Defending Against Universal Perturbations With Shared Adversarial Training

Chaithanya Kumar Mummadi
University of Freiburg
Bosch Center for Artificial Intelligence, Germany
ChaithanyaKumar.Mummadi@de.bosch.com

Thomas Brox
University of Freiburg
brox@cs.uni-freiburg.de

Jan Hendrik Metzen
Bosch Center for Artificial Intelligence, Germany
janhendrik.metzen@de.bosch.com

Abstract

Classifiers such as deep neural networks have been shown to be vulnerable against adversarial perturbations on problems with high-dimensional input space. While adversarial training improves the robustness of image classifiers against such adversarial perturbations, it leaves them sensitive to perturbations on a non-negligible fraction of the inputs. In this work, we show that adversarial training is more effective in preventing universal perturbations, where the same perturbation needs to fool a classifier on many inputs. Moreover, we investigate the trade-off between robustness against universal perturbations and performance on unperturbed data and propose an extension of adversarial training that handles this trade-off more gracefully. We present results for image classification and semantic segmentation to showcase that universal perturbations that fool a model hardened with adversarial training become clearly perceptible and show patterns of the target scene.

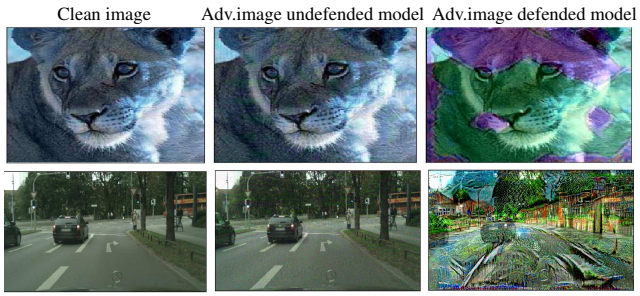

Figure 1. *Effectiveness of shared adversarial training against universal perturbations:* the top row shows an ImageNet example and the bottom row an example from Cityscapes. Adversarial images perturbed by universal perturbations generated for both the undefended models and models defended by our proposed method *shared adversarial training* are shown. The classification accuracy of the defended models deteriorates no more than 5% but robustness to universal adversarial attacks increases by 3x and 5x on image classification and semantic segmentation, respectively. Moreover, universal perturbations become clearly perceptible.

1. Introduction

While deep learning is relatively robust to random noise [11], it can be easily fooled by *adversarial perturbations* [44]. These perturbations are generated by adversarial attacks [15, 31, 5] that generate perturbed versions of the input which are misclassified by a classifier and remain quasi-imperceptible for humans. There have been different approaches for explaining properties of adversarial examples and provide rationale for their existence in the first place [15, 45, 12, 13]. Moreover, these perturbations have been shown to be relatively robust against various kinds of image transformations and are even successful when placed as artifacts in the physical world [21, 43, 10, 4]. Thus, adversarial perturbations might pose a safety and security risk for autonomous systems and also reduce trust on the models that are in principle vulnerable to these perturbations.

Several methods have been proposed for increasing the *robustness* of deep networks against adversarial examples, such as adversarial training [15, 22], virtual adversarial training [28], ensemble adversarial training [46], defensive distillation [36, 35], stability training [50], robust optimization [25], Parseval networks [7] and alternatively detecting and rejecting them as malicious [26]. While some of these approaches improve robustness against adversarial examples to some extent, the classifier remains vulnerable against adversarial perturbations on a non-negligible fraction of the inputs for all defenses [3, 47].

Most work has focused on increasing robustness in image classification tasks, where the adversary can choose a

data-dependent perturbation for each input. This setting is very much in favor of the adversary since the adversary can craft a high-dimensional perturbation "just" to fool a model on a single input. In this work, we argue that limited success in increasing the robustness under these conditions does not necessarily imply that robustness can not be achieved in other settings. Specifically, we focus on robustness against input-agnostic perturbations, namely *universal perturbations* [29], where the same perturbation needs to fool a classifier on many inputs. Moreover, we investigate robustness against such perturbations in dense prediction tasks such as *semantic image segmentation*, where a perturbation needs to fool a model on many decisions, e.g., the pixel-wise classifications. Data-dependent adversarial attacks need to know their input in advance and require online computation to generate perturbations for every incoming input whereas universal attacks work on unseen inputs.

Prior work has shown that standard models are vulnerable to both universal perturbations, which mislead a classifier on the majority of the inputs [29, 32], and to adversarial perturbations on semantic segmentation tasks [14, 48, 6]. The study of robustness against universal perturbations is important since they pose a realistic threat-model for certain physical-world attacks: for instance, Li et al. [23] show that an adversary could mount a semi-transparent adversarial sticker on a physical camera which effectively adds a universal perturbation to each unseen camera image. It was demonstrated by Metzen et al. [27] that such universal perturbations can hide nearby pedestrians in semantic segmentation which may allow deceiving an emergency braking system and would also pose a threat in surveillance scenarios. However, these and similar results have been achieved for undefended models. In this work, we focus on the case where models have been "hardened" by a defense mechanism, particularly adversarial training. While this technique can considerably increase robustness, there is an implicit trade-off between robustness against perturbations and high performance on unperturbed inputs. We show that explicitly tailoring adversarial training for universal perturbations allows handling this trade-off more gracefully.

Our main contributions are as follows: (1) We propose *shared adversarial training*, an extension of adversarial training that handles the inherent trade-off between accuracy on clean examples and robustness against universal perturbations more gracefully. (2) We evaluate our method on CIFAR10, a subset of ImageNet (with 200 classes), and Cityscapes to demonstrate that universal perturbations for the defended models become clearly perceptible as shown in Figure 1. (3) We are the first to scale defenses based on adversarial training to semantic segmentation. (4) We demonstrate empirically on CIFAR10 that the proposed technique outperforms other defense mechanisms [30, 37] in terms of robustness against universal perturbations.

2. Related Work

In this section, we review related work on the study of universal perturbations and adversarial perturbations for semantic image segmentation.

2.1. Universal Perturbations

Different methods for generating universal perturbations exist: Moosavi-Dezfooli et al. [29] uses an extension of the DeepFool adversary [31] to generate perturbations that fool a classifier on a maximum number of inputs from a training set. Metzen et al. [27] proposed a similar extension of the basic iterative adversary [22] for generating universal perturbations for semantic image segmentation. In contrast to former works, Mopuri et al. [33] proposed Fast Feature Fool, a data-independent approach for generating universal perturbations. In follow-up work [32], they show similar fooling rates of data-independent approaches as have been achieved by Moosavi-Dezfooli et al. [29]. Khrulkov and Oseledets [19] show a connection between universal perturbations and singular vectors. In another line of work, Hayes and Danezis [16], Mopuri et al. [40], and Poursaeed et al. [38] proposed generative models that can be trained to generate a diverse set of (universal) perturbations.

An analysis of universal perturbation and their properties is provided by Moosavi-Dezfooli et al. [30]. They connect the robustness to universal perturbations with the geometry of the decision boundary and prove the existence of small universal perturbation provided the decision boundary is systematically positively curved. Jetley et al. [18] build upon this work and provide evidence that directions in which a classifier is vulnerable to universal perturbations coincide with directions important for correct prediction on unperturbed data. They follow that predictive power and adversarial vulnerability are closely intertwined.

Prior procedures on robustness against universal perturbations define a distribution over (approximately optimal) such perturbations for a model (either by precomputing and random sampling [29], by learning a generative model [16], or by collecting an increasing set of universal perturbations for model checkpoints during training [37]), fine-tune model parameters to become robust against this distribution of perturbations, and (optionally) iterate. These procedures increase robustness against universal perturbations slightly, however, not to a satisfying level. This is probably caused by the model overfitting to the fixed distribution of universal perturbations that do not change during the optimization process. However, re-computing universal perturbations in every mini-batch anew is prohibitively expensive. In this work, we propose a method that can be performed efficiently by computing shared perturbations on each mini-batch and using them in adversarial training, i.e., the shared perturbations are computed on-the-fly rather than precomputed as in prior work [29, 37]. Concurrent to our work,

Shafahi et al.[42] recently proposed "universal adversarial training" where updates of the neural network's parameters and the universal perturbation happen concurrently. This reduces the overhead of determining a universal perturbation anew for every mini-batch; however, it is unclear if such an incrementally updated universal perturbation can track the changes of the network's weights sufficiently.

Alternative defense approaches add additional components to the model: Ruan and Dai [41] proposed to identify and reject universal perturbations by adding shadow classifiers, while Akhtar et al. [1] proposed to prepend a subnetwork in front of the model that is used to compensate for the added universal perturbation by detecting and rectifying the perturbation. Both methods have the disadvantage that the model becomes large and thus inference more costly. More severely, it is assumed that the adversary is not aware of the defense mechanism and it is unclear if a more powerful adversary could not fool the defense mechanism.

2.2. Adversarial Perturbations for Semantic Image Segmentation

Methods for generating adversarial perturbations have been extended to structured and dense prediction tasks like semantic segmentation and object detection [14, 48, 6]. Metzen et al. [27] even showed the existence of universal perturbations which result in an arbitrary target segmentation of the scene which has nothing in common with the scene a human perceives. A comparison of the robustness of different network architectures has been conducted by Arnab et al. [2]: they found that residual connections and multiscale processing actually increase robustness of an architecture, while mean-field inference for Dense Conditional Random Fields only masks gradient but does not increase robustness itself. In contrast to their work, we focus on modifying the training procedure for increasing robustness. Both approaches could be combined in the future.

3. Preliminaries

In this section, we introduce basic terms and notations relevant for this work. We aim to defend against an adversary under *white-box* attack settings. Please refer to Section A.1 in the supplementary material for details on capabilities of the adversary and the threat model.

3.1. Risks

Let L be a loss function (categorical crossentropy throughout this work), $\mathcal{D}$ be a data distribution, and θ be the parameters of a parametric model f_θ. Here, we define the *risk* $\rho(\theta)$ as the expected loss of the model f_θ for a data distribution. The following risks are relevant for this work (we extend the definitions of Uesato et al. [47]):

1. Expected Risk: $\rho_{exp}(\theta) = \mathbb{E}_{(x,y)\sim\mathcal{D}}\, L(\theta, x, y)$
2. Adversarial Risk: $\rho_{adv}(\theta, \mathcal{S}) = \mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\sup_{\xi(x)\in\mathcal{S}} L(\theta, x+\xi(x), y)\right]$
3. Universal Adversarial Risk: $\rho_{uni}(\theta, \mathcal{S}) = \sup_{\xi\in\mathcal{S}} \mathbb{E}_{(x,y)\sim\mathcal{D}}\left[L(\theta, x+\xi, y)\right]$

Here, $\xi(x)$ denotes an adversarial perturbation, ξ a universal perturbation, and $x + \xi(x)$ an adversarial example. The set $\mathcal{S}$ defines the space from which perturbations may be chosen. We would like to note that adversarial and universal risk are not equivalent since in the former case, $\xi(x)$ depends on the specific x sampled from $\mathcal{D}$, while the latter, ξ needs to generalize over the entire data distribution $\mathcal{D}$.

3.2. Adversaries

Since the worst-case perturbation $\xi(x)$ cannot be computed efficiently in typical settings, one needs to resort to an *adversary* which aims at finding a strong perturbation $\xi(x)$. Note that this corresponds to searching for a tight lower bound of ρ_{adv}. We define an adversary as a function $f_{adv} : \mathcal{D} \times \Theta \mapsto \mathcal{S}$, which maps a data point and model parameters θ onto a perturbation $\xi(x)$ that maximizes a loss $L_{adv}(\theta, x+\xi(x), y)$[1]. While different options for the adversaries f_{adv} exist [15, 31, 5, 29, 32], we focus on projected gradient descent (PGD) [25, 21], as it provides in our experience a good trade-off between being computationally efficient and powerful. PGD initializes $\xi^{(0)}$ uniformly at random in $\mathcal{S}$ (or subset of $\mathcal{S}$) and performs K iterations of the following update:

$\xi^{(k+1)} = \Pi_{\mathcal{S}}\left[\xi^{(k)} + \alpha_k \cdot \mathrm{sgn}(\nabla_x L_{adv}(\theta, x+\xi^{(k)}, y)\right]$, where $\Pi_{\mathcal{S}}$ denotes a projection on the space $\mathcal{S}$ and α_k denotes a step-size. Similarly, a targeted attack where the model shall output the target class y_t can be obtained by setting α_k to $-\alpha_k$ and y to y_t.

Similar to a standard adversary, we define a *universal adversary* denoted by f_{uni} as function mapping model parameters θ onto perturbation ξ with the objective of maximizing $\mathbb{E}_{(x,y)\sim\mathcal{D}}\left[L_{adv}(\theta, x+\xi, y)\right]$. One can modify PGD into a universal adversary by using the loss $L_{uni}(\theta, \{x_i, y_i\}_{i=1}^m, \xi) = \frac{1}{m}\sum_{i=1}^m L_{adv}(\theta, x_i+\xi, y_i)$. If the number of data points m is large (which is typically required for finding universal perturbations that generalize well to unseen data), one can employ stochastic PGD, where in every iteration k, a set of $\tilde{m}_k$ data points is sampled and L_{uni} is only evaluated on this subset of data points.

4. Shared Adversarial Training

We connect the above risks to show that adversarial training optimizes a loose upper bound on the universal risk and

[1] We note that one may choose $L_{adv} = L$ or one may also choose, e.g., L to be the 0-1 loss and L_{adv} be a differentiable surrogate loss.

motivate *shared adversarial training*, an extension of adversarial training that aims at maximizing robustness against universal perturbations. We show that this method minimizes an upper bound on the universal risk which is tighter than the one used in adversarial training.

4.1. Relationship between Risks

We show the following inequalities for the risks:$\rho_{exp}(\theta) \leq \rho_{uni}(\theta, \mathcal{S}) \leq \rho_{adv}(\theta, \mathcal{S})\ \forall\theta\ \forall\mathcal{S} \supset \{\mathbf{0}\}$. To see the validity of these inequalities, we set $\mathcal{S} = \{\mathbf{0}\}$ to obtain $\rho_{uni}(\theta, \mathcal{S}) = \rho_{exp}(\theta)$ (and $\mathcal{S} \supset \{\mathbf{0}\}$ can only increase $\rho_{uni}(\theta, \mathcal{S})$). For the second inequality, assume $\rho_{adv} < \rho_{uni}$. Let ξ be one of the multiple universal perturbations that maximize ρ_{uni}. Since ξ is an element of $\mathcal{S}$, we could certainly set $\xi(x) = \xi\ \forall x$ in the definition of the adversarial risk. This would result in $\rho_{adv} = \rho_{uni}$. This completes the proof by contradiction, and thus ρ_{adv} can only be larger than or equal to ρ_{uni} in general.

The objective of *adversarial training* is defined as minimizing the loss function $\sigma \cdot \rho_{adv}(\theta, \mathcal{S}) + (1 - \sigma) \cdot \rho_{exp}(\theta)$, where σ controls the trade-off between robustness and performance on unperturbed inputs. We note that if one is interested in minimizing the universal adversarial risk ρ_{uni}, then using ρ_{adv} in adversarial training with $\sigma = 1$ corresponds to minimizing an upper bound of ρ_{uni} because $\rho_{uni}(\theta, \mathcal{S}) \leq \rho_{adv}(\theta, \mathcal{S})$, provided that the adversaries find perturbations that are sufficiently close to the optimal perturbations. On the other hand, standard empirical risk minimization ERM ($\sigma = 0$), which minimizes the empirical estimate of ρ_{exp}, corresponds to minimizing a lower bound. As shown in previous work [15, 31, 5], this does confer only little robustness against (universal) perturbations. For $0 < \sigma < 1$, adversarial training corresponds to minimizing a convex combination of the upper bound ρ_{adv} and the lower bound ρ_{exp} but does not directly optimize on ρ_{uni}. As we show in Section 6, this standard version of adversarial training already provides strong robustness against universal perturbations at the cost of reducing performance on unperturbed data considerably.

4.2. Method

Directly employing ρ_{uni} in adversarial training is infeasible since evaluating $\rho_{uni}(\theta, \mathcal{S})$ with an adversary f_{uni} in every mini-batch is prohibitively expensive (because it requires large m). Hence, it would be desirable to use an upper bound of ρ_{uni} in adversarial training that is tighter than ρ_{adv} but cheaper to approximate than ρ_{uni}.

For this, we propose to use a so-called *heap adversary*, which we define as a function $f_{heap} : \mathcal{D}^m \times \Theta \mapsto \mathcal{S}$ that maps a set of m data points and model parameters θ onto a perturbation ξ. We use $L_{uni}(\theta, \{x_i, y_i\}_{i=1}^m, \xi) = \frac{1}{m}\sum_{i=1}^m L_{adv}(\theta, x_i + \xi, y_i)$ as loss function for the heap adversary. However, in contrast to a universal adversary, we do not require a heap adversary to find perturbations that generalize to unseen data. This allows choosing m relatively small.

More specifically, we split a mini-batch consisting of d data points into d/s heaps (subsets of the mini-batch) of size s (we denote s as *sharedness*). Rather than using the adversary f_{adv} for computing a perturbation on each of the d data points separately, we employ a heap adversary f_{heap} for computing d/s *shared* perturbations on the heaps with $m = s$. Thereupon, these perturbations are broadcasted to all d data points by repeating each of the shared perturbations s times for all elements of the heap. Employing this heap adversary implies a risk $\rho_{heap}^{(s)}$. We propose to use $\rho_{heap}^{(s)}$ in adversarial training when aiming at defending against universal perturbations and denote the resulting procedure as *shared adversarial training*. This entire process is illustrated in Figure 2. We can obtain the following relationship for $s = 2^i$ (please refer to Section A.2 for more details):

$$\rho_{adv} = \rho_{heap}^{(1)} \geq \rho_{heap}^{(2)} \geq \rho_{heap}^{(4)} \geq \cdots \geq \rho_{heap}^{(d)} \geq \rho_{uni}(\sigma, \mathcal{S})$$

Note that while all $\rho_{heap}^{(s)}$ are upper bounds on the universal risk ρ_{uni}, this does not imply that shared perturbations are strong universal perturbations. In contrast, the smaller s, the more "overfit" are the shared perturbations to the respective heap. However, $\rho_{heap}^{(s)}$ with $s \gg 1$ is typically a much tighter upper bound on ρ_{uni} than ρ_{adv} and can be approximated as efficiently as ρ_{adv}: for this, PGD is converted into a heap adversary by replacing L_{adv} with L_{uni}. By appropriately reshaping and broadcasting perturbations, we can compute d/s shared perturbations on the respective heaps of the mini-batch jointly by PGD with essentially the same cost as computing d adversarial perturbations with PGD.

4.3. Adversarial Loss Function

We recall that $L_{uni}(\theta, \{x_i, y_i\}_{i=1}^m, \xi) = \frac{1}{m}\sum_{i=1}^m L_{adv}(\theta, x_i + \xi, y_i)$. Because of limited capacity of the perturbation ($\xi \in \mathcal{S}$), there is "competition" between m data points: the maximizers of $L_{adv}(\theta, x_i + \xi, y_i)$ will typically be different and the data points will "pull" ξ into different directions. Hence, using the categorical cross-entropy as a proxy for the 0-1 loss is problematic for untargeted adversaries: since we are maximizing the loss and the categorical cross-entropy has no upper bound, there is a winner-takes-all tendency where the perturbation is chosen such that it leads to highly confident misclassifications on some data points and to correct classification on other data points (this incurs higher cost than misclassifying more data points but with lower confidence).

To prevent this, we employ loss thresholding on the categorical cross-entropy L to enforce an upper bound on L_{adv}: $L_{adv}(\theta, x, y) = \min(L(\theta, x, y), \kappa)$. We used $\kappa =$

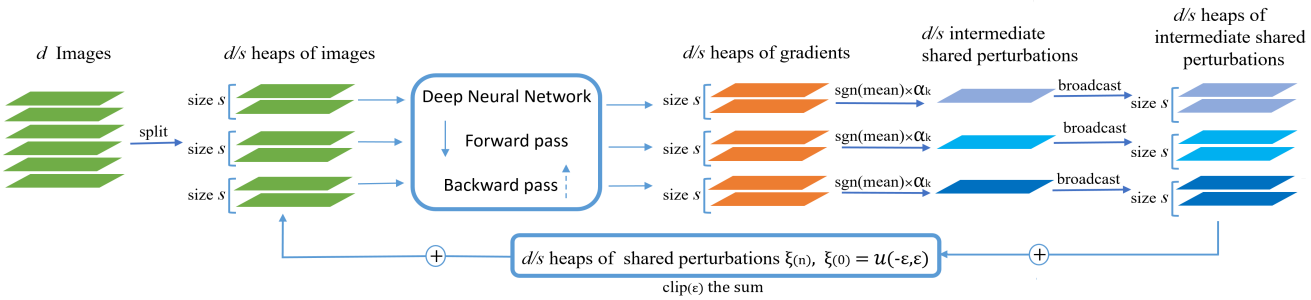

Figure 2. A pictorial representation of *shared adversarial training*. We split the mini-batch of d images into d/s heaps each with sharedness s and obtain the gradients of the loss with respect to the inputs. Here, the sharedness s corresponds to the number of inputs that are used for the generation of a shared perturbation. The gradients in each heap of size s are then processed and multiplied with step-size α_k to create a shared perturbation that is further broadcasted to size of the heap. The generated shared perturbations are aggregated and clipped after every iteration in order to confine the perturbations within a predefined magnitude ε. These perturbations are added to the images and this process is repeated iteratively. The adversarial inputs generated from the shared perturbations are used for adversarial training.

$-\log 0.2$, which corresponds to not encouraging the adversary to reduce confidence of the correct class below 0.2. A similar loss thresholding was also proposed by Shafahi et al. [42] concurrently. Besides, we also incorporate label smoothing and use the soft targets for the computation of loss in all our experiments.

5. Robustness Evaluation

In this section, we define the measure of robustness used in the experiments and detail how we approximate it.

5.1. Definition of Robustness

For the special case of the 0-1 loss, an n-dimensional input x, and $\mathcal{S} = \mathcal{S}(\varepsilon) = [-\varepsilon, \varepsilon]^n$, we define the *adversarial robustness* as the smallest perturbation magnitude ε that results in an adversarial risk (misclassification rate) of at least δ. More formally:

$$\varepsilon_{adv}(\delta) = \arg\min_{\varepsilon} \rho_{adv}(\theta, \mathcal{S}(\epsilon)) \text{ s.t. } \rho_{adv}(\theta, \mathcal{S}(\epsilon)) > \delta.$$

In other words, there are perturbations $\xi(x)$ with $||\xi(x)||_\infty < \varepsilon_{adv}(\delta)$ that result in a misclassification rate of at least δ. Analogously, we can also define the *universal robustness* as

$$\varepsilon_{uni}(\delta) = \arg\min_{\varepsilon} \rho_{uni}(\theta, \mathcal{S}(\epsilon)) \text{ s.t. } \rho_{uni}(\theta, \mathcal{S}(\epsilon)) > \delta.$$

Here, a perturbation ξ with $||\xi||_\infty < \varepsilon_{uni}(\delta)$ exists that results in a misclassification rate of at least δ.

5.2. Quantifying Robustness

Since the exact evaluation of $\varepsilon_{uni}(\delta)$ is intractable for our settings, we use an upper bound on the actual robustness $\varepsilon_{uni}(\delta)$ instead. For this, we tuned the PGD adversary as follows to make it more powerful (and thus the upper bound more tight): we performed a binary search of b iterations for perturbation magnitude ε of $\mathcal{S}(\varepsilon)$, i.e., the bound in the l_∞ norm on the perturbation, on the interval $\varepsilon \in [0, 255]$. In every iteration, we used the step-size annealing schedule $\alpha_k = \frac{\beta\varepsilon\gamma^k}{\sum_{j=0}^{K-1}\gamma^j}$ which guarantees that $\sum_{j=0}^{K-1}\alpha_k = \beta\varepsilon$. If a perturbation with misclassification rate δ is found in an iteration, the next iteration of binary search continues on the lower half of the interval for ε, otherwise on the upper half. The reported robustness is the smallest perturbation found in entire procedure that achieves a misclassification rate of δ. Note that this procedure was only used for evaluation; for training we used a predefined ε and constant step-size α_k.

6. Experimental Results

We present experimental results of *shared adversarial training* on robustness against universal perturbations in both image classification and semantic segmentation tasks. We extended the PGD implementation of Cleverhans [34] such that it supports shared adversarial perturbations and loss clipping as discussed in Section 4. For quantifying robustness, we extended Foolbox [39] such that universal perturbations (with minimal l_∞ norm) that achieve a misclassification rate of at least δ can be searched.

6.1. Experiments on CIFAR10

We present results on CIFAR10 [20] for ResNet20 [17] with 64-128-256 feature maps per stage. For evaluating robustness, we generate f_{uni} using stochastic PGD on 5000 validation samples with mini-batches of size $\tilde{m}_k = 16$ and evaluated on 512 test samples. We used $b = 10$ binary search iterations, $K = 200$ S-PGD iterations, and the step-size schedule values $\gamma = 0.975$ and $\beta = 4$. We pretrained ResNet20 with standard regularized empirical risk minimization (ERM) and obtained an accuracy of 93.25% on clean data and a robustness against universal perturbations of $\varepsilon_{uni}(\delta = 0.75) = 14.9$.

In general, we are interested in models that increase the robustness without decreasing the accuracy on clean

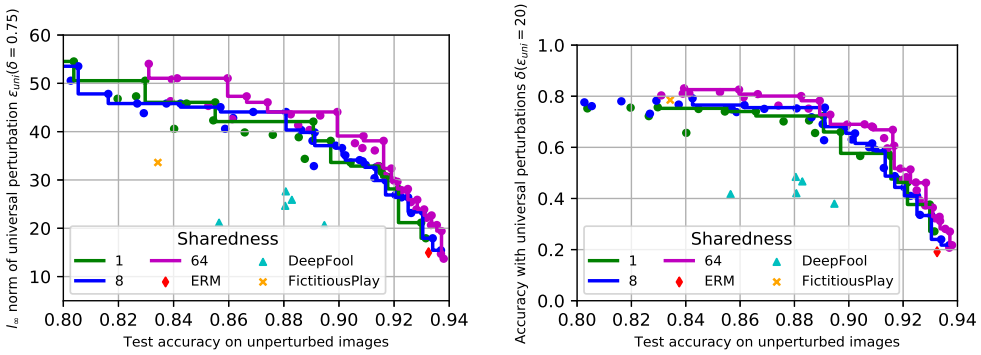

Figure 3. Pareto front on CIFAR10 for sharedness values $s \in \{1, 8, 64\}$. ERM corresponds to the model pretrained with empirical risk minimization, "DeepFool UAD" [29] to models trained with the procedure proposed by Moosavi-Dezfooli et al. [29], and "FictitiousPlay" to the procedure proposed by Perolat et al. [37]. (Left) Robustness with regard to S-PGD universal perturbations. (Right) Robustness with regard to DeepFool-based universal perturbations [29]. The Pareto front of the proposed defense is clearly above all previous defenses.

data considerably. We consider this as a multi-objective problems with two objectives (accuracy and robustness). In order to approximate the Pareto-front of different variants of adversarial and shared adversarial training (sharedness $s \in \{1, 8, 64\}$), we conducted runs for a range of attack parameters: maximum perturbation strength $\varepsilon \in \{2, 4, 6, 8, 10, 14, 18, 22, 26\}$ and $\sigma \in \{0.3, 0.5, 0.7, 0.9\}$ (controlling the trade-off between expected and adversarial risk). Model fine-tuning was performed with 65 epochs of SGD with batch-size 128, momentum 0.9, initial learning rate of 0.0025 and also performed 4 steps of PGD with step-size $\alpha_k = 0.5\varepsilon$ for each mini-batch. Here, the learning rate was annealed after 50 epochs by a factor of 10.

Figure 3 (left) shows the resulting Pareto fronts of different sharedness values (entries are provided in Table A1 in supplementary material). While sharedness $s = 1$ (standard adversarial training) and $s = 8$ perform similarly, $s = 64$ strictly dominates the other two settings. Without any loss on accuracy, a robustness of $\varepsilon_{uni}(\delta = 0.75) = 22.7$ can be achieved, and if one accepts an accuracy of 90%, a robustness of $\varepsilon_{uni}(\delta = 0.75) = 44.1$ is obtainable. This corresponds to nearly three times the robustness of the undefended model while accuracy only drops by less than 3.5%. We would also like to note that standard adversarial training is surprisingly effective in defending against universal perturbations and achieves a robustness that is smaller by approximately 5 than $s = 64$ at the same level of accuracy on unperturbed data. These findings suggest that increasing sharedness results in increased robustness. We found in preliminary experiments that this effect is strong for small s but has diminishing returns for sharedness beyond $s = 64$.

We also evaluated the defenses against universal perturbations proposed by Moosavi-Dezfooli et al. [29] and Perolat et al. [37] (please refer to Section A.3 in the supplementary material for details). It can be seen in Figure 3 (left) that these defenses are strictly dominated by all variants of (shared) adversarial training. In terms of computation, shared adversarial training required 189s (the same compute time as required by standard adversarial training) while the defense [29] required 3118s, and the defense [37] required 3840s per epoch on average. The proposed method thus outperforms the baseline defenses both in terms of computation and with regard to the robustness-accuracy trade-off.

Figure 3 (right) shows the Pareto front of the same models when attacked by the DeepFool-based method for generating universal perturbations [29]. In this case, the robustness is computed for a fixed perturbation magnitude $\varepsilon_{uni} = 20$ and the accuracy δ under this perturbation is reported. The qualitative results are the same as for an S-PGD attack: the Pareto-front of adversarial training (s=1) clearly dominates the results achieved by the defense proposed in [29]. Moreover, shared adversarial training with s=64 dominates standard adversarial training and the defense proposed by Perolat et al. [37]. This indicates that the increased robustness by shared adversarial training is not specific to the way the attacker generates universal perturbations. An illustration of the universal perturbations on this dataset is given in Section A.4 in the supplementary material.

6.2. Experiment on a Subset of ImageNet

We extend our experiments to a subset of ImageNet [9], which has more classes and higher resolution inputs than CIFAR10. Please refer to Section A.5 in the supplementary material for details on the selection of this subset. Similar to CIFAR10, we evaluate the robustness using stochastic PGD but generate perturbations on the training set with mini-batches of size $\tilde{m}_k = 10,000$ and evaluate on the total validation set. We used $b = 10$ binary search iterations, $K = 20$ S-PGD iterations, and the step-size schedule values $\gamma = 0.975$ and $\beta = 4$. We pre-trained wide resid-

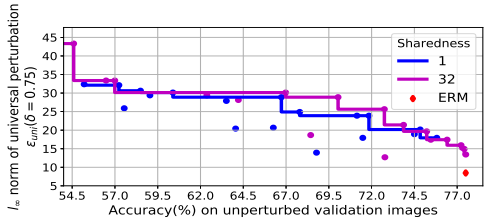

Figure 4. Pareto front on ImageNet for sharedness $s \in \{1, 32\}$. Shared adversarial training has doubled the robustness at the point of accuracy similar to baseline. With a slight loss of accuracy between 5% to 7%, the method increases the robustness by a factor of 3 and clearly dominates the standard adversarial training in terms of the robustness/accuracy trade-off.

ual network WRN-50-2-bottleneck [49] on this dataset with ERM using SGD for 100 epochs along with initial learning rate 0.1 and reduced it by a factor of 10 after every 30 epochs. We have obtained a top-1 accuracy of 77.57% on unperturbed validation data and a robustness against universal perturbations of $\varepsilon_{uni}(\delta = 0.75) = 8.4$.

We approximate the Pareto front of adversarial and shared adversarial training with sharedness $s \in \{1, 32\}$ and different $\varepsilon \in \{2, 4, 6, 8, 10, 14, 18, 22, 26\}$ and $\sigma \in \{0.5, 1.0\}$. We performed 5 steps of PGD with step-size $\alpha_k = 0.4\varepsilon$. The model was fine-tuned for 30 epochs of SGD with batch-size 128, momentum term 0.9, weight decay $5e^{-5}$, an initial learning rate of 0.01 that was reduced by a factor of 10 after 20 epochs and also performed 5 steps of PGD with step-size $\alpha_k = 0.4\varepsilon$ for each mini-batch.

Figure 4 compares the Pareto front of shared adversarial training with $s = 32$ and standard adversarial training $s = 1$ (entries are provided in Table A2 in the supplementary material). It can be clearly seen that shared adversarial training increases the robustness from $\varepsilon_{uni}(\delta = 0.75) = 8.4$ to 15.0 without any loss of accuracy. Moreover, shared adversarial training also dominates standard adversarial training for a target accuracy between 67%-74%, which corresponds to the sweet spot as a small loss in accuracy allows a large increase in robustness. The point with accuracy 72.74% and robustness $\varepsilon_{uni}(\delta = 0.75) = 25.64$ (obtained at $s = 32, \varepsilon = 10, \sigma = 1.0$) can be considered a good trade-off as accuracy drops by only 5% while robustness increases by a factor of 3, which results in clearly perceptible perturbations as shown in the top row of Figure 1 and Section A.6. Moreover, (shared) adversarial training also increases the entropy of the predicted class distribution for successful untargeted perturbations substantially (see Section A.7).

6.3. Semantic Image Segmentation

The results from above experiments show that shared adversarial training improves robustness against universal perturbations on image classification tasks where the adversary aims to fool the classifier's single decision on an input. In this section, we investigate our method against adversaries in a dense prediction task (semantic image segmentation), where the adversary aims at fooling the classifier on many decisions. To our knowledge, this is the first work to scale defenses based on adversarial training to this task.

We evaluate the proposed method on the Cityscapes dataset [8]. For computational reasons, all images and labels were downsampled from a resolution of 2048×1024 to 1024×512 pixels, where for images a bilinear interpolation and for labels a nearest-neighbor approach was used for downsampling. We pretrained the FCN-8 network architecture [24] on the whole training set of 2975 images and achieved 49.3% class-wise intersection-over-union (IoU) on the validation set of 500 images. Note that this IoU is relatively low because of downsampling the images.

We follow the experimental setup of Metzen et al. [27] which performed a targeted attack with a fixed target scene (monchengladbach_000000_026602_gtFine). They demonstrated that the desired target segmentation can be achieved despite the fact that the original scene has nothing in common with the target scene. We use the same target scene and consider this targeted attack successful if the average pixel-wise accuracy between the prediction on the perturbed images and the target segmentation exceeds $\delta = 0.95$. For evaluating robustness, we generate f_{uni} using stochastic PGD with mini-batches from the validation set of size $\tilde{m}_k = 5$ and tested on 16 samples from test set. We used $b = 10$ binary search iterations, $K = 200$ S-PGD iterations, the step-size schedule values $\gamma = 0.99$ and $\beta = 2$, and did not employ loss thresholding for targeted attacks. We find a universal perturbation that upper bounds the robustness of the model to $\varepsilon_{uni}(\delta = 0.95) \leq 19.92$.

We fine-tuned this model with adversarial and shared adversarial training. Since approximating the entire Pareto front of both methods would be computationally very expensive, we instead selected a target performance on unperturbed data of roughly 45% IoU (no more than 5% worse than the undefended model). The following two settings achieved this target performance (see Figure 5 left): adversarial training with $\varepsilon = 8$ and $\sigma = 0.5$ and shared adversarial training for sharedness $s = 5$, $\varepsilon = 30$, and $\sigma = 0.7$. The finetuning was performed for 20 epochs using Adam with batch-size 5 and a learning rate of 0.0001 that was annealed after 15 epochs to 0.00001. As heap adversary, we performed 5 steps of untargeted PGD with step-size $\alpha_k = 0.4\varepsilon$.

While both methods achieved very similar performance on unperturbed data, they show very different robustness against adversarial and universal perturbations (see Figure 5): standard adversarial training largely increases robustness against adversarial perturbations to $\varepsilon_{adv}(\delta = 0.95) \leq 11$, an increase by a factor of 4 compared to the undefended model. Shared adversarial training is less effec-

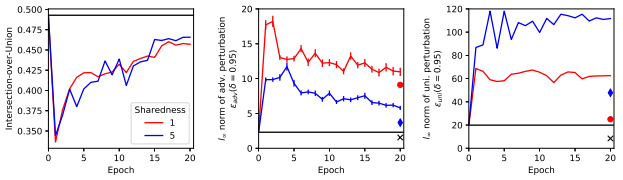

Figure 5. Learning curves on Cityscapes for adversarial (red, circle) and shared adversarial training (blue, diamond) with regard to performance on unperturbed images (left), and robustness against adversarial perturbations (middle, showing mean and standard error of mean) and universal perturbations (right). Black horizontal lines denote performance of undefended model. Isolated markers correspond to robustness against untargeted attacks. Performance of both standard and shared adversarial training are comparable on unperturbed data, but standard adversarial training dominates in terms of robustness against image-dependent adversarial perturbations, while shared adversarial training dominates in terms of robustness against targeted and untargeted universal perturbations.

tive against adversarial perturbations, its robustness is upper bounded by $\varepsilon_{adv}(\delta = 0.95) \leq 5.9$. However, shared adversarial training is more effective against targeted universal perturbations with an upper bound on robustness of $\varepsilon_{uni}(\delta = 0.95) \leq 111.7$, while adversarial training reaches $\varepsilon_{uni}(\delta = 0.95) \leq 62.5$. We also evaluated robustness against untargeted attacks: robustness increased from $\varepsilon_{uni}(\delta = 0.95) \leq 8.5$ of the undefended model to 25 and 47.8 for the models trained with standard and shared adversarial training respectively. The universal perturbation for the model trained with shared adversarial training clearly shows patterns of the target scene and dominates the original image, which is also depicted in the bottom row of Figure 1. We refer to Section A.8 in the supplementary material for illustrations of targeted and untargeted universal perturbations for different models.

6.4. Discussion

Results shown in Figure 5 indicate that there may exist a trade-off between robustness against image-dependent adversarial perturbations and universal perturbations. Figure 6 illustrates why these two kinds of robustness are not strictly related: adversarial perturbations fool a classifier by both adding structure from the target scene/class[2] to the image (e.g., vegetation on the middle left part of the image) and by destroying properties of the original scene (e.g., edges of the topmost windowsills). The latter is not possible for universal perturbations since the input images are not known in advance. As also shown in the figure, universal perturbations compensate this by adding stronger patterns of the target scene. Shared perturbations will become more similar

[2]For untargeted attacks, the attacks may choose a target scene/class arbitrarily such that fooling the model becomes as simple as possible.

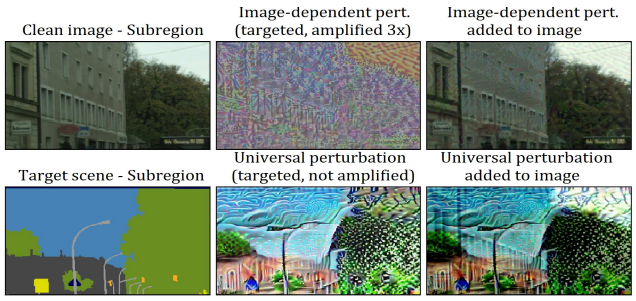

Figure 6. Illustration of image-dependent and universal perturbations for the same image and target scene (upper and lower left) that are generated on the model hardened with shared adversarial training. Image-dependent perturbations weaken patterns of existing structure like edges of the actual scene (upper right) whereas universal perturbations are restricted to adding structure indicative of the target scene (lower right). This qualitative difference between perturbations provides a possible explanation why shared adversarial training demonstrates different levels of robustness on image-dependent and universal perturbations: shared adversarial training improves robustnesss against additive structure but not against the perturbations that weaken the existing structure.

to universal perturbations with increasing sharedness since a single shared perturbation has fixed capacity and cannot destroy properties of arbitrarily many input images (even if they are known in advance). Accordingly, shared adversarial training will make the model mostly more robust against perturbations which add new structures and not against perturbations which destroy existing structure. Hence, it results in less robustness against image-specific perturbations (seen in Figure 5 middle). On the other hand, since shared adversarial training focuses on one specific kind of perturbations (those that add structure to the scene), it leads to models that are particularly robust against universal perturbations (shown in Figure 5 right).

7. Conclusion

We have shown that adversarial training is surprisingly effective in defending against universal perturbations. Since adversarial training does not explicitly optimize the trade-off between robustness against universal perturbations and performance on unperturbed data points, it handles this trade-off suboptimally. We have proposed *shared adversarial training*, which performs adversarial training on a tight upper bound of the universal adversarial risk. We have shown that our method allows achieving high robustness against universal perturbations on image classification tasks at smaller loss of accuracy. The proposed method also scales to semantic segmentation on high resolution images, where compared to adversarial training it achieves higher robustness against universal perturbations at the same level of performance on unperturbed images.

References

[1] Naveed Akhtar, Jian Liu, and Ajmal Mian. Defense against Universal Adversarial Perturbations. In *arXiv:1711.05929 [cs]*, Nov. 2017. arXiv: 1711.05929.

[2] Anurag Arnab, Ondrej Miksik, and Philip H. S. Torr. On the Robustness of Semantic Segmentation Models to Adversarial Attacks. In *arXiv:1711.09856 [cs]*, Nov. 2017. arXiv: 1711.09856.

[3] Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated Gradients Give a False Sense of Security: Circumventing Defenses to Adversarial Examples. In *arXiv:1802.00420 [cs]*, Feb. 2018.

[4] Anish Athalye and Ilya Sutskever. Synthesizing Robust Adversarial Examples. In *arXiv:1707.07397 [cs]*, July 2017.

[5] Nicholas Carlini and David Wagner. Towards Evaluating the Robustness of Neural Networks. In *IEEE Symposium on Security and Privacy (SP)*, May 2017.

[6] Moustapha Cisse, Yossi Adi, Natalia Neverova, and Joseph Keshet. Houdini: Fooling Deep Structured Prediction Models. In *Advances in Neural Information Processing Systems (NIPS) 30*, 2018.

[7] Moustapha Cisse, Piotr Bojanowski, Edouard Grave, Yann Dauphin, and Nicolas Usunier. Parseval Networks: Improving Robustness to Adversarial Examples. In *Proceedings of the 34th International Conference on Machine Learning*, Aug. 2017.

[8] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, Nevada, USA, 2016.

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. Ieee, 2009.

[10] Ivan Evtimov, Kevin Eykholt, Earlence Fernandes, Tadayoshi Kohno, Bo Li, Atul Prakash, Amir Rahmati, and Dawn Song. Robust Physical-World Attacks on Machine Learning Models. In *arXiv:1707.08945 [cs]*, July 2017.

[11] Alhussein Fawzi, Seyed-Mohsen Moosavi-Dezfooli, and Pascal Frossard. Robustness of classifiers: from adversarial to random noise. In *Advances in Neural Information Processing Systems (NIPS) 29*, pages 1632–1640, 2016.

[12] Alhussein Fawzi, Seyed-Mohsen Moosavi-Dezfooli, and Pascal Frossard. A Geometric Perspective on the Robustness of Deep Networks. In *IEEE Signal Processing Magazine*, 2017. accepted.

[13] Alhussein Fawzi, Seyed-Mohsen Moosavi-Dezfooli, Pascal Frossard, and Stefano Soatto. Classification regions of deep neural networks. In *arXiv:1705.09552 [cs, stat]*, May 2017.

[14] Volker Fischer, Chaithanya Kumar Mummadi, Jan Hendrik Metzen, and Thomas Brox. Adversarial Examples for Semantic Image Segmentation. In *Workshop of International Conference on Learning Representations (ICLR)*, Mar. 2017.

[15] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and Harnessing Adversarial Examples. In *International Conference on Learning Representations (ICLR)*, 2015.

[16] Jamie Hayes and George Danezis. Learning Universal Adversarial Perturbations with Generative Models. *arXiv:1708.05207 [cs, stat]*, Aug. 2017. arXiv: 1708.05207.

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *Computer Vision and Pattern Recognition (CVPR)*, 2016.

[18] Saumya Jetley, Nicholas A. Lord, and Philip H. S. Torr. With Friends Like These, Who Needs Adversaries? In *arXiv:1807.04200 [cs]*, July 2018. arXiv: 1807.04200.

[19] Valentin Khrulkov and Ivan Oseledets. Art of singular vectors and universal adversarial perturbations. In *arXiv:1709.03582 [cs]*, Sept. 2017.

[20] Alex Krizhevsky. Learning Multiple Layers of Features from Tiny Images. Master's thesis, University of Toronto, 2009.

[21] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *International Conference on Learning Representations (Workshop)*, Apr. 2017.

[22] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial Machine Learning at Scale. In *International Conference on Learning Representations (ICLR)*, 2017.

[23] Juncheng Li, Frank Schmidt, and Zico Kolter. Adversarial camera stickers: A physical camera-based attack on deep learning systems. In *International Conference on Machine Learning*, pages 3896–3904, 2019.

[24] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation. In *Proceedings of Computer Vision and Pattern Recognition (CVPR)*, Boston, 2015.

[25] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards Deep Learning Models Resistant to Adversarial Attacks. In *International Conference on Learning Representations (ICLR)*, 2018.

[26] Jan Hendrik Metzen, Tim Genewein, Volker Fischer, and Bastian Bischoff. On Detecting Adversarial Perturbations. In *International Conference on Learning Representations (ICLR)*, 2017.

[27] Jan Hendrik Metzen, Chaithanya Kumar Mummadi, Thomas Brox, and Volker Fischer. Universal Adversarial Perturbations Against Semantic Image Segmentation. *International Conference on Computer Vision (ICCV)*, Oct. 2017.

[28] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual Adversarial Training: a Regularization Method for Supervised and Semi-supervised Learning. *arXiv:1704.03976 [cs, stat]*, Apr. 2017.

[29] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. Universal adversarial perturbations. In *IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, Hawaii, USA, 2017.

[30] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, Pascal Frossard, and Stefano Soatto. Robustness of classifiers to universal perturbations: A geometric perspective. *International Conference on Learning Representations (ICLR)*, 2018.

[31] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. DeepFool: a simple and accurate method to fool deep neural networks. In *Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, Nevada, USA, 2016.

[32] Konda Reddy Mopuri, Aditya Ganeshan, and R. Venkatesh Babu. Generalizable Data-free Objective for Crafting Universal Adversarial Perturbations. *arXiv:1801.08092 [cs]*, Jan. 2018. arXiv: 1801.08092.

[33] Konda Reddy Mopuri, Utsav Garg, and R. Venkatesh Babu. Fast Feature Fool: A data independent approach to universal adversarial perturbations. In *arXiv:1707.05572 [cs]*, July 2017.

[34] Nicolas Papernot, Ian Goodfellow, Ryan Sheatsley, Reuben Feinman, and Patrick McDaniel. cleverhans v1.0.0: an adversarial machine learning library. *arXiv preprint arXiv:1610.00768*, 2016.

[35] Nicolas Papernot and Patrick McDaniel. Extending Defensive Distillation. *arXiv:1705.05264 [cs, stat]*, May 2017.

[36] Nicolas Papernot, Patrick McDaniel, Xi Wu, Somesh Jha, and Ananthram Swami. Distillation as a Defense to Adversarial Perturbations against Deep Neural Networks. In *Proceedings of the 37th IEEE Symposium on Security & Privacy*, pages 582–597, San Jose, CA, 2016.

[37] Julien Perolat, Mateusz Malinowski, Bilal Piot, and Olivier Pietquin. Playing the Game of Universal Adversarial Perturbations. *arXiv:1809.07802 [cs, stat]*, Sept. 2018. arXiv: 1809.07802.

[38] Omid Poursaeed, Isay Katsman, Bicheng Gao, and Serge Belongie. Generative adversarial perturbations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4422–4431, 2018.

[39] Jonas Rauber, Wieland Brendel, and Matthias Bethge. Foolbox v0.8.0: A python toolbox to benchmark the robustness of machine learning models. *arXiv preprint arXiv:1707.04131*, 2017.

[40] Konda Reddy Mopuri, Utkarsh Ojha, Utsav Garg, and R Venkatesh Babu. Nag: Network for adversary generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 742–751, 2018.

[41] Yibin Ruan and Jiazhu Dai. TwinNet: A Double Sub-Network Framework for Detecting Universal Adversarial Perturbations. *Future Internet*, 10(3), Mar. 2018.

[42] Ali Shafahi, Mahyar Najibi, Zheng Xu, John Dickerson, Larry S Davis, and Tom Goldstein. Universal adversarial training. *arXiv preprint arXiv:1811.11304*, 2018.

[43] Mahmood Sharif, Sruti Bhagavatula, Lujo Bauer, and Michael K. Reiter. Accessorize to a Crime: Real and Stealthy Attacks on State-of-the-Art Face Recognition. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, CCS '16, pages 1528–1540, New York, NY, USA, 2016. ACM.

[44] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations (ICLR)*, 2014.

[45] Thomas Tanay and Lewis Griffin. A Boundary Tilting Persepective on the Phenomenon of Adversarial Examples. *arXiv:1608.07690 [cs, stat]*, Aug. 2016.

[46] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Dan Boneh, and Patrick McDaniel. Ensemble Adversarial Training: Attacks and Defenses. In *International Conference on Learning Representations (ICLR)*, 2018.

[47] Jonathan Uesato, Brendan O'Donoghue, Aaron van den Oord, and Pushmeet Kohli. Adversarial Risk and the Dangers of Evaluating Against Weak Attacks. In *arXiv:1802.05666 [cs, stat]*, Feb. 2018.

[48] Cihang Xie, Jianyu Wang, Zhishuai Zhang, Yuyin Zhou, Lingxi Xie, and Alan Yuille. Adversarial Examples for Semantic Segmentation and Object Detection. In *International Conference on Computer Vision (ICCV)*, 2017.

[49] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.

[50] Stephan Zheng, Yang Song, Thomas Leung, and Ian Goodfellow. Improving the Robustness of Deep Neural Networks via Stability Training. In *Computer Vision and Pattern Recognition (CVPR)*, 2016.