Rethinking the Evaluation of Deep Neural Network Robustness

Mingyuan Fan[1], Fuyi Wang[2](✉), and Bosheng Yan[2]

[1] School of Data Science and Engineering, East China Normal University, Shanghai, China
[2] School of Information Technology, Deakin University, Geelong, Australia
wong_fuyi@outlook.com, yanbo@deakin.edu.au

Abstract. Evaluating the robustness of deep neural networks (DNNs) is crucial for ensuring the reliability and security of machine learning systems. Prior approaches quantify the probability of a DNN being compromised under a specified constraint. Despite their utility, these techniques suffer from low efficiency and effectiveness in evaluating the robustness of DNNs. The paper presents a promising evaluation approach, named typeII-EvaA, for accurately and efficiently assessing the robustness of DNNs against adversarial attacks. The typeII-EvaA overcomes the limitations of existing evaluation methods by devising several new assessment methods, called typeII-AssMs, which use attack success rate (ASR) constraints to minimize perturbation magnitudes. Additionally, we introduce a more effective human imperceptibility metric, CIEDE2000, which aligns with the human vision system to probe almost all human-imperceptible areas for obtaining the most threatening adversarial examples. Extensive experimental results corroborate that typeII-EvaA has practical implications for improving the security of DNN-based systems. And typeII-AssMs can achieve 100% ASR against various defense mechanisms. Our intention is for the typeII-EvaA to serve as a benchmark for future efforts toward developing robust DNNs that can withstand adversarial examples.

Keywords: Deep neural networks · Robustness evaluation · Adversarial attacks · Human imperceptibility metric

1 Introduction

Deep neural networks (DNNs) have garnered significant attention in recent years due to their superior performance and ability to address complex tasks across various domains. However, their vulnerability to attacks from adversaries has limited their deployment in security-critical applications [21]. Specifically, adversarial attacks exploit the susceptibility of DNNs by introducing **human-imperceptible** adversarial perturbations to natural examples, resulting in misclassifications from state-of-the-art (SOTA) DNNs [14,15,21]. As a result, there

M. Fan and F. Wang—The authors contribute equally to this work.

X. Yang et al. (Eds.): ADMA 2023, LNAI 14179, pp. 532–547, 2023.
https://doi.org/10.1007/978-3-031-46674-8_37

are significant incentives for researchers to explore the robustness of DNNs against adversarial attacks [1,7,9,18].

The task of exploring the robustness of DNNs against adversarial attacks primarily involves developing effective defense mechanisms and evaluation approaches that are analogous to the training and validation methods used for DNNs [2,12,14,17,21]. On the one hand, an ideal evaluation approach should be capable of accurately and efficiently assessing the ground-truth robustness of DNNs against adversarial attacks [2,14]. Evaluation methods that lack rigor are inadequate for evaluating the effectiveness of defenses and can yield misleading results, hampering progress in this field. On the other hand, evaluation methods that are overly resource-intensive are impractical for real-world use due to their high computational overhead.

This paper rethinks the effectiveness and efficiency of the existing evaluation approaches, named typeI-EvaAs. TypeI-EvaAs generally involve the following two steps: 1) generating threatening adversarial perturbations as much as possible via maximizing an attack effectiveness metric, under a certain distance constraint (known as the human perceptibility metric) of magnitude ϵ, where ϵ has to enable the resulting perturbations to be human-imperceptible; 2) adopting the perturbations to produce adversarial examples and estimating the corresponding probability of the DNNs being tricked, i.e., attack success rate (ASR). In brief, typeI-EvaAs report the probability of the DNNs being fooled under a given constraint, and we call these kinds of assessment methods typeI-AssMs. However, typeI-EvaAs are of low efficiency and poor effectiveness.

Low Efficiency. Pre-setting an appropriate constraint magnitude ϵ is necessary for typeI-EvaAs: a larger constraint magnitude allows a more broad search space to be navigated that can in general raise attack effectiveness, but also easily results in visually noticeable adversarial perturbations, i.e., viotibility; vice versa. To determine the proper magnitude, empirical observations are typically used. Evaluators observe and compare the crafted adversarial examples under various constraint magnitudes and then manually select an optimal magnitude. It is vastly cumbersome and computationally intensive, as crafting a single adversarial example may require backpropagation up to hundreds or thousands of times. Furthermore, many evaluators may not have the necessary expertise to efficiently tune ϵ, which can lead to additional overhead, particularly for large-scale datasets with modern ultra-huge DNNs.

Poor Effectiveness. TypeI-EvaAs commonly use norm-based constraints, specifically ∞-norm constraint, to resultant adversarial perturbations human-imperceptible. However, many more threatening and imperceptible adversarial perturbations are beyond the ∞-norm constraint. Specifically, Fig. 1(b,c) shows adversarial examples by common attacks with ∞-norm constraint of 16 and 32. As can be seen, some adversarial ones are considerably different from the original ones in vision and also cannot completely mislead models. In contrast, TypeII-EvaA crafts adversarial ones (Fig. 1(d)) which perturbation beyond 16 and 32 but are also significantly effective to models and quite similar to original ones. In fact, ∞-norm distance treats perturbations of different images, or even different

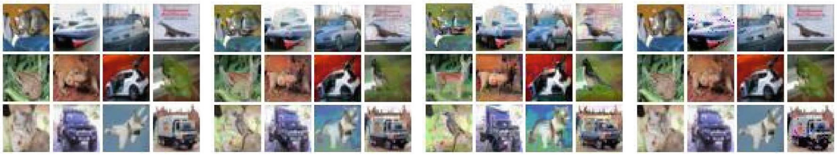

(a) Origin (b) PGD-16 (c) PGD-32 (d) Our

Fig. 1. We craft adversarial examples for images from the leftmost column. (b) and (c) are produced by PGD with perturbation budgets of 16 and 32. (d) is crafted by TypeII-EvaA (inner-joint-optim version).

pixels in an image, as equally important, and this is barely established for the human perceptual system. Therefore, typeI-EvaAs tend to overlook the area with low human-imperceptible distance and high ∞-norm distance from the original images, resulting in the failure to detect many threatening adversarial examples and leading to a false sense of robustness.

To address the two above-mentioned limitations, we revisit typeI-EvaAs which report ASR with a specified perturbation constraint (called typeI-AssMs) for the robustness of DNNs. And a natural idea is that can we assess the robustness of DNNs by estimating how much perturbations need to be imposed to reach a given ASR? To achieve a specific ASR, we identify the most vulnerable sample combination with minimal perturbations to induce full misclassification. However, finding the most vulnerable combination is challenging as enumerating all combinations equals an NP-hard problem. We simplify the problem into finding minimal threatening adversarial perturbations for each sample independently with linear time complexity and the overhead is considerably lower than typeI-AssMs that craft adversarial perturbations for each constraint magnitude over all samples. For a specific ASR, the attackers are at least required to add adversarial perturbations above the magnitude to reach the ASR. In a nutshell, our technical contributions are threefold.

- We develop a novel evaluation approach, dubbed typeII-EvaA, that can effectively and efficiently reap the accurate robustness estimations of DNNs. TypeII-EvaA is the *first* work to explore novel and remarkably efficient assessment methods called typeII-AssMs with a specific ASR. For the practicality of typeII-EvaA, we craft fresh attack paradigms that minimize perturbation magnitudes with ASR constraints.
- For the effectiveness of typeII-EvaA, we explore an effective human imperceptibility metric compared to *norm*-based metrics, i.e., CIEDE2000[1], that aligns well with the human vision system, such that the adversarial attacks with CIEDE2000 are allowed to probe almost all human-imperceptible areas for obtaining most threatening adversarial examples.

[1] CIEDE2000 is a perceptual color distance recently released by International Commission on Illumination.

- We design several proxy functions for finding adversarial examples. Additionally, we devise four search algorithms with various strategies in order to determine the perturbations of adversarial examples. We systematically evaluate these designs and show that typeII-EvaA can comprehensively evaluate the efficacy of defense mechanisms.

The rest of the paper is organized as follows. In the following section, we introduce the background, e.g., DNNs and adversarial examples. Section 3 formulates the challenge for typeII-AssMs. Section 4 develops the human imperceptibility and attack effectiveness metrics. We design search algorithms consisting of three ingredients: initialization strategy, search direction, and step size in Sect. 5 followed by experimental evaluation of typeII-EvaA in the large-scale CIFAR10 and ImageNet datasets in Sect. 6. Finally, the conclusion of this paper is made in Sect. 7.

2 Background

2.1 Deep Neural Networks

Deep neural networks (DNNs) are highly intricate mathematical models composed of numerous layers. Each layer is typically comprised of a linear function and an activation function. We denote a DNN with parameter θ by $F_\theta(\cdot) \in \mathbb{R}^m$ and $F_\theta(x)[i]$ $(i = 1, 2, \cdots, m)$ denotes the prediction confidence of the DNN for classifying x into i-th category, where m is the total number of categories. In order to obtain a probability distribution over all potential categories as the final prediction result, it is frequently customary to incorporate a softmax function to standardize the level of confidence in the prediction. Softmax function outputs the probability of i-th category as follows:

$$\text{softmax}(F_\theta(x))[i] = \frac{e^{F_\theta(x)[i]}}{\sum_{i=1}^{m} e^{F_\theta(x)[i]}}. \tag{1}$$

Given a dataset $D = \{(x_i, y_i), i = 1, 2, \cdots, n, x_i \in \mathbb{R}^{c\times h\times w}, y_i \in \mathbb{R}^m\}$ where c, h, w denote the channel, height, width of the input image, and the performance of DNNs is quantitated by accuracy as follows:

$$acc = \frac{\sum_{i=1}^{n} \mathbb{I}((\arg\max_{j=1,\cdots,m} F_\theta(x_i)[j]) = y_i)}{n}, \tag{2}$$

where $\mathbb{I}(\cdot)$ is a characteristic function that outputs 1 when the condition holds and otherwise outputs 0. Then, to make the optimal performance of the DNN over D, standard practice is leveraging the mini-batch gradient descent algorithm or its variants to optimize θ associated with accuracy in an end-to-end fashion. But the gradient-based optimization algorithms cannot be directly applied to optimize θ since accuracy blocks the gradient propagation process. Thus, accuracy generally is replaced with a differentiable objective function while the most frequently-used objective function is the cross-entropy function shortened by $CE(\cdot,\cdot)$.

$$CE(x, y) = -\log(\mathrm{softmax}(F_\theta(x))[y]). \tag{3}$$

The quality of objective functions has a huge influence on the resultant DNN and an inferior objective function probably leads to a worse θ.

2.2 Adversarial Examples

Adversarial examples are malicious inputs that are artificially synthesized with clean inputs and specific perturbations crafted by the attacker. With adversarial examples, the attacker can fool the target DNN model to output attacker-chosen (or random) predictions, as defined in Definition 1.

Definition 1 (Adversarial Attacks). *Given a DNN model $F_\theta(x)$ with parameter θ and an input x, the adversarial attack aims to find a specific perturbation δ for x that satisfies:*

$$\delta = \arg\min_{\delta} \mathcal{M}(x, x+\delta), \quad s.t., \arg\max_{j=1,\cdots,m} F_\theta(x+\delta)[j] \neq y, \tag{4}$$

where m is number of classes, the function $\mathcal{M}(\cdot,\cdot)$ evaluates the distance between x and $x+\delta$, which also reflects the human imperceptibility of the generated adversarial example [12].

Generally, most adversarial attacks follow the below paradigm to approximately resolve Eq. 4:

$$\delta = \arg\max_{\delta} L(F_\theta(x+\delta), y), \quad s.t., \mathcal{M}(x, x+\delta) \leq \epsilon, \tag{5}$$

where $L(\cdot,\cdot)$ is a proxy function of $\arg\max_{j=1,\cdots,m} F_\theta(x+\delta)[j] \neq y$ like $CE(\cdot,\cdot)$, and ϵ is the perturbation budget that constraints the distance between x and $x+\delta$. Commonly, $L(\cdot,\cdot)$ is positively correlated with the misclassification rate of the model and referred to as the attack effectiveness metric.

Definition 2 (Adversarial Perturbations and Examples). *Given an input x with the ground-truth label y and a target DNN, if perturbations δ are crafted by adversarial attacks, δ and $x+\delta$ are referred to as adversarial perturbations and adversarial examples. Furthermore, if the target DNN misidentifies $x+\delta$, the δ is threatening adversarial perturbations; otherwise, the δ is weak adversarial perturbations.*

If adversarial perturbations are described as minimal for input x, this implies that such perturbations result in the lowest possible value for $\mathcal{M}(x, x+\delta)$.

Different adversarial attacks can be reduced by solving Eq. 4. An intuitive idea of solving Eq. 4 is to impulse samples to move in the direction that makes the loss of the sample higher as possible, i.e., Eq. 5, and gradient directions can effectively match the direction. The fast gradient sign method (FGSM) harnesses the idea and approximately solves Eq. 5 by directly setting $\delta = \epsilon \ \nabla_x L(F_\theta(x), y)$, where $L(\cdot,\cdot)$ commonly is $CE(\cdot,\cdot)$ [21]. As suggested in its name, the main merit of FGSM is efficiently crafting adversarial examples due to backpropagation only being required to implement once. But, when a bigger tolerance for perturbations is allowed, FGSM performs poorly, as ϵ is seemingly too big and the gradient direction only works around the small neighborhood of x. Accordingly, the basic iterative method (BIM) [11] and projected gradient descent (PGD) [12] improve FGSM by using an iterative way with a small step size to solve Eq. 4. In detail, given total iterations T, BIM crafts $\delta = \delta_T$ by iteratively updating $\delta_t = Clip_\epsilon(\delta_{t-1} + \nabla_{x+\delta_{t-1}} L(F_\theta(x + \delta_{t-1}), y))$ $(t = 0, 1, 2, \cdots, T)$, where $Clip_\epsilon(\cdot)$ draws the perturbations back to the constraint domain, where the initial perturbations δ_0 are full-zero vectors. Due to the local extreme points in the vicinity of x, the PGD incorporates a randomized perturbation into the initial perturbation δ_0 to evade these local extreme points [12]. Apart from the above adversarial attack methods, another famous and effective adversarial attack is C&W attack [2]. Rather than optimizing perturbations subject to constraints, the C&W attack approach entails simultaneous optimization of both the loss function and perturbations, formulating various loss functions and choosing the optimal one experimentally to replace the traditional cross-entropy loss function [3].

3 Assessment Method

3.1 Problem Formulation

The objective of typeII-AssMs is to obtain the least adversarial perturbations on the dataset D to achieve the specified ASR p. This can be formulated as optimizing the following task to obtain adversarial perturbations $\delta_1, ..., \delta_n$:

$$\begin{aligned}
&\delta_1, \cdots, \delta_n = \underset{\delta_1,\ldots,\delta_n}{\arg\min} \sum_{i=1}^{n} I_i \cdot \mathcal{M}(x_i, x_i + \delta_i) \\
&s.t., \sum I_i(\underset{j=1,\cdots,m}{\arg\max} F_\theta(x_i)[j] \neq y_i) = n \cdot p, \\
&I_1 + \cdots I_n = n \cdot p \text{ and } I_i = 0 \text{ or } 1,
\end{aligned} \tag{6}$$

where $n \cdot p$ is assumed to be an integer and I_i is an indicator function that outputs 1 if the input condition establishes otherwise outputs 0. After obtaining the solution for Eq. 6, $d = \mathcal{M}(x_1, x_1 + \delta_1) + \cdots + \mathcal{M}(x_n, x_n + \delta_n)$ can be used to assess the robustness of DNNs against adversarial attacks.

3.2 Solution to Equation 6

Before developing the solution to Eq. 6, we consider a special case of it, where $p = 100\%$. If $p = 100\%$, there is $I_i = 1$ for $\forall i$ and we then search for the adversarial perturbations δ_i that cause the misclassification of x_i from the DNN

$F_\theta(\cdot)$ and minimize $\mathcal{M}(x_i, x_i + \delta_i)$. Since searching for adversarial perturbations δ_i for different x_i is independent, Eq. 6 can be simplified to solve the following optimization task for each x_i.

$$\begin{aligned} \delta_i &= \arg\min_{\delta_i} \mathcal{M}(x_i, x_i + \delta_i) \\ s.t., &\arg\max_{j=1,\cdots,m} F_\theta(x_i)[j] \neq y_i. \end{aligned} \tag{7}$$

Assuming that the adversarial perturbations δ_i for $\forall i$, are obtained by solving Eq. 7 with $p = 100\%$. The objective is to find the most vulnerable combination of $n \cdot p'$ instances after resetting p to a new value p'. To achieve this, one can greedily set $n \cdot (p - p')$ elements with the maximum $\mathcal{M}(x_i, x_i + \delta_i)$ in $\{\delta_1, \cdots, \delta_n\}$ to zero, which results in a minimal $\sum_i^n \mathcal{M}(x_i, x_i + \delta_i)$ among all combinations of size $n \cdot p'$ as the generation of each δ_i is independent. The resulting perturbations $\{\delta_1, \cdots, \delta_n\}$ are exactly the solution of Eq. 6 with p'. Furthermore, shrinking δ_i results in an increase in $\mathcal{M}(x_i, x_i + \delta_i)$, implying that the model $F_\theta(\cdot)$ will correctly identify x_i.

Time Complexity Comparison. Supposing that the time complexity of generating δ_i is $O(1)$. Then, the expected time complexity of solving Eq. 6 with our method is $O(n)$, whereas the expected time complexity of directly solving Eq. 6 is $O(C_n^k) = O(n!)$.

The Relationship to typeI-AssMs. We demonstrate that the results of typeI-AssMs can be readily obtained from the results of typeII-AssMs. Specifically, for typeI-AssMs, the objective is to determine the maximum achievable ASR under a given perturbation budget ϵ. By leveraging the fact that typeII-AssMs with ASR=100% generates the minimal magnitude of threatening adversarial perturbations for each instance x_i, imposing perturbations below this magnitude ensures that x_i is classified correctly. Therefore, the maximum achievable ASR can be computed as the ratio of samples for which the minimal magnitude of the threatening adversarial perturbations is smaller than ϵ. Consequently, we conclude that leveraging typeII-AssMs for evaluations is always preferable to typeI-AssMs, as the latter can be effortlessly derived from the former, but not vice versa. Furthermore, typeII-AssMs eliminate the significant burden of tuning the hyperparameter ϵ.

3.3 Solution to Equation 7

To simplify the notation, we omit the subscript i in Eq. 7. The problem we need to solve can be stated as follows:

$$\begin{aligned} \delta &= \arg\min_{\delta} \mathcal{M}(x, x + \delta) \\ s.t., &\arg\max_{j=1,\cdots,m} F_\theta(x)[j] \neq y. \end{aligned} \tag{8}$$

Intuitively, except directly optimizing $\mathcal{M}(x, x + \delta)$ under the optimization constraint, an alternative is to slack the constraint, putting

$\arg\max_{j=1,\cdots,m} F_\theta(x)[j] \neq y$ into objective function as a punishment term, formulated as follows:

$$\delta = \arg\min_{\delta} \mathcal{M}(x, x+\delta) - \alpha L(F_\theta(x+\delta), y), \tag{9}$$

where $F_\theta(x+\delta) \neq y$ is substituted by differentiable $L(F_\theta(x+\delta), y)$ for making gradient-based optimization methods applicable to this task.

Equation 9 is a more efficient and effective way of searching for δ compared to Eq. 8 for generating adversarial examples that are both effective and imperceptible to humans. The search direction[2] used in Eq. 9 is informed by both the attack effectiveness metric and human imperceptibility metric, while the search direction in Eq. 8 only considers one of the two metrics. Therefore, crafting δ via Eq. 9 appears to be a better option.

However, a significant challenge in solving Eq. 9 is determining an appropriate value of α that balances the effectiveness of the attack and the human imperceptibility metrics. We discuss the corresponding solution to this challenge in Sect. 5. In the next section, we define the metrics for measuring the effectiveness of the attack and the human imperceptibility of the perturbation.

4 Metric Design

Threatening adversarial examples possess two crucial characteristics: human imperceptibility and attack effectiveness, which dominate the quality of resultant adversarial examples.

4.1 Human Imperceptibility Metrics

The fundamental objective of human imperceptibility metrics is to approximate the ground-truth human perception distance[3] between two different images. However, most previous works have conveniently adopted norm-based distance functions as the similarity distance function as $\mathcal{M}(x, y) = ||x - y||_a$. The ∞-norm distance function is the most widely used method, which calculates the maximum absolute difference between the elements of two images $|x - y|$.

The primary flaw of norm-based distance functions is insufficiently aligned closely with the human perceptible distance function. Therefore, we introduce CIEDE2000, which has been shown to have better alignment with human perception than norm-based distance functions, to replace norm-based distance functions [19]. CIEDE2000 maps the two images from RGB space to CIELAB

[2] Gradient-based optimization methods are commonly used and effective for solving such tasks and we also follow it. Furthermore, the search direction of optimization methods is the gradient direction of the objective function.

[3] The similarity distance function in this paper is a loose version of the distance measure defined in mathematics, as a strict distance measure should satisfy non-negativity, symmetry, and triangle inequality but sometimes human perception distance may violate triangle inequality.

space since the human perceptible distance between two images is not uniformly affected by the RGB space distance. Specifically, it computes the distance between the two images as a weighted sum of the differences in lightness, chroma, and hue in CIELAB space. This mapping results in a distance metric that more accurately reflects the human visual system's response to differences in color and brightness.

4.2 Attack Effectiveness Metrics

Proxy functions for ASR considerably influence the crafted adversarial examples, motivating us to explore a variety of potential proxy functions to get better results. We design 7 proxy functions, expressed as follows:

$$\begin{aligned}
f_1(x,y) &= F(x)[y],\\
f_2(x,y) &= \text{softmax}(F(x))[y],\\
f_3(x,y) &= \log(f_2(x,y)),\\
f_4(x,y) &= \frac{1}{1-f_2(x,y)} f_2(x,y),\\
f_5(x,y) &= f_2(x,y) - \arg\max_{j\neq y}\{f_2(x,j)\},\\
f_6(x,y) &= \max\{f_4(x,y)+C, 0\},\ C \geq 0,\\
f_7(x,y) &= \frac{f_2(x,y)}{\arg\max_{j\neq y, j=1,\cdots,m,} f_2(x,j)}.
\end{aligned} \tag{10}$$

Functions f_1 and f_2 directly penalize the prediction confidence, normalized prediction confidence, and probability of the ground-truth label for input x. Function f_3 is a negative cross-entropy loss function commonly used in many adversarial attacks, such as FGSM, BIM, and PGD. Function f_4 is an improved version of f_2, taking into account the observation that higher values of $f_2(x,y)$ indicate a higher probability that x is correctly classified by the DNN. To account for this, we scale the magnitude of $f_2(x,y)$ by a regulatory factor $\frac{1}{1-f_2(x,y)}$, which amplifies the value of $f_2(x,y)$ when it is high. This weight tuning can be interpreted as implicitly adjusting the step size during the search process.

The proxy functions $f_1 \sim f_4$ have a limitation in that they only take into account the correct category of the input and do not consider other category information that could guide the search direction for effective adversarial examples. This can be addressed by incorporating similar information between categories. Therefore, we propose proxy functions $f_5 \sim f_7$, which consider the category most similar to the ground-truth label y that the model predicts as the target category for the adversarial attack. To further improve the performance of the proxy functions, we introduce an adaptive magnitude function in f_6 and f_7 that takes into account the model's confidence C in its misclassification of x. Specifically, f_6 disregards the attack effectiveness if the model confidently misclassifies x while f_7 always considers the attack effectiveness throughout the search process.

Notably, we derive $f_4 \sim f_7$ based on f_2 instead of f_1 or f_3 because we can easily tune the hyperparameter C and observe the prediction change trend of the model for x when the prediction is in probability form.

5 Search Algorithm Design

In the context of solving Eq. 8 and Eq. 9, the design of search algorithms is a crucial step that involves three main components: initialization strategy, search direction, and step size. The initialization strategy plays a critical role in determining the success of the search algorithm. There are two main types of initialization strategies: interior initialization and exterior initialization. For Eq. 8, we use inner-optim and outer-optim to refer to the search algorithm with interior initialization and exterior initialization, respectively. Similarly, the terms, inner-joint-optim and outer-joint-optim, are used for Eq. 9.

5.1 Inner-Optim

Initialization Strategy. In the inner-optim search algorithm, the initialization of adversarial perturbations δ needs to conform to the restrict condition $F_\theta(x+\delta) \neq y$, which implies that the model should identify $x+\delta$ as belonging to other categories. To achieve this, a simple way is to initialize δ such that $x+\delta$ becomes a sample belonging to a category different from y. Here, x' can be extracted from D_{train} and then $\delta = x' - x$.

Search Direction. We employ the gradient descent algorithm to move δ in the direction that $M(x, x+\delta)$ decreases the most, i.e., the negative gradient direction of $M(x, x+\delta)$ with respect to δ. However, simply using this algorithm can cause a violation of the optimization constraint since the similarity between x and $x+\delta$ increases with the number of iterations, leading to the increasing probability of x being correctly identified by the model. To prevent this issue, before updating δ in each iteration, the algorithm examines whether this update can result in $F_\theta(x+\delta) = y$. If $F_\theta(x+\delta) = y$, the update is abandoned, and the search process is terminated. Otherwise, the algorithm runs normally.

Adative Step Size Strategy. The appearance of $F_\theta(x+\delta) = y$ may be attributed to the large initialization step size and the smaller step size is worth exploring for searching more human-imperceptible δ. Therefore, the adaptive step size strategy is introduced into the search process and the strategy allows decreasing step size to implement more fine-grained search. In detail, if a certain update leads to $F_\theta(x+\delta) = y$, the step size will be reduced to half of the original one, and then examining the condition again. Also, the procedure is usually implemented several times. If all attempts fail, the search process is terminated.

5.2 Outer-Optim

Initialization Strategy. In the outer-optim algorithm, the initialization of δ should ensure that $F_\theta(x+\delta) = y$ for δ, and not allow $F_\theta(x+\delta) \neq y$. A simple approach to achieve this is to set δ as a full-zero vector, so that $x + \delta = x$ and $F_\theta(x + \delta) = y$.

Search Direction. With the above initialization strategy, the objective function can directly obtain the optimal value 0, but the perturbations are not threatening. Hence, outer-optim algorithm should move the perturbations towards the direction that induces $F_\theta(x + \delta) \neq y$ with minimal perturbations and this direction should be the optimal search direction. However, the gradient direction of $\mathcal{M}(x, x + \delta)$ alone is not sufficient to suggest the optimal direction because the gradient direction of $\mathcal{M}(x, x + \delta)$ not contain any information about $F_\theta(\cdot)$, considering the desired perturbations can give rise to $F_\theta(x + \delta) \neq y$. There are two alternatives to intuitively approximate the optimal direction. The first one is to jointly optimize two metrics for attack effectiveness and human imperceptibility and this is our inner-joint-optim and outer-joint-optim search algorithms; the last one is alternatively optimizing the two metrics and we discuss it in the next section. Here we more focus on leveraging the gradient direction of one of the two metrics as the search direction. The initial δ is the perturbations that enable x and $x + \delta$ to be most similar and thus we should attach more attention to the constraint, i.e., how move δ to obtain $F_\theta(x + \delta) \neq y$. If $F_\theta(x + \delta) \neq y$ is differentiable, the most effective direction is its gradient direction, but, unfortunately, it is not differentiable; thus, we use the gradient direction of the proxy function of $F_\theta(x + \delta) \neq y$. In addition, if $x + \delta$ is misclassified by the model, the search process should be ended as earlier as possible, because intuitively the move probably can increase $M(x, x + \delta)$.

Step Size Strategy. Similarly, we employ the adaptive step size strategy discussed in Sect. 5.1 to efficiently search for better adversarial perturbations.

5.3 Inner-Joint-Optim and Outer-Joint-Optim

Equation 9 generally performs better than individually optimizing one of the metrics. However, a key challenge is determining an appropriate value for the weighting parameter α. Setting a small α prioritizes human imperceptibility over attack effectiveness, potentially leading to ineffective adversarial perturbations. For instance, if $\alpha = 0$, the algorithm will exclusively focus on making $\delta = 0$. Intuitively, there are two approaches to solving the problem: 1) Setting a large α focuses solely on attack effectiveness and ignores human imperceptibility, resulting in overly perceptible perturbations; 2) An alternative approach of alternating between optimizing the two metrics based on whether the adversarial example is correctly classified has been proposed. Specifically, if the adversarial example is correctly classified, we optimize the attack effectiveness metric, otherwise, we optimize the similarity metric. This method still fails to fully explore the relationship between the two metrics.

We propose an adaptive method to find the optimal value of α that balances attack effectiveness and human imperceptibility. As α is increased, the model transitions from correctly classifying the sample to misclassifying it. This indicates that there is a tipping point to cause misclassification and the tipping point is the optimal value for α. With the optimal value for α, the produced adversarial perturbations are most human-imperceptible and also remain threatening. However, the optimal value of α is unknown in advance. Therefore, in each iteration, we increase the weight of the human-imperceptible metric if the sample with adversarial perturbations is misclassified, and we increase the weight of the attack effectiveness metric otherwise. This adaptive approach enables us to determine the optimal value of α and generate the optimal adversarial perturbations.

Initialization Strategy and Step Size. We introduce two variations of the joint optimization approach: inner-joint-optim when using the initialization strategy of inner-optim, and outer-joint-optim otherwise. We additionally incorporate a step size tuning strategy into the optimization process, which linearly decreases the step size to zero with iterations.

6 Experimental Evaluation

We implement a PyTorch-based prototype of typeII-EvaA based on CIEDE2000 to evaluate its performance on two commonly used *benchmark datasets*, CIFAR10 [10] and ImageNet [5]. We assess the effectiveness of typeII-EvaA attack methods against *SOTA defense mechanisms*, as Huang2021Exploring [8], Sridhar2021Robust [20], Pang2022Robustness [13] and Dai2021Parameterizing [4] for CIFAR10, and Standard, Wong2020Fast [22], Salman2020Do [16], and Engstrom2019Robustness [6] for ImageNet. To ensure fairness, the experimental model settings are consistent with those used in prior works and the step size is 0.005. Additionally, four advanced attacks are considered as *baselines* to estimate the effectiveness of our typeII-EvaA: FGSM [21], BIM [11], PGD [12], and C&W [2]. For all experiments, ASR, indicating the accuracy success rate, is regarded as the evaluation metric of typeII-AssMs. The typeII-AssMs' goal is to maximize ASR.

Evaluation of Attacks. We leverage four perturbation δ search algorithms of typeII-EvaA to evaluate the performance of SOTA defense mechanisms. We report the ASR in Fig. 2 over CIFAR10 and Fig. 3 over ImageNet along with various proxy functions. The results of Fig. 2 almost reaffirm the fact that, as the perturbation δ increases, the ASR of typeII-EvaA also increases. Obviously, The inner-joint-optim and outer-joint-optim outperform the inner-optim and outer-optim, respectively. This means that the strategy of slacking the constraint is more effective when against the defense mechanisms. For ImageNet dataset, we concentrate on inner-joint-optim and outer-joint-optim with the f_4 and f_7 functions. The effectiveness of inner-joint-optim with f_4 is dramatically improved. More specifically, when $\delta = 6.16$, the ASR of inner-joint-optim search algorithm with f_7 is 0.46, while when $\delta = 1.13$, the ASR of inner-joint-optim search algorithm with f_4 is up to 0.89. Additionally, the trend in ASR of outer-joint-optim search algorithm is the same as that over CIFAR10.

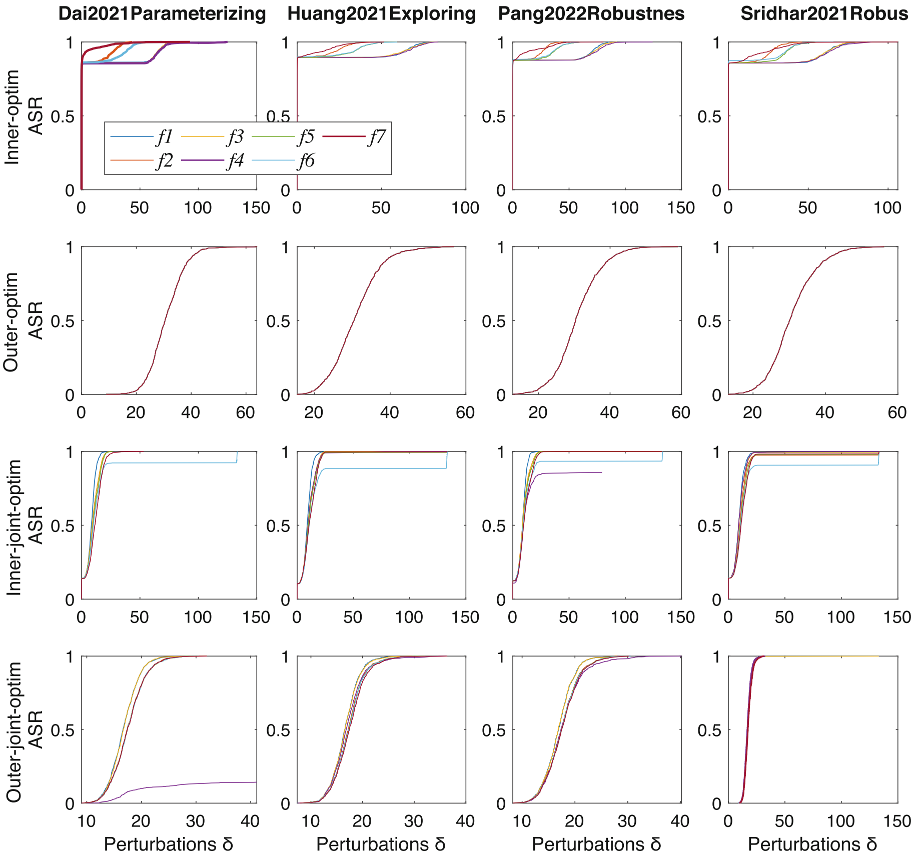
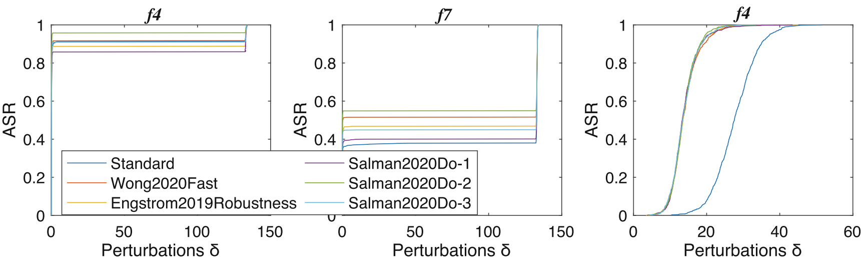


Fig. 2. The performance of SOTA defense mechanisms against typeII-EvaA over CIFAR10 dataset.



(a) Inner-joint-optim (b) Inner-joint-optim (c) Outer-joint-optim

Fig. 3. The performance of SOTA defense mechanisms against typeII-EvaA over ImageNet dataset.

Table 1. Evaluation between typeII-EvaA and advanced attacks with various perturbation parameter (para.) δ and proxy functions (ASR: %).

CIFAR10	Para.	[4]	[8]	[13]	[20]	Attack	Para.	[4]	[8]	[13]	[20]
FGSM	$\delta = 1$	14.96	11.27	14.84	17.3	PGD	$\delta = 1$	14.96	11.27	14.84	17.3
	$\delta = 2$	16.18	14.06	15.74	19.08		$\delta = 2$	16.18	14.17	15.85	19.2
	$\delta = 4$	20.76	17.86	20.65	21.99		$\delta = 4$	20.98	18.97	21.54	22.66
	$\delta = 8$	29.24	26.79	29.58	30.8		$\delta = 8$	33.93	33.26	33.37	36.05
	$\delta = 12$	40.4	33.04	39.4	38.17		$\delta = 12$	50.22	53.68	48.88	50.56
	$\delta = 16$	47.1	40.07	45.76	44.53		$\delta = 16$	65.74	70.31	67.3	66.18
BIM	$\delta = 1$	14.96	11.27	14.84	17.3	C&W	$\delta = 1$	15.18	11.72	15.07	17.52
	$\delta = 2$	16.18	14.06	15.85	19.2		$\delta = 2$	16.52	14.4	16.63	19.31
	$\delta = 4$	20.98	18.97	21.54	22.66		$\delta = 4$	21.76	19.87	22.54	23.21
	$\delta = 8$	33.82	33.15	33.15	35.71		$\delta = 8$	35.94	34.6	36.5	38.5
	$\delta = 12$	50.22	52.79	48.66	50.33		$\delta = 12$	52.23	57.03	53.79	54.13
	$\delta = 16$	65.62	69.53	66.96	65.62		$\delta = 16$	68.86	75.22	71.54	70.65
Inner-joint	f_1	100	100	100	99.33	Outer-joint	f_1	100	100	100	99.89
	f_2	100	100	100	97.43		f_2	100	100	100	100
	f_3	100	100	100	99.33		f_3	100	100	100	99.89
	f_4	90.4	100	97.54	99.78		f_4	91.29	100	100	100
	f_5	100	100	100	97.77		f_5	99.89	100	100	100
	f_6	96.88	98.1	100	96.65		f_6	99.89	100	100	100
	f_7	100	100	100	98.44		f_7	98.88	100	100	100
ImageNet	Para.	Standard	[22]	[6]	[16]	Attack	Para.	Standard	[22]	[6]	[16]
FGSM	$\delta = 16$	93.15	92.64	88.51	94.46	PGD	$\delta = 16$	100	98.89	98.89	99.19
BIM	$\delta = 16$	100	98.89	98.79	99.19	C&W	$\delta = 16$	100	99.4	99.6	99.7
Inner-joint	f_4	100	100	100	100	Outer-joint	f_4	100	100	100	100
	f_7	100	99.9	99.9	100		f_7	100	100	100	100

Comparison with SOTA. We report the evaluation results compared with existing attacks against defense mechanisms and demonstrate the superiority of the typeII-EvaA in Table 1. We evaluate the quality of the adversarial examples found on the CIFAR10 and ImageNet datasets. The parameters, like proxy functions and perturbation δ are identical between the two datasets, so for brevity, we report partial results for ImageNet. For CIFAR10, all of the previous attacks fail to find adversarial examples. In contrast, our inner-joint-optim and outer-joint-optim can achieve 100% ASR when against various defense mechanisms. For ImageNet, prior work achieves approximate 99% ASR against advanced defense mechanisms when $\delta = 16$. While our inner-joint-optim and outer-joint-optim succeed with 100% success probability for each of the seven proxy functions.

7 Conclusion

The vulnerability of deep learning models to adversarial examples presents a major challenge for their practical application in security-critical domains. In order to ensure the reliability and safety of these models, it is crucial to evaluate their robustness against adversarial attacks. In this paper, we propose powerful attacks typeII-EvaA that defeat advanced defense mechanisms, demonstrating that typeII-EvaA more generally can be used to evaluate the efficacy of potential defenses. By systematically comparing many attack approaches, we settle on one that can consistently find better adversarial examples than all existing approaches with linear time complexity. We encourage those who create defenses to perform the four evaluation approaches with various proxy functions we use in this paper.

References

1. Bastani, O., Ioannou, Y., Lampropoulos, L., Vytiniotis, D., Nori, A., Criminisi, A.: Measuring neural net robustness with constraints. In: Advances in Neural Information Processing Systems 29 (2016)
2. Carlini, N., Wagner, D.: Towards evaluating the robustness of neural networks. In: 2017 IEEE Symposium on Security and Privacy (SP), pp. 39–57. IEEE (2017)
3. Chen, K., Zhu, H., Yan, L., Wang, J.: A survey on adversarial examples in deep learning. J. Big Data **2**(2), 71 (2020)
4. Dai, S., Mahloujifar, S., Mittal, P.: Parameterizing activation functions for adversarial robustness. In: 2022 IEEE Security and Privacy Workshops (SPW), pp. 80–87. IEEE (2022)
5. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: a large-scale hierarchical image database (2009). https://doi.org/10.1109/CVPR.2009.5206848
6. Engstrom, L., Ilyas, A., Santurkar, S., Tsipras, D., Tran, B., Madry, A.: Adversarial robustness as a prior for learned representations. arXiv preprint arXiv:1906.00945 (2019)
7. Gu, S., Rigazio, L.: Towards deep neural network architectures robust to adversarial examples. arXiv preprint arXiv:1412.5068 (2014)
8. Huang, H., Wang, Y., Erfani, S., Gu, Q., Bailey, J., Ma, X.: Exploring architectural ingredients of adversarially robust deep neural networks. Adv. Neural. Inf. Process. Syst. **34**, 5545–5559 (2021)
9. Huang, R., Xu, B., Schuurmans, D., Szepesvári, C.: Learning with a strong adversary. arXiv preprint arXiv:1511.03034 (2015)
10. Krizhevsky, A., Nair, V., Hinton, G.: The CIFAR-10 dataset **55**(5) (2014). https://www.cs.toronto.edu/kriz/cifar.html
11. Kurakin, A., Goodfellow, I.J., Bengio, S.: Adversarial examples in the physical world. In: Artificial intelligence safety and security, pp. 99–112. Chapman and Hall/CRC (2018)
12. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083 (2017)
13. Pang, T., Lin, M., Yang, X., Zhu, J., Yan, S.: Robustness and accuracy could be reconcilable by (proper) definition. In: International Conference on Machine Learning, pp. 17258–17277. PMLR (2022)
14. Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z.B., Swami, A.: Practical black-box attacks against deep learning systems using adversarial examples. arXiv preprint arXiv:1602.02697 (2016)
15. Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z.B., Swami, A.: Practical black-box attacks against machine learning. In: Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security, pp. 506–519 (2017)

16. Salman, H., Ilyas, A., Engstrom, L., Kapoor, A., Madry, A.: Do adversarially robust imagenet models transfer better? Adv. Neural. Inf. Process. Syst. **33**, 3533–3545 (2020)
17. Samangouei, P., Kabkab, M., Chellappa, R.: Defense-gan: protecting classifiers against adversarial attacks using generative models. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 933–941 (2018)
18. Shaham, U., Yamada, Y., Negahban, S.: Understanding adversarial training: increasing local stability of neural nets through robust optimization. arXiv preprint arXiv:1511.05432 (2015)
19. Sharma, G., Wu, W., Dalal, E.N.: The CIEDE2000 color-difference formula: implementation notes, supplementary test data, and mathematical observations. Color Res. Appli. **30**(1), 21–30 (2005)
20. Sridhar, K., Sokolsky, O., Lee, I., Weimer, J.: Improving neural network robustness via persistency of excitation. In: 2022 American Control Conference (ACC), pp. 1521–1526. IEEE (2022)
21. Szegedy, C., et al.: Intriguing properties of neural networks. arXiv preprint arXiv:1312.6199 (2013)
22. Wong, E., Rice, L., Kolter, J.Z.: Fast is better than free: revisiting adversarial training. arXiv preprint arXiv:2001.03994 (2020)