UniCR: Universally Approximated Certified Robustness via Randomized Smoothing

Hanbin Hong[1], Binghui Wang[2], and Yuan Hong[1]

[1] University of Connecticut, Storrs CT 06269, USA
hanbin.hong@uconn.edu, yuan.hong@uconn.edu
[2] Illinois Institute of Technology, Chicago IL 60616, USA
bwang70@iit.edu

A Preliminary

We first briefly review the recent certified robustness scheme [4] for a general classification problem by classifying data point in $\mathbb{R}^d$ to classes in $\mathcal{Y}$. Given an arbitrary base classifier f, it can be converted to a "smoothed" classifier [4] g by adding isotropic Gaussian noise to the input x:

$$g(x) = \arg\max_{c \in \mathcal{Y}} \mathbb{P}(f(x+\epsilon) = c), where \;\; \epsilon \sim \mathcal{N}(0, \sigma^2 I) \tag{1}$$

Lemma 1. *(**Neyman-Pearson Lemma**) Let X and Y be random variables in $\mathbb{R}^d$ with densities μ_X and μ_Y. Let $f : \mathbb{R}^d \to \{0, 1\}$ be a random or deterministic function. Then:*

(1) If $S = \{z \in \mathbb{R}^d : \frac{\mu_Y(z)}{\mu_X(z)} \leq t\}$ for some $t > 0$ and $\mathbb{P}(f(X) = 1) \geq \mathbb{P}(X \in S)$, then $\mathbb{P}(f(Y) = 1) \geq \mathbb{P}(Y \in S)$;

(2) If $S = \{z \in \mathbb{R}^d : \frac{\mu_Y(z)}{\mu_X(z)} \geq t\}$ for some $t > 0$ and $\mathbb{P}(f(X) = 1) \leq \mathbb{P}(X \in S)$, then $\mathbb{P}(f(Y) = 1) \leq \mathbb{P}(Y \in S)$.

With Lemma 1, Cohen [4] derives the certified radius when the classifier is smoothed with the Gaussian noise. As shown in Theorem 1, when the smoothed classifier's prediction probabilities satisfy Equation (2), the prediction result is guaranteed to be the most probable class c_A when the perturbation is limited within a radius R in ℓ_2-norm.

Theorem 1. *(**Randomized Smoothing with Gaussian Noise [4]**) Let $f : \mathbb{R}^d \to \mathcal{Y}$ be any deterministic or random function, and let $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$. Denote g as the smoothed classifier in Equation (1), and the most probable and the second probable classes as $c_A, c_B \in \mathcal{Y}$, respectively. If the lower bound of the class c_A's prediction probability $\underline{p_A} \in [0, 1]$, and the upper bound of the class c_B's prediction probability $\overline{p_B} \in [0, 1]$ satisfy:*

$$\mathbb{P}(f(x+\epsilon) = c_A) \geq \underline{p_A} \geq \overline{p_B} \geq \max_{c \neq c_A} \mathbb{P}(f(x+\epsilon) = c) \tag{2}$$

Then $g(x + \delta) = c_A$ for all $||\delta||_2 \leq R$, where

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

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

Proof. See detailed proof in [4]. □

B Proofs

B.1 Proof of Theorem 1

Proof. We prove the theorem based on Neyman-Pearson Lemma (Lemma 1).

Let $x := x_0 + \epsilon$ be the random variable that follows any continuous distribution. δ be the perturbation added to the input image. $y = x_0 + \epsilon + \delta$ is the perturbed random variable. Thus, x and y are random variables with densities μ_x and μ_y. Define sets:

$$A := \{z : \frac{\mu_y(z)}{\mu_x(z)} \leq t_A\} \tag{4}$$

$$B := \{z : \frac{\mu_y(z)}{\mu_x(z)} \geq t_B\} \tag{5}$$

where t_A and t_B are picked to suffice:

$$\mathbb{P}(x \in A) = \underline{p_A} \tag{6}$$

$$\mathbb{P}(x \in B) = \overline{p_B} \tag{7}$$

Suppose $c_A \in \mathcal{Y}$ and $\underline{p_A}, \overline{p_B} \in [0, 1]$ satisfy:

$$\mathbb{P}(f(x + \epsilon) = c_A) \geq \underline{p_A} \geq \overline{p_B} \geq \max_{c \neq c_A} \mathbb{P}(f(x + \epsilon) = c) \tag{8}$$

Since $\mathbb{P}(f(x + \epsilon) = c_A) \geq \underline{p_A} = \mathbb{P}(x \in A)$ and $A = \{z : \frac{\mu_Y(z)}{\mu_X(z)} \leq t_A\}$, using Neyman-Pearson Lemma (Lemma 1), we have:

$$\mathbb{P}(f(y) = c_A) \geq \mathbb{P}(y \in A) \tag{9}$$

Similarly, we have:

$$\mathbb{P}(f(y) = c_B) \leq \mathbb{P}(y \in B) \tag{10}$$

To guarantee $\mathbb{P}(f(y) = c_A) \geq \mathbb{P}(f(y) = c_B)$, we need

$$\mathbb{P}(f(y) = c_A) \geq \mathbb{P}(y \in A) \geq \mathbb{P}(y \in B) \geq \mathbb{P}(f(y) = c_B) \tag{11}$$

In summary, to guarantee the certified robustness on class A, Equation (4), (5), (6), (7), (11) must be satisfied. The conditions can be rewritten as:

$$\mathbb{P}(\frac{\mu_y(x)}{\mu_x(x)} \leq t_A) = \underline{p_A} \tag{12}$$

$$\mathbb{P}(\frac{\mu_y(x)}{\mu_x(x)} \geq t_B) = \overline{p_B} \tag{13}$$

$$\mathbb{P}(\frac{\mu_y(y)}{\mu_x(y)} \leq t_A) \geq \mathbb{P}(\frac{\mu_y(y)}{\mu_x(y)} \geq t_B) \tag{14}$$

where Equation (12) is from Equation (4) and Equation (6), Equation (13) is from Equation (5) and Equation (7), and Equation (14) is from Equation (11).

Considering the relationship $y = x + \delta$, we can derive:

$$\mu_y(x) = \mu_x(x - \delta) \tag{15}$$

$$\mu_x(y) = \mu_x(x + \delta) \tag{16}$$

$$\mu_y(y) = \mu_x(y - \delta) = \mu_x(x) \tag{17}$$

Thus, the conditions (11), (12) and (13) can be rewritten as:

$$\mathbb{P}(\frac{\mu_x(x - \delta)}{\mu_x(x)} \leq t_A) = \underline{p_A} \tag{18}$$

$$\mathbb{P}(\frac{\mu_x(x - \delta)}{\mu_x(x)} \geq t_B) = \overline{p_B} \tag{19}$$

$$\mathbb{P}(\frac{\mu_x(x)}{\mu_x(x + \delta)} \leq t_A) \geq \mathbb{P}(\frac{\mu_x(x)}{\mu_x(x + \delta)} \geq t_B) \tag{20}$$

Any perturbation δ satisfying these conditions will not fool the smoothed classifier. In this case, these conditions construct a robustness area in δ space. If we want to find a ℓ_p ball within which the prediction is constant, the l_p ball should be in this robustness area. Therefore, the certified radii is the minimum $||\delta||_p$ on the boundary of this robustness area. In this case, the ℓ_p ball is exactly the maximum inscribed ball in the robustness area. Also, x can be replace by ϵ in these conditions since it is in the fraction, which means the optimization is independent to the input if given $\underline{p_A}$ and $\overline{p_B}$. Therefore, the whole optimization problem is summarized as:

$$\begin{aligned}
\underset{\delta}{\text{minimize}} \quad & R = ||\delta||_p \\
\text{subject to} \quad & \mathbb{P}(\frac{\mu_x(x - \delta)}{\mu_x(x)} \leq t_A) = \underline{p_A}, \\
& \mathbb{P}(\frac{\mu_x(x - \delta)}{\mu_x(x)} \geq t_B) = \overline{p_B}, \\
& \mathbb{P}(\frac{\mu_x(x)}{\mu_x(x + \delta)} \leq t_A) = \mathbb{P}(\frac{\mu_x(x)}{\mu_x(x + \delta)} \geq t_B)
\end{aligned}$$

If the noise is isotropic, each dimension is independent,

$$\mu_x(x) = \prod_{i=1}^{d} \mu_x(x_j) \tag{21}$$

Thus, conditions for the isotropic noise can be rewritten as:

$$\mathbb{P}(\prod_{j=1}^{d} \frac{\mu_x(x_j - \delta_j)}{\mu_x(x_j)} \leq t_A) = \underline{p_A} \tag{22}$$

$$\mathbb{P}(\prod_{j=1}^{d} \frac{\mu_x(x_j - \delta_j)}{\mu_x(x_j)} \geq t_B) = \overline{p_B} \tag{23}$$

$$\mathbb{P}(\prod_{j=1}^{d} \frac{\mu_x(x_j)}{\mu_x(x_j + \delta_j)} \leq t_A) = \mathbb{P}(\prod_{j=1}^{d} \frac{\mu_x(x_j)}{\mu_x(x_j + \delta_j)} \geq t_B) \tag{24}$$

Thus, this completes the proof. □

B.2 Binary Case for Theorem 2

Theorem 2. *(**Universal Certified Robustness (Binary Case)**) Let $f : \mathbb{R}^d \to \mathcal{Y}$ be any deterministic or random function, and let ϵ follows any continuous distribution. Let g be defined as in (1). Suppose the most probable class $c_A \in \mathcal{Y}$ and the lower bound of the probability $\underline{p_A}$ satisfy:*

$$\mathbb{P}(f + \epsilon) = c_A \geq \underline{p_A} \geq \frac{1}{2} \tag{25}$$

Then $g(x + \delta) = c_A$ for all $||\delta||_p \leq R$, where R is given by the optimization:

$$\begin{aligned} \underset{\delta}{\text{minimize}} \quad & R = ||\delta||_p \\ \text{subject to} \quad & \mathbb{P}(\frac{\mu_x(x - \delta)}{\mu_x(x)} \leq t_A) = \underline{p_A}, \\ & \mathbb{P}(\frac{\mu_x(x)}{\mu_x(x + \delta)} \leq t_A) = \frac{1}{2} \end{aligned}$$

B.3 UniCR (Binary Case)

Similar to the binary case of Theorem 2, the binary case of the two-phase optimization can be easily derived:

$$\begin{aligned} & R = ||\lambda\boldsymbol{\delta}||_p, \textit{where } \boldsymbol{\delta} \in \arg\min_{\delta} ||\lambda\boldsymbol{\delta}||_p \\ & s.t. \quad \lambda = \arg\min_{\lambda} |K| \\ & \qquad \mathbb{P}(\frac{\mu_x(x - \lambda\boldsymbol{\delta})}{\mu_x(x)} \leq t_A) = \underline{p_A} \\ & \qquad K = \mathbb{P}(\frac{\mu_x(x)}{\mu_x(x + \lambda\boldsymbol{\delta})} \leq t_A) - \frac{1}{2} \\ & \qquad \underline{p_A} \geq \frac{1}{2} \end{aligned}$$

B.4 UniCR Bound

The certified radius R approximated by the two-phase optimization is tight if achieving the optimality. Under this assumption, we analysis the confidence bound for the certification. We follow [4] to compute the probabilities $\underline{p_A}$ and $\overline{p_B}$ using Monte Carlo method with sample number n. The confidence is $\overline{1-\alpha_0}$, where $p_A >= \alpha_0^{1/n}$. To estimate the auxiliary parameters t_A and t_B, we use Dvoretzky–Kiefer–Wolfowitz inequality [6] to bound the CDFs of the random variables $\frac{\mu_x(x-\lambda\delta)}{\mu_x(x)}$ and $\frac{\mu_x(x)}{\mu_x(x+\lambda\delta)}$, then determine the t_A and t_B using Algorithm 1.

Lemma 2. *(**Dvoretzky–Kiefer–Wolfowitz inequality**(restate)) Let $X_1, X_2, ..., X_n$ be real-valued independent and identically distributed random variables with cumulative distribution function $F(\cdot)$, where $n \in \mathbb{N}$.Let F_n denotes the associated empirical distribution function defined by*

$$F_n(x) = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}_{\{X_i<=x\}}, x \in \mathbb{R} \tag{26}$$

The Dvoretzky–Kiefer–Wolfowitz inequality bounds the probability that the random function F_n differs from F by more than a given constant $\Delta \in \mathbb{R}^+$:

$$\mathbb{P}(\sup_{x\in\mathbb{R}} |F_n(x) - F(x)| > \Delta) \leq 2e^{-2n\Delta^2} \tag{27}$$

We use the Lemma 2 to estimate the CDFs in algorithm 1. In the robustness condition, we need to estimate 4 probabilities with confidence $1-2e^{-2n\Delta^2}$ as well as the p_A and p_B with confidence $1-\alpha_0$. Therefore, the confidence that deriving the correct radius is at least $(1-\alpha_0)^2(1-2e^{-2n\Delta^2})^4$. In Figure 1, we show the confidence on a varying number of samples when $\Delta = 0.1$ and $\alpha_0 = 0.999$. As the number of samples increases to around 400 (all our experiments use more than 400 samples), our confidence is very close to Cohen's confidence [4]. Thus, the confidence is nearly 1 in all our experiments.

B.5 Optimization Convergence

We analysis the convergence of the two-phase optimization and the certification accuracy in this section. On one hand, the optimality of the scalar optimization can be asymptotically achieved by binary search. On the other hand, it is hard to find the minimum $||\lambda\delta||_p$ in the highly-dimensional space, but some special symmetry in the direction of δ (e.g., spherical symmetry that is also found in [18, 17]), can help approximate the certified radius. The detailed algorithms are presented in Section 3.1. The defense performance of such universally approximated certified robustness against different real-world attacks is the same as certified robustness (as shown in Appendix D.2). Thus, such negligible approximation error is close to 0, but result in many significant new benefits in return.

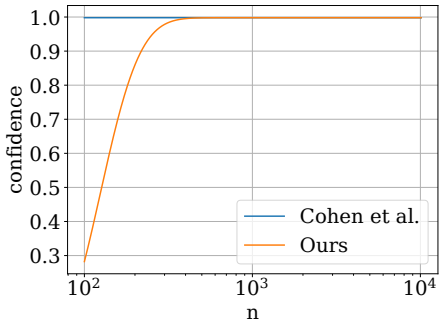


Fig. 1. Confidence vs. number of Monte Carlo samples.

C Algorithms

C.1 Computing t_A and t_B

We present the algorithm to compute the p_A and p_B in Algorithm 1.

Algorithm 1 Computing t_A and t_B

Input: Lower bound of the probabilities, $\underline{p_A}$; upper bound of the probabilities, $\overline{p_B}$; perturbation scalar, λ; perturbation, δ; noise PDF, μ_x; number of samples in the Monte Carlo method, n

Output: The auxiliary parameters, t_A and t_B

1: Sample n noise $\epsilon \in \mathbb{R}^{n \times d}$ from a discrete version of PDF.
2: Calculate $\frac{\mu_x(x-\lambda\delta)}{\mu_x(x)}$ using these n samples of noise, μ_x, λ and δ
3: Estimate the CDF Φ of $\frac{\mu_x(x-\lambda\delta)}{\mu_x(x)}$ using Monte Carlo method
4: **return** $t_A = \Phi^{-1}(p_A)$ and $t_B = \Phi^{-1}(p_B)$, with inverse CDF Φ

C.2 Scalar Optimization

We use the binary search to find a scale factor that minimizes $|K|$ (*the distance between δ and the robustness boundary*). When $K = 0$, the perturbation δ is exactly on the robustness boundary. Fixing the direction of δ, we find two scalars such that $K > 0$ and $K < 0$. Specifically, we start from a scalar λ_a and compute K. If $K > 0$, then the scaled perturbation $\lambda_a\delta$ is within the robustness boundary, thus we enlarge the scalar to find a λ_b such that $K < 0$ and vice versa. After that, we iteratively compute the K using $\lambda = \frac{1}{2}(\lambda_a + \lambda_b)$: if $K > 0$, we let $\lambda_a = \lambda$; otherwise, we let $\lambda_b = \lambda$. We repeat this iteration until K is less than a threshold or the number of iterations is sufficiently large. The procedures are summarized in Algorithm 2.

Algorithm 2 Scalar Optimization

Input: Lower bound of the probabilities, $\underline{p_A}$; upper bound of the probabilities, $\overline{p_B}$; perturbation scalar, λ; perturbation, δ; noise PDF, μ_x; number of samples in Monte Carlo method, n; threshold for K, K_m; number of iterations for binary search, N
Output: The scalar λ that minimizes $|K|$
Find initial scalar λ_a and λ_b such that $K > 0$ and $K < 0$
$\lambda = (\lambda_a + \lambda_b)/2$
Compute K using λ
while $N > 0$ and $|K| > K_m$ **do**
  if $K > 0$ **then**
    $\lambda_a = \lambda$
  else
    $\lambda_b = \lambda$
  $\lambda = (\lambda_a + \lambda_b)/2$
  Compute K using λ
  N=N-1
return λ

C.3 Direction Optimization

We show how to initialize the positions for different ℓ_p norms in PSO. Since some noise follows PDFs with symmetry [18, 17], we set the initial position of particles by considering this, e.g., setting the initial positions w.r.t. ℓ_p for $p \in \mathbb{R}^+$ as $[0, ..., 0, a, 0, ..., 0]$ and the initial positions w.r.t. ℓ_∞ as $[a, a, a, ..., a]$, where a is a small random number. Although the search space is highly-dimensional, empirical results show that the radius given by PSO can accurately approximate the theoretical radius given by other methods, e.g., Cohen's [4] (see Figure 4 in main context). Notice that, for more complicated PDFs without symmetry (which is indeed difficult for deriving the certified radius), PSO can also approximate the certified radius with more particles and iterations.

C.4 Noise Optimization Algorithms

With the UniCR for estimating the certified radius, we can further tune the noise PDF to each input or the classifier. Specifically, let $\mu(x, \boldsymbol{\alpha})$ denote the noise PDF, where $\boldsymbol{\alpha}$ is a set of hyper-parameters in the function, i.e., $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, ..., \alpha_m]$. We simply use grid-search algorithm to find the best hyper-parameters in the classifier smoothing (C-OPT). For the input noise optimization (I-OPT), we use the Hill-Climbing algorithm to find the optimal hyper-parameters in the function for each input. During the algorithm execution, hyper-parameter for the input is iteratively updated if a better solution is found in each round, until convergence. The procedures for the Hill Climbing algorithm are summarized in Algorithm 3.

Algorithm 3 I-OPT with Hill Climbing

Input: Input data, x; PDF of noise distribution, μ_x; universally approximated certified robustness, UniCR($\cdot$); initial hyper-parameters $\boldsymbol{\alpha}$; optimization range of hyper-parameters, $[\boldsymbol{L}, \boldsymbol{H}]$; optimization step of hyper-parameters, $\boldsymbol{S}$
Output: The optimal hyper-parameters, $\boldsymbol{\alpha}_{optimal}$

Initialize the certified radius $R_0 = \text{UniCR}(x, \mu(\boldsymbol{\alpha}))$
For each hyper-parameter α_i in $\boldsymbol{\alpha}$:
if $L_i < \alpha_i + S_i < H_i$ **then**
  $R' = \text{UniCR}(x, \mu(\boldsymbol{\alpha}|\alpha_i = \alpha_i + S_i))$
  if $R' > R_0$ **then**
    $\boldsymbol{\alpha}$ is updated with $\alpha_i = \alpha_i + S_i$
    $R_0 = R'$
  else if $L_i < \alpha_i - S_i < H_i$ **then**
    $R' = \text{UniCR}(x, \mu(\boldsymbol{\alpha}|\alpha_i = \alpha_i - S_i))$
    $R_0 = R'$
    if $R' > R$ **then**
      $\boldsymbol{\alpha}$ is updated with $\alpha_i = \alpha_i - S_i$
      $R_0 = R'$
  else
    break
else
  break
return $\boldsymbol{\alpha}_{optimal} = \boldsymbol{\alpha}$

D More Experimental Results

D.1 Metrics

We show the illustration of Robustness Score in Figure. 2

D.2 Defense against Real Attacks

We evaluate our UniCR's defense accuracy against a diverse set of state-of-the-art attacks, including universal attacks [3], white-box attacks [5, 15], and black-box attacks [1, 2]. We compare UniCR with other state-of-the-art certified schemes [17, 4, 13] against ℓ_1, ℓ_2 and ℓ_∞ perturbations. The certified radius R for each image in the test set ($10,000$ images in total) are computed beforehand, and the perturbation generation is constrained by $||\delta||_p = R$ for all the attack methods. We define the defense accuracy as the rate that the smoothed classifier can successfully defend against the perturbations with the ℓ_p size identical to the the certified radius:

$$acc_d = \mathbb{E}_{||\delta||_p = R}[\frac{\sum g(x+\delta) = c_A}{N}] \tag{28}$$

where $c_A = g(x)$, N is the total test number. In this defense study, we use 500 samples for both Monte Carlo method and testing.

Table 1 shows the defense accuracy on the smoothed classifier. The attack with "$*$" is re-scaled to the required norm (perturbation size R) based on their perturbation formats. UniCR universally provides a 100% defense accuracy against all the ℓ_1, ℓ_2 and ℓ_∞ perturbations generated by all the state-of-the-art attacks. These results validate our universally approximated certified robustness ensures the same defense performance as certified robustness in practice.

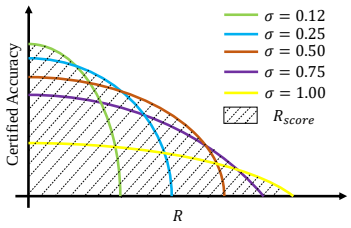


Fig. 2. An example of the Robustness Score.

Table 1. Defense against real attacks on CIFAR10 (results on MNIST & ImageNet are similar and not included due to space limit).

Defense Accuracy (%)	Gaussian*	Procedural*	[3] Auto-PGD	[5] Wasserstein*	[15] Square*	[1] HSJ* [2]
Teng's [13] ℓ_1-norm R	100.00	100.00	100.00	100.00	100.00	100.00
Our ℓ_1-norm R	100.00	100.00	100.00	100.00	100.00	100.00
Cohen's [4] ℓ_2-norm R	100.00	100.00	100.00	100.00	100.00	100.00
Our ℓ_2-norm R	100.00	100.00	100.00	100.00	100.00	100.00
Yang's [17] ℓ_∞-norm R	100.00	100.00	100.00	100.00	100.00	100.00
Our ℓ_∞-norm R	100.00	100.00	100.00	100.00	100.00	100.00

D.3 List of PDFs

The PDFs used in our experimental are summarized in Table 2.

Table 2. List of noise distributions.

Distribution	Probability Density Function
Gaussian	$\propto e^{-\lvert x/\alpha\rvert^2}$
Laplace	$\propto e^{-\lvert x/\alpha\rvert}$
Hyperbolic Secant	$\propto sech(\lvert x/\alpha\rvert)$
General Normal	$\propto e^{-\lvert x/\alpha\rvert^\beta}$
Cauthy	$\propto \frac{\alpha^2}{x^2+\alpha^2}$
Pareto	$\propto \frac{1}{(1+\lvert x/\alpha\rvert)^{\beta+1}}$
Laplace-Gaussian Mix.	$\propto \beta e^{-\lvert x/\alpha\rvert^1} + (1-\beta)e^{-\lvert x/\alpha\rvert^2}$
Exponential Mix.	$\propto e^{-\beta\lvert x/\alpha\rvert^1-(1-\beta)\lvert x/\alpha\rvert^2}$

D.4 Efficiency for Radius Derivation

We show the runime of our algorithms on deriving the certified radius for the inputs with various input dimensions in Figure 3. For the common input dimen-

sions, e.g., 24×24 for MNIST, $3 \times 32 \times 32$ for CIFAR10, and $3 \times 224 \times 224$ for ImageNet, it takes less than 10 seconds for certifying an image on average. Comparing with the theoretical certified radius deriving, our method's running time is undoubtedly larger since their radius is pre-derived. However, with the significant benefits on the universality and the automatically deriving, we believe the cost of the extra running time is worthwhile and acceptable in practice.

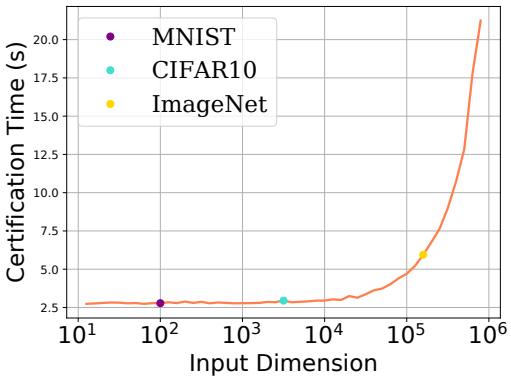


Fig. 3. Runtime of UniCR vs. input sizes (with RTX3080 GPU).

D.5 Any p (besides 1, 2, ∞)

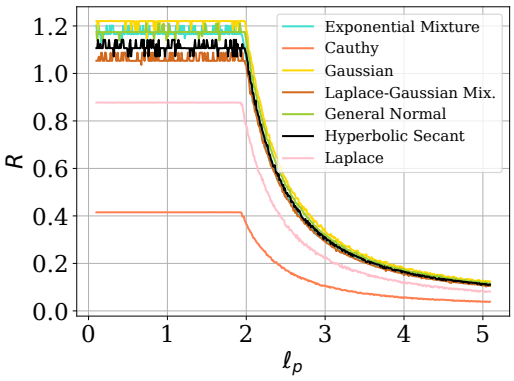


Fig. 4. Radius vs. various ℓ_p pert.

Existing methods [4, 13] usually focus on the certified radius in a specific norm, e.g., ℓ_1, ℓ_2 or ℓ_∞ norms. Some methods [18, 17] provide certified robustness theories for multiple norms but specific settings are usually needed for deriving the certified radii in different norms. None of the existing methods can automatically compute the certified radius in any ℓ_p norm. In this section, we show our UniCR can automatically approximate the certified radii for various p, in which p is a real number greater than 0.

In the experiments, we set the probability $\underline{p_A} = 0.9$ and draw the lines of certified radius w.r.t. different p for $p > 0$. We show the results computed with different noise distributions in Figure 4. We observe that when $p \in (0, 2]$, the certified radius for different p are approximately identical. This finding also matches the theoretical results in Yang et al. [17], in which the certified radii in ℓ_1 and ℓ_2 norm are exactly the same for multiple distributions. When $p > 2$, we observe that the certified radius decreases as p increases.

D.6 Evaluations on Complicated PDFs

We provide a fine-grained evaluation on the complicated distributions [12], e.g., General Normal, Laplace-Gaussian Mixture, and Exponential Mixture noises with various β. It shows that the Gaussian (i.e., $\beta = 2$ for General Normal, $\beta = 0$ for Laplace-Gaussian Mixture and Exponential Mixture) is the optimal noise in these β setting. We also observe the "crash" on Laplace-based distributions when p_A is small.

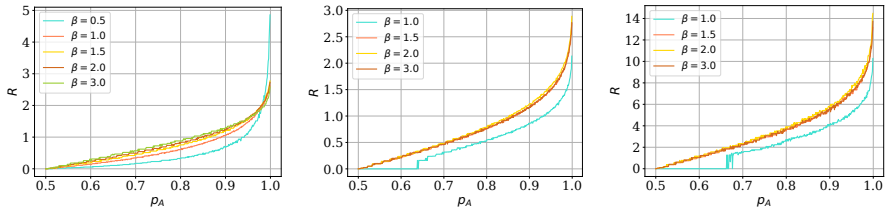
D.7 Certification on Non-Smoothed Classifier

Table 3. Certified accuracy on standard classifier.

radius R	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8
Yang's [17] vs. ℓ_1-norm	10.6	10.4	10.4	9.8	8.8	8.2	5.4	2.2	1.0
Ours vs. ℓ_1-norm	**98.8**	**47.0**	**22.4**	**17.8**	**13.8**	**10.2**	**7.0**	**3.8**	1.0
Cohen's [4] vs. ℓ_2-norm	10.6	10.4	10.4	9.6	8.8	8.2	5.6	2.2	1.2
Ours vs. ℓ_2-norm	**98.8**	**46.0**	**22.4**	**17.6**	**13.8**	**9.8**	**7.0**	**3.8**	1.2
Yang's [17] vs. ℓ_∞-norm (at $R/255$)	10.6	10.6	10.6	10.4	10.4	10.4	10.4	10.4	10.4
Ours vs. ℓ_∞-norm (at $R/255$)	**98.6**	**92.4**	**69.4**	**61.6**	**53.6**	**46.0**	**37.8**	**27.4**	**24.4**

Besides certifying inputs with the smoothed classifier, our input noise optimization (I-OPT) can certify input with a standard classifier without degrading the classifier accuracy on clean data (on the contrary, existing works have to trade off such accuracy for certified defenses).

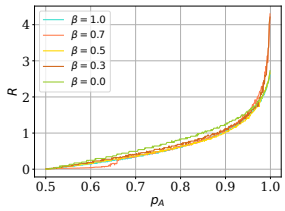
Specifically, since our I-OPT allows the noise for the input certification to be different from the noise used in training, a special case of the training noise is no noise ($\sigma = 0$). This means that we can certify a naturally-trained classifier



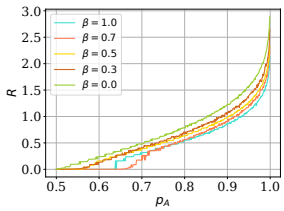
(a) General Normal vs. ℓ_1 perturbations

(b) General Normal vs. ℓ_2 perturbations

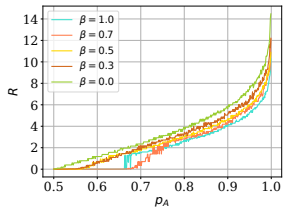
(c) General Normal vs. ℓ_∞ perturbations



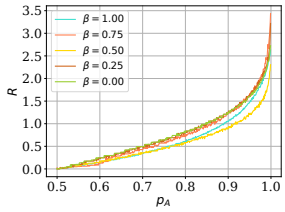
(d) Laplace-Gaussian Mixture vs. ℓ_∞ perturbations



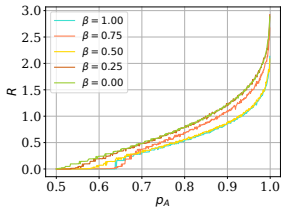
(e) Laplace-Gaussian Mixture vs. ℓ_∞ perturbations



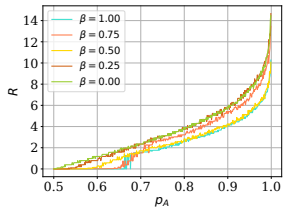
(f) Laplace-Gaussian Mixture vs. ℓ_∞ perturbations



(g) Exponential Mixture vs. ℓ_∞ perturbations



(h) Exponential Mixture vs. ℓ_∞ perturbations



(i) Exponential Mixture vs. ℓ_∞ perturbations

Fig. 5. p_A-R curves of General Normal, Laplace-Gaussian Mixture, and Exponential Mixture noise with a varying β.

(standard classifier). This provides an obvious benefit that the classifier can still execute normal classification on clean data with high accuracy since the standard classifier is trained without noise. Also, with I-OPT, we can tune the noise for the input to maintain the prediction accuracy. Thus, any classifier can be certifiably protected against perturbations without degrading the general performance on clean data.

To maintain the performance on standard classification, we add a condition while performing I-OPT:

$$g(x + \delta) = f(x) \tag{29}$$

We show this application on a standard ResNet110 classifier trained on CIFAR10 (see Table 3). For the baselines, we use Gaussian noise ($\sigma = 0.35$) and its corresponding theoretical radius [17, 4] for certification. Our method uses I-OPT

with General Normal noise and initializes it with the same σ. While approximating the certified radius with UniCR, we generate $4,000$ samples with the Monte Carlo method on CIFAR10.

The table shows that over 98.6% of the inputs are certified by our method with a radius $R > 0$. This means that over 98.6% of the samples are certifiably protected while only 10.6% of inputs are certified by the baselines, which is nearly the accuracy by random guessing. This significant improvement emerges since the I-OPT could optimize the noise PDF for each input even though the classifier is not trained with noise (non-smoothed classifier). Although the certified radii are low compared to smoothly-trained classifiers, it provides a certifiable protection on perturbed data while maintaining the high accuracy for classifying clean data.

E Visual Examples of I-OPT

We present some examples of I-OPT on the ImageNet dataset against ℓ_1, ℓ_2 and ℓ_∞ perturbations, respectively (see Figure 6). In the first case (ℓ_1 perturbations), without executing the I-OPT, our UniCR certifies the input with a radius $R = 1.24$. Our I-OPT optimizes the distribution as the right-most figure shows, then the certified radius is improved to 1.48 with our UniCR. Similarly, in the rest cases, we show I-OPT can improve the certified radius significantly by optimizing the noise distribution. Especially, we improve the radius from 0.35 to 1.30 in the second case.

F Discussions

F.1 Universal Certified Robustness

It might be impractical to make a universal framework satisfy all the theoretical conditions w.r.t. all ℓ_p perturbations, especially p can be any positive real number. Thus, we admit that UniCR may not strictly satisfy certified robustness all the time due to the approximated optimization. However, extensive empirical results confirm that our derived radii highly approximate the theoretical certified radii against different ℓ_p perturbations. In addition, the defense performance against real attacks also illustrate that our method is as reliable as different theoretical certified radii. We believe that with the negligible error in practice, UniCR can be deployed as a universal framework to significantly ease the process of achieving certified robustness in different scenarios.

F.2 Certifying Perturbed Data with Randomized Smoothing

Randomized smoothing usually assumes that the input is clean and empirical defenses [11, 7] are not applied, if the input data is perturbed before certification, then certification in I-OPT might be inaccurate. Indeed, the certification in traditional randomized smoothing (e.g., [4]) methods also depend on the inputs (since p_A is different for different inputs), they might be inaccurate if the

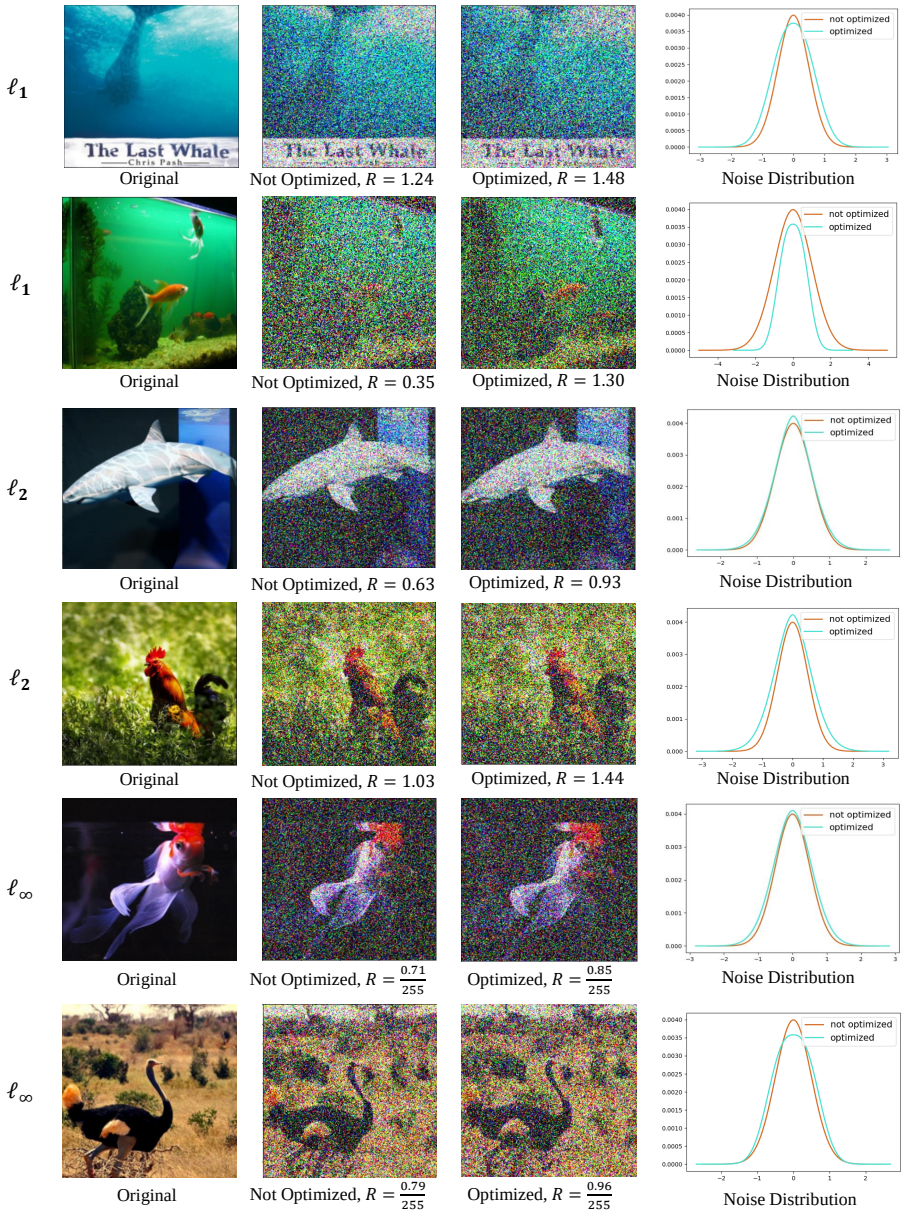


Fig. 6. Example images of applying I-OPT (based on UniCR) for smoothed classifier against different ℓ_p perturbations on the ImageNet dataset. From the left to right, the first figure shows the original image. The second and third figures show the smoothed image without I-OPT and with I-OPT, respectively. The fourth figure shows the corresponding distributions before/after I-OPT.

input data is perturbed, either. Thus, randomized smoothing focuses on certifying clean inputs rather than correcting perturbed inputs. We will study this interesting problem on certifying both clean and perturbed inputs in the future.

F.3 Can existing methods adopt noise optimization?

A question here is that if the noise optimization can improve the certified radius, can the theoretical methods provide personalized randomization for each input? The personalized randomization is actually not adaptable in the theoretical methods since they cannot automatically derive the certified radius for different noise distributions, especially for uncommon distributions, e.g., $e^{-|x/0.5|^{1.5}}$. Instead, our UniCR can automatically derive the certified radius for any distribution within the continuous parameter space.

F.4 Extensions

We evaluate our UniCR on the image classification. Indeed, our UniCR is a general method that can be directly applied to other tasks, e.g., video classification [10, 16], graph learning (e.g., node/graph classification [14] and community detection [8]), and natural language processing [9].

References

1. Andriushchenko, M., Croce, F., Flammarion, N., Hein, M.: Square attack: a query-efficient black-box adversarial attack via random search. In: European Conference on Computer Vision. pp. 484–501. Springer (2020)
2. Chen, J., Jordan, M.I., Wainwright, M.J.: Hopskipjumpattack: A query-efficient decision-based attack. In: 2020 IEEE Symposium on Security and Privacy (2020)
3. Co, K.T., Muñoz-González, L., de Maupeou, S., Lupu, E.C.: Procedural noise adversarial examples for black-box attacks on deep convolutional networks. In: ACM SIGSAC conference on computer and communications security (2019)
4. Cohen, J., Rosenfeld, E., Kolter, Z.: Certified adversarial robustness via randomized smoothing. In: International Conference on Machine Learning (2019)
5. Croce, F., Hein, M.: Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In: International conference on machine learning. pp. 2206–2216. PMLR (2020)
6. Dvoretzky, A., Kiefer, J., Wolfowitz, J.: Asymptotic minimax character of the sample distribution function and of the classical multinomial estimator. The Annals of Mathematical Statistics pp. 642–669 (1956)
7. Hong, H., Hong, Y., Kong, Y.: An eye for an eye: Defending against gradient-based attacks with gradients. arXiv preprint arXiv:2202.01117 (2022)
8. Jia, J., Wang, B., Cao, X., Gong, N.Z.: Certified robustness of community detection against adversarial structural perturbation via randomized smoothing. In: Proceedings of The Web Conference 2020. pp. 2718–2724 (2020)
9. Jia, R., Raghunathan, A., Göksel, K., Liang, P.: Certified robustness to adversarial word substitutions. In: EMNLP/IJCNLP (2019)
10. Li, S., Neupane, A., Paul, S., Song, C., Krishnamurthy, S.V., Roy-Chowdhury, A.K., Swami, A.: Stealthy adversarial perturbations against real-time video classification systems. In: NDSS. The Internet Society (2019)
11. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: Towards deep learning models resistant to adversarial attacks. In: International Conference on Learning Representations (2018)
12. Mohammady, M., Xie, S., Hong, Y., Zhang, M., Wang, L., Pourzandi, M., Debbabi, M.: R2dp: A universal and automated approach to optimizing the randomization mechanisms of differential privacy for utility metrics with no known optimal distributions. In: Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security. pp. 677–696 (2020)
13. Teng, J., Lee, G.H., Yuan, Y.: ℓ_1 adversarial robustness certificates: a randomized smoothing approach (2020)
14. Wang, B., Jia, J., Cao, X., Gong, N.Z.: Certified robustness of graph neural networks against adversarial structural perturbation. In: KDD. ACM (2021)
15. Wong, E., Schmidt, F., Kolter, Z.: Wasserstein adversarial examples via projected sinkhorn iterations. In: International Conference on Machine Learning (2019)
16. Xie, S., Wang, H., Kong, Y., Hong, Y.: Universal 3-dimensional perturbations for black-box attacks on video recognition systems. In: In Proceedings of the 43rd IEEE Symposium on Security and Privacy (Oakland'22) (2022)
17. Yang, G., Duan, T., Hu, J.E., Salman, H., Razenshteyn, I., Li, J.: Randomized smoothing of all shapes and sizes. In: International Conference on Machine Learning. pp. 10693–10705. PMLR (2020)
18. Zhang, D., Ye, M., Gong, C., Zhu, Z., Liu, Q.: Black-box certification with randomized smoothing: A functional optimization based framework (2020)