AIRI

Certified Robustness, High Dimensions and CV

Aleksandr Petiushko

AIRI (Artificial Intelligence Research Institute)
MSU (Lomonosov Moscow State University)

Certified Robustness - definitions

Suppose our NN function f(x) is the **classifier** to K classes: $f: R^d \to Y, Y = \{1, \dots, K\}$

- Usually we have NN $h(x): R^d \to R^K$, and $f(x) = argmax_i h(x)_i$
- **Deterministic approach**: we want to find the class of perturbation $S(x, f)$ so as the classifier's output doesn't not change, or more formally:

$$f(x + \delta) = f(x)\ \forall \delta \in S(x, f)$$

- **Probabilistic approach**: having the probability of robustness P, find the class of input perturbations $S(x, f, P)$ s.t.:

$$Prob_{\delta \in S(x,f,P)}\big(f(x + \delta) = f(x)\big) = P$$

If NN f(x) is the **regressor**: $f: R^d \to R$

- Having the upper and lower bounds on the output perturbation, find the class of input perturbations $S(x, f, f_{low}, f_{up})$:

$$f(x) - f_{low} \leq f(x + \delta) \leq f(x) + f_{up}, \forall \delta \in S(x, f, f_{low}, f_{up})$$

Certified Robustness – definitions (2)

Also, **inverse tasks** could be considered

If NN $f(x)$ is the **classifier** to K classes: $f: R^d \to Y, Y = \{1, \dots, K\}$

- **Probabilistic approach**: we want to measure the probability of retaining the classifier output under some class of input perturbations S:

$$Prob_{\delta \in S}(f(x+\delta) = f(x))$$

If NN $f(x)$ is the **regressor**: $f: R^d \to R$

- We want to find the upper and lower bounds of the output perturbation under some class of input perturbations S in the analytical form:

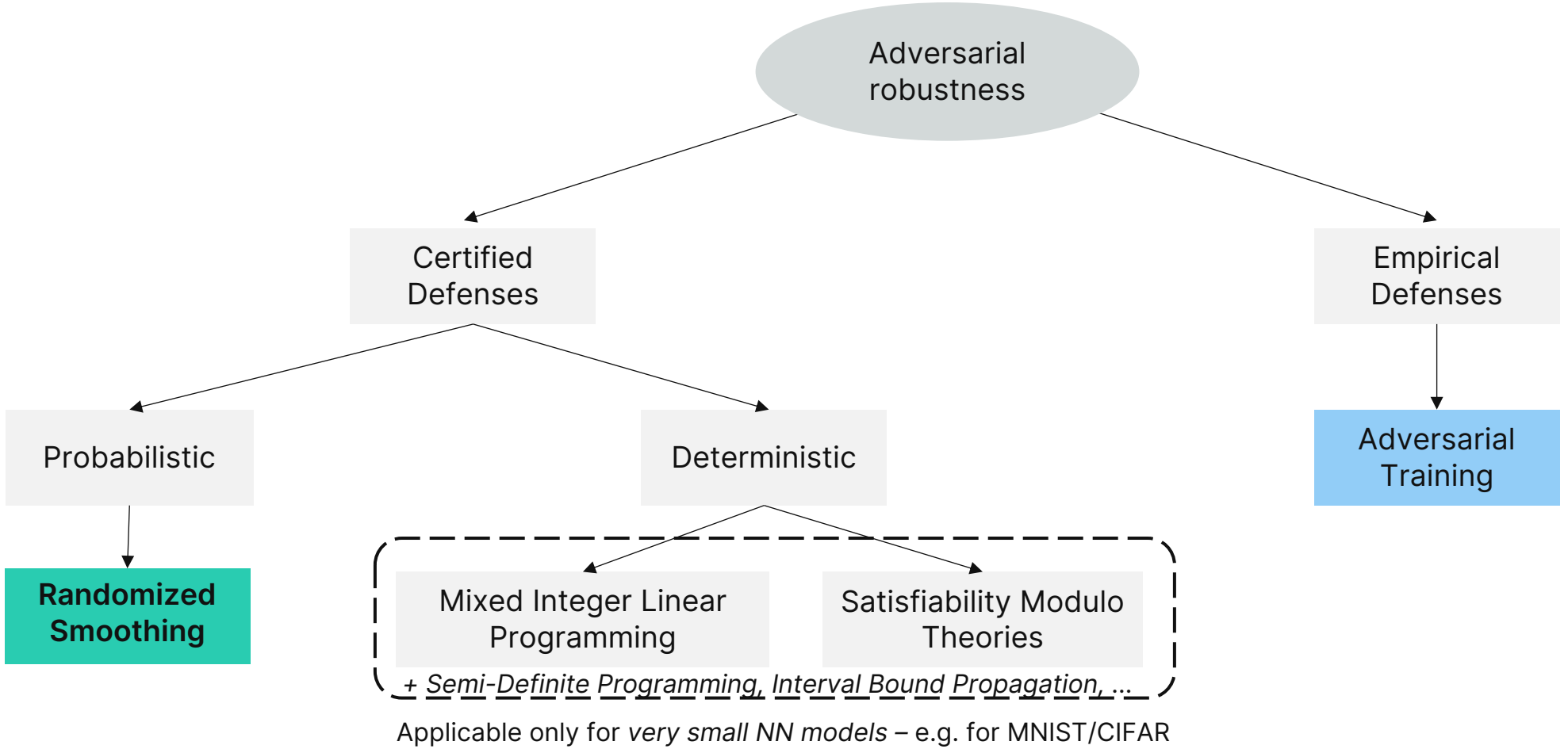
$$f(x) - f_{low}(f, x, S) \leq f(x+\delta) \leq f(x) + f_{up}(f, x, S), \forall \delta \in S$$

Can be measured by analyzing the output of $f(x+\delta)$

Certified Robustness and Lipschitz Function

- Neural Net output is $h: R^d \to R^K$, and the classifier itself is $f: R^d \to Y, Y = \{1, \dots, K\}$, where $f(x) = argmax_{i \in Y} h(x)_i$
- Consider binary case ($K = 2$), and probabilistic output: $h(x)_1 + h(x)_2 = 1, h(x)_i \geq 0$
- **Lipschitz function** $f: R^d \to R$ with a Lipschitz constant L: $\forall x_1, x_2$ it is true that
- $|f(x_1) - f(x_2)| \leq L||x_1 - x_2||$
- **Local Lipschitz function** f with a Lipschitz constant $L(x_0)$: $\forall x \in S(x_0)$ it is true that
- $|f(x_0) - f(x)| \leq L(x_0)||x_0 - x||$
- Let $j = argmax_{i \in Y} h(x_0)_i$, and $h(x_0)_j - h(x_0)_{i \neq j} \geq \epsilon \; \forall i \neq j$
- Let $h(x_0)_j$ - local Lipschitz function with a Lipschitz constant $L(x_0)$
- Then if $S(x_0) = \{x: \left||x_0 - x|\right| \leq \frac{\epsilon}{2L(x_0)}\}$, we have $\left|h(x_0)_j - h(x)_j\right| \leq L(x_0) \frac{\epsilon}{2L(x_0)} = \frac{\epsilon}{2}$
- As a consequence we'll have $j = argmax_{i \in Y} h(x)_i$, and $f(x) = f(x_0) = j$ in the vicinity $S(x_0) = \{x: \left||x_0 - x|\right| \leq \frac{\epsilon}{2L(x_0)}\}$, and certified robustness!
- Problems:
  - True certified radius can be much bigger than Lipschitz vicinity $S(x_0)$
  - Hard to provide the adequate Lipschitz constant for a DNN

Adversarial Robustness



Robustness: Empirical VS Certified

Empirical

Upper bound on the true robustness accuracy

But only *until* the new *stronger attack* appears

Certified

Lower bound on the true robustness accuracy

It is what has been *theoretically proven*, and no one attack can beat it

Empirical robustness: adversarial training

Main idea:

Train on the *most hard examples* using some class of perturbations $S(= \Delta)$ around training examples

$$\min_{\theta} \mathbf{E}_{x,y}[\text{Loss}(f_\theta(x), y)] \Longrightarrow \min_{\theta} \mathbf{E}_{x,y}\left[\max_{\delta \in \Delta} \text{Loss}\left(f_\theta(x+\delta), y\right)\right]$$

Drawbacks:

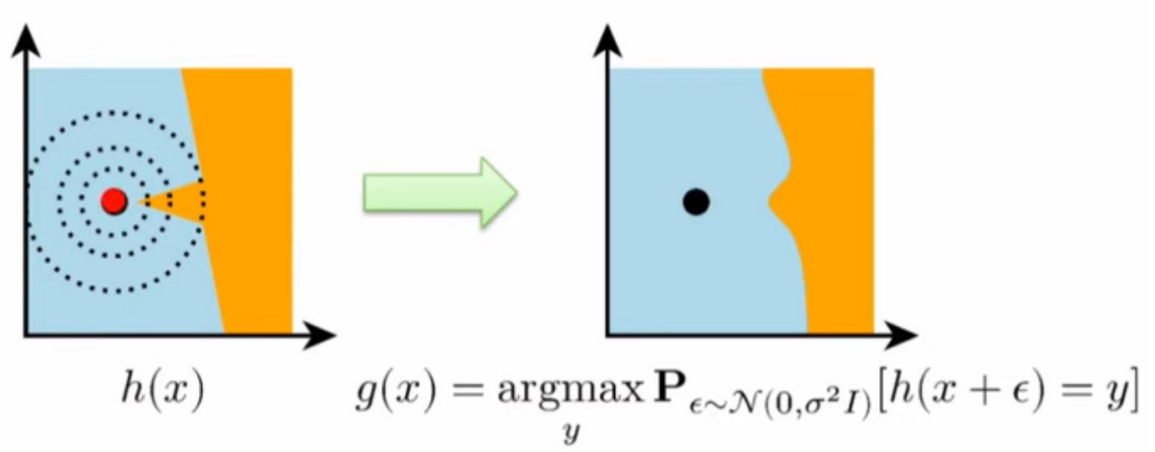
- Quite *inefficient training* (longer than usual because of finding of hard examples for every training sample for every iteration)
- The *accuracy on clean samples is lower* than for usual training

Adversarial examples and boundary curvature

Very **curved boundary** leads to *adversarial examples* looking very similar to ones near the classification boundary

So let's **diminish** this curvature **spike** influence!

Different approaches exist e.g. by *Lecuyer et al.*[1] *and Li et al.*[2], but the most famous one is by *Cohen et al.*



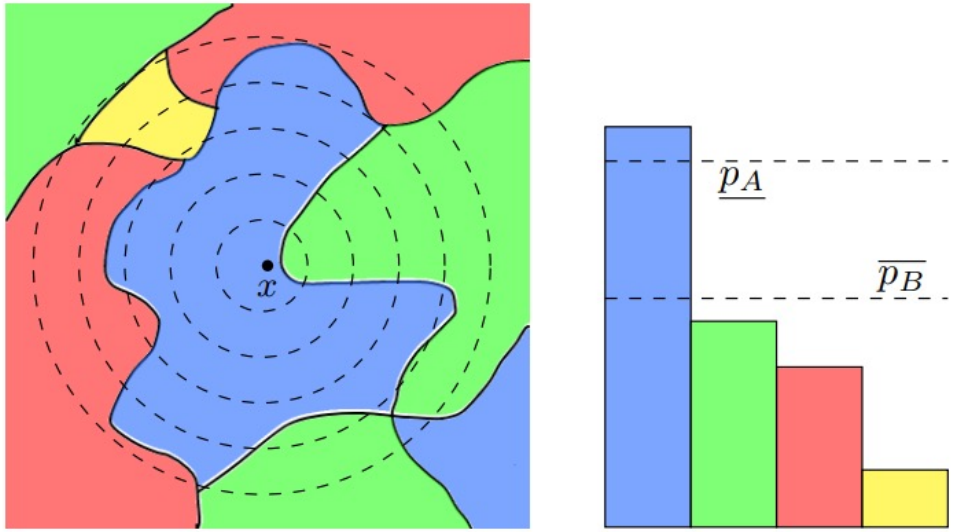
[1] Lecuyer, Mathias, et al. "Certified robustness to adversarial examples with differential privacy."
[2] Li, Bai, et al. "Certified Adversarial Robustness with Additive Noise."

Randomized Smoothing[1]

Main idea:

Let's use another definition of classifier!
New classifier (in fact, sort of TTA):

$$g(x) = \underset{c \in Y}{\operatorname{argmax}}\, P(f(x+\varepsilon) = c), \varepsilon \sim N(0, \sigma^2 I)$$



The main robustness result:

- If $f(x)$ classifier is robust under Gaussian noise,
- Then $g(x)$ classifier is robust under **ANY** noise

Theorem 1. *Let $f : \mathbb{R}^d \to \mathcal{Y}$ be any deterministic or random function, and let $\varepsilon \sim \mathcal{N}(0, \sigma^2 I)$. Let g be defined as in (1). Suppose $c_A \in \mathcal{Y}$ and $\underline{p_A}, \overline{p_B} \in [0,1]$ satisfy:*

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

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

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

The radius R in **Theorem 1** is tight: with the bigger radius there exists an adversarial example

Theorem 2. *Assume $\underline{p_A} + \overline{p_B} \leq 1$. For any perturbation δ with $\|\delta\|_2 > R$, there exists a base classifier f consistent with the class probabilities (2) for which $g(x+\delta) \neq c_A$.*

[1] Cohen, Jeremy M., Elan Rosenfeld, and J. Zico Kolter.
"Certified adversarial robustness via randomized smoothing."

Randomized smoothness: results

The authors propose the procedure to return the radius R and output class c based on input x and the deviance of noise σ

This procedure can even avoid (ABSTAIN) to provide the answer with some probability α

To certify the classifiers, authors **trained the base models with Gaussian noise from $\boldsymbol{N(0, \sigma^2 I)}$** – in fact, to make the classifier $f(x)$ more robust to Gaussian noise

Trained models are compared using "**approximate certified accuracy**":

- For each test radius $\delta = r$ the fraction of examples is returned on which procedure CERTIFY:
  - Provides the answer
  - Returns the correct class
  - Returns a radius R so as $r \leq R$

Also when estimating the $g(x)$ authors run Monte Carlo N times

Randomized smoothness: results on ImageNet

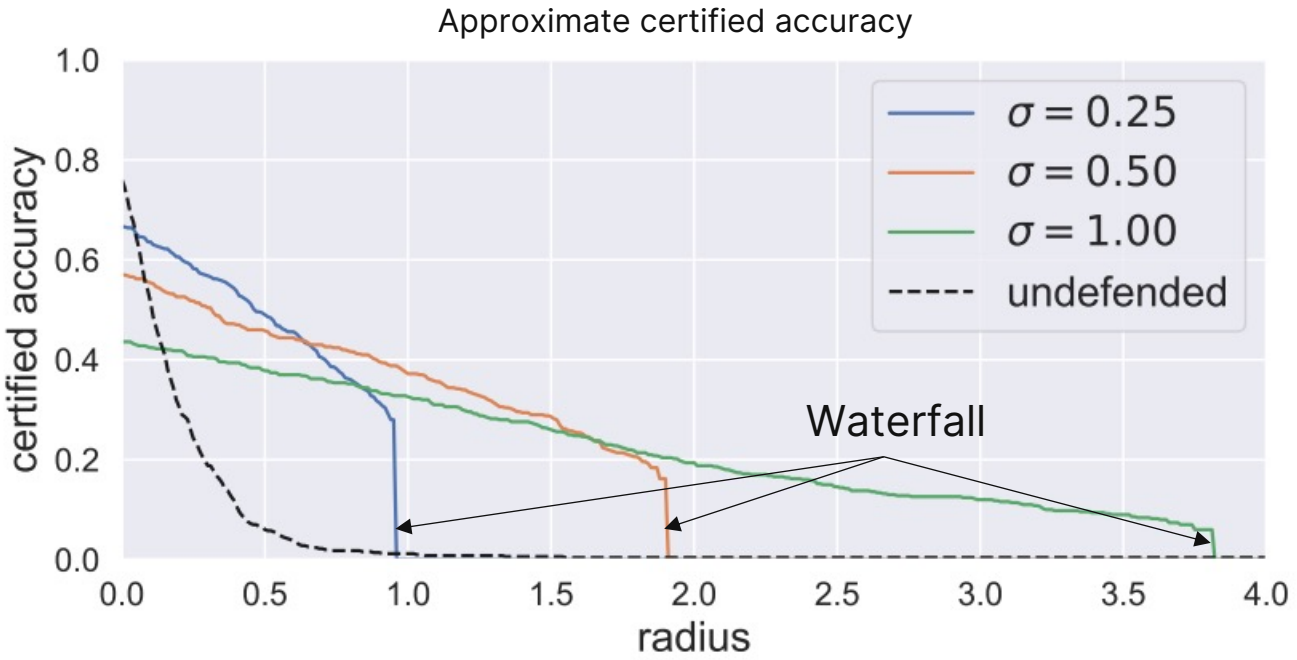


Table 1. Approximate certified accuracy on ImageNet. Each row shows a radius r, the best hyperparameter σ for that radius, the approximate certified accuracy at radius r of the corresponding smoothed classifier, and the standard accuracy of the corresponding smoothed classifier. To give a sense of scale, a perturbation with ℓ_2 radius 1.0 could change one pixel by 255, ten pixels by 80, 100 pixels by 25, or 1000 pixels by 8. Random guessing on ImageNet would attain 0.1% accuracy.

ℓ_2 RADIUS	BEST σ	CERT. ACC (%)	STD. ACC(%)
0.5	0.25	49	67
1.0	0.50	37	57
2.0	0.50	19	57
3.0	1.00	12	44

Waterfall just because the trained model is robust usually under some $r \leq R$

Improved training for smoothed classifier

I. Instead of simple augmenting the training example with Gaussian noise,
let's do in fact **adversarial training**[1] using attacks on $g(x)$!

That means not $\sum log$, but $log \sum$

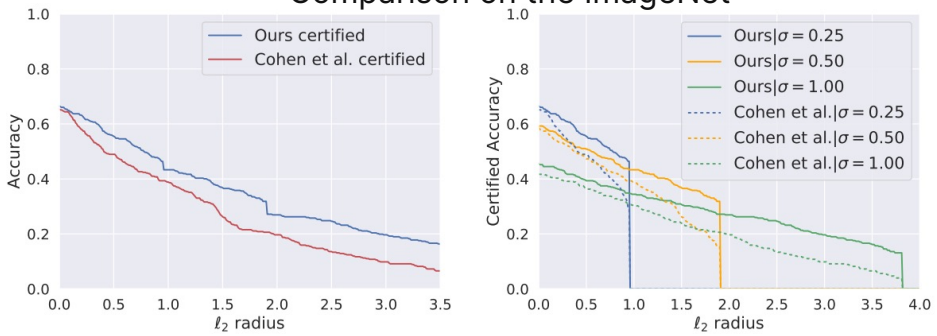
$$\hat{x} = \underset{\|x'-x\|_2 \leq \epsilon}{\arg\max}\ \ell_{\mathrm{CE}}(G(x'), y)$$

$$= \underset{\|x'-x\|_2 \leq \epsilon}{\arg\max} \left(-\log \underset{\delta \sim \mathcal{N}(0,\sigma^2 I)}{\mathbb{E}} \left[(F(x'+\delta))_y \right] \right)$$

$$\nabla_{x'} J(x') = \nabla_{x'} \left(-\log \underset{\delta \sim \mathcal{N}(0,\sigma^2 I)}{\mathbb{E}} [F(x'+\delta)_y] \right)$$

$$\nabla_{x'} J(x') \approx \nabla_{x'} \left(-\log \left(\frac{1}{m} \sum_{i=1}^{m} F(x'+\delta_i)_y \right) \right)$$

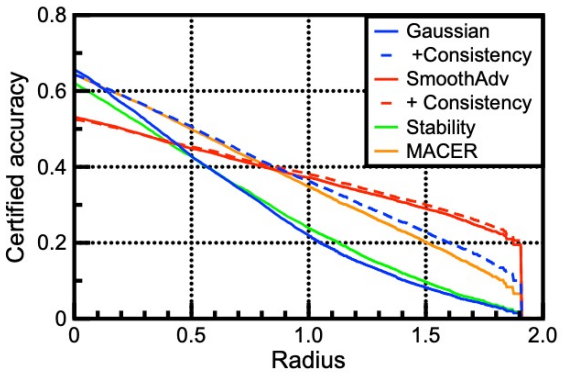
Comparison on the ImageNet



II. Another approach is
to force *similarity* between *smoothe*d and *current* predictions as well as minimizing the entropy of smoothed output via **consistency regularization**[2] loss term

$$\hat{F}(x) := \mathbb{E}[F(x+\delta)]$$

$$L^{\mathtt{con}} := \lambda \cdot \mathbb{E}_\delta \left[\mathrm{KL}(\hat{F}(x) || F(x+\delta)) \right] + \eta \cdot \mathrm{H}(\hat{F}(x))$$



(b) $\sigma = 0.50$

[1] Salman, Hadi, et al. "Provably robust deep learning via adversarially trained smoothed classifiers.»
[2] Jeong, Jongheon, and Jinwoo Shin. "Consistency regularization for certified robustness of smoothed classifiers."

BlackBox for Randomized Smoothing[1]

What if we **cannot change pretrained classifier**, but want to increase its certified robustness?

Let's train **denoiser** D used after we add Gaussian noise!

And then simply apply majority rule.

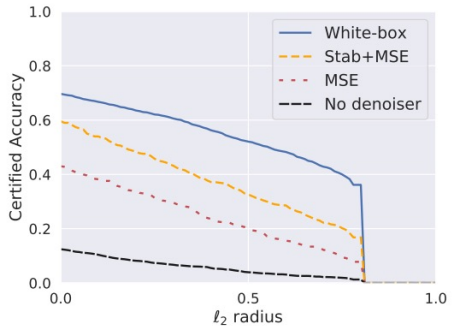
Denoiser: trained with two losses for every Gaussian σ

- MSE
- Stability (CE loss)

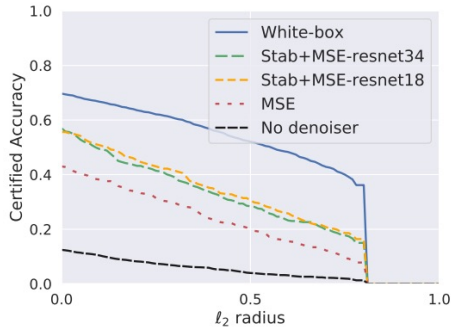
Table 1. Certified top-1 accuracy of ResNet-50 on **ImageNet** at various ℓ_2 radii (Standard accuracy is in parenthesis).

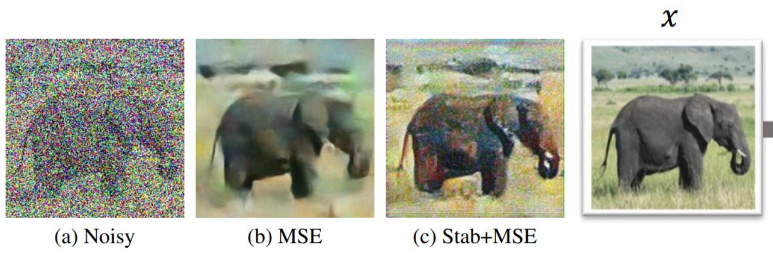
ℓ_2 RADIUS (IMAGENET)	0.25	0.5	0.75	1.0	1.25	1.5
WHITE-BOX SMOOTHING (COHEN ET AL., 2019) (%)	$^{(70)}62$	$^{(70)}52$	$^{(62)}45$	$^{(62)}39$	$^{(62)}34$	$^{(50)}29$
NO DENOISER (BASELINE) (%)	$^{(49)}32$	$^{(12)}4$	$^{(12)}2$	$^{(0)}0$	$^{(0)}0$	$^{(0)}0$
BLACK-BOX SMOOTHING (QUERY ACCESS) (%)	$^{(69)}48$	$^{(56)}31$	$^{(56)}19$	$^{(34)}12$	$^{(34)}7$	$^{(30)}4$
BLACK-BOX SMOOTHING (FULL ACCESS) (%)	$^{(67)}50$	$^{(60)}33$	$^{(60)}20$	$^{(38)}14$	$^{(38)}11$	$^{(38)}6$

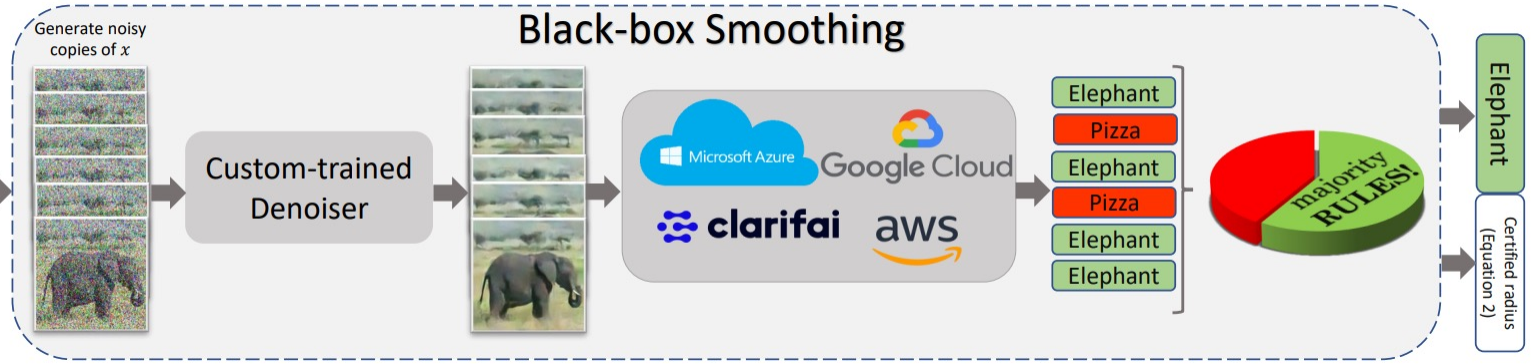
ImageNet, ResNet-50, Full-access



Query-access







[1] Salman, Hadi, et al.
"Black-box Smoothing: A Provable Defense for Pretrained Classifiers."

Randomized Smoothing for Regression[1]

The solution was proposed even for **Regression** problem, where the logit / probability of 2-class classification model can be assumed as the continuous output:

$$f : \mathbb{R}^d \to \mathbb{R}$$

$$g(x) = \mathbb{E}[f(x + G)], \quad \text{where } G \sim N(0, \sigma^2 I)$$

Corollary 1. *[30] For any* $f : \mathbb{R}^d \to [l, u]$*, the map* $\eta(x) = \sigma \cdot \Phi^{-1}(\frac{g(x)-l}{u-l})$ *is 1-Lipschitz, implying*

$$l + (u - l) \cdot \Phi\left(\frac{\eta(x) - \|\delta\|_2}{\sigma}\right) \leq g(x + \delta) \leq l + (u - l) \cdot \Phi\left(\frac{\eta(x) + \|\delta\|_2}{\sigma}\right) \qquad (2)$$

[1] Chiang, Ping-yeh, et al.
"Detection as Regression: Certified Object Detection by Median Smoothing."

Things to note [intermediate takeaway]

Certification is only for much smaller regions than humans can do

Certified robustness is better than empirical adversarial training in certification, but worse than clean performance (and too much time to train)

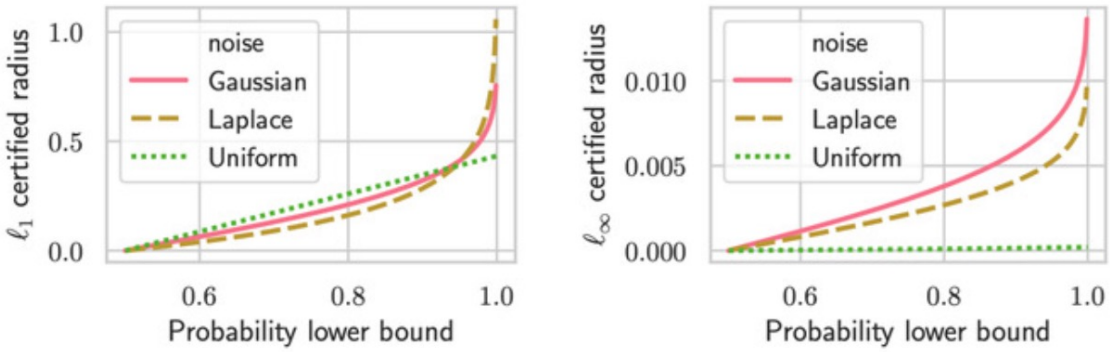
Using l_p-balls is neither necessary nor sufficient for perceptual robustness

Other types of randomized smoothing could be taking into account: e.g. *Uniform*[1] or *Laplacian*[2]

Randomized smoothing requires multiple inferences ☹

BTW some note about physical nature of l_p-balls:

- l_2: corresponds to the power of signals
- l_1: corresponds to the pixel mass
- l_∞: corresponds to the noise in camera sensors
- l_0: corresponds to the practical *patch* robustness



[1] Lee, Guang-He, et al. "Tight certificates of adversarial robustness for randomly smoothed classifiers."
[2] Teng, Jiaye, et al. "l_1 Adversarial Robustness Certificates: a Randomized Smoothing Approach"

High Dimension case for randomized smoothing

p = 1 and p = 2 are the **special cases**[1] of l_p:

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

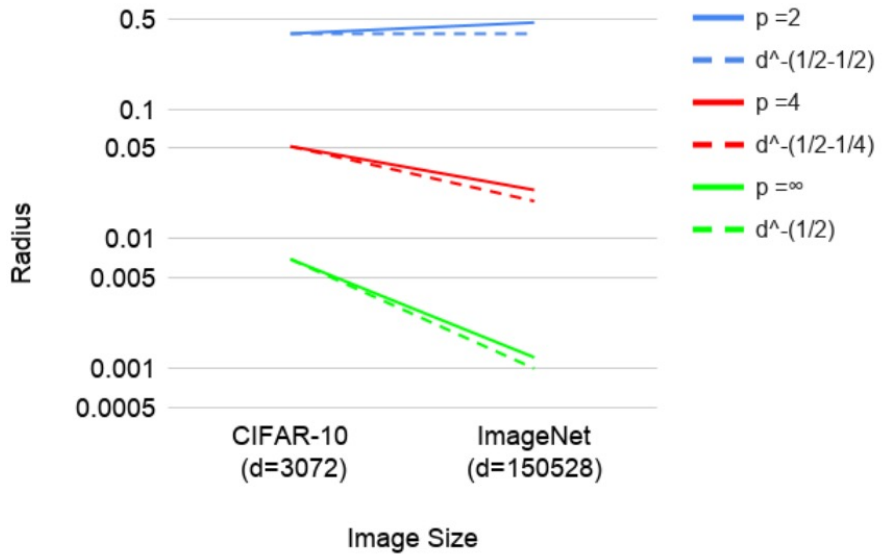
Unfortunately, these are **only** examples of **non-decreasing** with **input dimension *d***.

For $p \geq 2$, the certified radius[2] is decreasing with dim *d*:

$$r_p = \frac{\sigma}{2d^{\frac{1}{2}-\frac{1}{p}}}\left(\Phi^{-1}\left(p_1(x)\right) - \Phi^{-1}\left(p_2(x)\right)\right)$$

And the most important case in CV, $p = \infty$, means

$$R \sim {}^{1}/_{\sqrt{d}}$$



[1] Yang G. et al .“Randomized smoothing of all shapes and sizes”
[2] Kumar A. et al. “Curse of dimensionality on randomized smoothing for certifiable robustness”

High Dimension case in CV

Any **semantic-meaningful** perturbation in **CV** leads to **high l_∞-perturbation**, and the dimension of an image ($d = H \times W$) is very high
⇒ **no any practical certified radius**

E.g., for semantic-specific transformations like **contrast** and **brightness error** is **higher** than on clean images up to **50-60%** on *Common Corruptions*[1] on ImageNet

The same is true for **safety-critical** applications like **autonomous driving**[2]

Network	Error	Bright	Contrast
AlexNet	43.5	100	100
SqueezeNet	41.8	97	98
VGG-11	31.0	75	86
VGG-19	27.6	68	80
VGG-19+BN	25.8	61	74
ResNet-18	30.2	69	78
ResNet-50	23.9	57	71

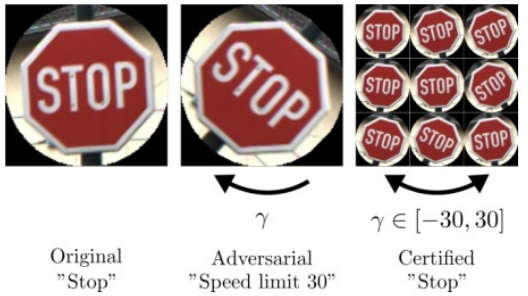
Transformation	#err
Brightness	97
Contrast	31

[1] Hendrycks, Dan, and Thomas Dietterich. "Benchmarking neural network robustness to common corruptions and perturbations."
[2] Tian, Yuchi, et al. "Deeptest: Automated testing of deep-neural-network-driven autonomous cars."

Certified robustness for Semantic perturbations (1)

Let's certify semantic perturbations!

- In fact, **rotations** and **translations** are studied $\psi_\beta : \mathbb{R}^n \to \mathbb{R}^n$
- Smoothed classifier: $g(\boldsymbol{x}) = \arg\max_c \mathbb{P}_{\beta \sim \mathcal{N}(0,\sigma^2\mathbb{1})}(f \circ \psi_\beta(\boldsymbol{x}) = c)$
- Also **interpolation** procedure is taken into account because after rotation we need to interpolate anyway



Theorem 4.2. *Let $\boldsymbol{x} \in \mathbb{R}^n$, $f : \mathbb{R}^m \to \mathcal{Y}$ be a classifier and $\psi_\beta : \mathbb{R}^n \to \mathbb{R}^m$ be a composable transformation as above. If*

$$\mathbb{P}_\beta(f \circ \psi_\beta(x) = c_A) = p_A \geq \underline{p_A} \geq \overline{p_B} \geq p_B = \max_{c_B \neq c_A} \mathbb{P}_\beta(f \circ \psi_\beta(x) = c_B),$$

then $g \circ \psi_\gamma(\boldsymbol{x}) = c_A$ for all γ satisfying

$$\|\gamma\|_2 < \tfrac{\sigma}{2}(\Phi^{-1}(\underline{p_A}) - \Phi^{-1}(\overline{p_B})) =: r_\gamma.$$

Rotation						r_γ percentile		
Dataset	$\mathcal{I}$	σ_γ	α_γ	f Acc.	g Acc.	25$^{\text{th}}$	50$^{\text{th}}$	75$^{\text{th}}$
ImageNet	bil.	10	0.001	0.39	0.29	10.81	10.81	10.81
ImageNet	bil.	10	0.001	0.39	0.29	18.29	18.29	18.29
ImageNet	bil.	30	0.001	0.39	0.28	9.09	16.59	28.60
ImageNet	bil.	30	0.001	0.39	0.28	20.22	25.36	30†
ImageNet	bic.	10	0.001	0.39	0.29	10.40	10.40	10.40
ImageNet	bic.	30	0.001	0.39	0.27	9.33	17.00	28.74
ImageNet	near.	10	0.001	0.39	0.29	9.62	9.62	9.62
ImageNet	near.	30	0.001	0.39	0.26	7.38	16.63	27.72

Translation						r_γ percentile		
Dataset	$\mathcal{I}$	σ_γ	α_γ	f Acc.	g Acc.	25$^{\text{th}}$	50$^{\text{th}}$	75$^{\text{th}}$
ImageNet	bil.	50	0.001	0.48	0.36	2.4%	2.4%	2.4%
ImageNet	bic.	50	0.001	0.48	0.36	2.4%	2.4%	2.4%

[1] Fischer, Marc, Maximilian Baader, and Martin Vechev.
"Certification of Semantic Perturbations via Randomized Smoothing."

Certified robustness for Semantic perturbations (2)

Further development of the semantic-specific transformations:

- More rigorous approach – TSS[1] - taking into account different types of perturbations and interpolation errors

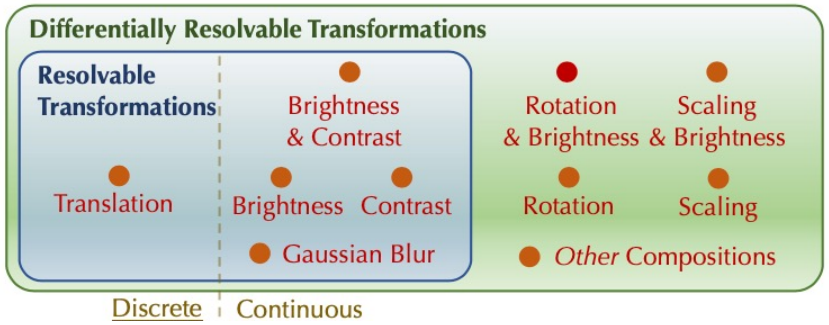
DEFINITION 2 (RESOLVABLE TRANSFORM). *A transformation* $\phi : \mathcal{X} \times \mathcal{Z} \to \mathcal{X}$ *is called resolvable if for any* $\alpha \in \mathcal{Z}$ *there exists a resolving function* $\gamma_\alpha : \mathcal{Z} \to \mathcal{Z}$ *that is injective, continuously differentiable, has non-vanishing Jacobian and for which*

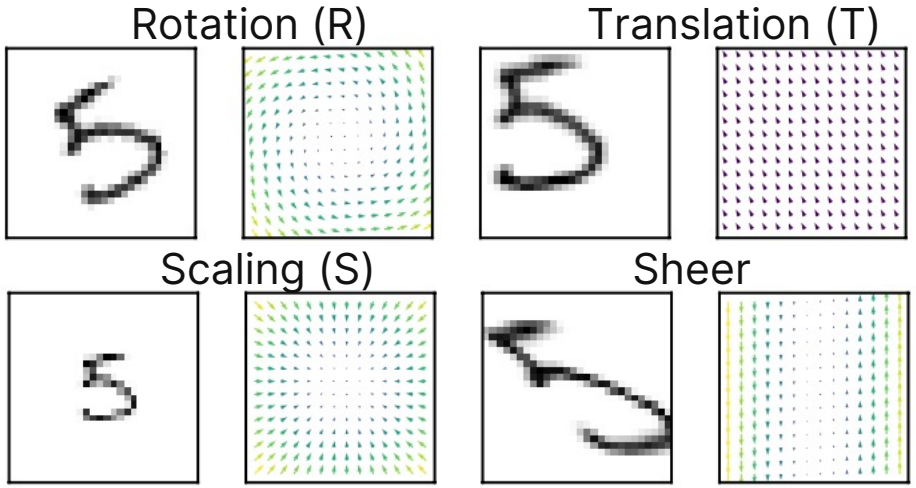
$$\phi(\phi(x, \alpha), \beta) = \phi(x, \gamma_\alpha(\beta)) \quad x \in \mathcal{X}, \beta \in \mathcal{Z}. \tag{7}$$

Furthermore, we say that ϕ *is additive, if* $\gamma_\alpha(\beta) = \alpha + \beta$.

- Approach – DeformRS[2] - based of Vector Fields

Certification	ImageNet R(10°)	ImageNet S (15%)	ImageNet T($\|\psi\|_2 \leq 5$)
Fischer [9]	17.25[(e)]	-	-
TTS [10]	33.00	31.00	**63.30**
DEFORMRS-PAR	**39.00**	**42.80**	48.20





[1] Li, Linyi, et al. "Tss: Transformation-specific smoothing for robustness certification."
[2] Alfarra, Motasem, et al. "DeformRS: Certifying Input Deformations with Randomized Smoothing."

Semantic perturbations for multiplicative parameters[1]

All research is concentrated on *additive* perturbations

We decided to investigate the **multiplicative** parameters (e.g., *gamma correction* $G_\gamma(x) = x^\gamma$ in CV):

Definition 3.1. *A parameterized map* $\psi_\delta : X \to X, \quad \delta \in \mathcal{B} \subset \mathbb{R}^n$ *is called multiplicatively composable if*

$$(\psi_\delta \circ \psi_\theta)(x) = \psi_{(\delta \cdot \theta)}(x), \ \forall x \in X, \ \forall \delta, \theta \in \mathcal{B},$$

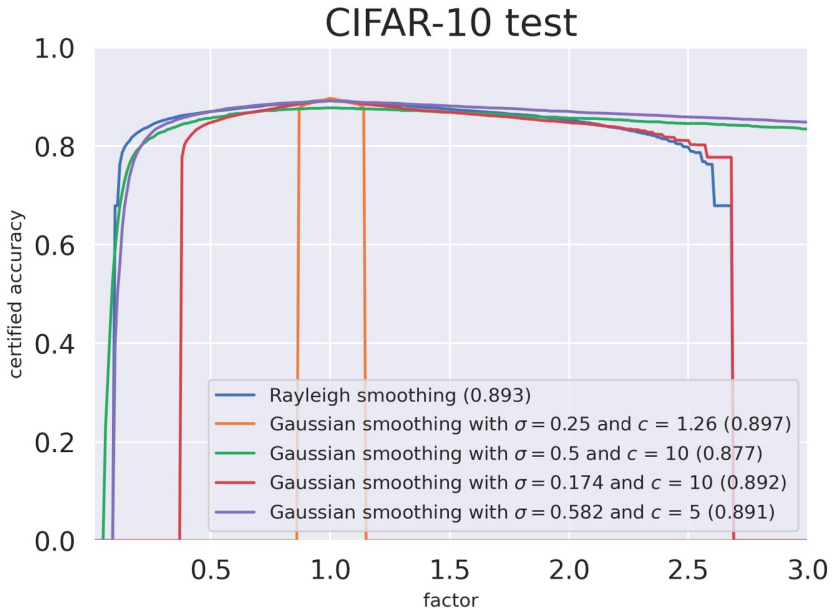
- Example: $G_\beta \circ G_\gamma(x) = (x^\gamma)^\beta = x^{\gamma \cdot \beta} = G_{\gamma \cdot \beta}(x)$
- To work under this limitation, the new type of smoothing distribution is needed (positive support, mean at 1):

Rayleigh distribution $p_\zeta(z) = \sigma^{-2} z e^{-z^2/(2\sigma^2)}, z \geq 0.$

We've managed to:

- Get analytical certification radius
- Better certification radius on factors less than 1

(a) $\gamma = 0.5$ (b) $\gamma = 1$ (c) $\gamma = 2$



[1] Muravev, Nikita, and Aleksandr Petiushko.
"Certified Robustness via Randomized Smoothing over Multiplicative Parameters."

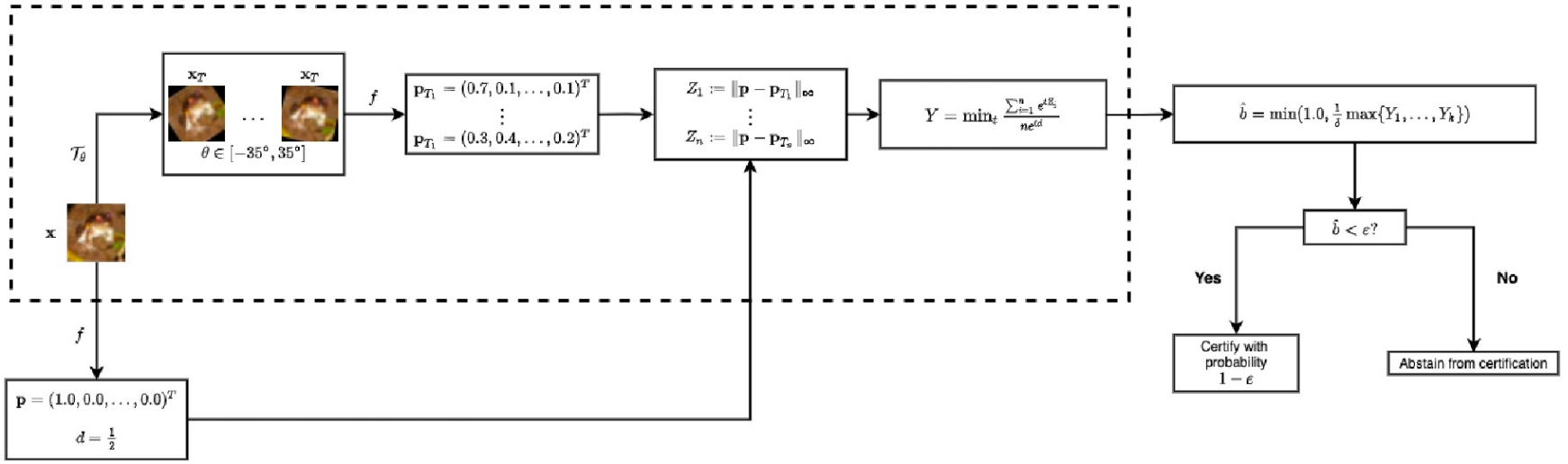
Inverse task: Randomized Smoothing for Probabilistic Certification[1]

We proposed the method to provide the statistically-grounded **estimations for the certification**, where perturbed **radius** is **already given**

Done based on implying *Chernoff-Cramer*[2] *inequality* (Markov's inequality Corollary)

Can be easily used for **any** semantical perturbation and **any** compositions

Dataset	Transform	Parameters	Training type	ERA	PCA(ε)		
					$\varepsilon = 10^{-10}$	$\varepsilon = 10^{-7}$	$\varepsilon = 10^{-4}$
	Brightness	$\theta_b \in [-40\%, 40\%]$	plain	58.4%	47.8%	51.6%	55.2%
			smoothing	65.0%	55.4%	59.4%	61.8%
	Contrast	$\theta_c \in [-40\%, 40\%]$	plain	91.6%	62.4%	67.0%	69.6%
			smoothing	88.0%	67.0%	72.8%	74.2%
	Rotation	$\theta_r \in [-10°, 10°]$	plain	73.4%	64.6%	69.0%	71.0%
			smoothing	72.4%	57.4%	63.6%	67.4%
	Contrast + Brightness	see Contrast & Brightness	plain	0.0%	0.0%	0.0%	0.0%
			smoothing	0.4%	0.0%	0.0%	0.0%
	Rotation + Brightness	see Rotation & Brightness	plain	22.6%	16.2%	20.6%	21.8%
			smoothing	30.4%	21.2%	24.6%	27.6%
	Scale + Brightness	see Scale & Brightness	plain	10.2%	10.4%	10.4%	10.4%
			smoothing	41.8%	40.6%	40.6%	40.6%



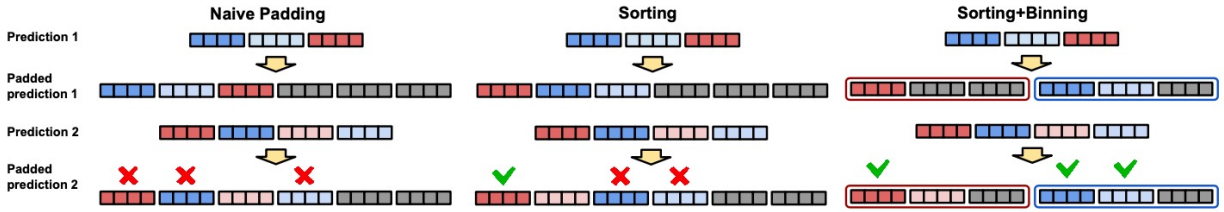
[1] Pautov, Mikhail, et al. "CC-Cert: A Probabilistic Approach to Certify General Robustness of Neural Networks."
[2] Boucheron, Stéphane, Gábor Lugosi, and Olivier Bousquet. "Concentration inequalities."

Randomized smoothing for Object Detection[1]

The approach treats certification for **Object Detection** as a **Regression problem** for **Black-box**[2] detectors

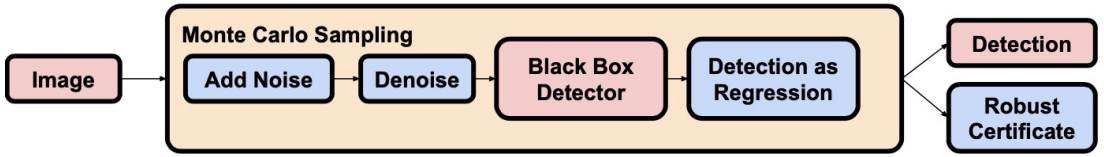
In order to certify multiple BB under different noise,
the **sorting** based on location +
binning based on label is needed

But the **certification** is still **out of practical use**



Architecture	Base Detector AP @ 50	Smoothed Detector AP @ 50	Smoothed Detector Certified AP @ 50
YOLOv3	48.66%	31.93%	4.21%
Mask RCNN	51.28%	30.53%	1.67%
Faster RCNN	50.47%	29.89%	1.54%

Figure 1: Samples of object detection certificates using the proposed method. Dotted lines represent the farthest a bounding box could move under an adversarial perturbation δ of bounded ℓ_2-norm. If the predicted bounding box can be made to disappear, or if the label can be made to change, after a perturbation with $\|\delta\|_2 < 0.36$, then we annotate the bounding box with a red X.



[1] Chiang, Ping-yeh, et al. "Detection as Regression: Certified Object Detection by Median Smoothing."
[2] Salman, Hadi, et al. "Black-box Smoothing: A Provable Defense for Pretrained Classifiers."

Randomized smoothing for Semantic Segmentation[1]

For **segmentation <u>every</u> pixel** needs to be classified. **Any atomic error** leads to the **overall certification fail**.

Authors proposed to **allow some pixels** to be **non-classified** if the smoothed classifier provides less probability than $\tau \in (\frac{1}{2}; 1]$ and redefine it:

$$\bar{f}_i^{\tau}(\boldsymbol{x}) = \begin{cases} c_{A,i} & \text{if } \mathbb{P}_{\epsilon \sim \mathcal{N}(0,\sigma)}(f_i(\boldsymbol{x}+\epsilon)) > \tau \\ \oslash & \text{else} \end{cases},$$

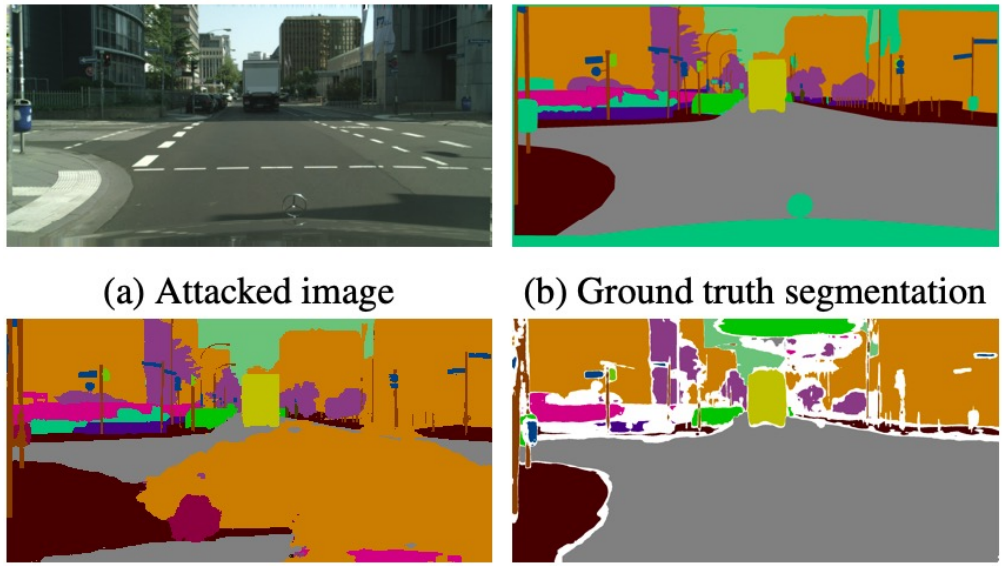
$$\text{where } c_{A,i} = \arg\max_{c \in \mathcal{Y}} \mathbb{P}_{\epsilon \sim \mathcal{N}(0,\sigma)}(f_i(\boldsymbol{x}+\epsilon) = c).$$

Making this assumption, they succeed to prove the certification Theorem and provide non-trivial results on segmentation benchmarks

Theorem 5.1. *Let* $\mathcal{I}_{\boldsymbol{x}} = \{i \mid \bar{f}_i^{\tau}(\boldsymbol{x}) \neq \oslash, i \in 1, \ldots, N\}$ *denote the set of non-abstain indices for* $\bar{f}^{\tau}(\boldsymbol{x})$*. Then,*

$$\bar{f}_i^{\tau}(\boldsymbol{x}+\delta) = \bar{f}_i^{\tau}(\boldsymbol{x}), \quad \forall i \in \mathcal{I}_{\boldsymbol{x}}$$

for $\delta \in \mathbb{R}^{N \times m}$ *with* $\|\delta\|_2 \leq R := \sigma \Phi^{-1}(\tau)$.



(a) Attacked image (b) Ground truth segmentation

(c) Attacked segmentation (d) Certified segmentation

				Cityscapes			
scale		σ	R	acc.	mIoU	%⊘	t
0.25	non-robust model	-	-	0.93	0.60	0.00	0.38
	base model	-	-	0.87	0.42	0.00	0.37
	SEGCERTIFY $n = 100, \tau = 0.75$	0.25	0.17	0.84	0.43	0.07	70.00
		0.33	0.22	0.84	0.44	0.09	70.21
		0.50	0.34	0.82	0.43	0.13	71.45
	SEGCERTIFY $n = 500, \tau = 0.95$	0.25	0.41	0.83	0.42	0.11	229.37
		0.33	0.52	0.83	0.42	0.12	230.69
		0.50	0.82	0.77	0.38	0.20	230.09

[1] Fischer, Marc, Maximilian Baader, and Martin Vechev. "Scalable certified segmentation via randomized smoothing."

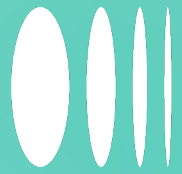
Takeaway

Certification in l_∞ is not working for high dimension input

In Computer Vision no need in any l_p (aside l_0 for patch attacks, but it is usually also combined with other perturbations)

Semantical perturbations harder to certify (+ interpolation!)

Current challenge: 3D and even non-rigid transformations of **real world**



Artificial Intelligence
Research Institute