

Universal Adversarial Perturbations

Author: M.A. Kiani, Mohammad Parsa Dini

Mentor: M.R. Rahmani

Instructor: Dr. Yassaee

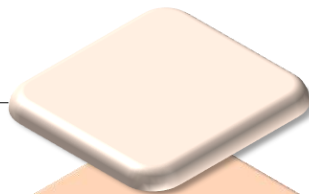
Acknowledgments

M.A. Kiani
B.Sc.C.E.
m.a.kiani@gmail.com

Mohammad Parsa Dini
B.Sc.E.E.
mohammadparsadinithefirst@gmail.com

**Introduction to
Definitions &
Motivations**

1



**Key idea &
Solutions for
Adversary**

3



Conclusion

5



2



**Problem
Setting & other
approaches**



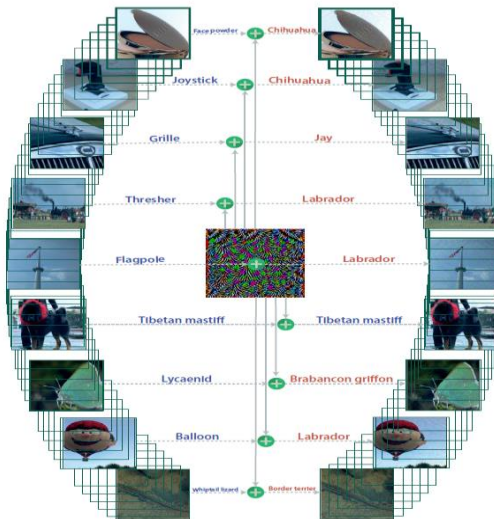
4



**Our contribution
& Experiments**

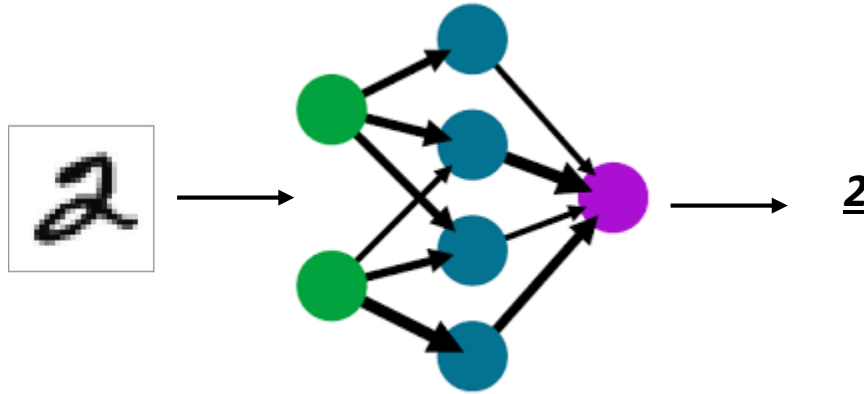


Definition:



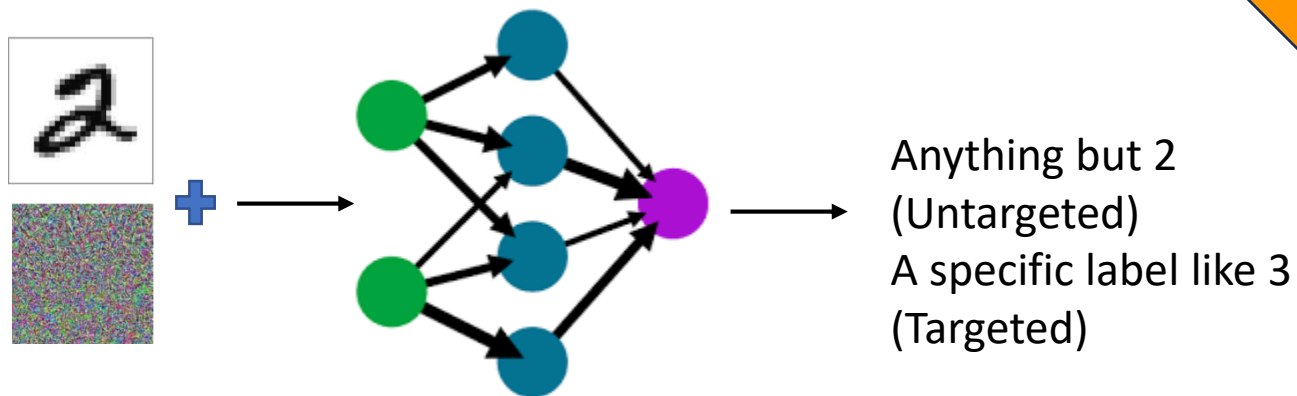
Can we find a **universal (image-agnostic)** and very small perturbation vector that causes natural images to be misclassified with high probability?

Preliminaries:



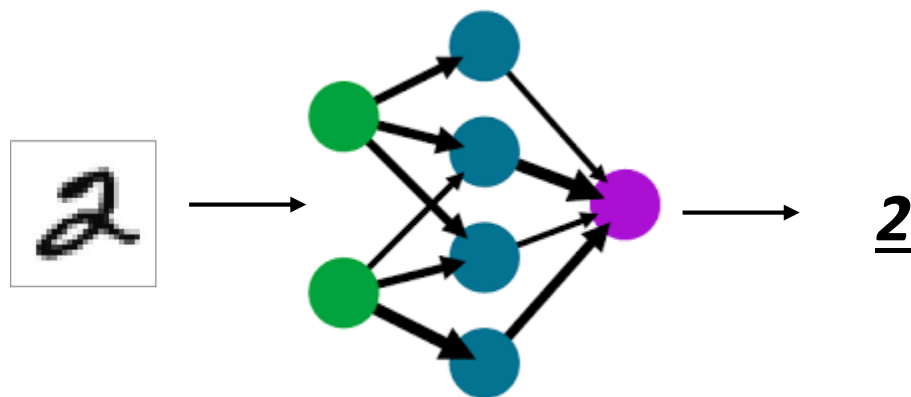
Feeding an image to the deep neural network and receive a softmax vector and then by decision-making we output the most probable label.

Preliminaries:



There are two categories in adversary models, targeted and untargeted attacks.

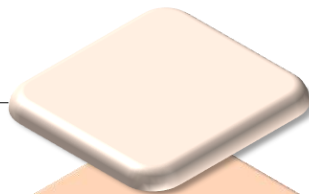
Preliminaries:



The attacks can access and use gradient and weights (White Box) or otherwise (Black Box).

**Introduction to
Definitions &
Motivations**

1



**Key idea &
Solutions for
Adversary**

3



Conclusion

5



2 Problem setting
& other
approaches

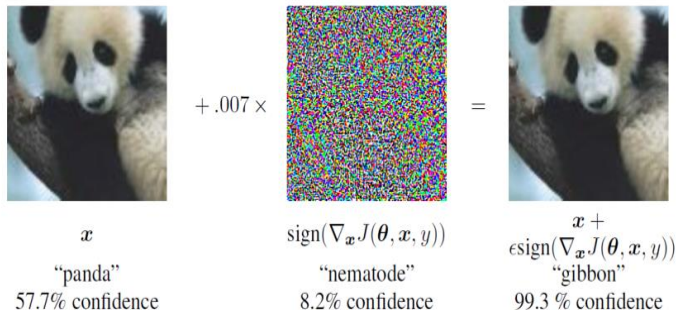


4 Our contribution
& Experiments



FGSM & L-BFGS

- In order to find the perturbed vector v , we use the gradient on which the model was trained.
- If the parameters to model are θ and y is our label then:
$$v = \epsilon \operatorname{sign} \nabla_x \{J(\theta, x, y)\}$$
- Given an image x , their method finds a different image x' that is similar to x under L2 distance, yet is labeled differently by the classifier.



$$\begin{aligned} &\text{minimize } c \cdot \|x - x'\|_2^2 + \text{loss}_{F,l}(x') \\ &\text{such that } x' \in [0, 1]^n \end{aligned}$$

UAP

● μ is the distribution of images in R^d . $\chi = \{x_1, \dots, x_m\}$ is the dataset.

● seeking perturbation vectors $v \in R^d$ that fool the classifier $f: R^d \rightarrow C$ on almost all data sampled from μ . In other words:
 $f(x + v) \neq f(x)$ for almost all $x \sim \mu$.

● The goal is to find v that satisfies the following two constraints:

1. $\|v\|_p \leq \xi$

2. $P_{x \sim \mu} [f(x + v) \neq f(x)] \geq 1 - \delta$

UAP Algorithm

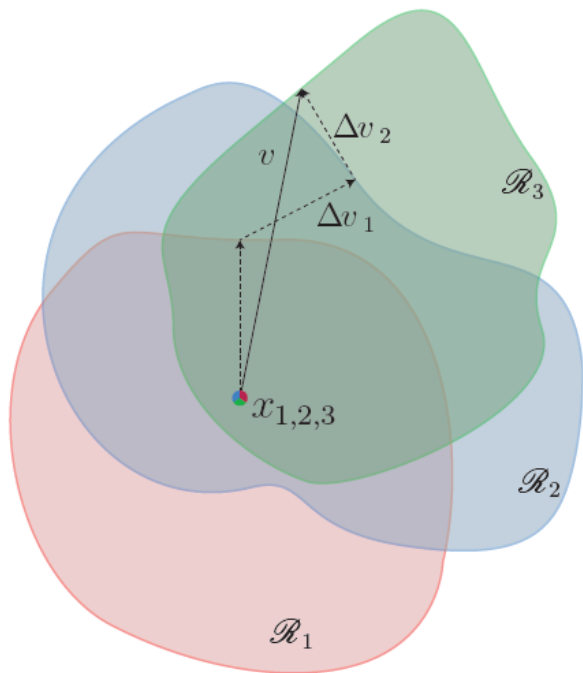
Algorithm 1 Computation of universal perturbations.

- 1: **input:** Data points X , classifier \hat{k} , desired ℓ_p norm of the perturbation ξ , desired accuracy on perturbed samples δ .
 - 2: **output:** Universal perturbation vector v .
 - 3: Initialize $v \leftarrow 0$.
 - 4: **while** $\text{Err}(X_v) \leq 1 - \delta$ **do**
 - 5: **for** each datapoint $x_i \in X$ **do**
 - 6: **if** $\hat{k}(x_i + v) = \hat{k}(x_i)$ **then**
 - 7: Compute the *minimal* perturbation that sends $x_i + v$ to the decision boundary:
$$\Delta v_i \leftarrow \arg \min_r \|r\|_2 \text{ s.t. } \hat{k}(x_i + v + r) \neq \hat{k}(x_i).$$
 - 8: Update the perturbation:
$$v \leftarrow \mathcal{P}_{p,\xi}(v + \Delta v_i).$$
 - 9: **end if**
 - 10: **end for**
 - 11: **end while**
-

$$\text{Err}(X_v) := \frac{1}{m} \sum_{i=1}^m 1_{\hat{k}(x_i+v) \neq \hat{k}(x_i)} \geq 1 - \delta.$$

$$\mathcal{P}_{p,\xi}(v) = \arg \min_{v'} \|v - v'\|_2 \text{ subject to } \|v'\|_p \leq \xi.$$

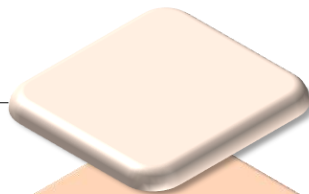
Visualization



This gradient indicates the direction in which the perturbation should be adjusted to increase the loss.

**Introduction to
Definitions &
Motivations**

1



**2 Problem setting
& approaches**

2



**Robustifying via
Randomized
smoothness**

3



**4 Our contribution
& Experiments
& Bounds**

4

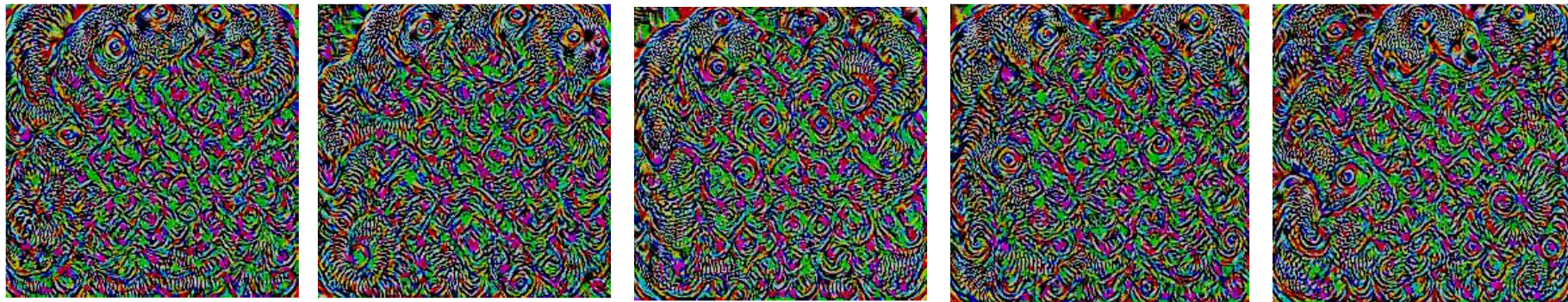


Conclusion

5



The space of perturbed vectors



Diversity of universal perturbations for the GoogLeNet architecture. The five perturbations are generated using different random shufflings of the set X . Note that the normalized inner products for any pair of universal perturbations does not exceed 0.1, which highlights the diversity of such perturbations

How does the norm effect the attack?

	VGG-F	CaffeNet	GoogLeNet	VGG-16	VGG-19	ResNet-152
VGG-F	93.7%	71.8%	48.4%	42.1%	42.1%	47.4 %
CaffeNet	74.0%	93.3%	47.7%	39.9%	39.9%	48.0%
GoogLeNet	46.2%	43.8%	78.9%	39.2%	39.8%	45.5%
VGG-16	63.4%	55.8%	56.5%	78.3%	73.1%	63.4%
VGG-19	64.0%	57.2%	53.6%	73.5%	77.8%	58.0%
ResNet-152	46.3%	46.3%	50.5%	47.0%	45.5%	84.0%

Randomized Smoothing

Consider a classification problem from \mathbb{R}^d to classes \mathcal{Y} . Randomized smoothing is a method for constructing a new, **smoothed classifier** \hat{f} from an arbitrary base classifier f .

- When queried at x , the smoothed classifier \hat{f} returns whichever class the base classifier f is most likely to return when x is perturbed by isotropic Gaussian noise:

$$\hat{f}(x) = \underset{c \in \mathcal{Y}}{\operatorname{argmax}} \mathbb{P}(f(x + \epsilon) = c) \quad (1)$$

where $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$

The noise level σ is a hyperparameter of the smoothed classifier \hat{f} which controls a robustness/accuracy tradeoff.

Theorem 1

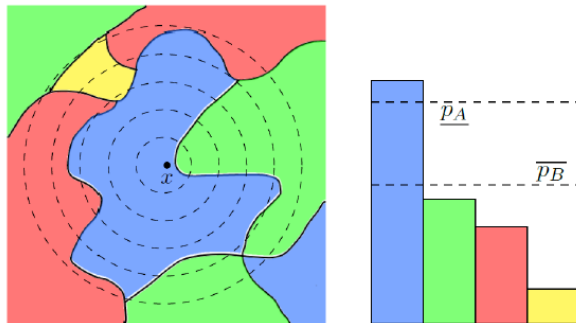
Theorem 1. Let $f : \mathbb{R}^d \rightarrow \mathcal{Y}$ be any deterministic or random function, and let $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$. Let \hat{f} be defined as in (1). Suppose $C_A \in \mathcal{Y}$ and $\underline{P}_A, \overline{P}_B \in [0, 1]$ satisfy:

$$\mathbb{P}(f(x + \epsilon) = C_A) \geq \underline{P}_A \geq \overline{P}_B \geq \max_{C \neq C_A} \mathbb{P}(f(x + \epsilon) = C)$$

Then $\hat{f}(x + \delta) = C_A$ for all $\|\delta\|_2 \leq R$, where

$$R = \frac{\sigma}{2} (\Phi^{-1}(\underline{P}_A) - \Phi^{-1}(\overline{P}_B))$$

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



Proof of Thrm 1:

To prove Theorem 1, we need to find the Lipschitz constant of the smoothed classifier \hat{f} .

Let

$$f : \mathbb{R}^d \rightarrow [0, 1]$$
$$\hat{f}(x) = \mathbb{E}_{z \sim \mathcal{N}(0, I_d)}[f(x + \sigma z)]$$

It is well-known that \hat{f} is Lipschitz (it has uniform bound on the Lipschitz constant). In practice, we can approximate \hat{f} by empirical average

$$y^{(k)} = \sum_{i=1}^k f(x + \sigma z_i), \quad \text{where } z_i \sim \mathcal{N}(0, I_d)$$

It can be shown that if $k \rightarrow \infty$, $y^{(k)}$ almost surely converges to \hat{f} .

$$\hat{f}(x) = \mathbb{E}_{z \sim \mathcal{N}(0, I_d)}[f(x + \sigma z)] = \frac{1}{(2\pi)^{d/2}} \int f(x + \sigma z) \exp\left\{-\frac{\|z\|_2^2}{2}\right\} dz$$

$\hat{f}(x)$ is the weighted average of $f(x)$ in the vicinity of x .

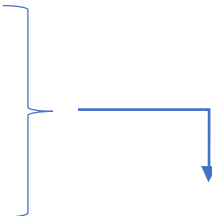
Hence, We calculate $\|\nabla_x \hat{f}(x)\|_2$ in order to find the **upper bound on Lipschitz constant** of \hat{f} .

Proof of Thrm 1:

$$\hat{f}(x) = f * g_\sigma$$

$$\nabla_x \hat{f}(x) = \nabla_x (f * g_\sigma)$$

$$\nabla_x \hat{f}(x) = \nabla_x (f * g_\sigma) = f * \nabla_x g_\sigma$$



$$\nabla_x \hat{f}(x) = \int f(x - w) \frac{-w}{\sigma^2} g_\sigma(w) dw$$

Change of variable: $w = -\sigma z$. We get

$$\begin{aligned} \nabla_x \hat{f}(x) &= \int f(x + \sigma z) \frac{\sigma z}{\sigma^2} g_\sigma(-\sigma z) \sigma^d dz = \int f(x + \sigma z) \frac{z}{\sigma} \frac{\sigma^d}{(2\pi)^{d/2} \sigma^d} \exp\left\{-\frac{\|-\sigma z\|_2^2}{2\sigma^2}\right\} dz \\ &= \int f(x + \sigma z) \frac{z}{\sigma} \underbrace{\frac{1}{(2\pi)^{d/2}} \exp\left\{-\frac{\|z\|_2^2}{2}\right\}}_{\mathcal{N}(0, I_d)} dz = \mathbb{E}_{z \sim \mathcal{N}(0, I_d)} \left[f(x + \sigma z) \frac{z}{\sigma} \right] \end{aligned}$$

Therefore, for Lipschitz constant of \hat{f} , we have

$$L_{\hat{f}} \leq \|\nabla_x \hat{f}(x)\|_2 \Rightarrow L_{\hat{f}} \leq \|\mathbb{E}_{z \sim \mathcal{N}(0, I_d)} [f(x + \sigma z) \frac{z}{\sigma}] \|_2$$

Proof of Thrm 1:

Change of variable: $w = -\sigma z$. We get

$$\begin{aligned}\nabla_x \hat{f}(x) &= \int f(x + \sigma z) \frac{\sigma z}{\sigma^2} g_\sigma(-\sigma z) \sigma^d dz = \int f(x + \sigma z) \frac{z}{\sigma} \frac{\sigma^d}{(2\pi)^{d/2} \sigma^d} \exp\left\{-\frac{\|-\sigma z\|_2^2}{2\sigma^2}\right\} dz \\ &= \int f(x + \sigma z) \frac{z}{\sigma} \underbrace{\frac{1}{(2\pi)^{d/2}} \exp\left\{-\frac{\|z\|_2^2}{2}\right\}}_{\mathcal{N}(0, I_d)} dz = \mathbb{E}_{z \sim \mathcal{N}(0, I_d)} \left[f(x + \sigma z) \frac{z}{\sigma} \right]\end{aligned}$$

Therefore, for Lipschitz constant of \hat{f} , we have

$$\begin{aligned}L_{\hat{f}} \leq \|\nabla_x \hat{f}(x)\|_2 &\Rightarrow L_{\hat{f}} \leq \|\mathbb{E}_{z \sim \mathcal{N}(0, I_d)} [f(x + \sigma z) \frac{z}{\sigma}]\|_2 \leq \mathbb{E}_{z \sim \mathcal{N}(0, I_d)} [\underbrace{\|f(x + \sigma z) \frac{z}{\sigma}\|_2}_{\in [0, 1]}] \\ &\leq \frac{1}{\sigma} \mathbb{E}_{z \sim \mathcal{N}(0, I_d)} [\|z\|_2]\end{aligned}$$

Proof of Thrm 1:

Therefore, for Lipschitz constant of \hat{f} , we have

$$\begin{aligned} L_{\hat{f}} &\leq \|\nabla_x \hat{f}(x)\|_2 \Rightarrow L_{\hat{f}} \leq \|\mathbb{E}_{z \sim \mathcal{N}(0, I_d)} [f(x + \sigma z) \frac{z}{\sigma}]\|_2 \leq \mathbb{E}_{z \sim \mathcal{N}(0, I_d)} [\underbrace{\|f(x + \sigma z) \frac{z}{\sigma}\|_2}_{\in [0,1]}] \\ &\leq \frac{1}{\sigma} \mathbb{E}_{z \sim \mathcal{N}(0, I_d)} [\|z\|_2] \leq \frac{1}{\sigma} \sqrt{\mathbb{E}_{z \sim \mathcal{N}(0, I_d)} [\|z\|_2^2]} \end{aligned}$$

Note: Since $f(x) = x^2$ is a convex function, for random variable X , we have

$$f(\mathbb{E}[g(X)]) \underset{\text{Jensen's Inequality}}{\leq} \mathbb{E}[f(g(X))] \Rightarrow \mathbb{E}^2[g(X)] \leq \mathbb{E}[g^2(X)] \Rightarrow \mathbb{E}[g(X)] \leq \sqrt{\mathbb{E}[g^2(X)]}$$

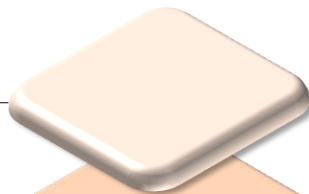
$$L_{\hat{f}} \leq \frac{1}{\sigma} \sqrt{\mathbb{E}_{z \sim \mathcal{N}(0, I_d)} [\|z\|_2^2]}$$



$$L_{\hat{f}} \leq \frac{\sqrt{d}}{\sigma}$$

**Introduction to
Definitions &
Motivations**

1



**Key idea &
Solutions for
Adversary**

3



Conclusion

5



2

**Problem setting
& other
approaches**

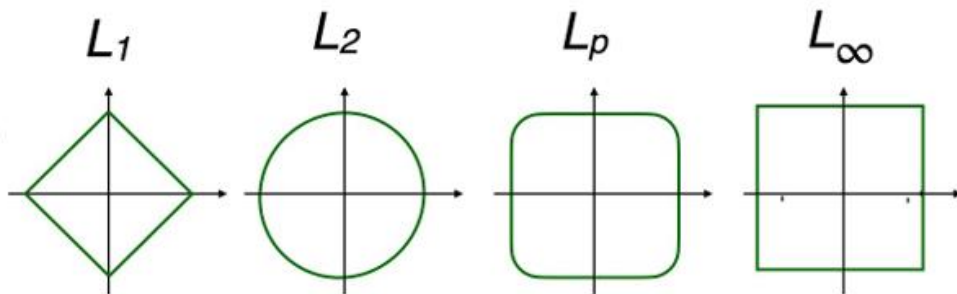


4

**Our contribution
& Experiments**

How does the norm effect the attack?

The L_p Norm



Since l_∞ norm measures the maximum change to any pixel in the image, Perturbations under the l_∞ norm are also small but can be more effective in causing misclassification compared to l_2 and l_1 . And As we can see, the results are as we expected.

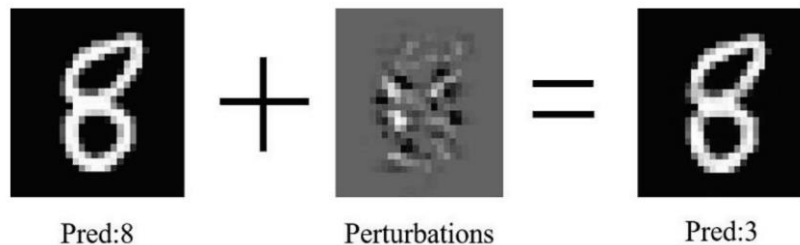
Our method's algorithm

Algorithm 1 Computation of Universal Perturbations

Require: Data points X , classifier \hat{k} , desired p -norm of the perturbation ξ , desired accuracy on perturbed samples ϵ .

Ensure: Universal perturbation vector v .

```
1: Initialize  $v \leftarrow 0$ .
2: while  $Err(X + v) \geq 1 - \epsilon$  do
3:   for each datapoint  $x_i \in X$  do
4:     if  $\hat{k}(x_i + v) = \hat{k}(x_i)$  then
5:       * Compute the minimal perturbation that sends  $x_i + v$  to the decision
         boundary:  $\Delta v_i \leftarrow \arg \min_r \|r\|_\infty + \quad s.t. \quad \hat{k}(x_i + v + r) \neq \hat{k}(x_i)$ 
6:       * Update the perturbation:  $v \leftarrow P_{p,\xi}(v + \Delta v_i)$ 
7:     end if
      smooth the image with RBF Kernel.  $X_{out} = (X + v) * K_{RBF}$ 
8:   end for
9: end while
```



Accuracy dump after attack on LeNet model &

Model	Acc. Before attack	Acc. After Attack
FGSM	94.2%	18.3%
PGD	94.2%	21.7%
LBFGS	94.2%	27.6%
UAP	94.2%	29.1%
Our prop. method	94.2%	38.5%



$$E_{X \sim P_X(x)} \left\{ \sup_{\tilde{t} \in T} H(\hat{f}(x + t)) - H(\hat{f}(x)) \right\} \leq \int_0^\infty \sqrt{\log(N(m, d, \chi))} \, dm$$

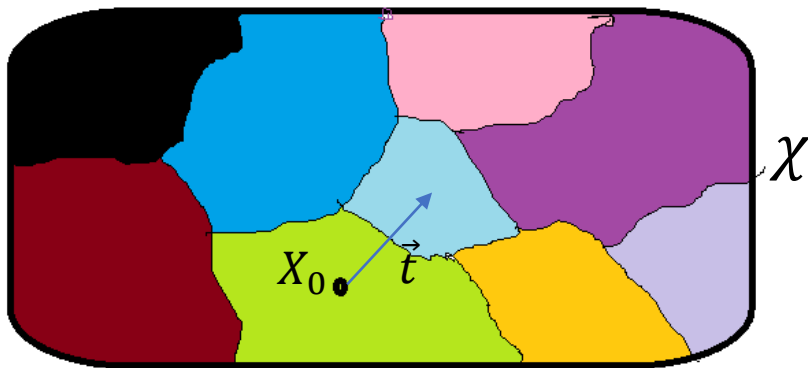
$$L_{\hat{f}} \leq \frac{\sqrt{d}}{\sigma}$$

Lipschitz constant via
Randomized smoothing

Index $\tilde{t} \in T$ in Hamming space of
the corresponding vector $x + t$

We used Lipschitzness on classifier (our metric):

$$(f(x + t), f(x + s))_H \leq c \cdot L_{\hat{f}} \|\tilde{t} - \tilde{s}\|_p$$



num of labels: C
Maximum num of adjacent
classes in hamming space $d < C$

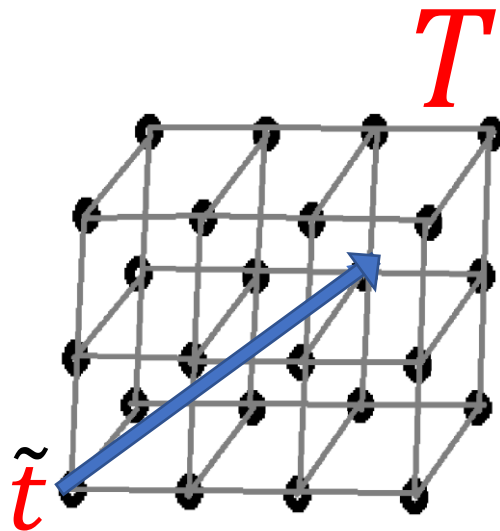
$$E_{X \sim P_X(x)} \left\{ \sup_{\tilde{t} \in T} H(\hat{f}(X + t)) - H(\hat{f}(X)) \right\} \leq \sup_m m \sqrt{\log(N(m, d, \chi))}$$

Where for hamming distance we know:

$$\frac{2^d}{\sum_{k=0}^m \binom{d}{k}} \leq N(m, d, \chi) \leq \frac{2^d}{\sum_{k=0}^{m/2} \binom{d}{k}}$$

Which finally leads to:

$$E_{X \sim P_X(x)} \left\{ \sup_{\tilde{t} \in T} |H(f(X + t)) - H(\hat{f}(x))| \right\} \leq \sup_m m \frac{2^d}{\sum_{k=0}^{m/2} \binom{d}{k}}$$



**Introduction to
Definitions &
Motivations**

1



**Key idea &
Solutions for
Adversary**

3



2 **Problem setting
& other
approaches**



4

**Our contribution
& Experiments**



Conclusion

5



Conclusion & Future Works

In this presentation we have reviewed the general literature regarding adversarial attack algorithms and possible defenses against it.

- We have simulated a different attacking strategy using some p-norms and have reached noticing accuracy.
- For structural difference images it can adapt itself and have **creativity**.
- There are many more possible future research directions regarding this problem. For instance, we can assume upper bound on Wasserstein distance between perturbed and original distributions, and explore the expected adversarial error for gaussian smoothing distribution.

References

Additional Resources

itinerai/us_places.
https://huggingface.co/datasets/itinerai/us_places

pcuenq/lsun-bedrooms.
<https://huggingface.co/datasets/pcuenq/lsun-bedrooms>.

Yaron Cruz, Image Processing, Aug 2024.
<https://www.slideserve.com/yaron/image-processing>.

Main Sources

Certified Adversarial Robustness via Randomized Smoothing

[arXiv:2310.16047](https://arxiv.org/abs/2310.16047)

Universal Adversarial Perturbation (UAP)

[arXiv:2310.16047](https://arxiv.org/abs/2310.16047)

