

# Secure Learning in Adversarial Deep Neural Networks

Bo Li, Dawn Song

UC, Berkeley

# Machine Learning in Physical World



Autonomous Driving



Healthcare



Smart City



Malware Classification



Fraud Detection



Biometrics Recognition

# Security & Privacy Problems

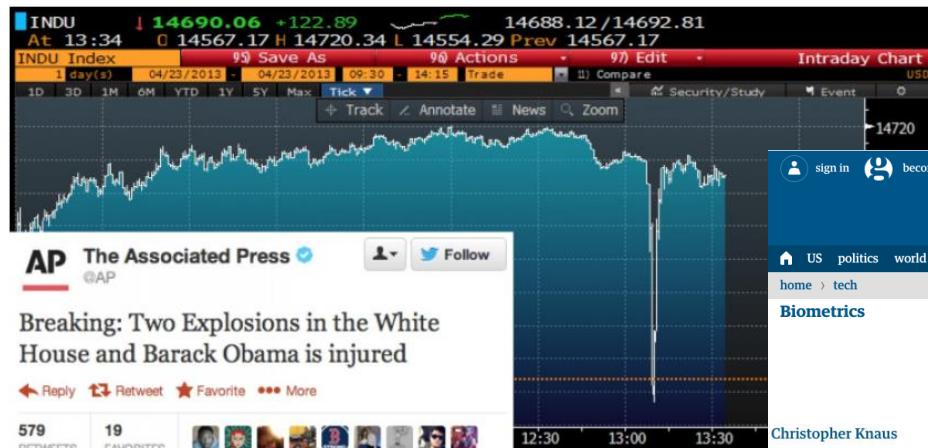
Sections 

The Washington Post

WorldViews

## Syrian hackers claim AP hack that tipped stock market by \$136 billion. Is it terrorism?

By Max Fisher April 23, 2013

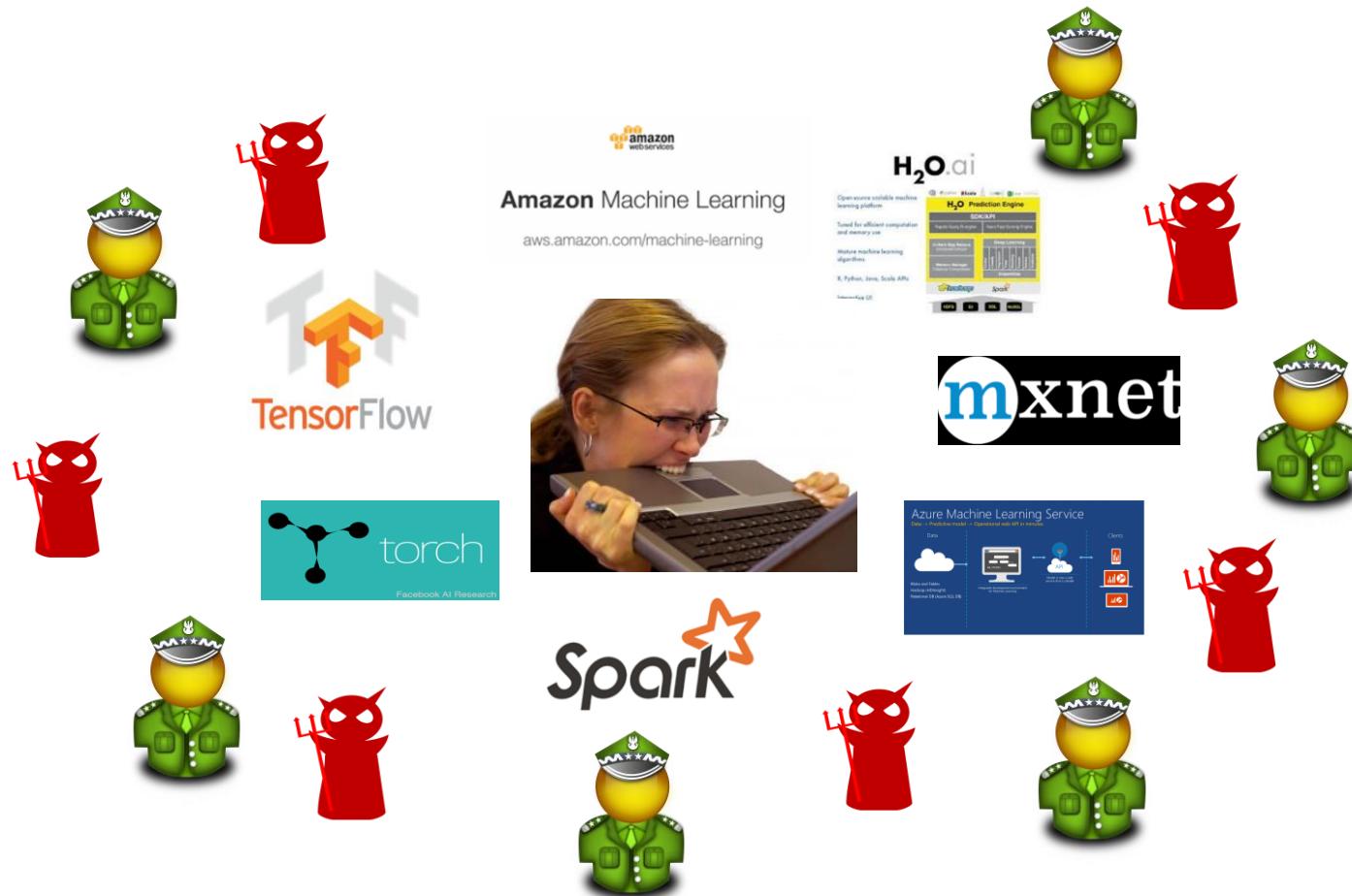


Privacy Concerns

Security Problems



# We Are in Adversarial Environments



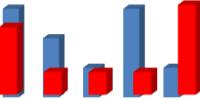


While cybersecurity R&D needs are addressed in greater detail in the NITRD Cybersecurity R&D Strategic Plan, some cybersecurity risks are specific to AI systems. **One key research area is “adversarial machine learning”**, that explores the degree to which AI systems can be compromised by “contaminating” training data, by modifying algorithms, or by making subtle changes to an object that prevent it from being correctly identified....

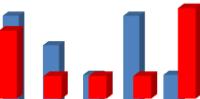
- National Science and Technology Council  
2016

# Perils of Stationary Assumption

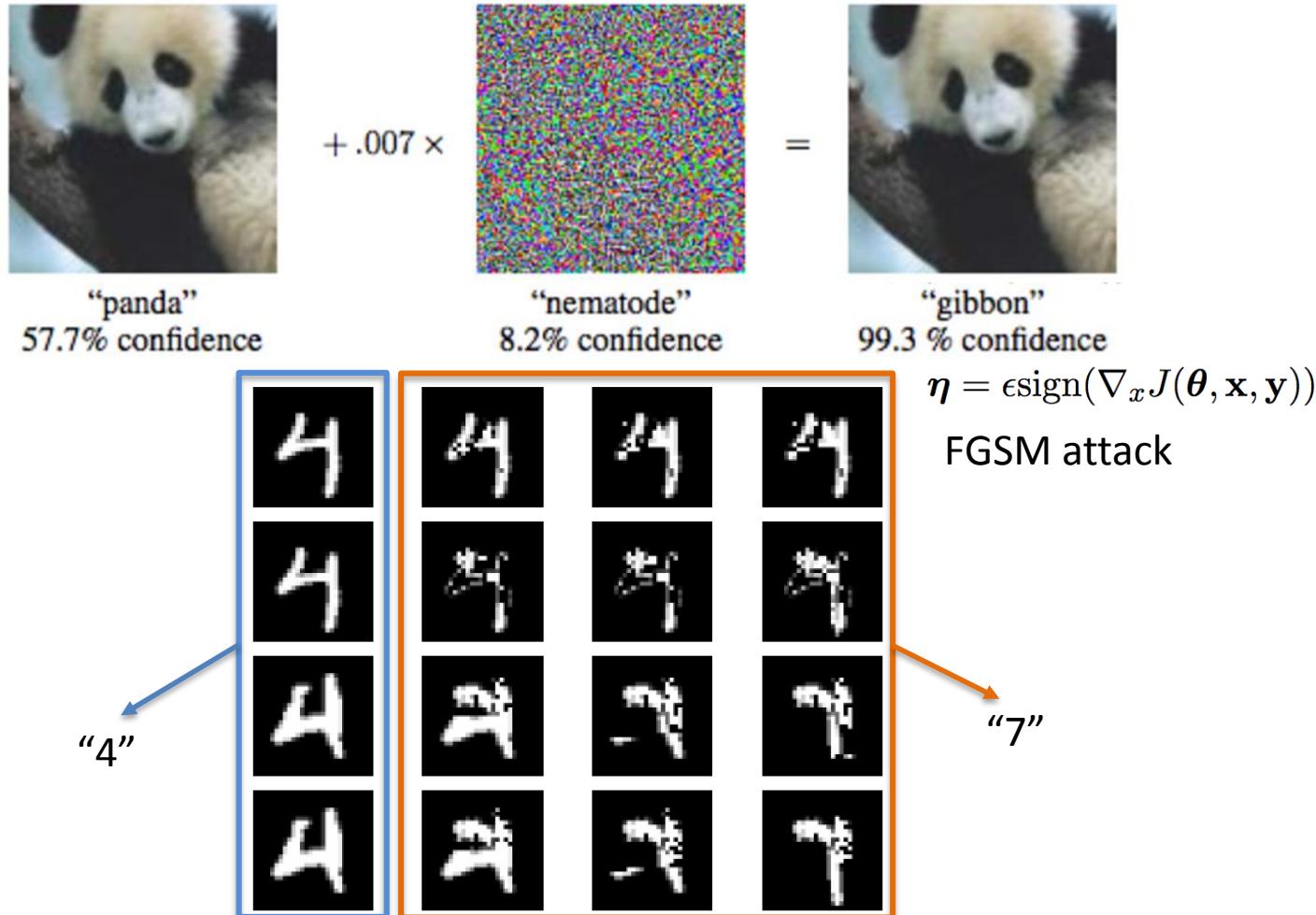
Traditional machine learning approaches assume

Training Data 

$\approx$

Testing Data 

# Adversarial Examples



Goodfellow, Ian J., Jonathon Shlens, and Christian Szegedy. "Explaining and harnessing adversarial examples." *ICLR 2015*.  
[Li, Bo](#), Yevgeniy Vorobeychik, and Xinyun Chen. "A General Retraining Framework for Scalable Adversarial Classification." *ICLR*. (2016).

# Optimization Based Attack

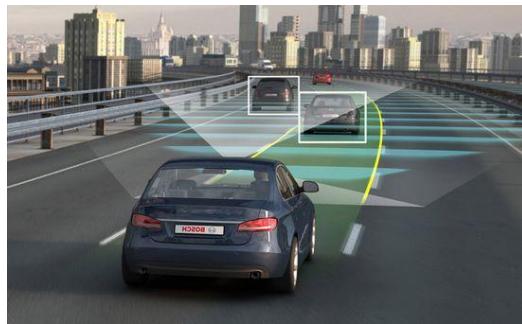
$$\begin{aligned} & \text{minimize } \mathcal{D}(x, x + \delta) \\ \text{such that } & C(x + \delta) = t \\ & x + \delta \in [0, 1]^n \end{aligned}$$

$$\begin{aligned} & \text{minimize } \mathcal{D}(x, x + \delta) + c \cdot f(x + \delta) \\ \text{such that } & x + \delta \in [0, 1]^n \end{aligned}$$

	Best Case				Average Case				Worst Case			
	MNIST		CIFAR		MNIST		CIFAR		MNIST		CIFAR	
	mean	prob	mean	prob	mean	prob	mean	prob	mean	prob	mean	prob
Our $L_0$	8.5	100%	5.9	100%	16	100%	13	100%	33	100%	24	100%
JSMA-Z	20	100%	20	100%	56	100%	58	100%	180	98%	150	100%
JSMA-F	17	100%	25	100%	45	100%	110	100%	100	100%	240	100%
Our $L_2$	1.36	100%	0.17	100%	1.76	100%	0.33	100%	2.60	100%	0.51	100%
Deepfool	2.11	100%	0.85	100%	-	-	-	-	-	-	-	-
Our $L_\infty$	0.13	100%	0.0092	100%	0.16	100%	0.013	100%	0.23	100%	0.019	100%
Fast Gradient Sign	0.22	100%	0.015	99%	0.26	42%	0.029	51%	-	0%	0.34	1%
Iterative Gradient Sign	0.14	100%	0.0078	100%	0.19	100%	0.014	100%	0.26	100%	0.023	100%

[Carlini, Wagner, Towards robustness of neural networks. 2017]

# Autonomous Driving is the Trend...

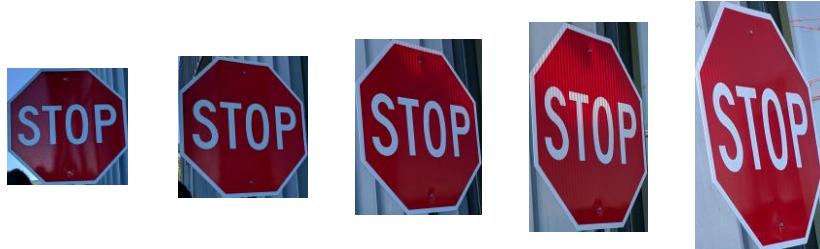


# However, What We Can See Everyday...



# The Physical World Is... Messy

Varying Physical Conditions (Angle, Distance, Lighting, ...) Physical Limits on Imperceptibility



Fabrication/Perception Error (Color Reproduction, etc.)



Digital Noise  
(What you want)

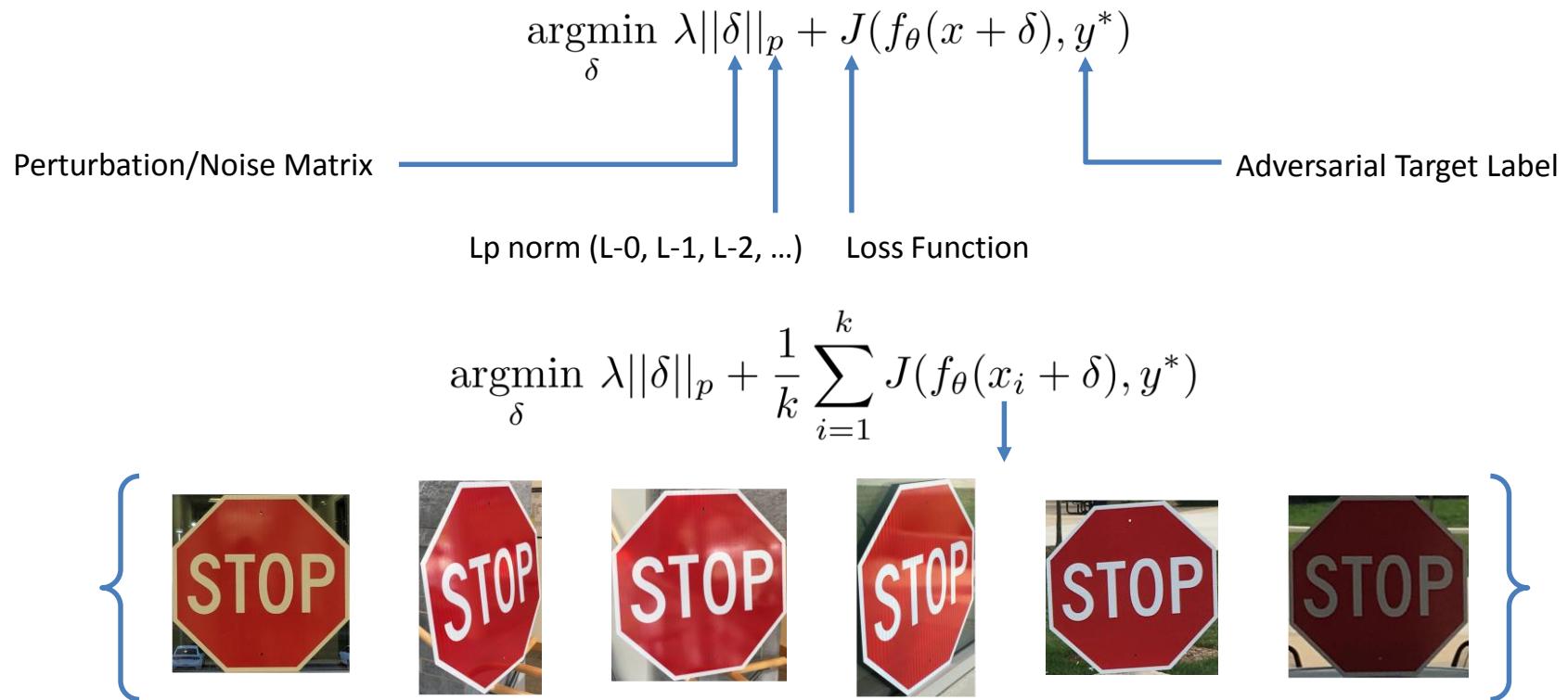
What is  
printed

What a camera  
may see

Background Modifications\* Image Courtesy, OpenAI

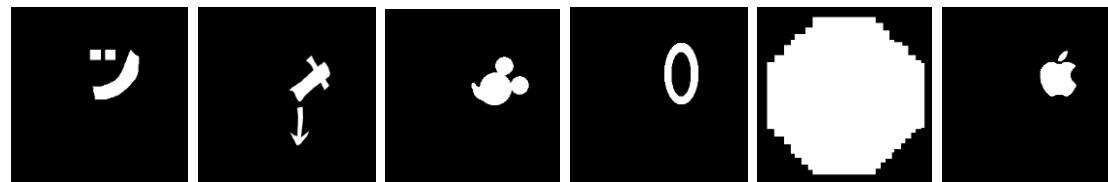


# An Optimization Approach To Creating Robust Physical Adversarial Examples



# Optimizing Spatial Constraints (Handling Limits on Imperceptibility)

$$\operatorname{argmin}_{\delta} \lambda ||M_x \cdot \delta||_p + \frac{1}{k} \sum_{i=1}^k J(f_{\theta}(x_i + M_x \cdot \delta), y^*)$$



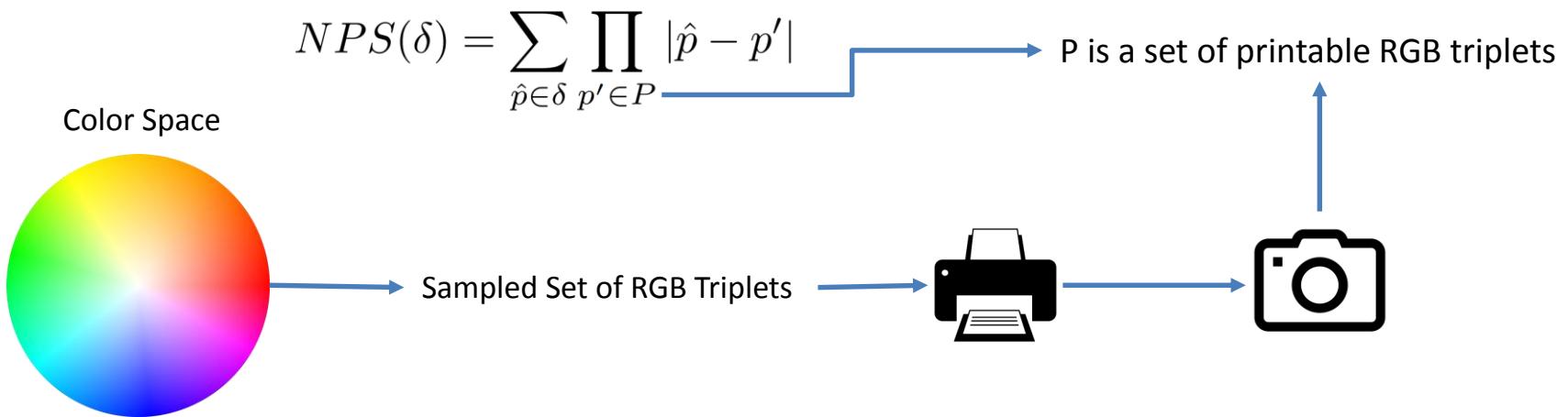
Subtle Poster  
Camouflage Sticker

Mimic vandalism  
"Hide in the human psyche"



# Handling Fabrication/Perception Errors

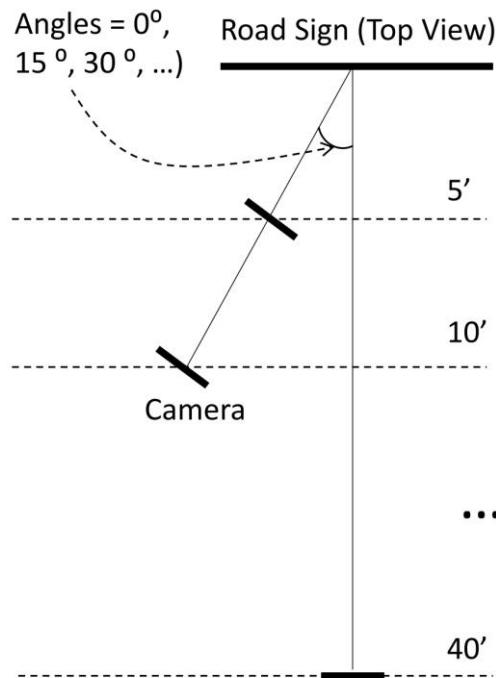
$$\operatorname{argmin}_{\delta} \lambda \|M_x \cdot \delta\|_p + \frac{1}{k} \sum_{i=1}^k J(f_{\theta}(x_i + M_x \cdot \delta), y^*) + NPS(M_x \cdot \delta)$$



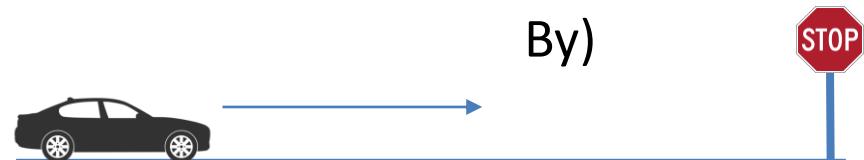
NPS based on Sharif et al., "Accessorize to a crime," CCS 2016

# How Can We Realistically Evaluate Attacks?

## Lab Test (Stationary)



## Field Test (Drive-By)



Record video

Sample frames every k frames

Run sampled frames through DNN



## Lab Test Summary (Stationary)

Target Class: Speed Limit 45

GTSRB\*-CNN

Subtle Poster

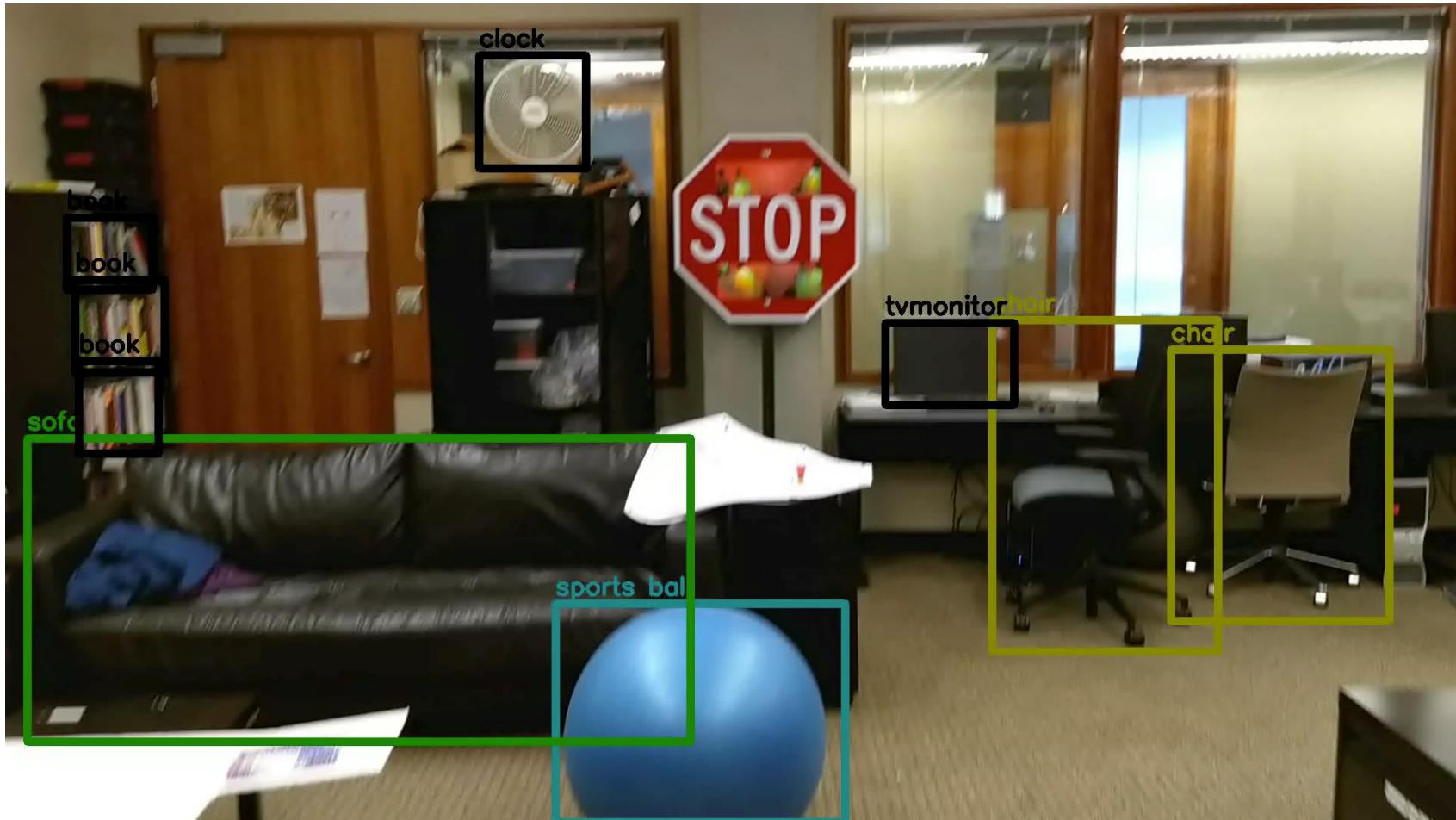
# Art Perturbation



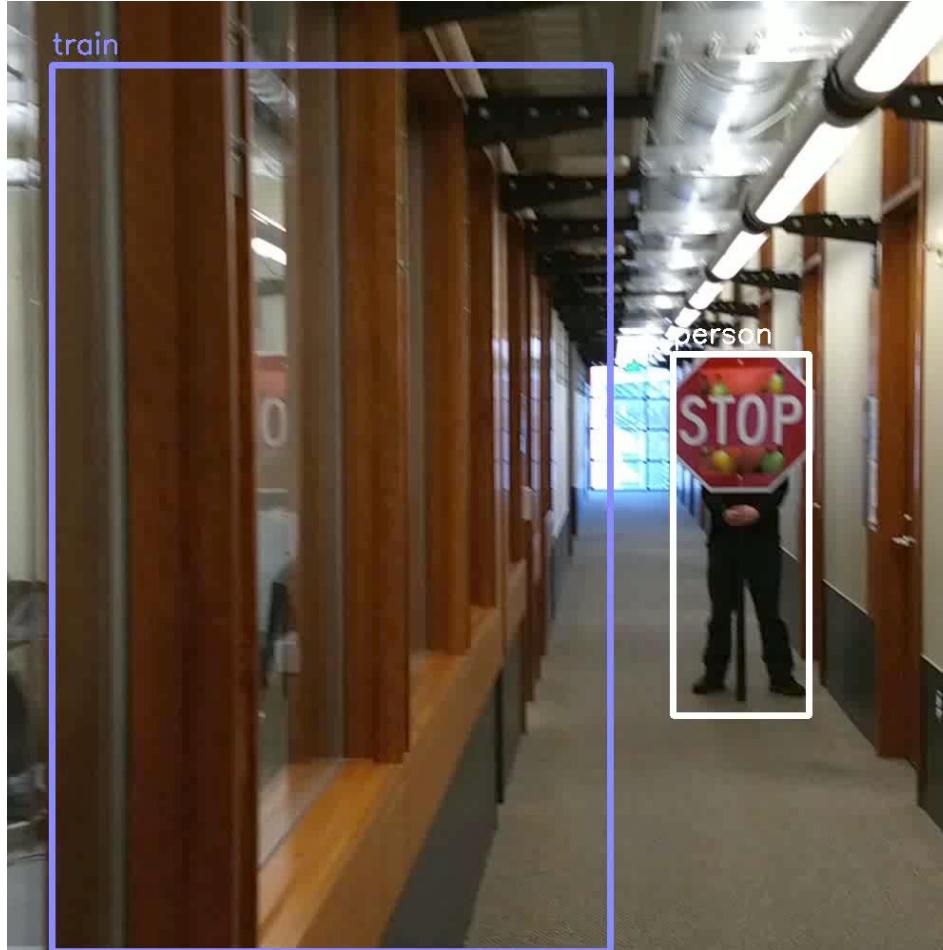
# Subtle Perturbation



# Physical Attacks Against Detectors



# Physical Attacks Against Detectors



# Adversarial Examples in Physical World

**Adversarial perturbations are possible in physical world under different conditions and viewpoints, including the distances and angles.**

# Different approaches to optimize the objective

- Fast approaches
  - Fast gradient sign ( $d = \|\cdot\|_\infty$ ):  $x^* = x + B \text{sgn}(\nabla_x \ell(f_\theta(x), y))$
  - Fast gradient ( $d = \|\cdot\|_2$ ):  $x^* = x + B \left( \frac{\nabla_x \ell(f_\theta(x), y)}{\|\nabla_x \ell(f_\theta(x), y)\|_2} \right)$
- Iterative approaches
  - E.g., use a SGD optimizer, such as Adam, to optimize

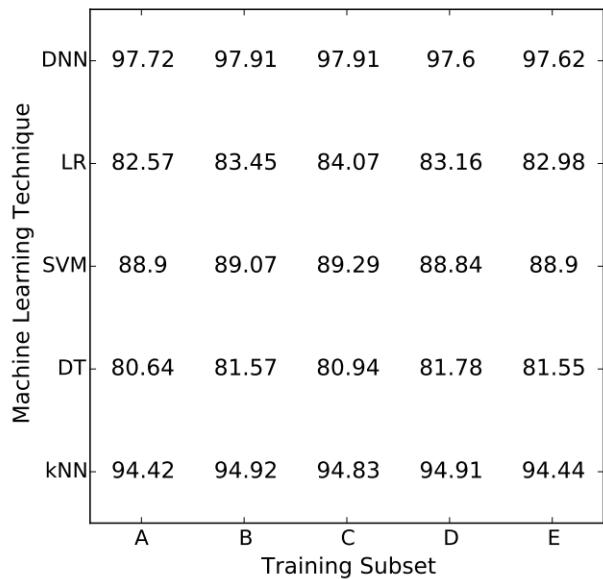
$$\max_{x^*} \ell(f_\theta(x^*), y) + \lambda d(x, x^*)$$

- Optimization  $\underset{\delta}{\operatorname{argmin}} \lambda \|\delta\|_p + J(f_\theta(x + \delta), y^*)$
- **Need to know model  $f_\theta$**

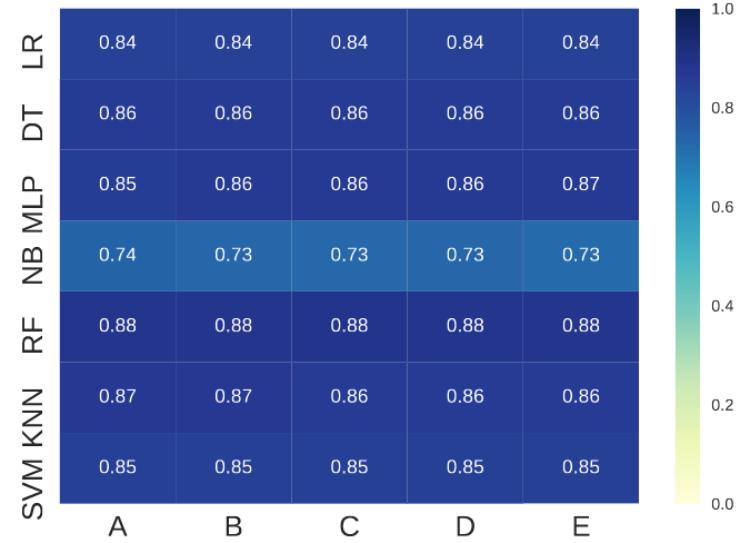
# A General Framework for Black-box attacks

- Zero-Query Attack
  - Random perturbation
  - Difference of means
  - *Transferability-based attack*
    - Practical Black-Box Attacks against Machine Learning
    - Ensemble transferability-based attack
- Query Based Attack
  - Finite difference gradient estimation
  - Query reduced gradient estimation
  - Results: similar effectiveness to whitebox attack
  - A general active query game model

# Transferability



MNIST



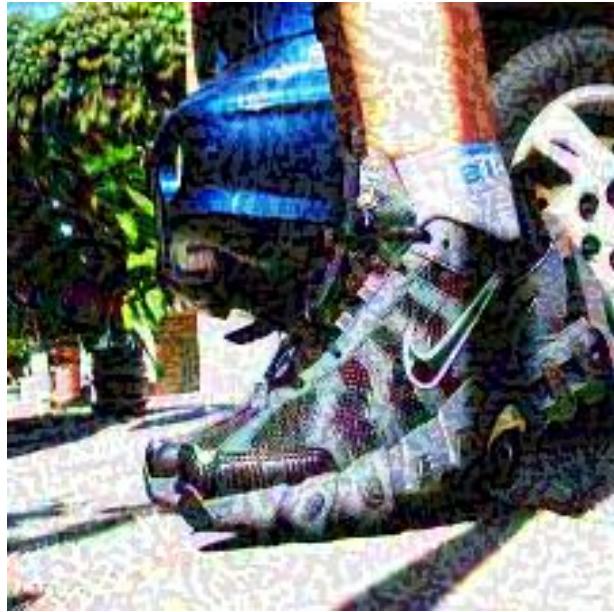
PDF Malware

Papernot, McDaniel, Goodfellow, Transferability in Machine Learning: from Phenomena to Black-Box Attacks using Adversarial Samples. 2016

Xiao, Li, Malware Evasion Attacks Based on Generative Adversarial Networks (GANs), 2017.

# Targeted vs Non-targeted

- Non-targeted adversarial examples
  - The goal is to mislead the classifier to predict **any labels** other than the ground truth
  - Most existing work deals with this goal
- Targeted adversarial examples
  - The goal is to mislead the classifier to predict a **target label** for an image
  - **Harder!**



Ground truth: running shoe

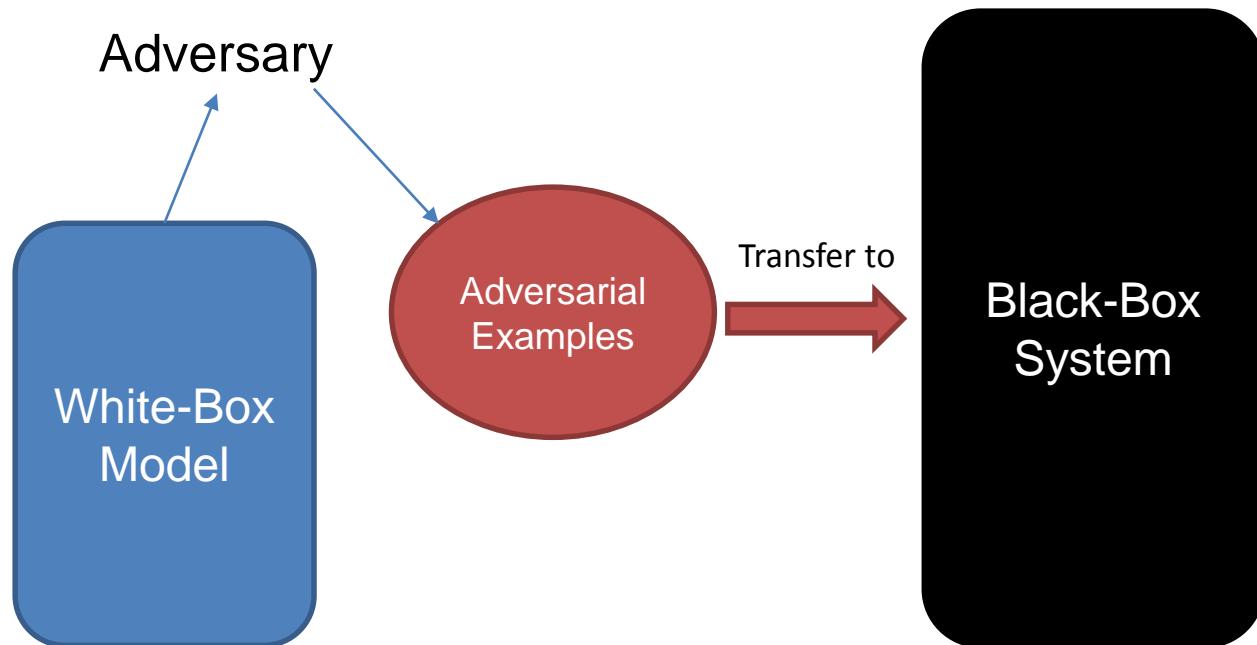
<b>VGG16</b>	<b>Military uniform</b>
ResNet50	Jigsaw puzzle
ResNet101	Motor scooter
ResNet152	Mask
GoogLeNet	Chainsaw

# Targeted Adversarial Example's Transferability Among Two Models is Poor!

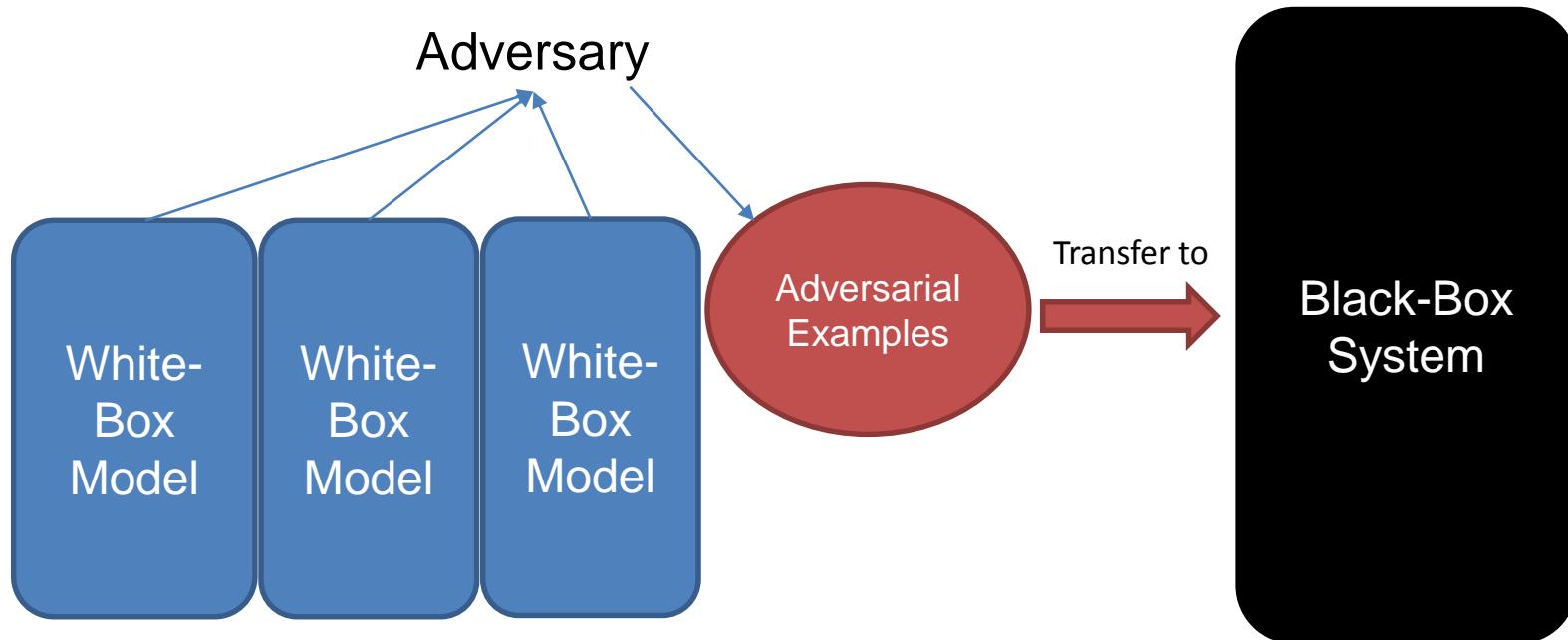
	ResNet152	ResNet101	ResNet50	VGG16	GoogLeNet	Incept-v3
ResNet152	100%	2%	1%	1%	1%	0%
ResNet101	3%	100%	3%	2%	1%	1%
ResNet50	4%	2%	100%	1%	1%	0%
VGG16	2%	1%	2%	100%	1%	0%
GoogLeNet	1%	1%	0%	1%	100%	0%
Incept-v3	0%	0%	0%	0%	0%	100%

Only 2% of the adversarial images generated for VGG16 (row) can be predicted as the targeted label by ResNet50 (column)

# Black-box Attacks Based On Transferability



# Ensemble Targeted Black-box Attacks Based On Transferability



# Clarifai.com

Ground truth from ImageNet: broom

The image shows a screenshot of the Clarifai demo interface. At the top, there is a navigation bar with three horizontal lines and the word "clarifai". Below the navigation bar is a large image of a broom leaning against a textured wall. Underneath the image, there are two buttons: "Clarifai Demo" and "Configure". At the bottom of the interface, the text "GENERAL-V1.3" is displayed, followed by a grid of classification options in small circles. The visible categories include: wall, dirty, old, no person; architecture, stone, building, dust; rope, rustic, brick, ancient, soil.



**jacamar**



# Adversarial Example on Clarifai.com

- Ground truth: **broom**
- Target label: **jacamar**

Clarifai Demo [Configure](#)

---

GENERAL-V1.3

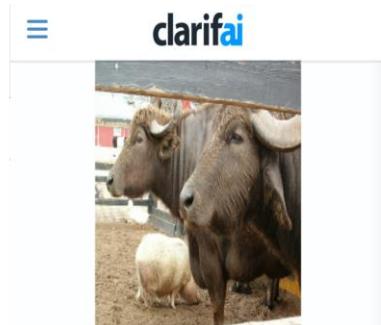


bird nature desktop color art tree  
pattern bright feather painting texture  
design decoration flora no person  
beautiful leaf garden old illustration

---

# Clarifai.com

Ground truth on ImageNet: Waterbuffalo



The screenshot shows the Clarifai demo interface. At the top, there's a navigation bar with three horizontal lines and the word "clarifai". Below the image, there are two buttons: "Clarifai Demo" and "Configure". Underneath the image, the text "GENERAL-V1.3" is displayed. At the bottom, there are four rows of tags in rounded rectangles: "cattle", "agriculture", "livestock", "animal"; "bull", "horn", "cow", "mammal", "farm"; "rural", "herd", "nature", "field", "milk"; and "grass", "countryside", "farmland", "pasture".



rugby ball



# Adversarial Example on Clarifai.com

- Ground truth: **water buffalo**
- Target label: **rugby ball**

Clarifai Demo [Configure](#)

---

GENERAL-V1.3



pastime print illustration art nature  
animal color ball old man one  
vintage sport game people

---

# Clarifai.com

Ground truth from ImageNet: rosehip

The image shows the Clarifai demo interface. At the top, there is a logo with three horizontal bars and the word "clarifai". Below the logo is a photograph of two red rosehips on a stem. To the left of the image is the text "Clarifai Demo" and to the right is a blue "Configure" button. Below the image is the text "GENERAL-V1.3". Underneath this, there are two rows of four buttons each, all of which are white with black text. The first row contains: "no person", "nature", "wildlife", and "little". The second row contains: "fall", "fruit", "food", and "outdoors".



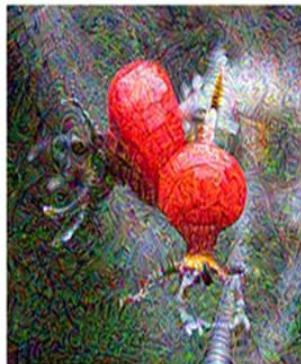
stupa



LCLS17. Delving into Transferable Adversarial Examples and Black-box Attacks, ICLR 2017

# Adversarial Example on Clarifai.com

- Ground truth: **rosehip**
- Target label: **stupa**



LCLS17. Delving into Transferable Adversarial Examples and Black-box Attacks, ICLR 2017

# Black-box attacks

- Zero-Query Attack (Previous methods)
  - Random perturbation
  - Difference of means
  - Transferability based attack
- Query Based Attack (Our methods)
  - Finite difference gradient estimation
  - Query reduced gradient estimation

*The zero-query attack can be viewed as a special case for the query based attack, where the number of queries made is zero*

# Query Based attacks

- Finite difference gradient estimation
  - Given  $d$ -dimensional vector  $\mathbf{x}$ , we can make  $2d$  queries to estimate the gradient as below

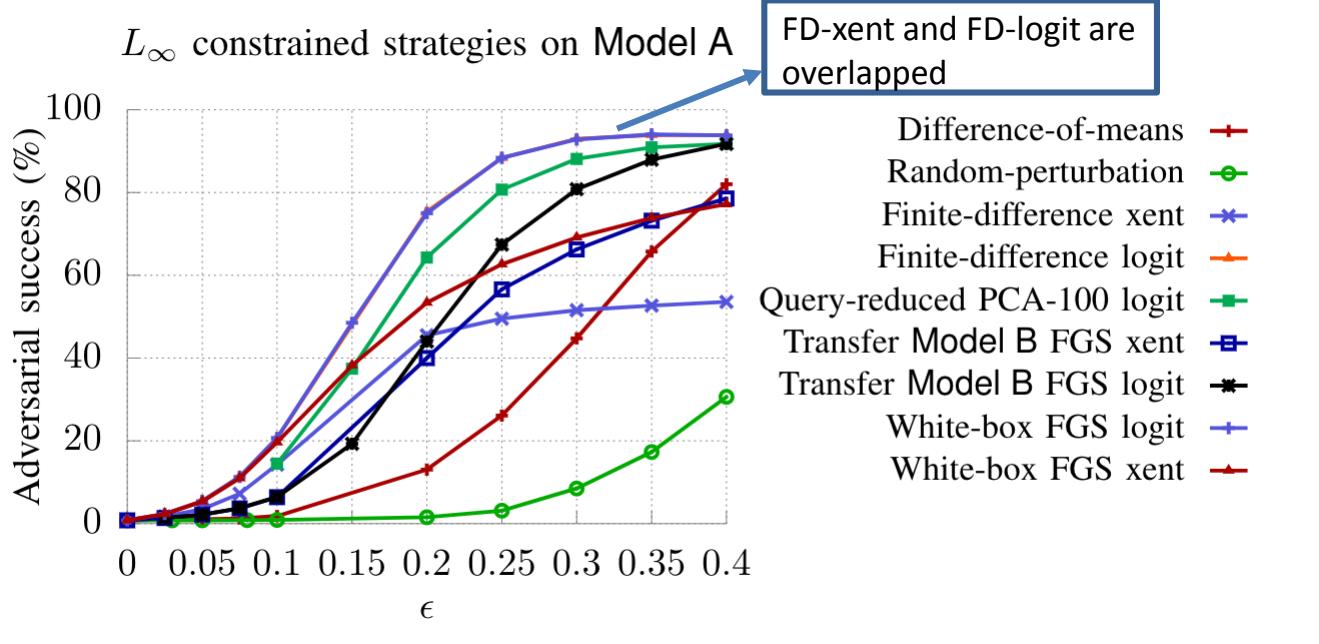
$$\text{FD}_{\mathbf{x}}(g(\mathbf{x}), \delta) = \begin{bmatrix} \frac{g(\mathbf{x} + \delta \mathbf{e}_1) - g(\mathbf{x} - \delta \mathbf{e}_1)}{2\delta} \\ \vdots \\ \frac{g(\mathbf{x} + \delta \mathbf{e}_d) - g(\mathbf{x} - \delta \mathbf{e}_d)}{2\delta} \end{bmatrix}$$

- An example of approximate FGS with finite difference

$$x_{adv} = \mathbf{x} + \epsilon \cdot \text{sign}(\text{FD}_{\mathbf{x}}(\ell_f(\mathbf{x}, y), \delta))$$

- Query reduced gradient estimation
  - Random grouping
  - PCA

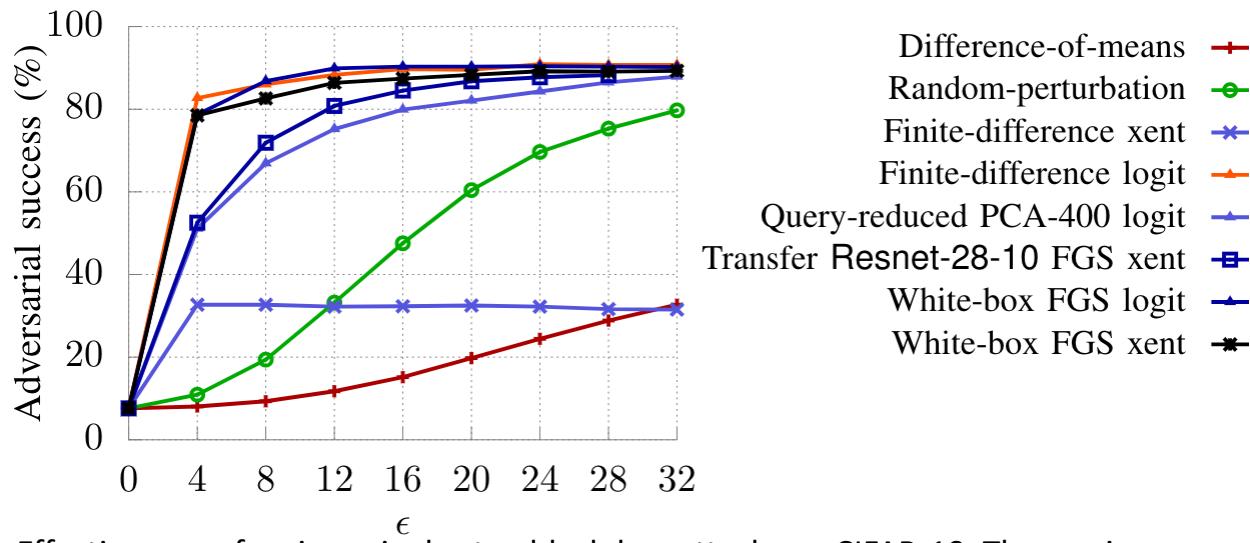
Similarly, we can also approximate for logit-based loss by making  $2d$  queries



Effectiveness of various single step black-box attacks on MNIST. The y-axis represents the variation in adversarial success as  $\epsilon$  increases.

Finite Differences method outperform other black-box attacks and achieves similar attack success rate with the white-box attack

## $L_\infty$ constrained strategies on Resnet-32

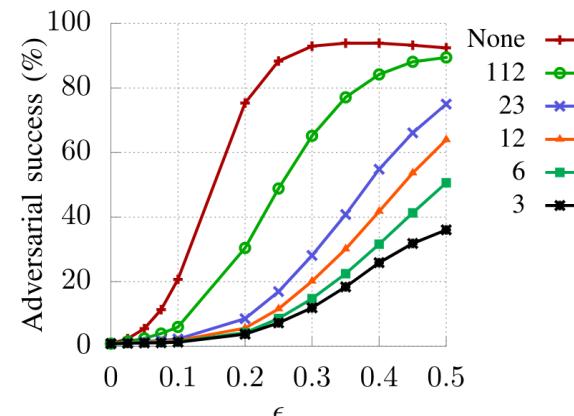


$\epsilon$   
Effectiveness of various single step black-box attacks on CIFAR-10. The y-axis represents the variation in adversarial success as  $\epsilon$  increases.

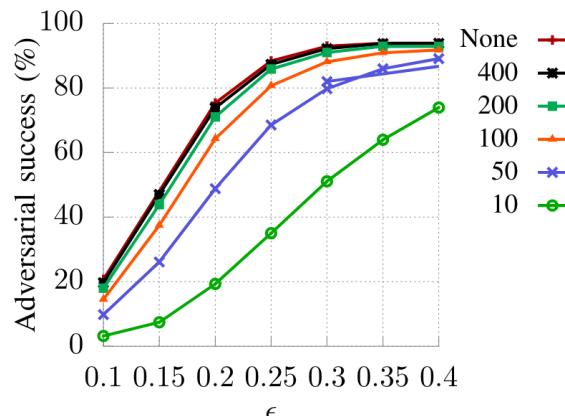
Finite Differences method outperform other black-box attacks and achieves similar attack success rate with the white-box attack

# Gradient Estimation Attack with Query Reduction

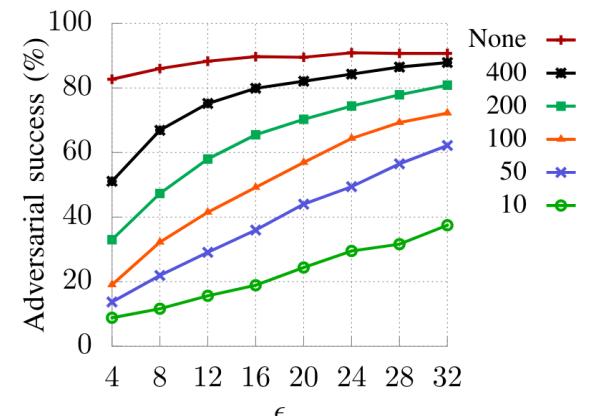
Random feature groupings for Model A



PCA-based query reduction for Model A



PCA-based query reduction for Resnet-32



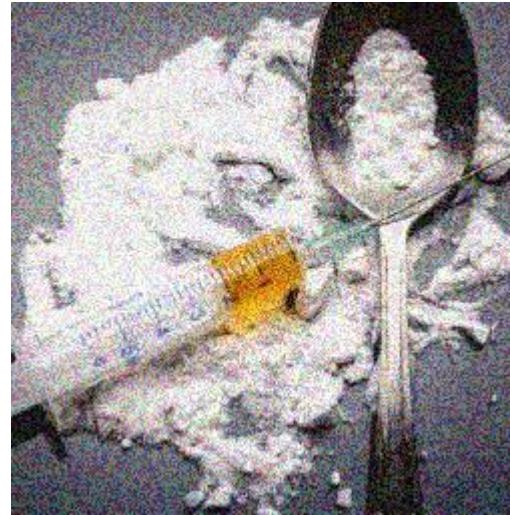
Adversarial success rates for Gradient Estimation attacks with query reduction on Model A (MNIST) and Resnet-32 (CIFAR-10).

Finite Differences method with query reduction perform approximately similar with the gradient estimation black-box attack

# Black-box Attack Clarifai



Original image, classified as “drug”  
with a confidence of 0.99



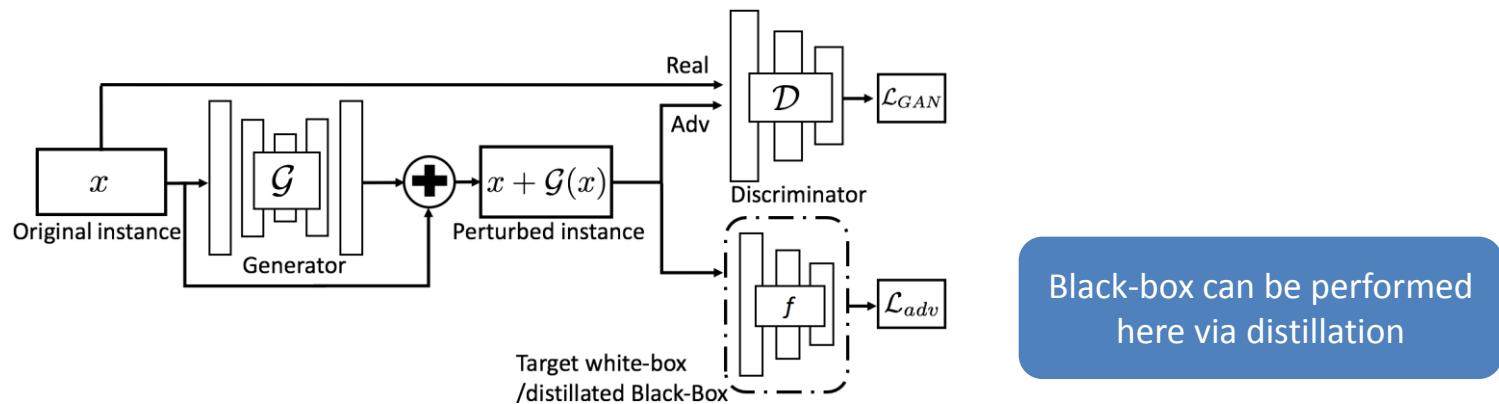
Adversarial example, classified as  
“safe” with a confidence of 0.96

The Gradient Estimation black-box attack on Clarifai’s Content Moderation Model

# Black-box Attacks

**Black-box attacks are possible on deep neural networks with query access.**  
**The number of queries needed can be reduced.**

# Generating Adversarial Examples with Adversarial Networks

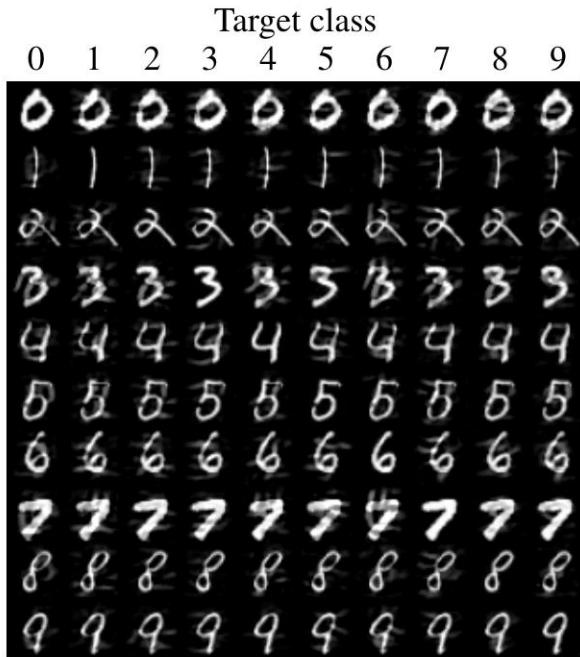


$$\mathcal{L}_{GAN} = \mathbb{E}_{x \sim \mathcal{P}_{\text{data}}(x)} \log \mathcal{D}(x) + \mathbb{E}_{x \sim \mathcal{P}_{\text{data}}(x)} \log(1 - \mathcal{D}(x + \mathcal{G}(x)))$$

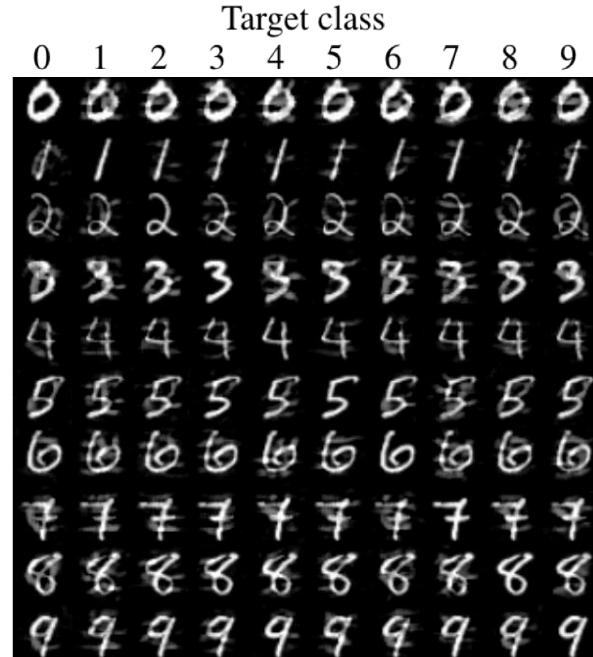
$$\mathcal{L} = \mathcal{L}_{adv}^f + \alpha \mathcal{L}_{GAN} + \beta \mathcal{L}_{hinge}$$

*The GAN loss here tries to ensure the diversity of adversarial examples*

[Chaowei Xiao, Bo Li, Jun-yan Zhu, Warren He, Mingyan Liu, Dawn Song, 2017]

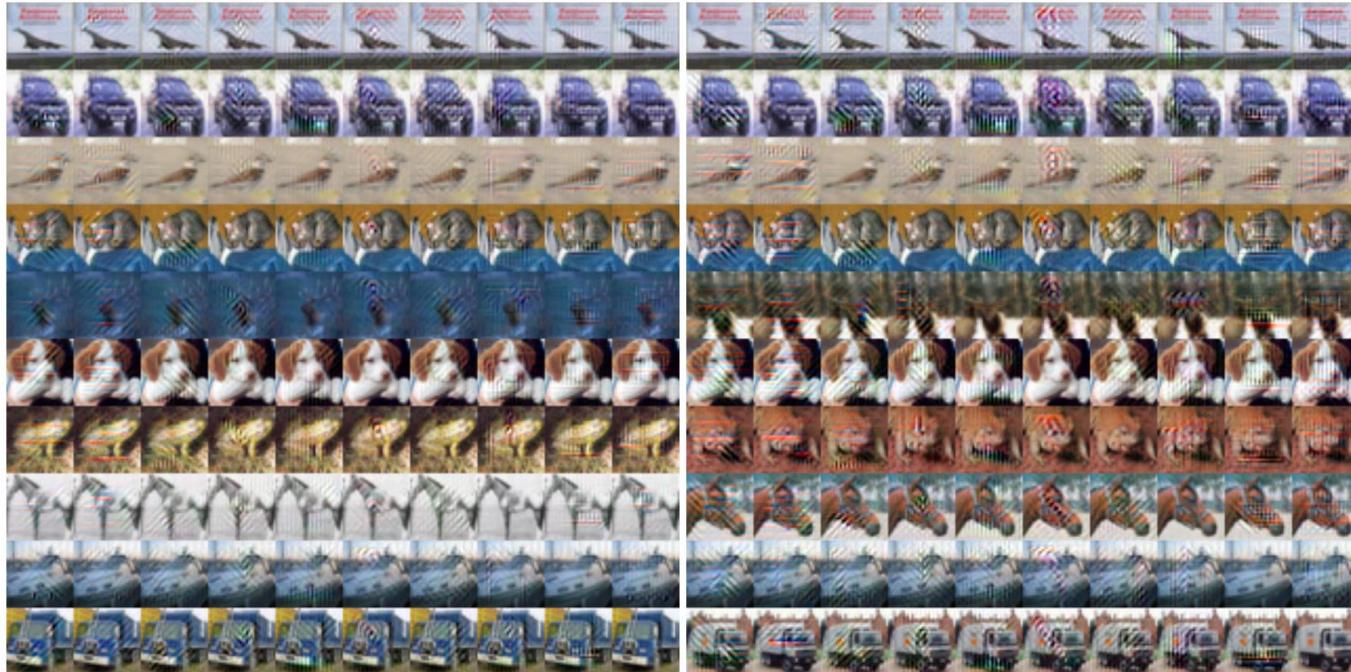


Semi-white box attack on MNIST



Black-box attack on MNIST

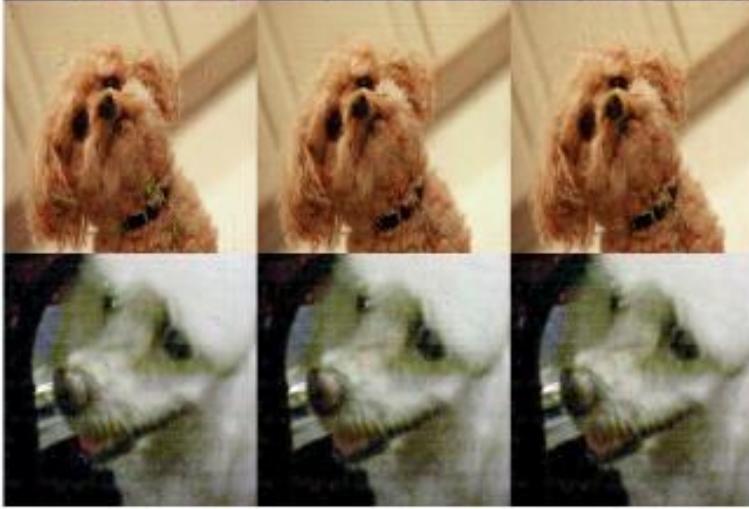
The perturbed images are very close to the original ones. The original images lie on the diagonal.



(a) Semi-whitebox setting

(b) Black-box setting

The perturbed images are very close to the original ones. The original images lie on the diagonal.



Poodle

Ambulance

Basketball

Electric guitar



(a) Strawberry



(b) Toy poodle



(c) Buckeye



(d) Toy poodle

# Attack Effectiveness Under Defenses

Data	Model	Defense	FGSM	Opt.	AdvGAN
MNIST	A	Adv.	4.3%	4.6%	<b>8.0%</b>
		Ensemble	1.6%	4.2%	<b>6.3%</b>
		Iter.Adv.	4.4%	2.96%	<b>5.6%</b>
	B	Adv.	6.0%	4.5%	<b>7.2%</b>
		Ensemble	2.7%	3.18%	<b>5.8%</b>
		Iter.Adv.	<b>9.0%</b>	3.0%	6.6%
	C	Adv.	2.7%	2.95%	<b>18.7%</b>
		Ensemble	1.6%	2.2%	<b>13.5%</b>
		Iter.Adv.	1.6%	1.9%	<b>12.6%</b>
CIFAR	ResNet	Adv.	13.10%	11.9%	<b>16.03%</b>
		Ensemble.	10.00%	10.3%	<b>14.32%</b>
		Iter.Adv	22.8%	21.4%	<b>29.47%</b>
	Wide ResNet	Adv.	5.04%	7.61%	<b>14.26%</b>
		Ensemble	4.65%	8.43%	<b>13.94 %</b>
		Iter.Adv.	14.9%	13.90%	<b>20.75%</b>

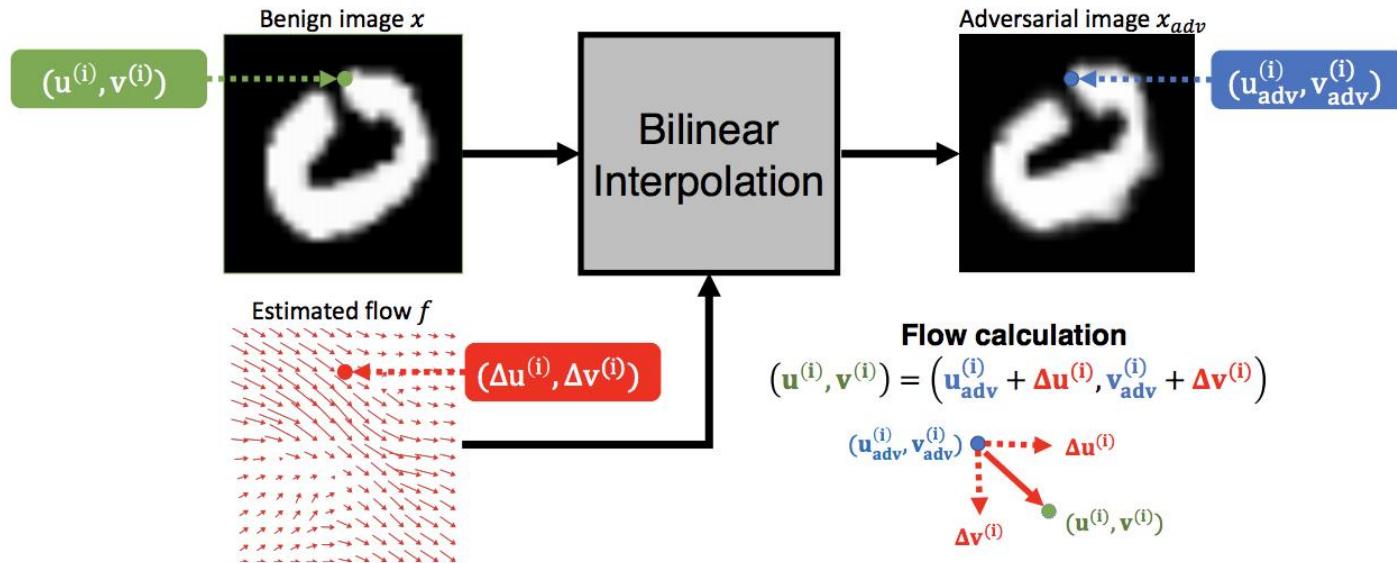
Attack success rate of adversarial examples generated by AdvGAN in semi-whitebox setting under defenses on MNIST and CIFAR-10

# Attack Effectiveness Under Defenses

## Black-Box Leaderboard (Original Challenge)

Attack	Submitted by	Accuracy	Submission Date
AdvGAN from " <a href="#">Generating Adversarial Examples with Adversarial Networks</a> "	AdvGAN	92.76%	Sep 25, 2017
PGD against three independently and adversarially trained copies of the network	<a href="#">Florian Tramèr</a>	93.54%	Jul 5, 2017
FGSM on the <a href="#">CW</a> loss for model B from " <a href="#">Ensemble Adversarial Training [...]</a> "	<a href="#">Florian Tramèr</a>	94.36%	Jun 29, 2017
FGSM on the <a href="#">CW</a> loss for the naturally trained public network	(initial entry)	96.08%	Jun 28, 2017
PGD on the cross-entropy loss for the naturally trained public network	(initial entry)	96.81%	Jun 28, 2017
Attack using Gaussian Filter for selected pixels on the adversarially trained public network	Anonymous	97.33%	Aug 27, 2017
FGSM on the cross-entropy loss for the adversarially trained public network	(initial entry)	97.66%	Jun 28, 2017
PGD on the cross-entropy loss for the adversarially trained public network	(initial entry)	97.79%	Jun 28, 2017

# Spatially Transformed Adversarial Examples



$$f^* = \operatorname{argmin}_f \quad \mathcal{L}_{adv}(x, f) + \tau \mathcal{L}_{flow}(f),$$

[Xiao, Zhu, Li, He, Liu, Song. ICLR 2018]

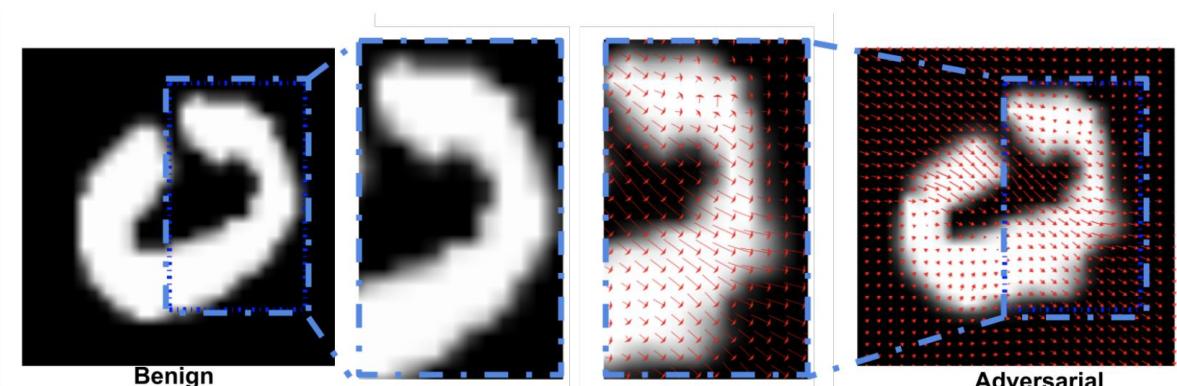
# Examples generated by stAdv

Target class

0 1 2 3 4 5 6 7 8 9



Adversarial examples generated by stAdv on MNIST  
The ground truth images are shown in the diagonal



# Attack Effectiveness Under Defenses

Model	Def.	FGSM	C&W.	stAdv
A	Adv.	4.3%	4.6%	<b>32.62%</b>
	Ens.	1.6%	4.2%	<b>48.07%</b>
	PGD	4.4%	2.96%	<b>48.38%</b>
B	Adv.	6.0%	4.5%	<b>50.17%</b>
	Ens.	2.7%	3.18%	<b>46.14%</b>
	PGD	9.0%	3.0%	<b>49.82%</b>
C	Adv.	3.22%	0.86%	<b>30.44%</b>
	Ens.	1.45%	0.98%	<b>28.82%</b>
	PGD	2.1%	0.98%	<b>28.13%</b>

Model	Def.	FGSM	C&W.	stAdv
ResNet32	Adv.	13.10%	11.9%	<b>43.36%</b>
	Ens.	10.00%	10.3%	<b>36.89%</b>
	PGD	22.8%	21.4%	<b>49.19%</b>
wide ResNet34	Adv.	5.04%	7.61%	<b>31.66%</b>
	Ens.	4.65%	8.43%	<b>29.56%</b>
	PGD	14.9%	13.90%	<b>31.6%</b>

Attack success rate of adversarial examples generated by stAdv against different models under standard defense on MNIST and CIFAR-10

# Attention of network



(a) mountain bike

(b) goldfish

(c) Maltese dog

(d) tabby cat



(e)

(f)

(g)

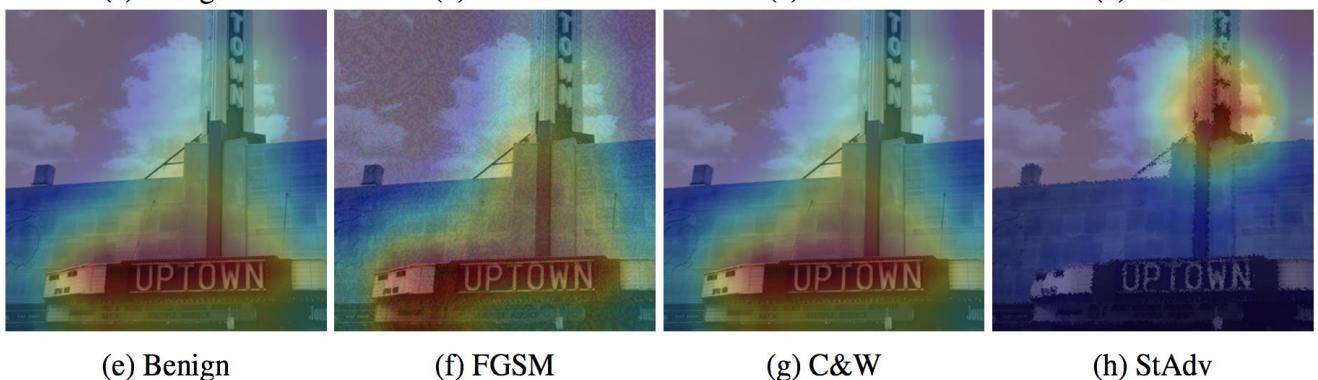
(h)

**CAM attention visualization for ImageNet inception\_v3 model. (a) the original image and (b)-(d) are stAdv adversarial examples targeting different classes. Row 2 shows the attention visualization for the corresponding images above.**

## **inception\_v3 model**



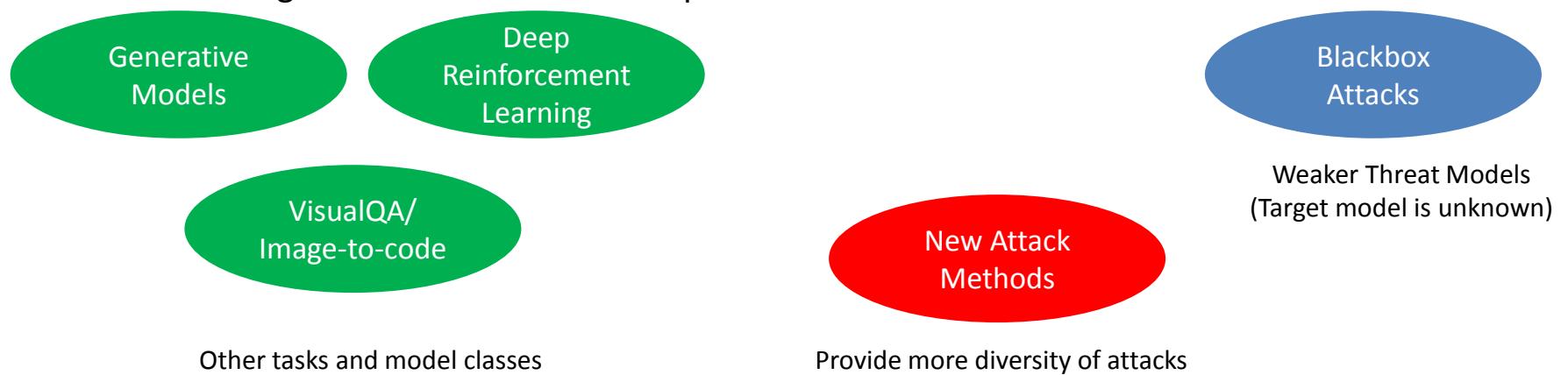
## **Adversarial trained inception\_v3 model**



**CAM attention visualization for ImageNet inception\_v3 model. Column 1 shows the CAM map corresponding to the original image. Column 2-4 show the adversarial examples generated by different methods. (a) and (e)-(g) are labeled as the ground truth “cinema”, while (b)-(d) and (h) are labeled as the adversarial target “missile.”**

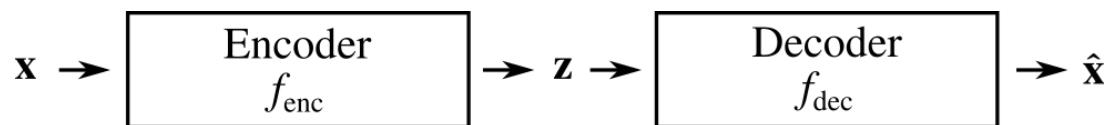
# Adversarial Examples Prevalent in Deep Learning Systems

- Most existing work on adversarial examples:
  - Image classification task
  - Target model is known
- Our investigation on adversarial examples:



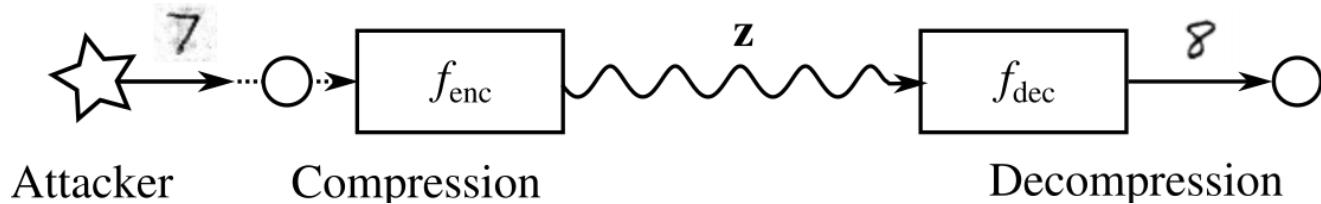
## Generative models

- VAE-like models (VAE, VAE-GAN) use an intermediate latent representation
- An **encoder**: maps a high-dimensional input into lower-dimensional latent representation  $\mathbf{z}$ .
- A **decoder**: maps the latent representation back to a high-dimensional reconstruction.



# Adversarial Examples in Generative Models

- An example attack scenario:
  - Generative model used as a compression scheme



- Attacker's goal: for the decompressor to reconstruct a different image from the one that the compressor sees.

# Adversarial Examples for VAE-GAN in MNIST

7210414959  
0690159784  
9665407401  
3134727121  
1742351244  
6355604195  
7893746430  
7029173297  
1627847361  
3693141769

Original images

7214149596  
9159784964  
5474131347  
2712117423  
5124463556  
4195789374  
0437291732  
9776278473  
6136931417  
6965499219

Adversarial examples

7210414967  
0690159734  
9665407401  
3134727121  
1742351244  
6355604195  
7293796430  
7027173297  
1627847361  
3693141769

Reconstruction of original images

000000000000  
000000000000  
000000000000  
000000000000  
000000000000  
000000000000  
000000000000  
000000000000  
000000000000  
000000000000

Reconstruction of adversarial examples

Target Image



# Adversarial Examples for VAE-GAN in SVHN



Original images



Adversarial examples



Reconstruction of original images



Reconstruction of adversarial examples

Target Image



Jernej Kos, Ian Fischer, Dawn Song: Adversarial Examples for Generative Models

# Adversarial Examples for VAE-GAN in SVHN



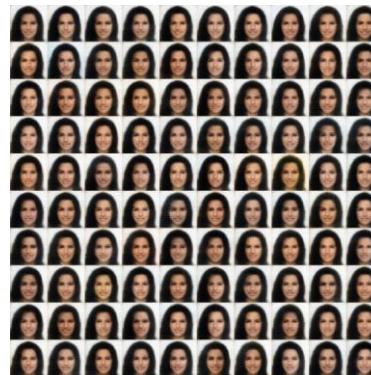
Original images



Adversarial examples



Reconstruction of original images



Reconstruction of adversarial examples

Target Image

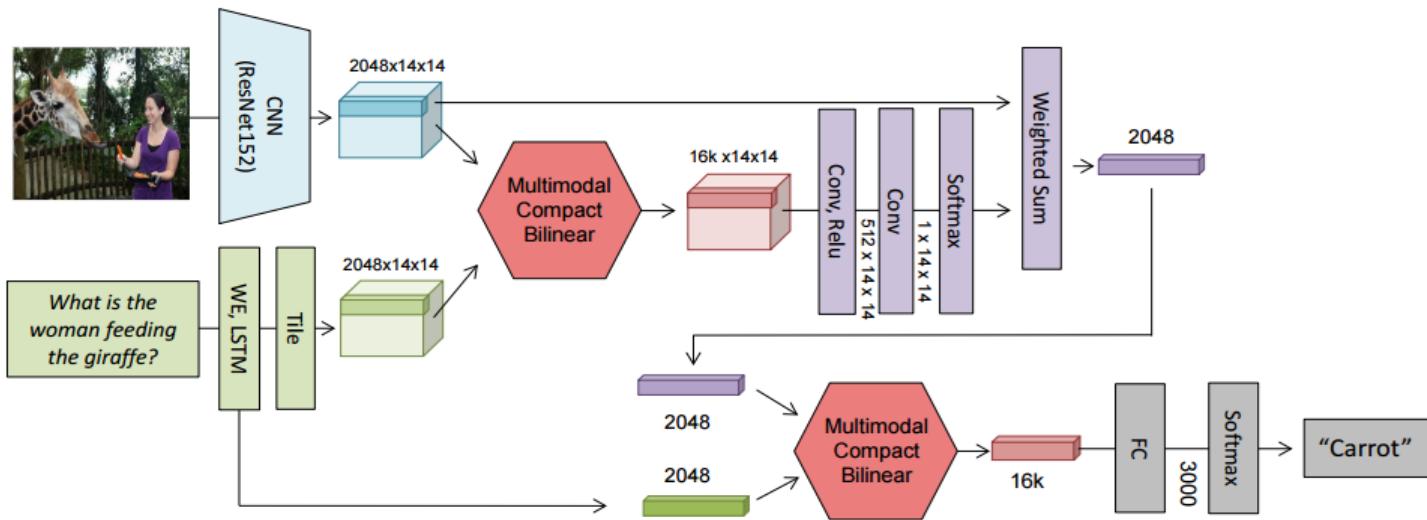


Jernej Kos, Ian Fischer, Dawn Song: Adversarial Examples for Generative Models

# Takeaways

VAE-like **generative models** are **vulnerable** to adversarial examples

# Visual Question & Answer (VQA)



Multimodal Compact Bilinear Pooling for Visual Question Answering and Visual Grounding, Fukui et al., <https://arxiv.org/abs/1606.01847>

Q: Where is  
the plane?



Benign image



Answer:  
Runway

Fooling VQA

Target: Sky



Adversarial example



Sky

Q: How many cats are there?



Benign image



Answer:  
1

Fooling VQA

Target: 2

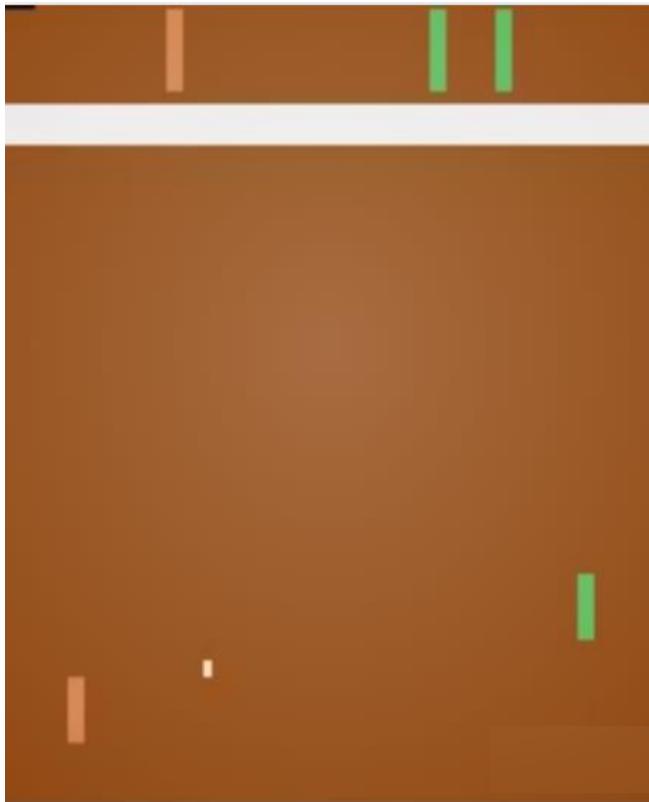


Adversarial example



2

# A3C: A Deep Policy on Pong



Reinforcement learning algorithms:

- Actor – **policy network** to predict the action based on each frame
- Critics – **value function** to predict the value of each frame, and the action is chosen to maximize the expected value
- Actor-critics (A3C) – combine value function into the policy network to make prediction

# Agent in Action: attack the policy network



Original Frames



Adversarial perturbation  
injected into **every frame**

# Agent in Action: attack the value function



Original Frames



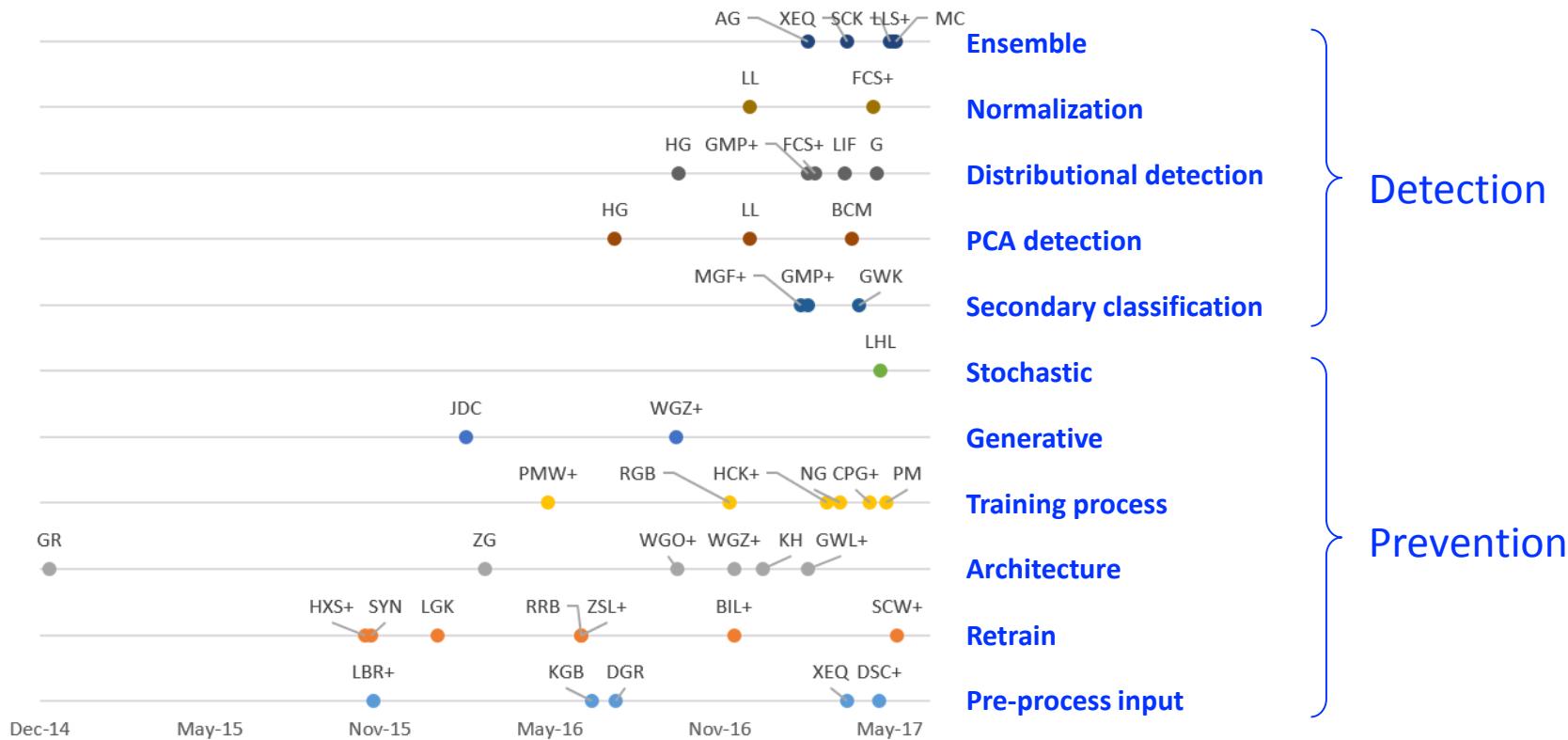
Adversarial perturbation  
injected into **every other 10  
frames**

# Takeaways

**Reinforcement learning** systems (e.g., robotics, self-driving systems) are also **vulnerable** to adversarial examples

To attack a reinforcement learning system, **adversarial perturbations need not be injected to every frame.**

# Numerous Defenses Proposed

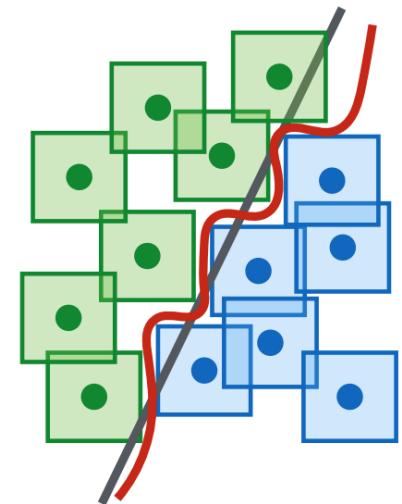
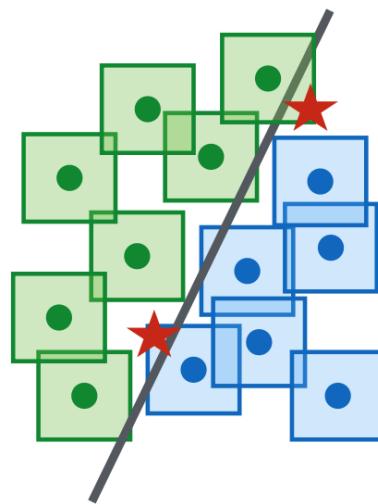
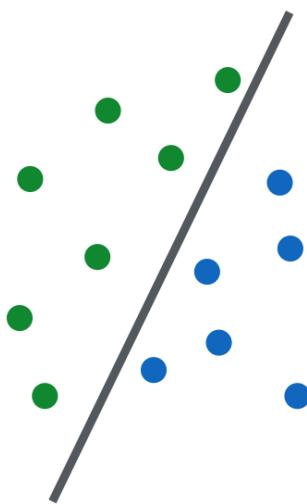


# Towards Deep Learning Models Resistant to Adversarial Attacks

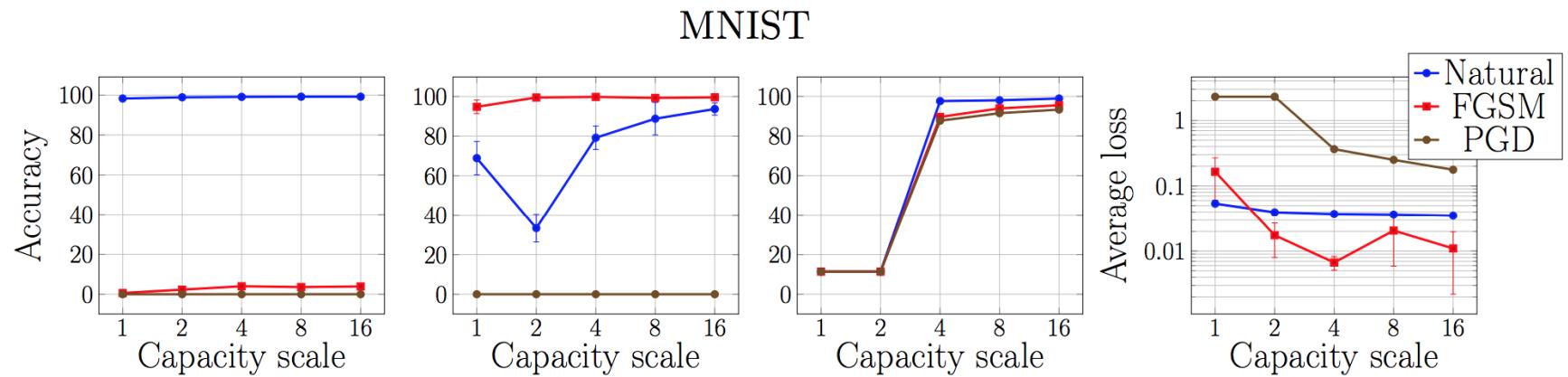
$$\min_{\theta} \rho(\theta), \quad \text{where} \quad \rho(\theta) = \mathbb{E}_{(x,y) \sim \mathcal{D}} \left[ \max_{\delta \in \mathcal{S}} L(\theta, x + \delta, y) \right]$$

- Use a natural saddle point (min-max) formulation to capture the notion of security against adversarial attacks in a principled manner.
- The formulation casts both attacks and defenses into a common theoretical framework.
- Motivate projected gradient descent (PGD) as a universal “first-order adversary”.

# Model Capacity



# Towards Deep Learning Models Resistant to Adversarial Attacks



Benign

OPTBRITTLE

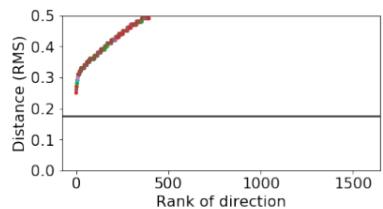
OPTMARGIN  
(ours)

FGSM

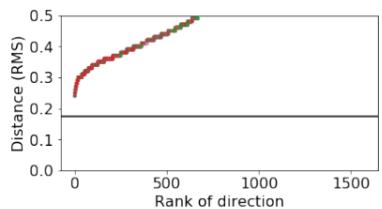
MNIST Test image 3153

CIFAR-10 Test image 5415

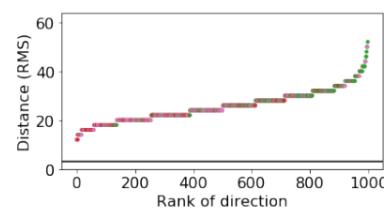
No defense



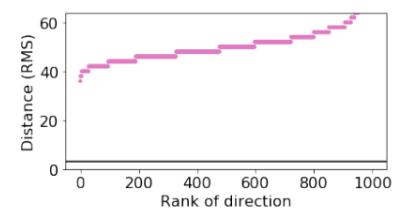
Adv. training



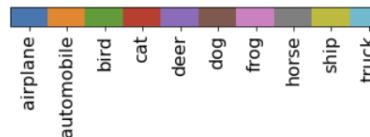
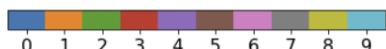
No defense



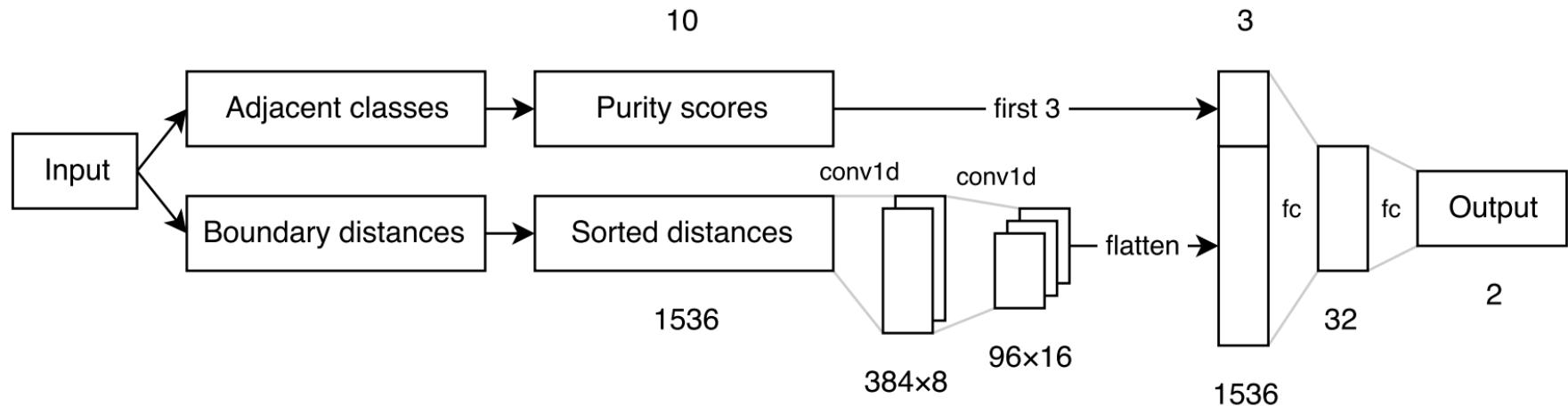
Adv. training



(unsuccessful)



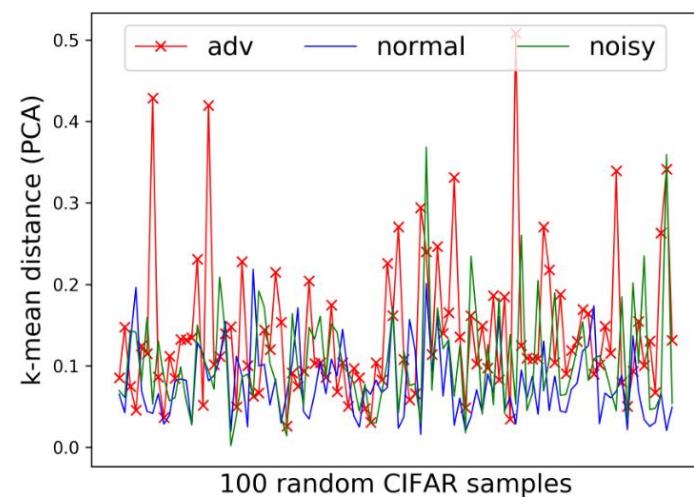
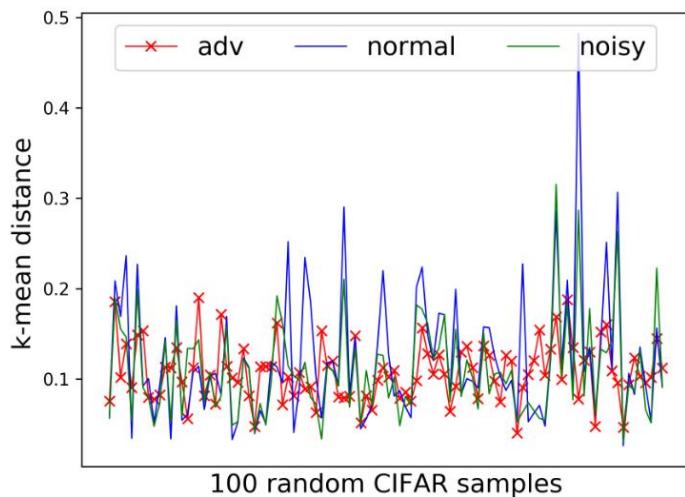
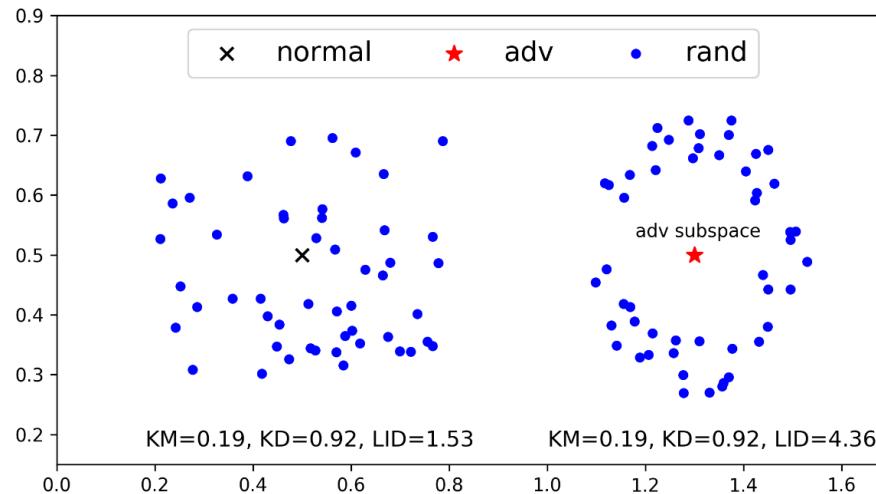
# Decision Boundary Analysis of Adversarial Examples



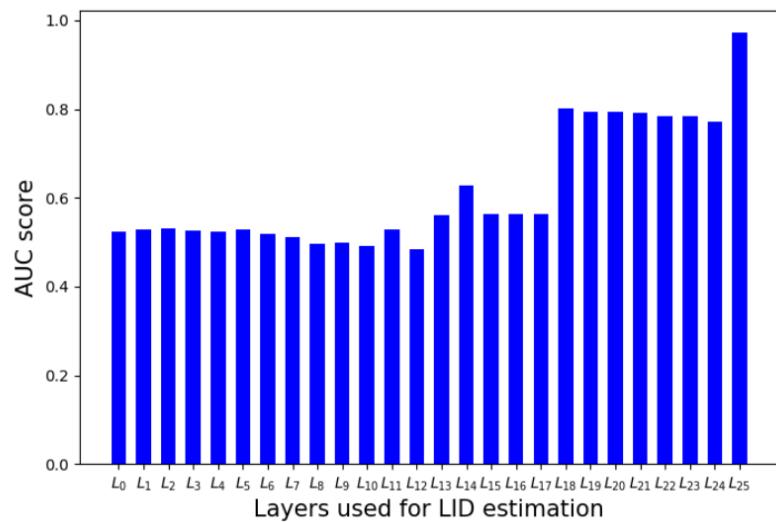
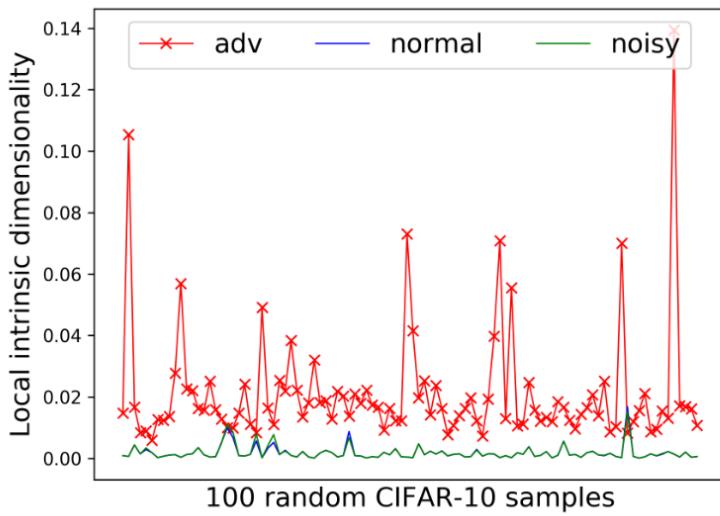
He, Li, Song, Decision Boundary Analysis of Adversarial Examples, ICLR 2017.

Training attack	False pos.		False neg.		Accuracy	
	Benign	OPTBRITTLE	OPTMARGIN	Our approach	Cao & Gong	
MNIST, normal training						
OPTBRITTLE	1.0%	1.0%	74.1%			
OPTMARGIN	<b>9.6%</b>	0.6%	7.2%	90.4%		10%
MNIST, PGD adversarial training						
OPTBRITTLE	2.6%	2.0%	39.8%			
OPTMARGIN	10.3%	0.4%	14.5%			
CIFAR-10, normal training						
OPTBRITTLE	5.3%	3.2%	56.8%			
OPTMARGIN	8.4%	7.4%	5.3%	96.4%		5%
CIFAR-10, PGD adversarial training						
OPTBRITTLE	0.0%	2.4%	51.8%			
OPTMARGIN	<b>3.6%</b>	0.0%	1.2%			

# Adversarial Examples Detection

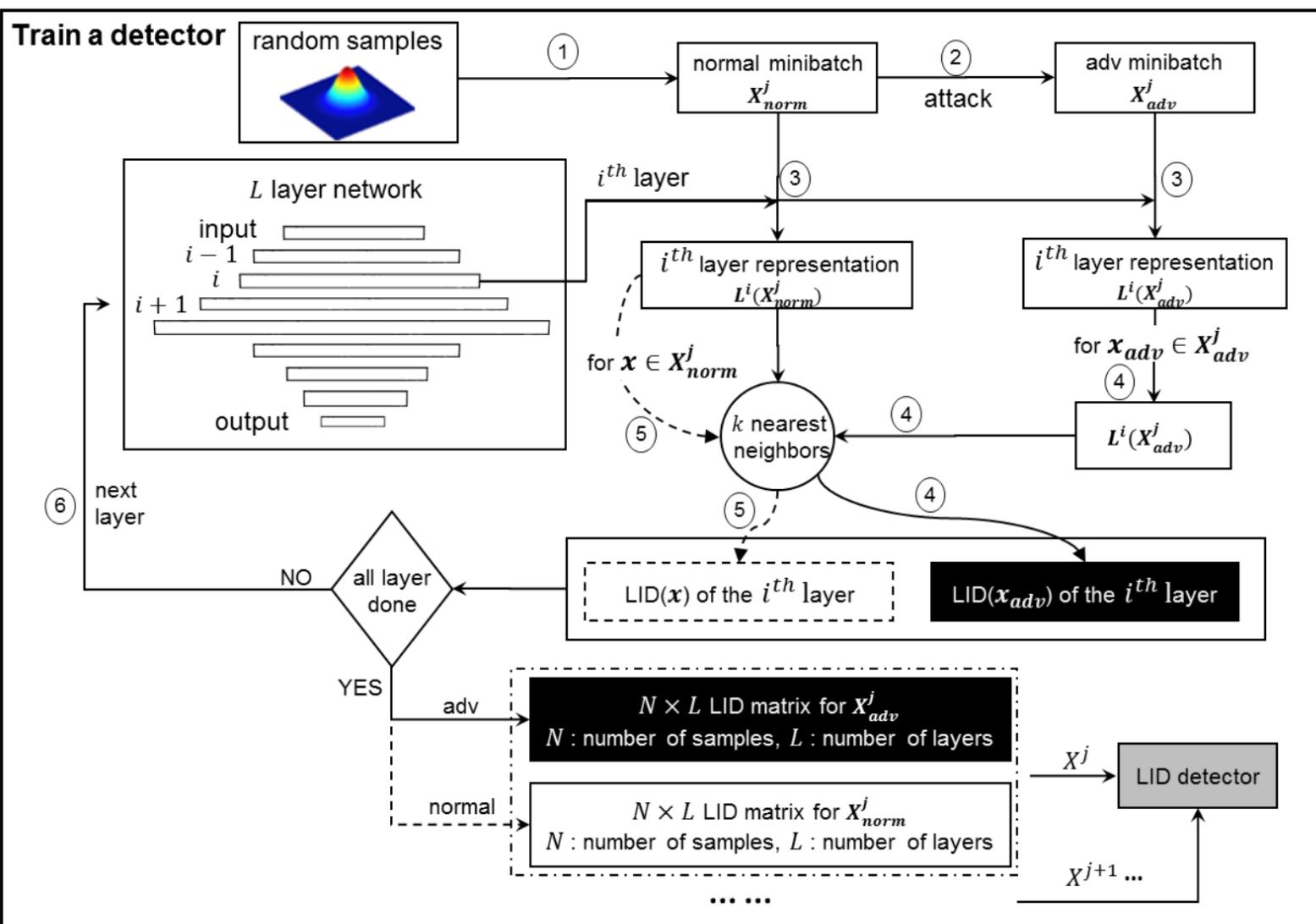


# Adversarial Examples Detection via Local Intrinsic Dimensionality (LID)



$$\widehat{\text{LID}}(x) = - \left( \frac{1}{k} \sum_{i=1}^k \log \frac{r_i(x)}{r_k(x)} \right)^{-1}$$

## Train a detector



# Characterizing Adversarial Examples

Dataset	Feature	FGM	BIM-a	BIM-b	JSMA	Opt
MNIST	KD	78.12%	99.14%	98.61%	68.77%	95.15%
	BU	32.37%	91.55%	25.46%	88.74%	71.29%
	KD+BU	82.43%	99.20%	98.81%	90.12%	95.35%
	LID	<b>96.89%</b>	<b>99.60%</b>	<b>99.83%</b>	<b>92.24%</b>	<b>99.24%</b>
CIFAR-10	KD	64.92%	68.38%	98.70%	85.77%	91.35%
	BU	70.53%	81.60%	97.32%	87.36%	91.39%
	KD+BU	70.40%	81.33%	98.90%	88.91%	93.77%
	LID	<b>82.38%</b>	<b>82.51%</b>	<b>99.78%</b>	<b>95.87%</b>	<b>98.93%</b>
SVHN	KD	70.39%	77.18%	99.57%	86.46%	87.41%
	BU	86.78%	84.07%	86.93%	91.33%	87.13%
	KD+BU	86.86%	83.63%	99.52%	93.19%	90.66%
	LID	<b>97.61%</b>	<b>87.55%</b>	<b>99.72%</b>	<b>95.07%</b>	<b>97.60%</b>

	MNIST	CIFAR-10	SVHN
Attack Failure Rate (one-layer)	100%	95.7%	97.2%
Attack Failure Rate (all-layer)	100%	100%	100%