MagNet: a Two-Pronged Defense against Adversarial Examples

Dongyu Meng, ShangHaiTech University

Hao Chen, University of California Davis

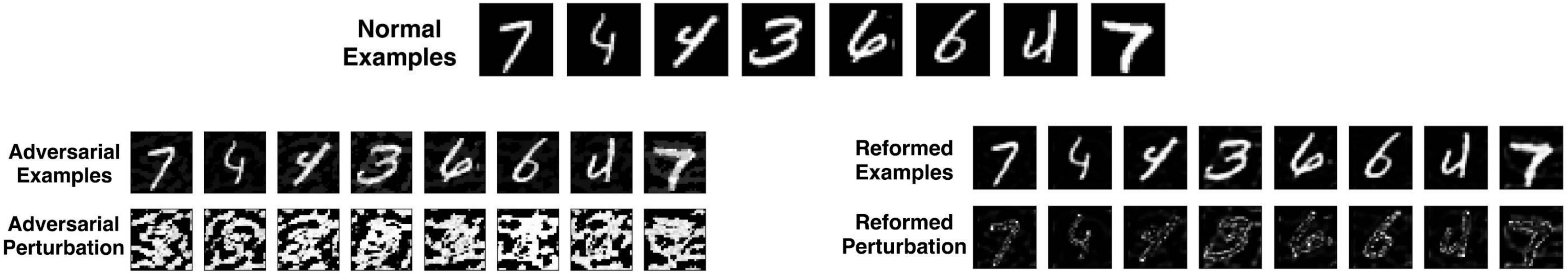
ACM CCS 2017

Web Security & Privacy Lab

KAIST

Problems and Solutions :

- Problem : How to defend Machine learning models against adversarial attacks ?
- Proposed solution : MagNet , a framework that detects or reforms adversarial examples using two types of networks.
- Results : Successfully protected the targeted network from the attack by detecting or reforming the adversarial example.



Contributions of the paper :

- Formally defines adversarial examples :
  - Including metrics for evaluating defenses
- Proposes general defense against adversarial examples :
  - MageNet is independent to the target classifier it protects
  - MageNet is independent to the process of adversarial examples generation
- Proposes Gray-box attack model :
  - Gives an example of black-box attack
  - Claims that Gray-box is a reasonable attack level to consider for defenses
  - Proposes diversity-based defense against Gray-box attacks

Meaning of the paper :

- The authors could build defense elements corresponding to each causes of adversarial attacks.

- The resulting defense, which is a combination of a detector and a reformer, provides a simple but strong protection against known adversarial example generators

Adversarial examples:

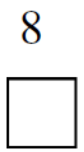
- Normal Example :
  - Occur naturally : examples generated do not differ from the classifying task
- Adversarial Example:
  - Is not a normal example
  - Humans judgement will differ from the classifier's decision

Definition of adversarial examples

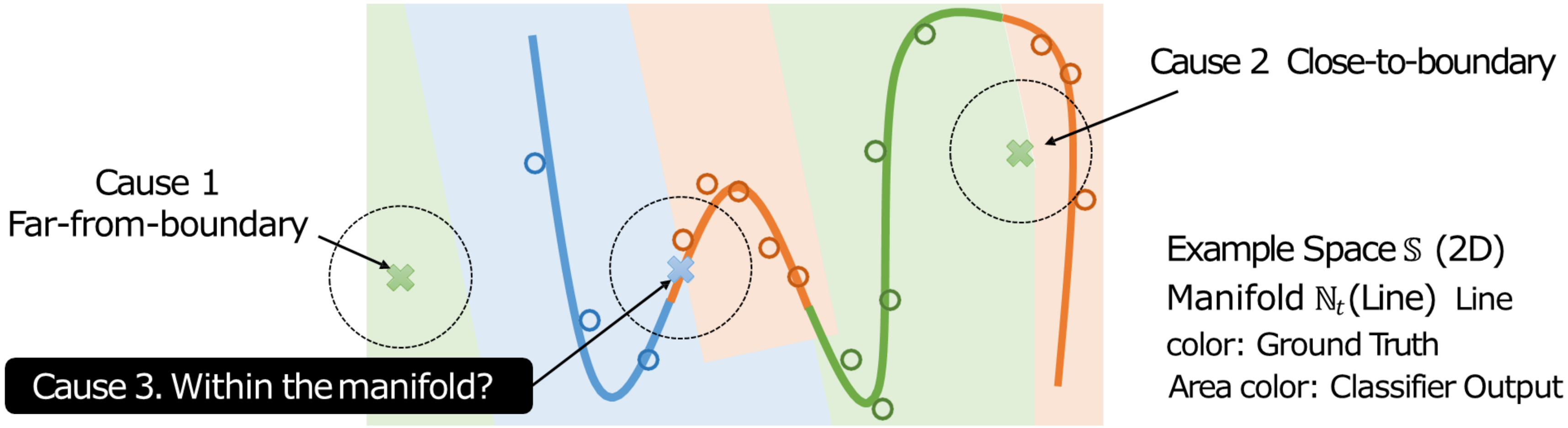
- $\mathbb{S}$: Set of all examples (all the images or the handwritten digits)
- $\mathbb{C}_t$: Set of classes for the task t (e.g $\mathbb{C}_t$={0,1,2,…,9})
- Normal examples for the task t : $\mathbb{N}_t=\{x \mid x \in \mathbb{S}$ and p(x) is non-negligible}
- p(x) is the probability of the natural generation process for the task to emit x.
- Classifier for a task t is $f_t : \mathbb{S} \rightarrow \mathbb{C}_t$
- Ground Truth for a task t is $g_t : \mathbb{S} \rightarrow \mathbb{C}_t \cup \{\perp\}$
- Adversarial example for x for task t and a classifier f_t :
  - $x \in \mathbb{S} \setminus \mathbb{N}$ (Not a normal example)
  - $f_t(x) \neq g_t(x)$ (Classification different from ground-truth)

Causes of mis-classification (adversarial example)

1) The adversarial example is far from the boundary of the manifold of the task (e.g. blank image classified as a handwritten digit)



2) The adversarial example is close to the boundary of the manifold.



Distance metrics :

- Definition : adversarial examples and normal examples should be visually indistinguishable for the human perception.
- Usage of L_p norm to model the human perception

$$\|x\|_p = \left(\sum_{i=1}^{n} |x_i|^p\right)^{\frac{1}{p}}$$

- Closest L_p norm to the human perception process are : L_1, L_2 and L_∞

Existing attacks :

- Fast Gradient Sign Method (FGSM)
  - One-step generation based on gradient sign of loss function
  - Every data (pixels in the case of images) become either incremented or decremented by small ϵ.

$$x' = x + \epsilon \cdot sign(\nabla_x Loss(x, l_x))$$

- Iterative Gradient Sign Method
  - Using smaller steps than FGSM, and iterates the process.
  - Clips according to distance ϵ, ensuring the perturbation within that boundary.

$$x'_{i+1} = clip_{\epsilon,x}(x'_i + \alpha \cdot sign(\nabla_x Loss(x, l_x)))$$

Existing attacks :

- DeepFool
  - Finds nearest boundary and performs a variant of Newton's method.
  - Iterative method, which extended into multiclass differentiable classifiers

- Carlini's Attack
  - Solves an optimization problem on δ, minimizing $||\delta|| + c \cdot f(x+\delta)$
  - Confidence κ is picked to change the confidence level of the adversarial example.
  - c : Balancing hyperparameter

Existing defenses :

- Adversarial Training
  - Augments adversarial data with correct classes, to the training input.
- Defensive Distillation
  - Hides the gradient between the pre-softmax layer and the outputs.
  - Bypasses: proper loss function, calculation on pre-softmax layer, …
- Adversarial Example Detection
  - Detectors: Binary classifiers to decide adversarial inputs
  - MagNet : Learns manifolds of the normal examples and uses multiple detectors and reformers.

Formal definition of the defense

- Defense against adversarial examples for a classifier f_t : $d_{f_t} : \mathbb{S} \rightarrow \mathbb{C}_t \cup \{\perp\}$
- Three ways d_{f_t} may use f_t :
  - ✓ d_{f_t} does not read the data in f_t and does not modify its parameters
  - ✓ d_{f_t} reads data in f_t and does not modify its parameters
  - ✓ d_{f_t} modifies f_t's parameters
- Successful defense if :
  - ✓ $x \in \mathbb{N}_t$, $d_{f_t}(x) = gt(x)$
  - ✓ $x \in \mathbb{S} \setminus \mathbb{N}_t$ and $(d_{f_t}(x) = \perp$ or $d_{f_t}(x) = g_t(x))$

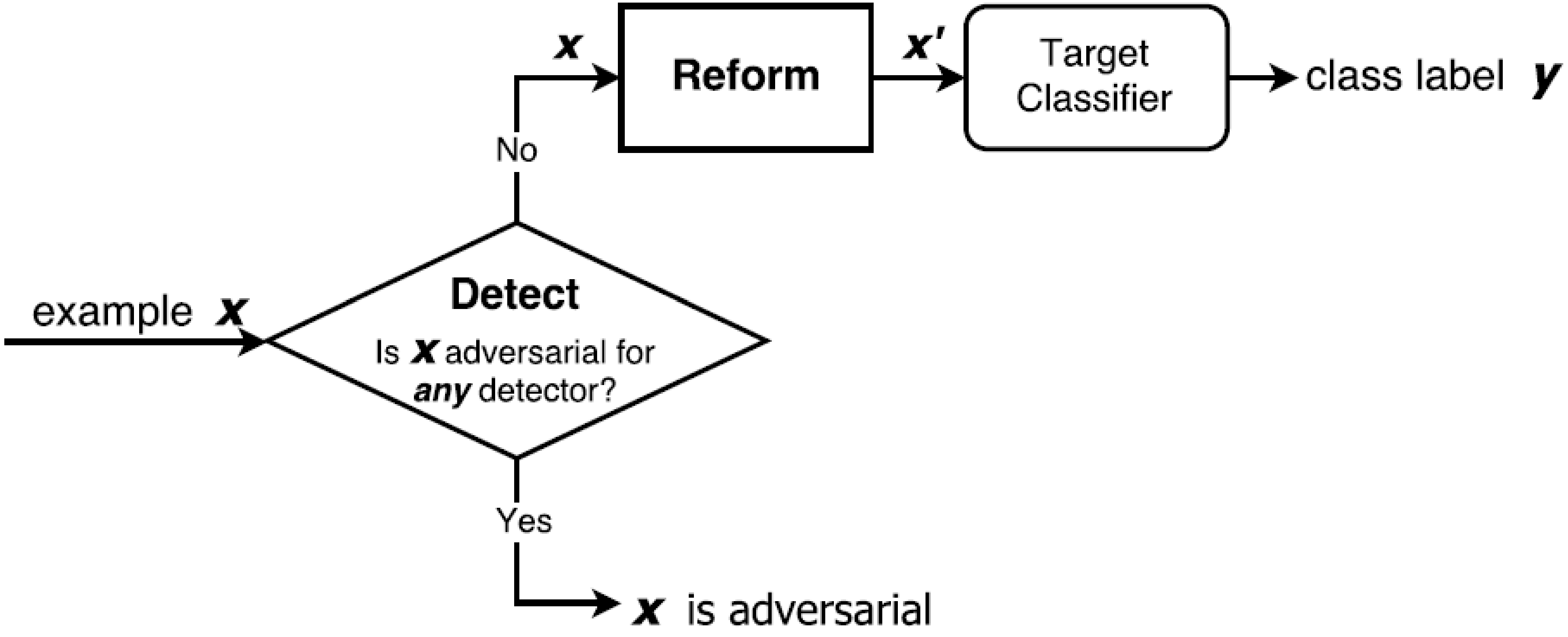
Threat model

- Attacker knows everything about the target classifier (f_t)
- Defender knows nothing about the attacker's generation process
- Different level of knowledge on the defense by the attacker :

Knowledge on d_{ft}	Black-box	Gray-box	White-box
Oracle of d_{ft}	O	O	O
Parameters	X	X	O
Model structure, Hyperparameters, Training set, Number of epochs, other non parameters	X	O	O
Random seed	X	X	O

MagNet design

A. Structure :



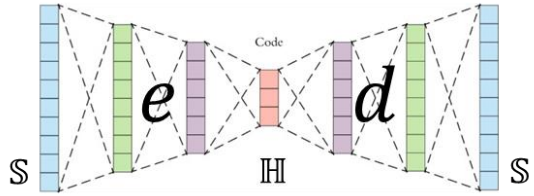
Web Security & Privacy Lab

KAIST

MagNet design

B. Autoencoder (ae) :

- ae = d ∘ e
- Encode e : $\mathbb{S} \rightarrow \mathbb{H}$, Decoder d : $\mathbb{H} \rightarrow \mathbb{S}$ ($\mathbb{H}$ is the hidden representation)

x  $ae(x)$

- Reconstruction error : $E(x) = \|x - ae(x))\|_p$
- Training an autoencoder :
  - Minimize the loss function over the training set

$$L(\mathbb{X}_{\text{train}}) = \frac{1}{|\mathbb{X}_{\text{train}}|} \sum_{x \in \mathbb{X}_{\text{train}}} \|x - ae(x))\|_2$$

Web Security & Privacy Lab

KAIST

MagNet design

C. Detector :

- Detector's function d : $\mathbb{S} \rightarrow \{0,1\}$
- d(x)=0 if x is too far from the manifold (far=distance metric , too=threshold)
- Detector based on reconstruction error :
  - ✓ Do not need adversarial examples for training
  - ✓ Uses the reconstruction error to approximate the distance between test example and boundary of the manifold of the normal example
  - ✓ t_{re} = threshold of reconstruction error to decide far-from-manifold data
  - ✓ Used L_1 and L_2 norm for reconstruction error

MagNet design

- Detector based on probability divergence :
  - ✓ Takes advantage of the target classifier
  - ✓ Compares probability mass function between f(x) and f(ae(x))
  - ✓ Uses Jensen-Shanon divergence on last layer (softmax)
  - ✓ Normal softmax function can saturate if largest element is much largest than the second

$$\text{softmax}(\boldsymbol{l})_i = \frac{\exp(l_i/T)}{\sum_{j=1}^{n} \exp(l_j/T)}$$

T: Temperature

More temperature produces 'smoother' probability.

MagNet design

D. Reformer :

- Reformer's function r : $\mathbb{S} \rightarrow \mathbb{N}_t$, r(x)=x' and $f_t(x') = g_t(x)$
- Ideal reformer should not change the classification results of normal examples and should reform adversarial examples

- Noise-based reformer :
  - ✓ Adds noise from normal distribution and clips the data

$$r(\boldsymbol{x}) = clip(\boldsymbol{x} + \epsilon \cdot \mathbf{y})$$

  - ✓ Problem : changes normal and adversarial examples blindly

MagNet design

D. Reformer :

- <u>Autoencoder-based reformer :</u>
  - ✓ Train AE to minimize reconstruction error on training set
  - ✓ Good generalization of the training set
  - ✓ r(x)=ae(x)
  - ✓ ae(x) = x′ , x′ is very similar to x if x is a normal example
  - ✓ ae(y)=y′ , y′ is very close to the manifold of the normal example is y is an adversarial example.

Is it possible to use the same AE for a detector and reformer ?
Will it be as good as if we were using different AEs from the same architecture?

Diversity to mitigate Graybox attacks

- Introduce randomness to diversify the defense (same style as cryptography)
  - ✓ Train multiples AEs as candidates
  - ✓ For every session, MagNet picks randomly an AE (from a pool) for the defense
  - ✓ Possible countermeasure : the attacker trains his attack on all the AEs however the authors can increase and diversify the pool of AEs available to make it harder

- How to find a large number of diverse AEs ?
  - ✓ Train n autoencoders (same or different architecture) at the same time with random initialization.
  - ✓ Add a regularization term to the loss function to penalize resemblance of the AEs

$$L(x) = \sum_{i=1}^{n} \mathrm{MSE}(x, ae_i(x)) - \alpha \sum_{i=1}^{n} \mathrm{MSE}(ae_i(x), \frac{1}{n} \sum_{j=1}^{n} ae_j(x))$$

How can we satisfy both of these constraints ?

Evaluation step

1. Train target classifiers to be defended by MagNet.
2. Deploy the known attacks to generate adversarial examples.
3. Construct defensive devices using autoencoders.
4. Measure classification accuracy of the normal/adversarial examples.

Classifiers to be defended by MagNet

- CNN models for :
  - CIFAR-10 : Image classification (10 classes)
  - MNIST : 10 handwritten digits

Table 2: Training parameters of classifiers to be protected

Parameters	MNIST	CIFAR
Optimization Method	SGD	SGD
Learning Rate	0.01	0.01
Batch Size	128	32
Epochs	50	350
Data Augmentation	-	Shifting + Horizontal Flip

MNIST		CIFAR	
Conv.ReLU	$3 \times 3 \times 32$	Conv.ReLU	$3 \times 3 \times 96$
Conv.ReLU	$3 \times 3 \times 32$	Conv.ReLU	$3 \times 3 \times 96$
Max Pooling	2×2	Conv.ReLU	$3 \times 3 \times 96$
Conv.ReLU	$3 \times 3 \times 64$	Max Pooling	2×2
Conv.ReLU	$3 \times 3 \times 64$	Conv.ReLU	$3 \times 3 \times 192$
Max Pooling	2×2	Conv.ReLU	$3 \times 3 \times 192$
Dense.ReLU	200	Conv.ReLU	$3 \times 3 \times 192$
Dense.ReLU	200	Max Pooling	2×2
Softmax	10	Conv.ReLU	$3 \times 3 \times 192$
		Conv.ReLU	$1 \times 1 \times 192$
		Conv.ReLU	$1 \times 1 \times 10$
		Global Average Pooling	
		Softmax	10

- Accuracy on MNIST : 99.4%
- Accuracy on CIFAR-10 : 90.4%

Detector and reformer architecture

- For MNIST :

Detector I & Reformer		Detector II	
Conv.Sigmoid	$3 \times 3 \times 3$	Conv.Sigmoid	$3 \times 3 \times 3$
AveragePooling	2×2	Conv.Sigmoid	$3 \times 3 \times 3$
Conv.Sigmoid	$3 \times 3 \times 3$	Conv.Sigmoid	$3 \times 3 \times 1$
Conv.Sigmoid	$3 \times 3 \times 3$		
Upsampling	2×2		
Conv.Sigmoid	$3 \times 3 \times 3$		
Conv.Sigmoid	$3 \times 3 \times 1$		

- For CIFAR-10 :

Detectors & Reformer	
Conv.Sigmoid	$3 \times 3 \times 3$
Conv.Sigmoid	$3 \times 3 \times 3$
Conv.Sigmoid	$3 \times 3 \times 1$

Why would they use simpler Reformers for CIFAR-10 as the images are more complicated than MNIST images ?

Results of MagNet :

Accuracy	Normal Examples		Adversarial Examples	
	No Defense	With MagNet	No Defense	With MagNet
MNIST	99.4%	99.1%	0~96.8%	92.0~100%
CIFAR10	90.6%	86.8%	0~46.0%	76.3~100%

The authors of the paper always talked about having a low false positive rate, however they never showed any results on those rates !

(a) MNIST

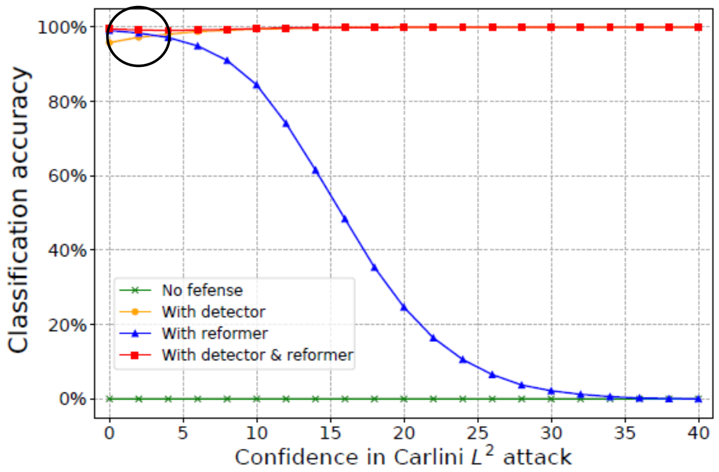
Attack	Norm	Parameter	No Defense	With Defense
FGSM	L^∞	$\epsilon = 0.005$	96.8%	100.0%
FGSM	L^∞	$\epsilon = 0.010$	91.1%	100.0%
Iterative	L^∞	$\epsilon = 0.005$	95.2%	100.0%
Iterative	L^∞	$\epsilon = 0.010$	72.0%	100.0%
Iterative	L^2	$\epsilon = 0.5$	86.7%	99.2%
Iterative	L^2	$\epsilon = 1.0$	76.6%	100.0%
Deepfool	L^∞		19.1%	99.4%
Carlini	L^2		0.0%	99.5%
Carlini	L^∞		0.0%	99.8%
Carlini	L^0		0.0%	92.0%

(b) CIFAR

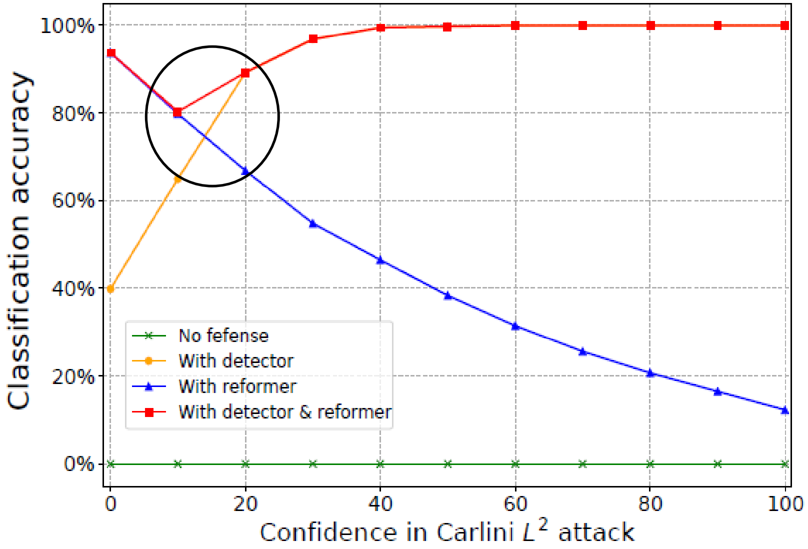
Attack	Norm	Parameter	No Defense	With Defense
FGSM	L^∞	$\epsilon = 0.025$	46.0%	99.9%
FGSM	L^∞	$\epsilon = 0.050$	40.5%	100.0%
Iterative	L^∞	$\epsilon = 0.010$	28.6%	96.0%
Iterative	L^∞	$\epsilon = 0.025$	11.1%	99.9%
Iterative	L^2	$\epsilon = 0.25$	18.4%	76.3%
Iterative	L^2	$\epsilon = 0.50$	6.6%	83.3%
Deepfool	L^∞		4.5%	93.4%
Carlini	L^2		0.0%	93.7%
Carlini	L^∞		0.0%	83.0%
Carlini	L^0		0.0%	77.5%

Case study on Carlini's attack :

MNIST :



CIFAR-10 :



- Reformer is good for low confidence attacks (near manifold)
- Detector is good for high confidence attacks (far from the manifold)
- They are complementary to each other, best accuracy when both are used.

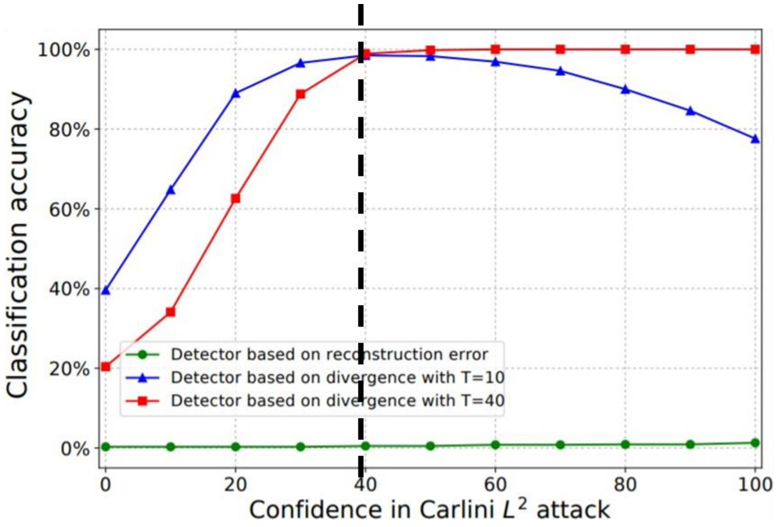
Web Security & Privacy Lab

KAIST

Case study on Carlini's attack :

- Effect of the temperature T on the accuracy of the probability divergence -based detector :

Low-temperature detectors are more accurate on low-confidence attacks



High-temperature detectors are more accurate on high-confidence attacks

Defense against Graybox attacks

- Classification accuracy table for Carlini's attack using 8 AEs for CIFAR-10 task :

	A	B	C	D	E	F	G	H
A	0.0	92.8	92.5	93.1	91.8	91.8	92.5	93.6
B	92.1	0.0	92.0	92.5	91.4	92.5	91.3	92.5
C	93.2	93.8	0.0	92.8	93.3	94.1	92.7	93.6
D	92.8	92.2	91.3	0.0	91.7	92.8	91.2	93.9
E	93.3	94.0	93.4	93.2	0.0	93.4	91.0	92.8
F	92.8	93.1	93.2	93.6	92.2	0.0	92.8	93.8
G	92.5	93.1	92.0	92.2	90.5	93.5	0.1	93.4
H	92.3	92.0	91.8	92.6	91.4	92.3	92.4	0.0
Random	81.1	81.4	80.8	81.3	80.3	81.3	80.5	81.7

- Classification accuracy on test set for cifar-10 :

AE	A	B	C	D	E	F	G	H	Rand
Acc	89.2	88.7	89.0	89.0	88.7	89.3	89.2	89.1	89.0

Sum up of the paper :

- Creation of MagNet : a framework against adversarial perturbation
- Uses two networks : a detector (example far or close from the manifold) and a reformer (recreates the examples)
- Strong against state-of-art attacks
- Defense mechanism should be attack-independent !

Pros	Cons
Simple and effective defense system	Dependent to the model
Tested against state-of-art attacks	Only tested on image datasets
Attack-independant	No explanation on why they used autoencoder for reformers

Questions ?