AI 539: TRUSTWORTHY ML

PRELIMINARIES ON ADVERSARIAL EXAMPLES

Instructor: Sanghyun Hong

sanghyun.hong@oregonstate.edu

NOTES

- Call for actions
  - In-class presentation sign-ups
  - Term project team-up
  - On-boarding quiz on Canvas
  - GitHub classroom registration

NOTES

- Checkpoint Presentation I (on the 20th)
  - 15 min presentation + 3-5 min Q&A
  - Presentation MUST cover:
    - A research problem your team chose
    - A review of the prior work relevant to your problem
      - How is your team's work different from the prior work?
      - What's the paper your team picked and the results your team will reproduce?
    - Next steps (+ how each member will contribute to the work)

ADVERSARIAL EXAMPLES

- A test-time input to a neural network
  - Crafted with the objective of fooling the network's decision(s)

NOT EVERY ADVERSARIAL EXAMPLES ARE INTERESTING

- A test-time input to a neural network
  - Crafted with the objective of fooling the network's decision(s)
  - That looks like a natural test-time input

Noisy test-time input

NOT EVERY ADVERSARIAL EXAMPLES ARE INTERESTING

- A test-time input to a neural network
  - Crafted with the objective of fooling the network's decision(s)
  - That looks like a natural test-time input

Prediction: **Panda**

$+ 0.007 \times$

Human-imperceptible Noise

$=$

Prediction: **Gibbon**

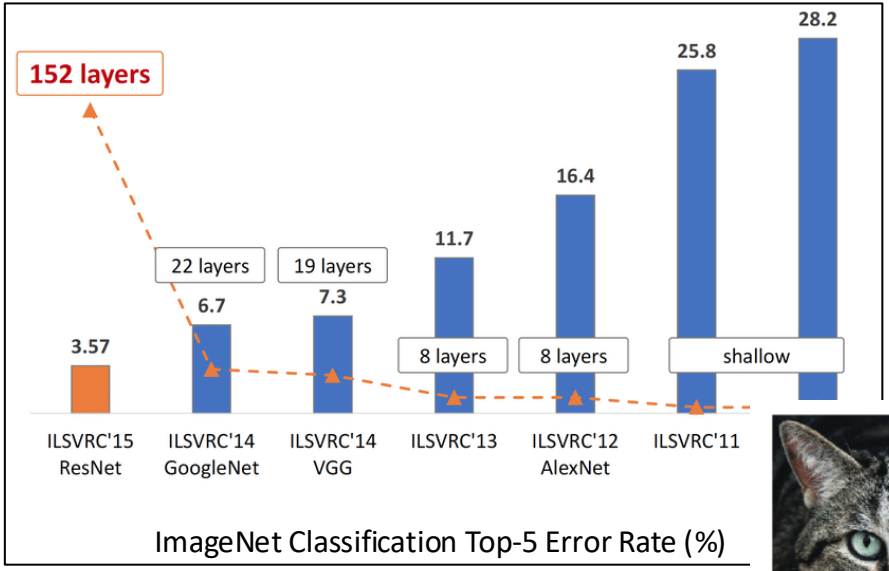
Goodfellow et al., *Explaining and Harnessing Adversarial Examples, International Conference on Learning Representations (ICLR)*, 2015.

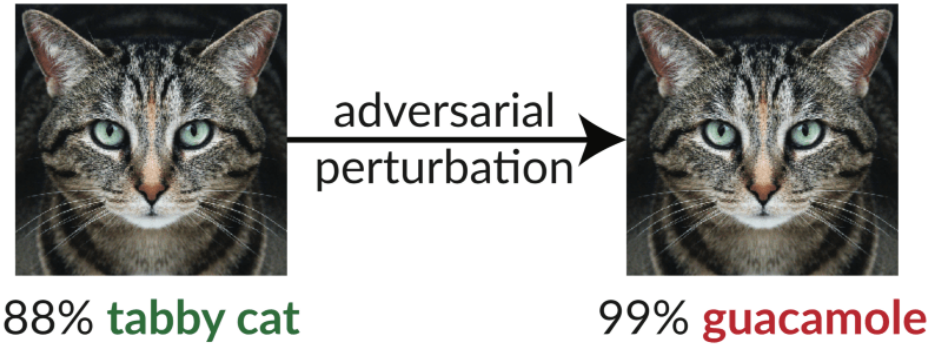
WHY DO THEY MATTER?

- from the security perspective: it makes ML-enabled systems unavailable

WHY DO THEY MATTER?

- from the ML perspective: it is counter-intuitive





TOPICS FOR PART I – ADVERSARIAL EXAMPLES

- Research questions
  - What are the adversarial examples?
  - How can we find adversarial examples?
  - How can we exploit them in practice?
  - How can we defeat adversarial examples?

WHAT ARE THE ADVERSARIAL EXAMPLES?

EXPLAINING AND HARNESSING ADVERSARIAL EXAMPLES, GOODFELLOW ET AL., ICLR 2015

WHAT DID WE BELIEVE AT THAT TIME?

- Two common beliefs about neural networks
  - Neurons represent certain features
    - People use this intuition to find *semantically-similar* inputs
    - Neural networks may have the ability to *disentangle* features at neuron-level
  - Neural Networks are stable when there is small perturbations to their inputs
    - *Random perturbations* to inputs are difficult to change networks' predictions

WHAT DID WE BELIEVE AT THAT TIME?

- Neurons represent certain features
- Re-visit this hypothesis[1]:
  - Find a set of inputs that maximally increases
    - The activation of i-th hidden neuron
    - The activation of random vector
  - Compare those two sets of inputs
  - More formally:

$$x' = \arg\max_{x \in \mathcal{I}} \langle \phi(x), e_i \rangle$$

$$x' = \arg\max_{x \in \mathcal{I}} \langle \phi(x), v \rangle$$

[1] Szegedy et al., Intriguing Properties of Neural Networks, ICLR

WHAT DID WE BELIEVE AT THAT TIME?

(a) Unit sensitive to white flowers.

(b) Unit sensitive to postures.

(c) Unit senstive to round, spiky flowers.

(d) Unit senstive to round green or yellow objects.

Images that activates a certain neuron the most

(a) Direction sensitive to white, spread flowers.

(b) Direction sensitive to white dogs.

(c) Direction sensitive to spread shapes.

(d) Direction sensitive to dogs with brown heads.

Images that activates a random dir. the most

WHAT DID WE BELIEVE AT THAT TIME?

- Neural networks are resilient to *small* input perturbations
- Re-visit this hypothesis[1]:
  - Let's find a small perturbation that changes a model's classification result
  - Initial work formulates this problem like:

> • Minimize $\|r\|_2$ subject to:
>
> 1. $f(x+r) = l$
> 2. $x + r \in [0,1]^m$

  - Formally:

> • Minimize $c|r| + \text{loss}_f(x+r, l)$ subject to $x + r \in [0,1]^m$

[1] Szegedy et al., Intriguing Properties of Neural Networks, ICLR

How to solve this constrained optimization?

- Intuitions
  - Non-linearity, from activation functions like ReLU, is the root-cause
  - Downside:
    - Computationally demanding, if we find adversarial examples in non-linear models
    - It's also not theoretically proven that non-linearity is the primary issue
  - This work:
    - let's only consider linearity in non-linear models!
    - I will show the existence of adversarial examples exploiting the linearity

How to solve this constrained optimization?

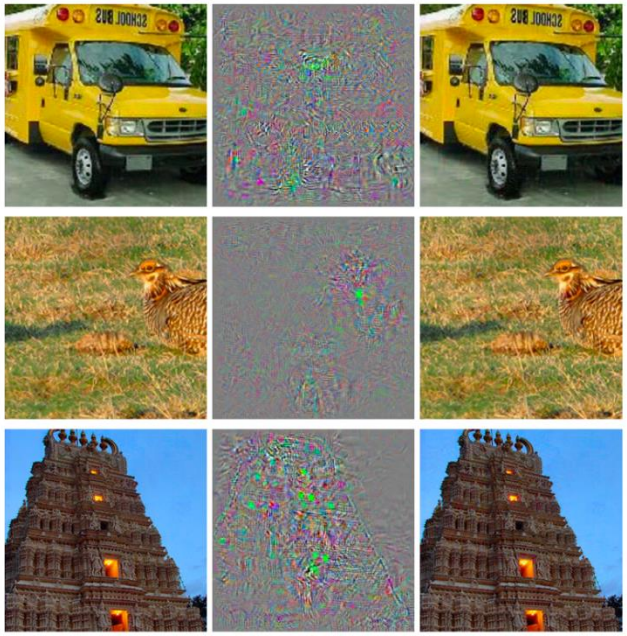
- Fast gradient sign method (FGSM)
  - A test-time input x and its true label y
  - A NN model f and its parameters θ
  - A loss (or a cost) function $J(\theta, x, y)$
  - Find an adversarial perturbation η such that $f(x+\eta) \neq y$ and $||\eta||_\infty < \varepsilon$

$$\boldsymbol{\eta} = \epsilon \text{sign}\left(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y)\right).$$

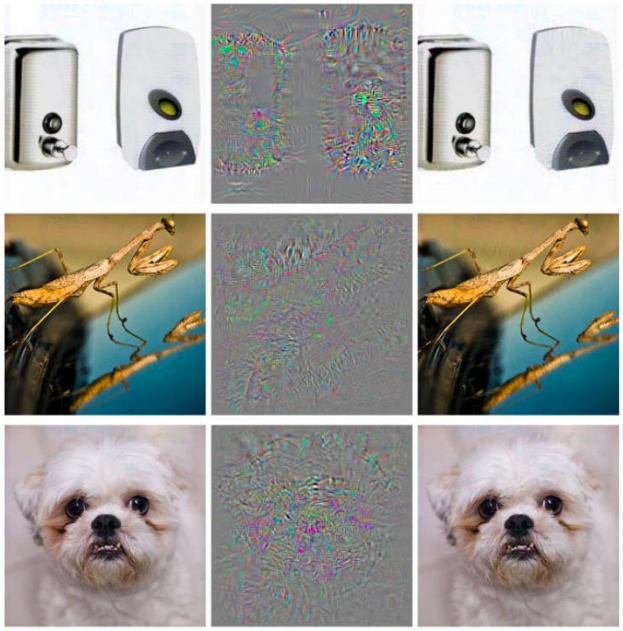
- Results on the test-sets
  - On MNIST: 99.9% error rate with an avg. confidence of 79.3% (eps = 0.25)
  - On CIFAR10: 87.2% error rate with an avg. confidence of 96.6% (eps = 0.1)

RESULTS

- Attacking AlexNet models trained on ImageNet



(a) Incorrect



(b) Incorrect

RESULTS

- Empirical findings:

	FC10(10^{-4})	FC10(10^{-2})	FC10(1)	FC100-100-10	FC200-200-10	AE400-10	Av. distortion
FC10(10^{-4})	100%	11.7%	22.7%	2%	3.9%	2.7%	0.062
FC10(10^{-2})	87.1%	100%	35.2%	35.9%	27.3%	9.8%	0.1
FC10(1)	71.9%	76.2%	100%	48.1%	47%	34.4%	0.14
FC100-100-10	28.9%	13.7%	21.1%	100%	6.6%	2%	0.058
FC200-200-10	38.2%	14%	23.8%	20.3%	100%	2.7%	0.065
AE400-10	23.4%	16%	24.8%	9.4%	6.6%	100%	0.086
Gaussian noise, stddev=0.1	5.0%	10.1%	18.3%	0%	0%	0.8%	0.1
Gaussian noise, stddev=0.3	15.6%	11.3%	22.7%	5%	4.3%	3.1%	0.3

  - Random perturbations are **NOT** the right way to measure the stability of neural networks
  - Adversarial examples **transfer**
    - Adversarial examples crafted on a model often work against others
    - AEs crafted on a model (trained with a disjoint training set) also works against the others

SRQ 4: WHAT PROPERTIES DO ADVERSARIAL EXAMPLES EXPLOIT?

- Observations from the work by Szegedy *et al.*
  - NNs are vulnerable to adv. examples
    - False sense of security
      - They are resilient to trivial, random (Gaussian) perturbations
      - However, it does **NOT** mean NNs are resilient to the worst-case perturbations
    - The vulnerability reduces when
      - We use regularization in training
      - We use linear models
    - Adv. examples **transfer**!

SRQ 3: How can we find adversarial examples, efficiently?

- Results from the prior work

Model Name	Description	Training error	Test error	Av. min. distortion
FC10(10^{-4})	Softmax with $\lambda = 10^{-4}$	6.7%	7.4%	0.062
FC10(10^{-2})	Softmax with $\lambda = 10^{-2}$	10%	9.4%	0.1
FC10(1)	Softmax with $\lambda = 1$	21.2%	20%	0.14
FC100-100-10	Sigmoid network $\lambda = 10^{-5}, 10^{-5}, 10^{-6}$	0%	1.64%	0.058
FC200-200-10	Sigmoid network $\lambda = 10^{-5}, 10^{-5}, 10^{-6}$	0%	1.54%	0.065
AE400-10	Autoencoder with Softmax $\lambda = 10^{-6}$	0.57%	1.9%	0.086

- Linear vs. non-linear models
  - Observations:
    - The min. distortion required to make a model's acc. to 0% is larger in the non-linear models (Row 4-6) than the linear models (Row 1-3)
    - **Non-linearity** may be the primary cause of adversarial examples

SRQ 3: FINDING ADVERSARIAL EXAMPLES ON NON-LINEAR MODELS

- Intuitions in the work by Goodfellow *et al.*[1]:
  - Finding adv. examples in non-linear models are computationally demanding
  - **(Hypothesis)** Let's only consider linearity in non-linear models!
  - **(Evaluation)** Show the existence of adversarial examples in linear models
    - Suppose an input x and its adv. input $x + \eta$, where $||\eta||_\infty < \varepsilon$, and a linear model

$$\boldsymbol{w}^\top \tilde{\boldsymbol{x}} = \boldsymbol{w}^\top \boldsymbol{x} + \boldsymbol{w}^\top \boldsymbol{\eta}.$$

  - **(Potential implications)**
    - Its linearity (and also the direction) matters
    - Introduce an easy way to find adversarial examples

SRQ 3: FAST GRADIENT SIGN METHOD (FGSM)

- Given
  - A test-time input (x, y)
  - A NN model f and its parameters θ
  - A loss (or a cost) function $J(\theta, x, y)$
- Find
  - An adversarial perturbation η such that $f(x + \eta) \neq y$ and $||\eta||_\infty < \varepsilon$

$$\boldsymbol{\eta} = \epsilon \text{sign}\left(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y)\right).$$

- Results on the test-sets
  - On MNIST: 99.9% error rate with an avg. confidence of 79.3% (eps = 0.25)
  - On CIFAR10: 87.2% error rate with an avg. confidence of 96.6% (eps = 0.1)

SRQ 3: BASIC ITERATIVE METHOD (BIM)

- Objectives
  - To craft **powerful** AEs

- BIM Method
  - Run FGSM over multiple iterations

$$\boldsymbol{X}_0^{adv} = \boldsymbol{X}, \quad \boldsymbol{X}_{N+1}^{adv} = Clip_{X,\epsilon}\left\{\boldsymbol{X}_N^{adv} + \alpha\,\mathrm{sign}\left(\nabla_X J(\boldsymbol{X}_N^{adv}, y_{true})\right)\right\}$$

- Iterative Least-Likely (ILL) Class Method
  - Choose a desired class as the class with the lowest logit value (y_{LL})

$$\boldsymbol{X}_0^{adv} = \boldsymbol{X}, \quad \boldsymbol{X}_{N+1}^{adv} = Clip_{X,\epsilon}\left\{\boldsymbol{X}_N^{adv} - \alpha\,\mathrm{sign}\left(\nabla_X J(\boldsymbol{X}_N^{adv}, y_{LL})\right)\right\}$$

TOPICS FOR TODAY

- Motivation
  - What is it?
  - Why do we care about adversarial examples?
- Research questions
  - How can we find adversarial examples?
  - How can a real-world attacker exploit them in practice?
  - How can we remove adversarial examples?

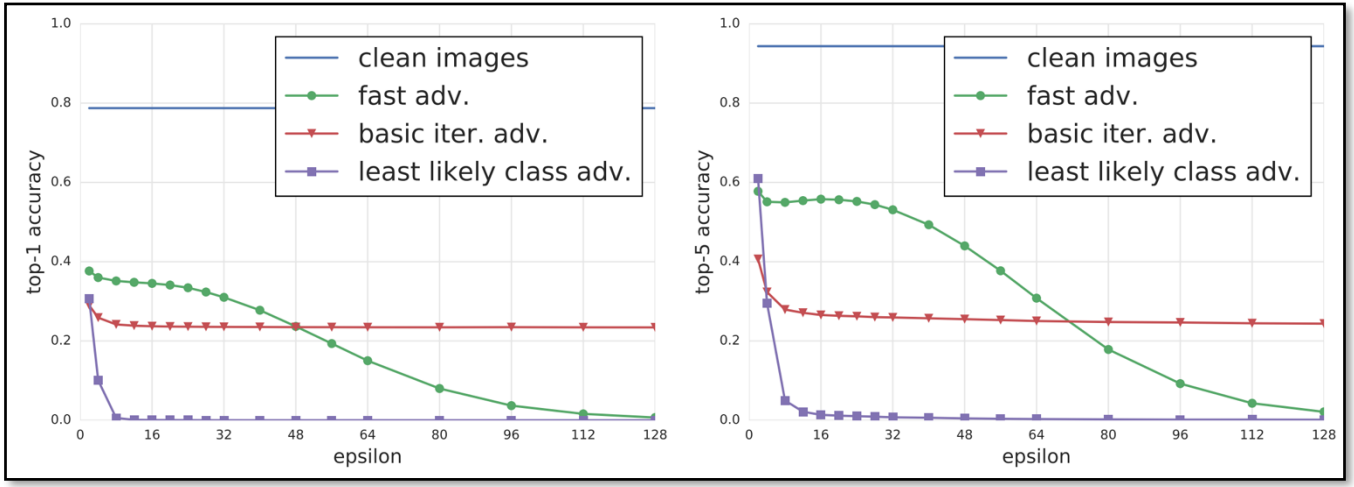
RQ 2: How does the attacker exploit AEs in practice?

- **C1:** AE in the numerical world ≠ AE in the physical world
  - Numerical perturbations by FGSM lead to the input values like 34.487
  - In the pixel space, such perturbations do not exist (*i.e.*, quantized pixel values)
  - One may take only classification results with a high probability (*e.g.*, > 0.8)
  - …

- Evaluation on CIFAR-10
  - Craft AEs on a DNN model (~an error rate of 99.9%)
  - Store these AEs into PNG files
  - Upload them to object recognition services (~an error rate of 10%)

RQ 2: How does the attacker exploit AEs in practice?

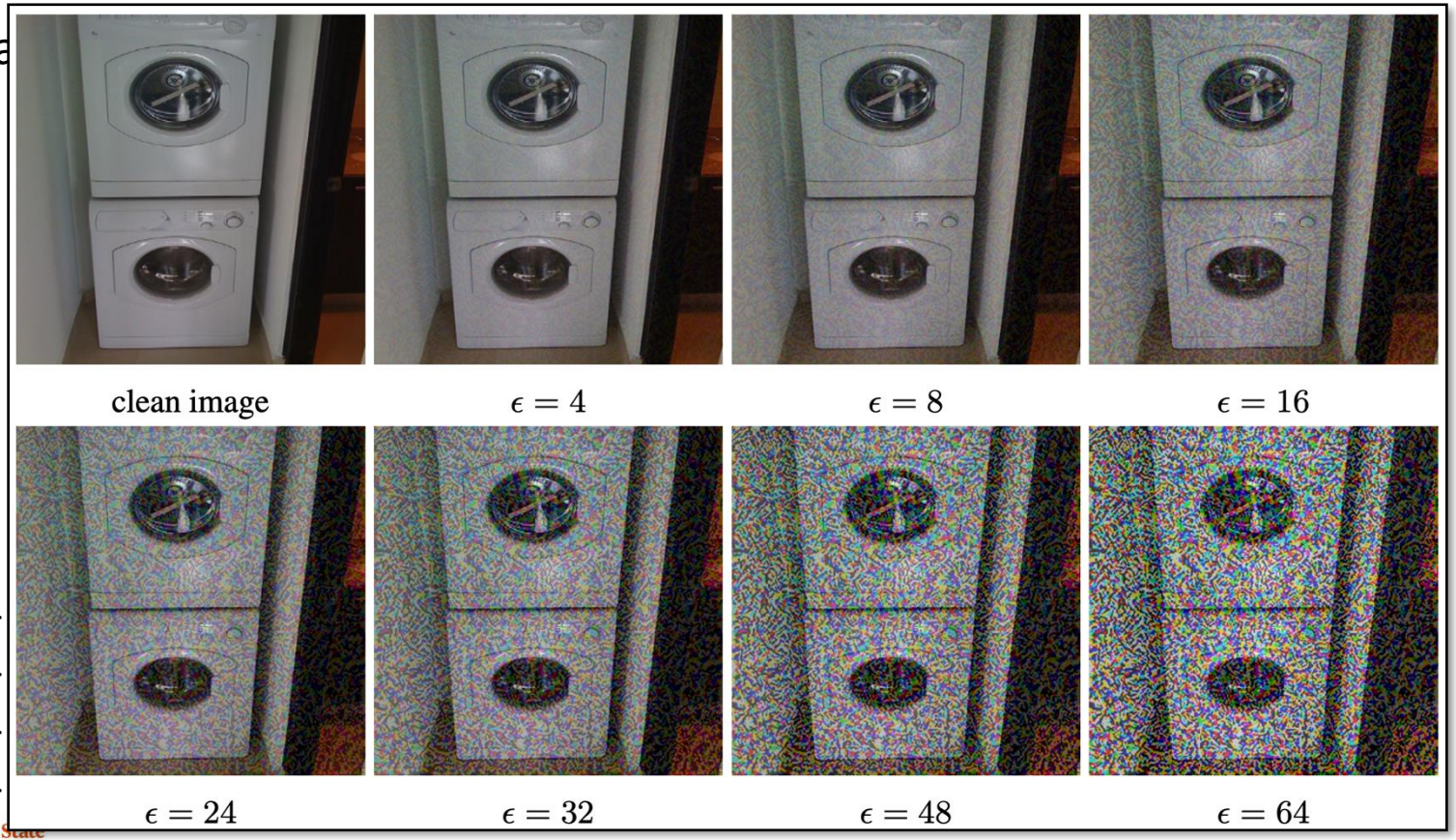
- Evaluation results of attacks on the ImageNet Inception-v3



- In FGSM, the error rate increases as we increase epsilon
- In the large eps, the error rate is ILL > FGSM > BIM
- In the smaller eps, the error rate is ILL > BIM > FGSM
- ILL achieves the highest error rate in both Top1 and Top5

RQ 2: How does the attacker exploit AEs in practice?

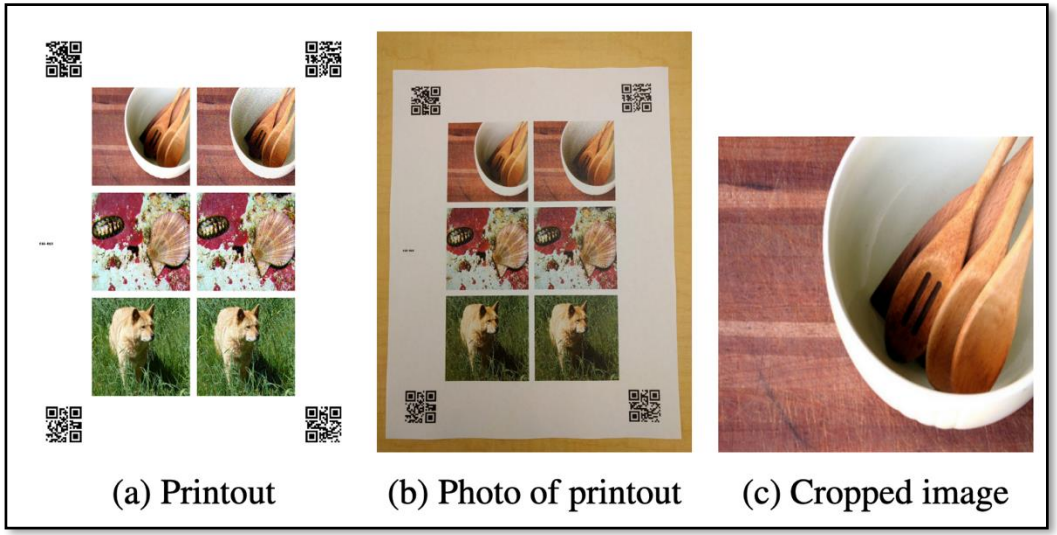
- Eva



RQ 2: How does the attacker exploit AEs in practice?

- Evaluation of attacks in realistic setup
  1. Craft AEs, store them in PNG, and print them
  2. Take photos of printed AEs with a cell phone
  3. Resize and center-crop the images from 2
  4. Run classification on the images from 3

- Measure
  - Classification accuracy
  - Destruction rate (error)



RQ 2: How does the attacker exploit AEs in practice?

- Observations
  - AEs work in the physical world
    - Misclassification rate is higher in AEs than what we observe with clean examples
    - Chances increase when we increase the perturbations (*i.e.*, eps from 2 to 16)
  - Prefiltering can reduce the misclassification significantly
    - **Prefilter:** only accept the classification with a high probability > 0.8
    - It reduces an error rate by 40 – 90%

RQ 2: STILL, I CAN'T BELIEVE IF IT WORKS

- Link, Link, Link

How can we find adversarial examples?

- Sub research questions
  - How can we define the adversarial examples?
  - What are the methods we can develop for finding adversarial examples?
  - What are the computational properties adversarial examples exploit?

Thank You!

Instructor: Sanghyun Hong

https://secure-ai.systems/courses/MLSec/current