Physical-World Attacks on Machine Learning

Mahmood Sharif
PhD Candidate, ECE, CMU

Security and Fairness of Deep Learning
Spring 2019

Carnegie Mellon University

Today’s Topics

1. Adversarial Machine Learning

2. Misleading Face Recognition Systems
    - *"Accessorize to a Crime: Real and Stealthy Attacks on State-of-the-Art Face Recognition," Sharif et al., CCS ’16*
    - "A General Framework for Adversarial Examples with Objectives," Sharif et al., TOPS ’19 (to appear)

3. Misleading Speech Recognition
    - "Hidden Voice Commands," Carlini et al., USENIX Security ’16
    - "DolphinAttack: Inaudible voice commands," Zhang et al., CCS, ’17

Predecessor: "Cocaine Noodles," WOOT ’15

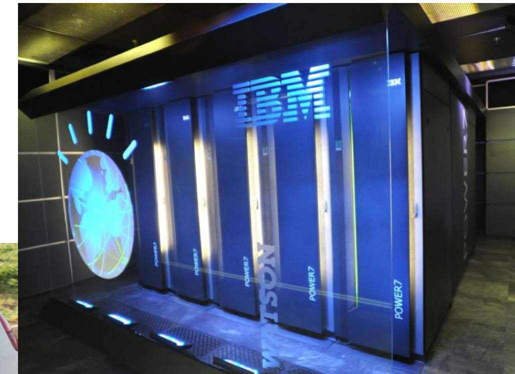
Today’s Topics

1. Adversarial Machine Learning

2. Misleading Face Recognition Systems
   - "Accessorize to a Crime: Real and Stealthy Attacks on State-of-the-Art Face Recognition," Sharif et al., CCS ’16
   - "A General Framework for Adversarial Examples with Objectives," Sharif et al., TOPS ’19 (to appear)

3. Misleading Speech Recognition
   - "Hidden Voice Commands," Carlini et al., USENIX Security ’16
   - "DolphinAttack: Inaudible voice commands," Zhang et al., CCS, ’17

Machine Learning Is Ubiquitous



- Cancer diagnosis

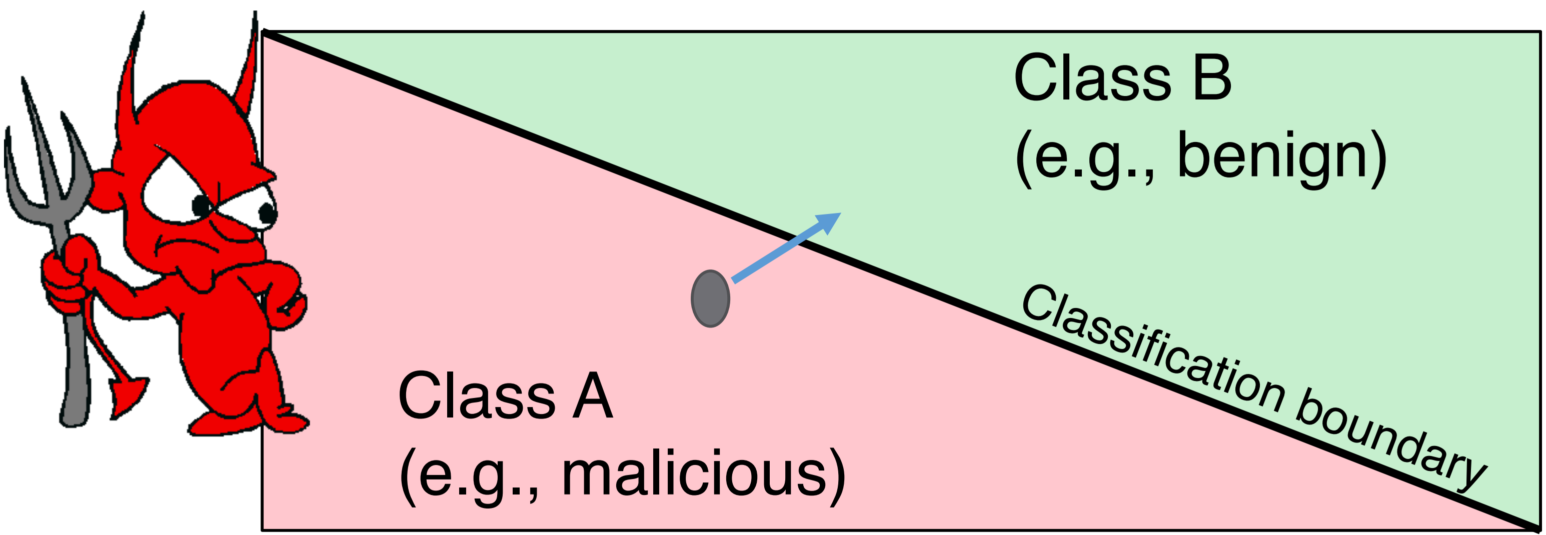
- Self-driving cars

- Surveillance and access-control
- Anomaly-based NIDS

...

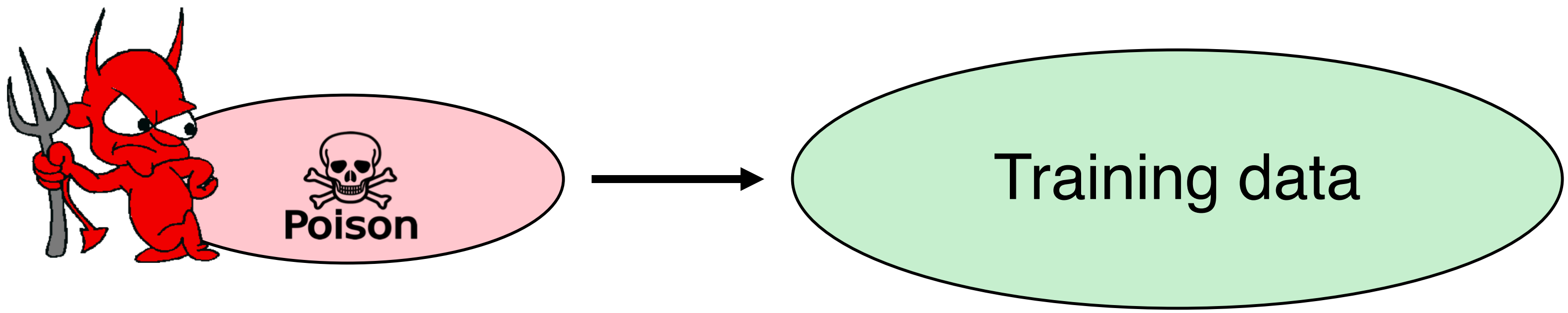
Misleading Machine Learning: Evasion

- Change input slightly, such that it remains in A, but is classified in B. Examples:
  - Malicious packet classified as benign
  - Person A confused as person B

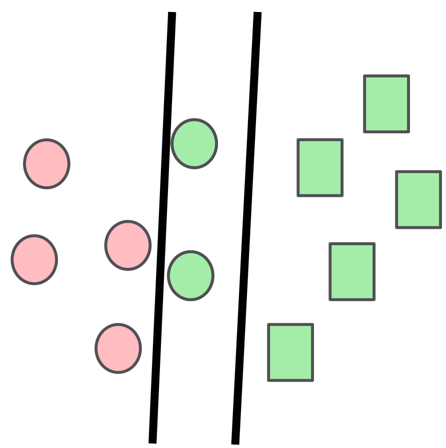


Misleading Machine Learning: Poisoning

- Cause classifier to learn wrong concepts by poisoning training data



- Result:



Today’s Topics

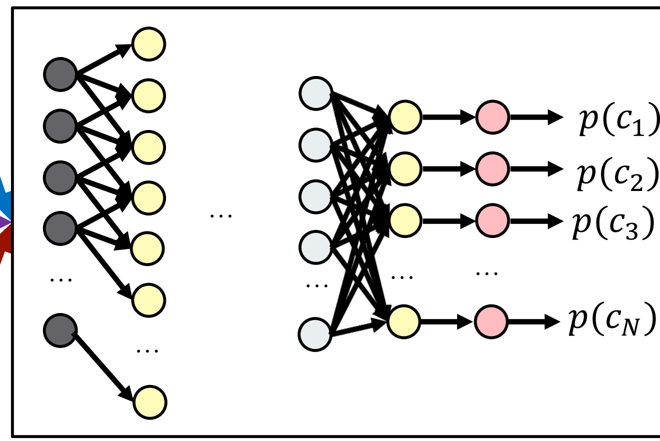
1. Adversarial Machine Learning

2. Misleading Face Recognition Systems
    - "Accessorize to a Crime: Real and Stealthy Attacks on State-of-the-Art Face Recognition," Sharif et al., CCS ’16
    - "A General Framework for Adversarial Examples with Objectives," Sharif et al., TOPS ’19 (to appear)

3. Misleading Speech Recognition
    - "Hidden Voice Commands," Carlini et al., USENIX Security ’16
    - "DolphinAttack: Inaudible voice commands," Zhang et al., CCS, ’17

What Do You See?

Deep Neural Network (DNN)



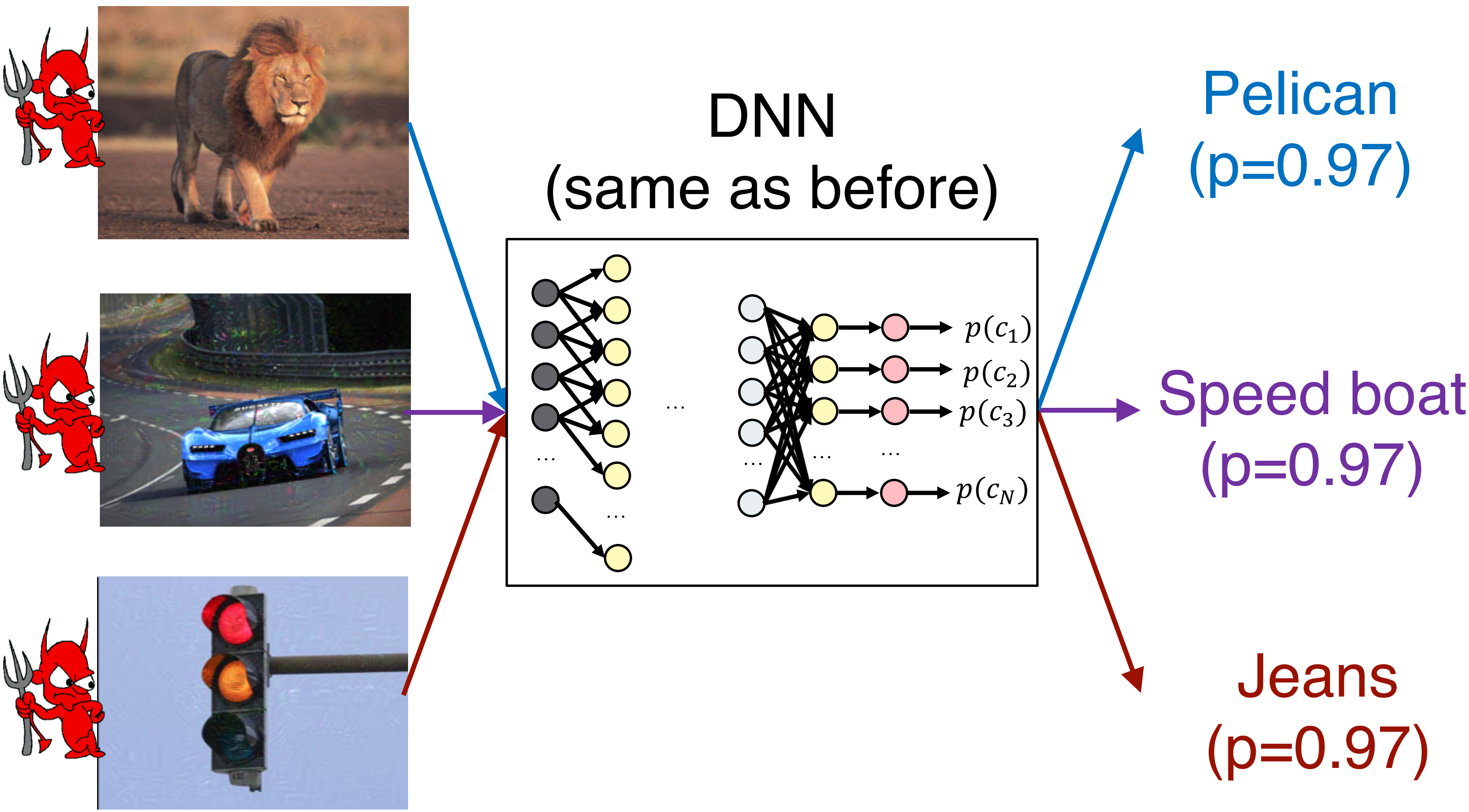
Lion (p=0.99)

Race car (p=0.74)

Traffic light (p=0.99)

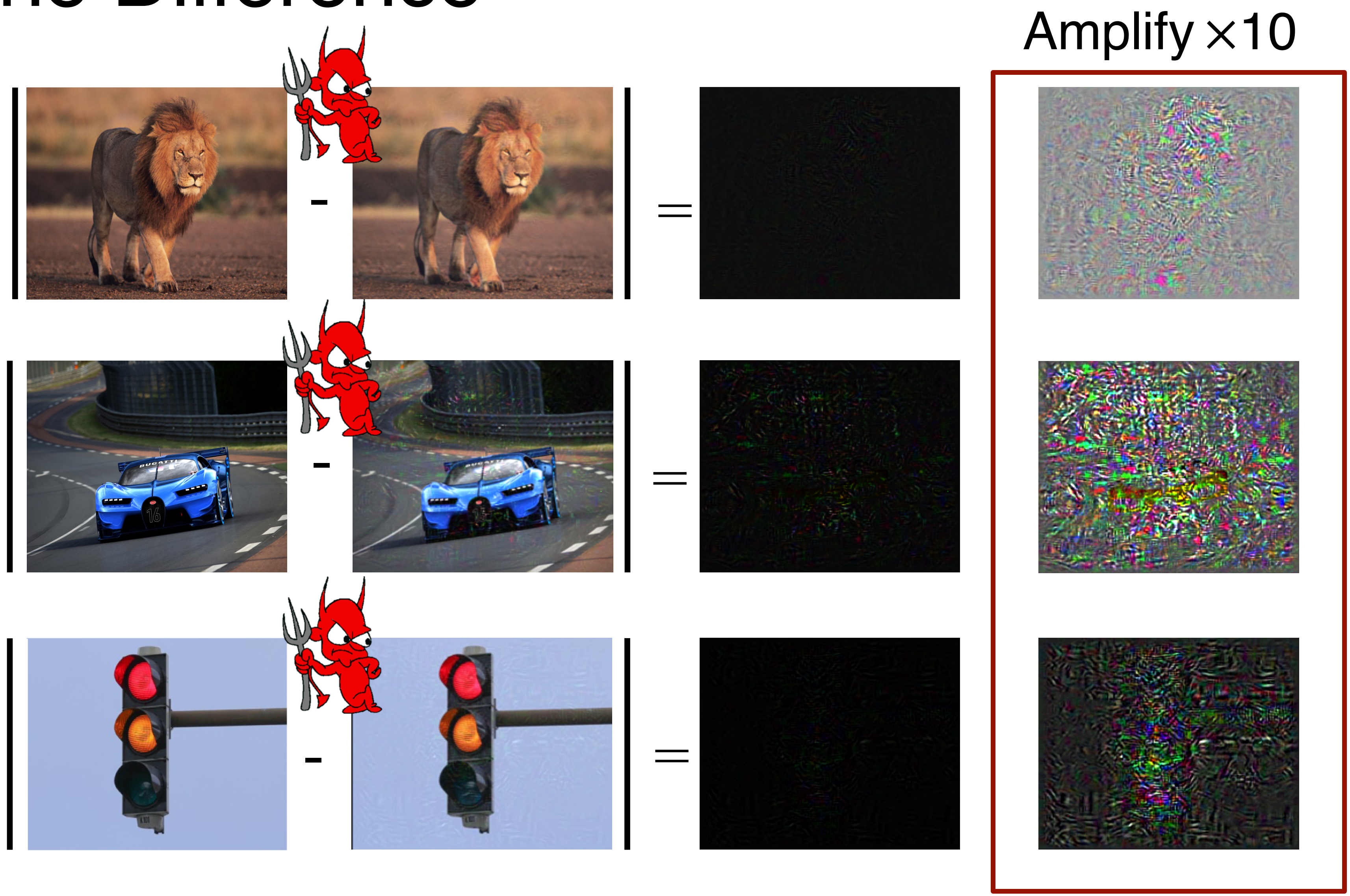
[Chatfield et al., BMVC ’14]

What Do You See Now?



[Szegedy et al., ICLR ’14]

The Difference



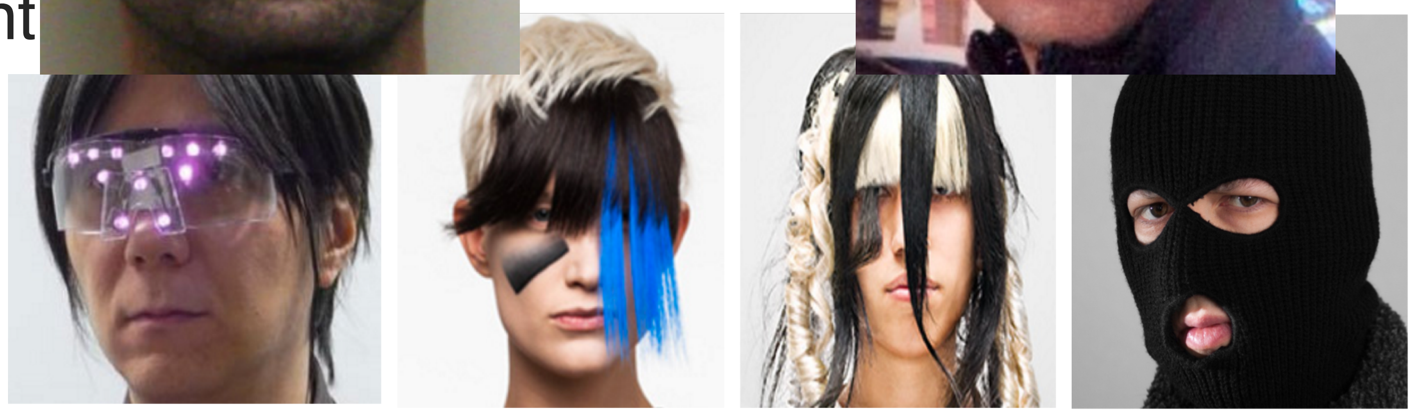
This Work

Physical realizability:

- Attacker can only change own appearance
- Robu… imaging co…

Incons…

- Do no… h) suspicion
- Want…

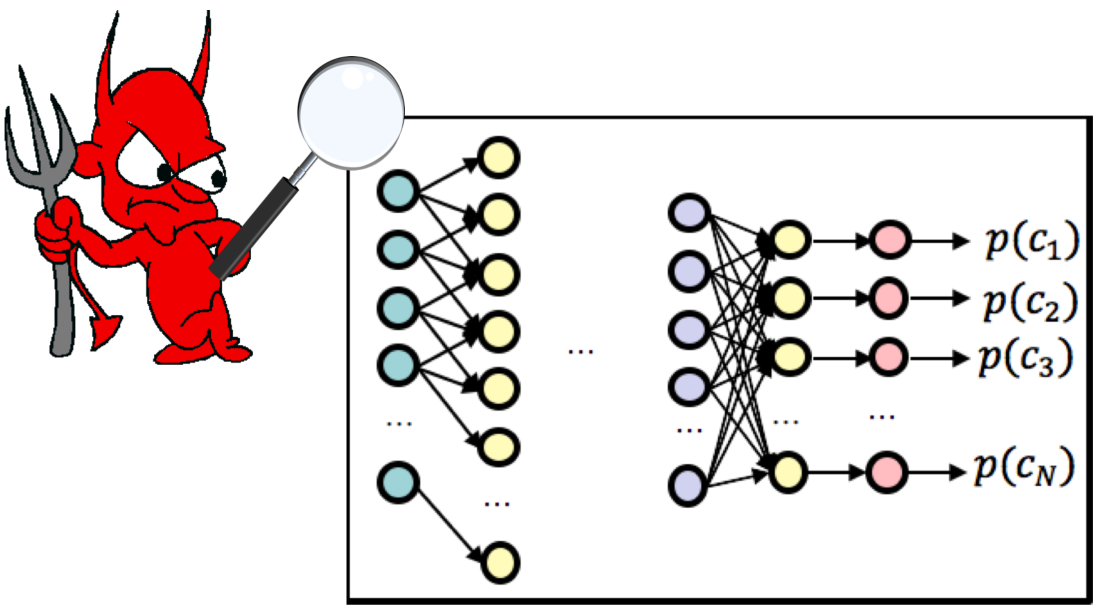


What Are the Adversary's Capabilities?

To generate attacks, attacker needs to know how changing input affects output

$\Downarrow$

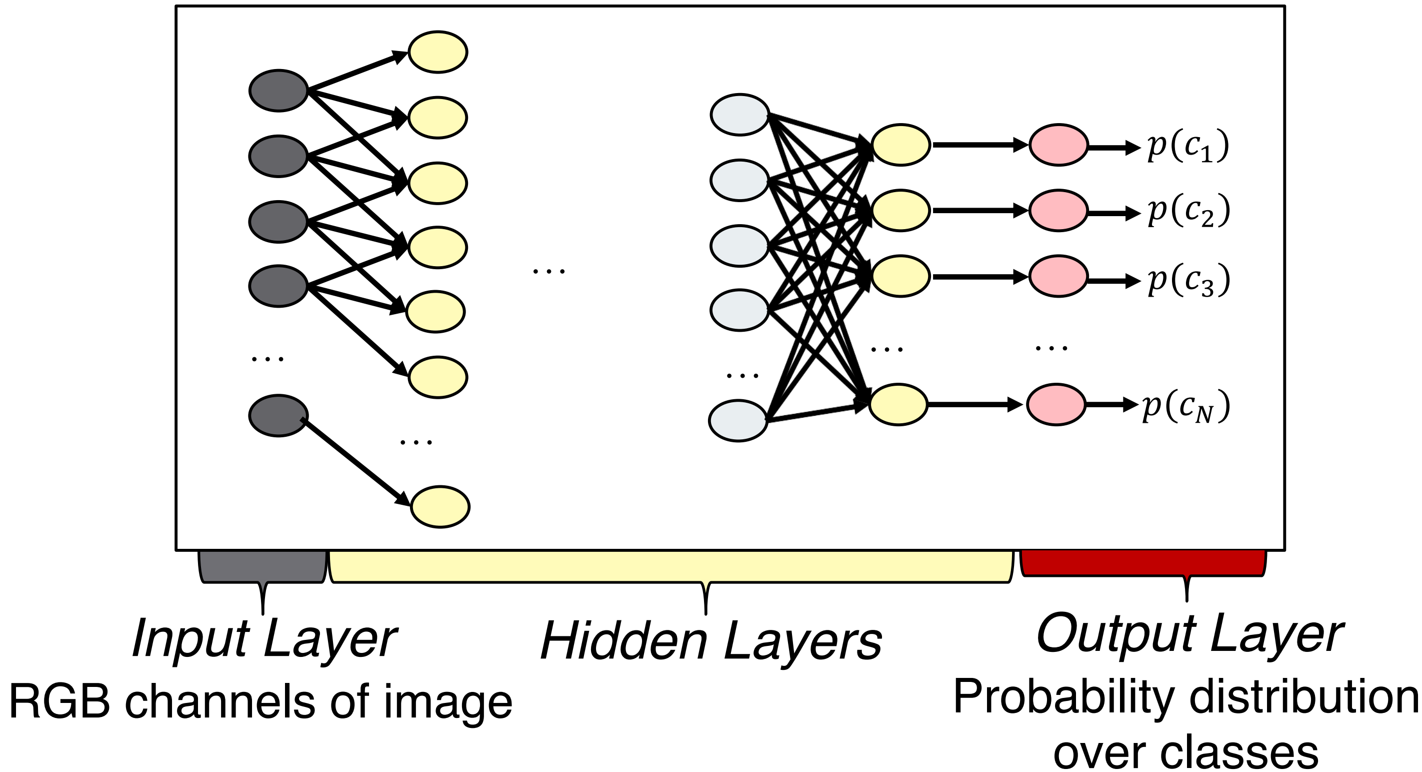
White-box setting



Background: Misleading DNNs
(and other classifiers)

What's a (Deep) Neural Network?

- Idea: simulate how brain cells work
- Basic building block: neuron, a simple computational unit



Classification DNNs are functions from inputs to classes (or probability distribution over classes)

How to Mislead DNNs?

Given DNN and input, find minimal change that causes specific misclassification

$\Downarrow$

Imperceptible adversarial examples
[Szegedy et al., ICLR '14]

- Defined as an optimization problem:

$$\underset{r}{\operatorname{argmin}} \; \boxed{\underbrace{|f(x+\boldsymbol{r}) - c_t|}_{\text{misclassification}}} + \kappa \cdot \underbrace{|\boldsymbol{r}|}_{\text{imperceptibility}}$$

Refer to as:
$\operatorname{distance}(f(x+r), c_t)$

x: input image
$f(\cdot)$: classification function (e.g., DNN)
$|\cdot|$: norm function (e.g., Euclidean norm)
c_t: target class
r: perturbation
κ: tuning parameter

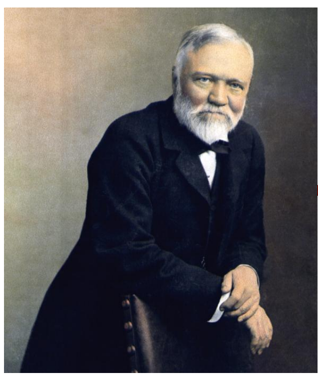
Optimization can be solved via gradient descent, L-BFGS, …

Fooling Face Recognition
(*Impersonation & Dodging*)

Facial Biometric Systems

Detection and recognition are usually pipelined:

1. Detect the face
2. Recognize the person



Andrew Carnegie

→ Face Detection →

→ Face Recognition →

$$\begin{array}{cl} \begin{pmatrix} 0.01 \\ 0.97 \\ 0.02 \end{pmatrix} & \begin{array}{l} \text{Brad Pitt} \\ \text{Carnegie} \\ \text{Harrison Ford} \end{array} \end{array}$$

Attacks may target detection or recognition

Face Recognition: Our Attacks

Impersonation

I want to break into the Blade Runner filming location


Andrew Carnegie

Face Recognition

$\begin{pmatrix} 0.01 \\ 0.02 \\ {\color{red}0.97} \end{pmatrix}$ Brad Pitt, Carnegie, Harrison Ford

- Targeting a specific subject
- To access specific resources or cause blame to be laid on a target

Dodging

I don't want to be recognized at Justin Bieber's concert


Andrew Carnegie

Face Recognition

$\begin{pmatrix} {\color{red}0.94} \\ 0.01 \\ 0.05 \end{pmatrix}$ Brad Pitt, Carnegie, Harrison Ford

- Being recognized incorrectly
- To achieve privacy, or if target doesn't matter

Deep Face Recognition

We use and build on DNN proposed by Parkhi et al. [BMVC ‘15]:

- Trained to recognize 2622 celebrities
- Evaluated on Labeled Faces in the Wild [Huang et al., ’07]:
  - 13233 face images collected in the wild (uncontrolled conditions)
- Outperforms humans:

Accuracy of humans	Accuracy of Parkhi et al.’s DNN
97.53%	**98.95%**

Strawman Formalization

- Like Szegedy et al., achieve impersonation by:

$$\underset{r}{\operatorname{argmin}} \ \overbrace{\boxed{\text{distance}(f(x+r), c_t)}}^{\text{misclassification}} + \overbrace{\boxed{\kappa \cdot |r|}}^{\text{imperceptibility}}$$

- Example of impersonation:

Vicky McClure

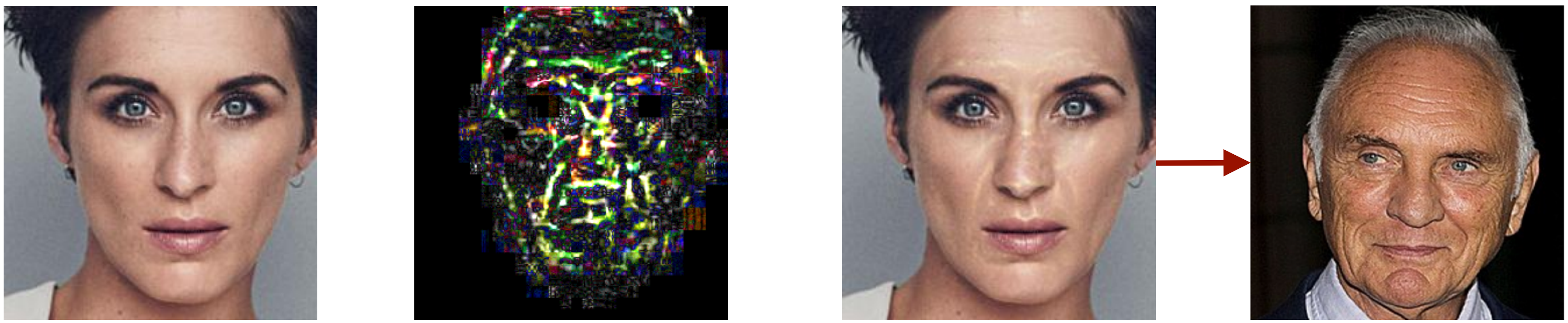
$10 \times abs(\text{perturbation})$

Terence Stamp

Caveat: may be hard to control background

Phase #1: Apply Changes to Face Only

- Image segmentation to find the face
- Only change pixels that overlay the face



Vicky McClure $20\times abs(\text{perturbation})$ Terence Stamp

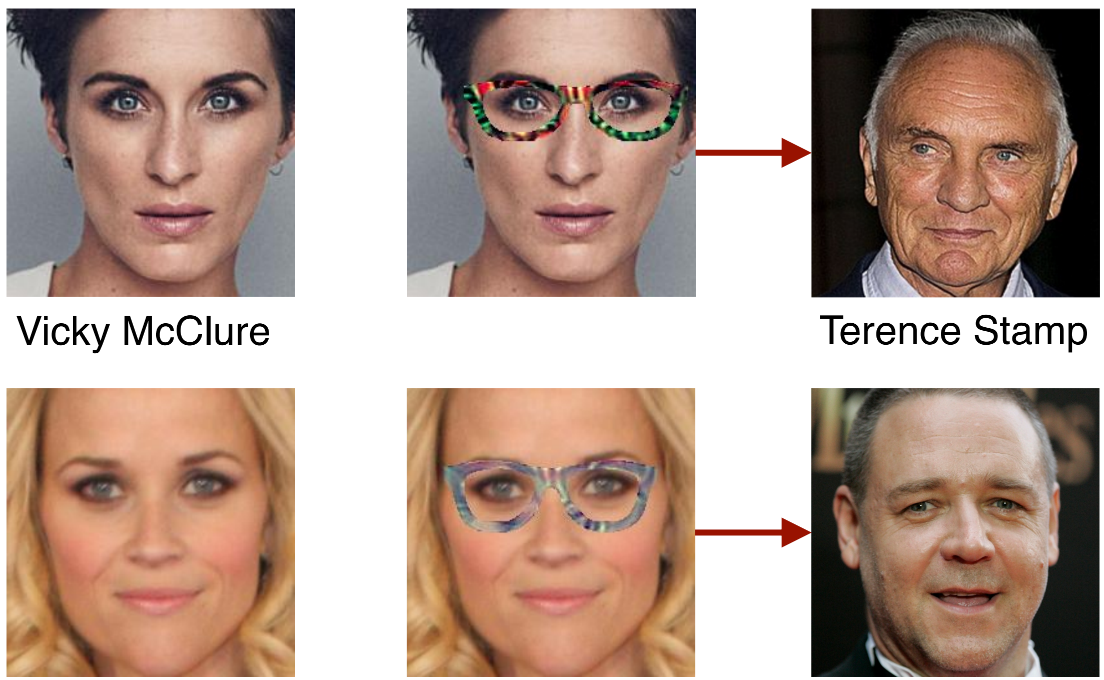
- Every impersonation attempt works

Caveats:

1. May be hard to realize the perturbations
2. Perturbations are smaller than camera's sampling error

Phase #2: Apply Changes to Eyeglasses

1. Easier to realize (2D or 3D printing)
2. Wearing eyeglasses isn't associated with adversarial intent



Vicky McClure → Terence Stamp

Reese Witherspoon → Russell Crowe

Experiments in Digital Environment

- 20 random pairs of attackers + targets
- 92% of impersonation attempts succeeded

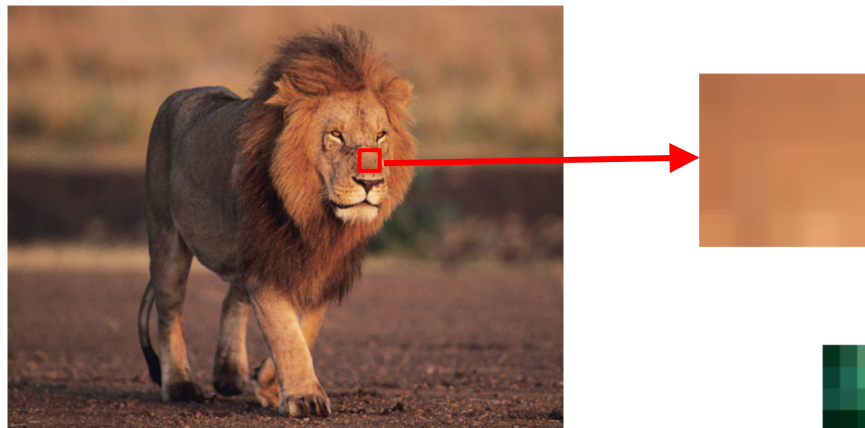
Reese Witherspoon

Russell Crowe

Can We Make Attacks Physically Realizable?

Phase #3: Smooth Transitions

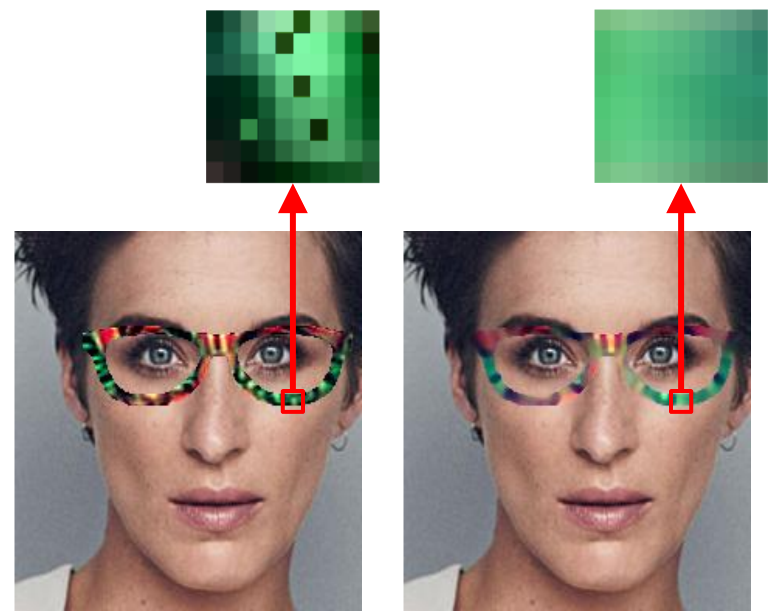
- Natural images tend to be smooth:



- We achieve this by minimizing total variations:

$$\mathrm{TV}(r) = \boxed{\sum_{i,j}\sqrt{\left(r_{i,j+1} - r_{i,j}\right)^2 + \left(r_{i+1,j} - r_{i,j}\right)^2}}$$

Sum of differences of neighboring pixels



Without min *TV()*   With min *TV()*

Phase #4: Printable Eyeglasses

- Challenge: Cannot print all colors
- Find printable colors by printing color palette

Ideal
color palette

Printed
color palette

- Define non-printability score *(NPS)*:
  - *NPS* is high if colors are not printable, and low otherwise
- Generate printable eyeglasses by minimizing *NPS*

Phase #5: Robust Perturbations

- Two samples of the same face are almost never the same ⇒ attack should generalize *beyond one* image
- Achieved by finding one attack accessory that leads any image in a set of images to be misclassified:

$$\underset{r}{\operatorname{argmin}} \left(\sum_{x \in X} \operatorname{distance}(f(x+r), c_t) \right)$$

X is a set of images, e.g., $X = \left\{ \right.$  $\left. \right\}$

Putting All the Pieces Together

- Physically realizable impersonation:

$$\underset{r}{\operatorname{argmin}} \boxed{\left(\sum_{x \in X} \operatorname{distance}(f(x+r), c_t)\right)} + \boxed{\kappa_1 \cdot \operatorname{TV}(r)} + \boxed{\kappa_2 \cdot \operatorname{NPS}(r)}$$

misclassify as c_t
(set of images)

smoothness

printability

Does This Work?

To test our approach, we need:

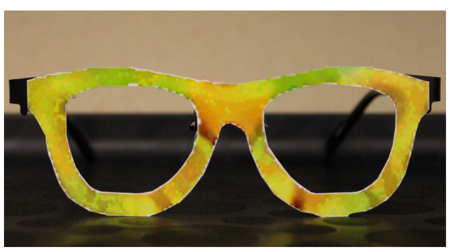
1. People to play role of the attacker

Lujo

Sruti

Mahmood

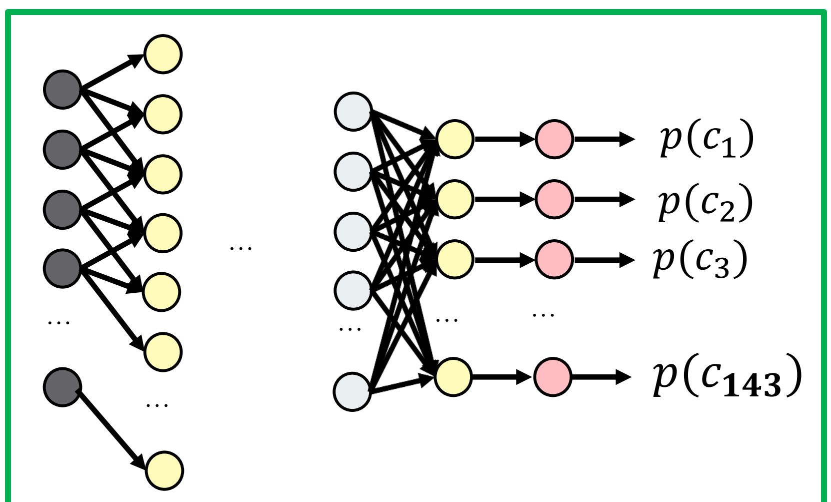
2. Realize the eyeglasses



3. DNN that recognizes the attackers

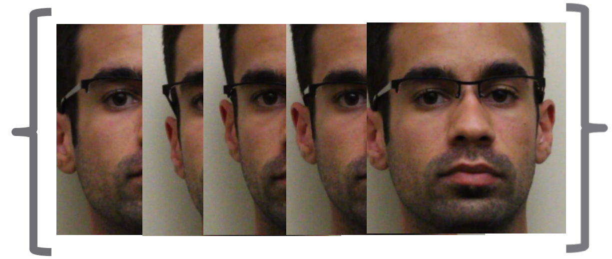
A DNN That Recognizes Us

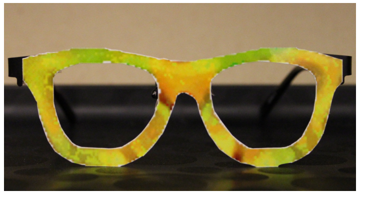
- Hard to train DNN from scratch ⇒ Use standard technique (transfer learning) to retrain DNN from Parkhi et al.'s
- New DNN recognizes **143** subjects:
  - First 3 authors + 140 Celebrities from PubFig dataset
- Accuracy: 96.75%

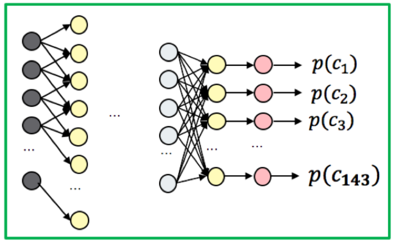


Experiment: Realized Impersonations

- Procedure:
    1. Collect images of attacker
    2. Choose random target
    3. Generate and print eyeglasses
    4. Collect 30 to 50 images of attacker wearing eyeglasses
    5. Classify collected images







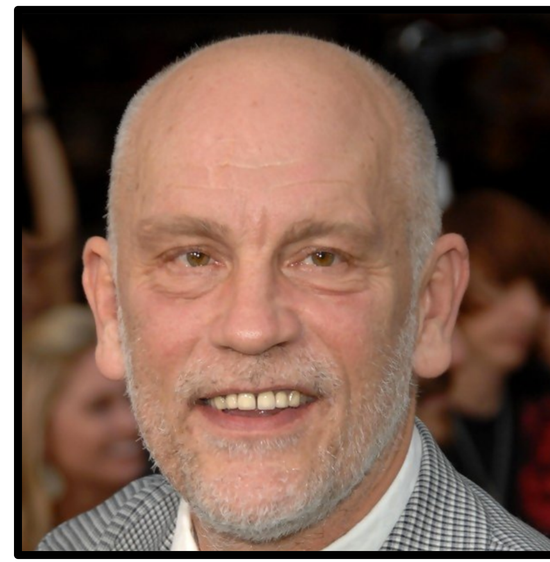
- Success metric: fraction of collected images misclassified as target

- Limitation: small set of variations in lighting

Impersonation Attacks Pose Real Risk!

Lujo

John Malkovich



100% success

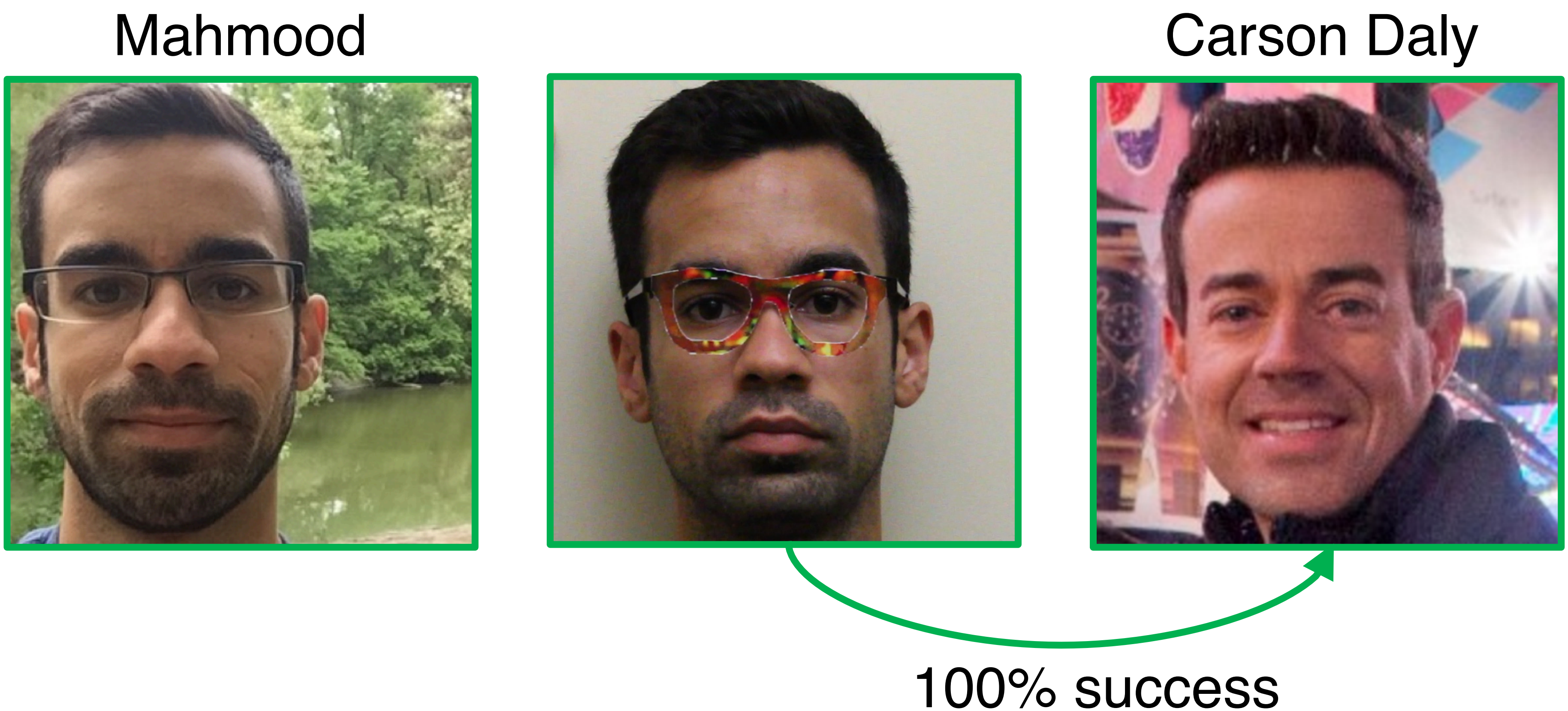
Impersonation Attacks Pose Real Risk!

Sruti

Colin Powell

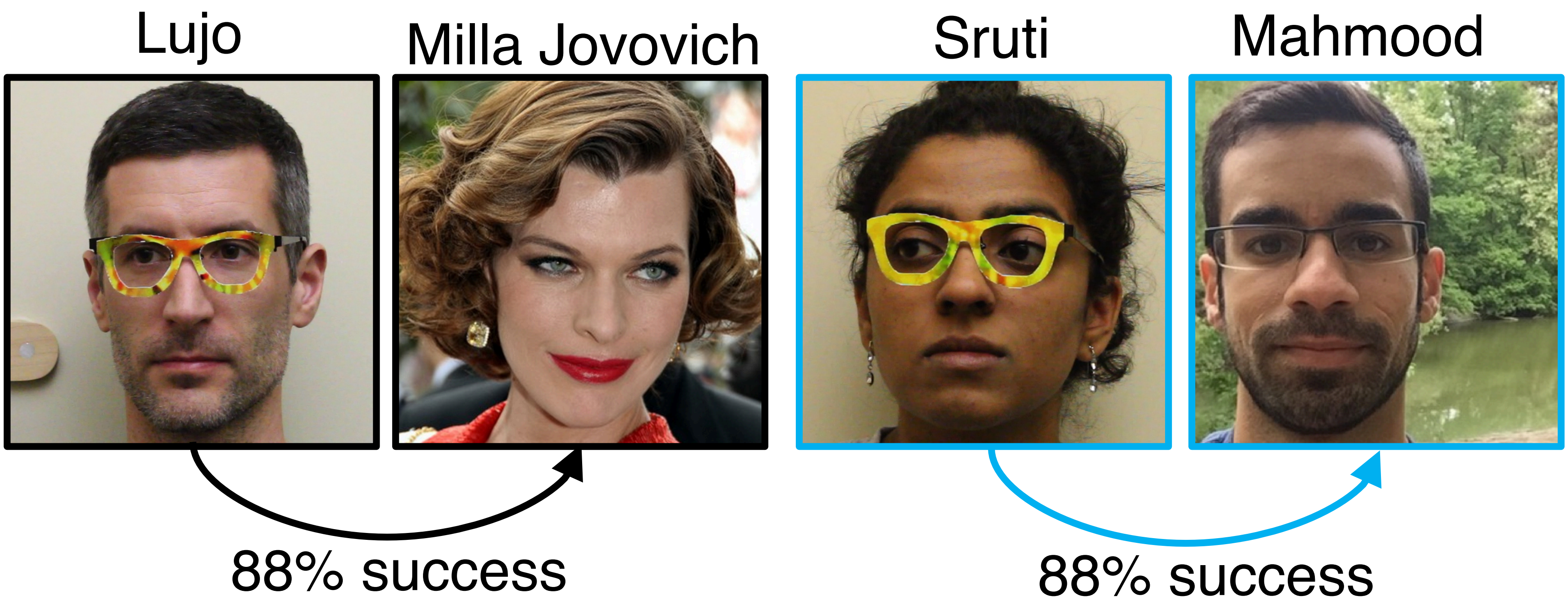
16% success

Impersonation Attacks Pose Real Risk!



More Realized Impersonations

- Against another DNN trained to recognize 10 subjects (including first 3 authors)



Question: How to Formalize Dodging?

- For reference, impersonation is formalized as:

$$\underset{r}{\operatorname{argmin}} \left(\sum_{x \in X} \operatorname{distance}(f(x+r), c_t) \right) + \kappa_1 \cdot \operatorname{TV}(r) + \kappa_2 \cdot \operatorname{NPS}(r)$$

misclassify as c_t (set of images)

smoothness

printability

- Dodging:

$$\underset{r}{\operatorname{argmin}} \left(\sum_{x \in X} -\operatorname{distance}(f(x+r), c_x) \right) + \kappa_1 \cdot \operatorname{TV}(r) + \kappa_2 \cdot \operatorname{NPS}(r)$$

?

misclassify as $\sim c_x$ (set of images)

smoothness

printability

Dodging Examples

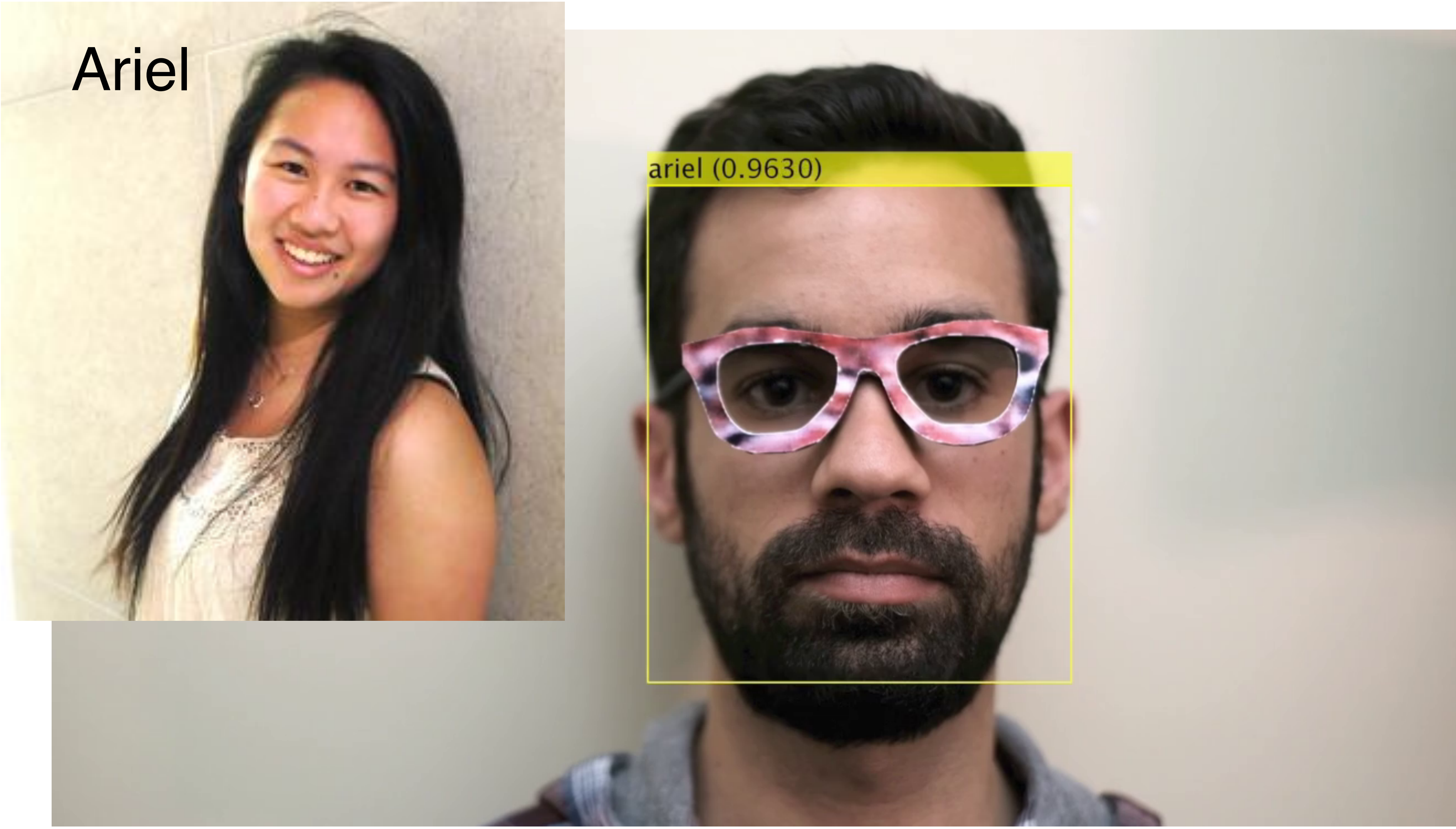
Not Sruti

Not Lujo

Not Mahmood

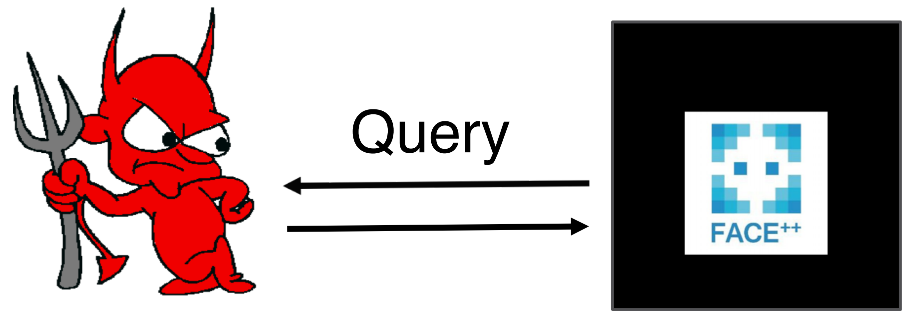
Probability assigned to correct classes is low (<0.03 in all cases)

Demo

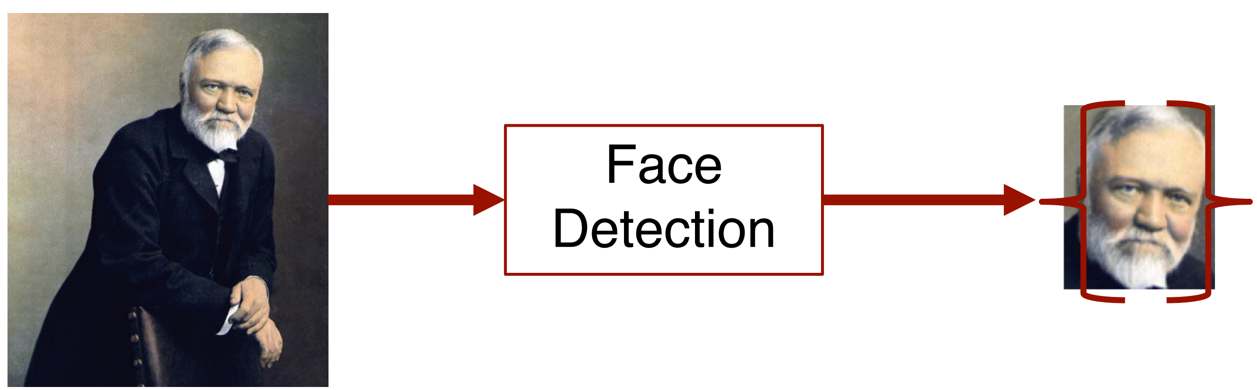


Extensions (vs. Online Re-identification)

- Impersonations against commercial face-recognition (Face++)
  - Threat model: black-box



- Invisibility against Viola-Jones:



(Possible) Defenses

- Ask subjects to remove accessories before recognition
  - Caveats: requires expensive enforcement (e.g., human operator), enforcement isn't always possible (e.g., surveillance or mobile phones)
- Train a model with provable accuracy guarantees
  - Works mainly for "imperceptible" perturbations ☹
- Show recognition system samples of attacks at training
  - Attacks can still be found at deployment time ☹
- Use machine-learning classifier to detect attacks
  - Detector and recognition system can be simultaneously fooled ☹

Today’s Topics

1. Adversarial Machine Learning

2. Misleading Face Recognition Systems
   - "Accessorize to a Crime: Real and Stealthy Attacks on State-of-the-Art Face Recognition," Sharif et al., CCS ’16
   - "A General Framework for Adversarial Examples with Objectives," Sharif et al., TOPS ’19 (to appear)

3. Misleading Speech Recognition
   - "Hidden Voice Commands," Carlini et al., USENIX Security ’16
   - "DolphinAttack: Inaudible voice commands," Zhang et al., CCS, ’17

Hidden Voice Commands

Sounds that are hard/impossible for humans to understand, but interpreted as voice commands by speech recognition

Risks?

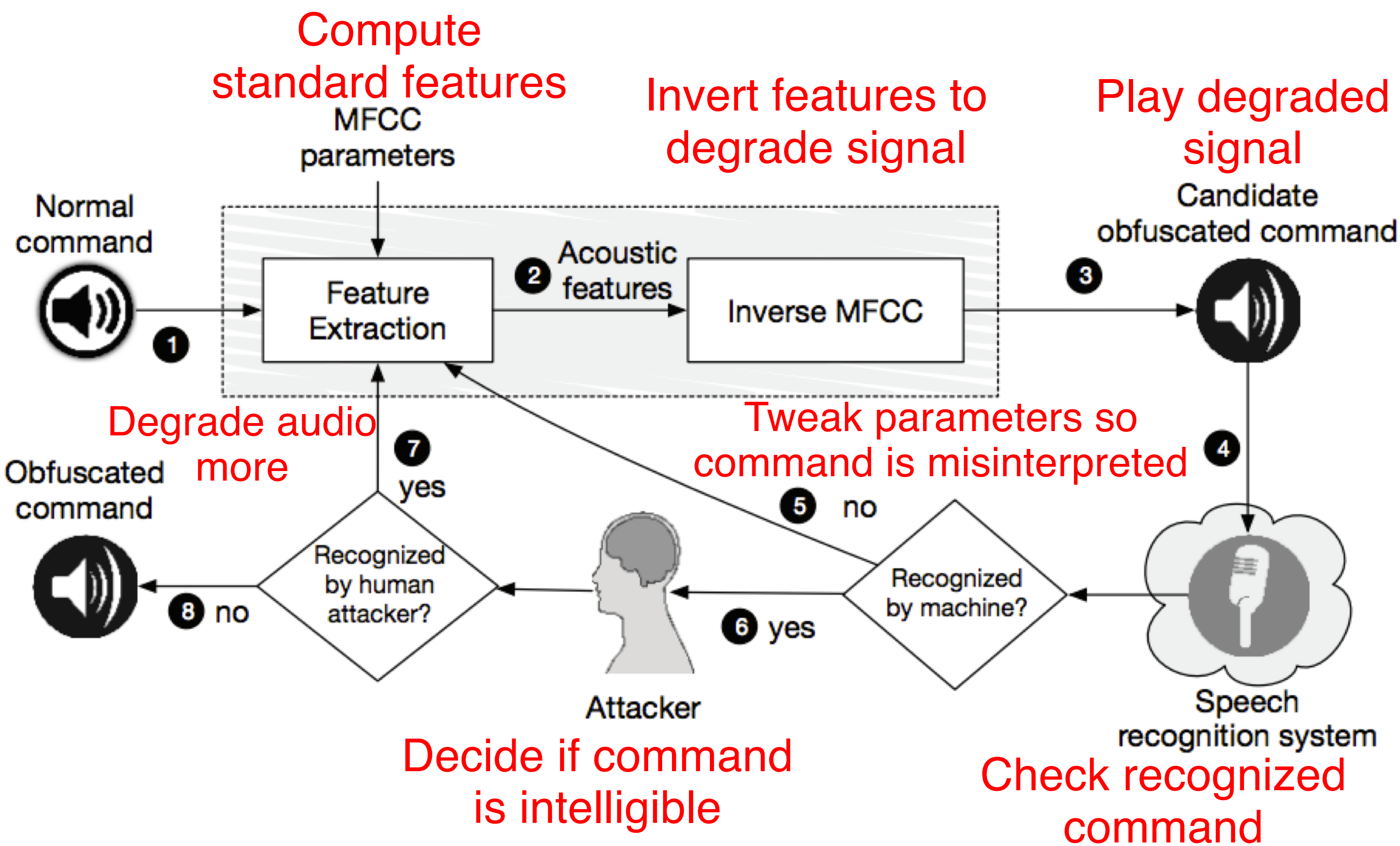
1. Compromise privacy (e.g., "call …", "upload contacts ...")
2. Compromise security (e.g., "open malicious.com", ...)
3. Monetary loss (e.g., send premium text message)

What is Being Said?

What is Being Said? (#2)

How Does this Work?

Black-box attack:



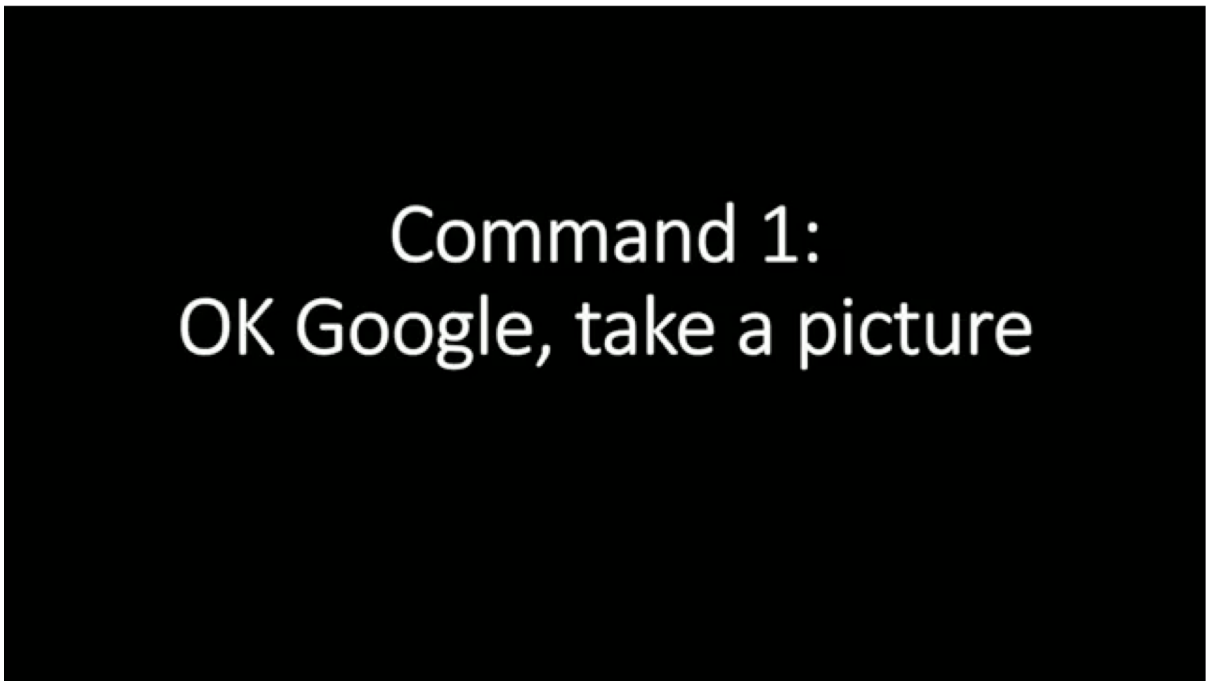
White-box Attack

Attacker that knows system's internals has more power

What is being said?

Recently: Inaudible Voice Commands

Idea: sounds outside of hearing range (20Hz-20KHz) interpreted as commands (by Google Now, Alexa, …)



[1] Zhang et al. "DolphinAttack: Inaudible voice commands," CCS, ’17.
[2] Song and Mittal. "Inaudible Voice Commands." arXiv, ’17.

(Possible) Defenses

- Perform speaker recognition: only authorized people can issue commands

- Machine-learning classifier that detects attacks
  - Caveat: Can attackers fool both the recognition system and detector?

- Filters:
  - Hidden commands: Sampling input uniformly harms attacks, but does not affect benign commands
  - Inaudible commands: Low pass filters allow only frequencies $\leq$ 20KHz

Takeaways

- Machine-learning algorithms *are not* foolproof; practical and stealthy attacks (affecting privacy, security, …) are possible
- Attacks on machine-learning have different forms. Examples:
  - Physical or digital domain
  - White-box or black-box settings
- These vulnerabilities should be taken into account when designing systems