The Cost of Learning from the Best:

How Prior Knowledge Weakens the Security of Deep Neural Networks

Baidu X-Lab

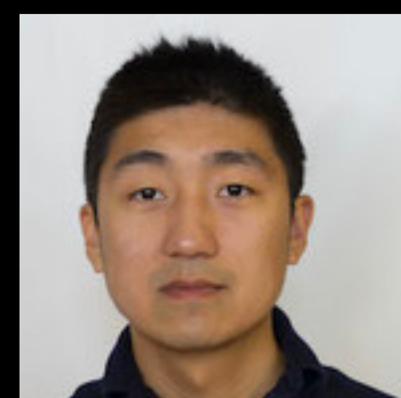
black hat ASIA 2019

Our Team X-Lab

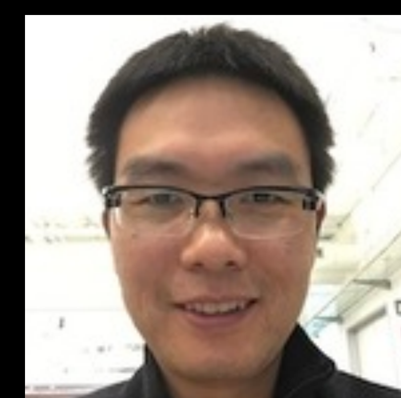
AI Security Research @ Baidu X-Lab

Tao Wei

Yulong Zhang



Yunhan Jia



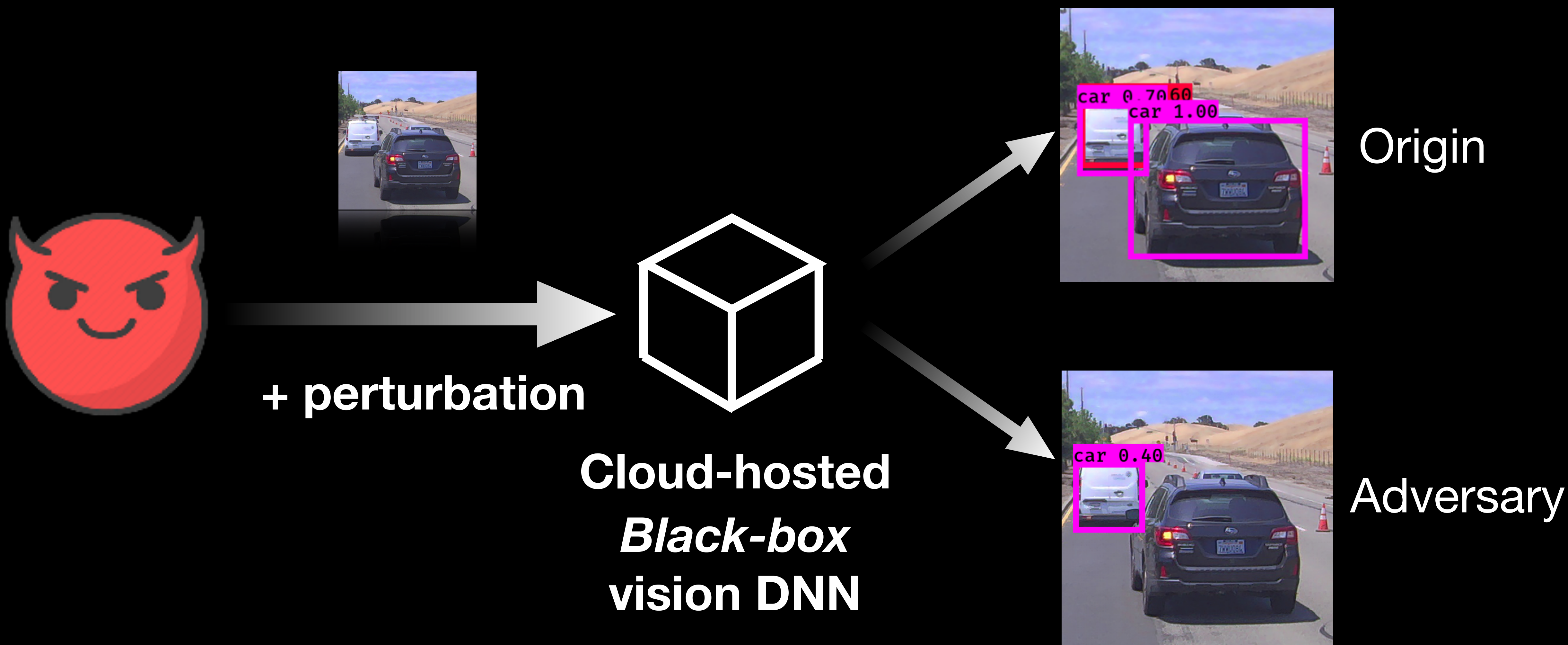
Zhenyu Zhong

Yantao Lu

Open Source Projects:

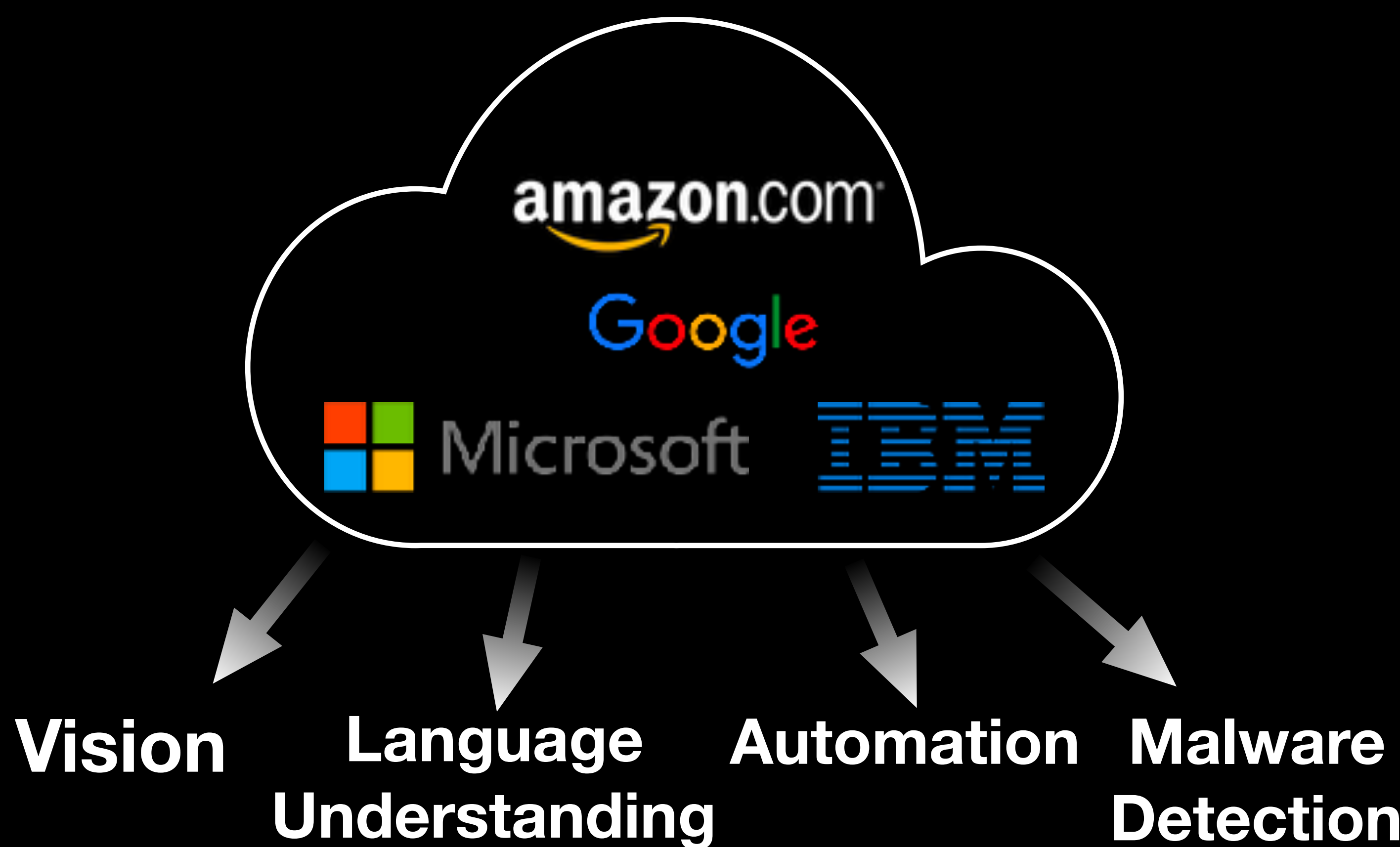
MesaLock MesaLink MesaPy MesaTEE

Today's Topics



AI Models in the Cloud

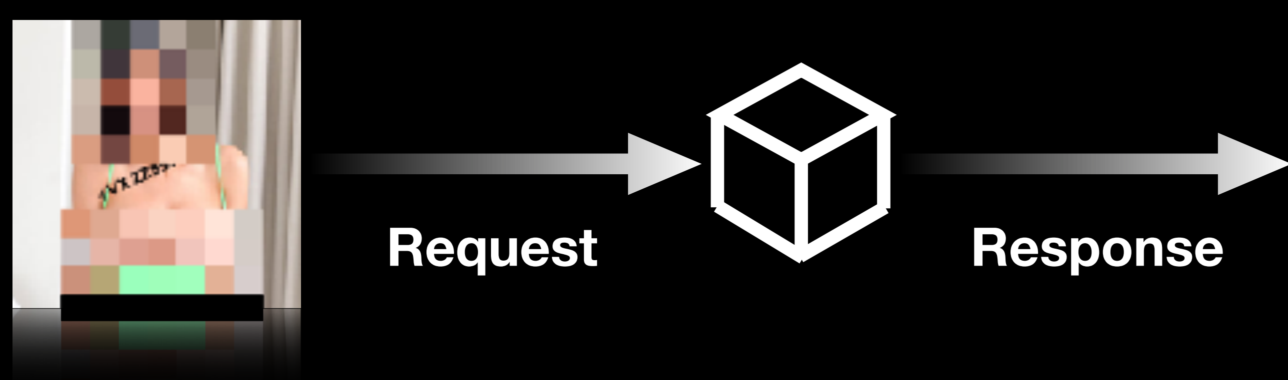
- **ML services are provided through Cloud APIs**



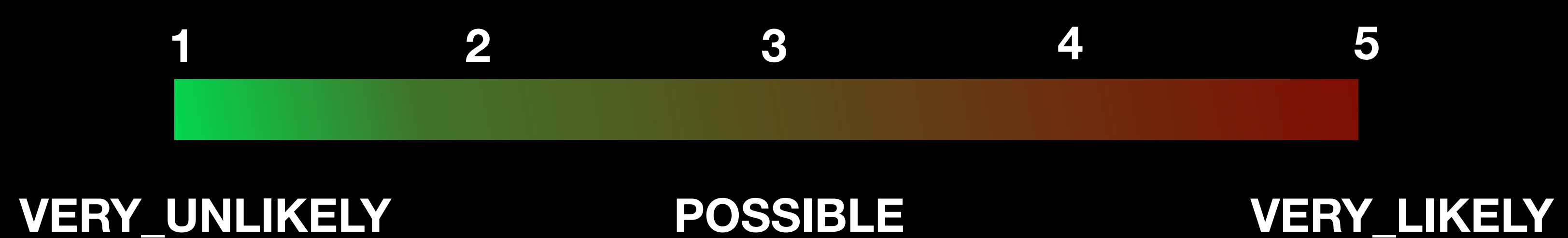
Are cloud models safe against adversary?

Case Study: SafeSearch API

- **Detect explicit content such as adult or violent content within an image sent in the query**




```
"safeSearchAnnotation": {
    "adult": "VERY_LIKELY",
    "spoof": "VERY_UNLIKELY",
    "medical": "VERY_UNLIKELY",
    "violence": "VERY_UNLIKELY",
    "racy": "VERY_LIKELY",
}
```

Is this black-box model safe against fraudster?

Spatial Attacks

- **We implement adversarial spatial transformations on images with explicit contents that allow evasion**

(adult, racy)	**Origin**	**Framing**	**Perspective**	**Affine**
	(5, 5)	**(2, 3)**	**(2, 4)**	**(1, 2)**

- **Attack evaluation:** **100** crawled porn images with **100** queries each to the *Safe Search* API using our mixed spatial attack transformations

Spatial Attacks

- Empirical results show that Safe Search API is vulnerable to spatial attacks
  - 69% images adult $\leq$ 2
  - 40% images (adult, racy) $\leq$ (2, 2)
- Potential causes:
  - Not enough spatial data augmentations
  - Preprocessing not cropping out region of interest

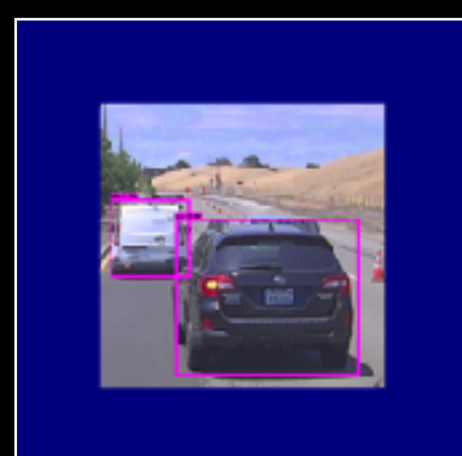
Is spatial attack generally applicable to cloud vision models?

Object Detection API

- **Object localization API is Robust against spatial attacks:**
  - **Multiple objects**
  - **Knowledge Graph**
  - **Bounding boxes**
  - **Scores**


```
"localizedObjectAnnotations": [
    {
        "name": "Van",
        "score": 0.89648587,
        "normalizedVertices": [
            {"x": 0.32076266,"y": 0.78941387},
            {"x": 0.43812272,"y": 0.78941387},
            {"x": 0.43812272,"y": 0.97331065},
            {"x": 0.32076266,"y": 0.97331065}
        ]
    }]
```


Origin



Framing

Perspective

Affine

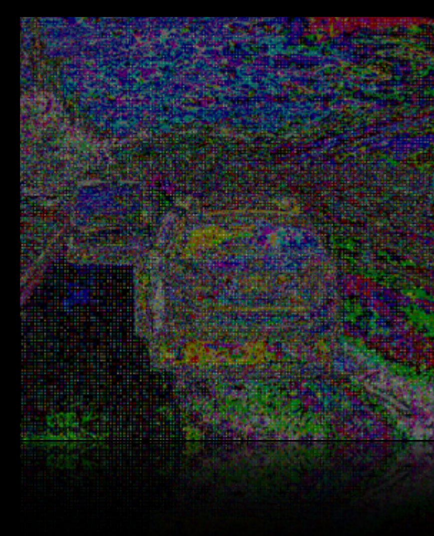
Attacks Overview

Introducing Fingerprinting attack that generates adversary examples efficiently against cloud vision models.

Adversarial Threat to DNN

- **Adversarial Examples**: inputs to ML models that an attacker has **intentionally** designed to **fool** the models such as:

+

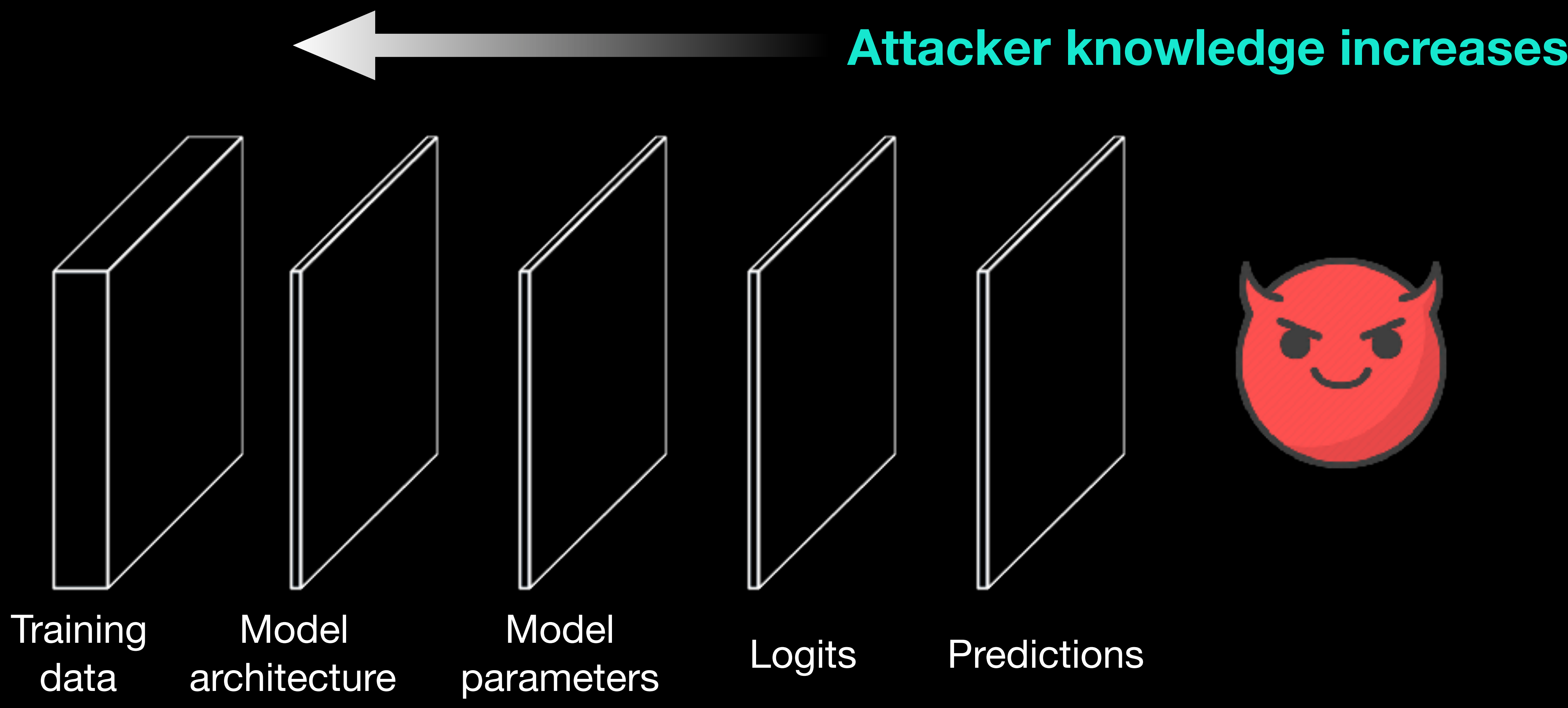


=

x ϵ (* 100) $f(x+\epsilon)$

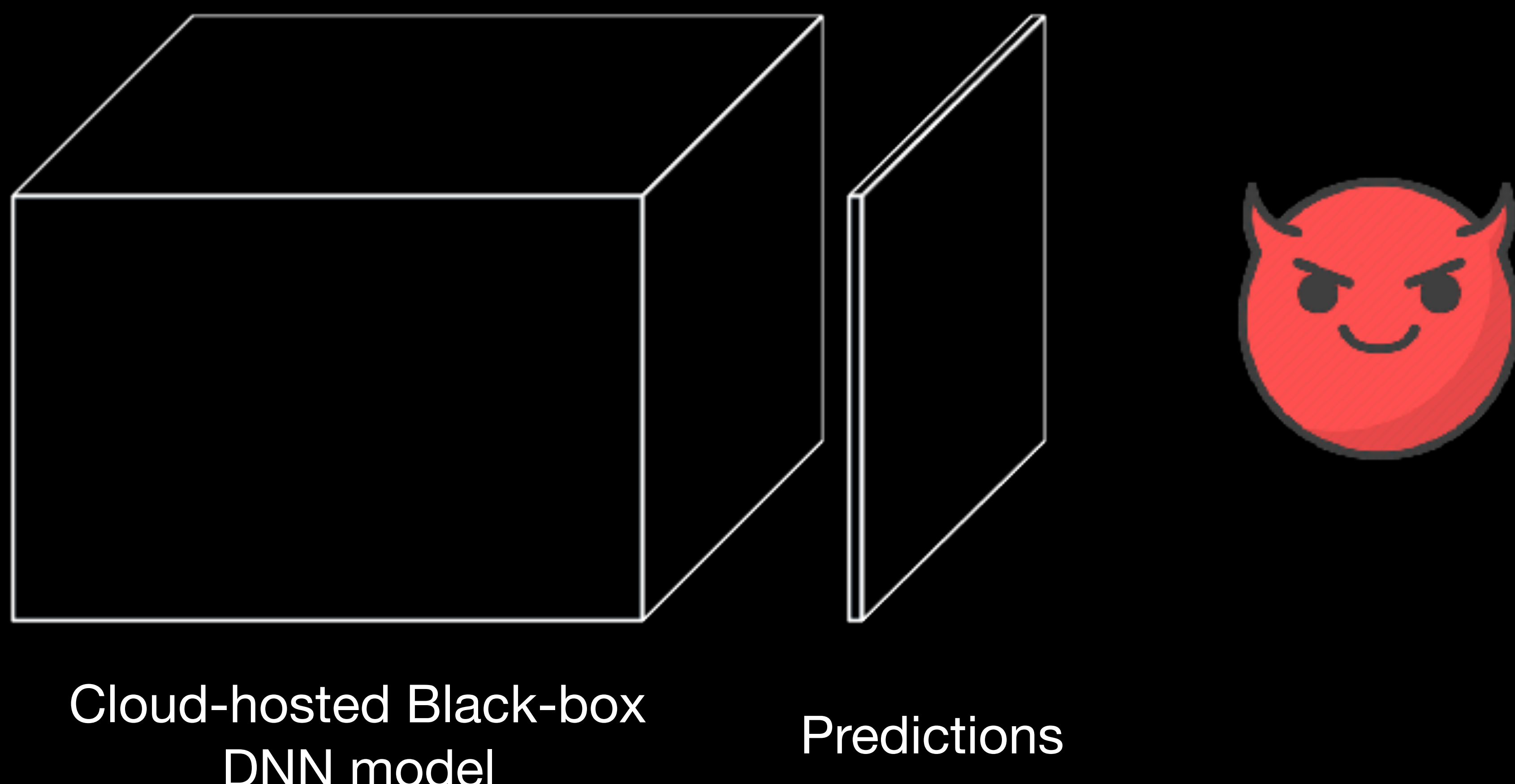
White-box vs. Black-box

- **Adversarial usually requires white-box access to model**
  - Requires **gradient** information to generate adversarial perturbations



White-box vs. Black-box

- **Cloud AI models are black-box to attackers**



Black-box provides a FALSE sense of security

Stealing the secret sauce of cloud models leveraging transfer learning

Transfer Learning

- **Pre-trained ConvNet used as feature extractor**

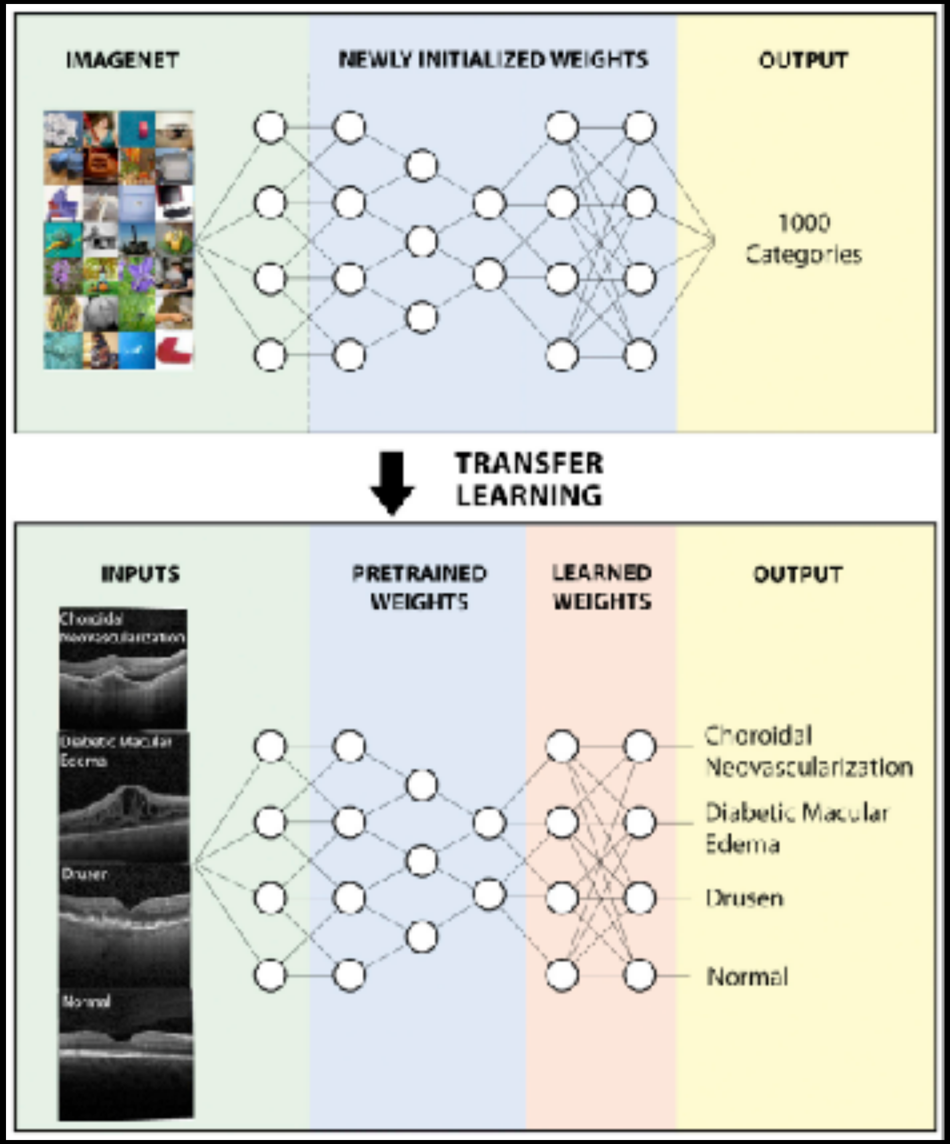


Image from Intel AI Academy

Transfer Learning

- **Pre-trained ConvNet used as feature extractor**
  - **Deep-layer feature extractor**
  - **Mid-layer feature extractor with fine-tune**

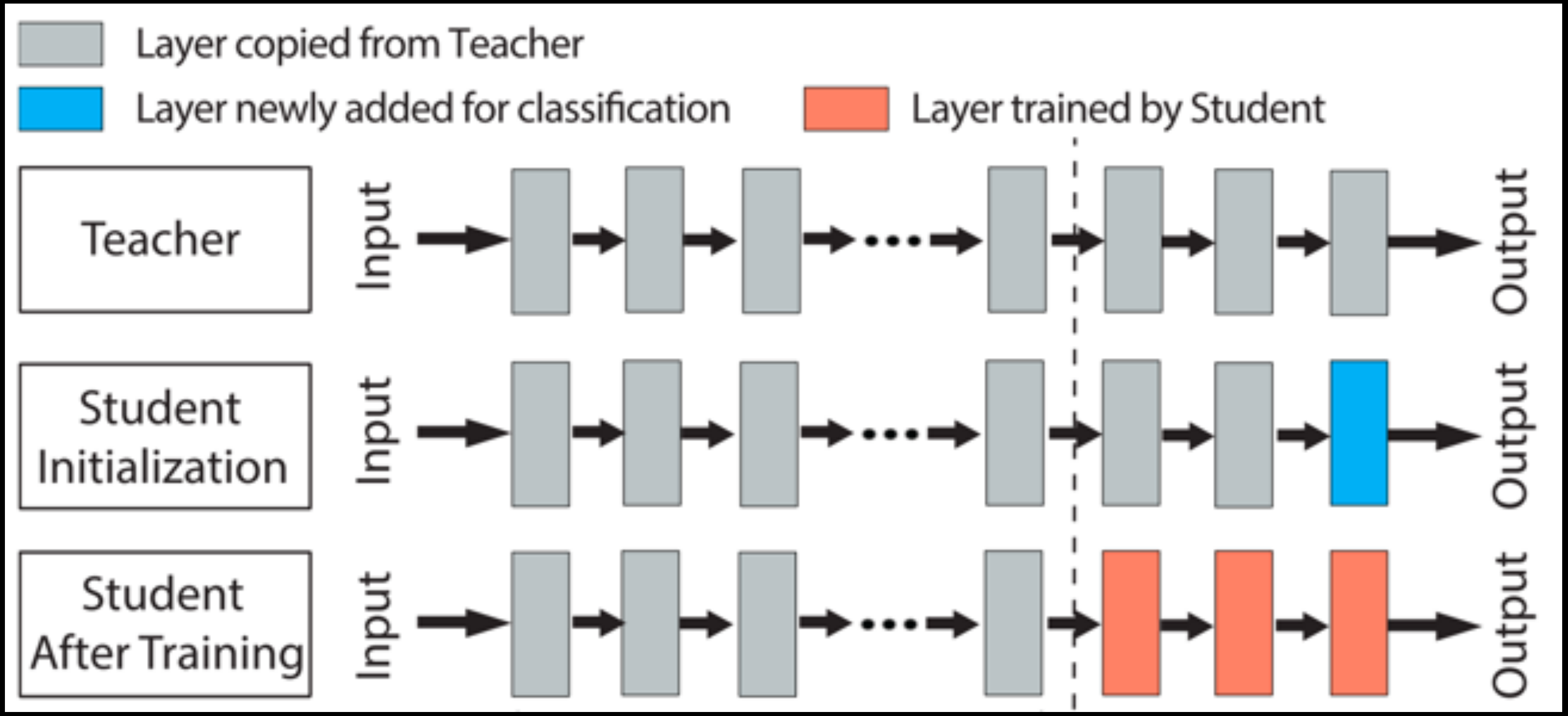


Image from Wang et al.

Object Detection Models

- YOLO v3 as an example

Insights: Adversarial sample fools layer *K* also fools the model

Fingerprinting attack
against *Object Detection* API

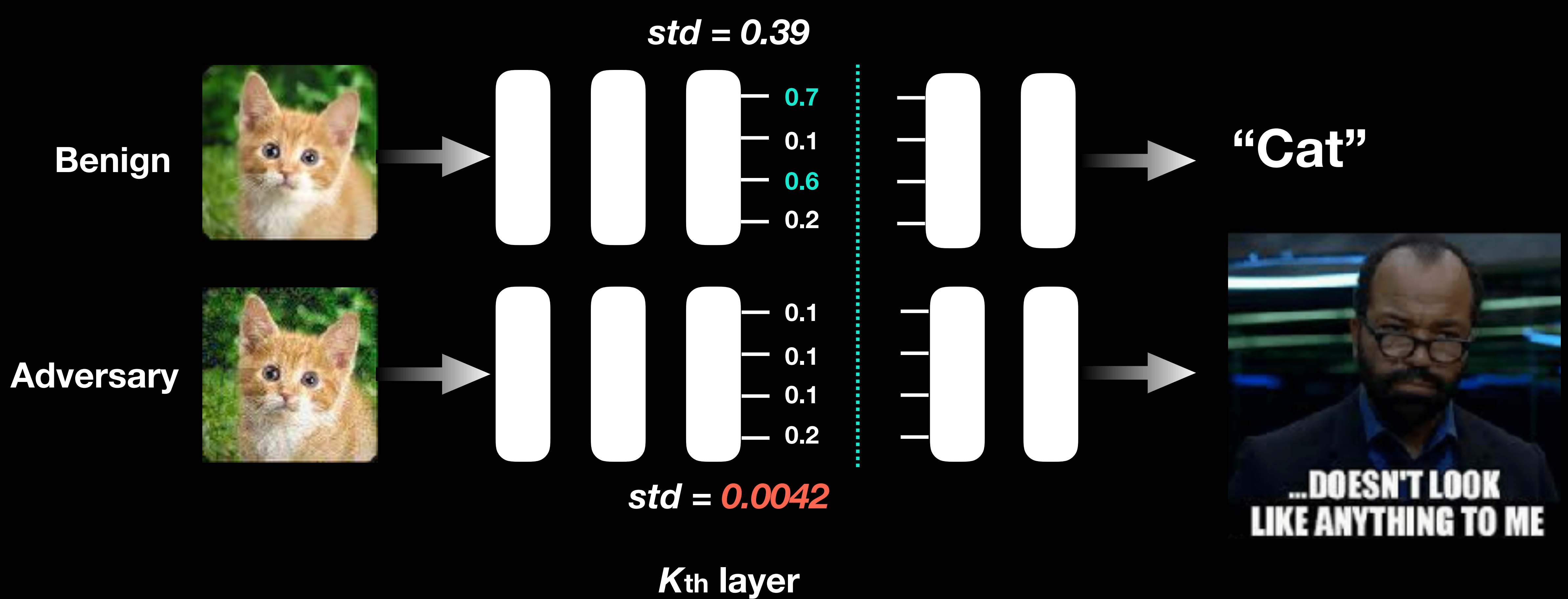
1. Identify the feature extractor that the target model is pre-trained on with a few queries

2. Generating adversarial samples on white-box pre-trained model

3. Attack black-box model using the samples

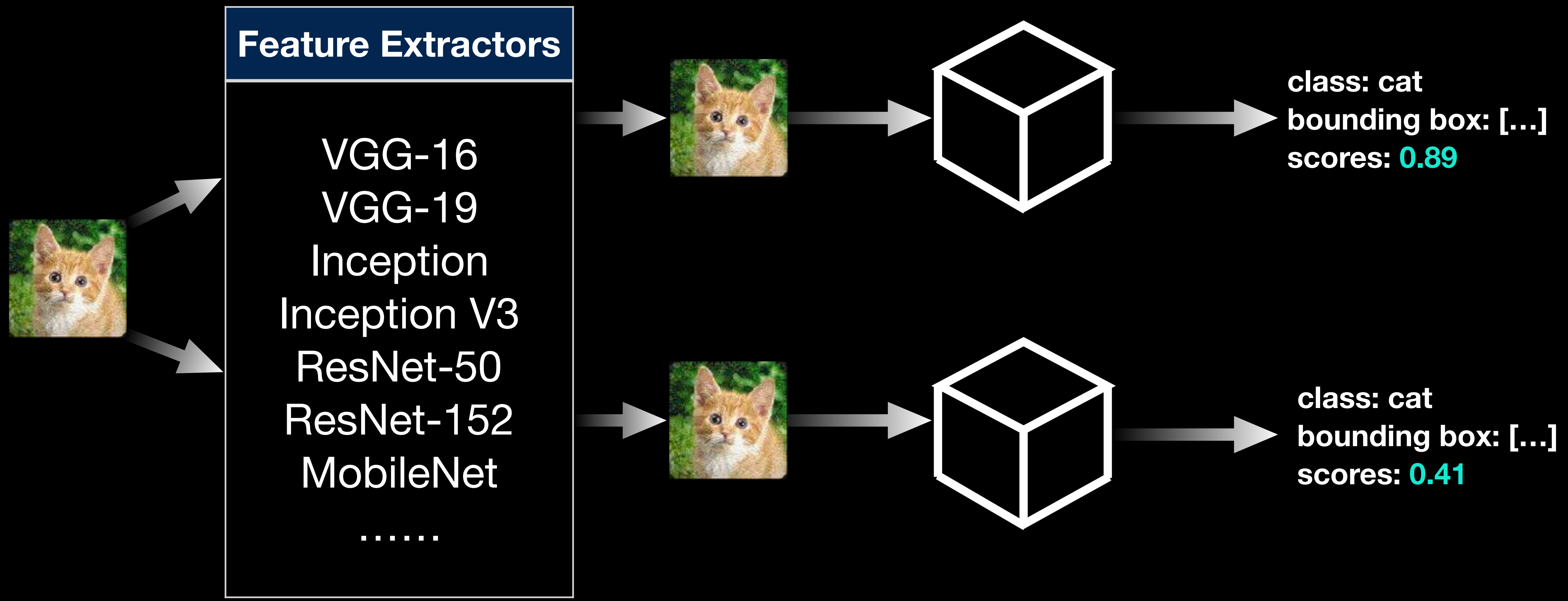
Target Internal Layer

- Target: Minimize "**dispersion**" of logits at layer K
  - Dispersion measures: Gini coefficient, standard deviation, etc.
  - "Recognizable" images will have **high** dispersion
  - **Low** dispersion at layer K results in **low** confidence score at final layer



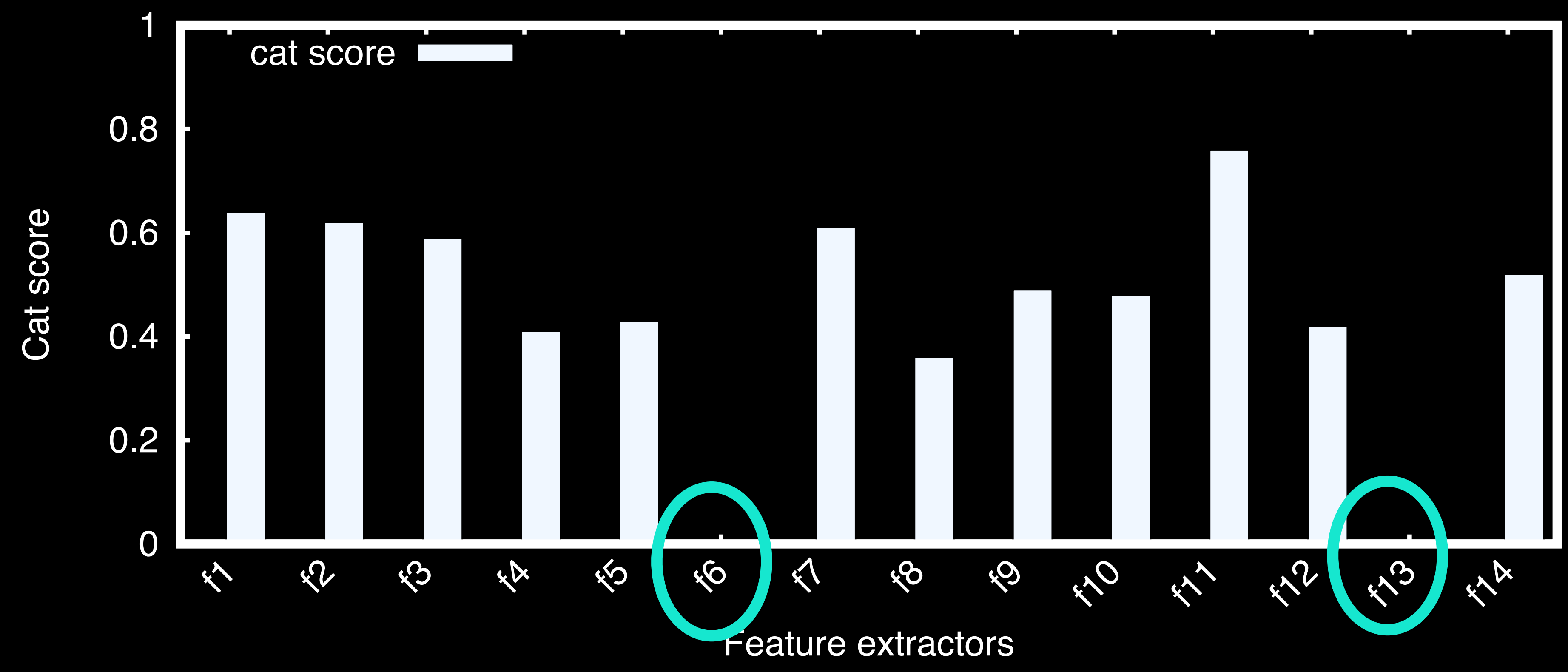
Fingerprinting Feature Extractor (1)

- For each popular feature extractor, generate samples that minimize the dispersions of each of the last few layers.
- Query with the samples and monitors the variation of score



Fingerprinting Feature Extractor (2)

- Identifying the feature extractor used in cloud models

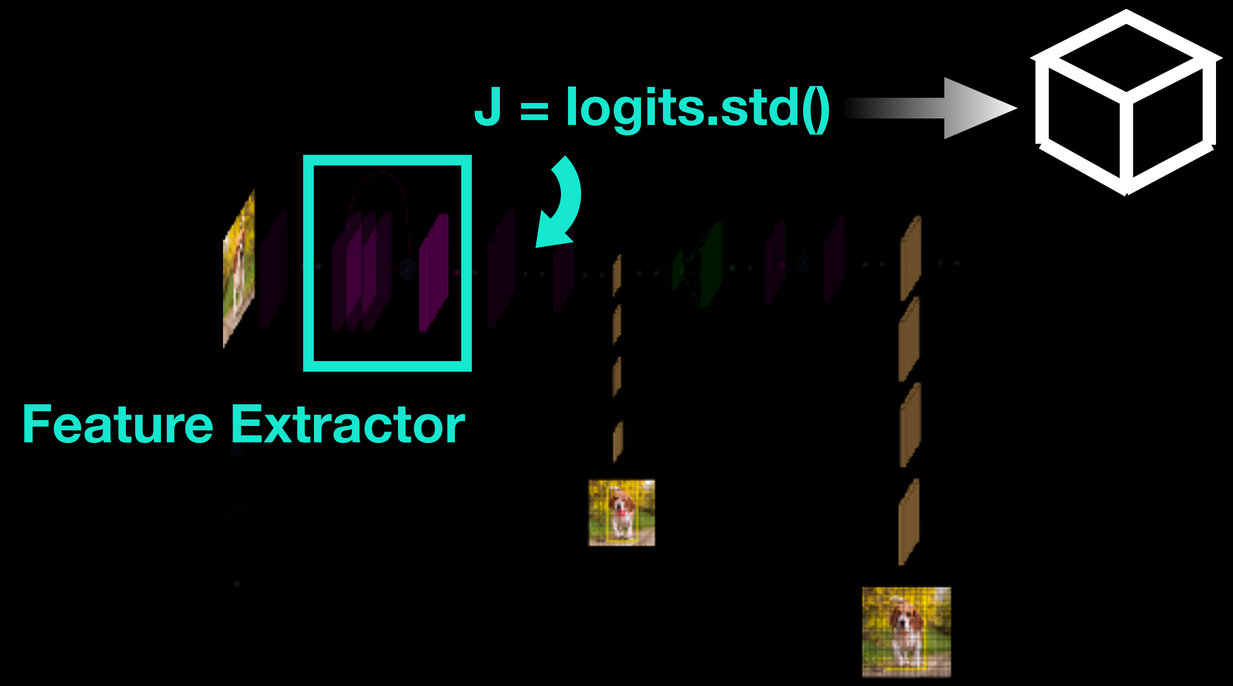


- Iterative gradient sign method on $\boldsymbol{f_6}$ and $\boldsymbol{f_{13}}$

$$x^{adv} = x - \epsilon \cdot sign(\nabla_x J(\cdot))$$

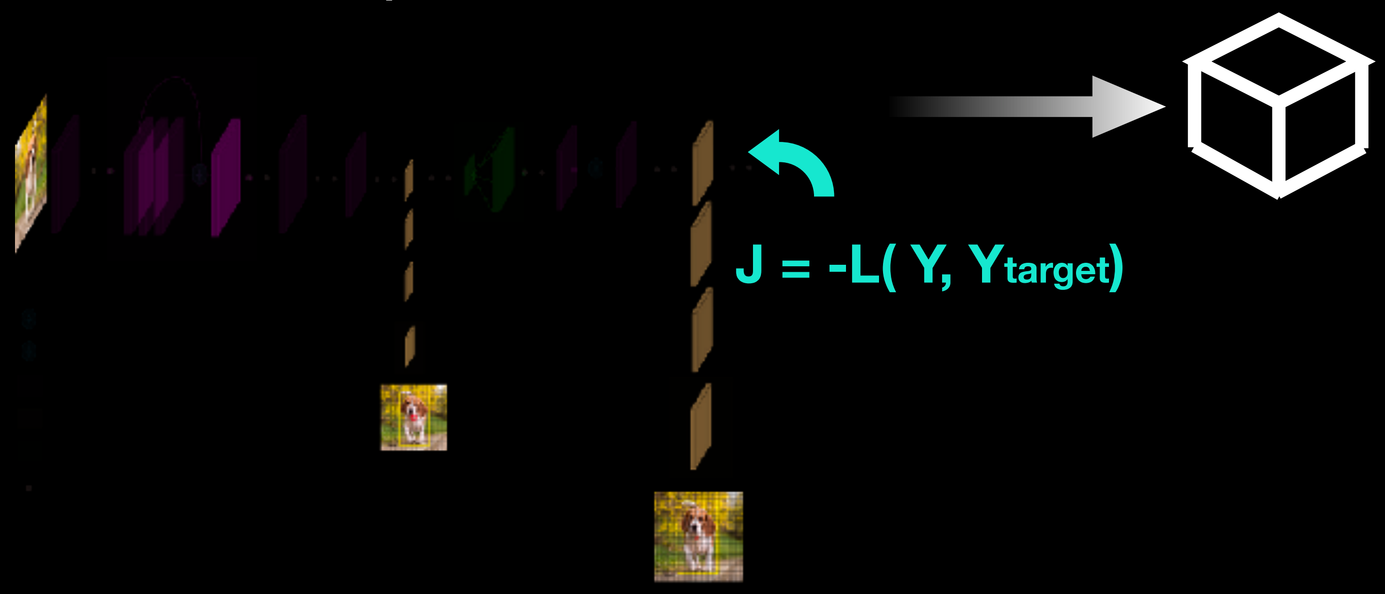
White-box Generation

- Choices of attack target ***J***:
  - Dispersion of feature extractor: **high** success rate, requires **large** perturbation



White-box Generation

- Choices of attack target ***J***:
  - Dispersion of feature extractor: **high** success rate, requires **large** perturbation
  - Target object score: **minimum** perturbation, **lower** success rate



Attack Evaluation

- **Achieved high evasion rate with limited budget (queries)**

Method	# of queries attempted	Evasion rate
Dispersion	Limit attack budget (2 queries)	33%
	No budget limit (100 queries):	86%
Target	Limit attack budget (2 queries)	16%
	No budget limit (100 queries):	65%

Van: 0.89
Car: 0.93

Origin

Van: 0.81

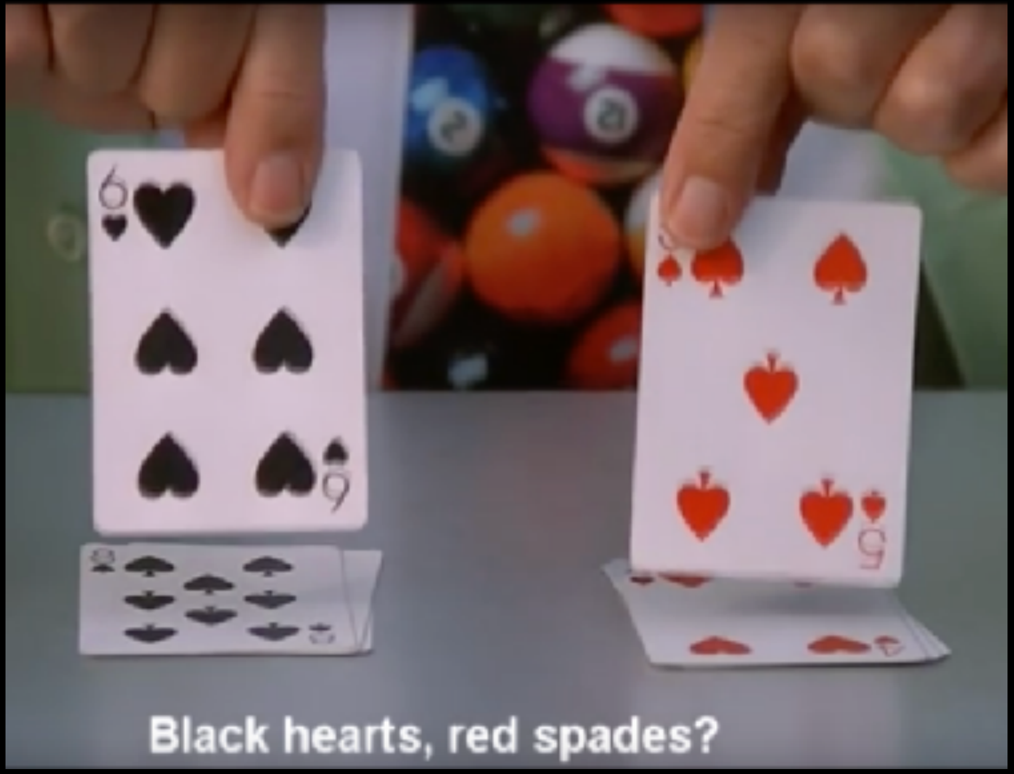
Target score attack

Van: 0.59

Dispersion attack

Conclusion

- **Black-box only provides a false sense of security.**
  - **Fooling prediction result by targeting internal layers is generally applicable to DNNs**
  - **Potential solution: hardening model with adversarial training**



Adversarial example to human from *Interstate 60*