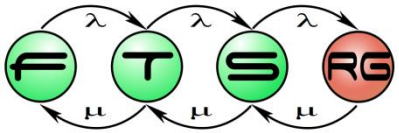
Adversarial attacks in practice

Gábor Rabatin

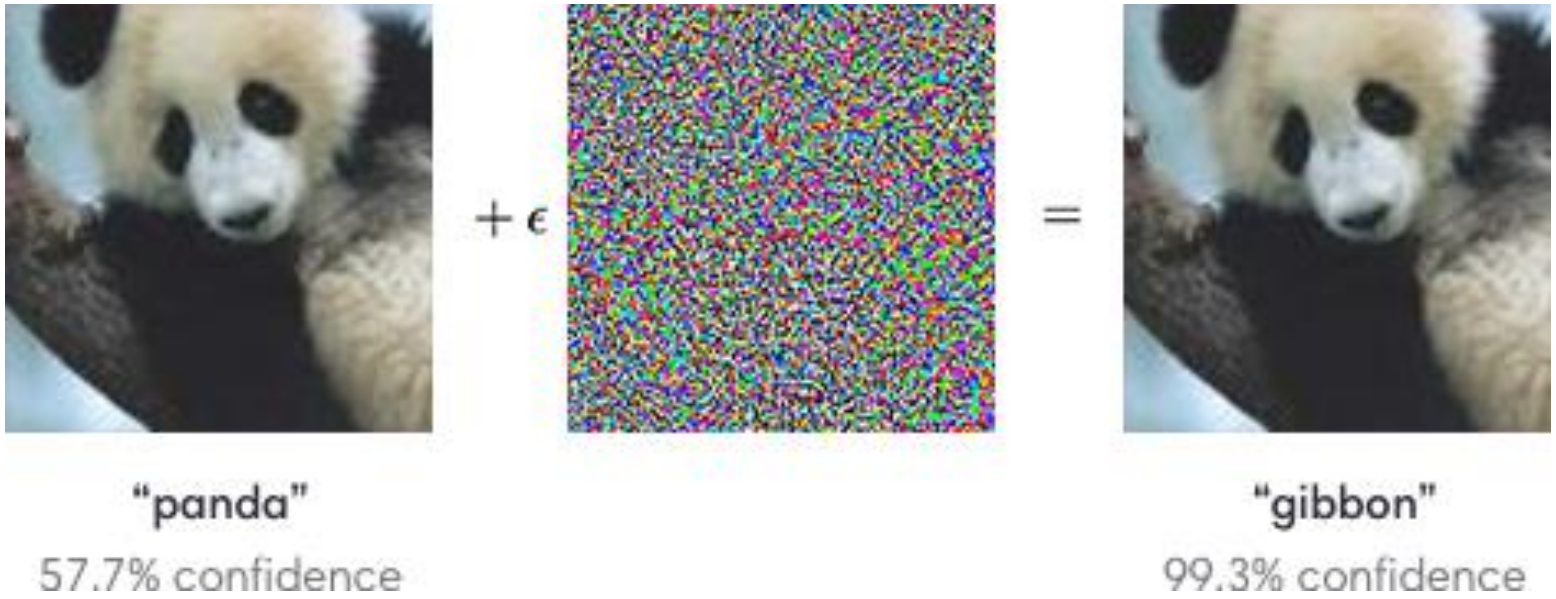
Budapest University of Technology and Economics
Fault Tolerant Systems Research Group



Agenda

- Background
- Attacking approaches
- Defending approaches
- Verification
- Demo

Adversarial attack



- **Adversarial example:** *„a pair of inputs x; x' is an adversarial example for a classifier, if a reasonable person would say they are of the same class but the classifier produces significantly different outputs."*
- *„they're like optical illusions for machines"*

BACKGROUND

Fast gradient sign method (FGSM)

- *Ian J. Goodfellow, Jonathon Shlens & Christian Szegedy: Explaining And Harnessing Adversarial Examples*

$$x^{adv} = x + \varepsilon \cdot \mathtt{sign}(\nabla_x J(x, y_{true})),$$

- Pixel-wide perturbation in the direction of gradient
- Computed in one step → very efficient

Targeted-FGSM

- *Alexey Kurakin, Ian J. Goodfellow, Samy Bengio: Adversarial Examples In The Physical World*

$$x^{adv} = x - \varepsilon \cdot \mathtt{sign}(\nabla_x J(x, y_{target})),$$

- In the negative direction in respect to the target class

Iterative-FGSM

- *Alexey Kurakin, Ian J. Goodfellow, Samy Bengio: Adversarial Machine Learning At Scale*

$$x_0^{adv} = x, \quad x_{t+1}^{adv} = x_t^{adv} + \alpha \cdot \mathtt{sign}\left(\nabla_x J(x_t^{adv}, y)\right).$$

- Smaller steps
- Higher success rate in white box attacks

NIPS 2017 Competition

- *„Adversarial Attacks and Defences”* Kaggle competition in 2017 by Google Brain

- 3 categories:
  - targeted adversarial attack,
  - non-targeted adversarial attack
  - and defense against adversarial attacks

Momentum Iterative-FGSM

- *Tsinghua University, Intel Labs China: Boosting Adversarial Attacks with Momentum*
- Good transferability
- Performs well in black box attacks

Algorithm 1 MI-FGSM

Input: A classifier f with loss function J; a real example $\boldsymbol{x}$ and ground-truth label y;

Input: The size of perturbation ϵ; iterations T and decay factor μ.

Output: An adversarial example $\boldsymbol{x}^*$ with $\|\boldsymbol{x}^* - \boldsymbol{x}\|_\infty \leq \epsilon$.

1: $\alpha = \epsilon/T$;

2: $\boldsymbol{g}_0 = 0$; $\boldsymbol{x}_0^* = \boldsymbol{x}$;

3: **for** $t = 0$ to $T - 1$ **do**

4: Input $\boldsymbol{x}_t^*$ to f and obtain the gradient $\nabla_{\boldsymbol{x}} J(\boldsymbol{x}_t^*, y)$;

5: Update $\boldsymbol{g}_{t+1}$ by accumulating the velocity vector in the gradient direction as

$$\boldsymbol{g}_{t+1} = \mu \cdot \boldsymbol{g}_t + \frac{\nabla_{\boldsymbol{x}} J(\boldsymbol{x}_t^*, y)}{\|\nabla_{\boldsymbol{x}} J(\boldsymbol{x}_t^*, y)\|_1}; \quad (6)$$

6: Update $\boldsymbol{x}_{t+1}^*$ by applying the sign gradient as

$$\boldsymbol{x}_{t+1}^* = \boldsymbol{x}_t^* + \alpha \cdot \text{sign}(\boldsymbol{g}_{t+1}); \quad (7)$$

7: **end for**

8: **return** $\boldsymbol{x}^* = \boldsymbol{x}_T^*$.

Attack Ensemble Models

- What if there are more than models?

Algorithm 2 MI-FGSM for an ensemble of models

Input: The logits of K classifiers $l_1, l_2, ..., l_K$; ensemble weights $w_1, w_2, ..., w_K$; a real example x and ground-truth label y;

Input: The size of perturbation ϵ; iterations T and decay factor μ.

Output: An adversarial example x^* with $\|x^* - x\|_\infty \leq \epsilon$.

$\alpha = \epsilon/T$;
$g_0 = 0$; $x_0^* = x$;
for $t = 0$ to $T - 1$ **do**
    Input x_t^* and output $l_k(x_t^*)$ for $k = 1, 2, ..., K$;
    Fuse the logits as $l(x_t^*) = \sum_{k=1}^{K} w_k l_k(x_t^*)$;
    Get softmax cross-entropy loss $J(x_t^*, y)$ based on $l(x_t^*)$ and Eq. (9);
    Obtain the gradient $\nabla_x J(x_t^*, y)$;
    Update g_{t+1} by Eq. (6);
    Update x_{t+1}^* by Eq. (7);
end for
return $x^* = x_T^*$.

$$J(x, y) = -1_y \cdot \log(\text{softmax}(l(x))), \tag{9}$$

Types of Adversarial Attack

- White-box attack
  - Attacker has access to the model's parameters
- Black-box attack
  - No access to parameters, gradients
  - Uses a different model or no model
  - With hope that the examples will transfer to the target model

Examples of Adversarial Attack

- One Pixel Attack
- Physical Adversarial Examples
- Adversarial Patch
- Examples Fool Both Human and Computer
- Unrecognizable examples
- Adversarial Attack in Reinforcement Learning
- Robust Adversarial Examples

One Pixel Attack for Fooling Deep Neural Networks

- Limited scenario: only one pixel is modified

HORSE

DOG(88.0%)

SHIP

AIRPLANE(62.7%)

CAT

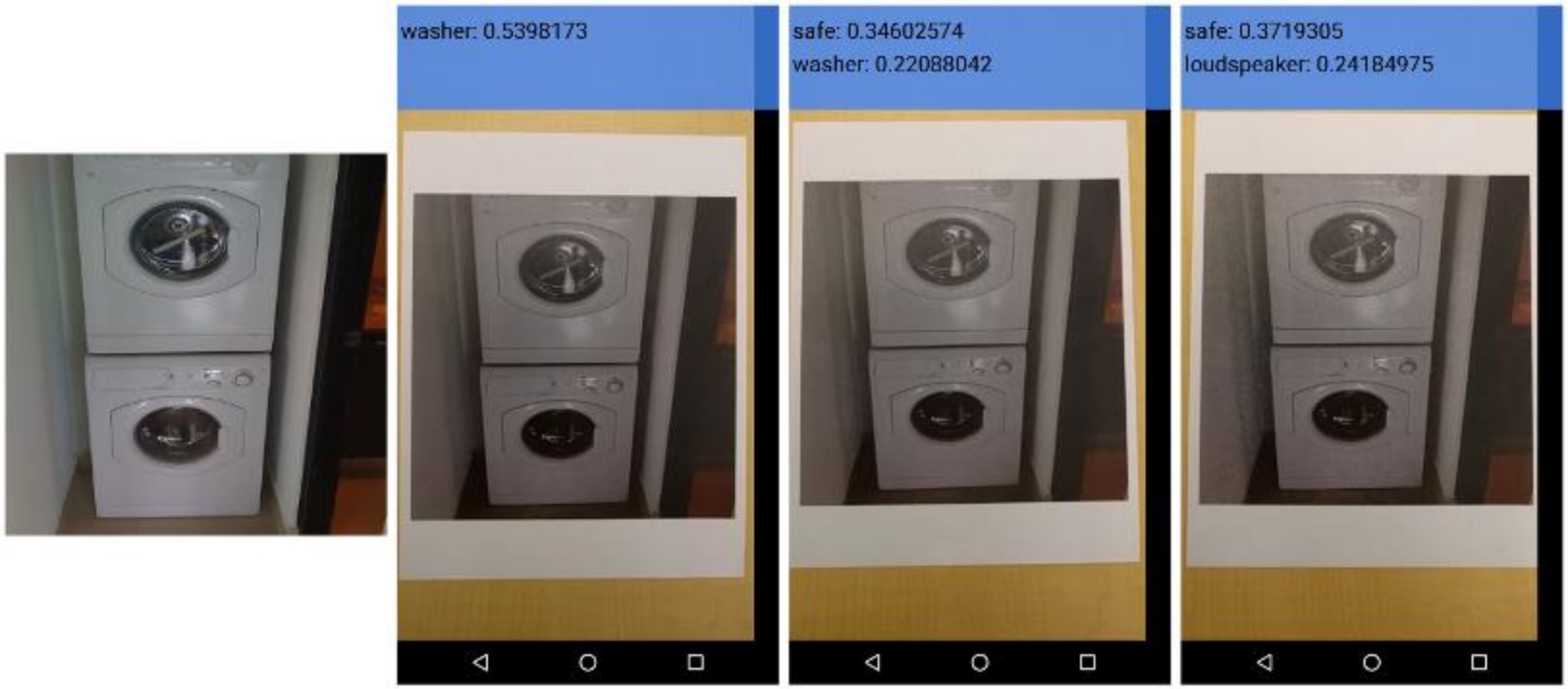
DOG(78.2%)

Planetarium

Mosque(7.81%)

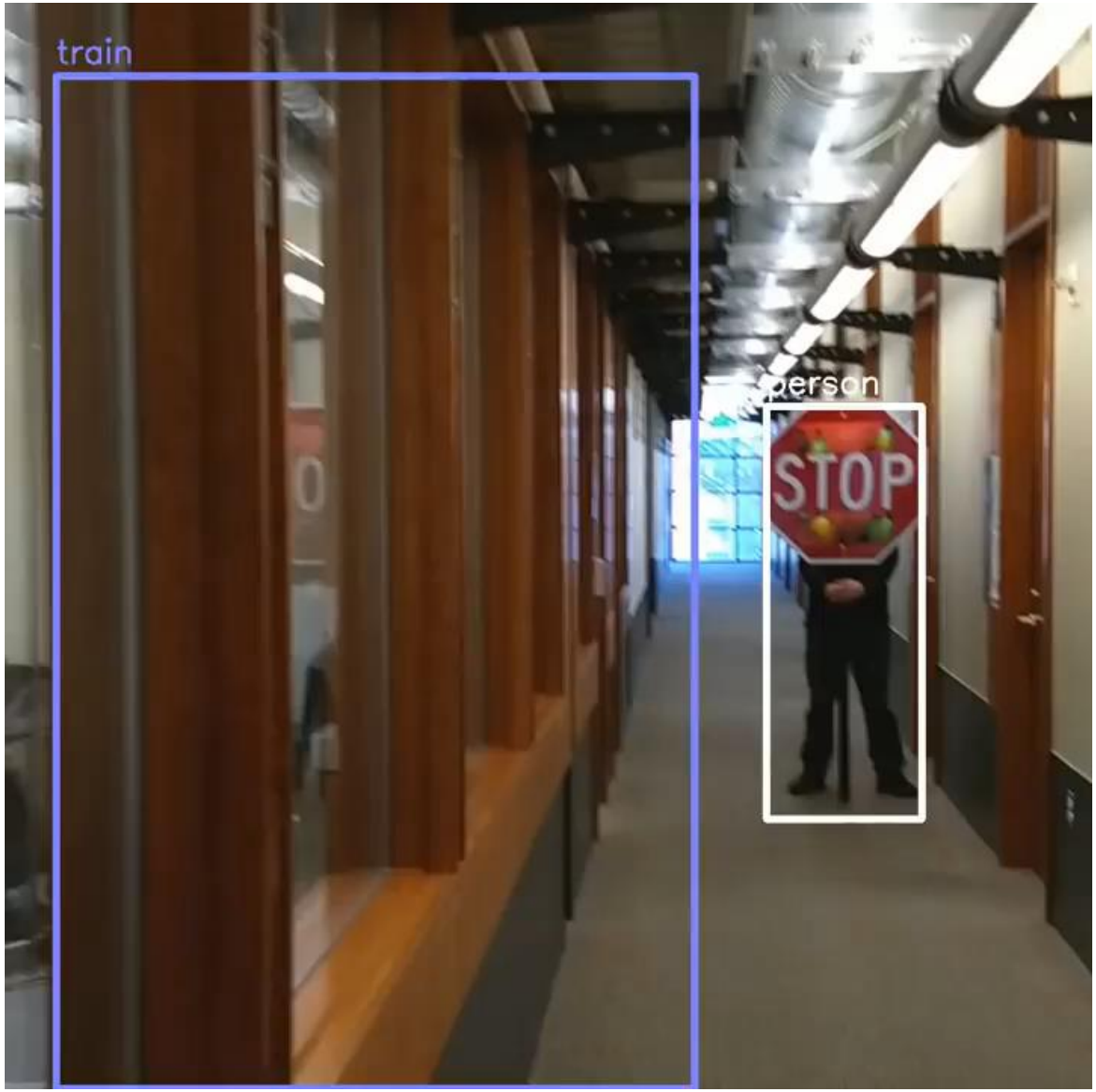
Adversarial Examples in the Physical World

- *Alexey Kurakin, Ian J. Goodfellow, Samy Bengio*
- Attacks also work in real life

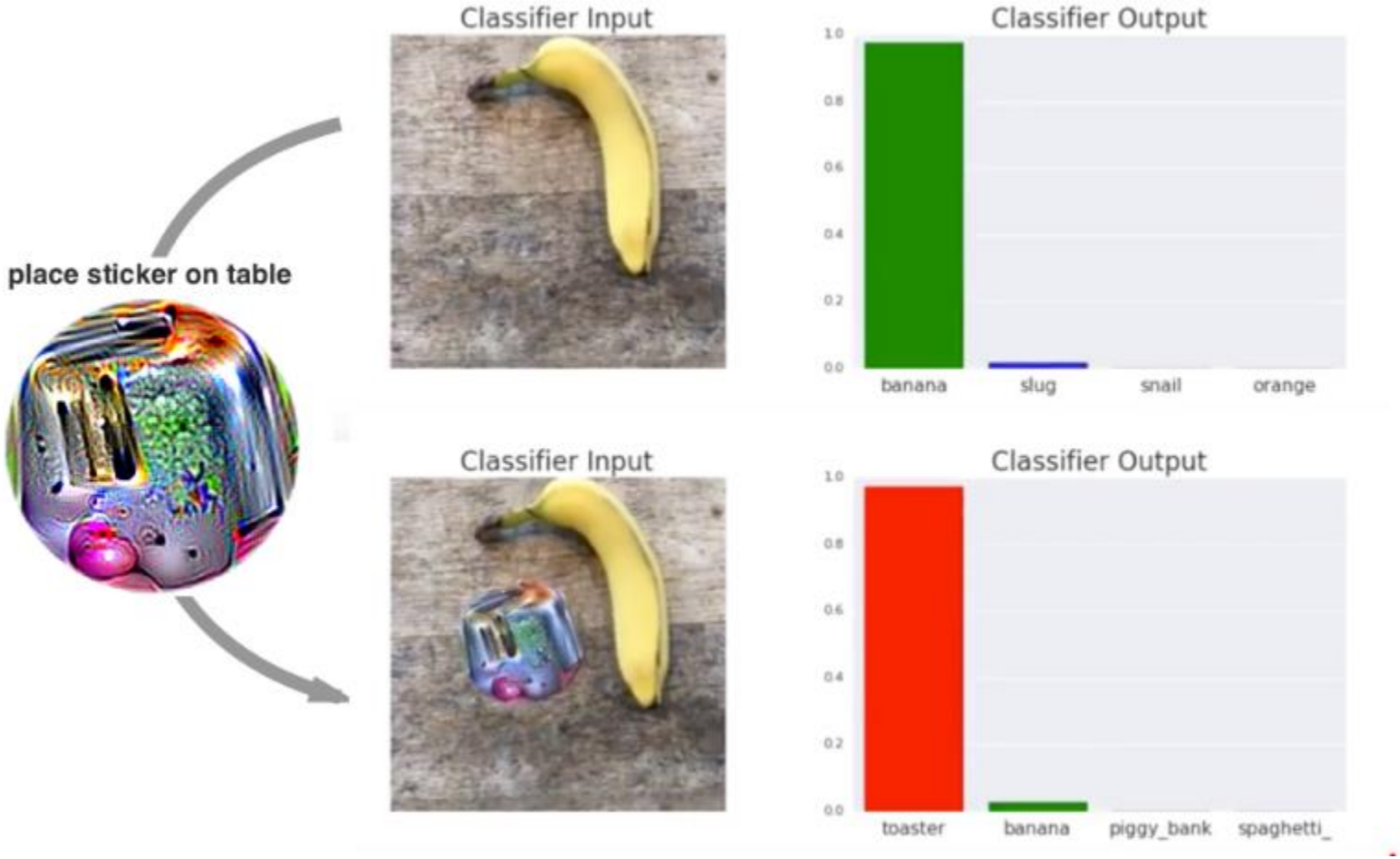


(a) Image from dataset (b) Clean image (c) Adv. image, $\epsilon = 4$ (d) Adv. image, $\epsilon = 8$

Adversarial Examples in the Physical World



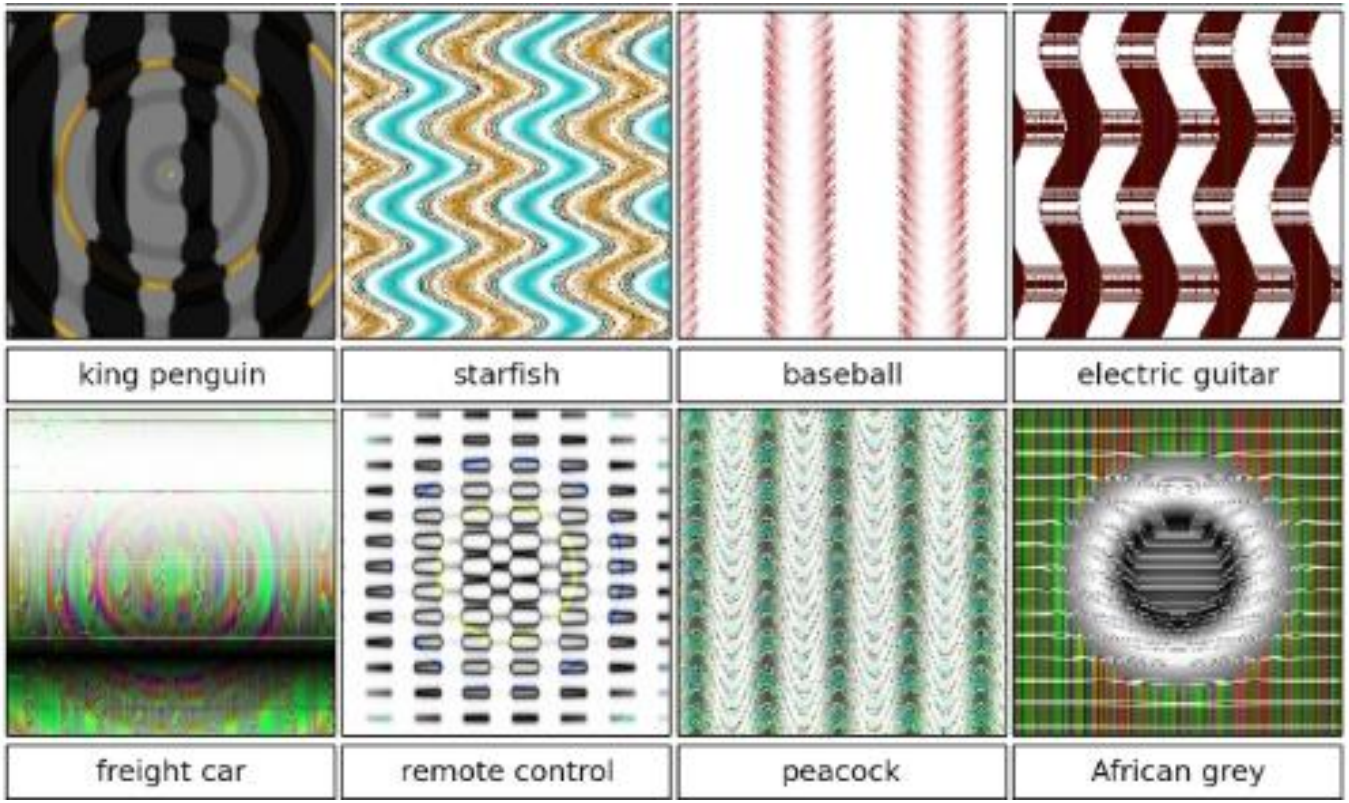
Adversarial Patch



Adversarial Examples that Fool Both Human and Computer

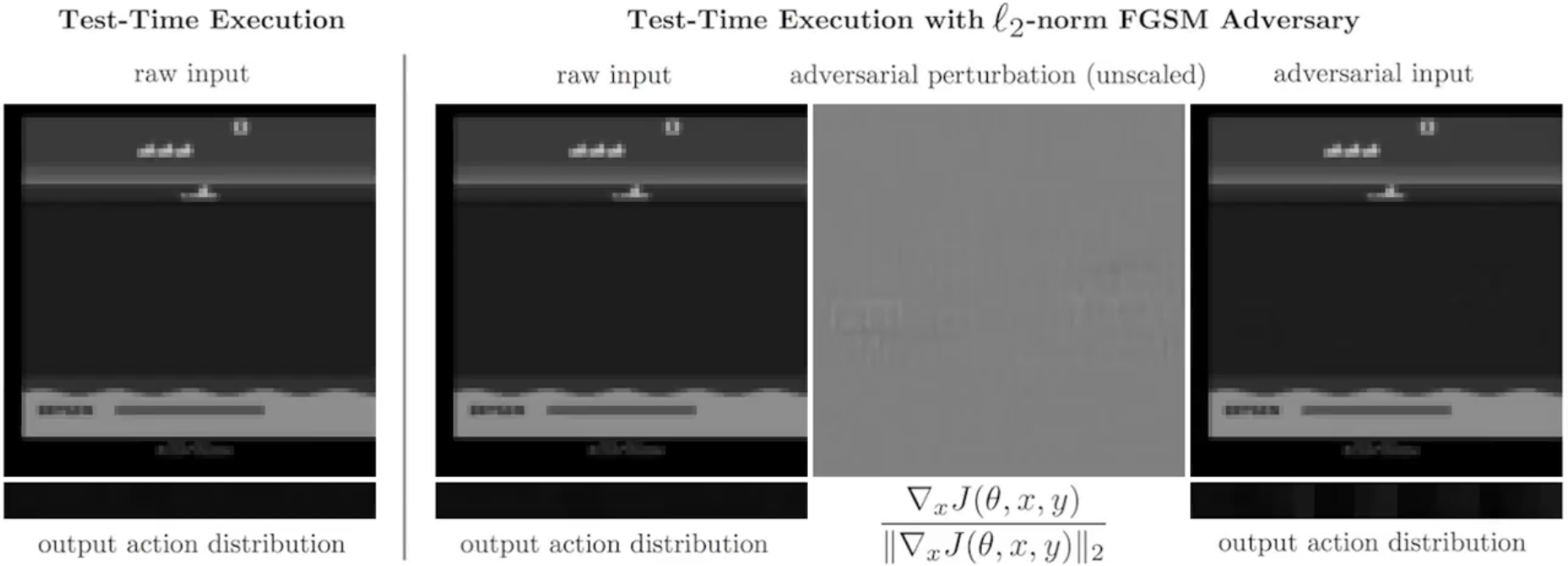
High Confidence Predictions for Unrecognizable Images

- Unrecognizable for humans, but „easily recognized” by DNNs
- Evolutionary algorithms are used

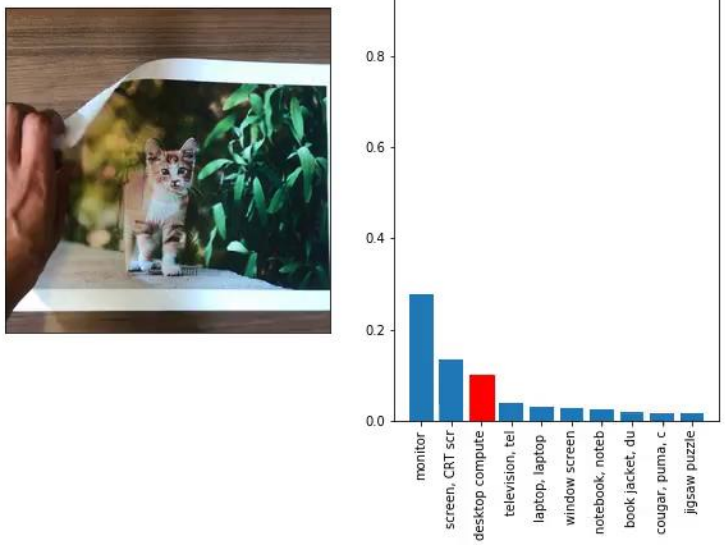


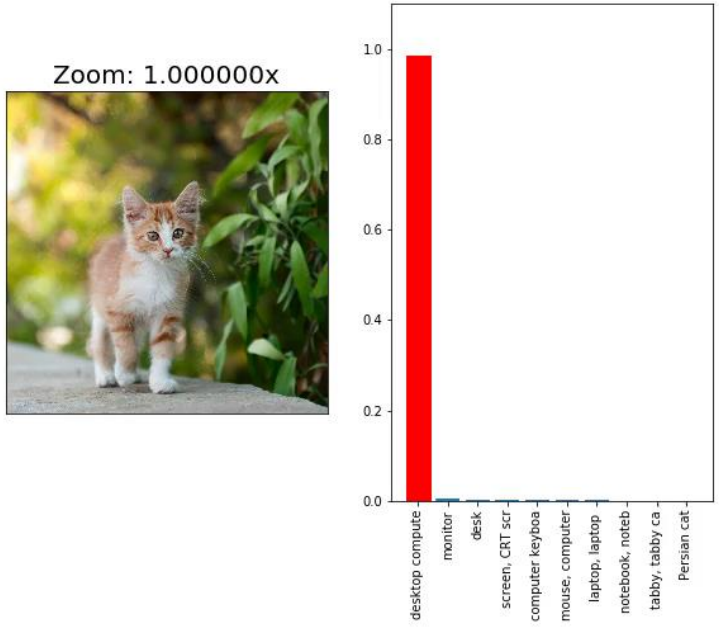
Adversarial Attack in Reinforcement Learning

- Widely used deep reinforcement learning algorithms are vulnerable too

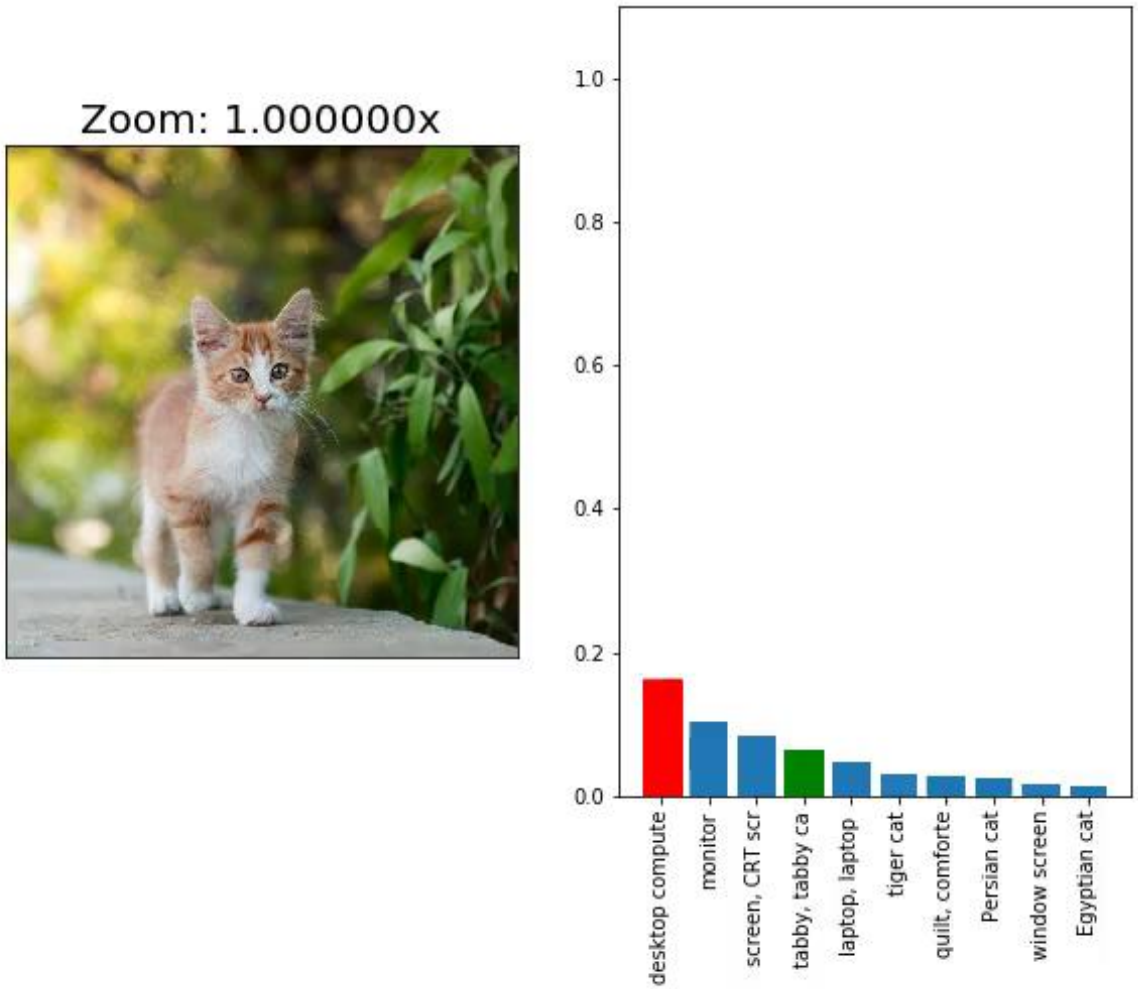


Are they robust?

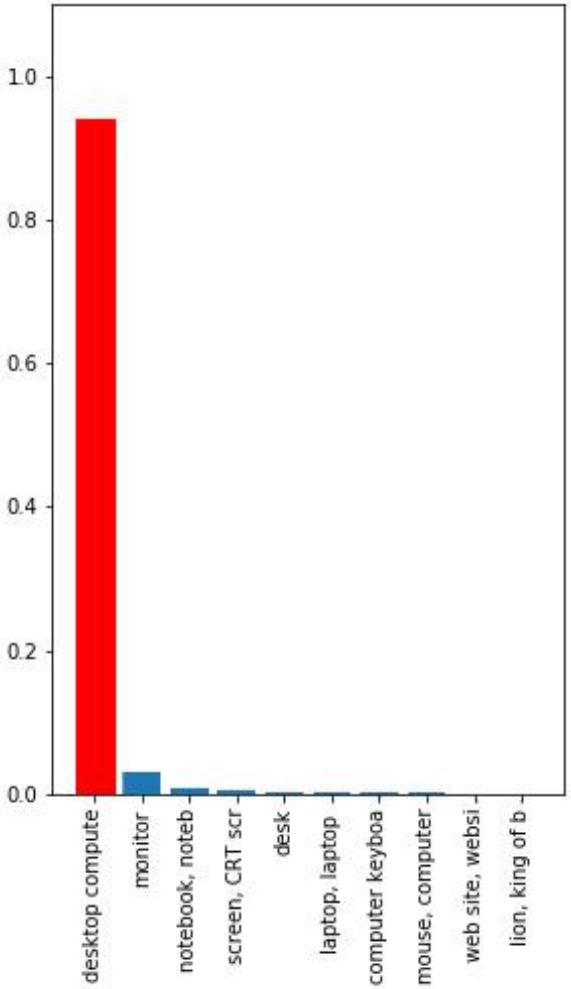




Scale-Invariant Adversarial Examples



Transformation-Invariant Adversarial Examples



Examples of Defenses

- Adversarial Training
- Defensive Distillation
- Gradient Masking
- Denoiser

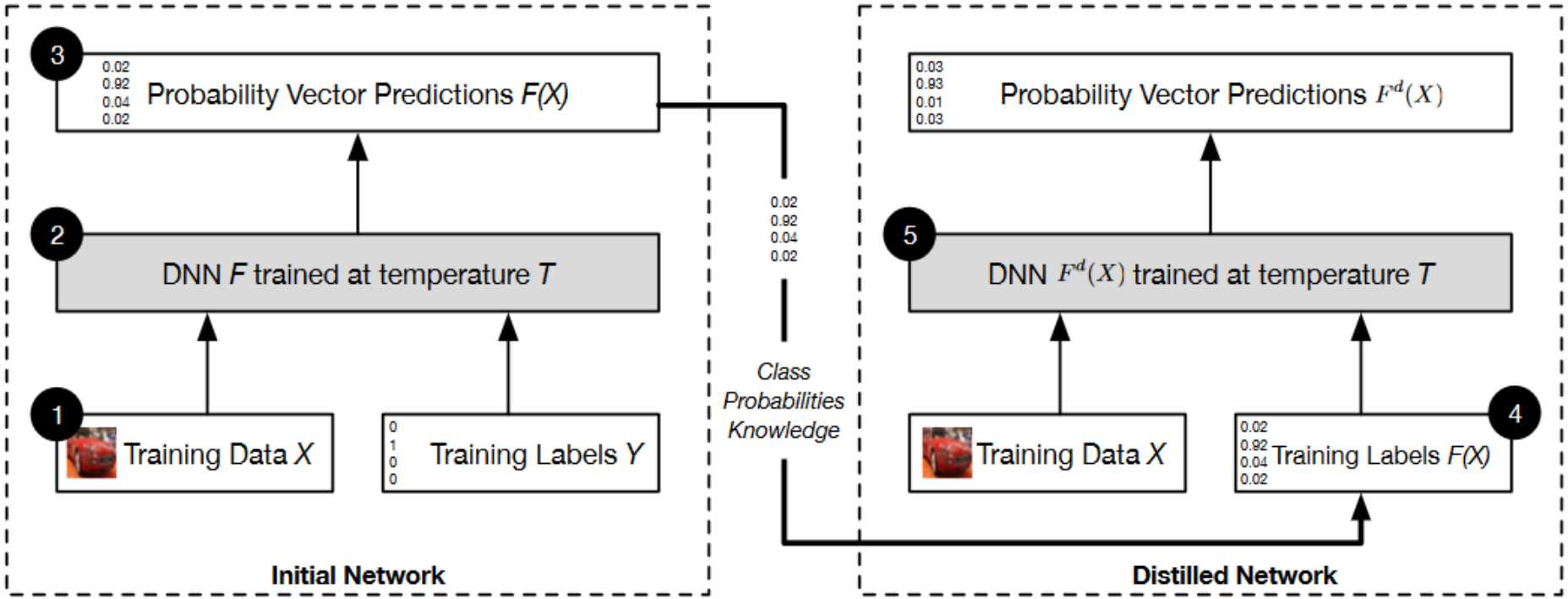
Adversarial Training

- Algorithm:
  - Generate a lot of adversarial examples
  - Retrain the model not to be fooled by them
  - Do this iteratively

- Danger of overfitting
- Less effective against black-box attacks

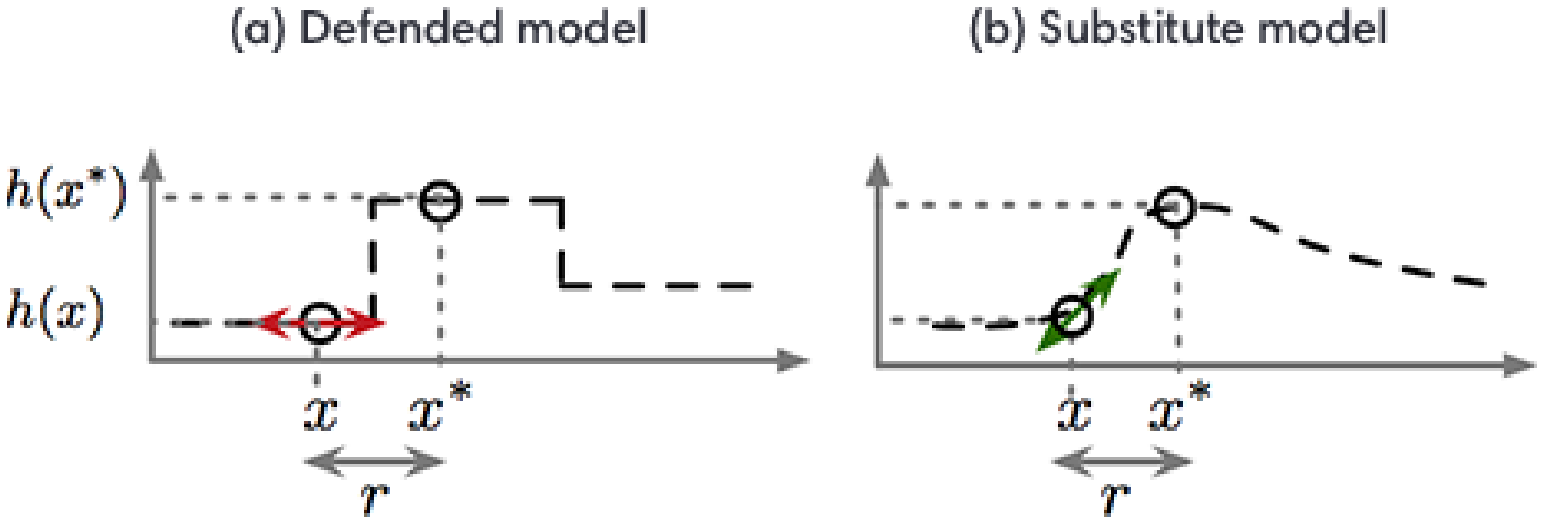
Defensive Distillation

- Train a new model with a pretrained model's output probalities
- Inspired by Geoffrey Hinton's knowledge compressing paper



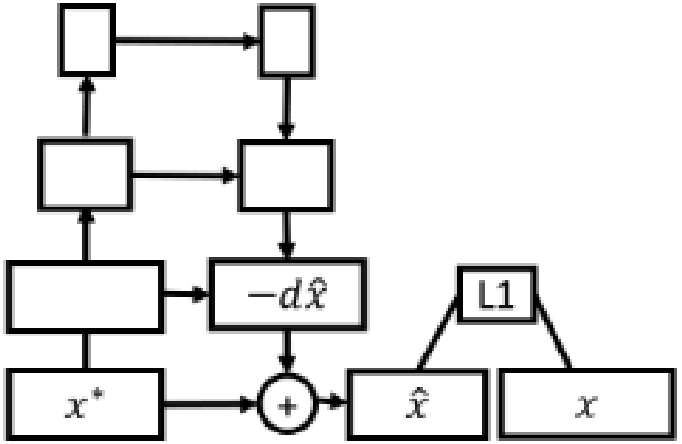
Gradient Masking – a failed defense

- Deny the attacker's access to a useful gradient
- „Most likely class" output mode, a smooth change in input doesn't change the output
- However, the model is not more robust, just fewer clues to finding the holes



High-Level Representation Guided Denoiser

- Feature guided denoiser
  - Denoising U-Net (denoising autoencoder with lateral connections)
  - Learning objective: adversarial noise
- NIPS 1st place!



- More robust to white-box and black-box attacks
- Can be trained on small subset of the images
- Can be transferred to defend other models

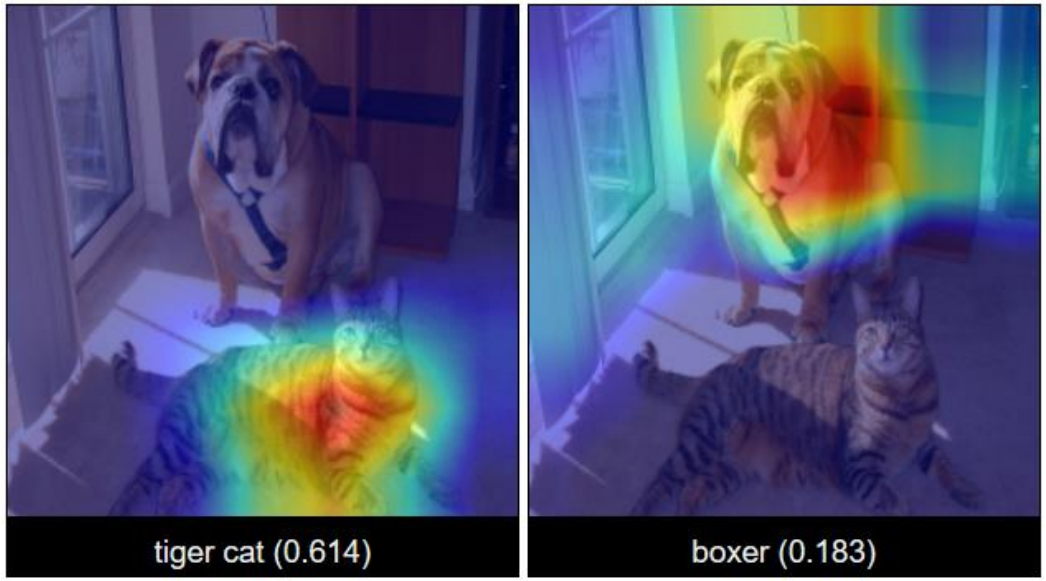
The problems with defending

- It requires models to produce good outputs for **every possible input**
- Techniques are not adaptive

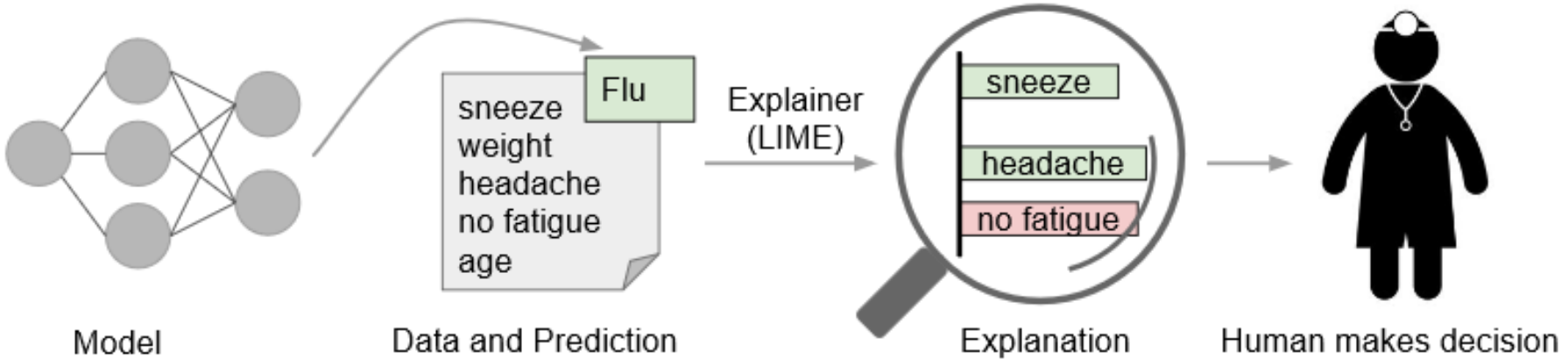
- But there are some tools…

Tools

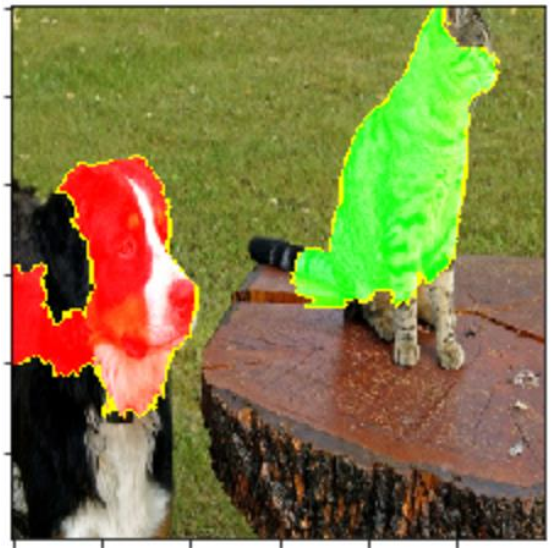
- Cleverhans
  - *Ian J. Goodfellow and Nicolas Papernot*
  - Tool for developing more robust models
  - Attacking and defending techniques implemented
- Darkon
  - Helps understanding the decision of DNNs
  - Filters bad training examples
  - Grad-CAM

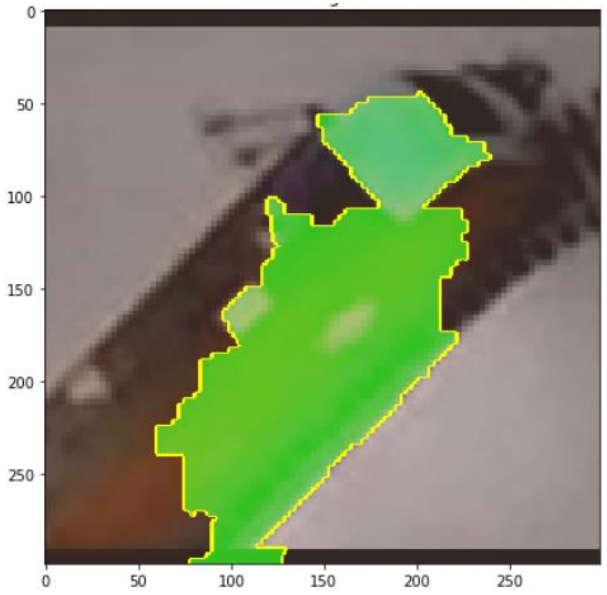


Tools



- LIME
- Helps interpretability

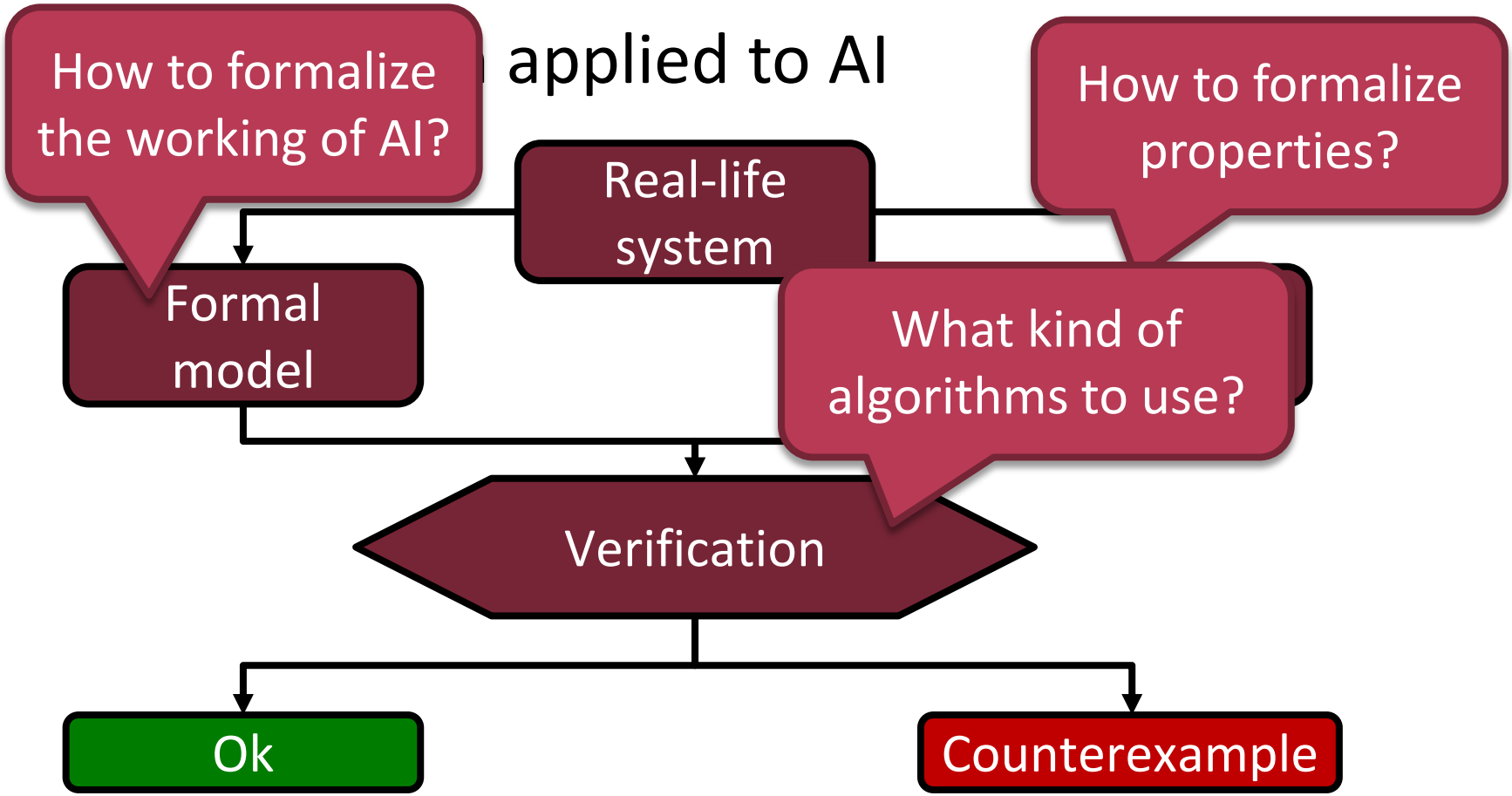




VERIFICATION

Verification

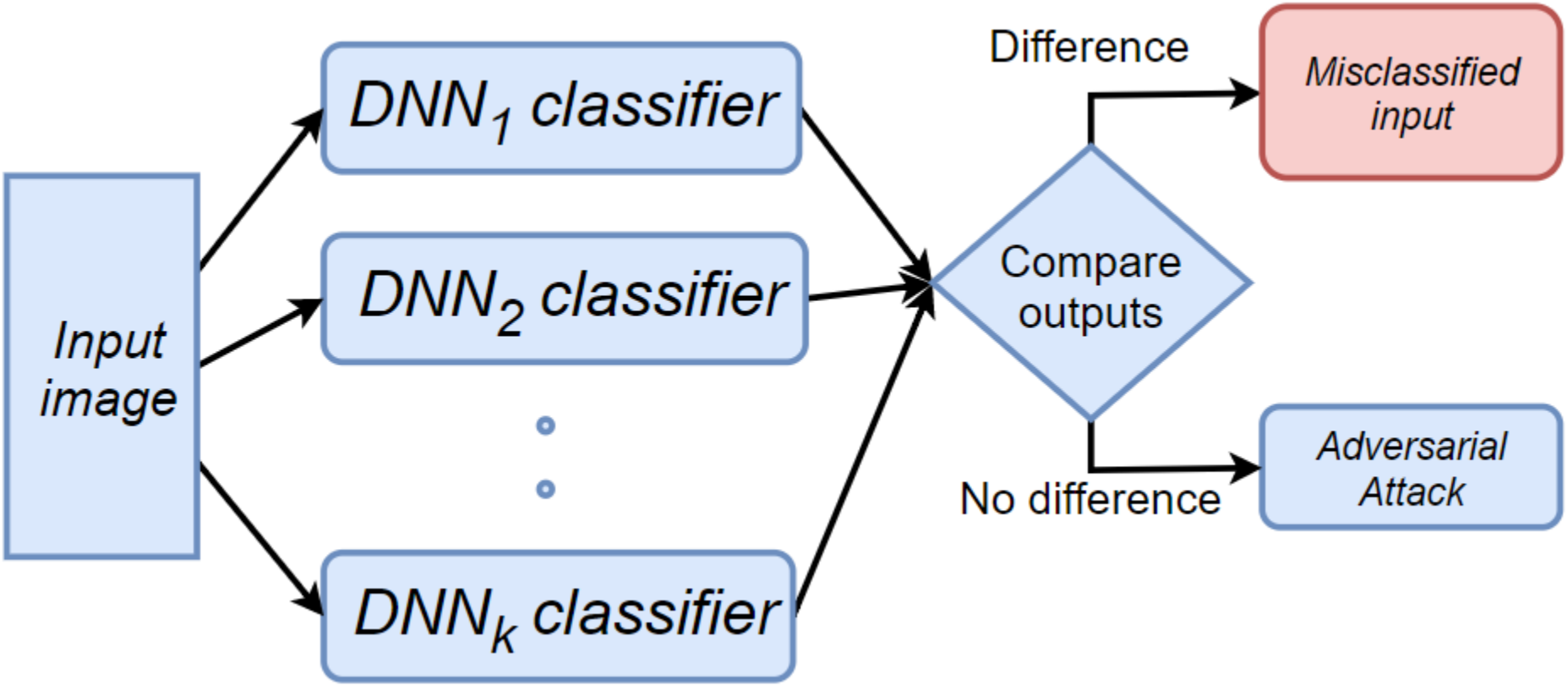
- Formal verification analyzes if the formal model satisfies the specification (properties)
- applied to AI



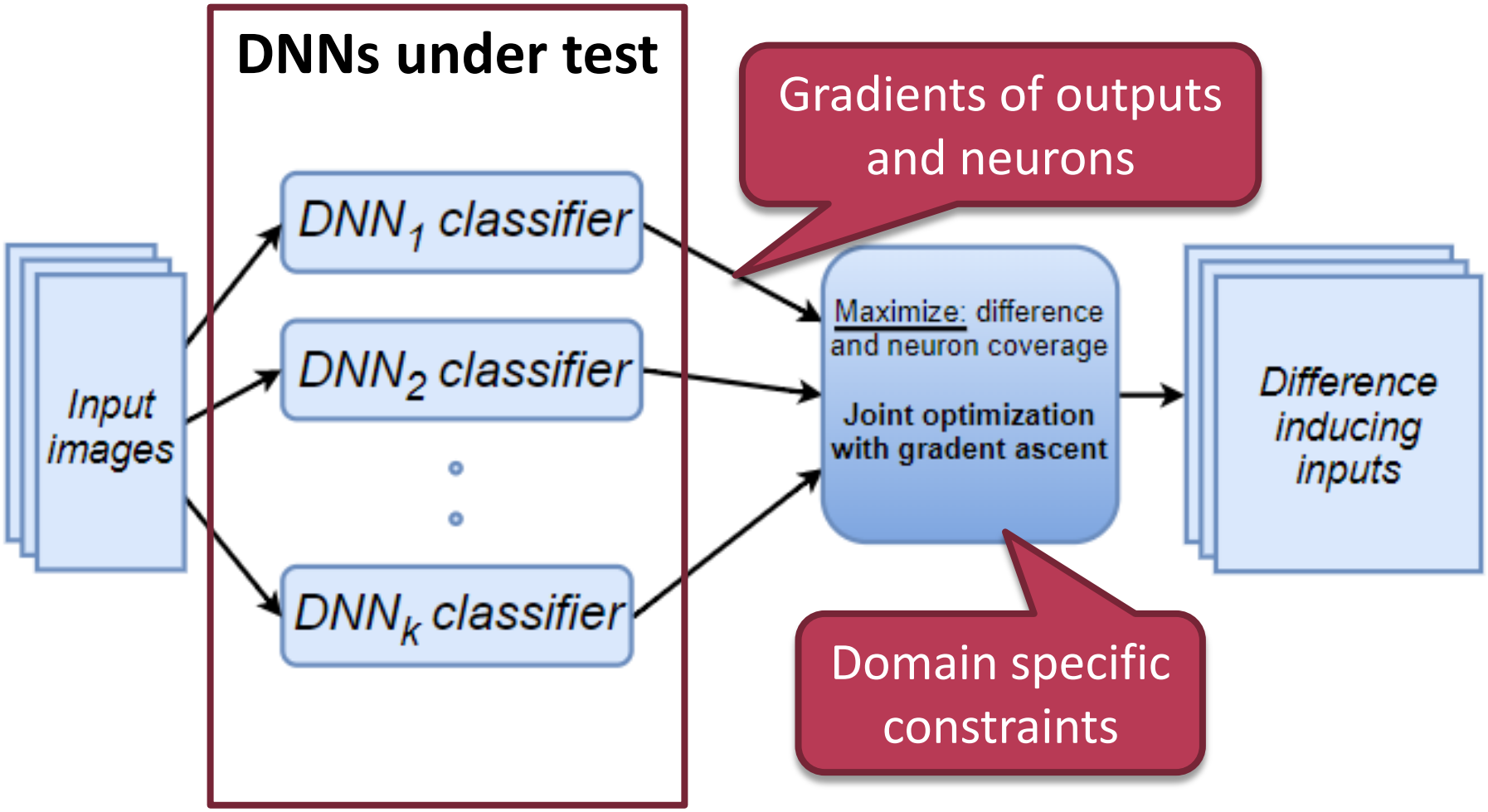
DeepXplore

- Differential testing approach
  - Running more versions of the same program (in our case: DNN)
- No difference found
  - Adversarial example generation
  - Image transformations on the input
- Two objectives
  - Modify the output of the target model, while keeping the original output of the other models
  - Increase the **neuron coverage** of the neural network

Searching for misclassified images

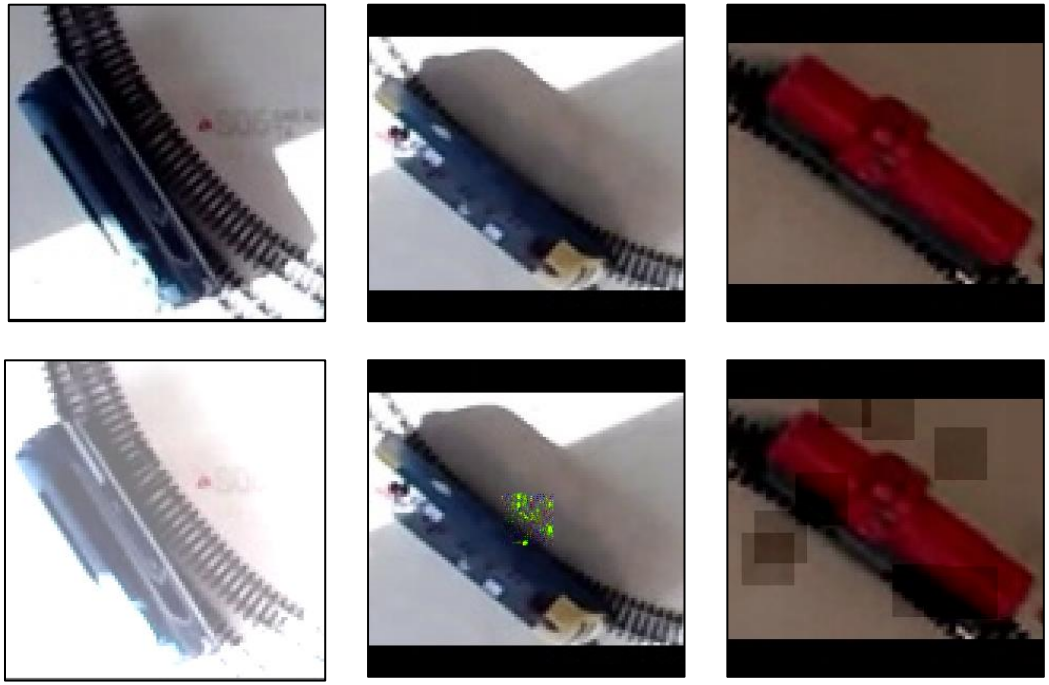


Adversarial attack (DeepXplore workflow)



Evaluation

- Retraining with the generated samples
- **Critical situation and counter-examples can be found**



Future Work?

- Active research area
  - Join us!
- Demonstrator development
  - MoDeS3 intelligent control
  - Industrial partners
- Project laboratory, Student scientific report (TDK)
- International collaboration

THANK YOU
FOR YOUR ATTENTION!

EMBERI ERŐFORRÁSOK
MINISZTÉRIUMA

DEMO