Forensics Through Stega Glasses: the Case of Adversarial Images

Benoit Bonnet
benoit.bonnet@inria.fr

Teddy Furon
teddy.furon@inria.fr

Patrick Bas
patrick.bas@centralelille.fr

Univ. Rennes, Inria, CNRS, IRISA

MMForWILD 2021

Introduction

- Image classification: most common task in Artificial Intelligence
- Lead by state-of-the art Deep Neural Networks (DNNs)

Introduction

- Image classification: most common task in Artificial Intelligence
- Lead by state-of-the art Deep Neural Networks (DNNs)

→ Sensitive to adversarial attacks !

What is an Adversarial Attack ?

- An Attack produces an Adversarial Sample
- Adversarial Sample = Original Image + Perturbation
- Perturbation:

  - Mostly imperceptible for a human

  - but enough to fool a classifier

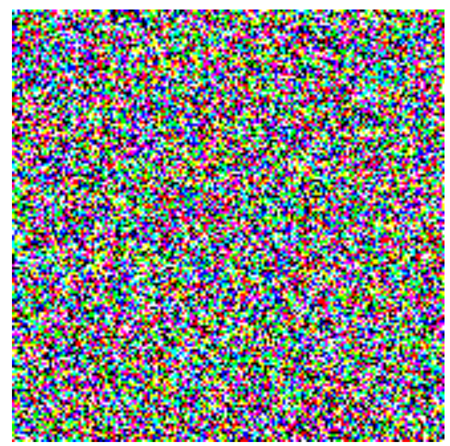
What is an Adversarial Attack ?

- An Attack produces an Adversarial Sample

Original image

"baseball"

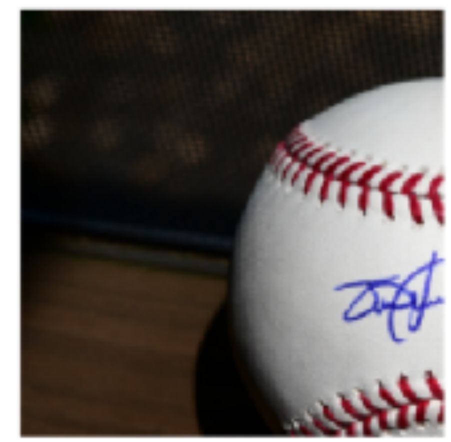
+



Perturbation

(crafted by the attack)

=



Adversarial Sample

"golf ball"

Attack Scenarios

- Several scenarios of attacks:

  - Targeted: Incorrect classification with specific label

1

  - **Untargeted**: Incorrect classification only

Attack Scenarios

- Several scenarios of attacks:

1

- Targeted: Incorrect classification with specific label

- **Untargeted**: Incorrect classification only

2

- Black-box: Attack only observes output of classifier

- **White-box**: Attack knows classifier and its parameters

White-box Attacks

- White-box setup:

White-box Attacks

- White-box setup:

    - Most popular attacks FGSM, IFGSM, PGD, DDN, C&W …

White-box Attacks

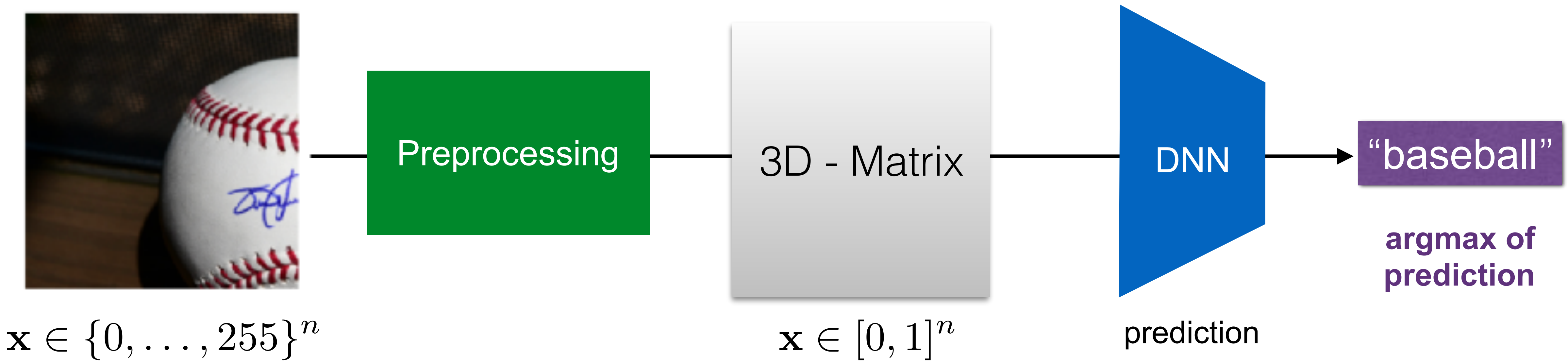
- White-box setup:

  - Most popular attacks FGSM, IFGSM, PGD, DDN, C&W …

  - Maximize success rate while minimizing Distortion

  - Core mechanism: gradient back propagation

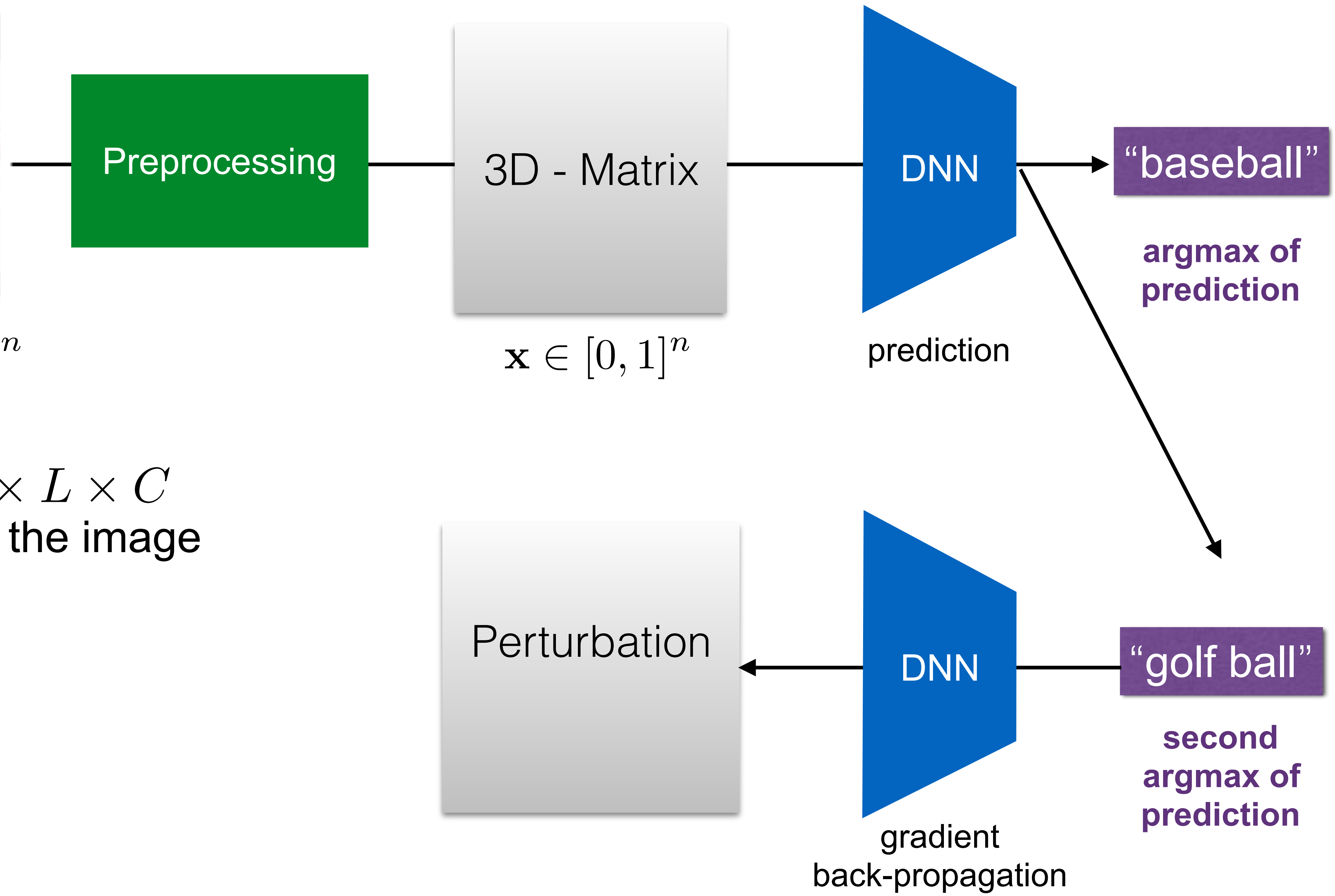
What Attacks usually do



With $n = 3 \times L \times C$
Dimension of the image

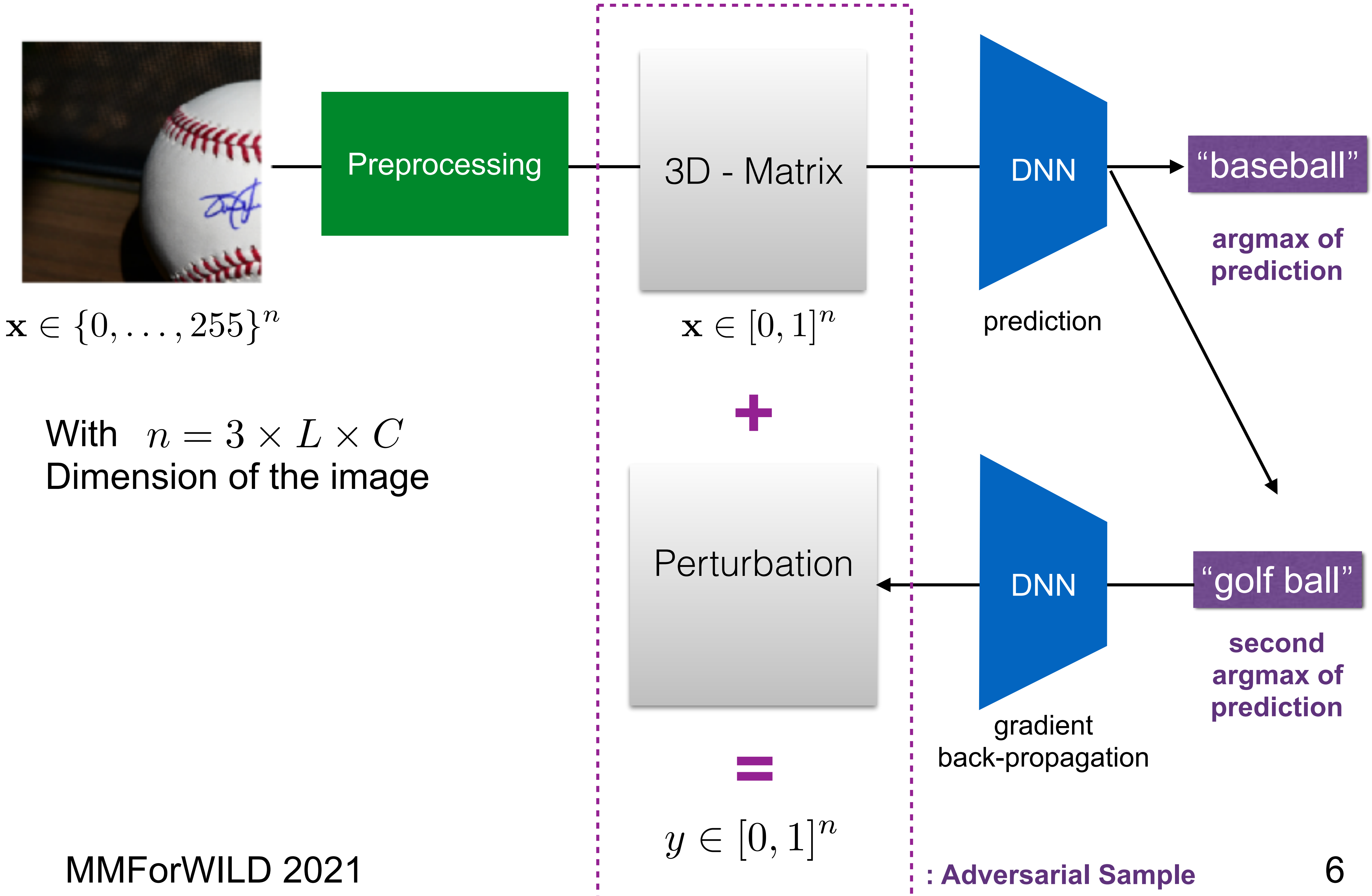
What Attacks usually do

$\mathbf{x} \in \{0, \ldots, 255\}^n$



With $n = 3 \times L \times C$
Dimension of the image

What Attacks usually do



What Attacks usually do

$\mathbf{x} \in \{0, \dots, 255\}^n$

Preprocessing

3D - Matrix

$\mathbf{x} \in [0, 1]^n$

+

Perturbation

=

$y \in [0, 1]^n$

: Adversarial Sample

DNN

prediction

"baseball"

argmax of prediction

DNN

gradient back-propagation

"golf ball"

second argmax of prediction

With $n = 3 \times L \times C$
Dimension of the image

Ultimate goal: Solve

$$\min_{y \in [0,1]^n : c(y) \neq c_0} ||y - x||$$

A missing Constraint ?

- A digital image is in the **discrete** RGB domain
- But attack is performed in the preprocessed domain

→ The sample is in the **continuous** domain

A missing Constraint ?

- A digital image is in the **discrete** RGB domain

- But attack is performed in the preprocessed domain

→ The sample is in the **continuous** domain

- This issue is adressed in previous work *"What if Adversarial Samples were Digital Images?" (IH&MMSec 2020)*

What if Adversarial Samples were Digital Images?

- Rounding is ineffective (erases most of the attack)

➡ Introduces a post-processing after any attack to effectively quantize the perturbation:

What if Adversarial Samples were Digital Images?

- Rounding is ineffective (erases most of the attack)

Introduces a post-processing after any attack to effectively quantize the perturbation:

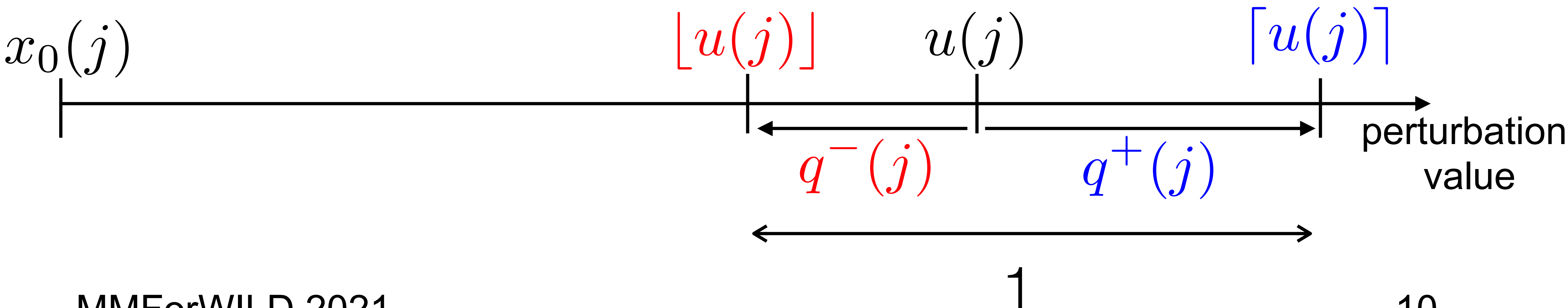
- **fast** (post-processing << attack)
- **effective** (sample remains adversarial)
- **optimized** (minimizes L2 distortion)

Overview of the Problem

- Notations:
  - x_0: original image
  - $x_a = x_0 + u$: unquantized adversarial sample
  - u : unquantized perturbation
  - $x_q = x_0 + u + q$: quantized adversarial sample
  - q : the quantization noise vector s.t. $u + q$ is an integer vector

Overview of the Problem

- Objective: find q
- for any pixel j, we consider 2 cases:
  - $q^+(j) = \lceil u(j) \rceil - u(j)$ s.t. $q^+(j) \geq 0$
  - $q^-(j) = \lfloor u(j) \rfloor - u(j)$ s.t. $q^-(j) \leq 0$



Classifier Loss

- Loss used to handle classification: $L_Q(q) = p_t(x_q) - p_k(x_q)$
  - p_t = probability output for class t which is the class of the **original image** $argmax f(x_0) = t$
  - p_k = probability output for class k which is the class of the **adversarial sample** $argmax f(x_a) = k$

Classifier Loss

- Loss used to handle classification: $L_Q(q) = p_t(x_q) - p_k(x_q)$
  - p_t = probability output for class t which is the class of the **original image** $argmax f(x_0) = t$
  - p_k = probability output for class k which is the class of the **adversarial sample** $argmax f(x_a) = k$

$\longrightarrow$ $L_Q(q) < 0 \Leftrightarrow$ adversariality

A Lagrangian Formulation

- Goal: find a tradeoff between distortion and adversariality
- We look for q^* s.t.:

$$q^* = \arg\min_q (D(q) + \lambda \times L_Q(q))$$

A Lagrangian Formulation

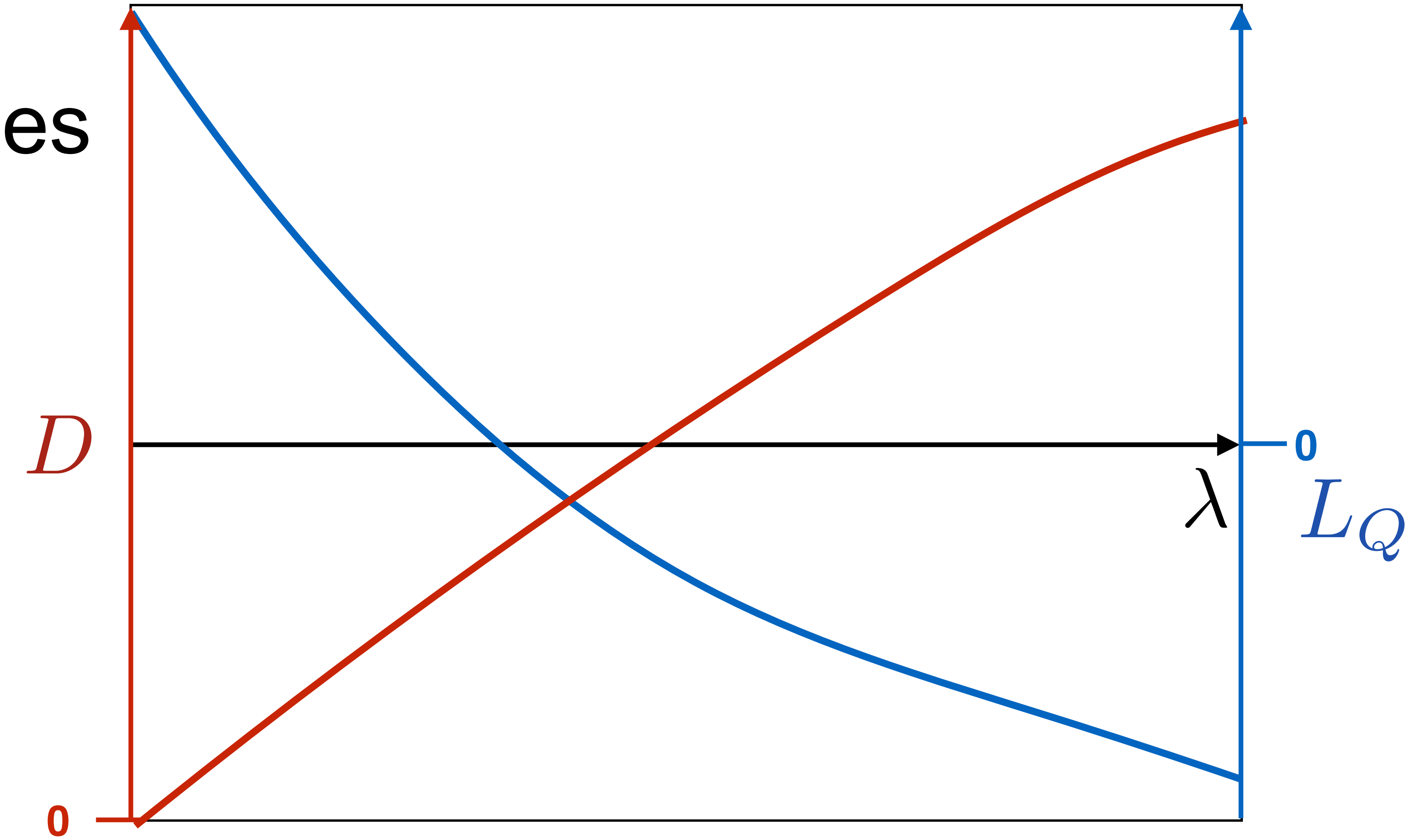
- Goal: find a tradeoff between distortion and adversariality
- We look for q^* s.t.:

$$q^* = \arg\min_q \big(\overbrace{D(q)}^{\boxed{\text{Optimization criterion}}} + \lambda \times \overbrace{L_Q(q)}^{\boxed{\text{Success criterion}}}\big)$$

- $D(q) = \|x_o - x_q\|^2$ = distortion after quantization
- $L_Q(q) = p_t(x_q) - p_k(x_q)$ = classification loss
- λ = Lagrangian multiplier

Behavior along λ

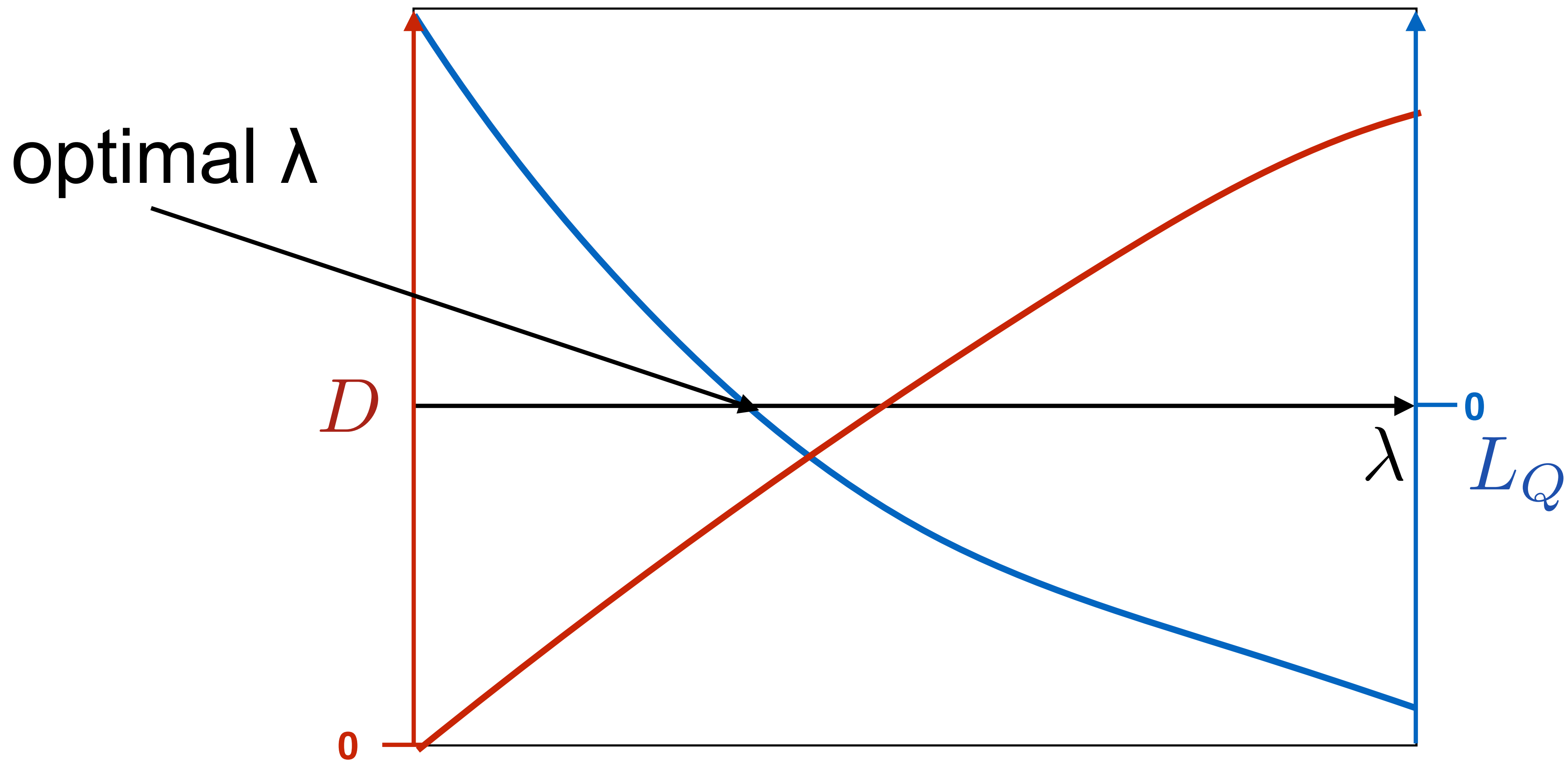
- As λ increases:
  - Distortion D increases
  - Loss L_Q decreases



Finding the optimal λ

- Optimal λ = smallest λ s.t. $L_Q < 0$

Keep adversariality while minimizing distortion



Finding the optimal λ

- Optimal λ = smallest λ s.t. $L_Q < 0$
- We compute and sort all values of λ that make a $q(j)$ swap from $q^+(j)$ to $q^-(j)$ and vice-versa
- Optimal λ found in less than $\log_2(n)$ steps of binary search

Detecting adversarial samples

- Adversarial samples are imperceptible for a human **but they are still statistically detectable !**
- Steganalysis: detection of hidden messages through statistical anomalies
- Steganalysis detectors: **SRM, SCRMQ1** (+ Linear classifier) and **SRNet** (DNN binary classifier)

Detectors

- **SRM**: **S**patial **R**ich **M**odel: feature vector of dimension 34,671. Only one channel→ used on luminance of the sample
- **SCRMQ1**: Color version of **SRM**: feature vector of dimension 18,157. On all 3 channels
- **SRNet**: DNN trained over 180 epochs
- Detectors trained on 15,651 pairs of images (original + adversarial sample crafted with *best-effort* FGSM)

Detection Results

- True Positive Rate over 1000 test images for False Positive Rate = 5%
- 4 attacks (FGSM, PGD and C&W quantized with post-processing, DDN natively quantized)

	P_{suc}	$\overline{L_2}$	SRM(%)	SCRMQ1(%)	SRNet(%)
FGSM+[4]	89.7	286	72.00	83.3	**93.5**
PGD_2+[4]	98.6	113	65.02	83.1	**93.8**
CW+[4]	89.7	97	68.78	83.6	**94.5**
DDN	83.2	186	79.53	91.9	**94.8**

Average L2 distortion

Detection Results

- Adversarial samples optimized with L2 Distortion are **highly detectable**
- Even if trained of a basic FGSM attack, detectors generalize well to finer attacks

Detection Results

- Adversarial samples optimized with L2 Distortion are **highly detectable**
- Even if trained of a basic FGSM attack, detectors generalize well to finer attacks
- **Idea**: We can use steganographic embedding strategies to quantize our image

Steganographic Cost

- To each pixel i is associated a weight $w(l)$ reflecting the detectability of modifying i by a quantum l
- usually $w(l) = w(-l)$

  $w(0) = 0$

  $|l_1| > |l_2| \rightarrow w(|l_1|) > w(|l_2|)$
- The total steganographic cost is $\sum_{i=1}^{n} w_i(l_i)$

Costs and quantization

- Distortion is replaced by stega cost in the lagrangian formulation

Costs and quantization

- Distortion is replaced by stega cost in the lagrangian formulation
- Costs: HILL computed using two low-pass filters
    - naive and simple
    - but only for a modification of ± 1

Costs and quantization

- Distortion is replaced by stega cost in the lagrangian formulation
- Costs: HILL computed using two low-pass filters

  MiPod computed through estimated variance with Wiener filtering

  - more complex
  - handles modifications others than ± 1

Costs and quantization

- Distortion is replaced by stega cost in the lagrangian formulation

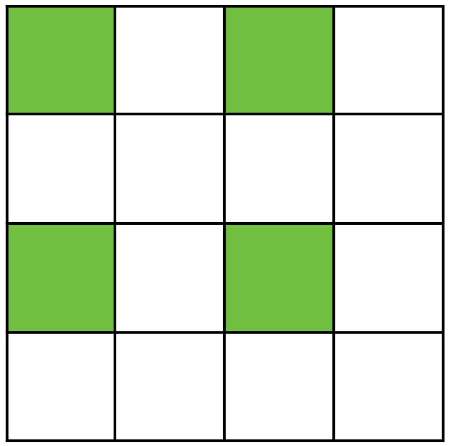
- Costs: HILL computed using two low-pass filters

  MiPod computed through estimated variance with Wiener filtering

- GINA : quantization strategy using MiPod costs

GINA strategy

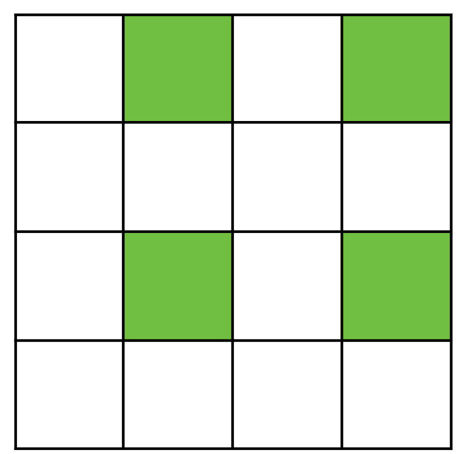
- The image is divided in 12 lattices (4 per color channel)



First channel (Green) First lattice

GINA strategy

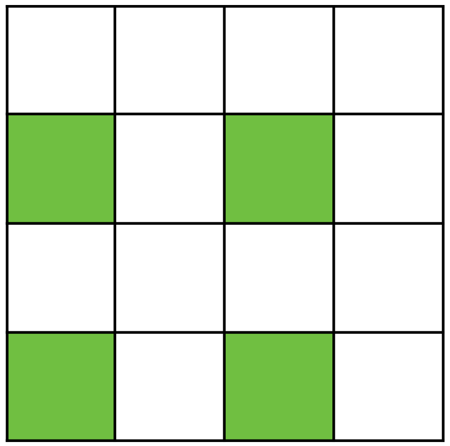
- The image is divided in 12 lattices (4 per color channel)



First channel (Green) Second lattice

GINA strategy

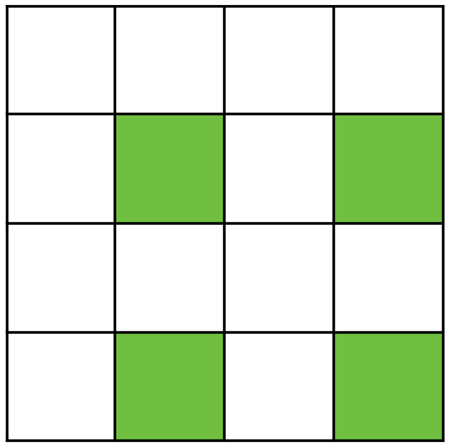
- The image is divided in 12 lattices (4 per color channel)



First channel (Green) Third lattice

GINA strategy

- The image is divided in 12 lattices (4 per color channel)



First channel (Green) Fourth lattice

GINA strategy

- The image is divided in 12 lattices (4 per color channel)
- Each lattice is quantized so it contributes to 1/12 of the initial L_Q (at $\lambda = 0$)

GINA strategy

- The image is divided in 12 lattices (4 per color channel)
- Each lattice is quantized so it contributes to 1/12 of the initial L_Q (at $\lambda = 0$)
- After each lattice is quantized, costs are recomputed and updated with **CMD**[1] strategy favoring same modifications in a neighbourhood

1: A strategy of clustering modification directions in spatial image steganography, Li et al. 2015

Detection results (bis)

	d	P_{suc} (%)		$\overline{L_2}$		SCRMQ1(%)		SRNet(%)	
		Van	Rob	Van	Rob	Van	Rob	Van	Rob
[30]	2	98.6	98.3	**101**	**167**	83.1	84.6	93.8	90.1
HILL	2	98.6	98.3	113	177	78.0	76.6	87.6	88.5
HILL	4	**98.9**	**98.5**	125	181	76.0	73.3	87.4	88.2
MiPod	2	98.3	98.3	176	242	77.4	76.2	86.6	87.7
MiPod	4	98.7	98.0	164	247	74.4	70.2	84.5	87.7
GINA	2	98.5	98.1	283	337	24.4	32.4	68.3	**82.9**
GINA	4	98.8	98.2	300	330	**18.6**	**24.3**	**50.9**	85.2

Detection results (bis)

degree of liberty: maximum distortion = $\pm\frac{d}{2}$

	d	P_{suc} (%)		$\overline{L_2}$		SCRMQ1(%)		SRNet(%)	
		Van	Rob	Van	Rob	Van	Rob	Van	Rob
[30]	2	98.6	98.3	**101**	**167**	83.1	84.6	93.8	90.1
HILL	2	98.6	98.3	113	177	78.0	76.6	87.6	88.5
HILL	4	**98.9**	**98.5**	125	181	76.0	73.3	87.4	88.2
MiPod	2	98.3	98.3	176	242	77.4	76.2	86.6	87.7
MiPod	4	98.7	98.0	164	247	74.4	70.2	84.5	87.7
GINA	2	98.5	98.1	283	337	24.4	32.4	68.3	**82.9**
GINA	4	98.8	98.2	300	330	**18.6**	**24.3**	**50.9**	85.2

Van = EfficientNet-b0 (vanilla)
Rob = EfficientNet-b0 with adversarial training (robust)

Detection results (bis)

	d	P_{suc} (%)		$\overline{L_2}$		SCRMQ1(%)		SRNet(%)	
		Van	Rob	Van	Rob	Van	Rob	Van	Rob
[30]	2	98.6	98.3	**101**	**167**	83.1	84.6	93.8	90.1
HILL	2	98.6	98.3	113	177	78.0	76.6	87.6	88.5
HILL	4	**98.9**	**98.5**	125	181	76.0	73.3	87.4	88.2
MiPod	2	98.3	98.3	176	242	77.4	76.2	86.6	87.7
MiPod	4	98.7	98.0	164	247	74.4	70.2	84.5	87.7
GINA	2	98.5	98.1	283	337	24.4	32.4	68.3	**82.9**
GINA	4	98.8	98.2	300	330	**18.6**	**24.3**	**50.9**	85.2

GINA is significantly harder to detect

Detection results (bis)

	d	P_{suc} (%)		$\overline{L_2}$		SCRMQ1(%)		SRNet(%)	
		Van	Rob	Van	Rob	Van	Rob	Van	Rob
[30]	2	98.6	98.3	**101**	**167**	83.1	84.6	93.8	90.1
HILL	2	98.6	98.3	113	177	78.0	76.6	87.6	88.5
HILL	4	**98.9**	**98.5**	125	181	76.0	73.3	87.4	88.2
MiPod	2	98.3	98.3	176	242	77.4	76.2	86.6	87.7
MiPod	4	98.7	98.0	164	247	74.4	70.2	84.5	87.7
GINA	2	98.5	98.1	283	337	24.4	32.4	68.3	**82.9**
GINA	4	98.8	98.2	300	330	**18.6**	**24.3**	**50.9**	85.2

GINA is significantly harder to detect

Rob is significantly harder to fool

Detection results (bis)

	d	P_{suc} (%)		$\overline{L_2}$		SCRMQ1(%)		SRNet(%)	
		Van	Rob	Van	Rob	Van	Rob	Van	Rob
[30]	2	98.6	98.3	**101**	**167**	83.1	84.6	93.8	90.1
HILL	2	98.6	98.3	113	177	78.0	76.6	87.6	88.5
HILL	4	**98.9**	**98.5**	125	181	76.0	73.3	87.4	88.2
MiPod	2	98.3	98.3	176	242	77.4	76.2	86.6	87.7
MiPod	4	98.7	98.0	164	247	74.4	70.2	84.5	87.7
GINA	2	98.5	98.1	283	337	24.4	32.4	68.3	**82.9**
GINA	4	98.8	98.2	300	330	**18.6**	**24.3**	**50.9**	85.2

GINA is significantly harder to detect

Rob is significantly harder to fool

SRNet outperforms SCRMQ1

Conclusion

- We explored detectors from **steganalysis** to detect adversarial samples with succes
- **SRNet** is in most cases the best detector

Conclusion

- We explored detectors from **steganalysis** to detect adversarial samples with succes
- **SRNet** is in most cases the best detector
- We explored strategies for less detectable adversarial samples through **quantization**
- **GINA** offers less detectability at the cost of a lot more distortion
- However scanning through the 1000 test images, none had visible artifacts