PatchGuard: A Provably Robust Defense against Adversarial Patches via Small Receptive Fields and Masking

Chong Xiang[†], Arjun Nitin Bhagoji[‡], Vikash Sehwag[†], Prateek Mittal[†]

[†]Princeton University [‡]University of Chicago

USENIX Security Symposium 2021

PatchGuard: A Provably Robust Defense against Adversarial Patches via Small Receptive Fields and Masking

Chong Xiang[†], Arjun Nitin Bhagoji[‡], Vikash Sehwag[†], Prateek Mittal[†]

[†]Princeton University [‡]University of Chicago

USENIX Security Symposium 2021

PatchGuard: A Provably Robust Defense against Adversarial Patches via Small Receptive Fields and Masking

Chong Xiang†, Arjun Nitin Bhagoji‡, Vikash Sehwag†, Prateek Mittal†

†Princeton University ‡University of Chicago

USENIX Security Symposium 2021

Adversarial Example Attacks: Small Perturbations for Test-Time Model Misclassification

Normal Example $(\boldsymbol{x})$
Dog (y)

$+$

Add imperceptible perturbations δ

$\Rightarrow$

Adversarial Example $(\boldsymbol{x} + \boldsymbol{\delta})$
Cat (y')

$$\max_{\delta} L(M(x + \delta), y)$$

$L(\cdot)$- Loss function; $M(\cdot)$ - Model

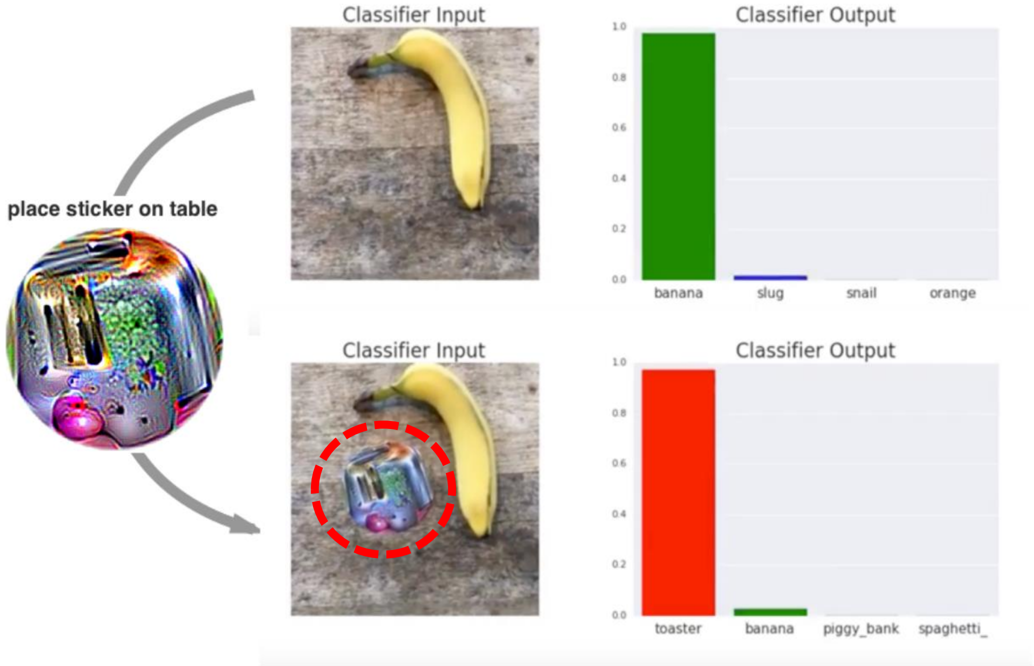
A threat to ML models!

Challenge: Requires global perturbations

Our Focus: Localized Adversarial Patch Attacks

1. All perturbations within *one* local region (patch)
2. Patch pixels can take arbitrary values
3. Realizable in the physical world – print and attach the patch!
   - **A REAL-WORLD threat**

LaVAN Attack [2]



Adversarial Patch [1]

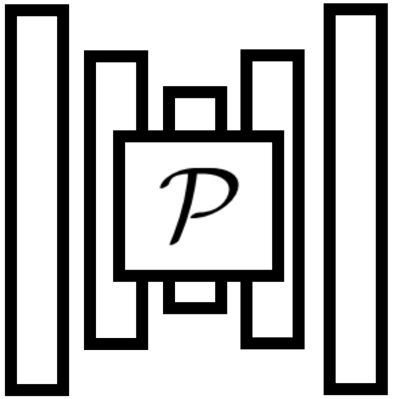
4. Patch can be *anywhere* on the image
5. Patch size should be reasonable (shouldn't block the entire salient object)

[1] Brown et al., "Adversarial Patch," NeurIPS Workshops 2017
[2] Karmon et al., "LaVAN: Localized and Visible Adversarial Noise," ICML 2018.

Defense Objective: Provable Robustness on Certified Test Images

Test Image

Threat Model
(patch sizes, shapes, and location set)



Provable Analysis

Ground-truth y

The prediction is **always** correct; Image certified!

"I think it is a $\boxed{\text{dog}}$"
"My prediction will never change"

or

"I can't say anything for sure"

Provable robust accuracy / certified accuracy: the fraction of test images that are

1. Correctly classified
2. Provably robust to any (adaptive) localized patch attack within the threat model

Our Contribution: PatchGuard Defense Framework with Provable Robustness

PatchGuard aims to prevent the localized patch from dominating the global prediction

Tiger Cat (94.4%)

PatchGuard: A Provably Robust Defense Framework

Small Receptive Field

Bound the number of corrupted features

Secure Feature Aggregation

Do robust prediction on partially corrupted features

Our Contribution: PatchGuard Defense Framework with Provable Robustness

PatchGuard aims to prevent the localized patch from dominating the global prediction

Tiger Cat (94.4%)

PatchGuard: A Provably Robust Defense Framework

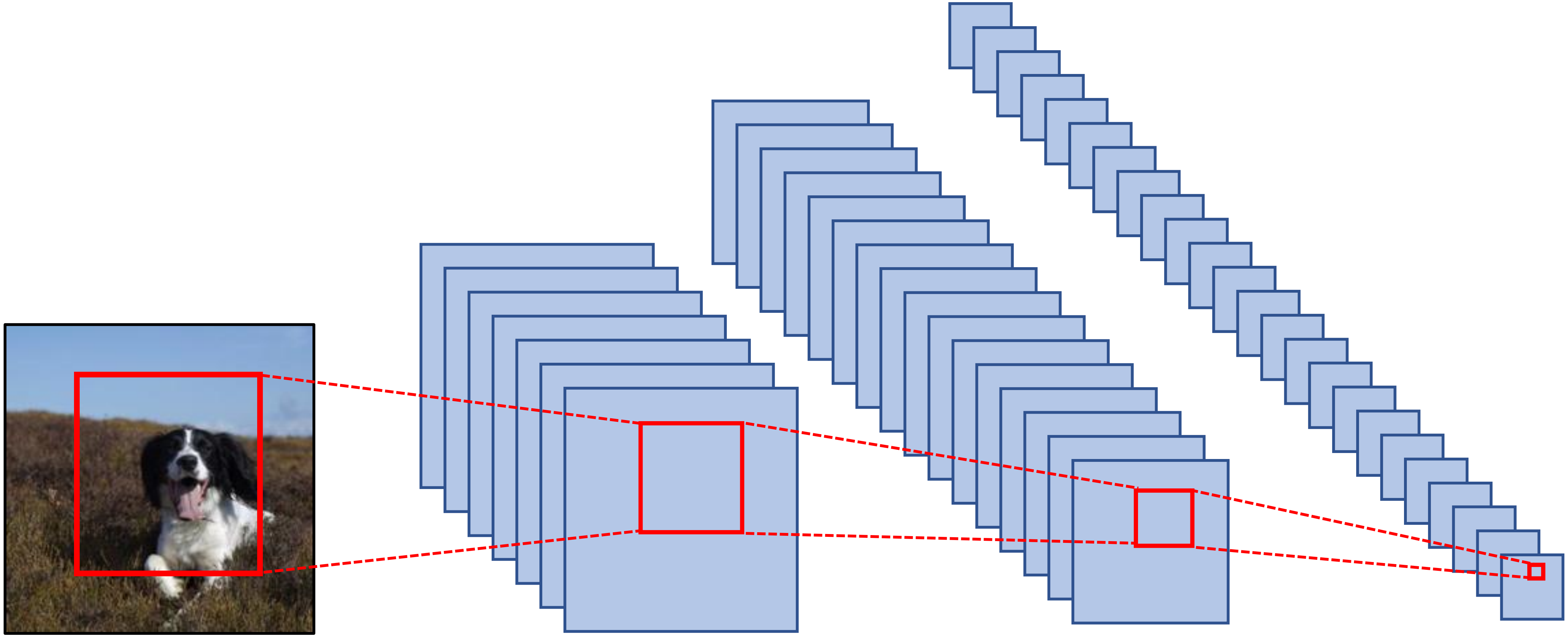
Small Receptive Field

Bound the number of corrupted features

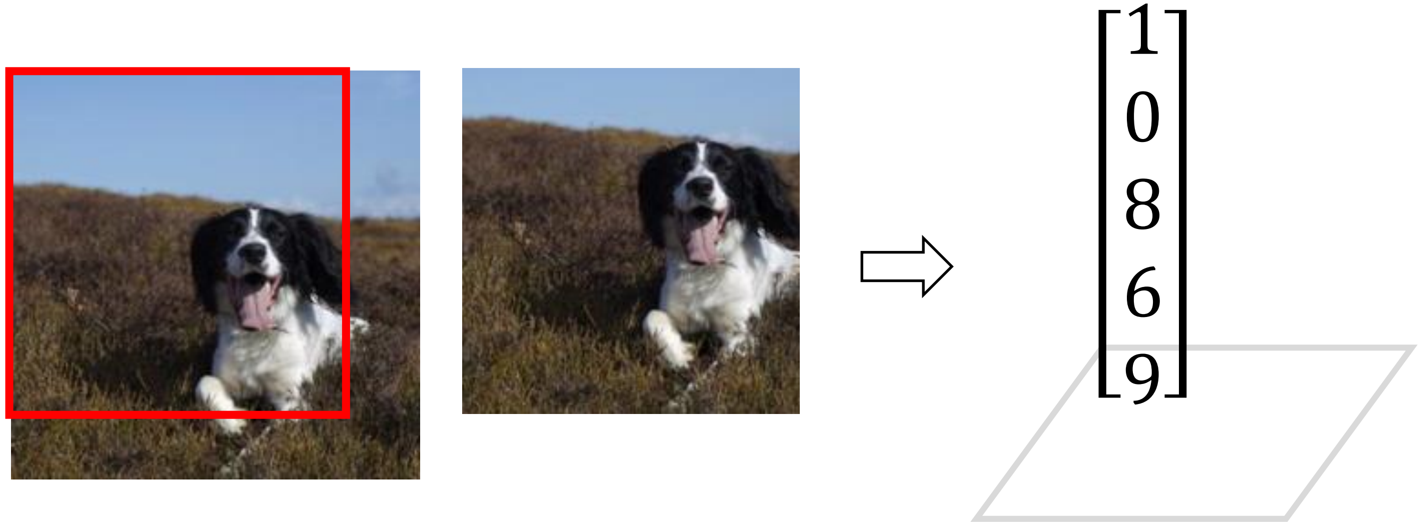
Secure Feature Aggregation

Do robust prediction on partially corrupted features

Receptive Field: a Region of the Input Image that an Extracted Feature is Looking at

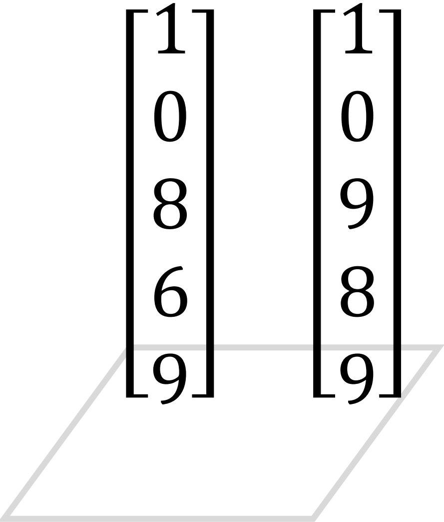


Receptive Field: a Region of the Input Image that an Extracted Feature is Looking at



Local feature map

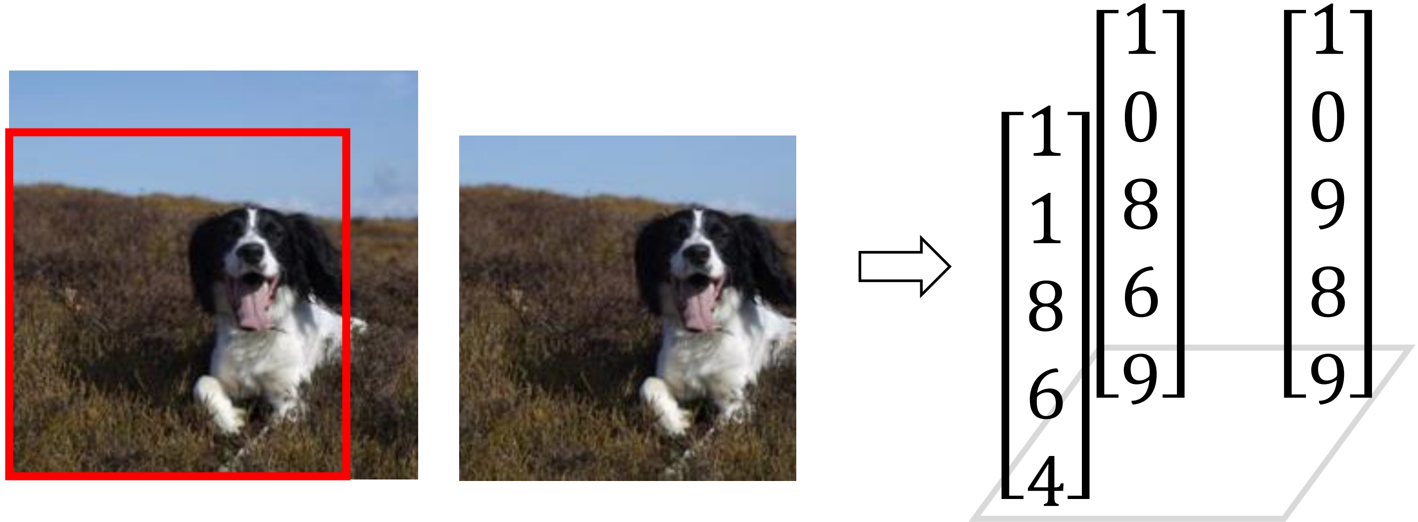
Receptive Field: a Region of the Input Image that an Extracted Feature is Looking at



$$\begin{bmatrix}1\\0\\8\\6\\9\end{bmatrix} \quad \begin{bmatrix}1\\0\\9\\8\\9\end{bmatrix}$$

Local feature map

Receptive Field: a Region of the Input Image that an Extracted Feature is Looking at



Local feature map

Receptive Field: a Region of the Input Image that an Extracted Feature is Looking at

$$\begin{bmatrix}1\\1\\8\\6\\4\end{bmatrix}\begin{bmatrix}1\\0\\8\\6\\9\end{bmatrix}\begin{bmatrix}0\\0\\9\\9\\9\end{bmatrix}\begin{bmatrix}1\\0\\9\\8\\9\end{bmatrix}$$

Local feature map

Aggregate Local Features for Global Prediction

$$\Rightarrow \begin{bmatrix}1\\1\\8\\6\\4\end{bmatrix} \begin{bmatrix}1\\0\\8\\6\\9\end{bmatrix} \begin{bmatrix}0\\0\\9\\9\\9\end{bmatrix} \begin{bmatrix}1\\0\\9\\8\\9\end{bmatrix} \Rightarrow \begin{bmatrix}3\\1\\34\\29\\31\end{bmatrix} \Rightarrow \begin{array}{c|l} \begin{bmatrix}4\\34\end{bmatrix} & \begin{array}{l}\text{cat}\\ \textbf{Dog!}\end{array}\end{array}$$

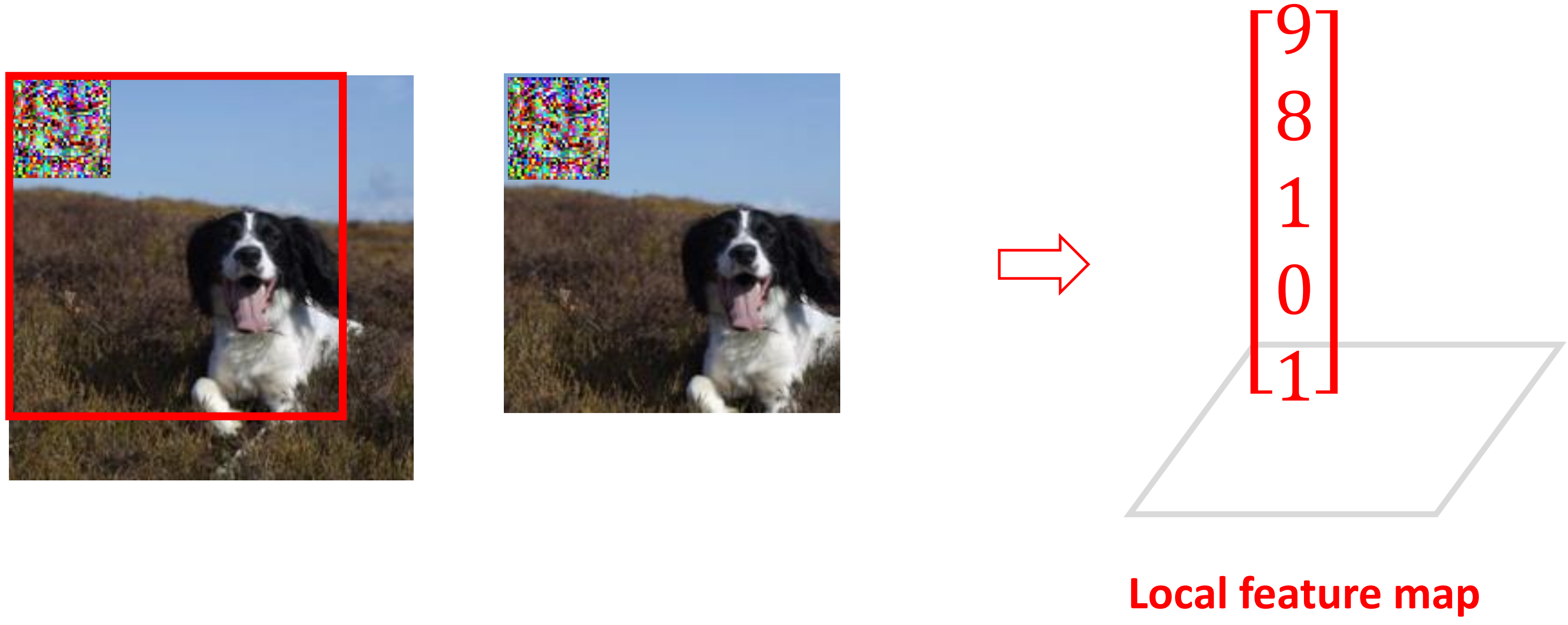
Local feature map

Global feature

Global prediction / logits

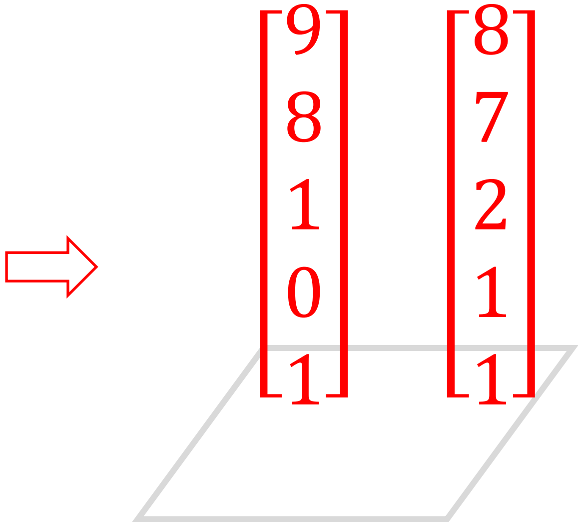
Key insight: the Receptive Field Size Determines the Number of Features Corrupted by the Adversarial Patch

Example 1: CNN with *large* receptive fields (e.g., ResNet with 483×483 px)



Key insight: the Receptive Field Size Determines the Number of Features Corrupted by the Adversarial Patch

Example 1: CNN with *large* receptive fields (e.g., ResNet with 483×483 px)



Local feature map

Key insight: the Receptive Field Size Determines the Number of Features Corrupted by the Adversarial Patch

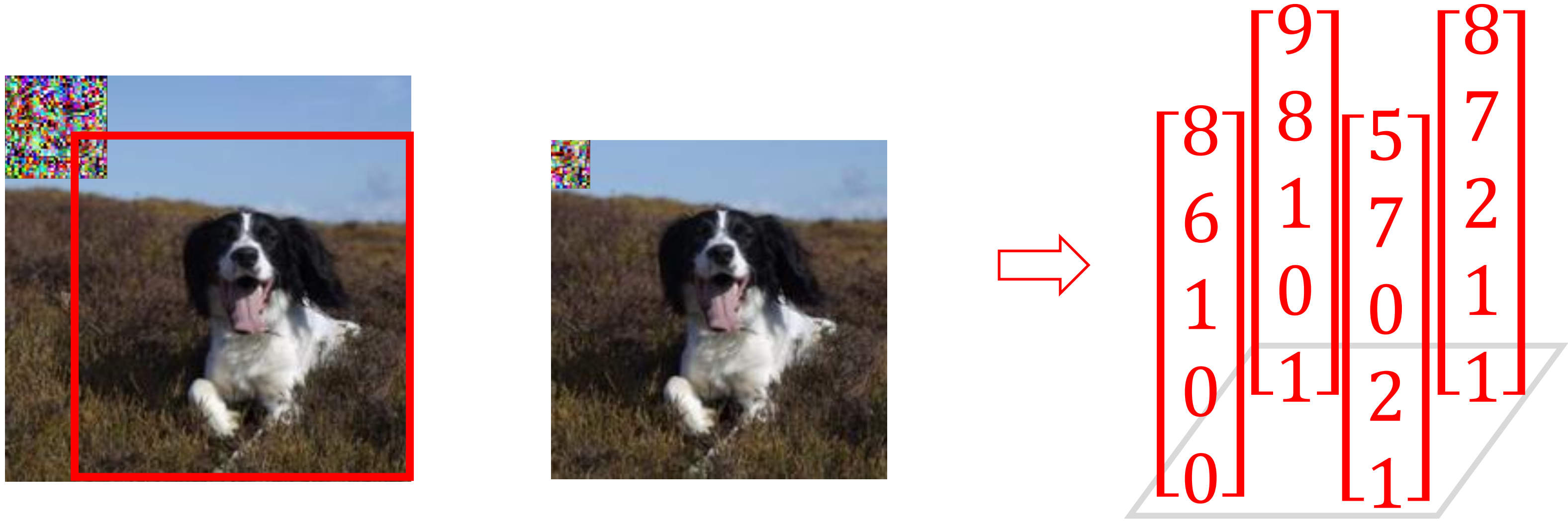
Example 1: CNN with *large* receptive fields (e.g., ResNet with 483×483 px)

$$\Rightarrow \begin{bmatrix}8\\6\\1\\0\\0\end{bmatrix} \begin{bmatrix}9\\8\\1\\0\\1\end{bmatrix} \begin{bmatrix}8\\7\\2\\1\\1\end{bmatrix}$$

Local feature map

Key insight: the Receptive Field Size Determines the Number of Features Corrupted by the Adversarial Patch

Example 1: CNN with *large* receptive fields (e.g., ResNet with 483×483 px)

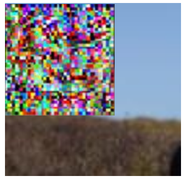


Local feature map

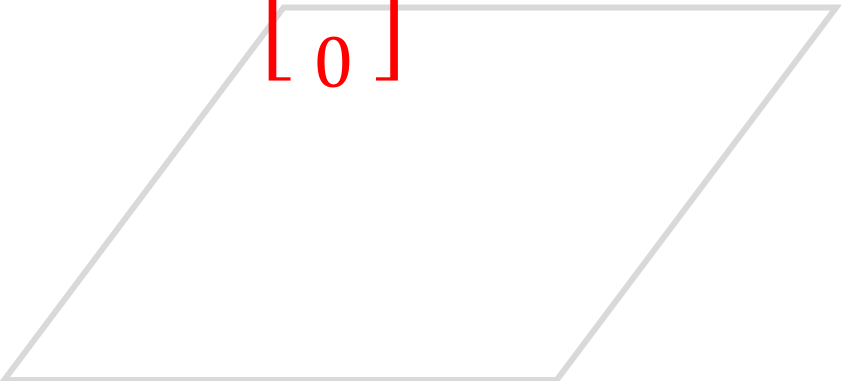
Note: *all* feature corrupted!
Little hope for us to do a robust prediction

Key insight: the Receptive Field Size Determines the Number of Features Corrupted by the Adversarial Patch

Example 2: CNN with *small* receptive fields (e.g., BagNet with 17 × 17 px)



$$\Rightarrow \begin{bmatrix} 20 \\ 18 \\ 10 \\ 1 \\ 0 \end{bmatrix}$$



Key insight: the Receptive Field Size Determines the Number of Features Corrupted by the Adversarial Patch

Example 2: CNN with *small* receptive fields (e.g., BagNet with 17×17 px)

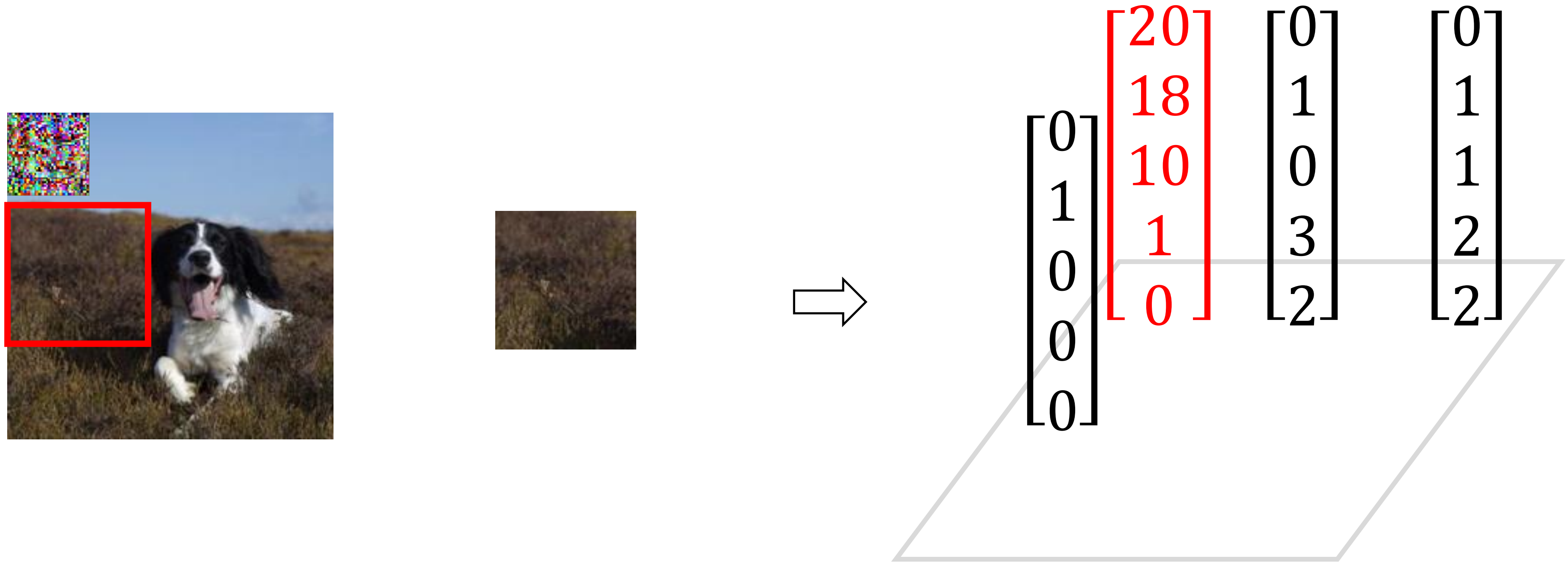
Key insight: the Receptive Field Size Determines the Number of Features Corrupted by the Adversarial Patch

Example 2: CNN with *small* receptive fields (e.g., BagNet with 17×17 px)

$$\color{red}{\begin{bmatrix} 20 \\ 18 \\ 10 \\ 1 \\ 0 \end{bmatrix}} \quad \begin{bmatrix} 0 \\ 1 \\ 0 \\ 3 \\ 2 \end{bmatrix} \quad \begin{bmatrix} 0 \\ 1 \\ 1 \\ 2 \\ 2 \end{bmatrix}$$

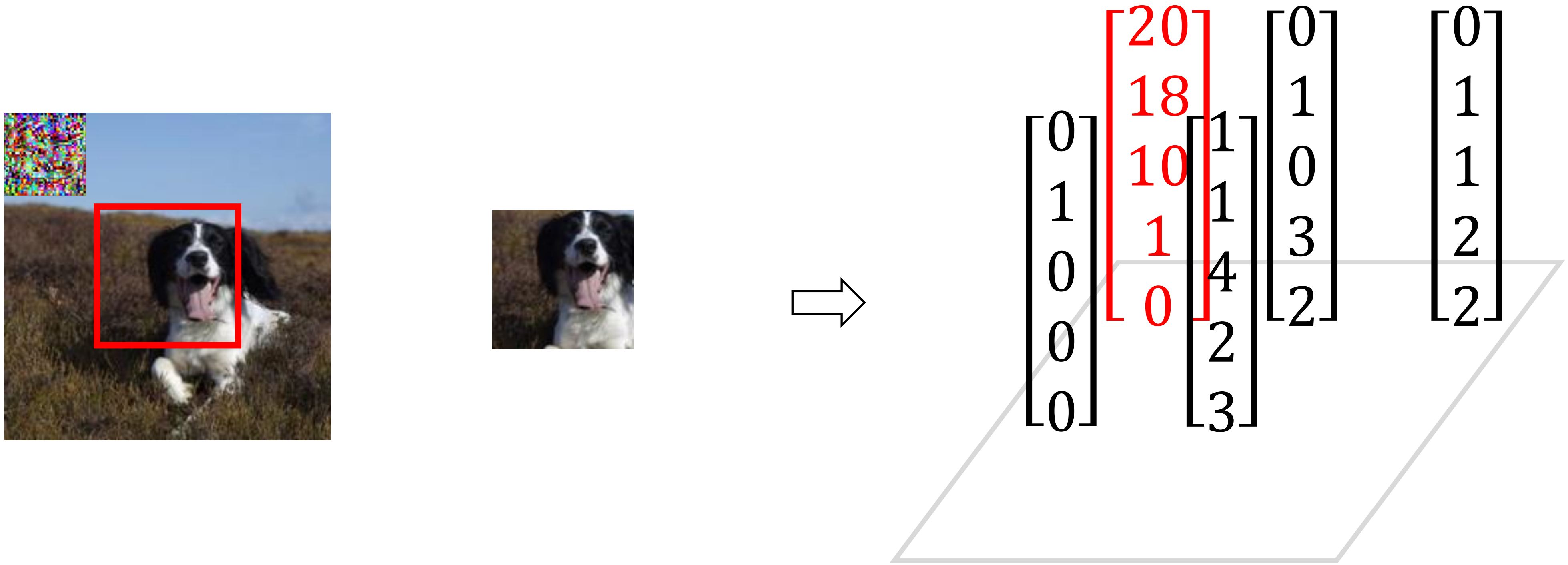
Key insight: the Receptive Field Size Determines the Number of Features Corrupted by the Adversarial Patch

Example 2: CNN with *small* receptive fields (e.g., BagNet with 17 × 17 px)



Key insight: the Receptive Field Size Determines the Number of Features Corrupted by the Adversarial Patch

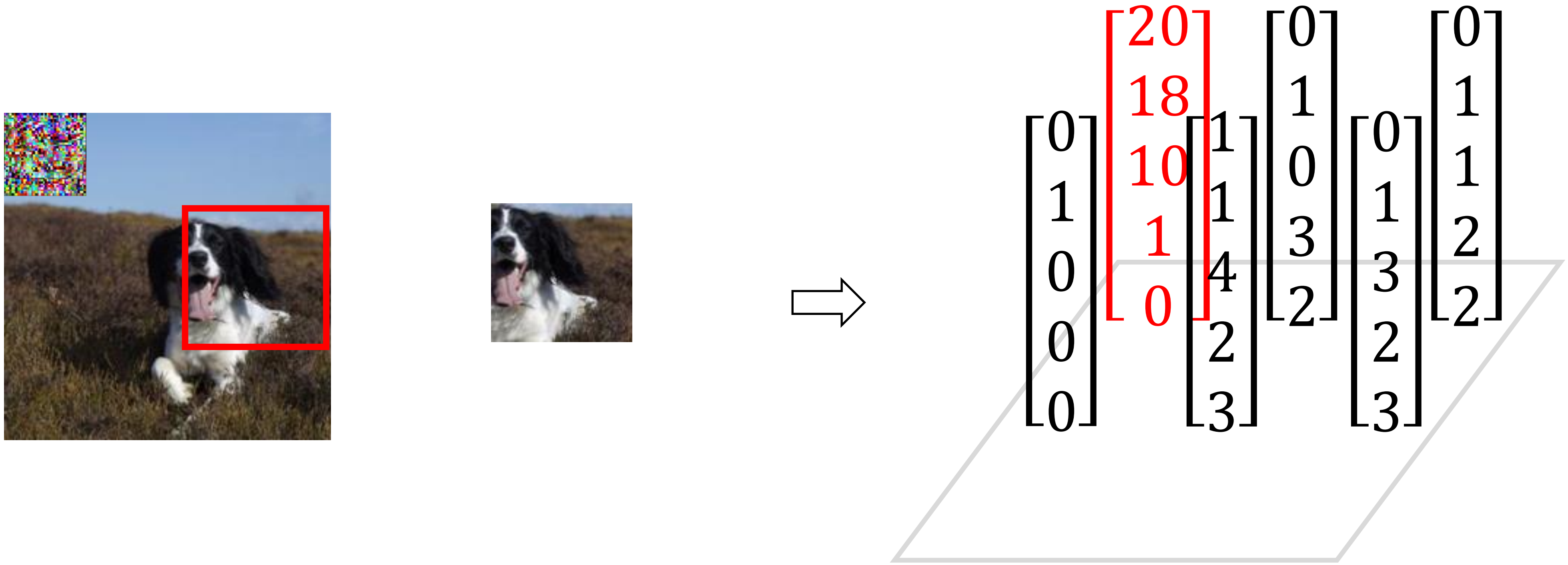
Example 2: CNN with *small* receptive fields (e.g., BagNet with 17 × 17 px)



$$\begin{bmatrix}0\\1\\0\\0\\0\end{bmatrix} \quad \begin{bmatrix}20\\18\\10\\1\\0\end{bmatrix} \quad \begin{bmatrix}1\\1\\4\\2\\3\end{bmatrix} \quad \begin{bmatrix}0\\1\\0\\3\\2\end{bmatrix} \quad \begin{bmatrix}0\\1\\1\\2\\2\end{bmatrix}$$

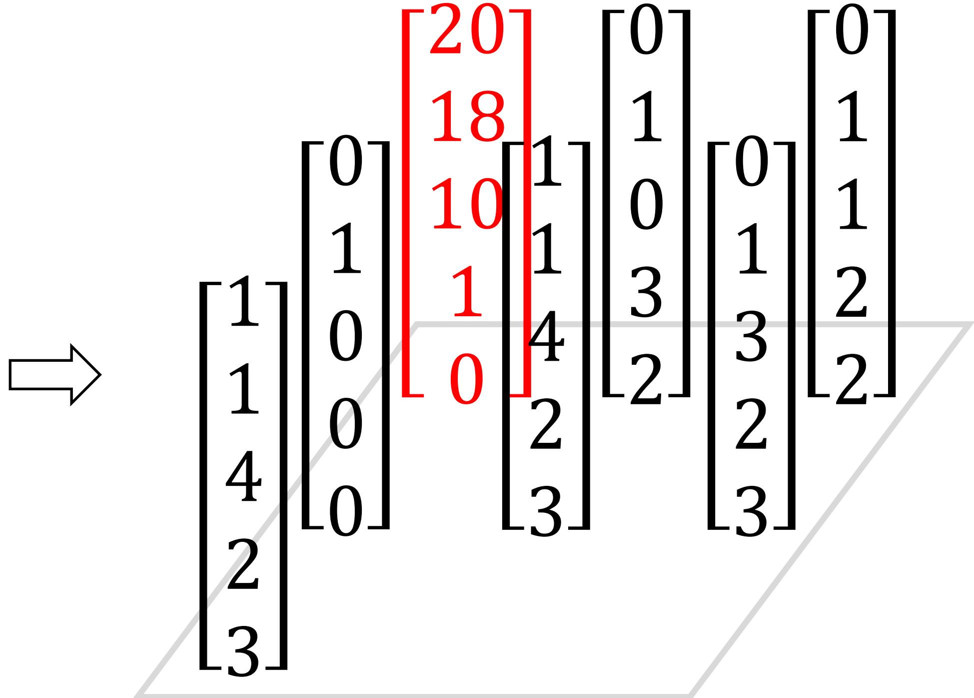
Key insight: the Receptive Field Size Determines the Number of Features Corrupted by the Adversarial Patch

Example 2: CNN with *small* receptive fields (e.g., BagNet with 17×17 px)



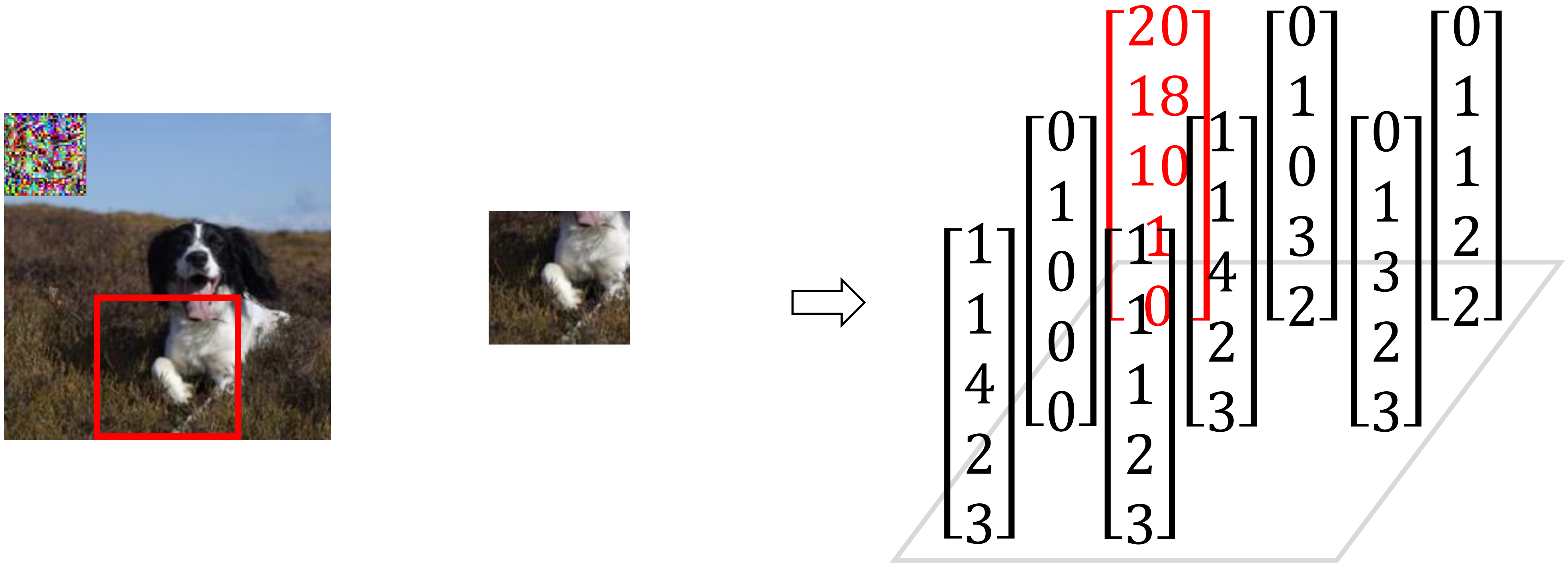
Key insight: the Receptive Field Size Determines the Number of Features Corrupted by the Adversarial Patch

Example 2: CNN with *small* receptive fields (e.g., BagNet with 17 × 17 px)



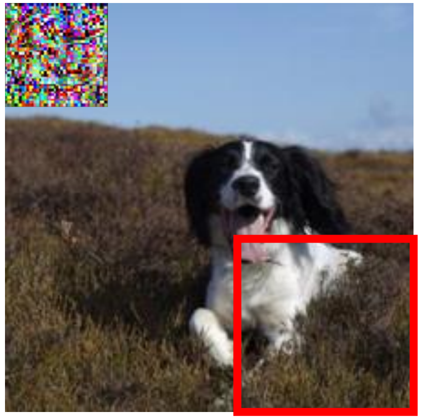
Key insight: the Receptive Field Size Determines the Number of Features Corrupted by the Adversarial Patch

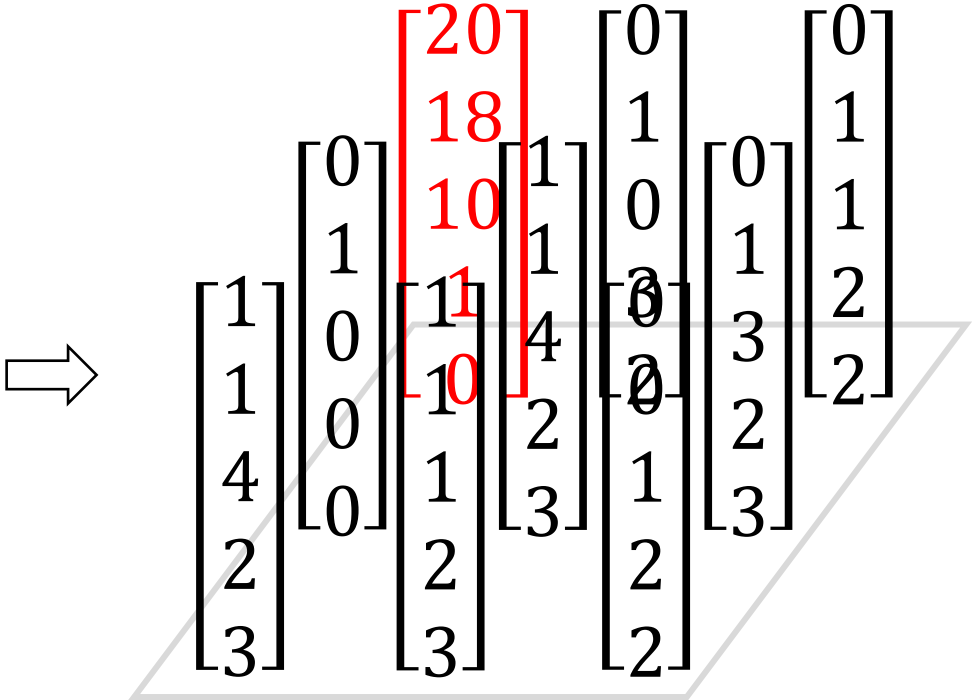
Example 2: CNN with *small* receptive fields (e.g., BagNet with 17 × 17 px)



Key insight: the Receptive Field Size Determines the Number of Features Corrupted by the Adversarial Patch

Example 2: CNN with *small* receptive fields (e.g., BagNet with 17 × 17 px)





Note: *only one* feature corrupted!
A major step towards robust prediction!

Key insight: the Small Receptive Field Size Bounds the Number of Features Corrupted by the Adversarial Patch

Number of corrupted features k (along one axis) satisfies:

$$k = \frac{p + r - 1}{s}$$

p patch size; r receptive field size; s receptive field stride
(more details are in the paper)

A smaller receptive field gives fewer corrupted features!

Our Contribution: PatchGuard Defense Framework with Provable Robustness

PatchGuard aims to prevent the localized patch from dominating the global prediction

PatchGuard: A Provably Robust Defense Framework

Small Receptive Field

Bound the number of corrupted features

Secure Feature Aggregation

Do robust prediction on partially corrupted features

Our Contribution: PatchGuard Defense Framework with Provable Robustness

PatchGuard aims to prevent the localized patch from dominating the global prediction

PatchGuard: A Provably Robust Defense Framework

Small Receptive Field

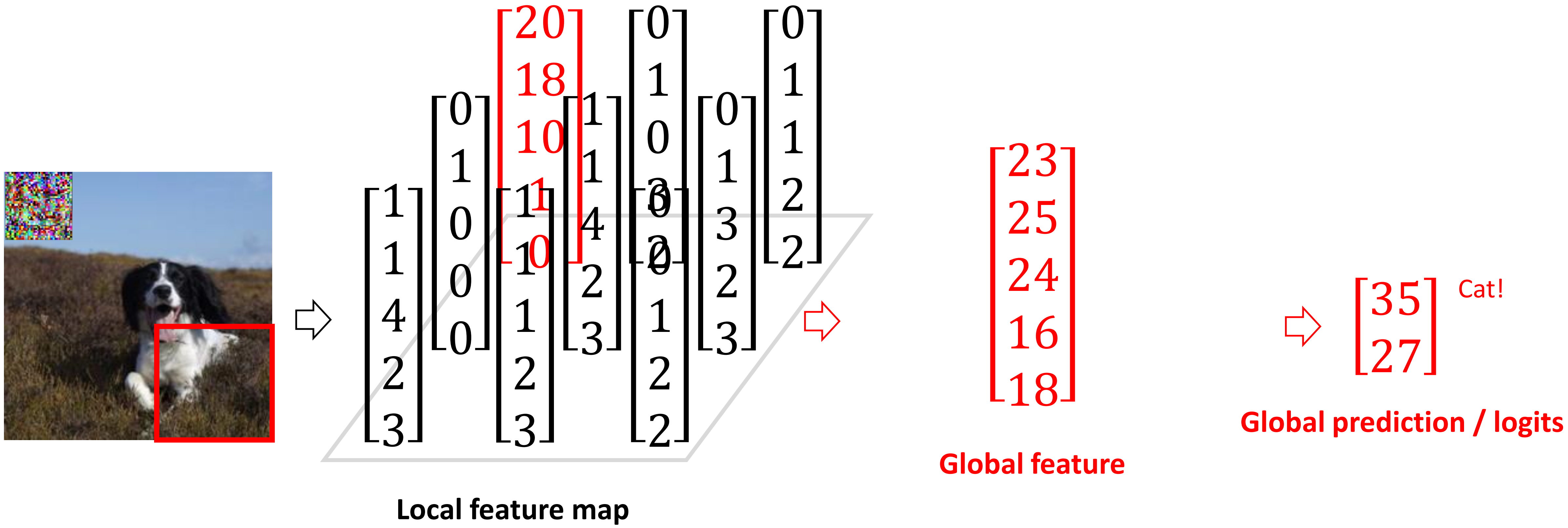
Bound the number of corrupted features

Secure Feature Aggregation

Do robust prediction on partially corrupted features

Vulnerability of Insecure Feature Aggregation

<u>Extremely large</u> malicious values dominate the insecure feature aggregation and global prediction

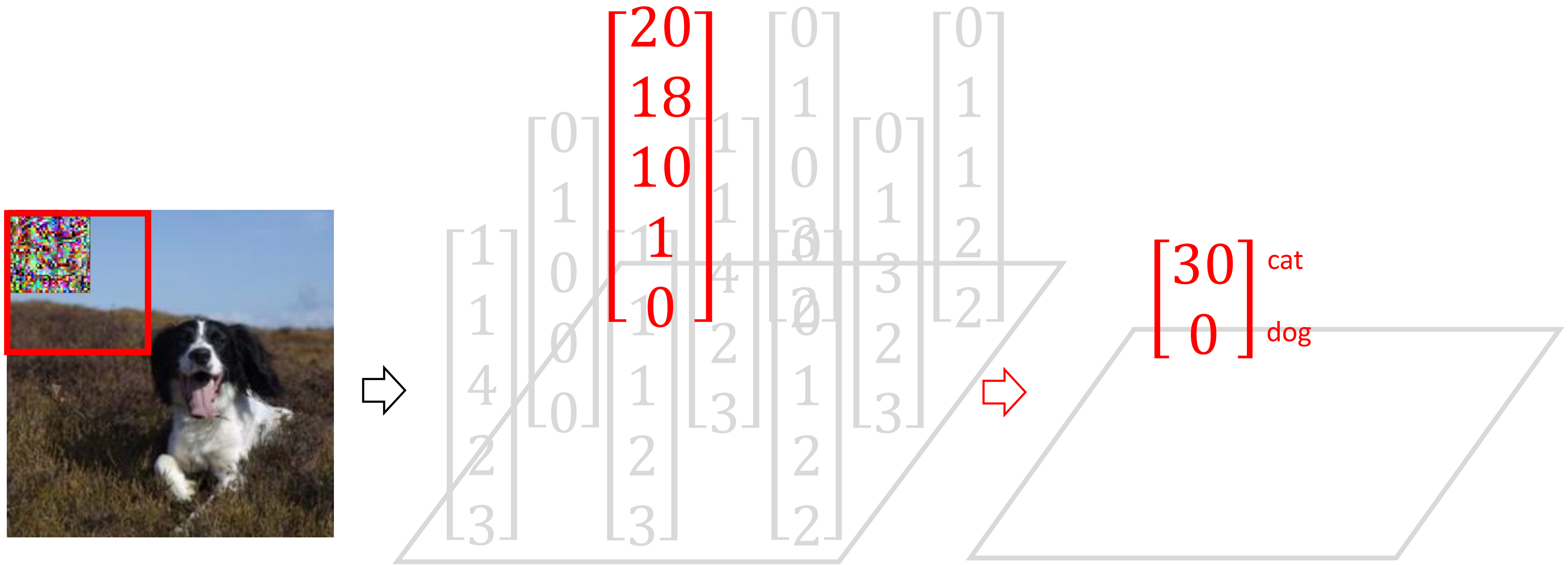


Secure feature aggregation to limit the adversarial effect!

- Robust masking to detect and remove large values

Leveraging Local Logits for Robust Masking

Local logits: making local prediction based on the local feature

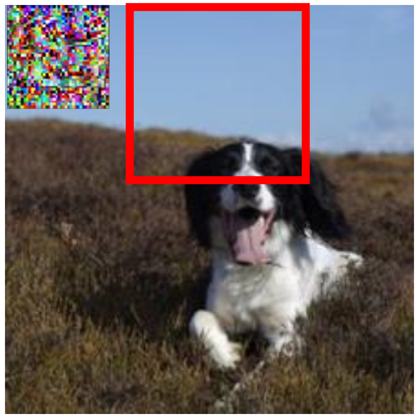


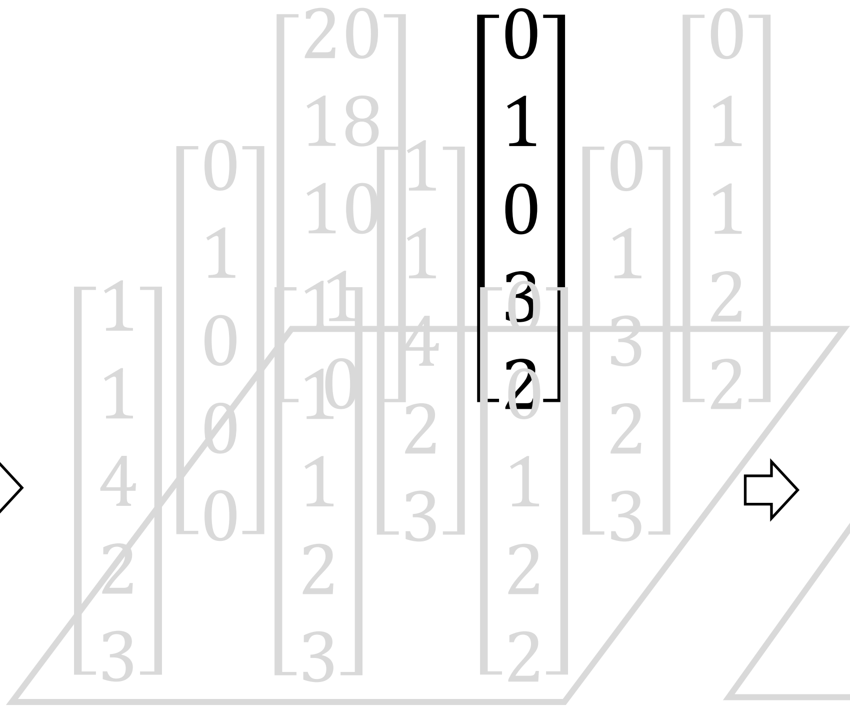
Local feature map

Local prediction / logits map

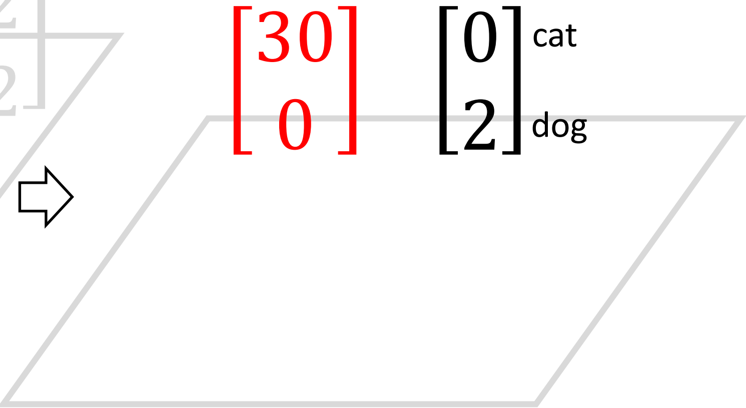
Leveraging Local Logits for Robust Masking

Local logits: making local prediction based on the local feature





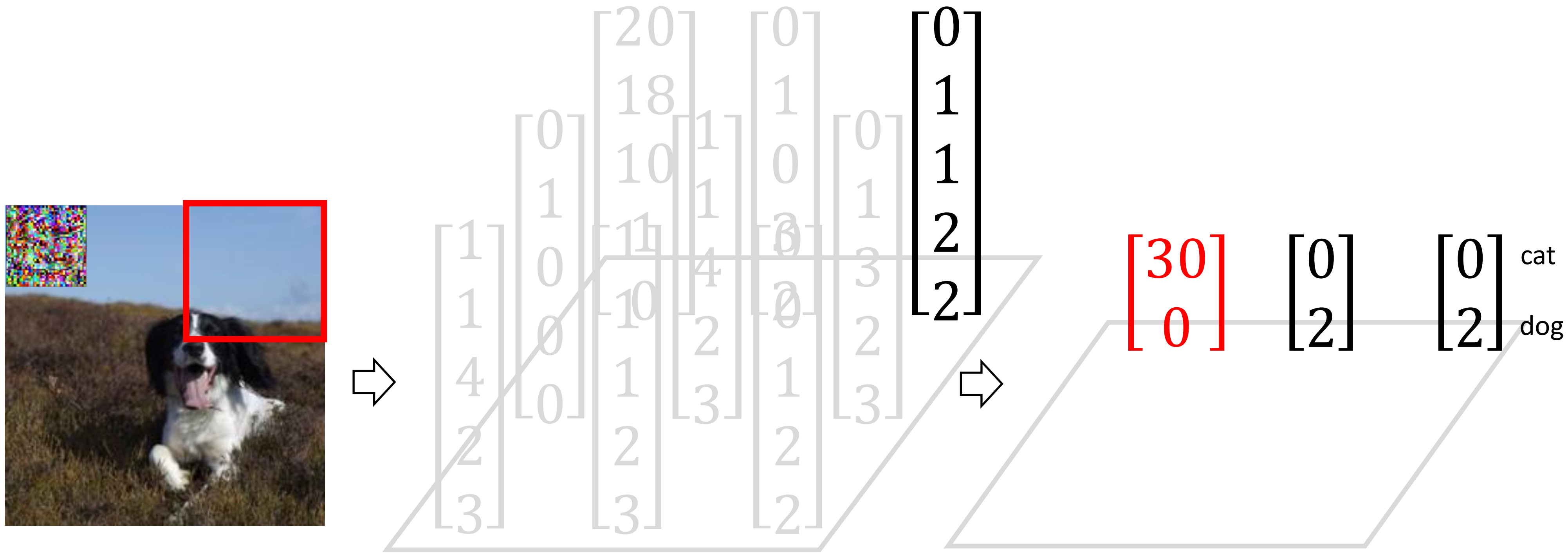
Local feature map



Local prediction / logits map

Leveraging Local Logits for Robust Masking

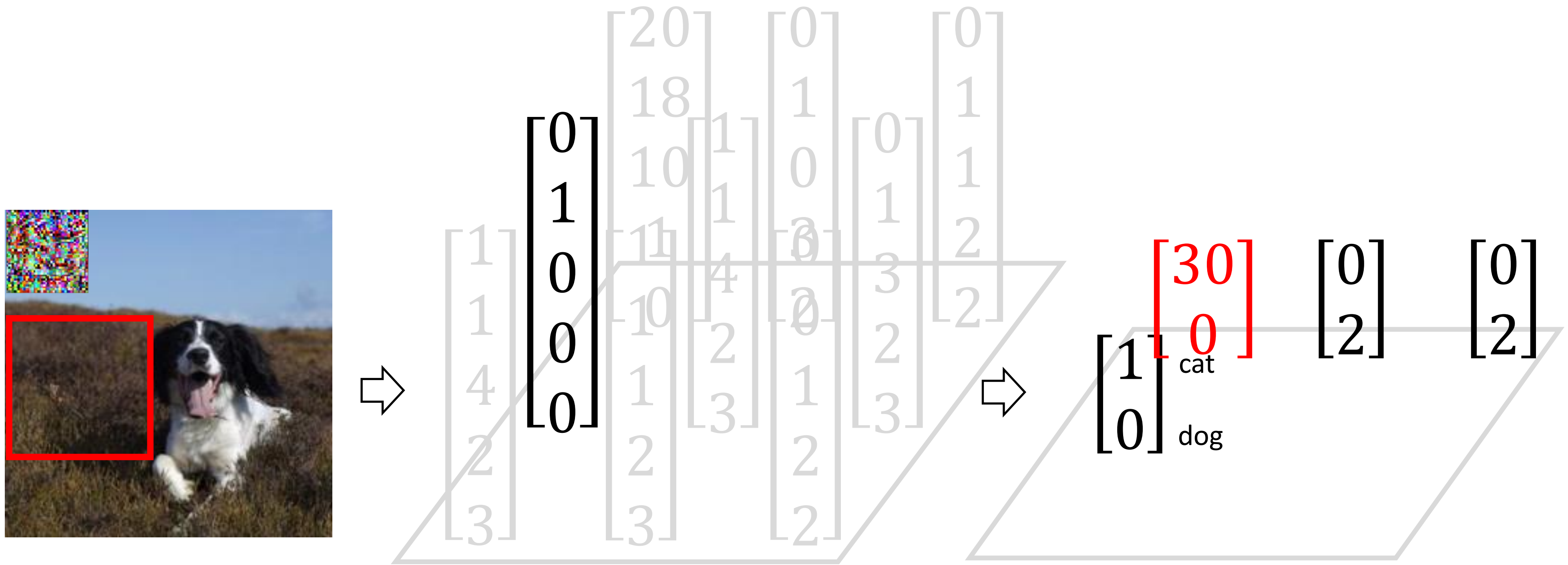
Local logits: making local prediction based on the local feature



Local feature map **Local prediction / logits map**

Leveraging Local Logits for Robust Masking

Local logits: making local prediction based on the local feature

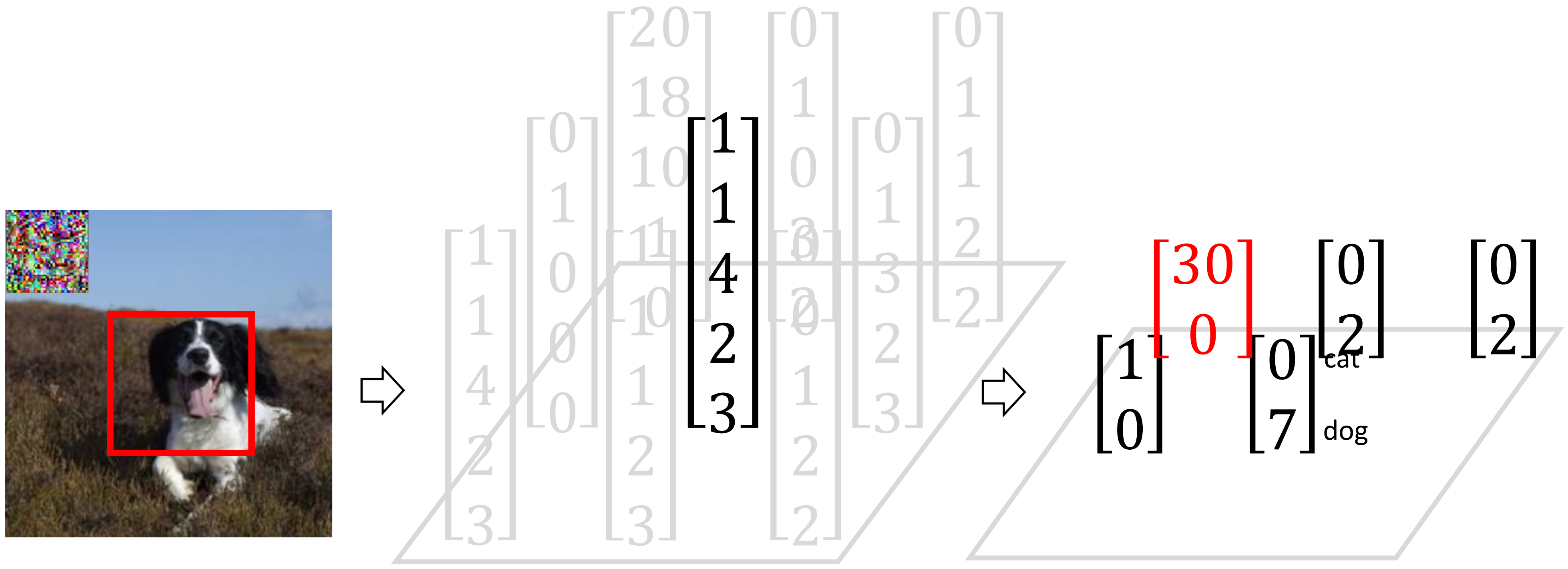


Local feature map

Local prediction / logits map

Leveraging Local Logits for Robust Masking

Local logits: making local prediction based on the local feature

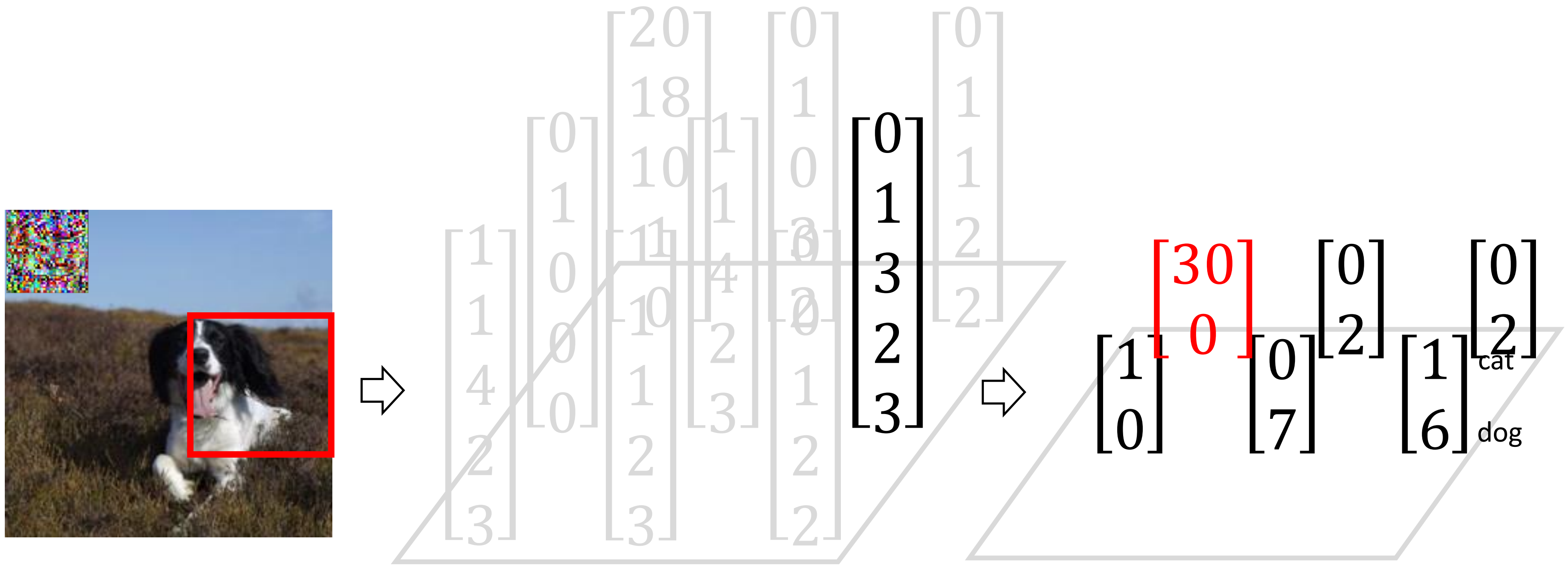


Local feature map

Local prediction / logits map

Leveraging Local Logits for Robust Masking

Local logits: making local prediction based on the local feature

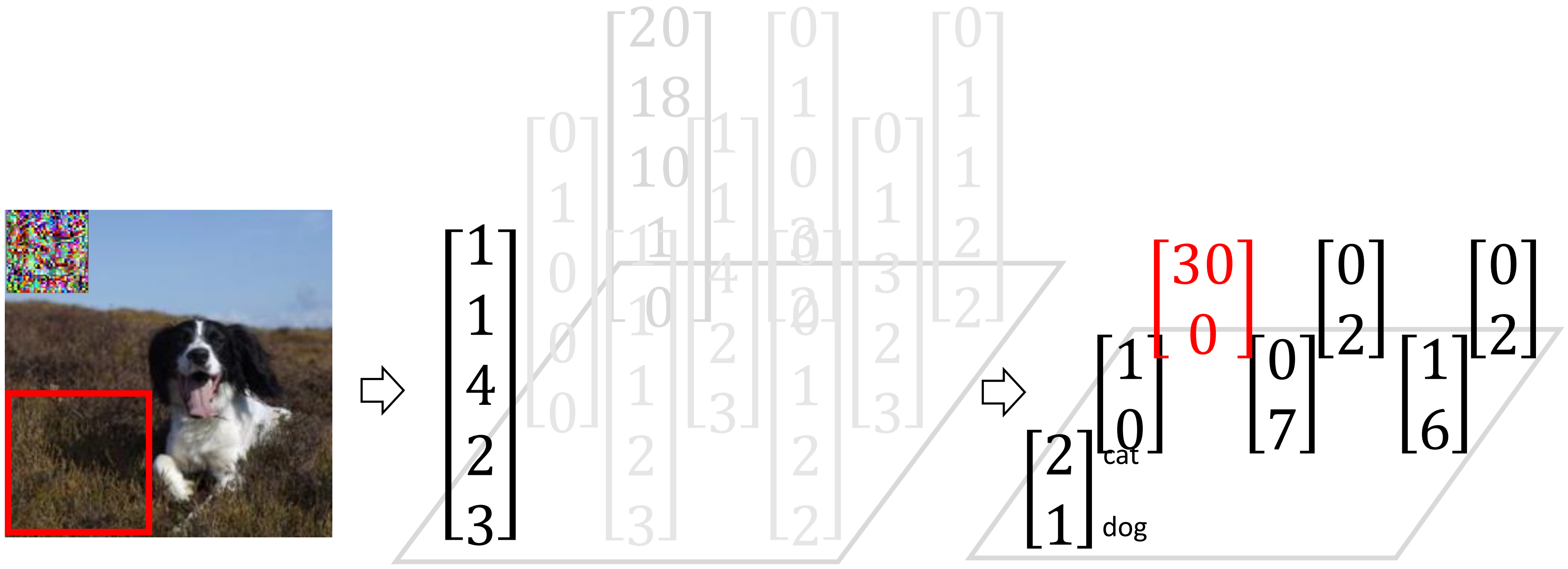


Local feature map

Local prediction / logits map

Leveraging Local Logits for Robust Masking

Local logits: making local prediction based on the local feature

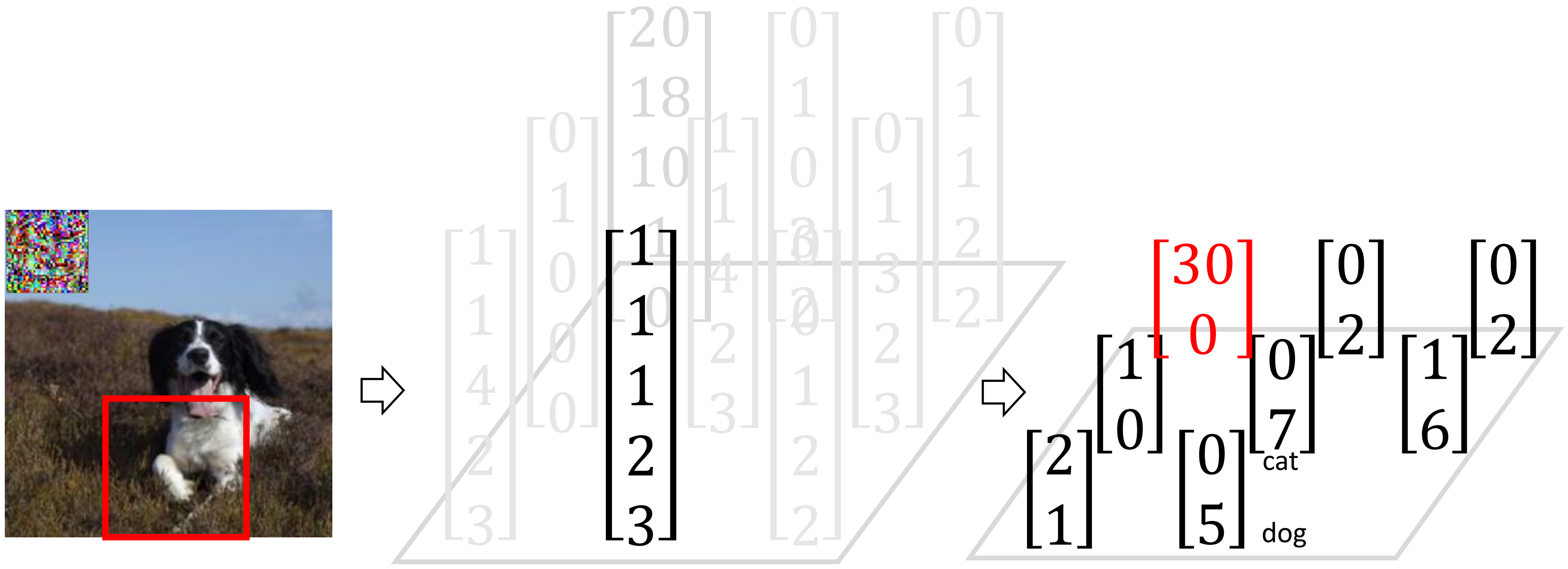


Local feature map

Local prediction / logits map

Leveraging Local Logits for Robust Masking

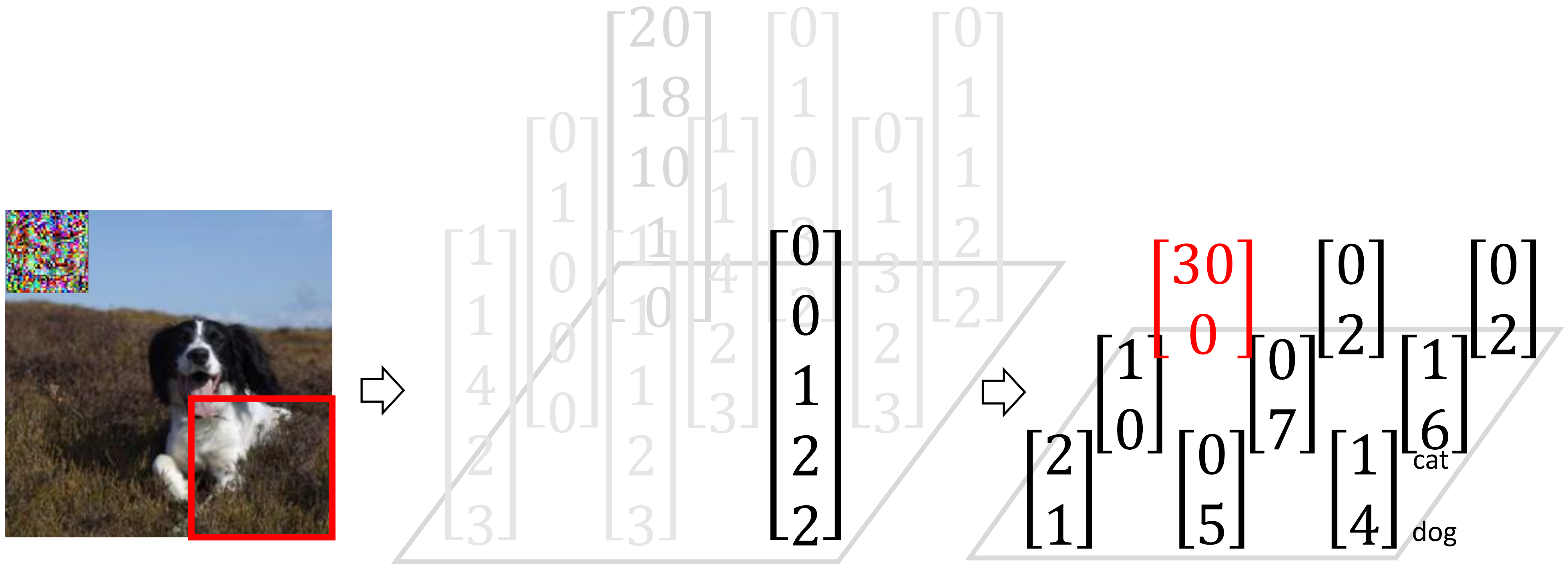
Local logits: making local prediction based on the local feature



Local feature map **Local prediction / logits map**

Leveraging Local Logits for Robust Masking

Local logits: making local prediction based on the local feature

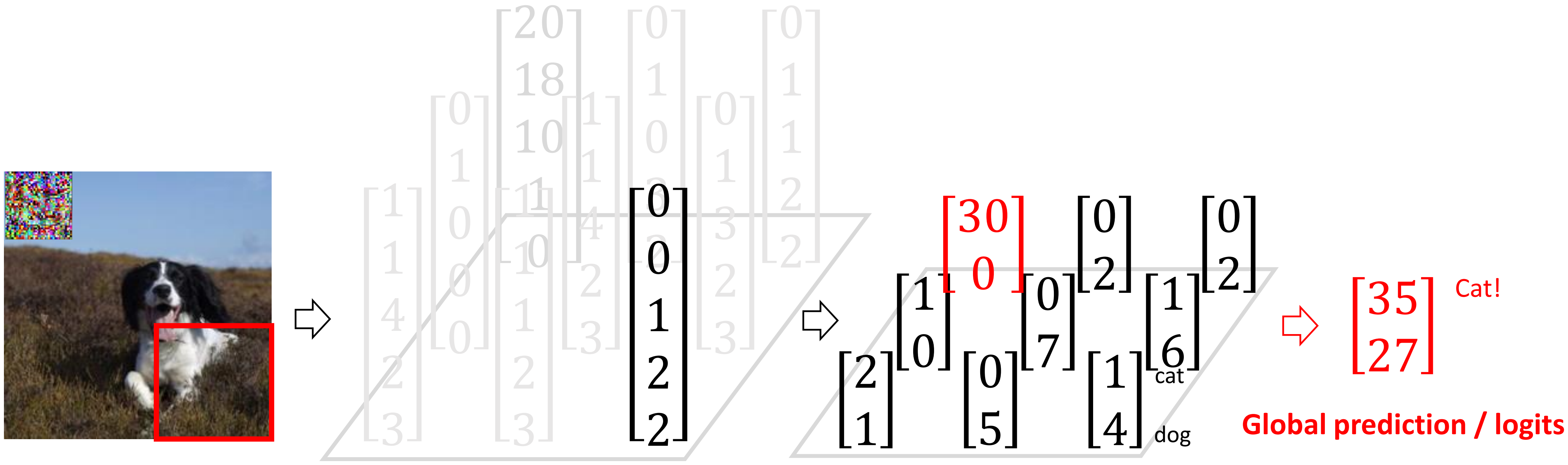


Local feature map

Local prediction / logits map

Leveraging Local Logits for Robust Masking

Aggregating local logits gives the same global logits prediction

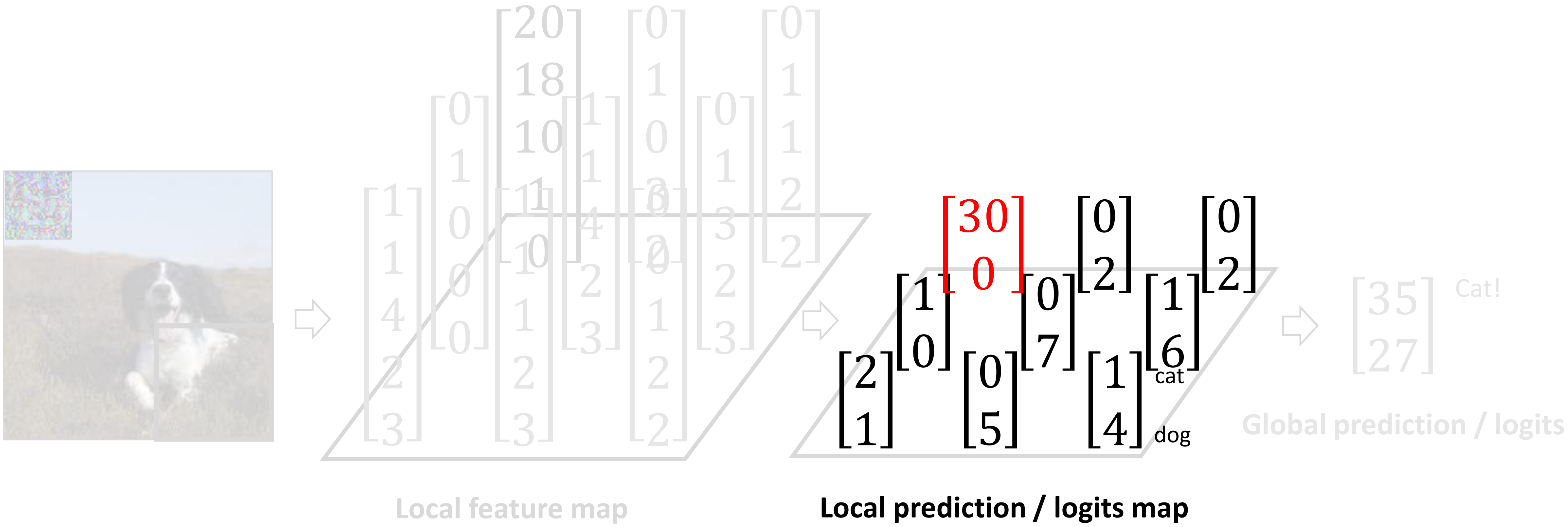


Local feature map

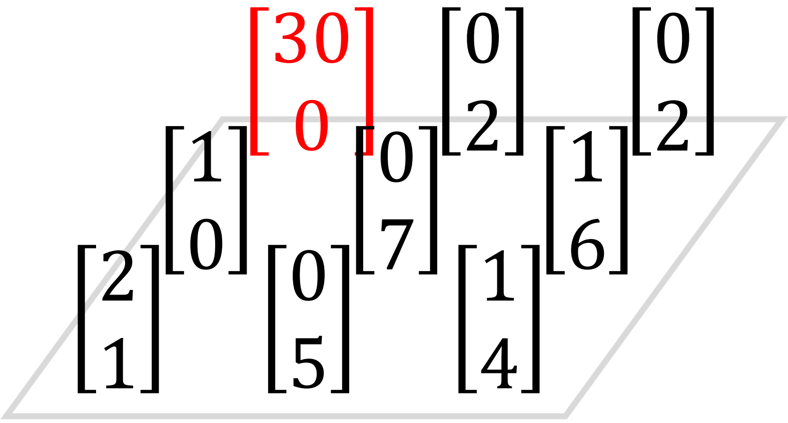
Local prediction / logits map

Leveraging Local Logits for Robust Masking

Aggregating local logits gives the same global logits prediction



A Better Visualization: Local Logits Map Slice



$$\begin{bmatrix} \color{red}{30} & 0 & 0 \\ 1 & 0 & 1 \\ 2 & 0 & 1 \end{bmatrix}$$

local logits map slice for cat

<u>Cat: 35</u>

$$\begin{bmatrix} \color{red}{0} & 2 & 2 \\ 0 & 7 & 6 \\ 1 & 5 & 4 \end{bmatrix}$$

local logits map slice for dog

Dog: 27

- One local logits map slice for one class
- Class evidence: elements of each slice

Robust Masking: Algorithm

$\square$ $$\begin{bmatrix} 30 & 0 & 0 \\ 1 & 0 & 1 \\ 2 & 0 & 1 \end{bmatrix}$$

local logits map slice for cat

Cat: 35

$\square$ $$\begin{bmatrix} 0 & 2 & 2 \\ 0 & 7 & 6 \\ 1 & 5 & 4 \end{bmatrix}$$

local logits map slice for dog

Dog: 27

Robust Masking:

1. Clip all negative values to zeros
2. Move a sliding window over each local logits slice (1×1 window here)
3. Calculate class evidence sum within each window
4. Mask the window with the highest sum

Robust Masking: Prediction in the Adversarial Setting

$$\begin{bmatrix} 30 & 0 & 0 \\ 1 & 0 & 1 \\ 2 & 0 & 1 \end{bmatrix}$$

local logits map slice for cat

Cat: 35

$$\begin{bmatrix} 0 & 2 & 2 \\ 0 & 7 & 6 \\ 1 & 5 & 4 \end{bmatrix}$$

local logits map slice for dog

Dog: 27

Robust Masking:

1. Clip all negative values to zeros
2. Move a <u>sliding window</u> over each local logits slice (1×1 window here)
3. Calculate class evidence <u>sum</u> within each window
4. Mask the window with the <u>highest</u> sum

Robust Masking: Prediction in the Adversarial Setting

$$\begin{bmatrix} \boxed{\cancel{30}} & 0 & 0 \\ 1 & 0 & 1 \\ 2 & 0 & 1 \end{bmatrix}$$

local logits map slice for cat

Cat: 5

$$\begin{bmatrix} 0 & 2 & 2 \\ 0 & \boxed{\cancel{7}} & 6 \\ 1 & 5 & 4 \end{bmatrix}$$

local logits map slice for dog

<u>Dog: 20</u>

Robust Masking:

1. Clip all negative values to zeros
2. Move a <u>sliding window</u> over each local logits slice (1×1 window here)
3. Calculate class evidence <u>sum</u> within each window
4. Mask the window with the <u>highest</u> sum

The prediction in the adversarial setting is <u>subject to partial feature masking</u>

Robust Masking: Prediction in the Clean Setting

$$\begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 2 & 0 & 1 \end{bmatrix}$$

local logits map slice for cat

Cat: 5

$$\begin{bmatrix} 1 & 2 & 2 \\ 0 & 7 & 6 \\ 1 & 5 & 4 \end{bmatrix}$$

local logits map slice for dog

<u>Dog: 28</u>

Robust Masking:

1. Clip all negative values to zeros
2. Move a <u>sliding window</u> over each local logits slice (1×1 window here)
3. Calculate class evidence <u>sum</u> within each window
4. Mask the window with the <u>highest</u> sum

Robust Masking: Prediction in the Clean Setting

$$\begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ \boxed{\cancel{2}} & 0 & 1 \end{bmatrix}$$

local logits map slice for cat

Cat: 3

$$\begin{bmatrix} 1 & 2 & 2 \\ 0 & \boxed{\cancel{7}} & 6 \\ 1 & 5 & 4 \end{bmatrix}$$

local logits map slice for dog

Dog: 21

Robust Masking:

1. Clip all negative values to zeros
2. Move a sliding window over each local logits slice (1×1 window here)
3. Calculate class evidence sum within each window
4. Mask the window with the highest sum

The prediction in the clean setting is generally invariant to partial feature masking

Our Contribution: PatchGuard Defense Framework with Provable Robustness

PatchGuard aims to prevent the localized patch from dominating the global prediction

PatchGuard: A Provably Robust Defense Framework

Small Receptive Field

Bound the number of corrupted features

Secure Feature Aggregation

Do robust prediction on partially corrupted features

Our Contribution: PatchGuard Defense Framework with Provable Robustness

PatchGuard aims to prevent the localized patch from dominating the global prediction

PatchGuard: A Provably Robust Defense Framework

Small Receptive Field

Bound the number of corrupted features

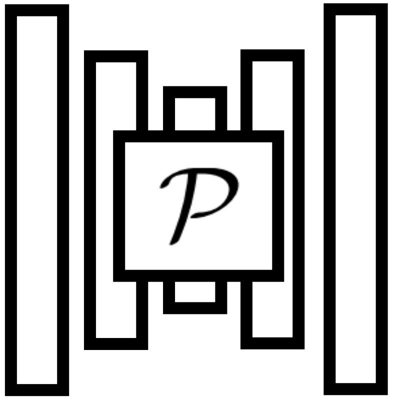
Secure Feature Aggregation

Do robust prediction on partially corrupted features

Recall: Provable Robustness on Certified Test Images

Test Image

Threat Model
(patch sizes, shapes, and location set)



Provable Analysis

Ground-truth y

The prediction is <u>**always**</u> correct

"I think it is a $\boxed{\text{dog}}$"
<u>"My prediction will never change"</u>

or

"I can't say anything for sure"

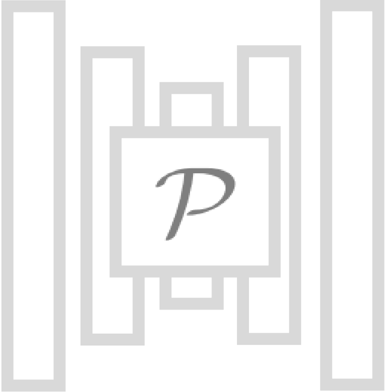
Provable robust accuracy / certified accuracy: the fraction of test images that are

1. <u>Correctly</u> classified
2. <u>Provably robust</u> to any (adaptive) localized patch attack within the threat model

Recall: Provable Robustness on Certified Test Images

Test Image

Threat Model
(patch sizes, shapes, and location set)



Provable Analysis

Ground-truth y

The prediction is **always** correct

"I think it is a $\boxed{\text{dog}}$"
"My prediction will never change"

or

"I can't say anything for sure"

Provable robust accuracy / certified accuracy: the fraction of test images that are

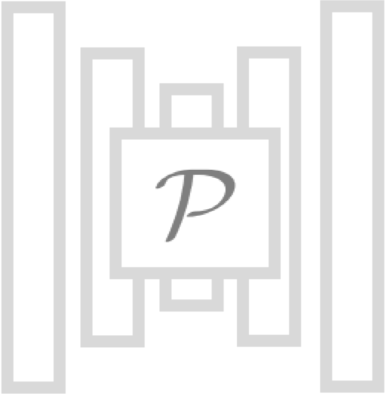
1. Correctly classified
2. Provably robust to any (adaptive) localized patch attack within the threat model

Recall: Provable Robustness on Certified Test Images

Test Image

Threat Model
(patch sizes, shapes, and location set)

⇨



Provable Analysis

⇨

Ground-truth y

The prediction is **always** correct

"I think it is a dog"
"My prediction will never change"

or

"I can't say anything for sure"

Provable robust accuracy / certified accuracy: the fraction of test images that are

1. Correctly classified
2. Provably robust to any (adaptive) localized patch attack within the threat model

Provable Analysis

$$\begin{bmatrix} ? & 0 & 0 \\ 1 & 0 & 1 \\ 2 & 0 & 1 \end{bmatrix}$$

local logits map slice for cat

Cat: ?

$$\begin{bmatrix} ? & 2 & 2 \\ 0 & 7 & 6 \\ 1 & 5 & 4 \end{bmatrix}$$

local logits map slice for dog

Dog: ?

The adversary can control values within a small window (1×1 window here)

Provable Analysis: Upper Bound of Class Evidence

$$\begin{bmatrix} \color{red}{?} & 0 & 0 \\ 1 & 0 & 1 \\ 2 & 0 & 1 \end{bmatrix}$$

local logits map slice for cat

Cat: ?

$$\begin{bmatrix} \color{red}{?} & 2 & 2 \\ 0 & 7 & 6 \\ 1 & 5 & 4 \end{bmatrix}$$

local logits map slice for dog

Dog: ?

The adversary can control values within a small window (1×1 window here)

1. **The adversary cannot increase the malicious class evidence too much**

Provable Analysis: Upper Bound of Class Evidence

$$\begin{bmatrix} \color{red}{3} & 0 & 0 \\ 1 & 0 & 1 \\ 2 & 0 & 1 \end{bmatrix}$$

local logits map slice for cat

Cat: ?

$$\begin{bmatrix} ? & 2 & 2 \\ 0 & 7 & 6 \\ 1 & 5 & 4 \end{bmatrix}$$

local logits map slice for dog

Dog: ?

The adversary can control values within a small window (1×1 window here)

1. **The adversary cannot increase the malicious class evidence too much**
   - A large value will be masked

Provable Analysis: Upper Bound of Class Evidence

$$\begin{bmatrix} \cancel{3} & 0 & 0 \\ 1 & 0 & 1 \\ 2 & 0 & 1 \end{bmatrix}$$

local logits map slice for cat

Cat: 5

$$\begin{bmatrix} ? & 2 & 2 \\ 0 & 7 & 6 \\ 1 & 5 & 4 \end{bmatrix}$$

local logits map slice for dog

Dog: ?

The adversary can control values within a small window (1×1 window here)

1. **The adversary cannot increase the malicious class evidence too much**
   - A large value will be masked
   - The robust masking imposes an upper bound of the class evidence sum

Provable Analysis: Lower Bound of Class Evidence

$$\begin{bmatrix} ? & 0 & 0 \\ 1 & 0 & 1 \\ 2 & 0 & 1 \end{bmatrix}$$

local logits map slice for cat

Cat: 5

$$\begin{bmatrix} ? & 2 & 2 \\ 0 & 7 & 6 \\ 1 & 5 & 4 \end{bmatrix}$$

local logits map slice for dog

Dog: ?

The adversary can control values within a small window (1×1 window here)

1. **The adversary cannot increase the malicious class evidence too much**
   - A large value will be masked
   - The robust masking imposes an upper bound of the class evidence sum
2. **The adversary cannot decrease the benign class evidence too much**

Provable Analysis: Lower Bound of Class Evidence

$$\begin{bmatrix} ? & 0 & 0 \\ 1 & 0 & 1 \\ 2 & 0 & 1 \end{bmatrix}$$

local logits map slice for cat

Cat: 5

$$\begin{bmatrix} 0 & 2 & 2 \\ 0 & 7 & 6 \\ 1 & 5 & 4 \end{bmatrix}$$

local logits map slice for dog

Dog: ?

The adversary can control values within a small window (1×1 window here)

1. **The adversary cannot increase the malicious class evidence too much**
   - A large value will be masked
   - The robust masking imposes an upper bound of the class evidence sum
2. **The adversary cannot decrease the benign class evidence too much**
   - Can only push malicious values to zero

Provable Analysis: Lower Bound of Class Evidence

$$\begin{bmatrix} \cancel{2} & 0 & 0 \\ 1 & 0 & 1 \\ 2 & 0 & 1 \end{bmatrix}$$

local logits map slice for cat

Cat: 5

$$\begin{bmatrix} \color{red}{0} & 2 & 2 \\ 0 & \cancel{7} & 6 \\ 1 & 5 & 4 \end{bmatrix}$$

local logits map slice for dog

Dog: 20

The adversary can control values within a small window (1×1 window here)

1. **The adversary cannot increase the malicious class evidence too much**
   - A large value will be masked
   - The robust masking imposes an upper bound of the class evidence sum
2. **The adversary cannot decrease the benign class evidence too much**
   - Can only push malicious values to zero
   - Clipping all negative values imposes a lower bound of the class evidence sum

Provable Analysis: Bounds hold for Any Attack Strategy

$$\begin{bmatrix} \cancel{2} & 0 & 0 \\ 1 & 0 & 1 \\ 2 & 0 & 1 \end{bmatrix}$$

local logits map slice for cat

Cat: 5

$$\begin{bmatrix} 0 & 2 & 2 \\ 0 & \cancel{7} & 6 \\ 1 & 5 & 4 \end{bmatrix}$$

local logits map slice for dog

Dog: 20

The adversary can control values within a small window (1×1 window here)

1. **The adversary cannot increase the malicious class evidence too much**
   - A large value will be masked
   - The robust masking imposes an upper bound of the class evidence sum
2. **The adversary cannot decrease the benign class evidence too much**
   - Can only push malicious values to zero
   - Clipping all negative values imposes a lower bound of the class evidence sum

We can derive bounds that apply to any attack strategy! (formal proof in the paper)

Provable Analysis: Example

$$\begin{bmatrix} \color{red}{?} & 0 & 0 \\ 1 & 0 & 1 \\ 2 & 0 & 1 \end{bmatrix}$$

local logits map slice for cat

Cat: ?

$$\begin{bmatrix} \color{red}{?} & 2 & 2 \\ 0 & 7 & 6 \\ 1 & 5 & 4 \end{bmatrix}$$

local logits map slice for dog

Dog: ?

	Lower Bound	Upper Bound
Cat	3	5
Dog	20	27

Provable Analysis: Example

$$\begin{bmatrix} ? & 0 & 0 \\ 1 & 0 & 1 \\ 2 & 0 & 1 \end{bmatrix}$$

local logits map slice for cat

Cat: ?

$$\begin{bmatrix} ? & 2 & 2 \\ 0 & 7 & 6 \\ 1 & 5 & 4 \end{bmatrix}$$

local logits map slice for dog

Dog: ?

	Lower Bound	Upper Bound
Cat	3	**5**
Dog	**20**	27

- **20 (lower bound of dog) > 5 (upper bound of cat)**
  - Provably Robust (always predicts dog)!
- Try all possible patch locations
  - This image is certified :)

Test Image

Threat Model

(patch sizes, shapes, and location set)

Evaluation: Substantial Provable Robustness

	10-class ImageNette	
Accuracy	Clean	Robust
PatchGuard	95.0%	86.7%

1. PatchGuard achieves substantial provable robustness

   (robustness evaluated against a 2%-pixel square patch *anywhere* on the image)

Evaluation: Substantial Provable Robustness

	10-class ImageNette		1000-class ImageNet	
Accuracy	Clean	Robust	Clean	Robust
PatchGuard	95.0%	86.7%	54.6%	26%

1. PatchGuard achieves substantial provable robustness
   (robustness evaluated against a 2%-pixel square patch *anywhere* on the image)

Evaluation: Substantial Provable Robustness

	10-class ImageNette		1000-class ImageNet	
Accuracy	Clean	Robust	Clean	Robust
PatchGuard	95.0%	86.7%	54.6%	26%
PatchGuard-Top-5	--	--	76.6%	56.9%

Top-5 accuracy for ImageNet is good!

1. PatchGuard achieves substantial provable robustness

    (robustness evaluated against a 2%-pixel square patch *anywhere* on the image)

Evaluation: State-of-the-art Clean Accuracy and Provable Robust Accuracy

	10-class ImageNette		1000-class ImageNet	
Accuracy	Clean	Robust	Clean	Robust
PatchGuard	**95.0%**	**86.7%**	**54.6%**	**26%**
IBP [1]	Computationally infeasible			
CBN [2]	94.9%	60.9%	49.5%	7.1%
DS [3]	92.1%	79.1%	44.4%	14.4%

2. IBP is too computationally expensive to scale to high-resolution images
3. PatchGuard significantly outperforms CBN and DS
    - Improvement from CBN on ImageNet:
        - 5% clean accuracy; 19% provable robust accuracy (2x better!)
    - Improvement from DS on ImageNet:
        - 10% clean accuracy; 12% provable robust accuracy (1x better!)

[1] Chiang et al., "Certified Defenses for Adversarial Patches," ICLR 2020
[2] Zhang et al., "Clipped bagnet: Defending against sticker attacks with clipped bag-of-features," DLS Workshop 2020
[3] Levine et al., "(De)randomized smoothing for certifiable defense against patch attacks," NeurIPS 2020

Discussion: Generalizability of PatchGuard

PatchGuard as a general defense framework

Provably Robust Defense	Small receptive field	Secure feature aggregation
PatchGuard (ours)	BagNet	Robust masking

Discussion: Generalizability of PatchGuard

PatchGuard as a general defense framework

Provably Robust Defense	Small receptive field	Secure feature aggregation
PatchGuard (ours)	BagNet	Robust masking
Clipped BagNet (CBN) [1]	BagNet	Clipping + Average pooling
Derandomized Smoothing (DS) [2]	Pixel patches to ResNet	Majority voting

Discussion: Generalizability of PatchGuard

PatchGuard as a general defense framework

Provably Robust Defense	Small receptive field	Secure feature aggregation
PatchGuard (ours)	BagNet	Robust masking
Clipped BagNet (CBN) [1]	BagNet	Clipping + Average pooling
Derandomized Smoothing (DS) [2]	Pixel patches to ResNet	Majority voting
BagCert [3]	Modified BagNet	Majority voting
Randomized Cropping [4]	Cropped images to ResNet	Majority voting

[1] Zhang et al., "Clipped bagnet: Defending against sticker attacks with clipped bag-of-features," DLS Workshop 2020
[2] Levine et al., "(De)randomized smoothing for certifiable defense against patch attacks," NeurIPS 2020
[3] Metzen et al., "Efficient certified defenses against patch attacks on image classifiers," ICLR 2021
[4] Lin et al. "Certified robustness against physically-realizable patch attack via randomized cropping," ICLR Open Review 2021

Discussion: Limitations

1. The small receptive field hurts the clean accuracy (provable robustness vs. clean accuracy trade-off)
   - The accuracy drop is especially obvious for ImageNet (from 76.1% to 56.5%)

	10-class ImageNette		1000-class ImageNet	
	Clean	Robust	Clean	Robust
ResNet-50 (483×483)	99.6%	--	76.1%	--
BagNet-17 (17×17)	95.9%	--	56.5%	--
PatchGuard	95.0%	86.7%	54.6%	26%
PatchGuard-Top-5			76.6%	56.9%

2. The masking operation requires additional parameters (e.g., number of masks, mask size, mask shape)

Takeaways

1. **PatchGuard: a General Defense Framework**
    - Small receptive field
    - Secure feature aggregation

2. **Provably Robust Defense**
    - Predictions are always correct on certified images

3. **State-of-the-art Defense Performance**
    - Clean accuracy
    - Provable robust accuracy

Thank you!

Chong Xiang
Princeton University
cxiang@princeton.edu

Arjun Nitin Bhagoji
University of Chicago
abhagoji@uchicago.edu

Vikash Sehwag
Princeton University
vvikash@princeton.edu

Prateek Mittal
Princeton University
pmittal@princeton.edu

Technical Report

GitHub