THE UNIVERSITY OF MELBOURNE

AJCAI 2024 Encore Track:

Trustworthy AI: Robustness and Ethics

Improving Accuracy-robustness Trade-off via Pixel Reweighted Adversarial Training

Presenter: Jiacheng Zhang

School of Computing and Information Systems

The University of Melbourne

Date: 26 November 2024

What is an adversarial example (attack)?

- **Left-or-right challenge:** Guess which one is the adversarial example?

What is an adversarial example (attack)?

99% Guacamole

88% Tabby Cat

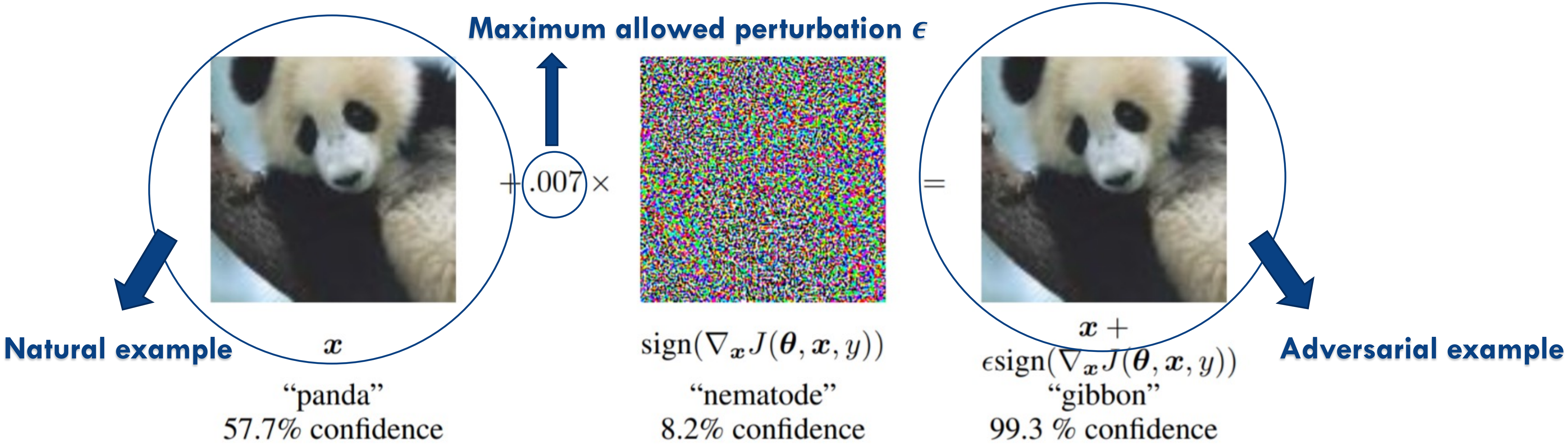
What is an adversarial example (attack)?

Adversarial examples can significantly drop the classification accuracy to **0%.**

How it works?

What is an adversarial example (attack)?

- Adding **imperceptible, non-random perturbations** to input data.

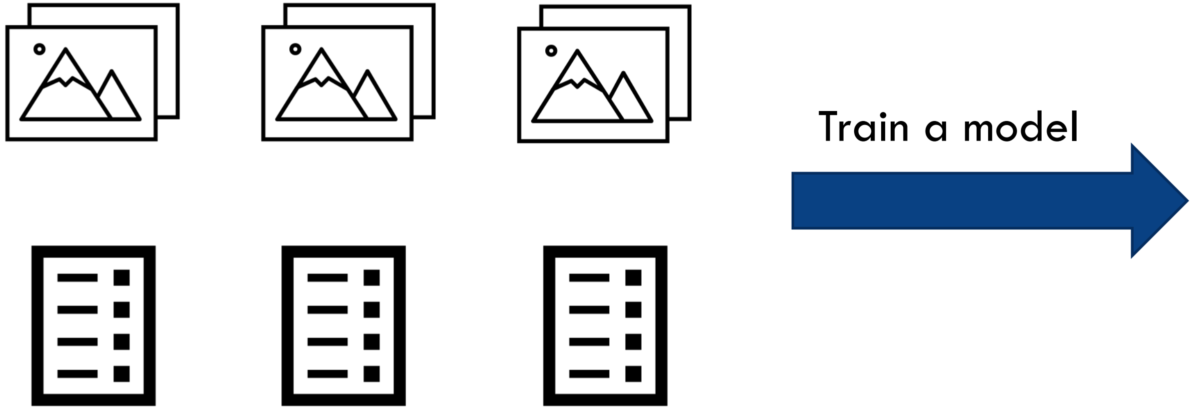


- Cannot fool human eyes, but **can easily fool** state-of-the-art neural networks.

Explaining and Harnessing Adversarial Examples. In *ICLR*, 2015.

What is an adversarial example (attack)?

Conventional Machine Learning Pipeline (classification):

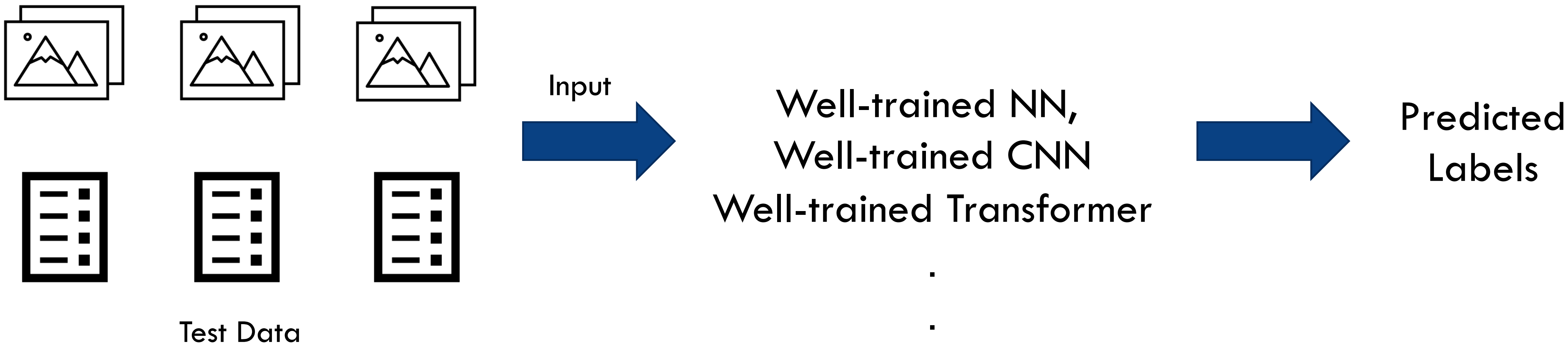


NN,
CNN
Transformer
.
.

Training Data with labels

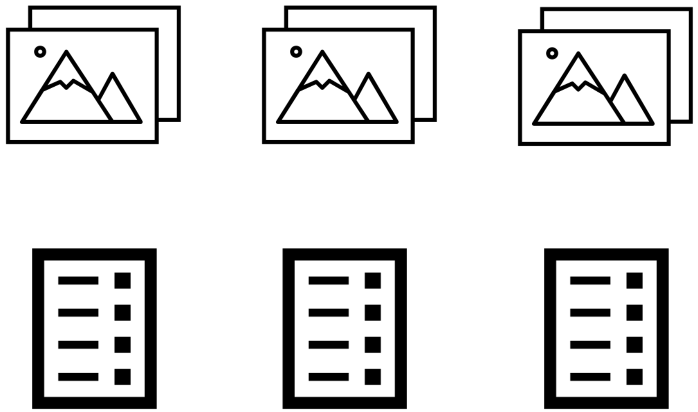
What is an adversarial example (attack)?

Conventional Machine Learning Pipeline (classification):



What is an adversarial example (attack)?

Adversarial attack happens:



Test Data + Adversarial **Perturbations**

Input →

Well-trained NN,
Well-trained CNN
Well-trained Transformer

.

.

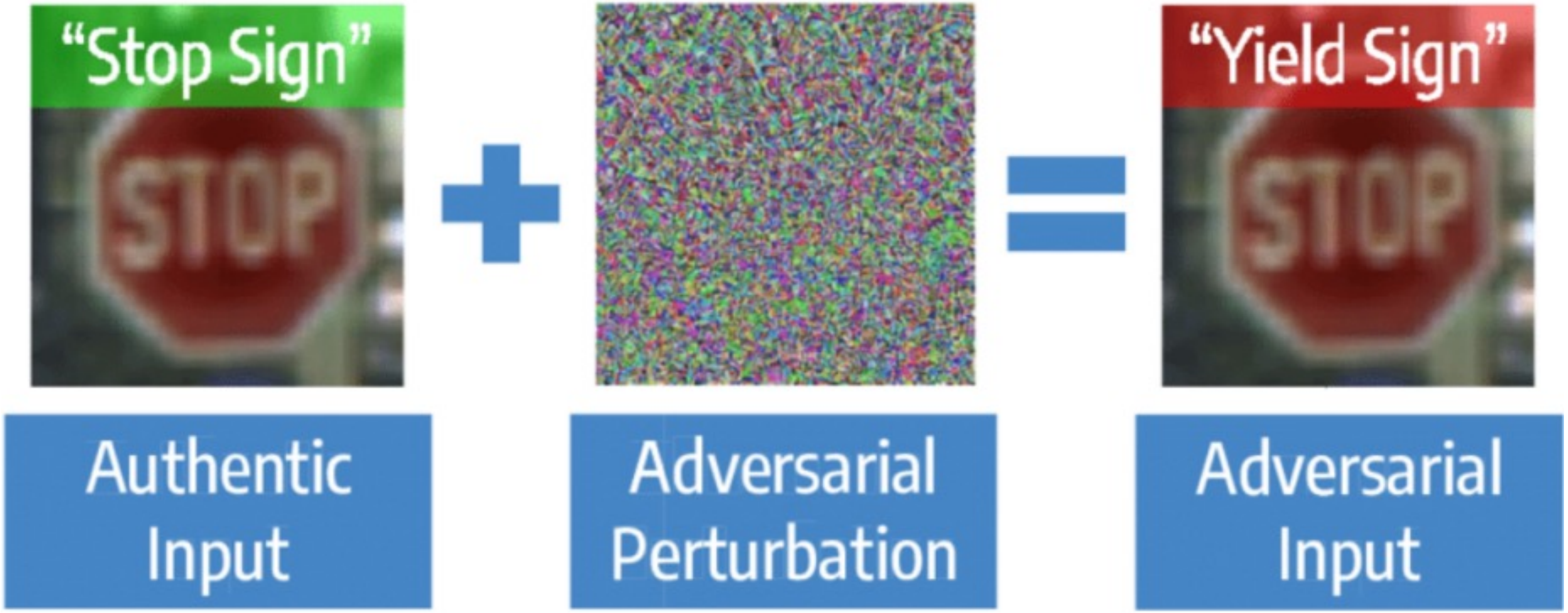
→

Predicted Labels

Untrusted

Why do we care?

- ❑ Cause **security and reliability issues** in the deployment of machine learning systems.
- ❑ E.g., mislead the autonomous driving system to recognize **a stop sign** into **something else**.



Black-box adversarial attacks in autonomous vehicle technology. In 49th IEEE Applied Imagery Pattern Recognition Workshop, 2020.

Why do we care?

- ❑ Adding **adversarial examples** on T-shirts can bypass the AI detection system.
- ❑ Let you be invisible to the AI detection system!
- ❑ It's cool but it can cause **security and reliability issues.**

Adversarial T-shirt! Evading Person Detectors in A Physical World. ArXiv 2019.

How to defend against adversarial attacks?

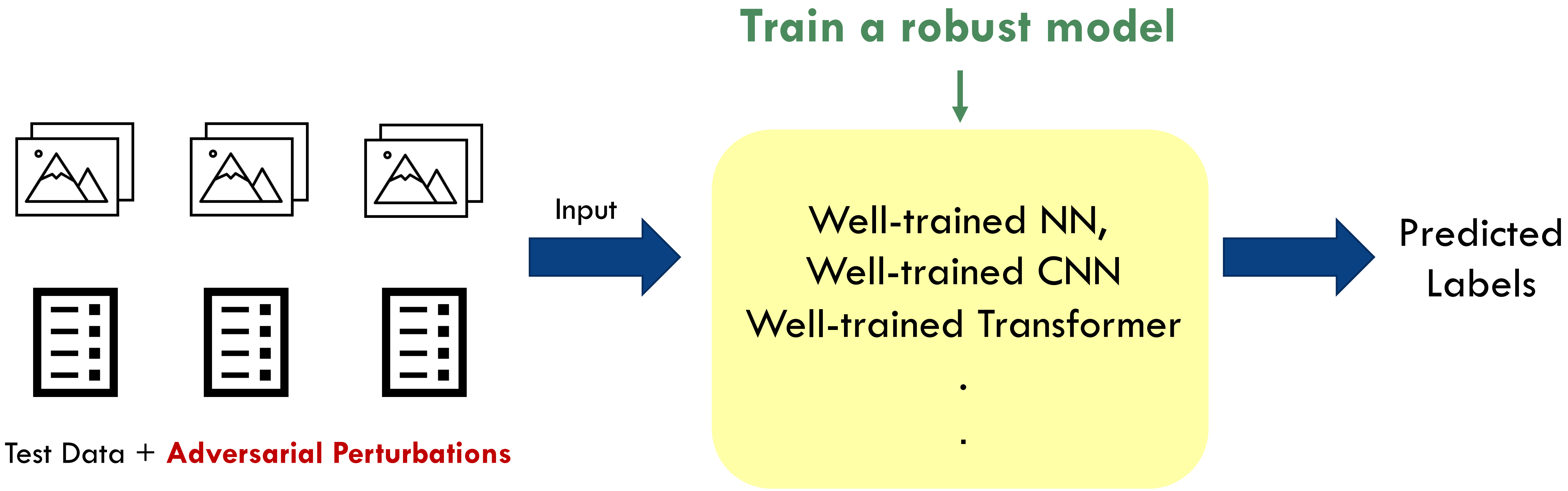
Today’s Focus

Trustworthy Machine Learning Under Adversarial Data

- Actively Handle Adversarial Data
  - Model Perspective — **Adversarial Training**
  - Data Perspective — **Adversarial Purification**
- Passively Handle Adversarial Data — **Adversarial Detection**

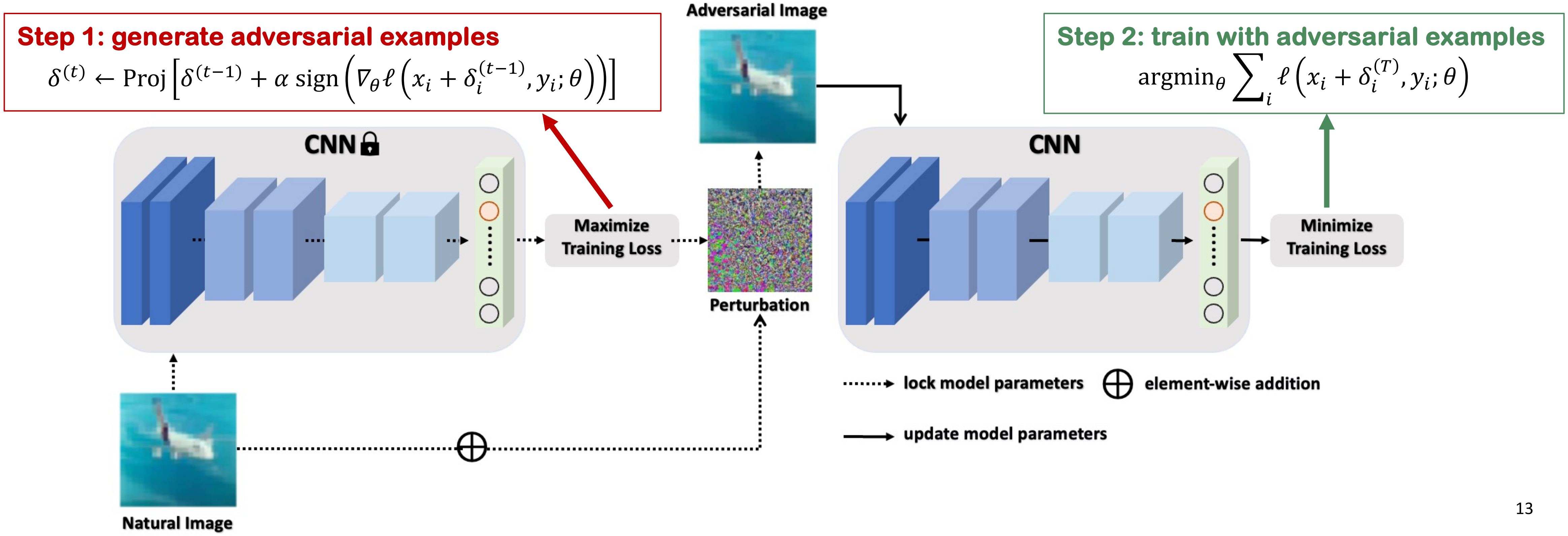
Adversarial training

- ❑ *Adversarial Training* (AT): aims to train a robust model on *adversarial examples* (AEs)



Adversarial training

- *Adversarial Training* (AT): aims to train a robust model on *adversarial examples* (AEs)



What's the problem of adversarial training?

What's good about AT?

- ❑ AT improves robustness (i.e., accuracy on adversarial examples).

What's bad about AT?

- ❑ AT drops natural accuracy (i.e., accuracy on natural examples).

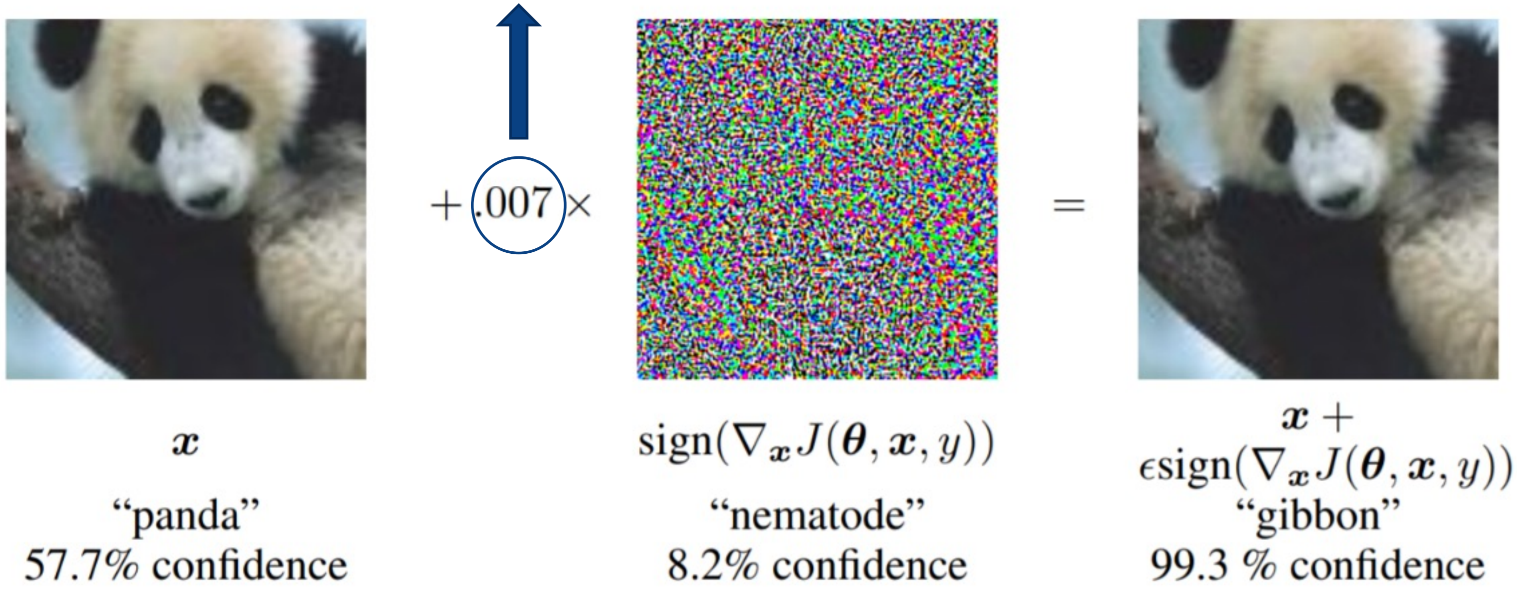
Problem: there is an accuracy-robustness trade-off!!!

- ❑ Higher robustness, lower accuracy; higher accuracy, lower robustness.

What we really want: a model that has high robustness without sacrificing natural accuracy.

How to mitigate this problem?

What will happen if we change ϵ?



- ❑ **Extreme case:** we decrease ϵ to 0, AT will converge to natural training.
- ❑ **Conclusion:** decrease the budget can improve natural accuracy.

Research gap and research question

- **Research gap:** Existing AT methods apply a fixed ϵ for all pixels in an image. Therefore, changing ϵ must sacrifice natural accuracy or robustness.
- **Research question:** Can we design an adaptive method to **reweight ϵ for only partial pixels in an image** so that we can increase natural accuracy without sacrificing robustness?
- In our recent work, we show that the answer to this question is **YES**.

Improving Accuracy-robustness Trade-off via Pixel Reweighted Adversarial Training

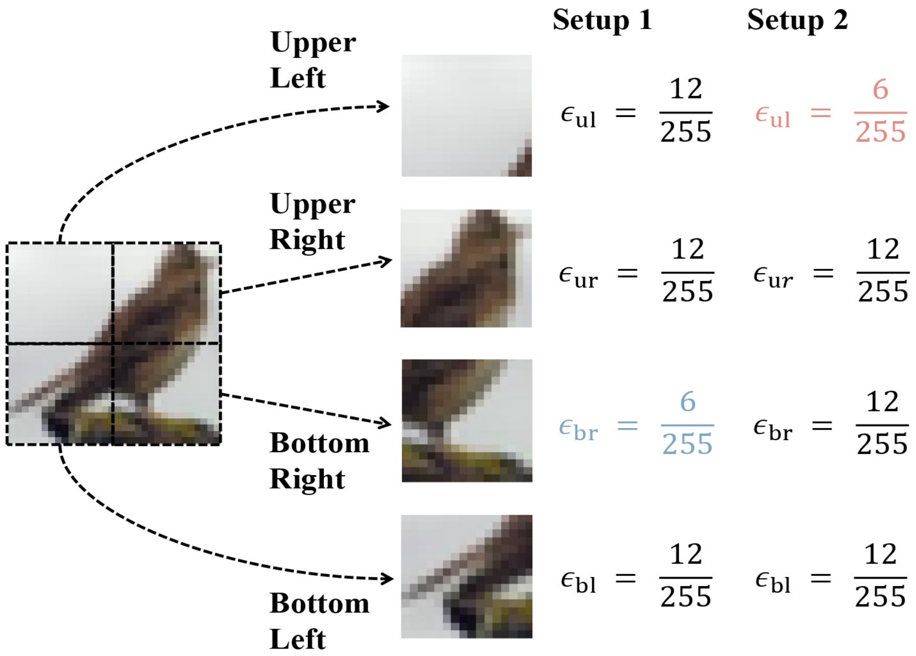
Jiacheng Zhang, Feng Liu*, Dawei Zhou, Jingfeng Zhang, Tongliang Liu*

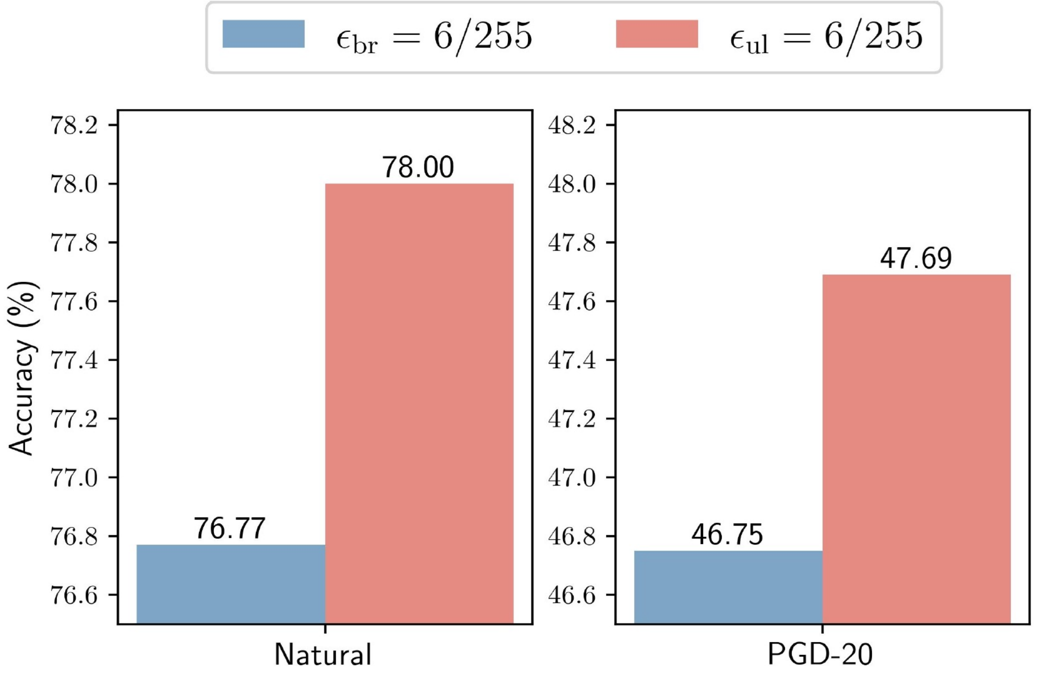
Paper

Code

Are all pixels equally important in robust classification?

- **Proof-of-concept experiment:** changing the perturbation budgets for different parts of an image has the potential to boost robustness and accuracy **at the same time**.





Pixel-reweighted AdversaRial Training (PART)

- **Proof-of-concept experiment:** changing the perturbation budgets for different parts of an image has the potential to boost robustness and accuracy **at the same time**.

↓

Pixels in an image have different importances in classification.

↓

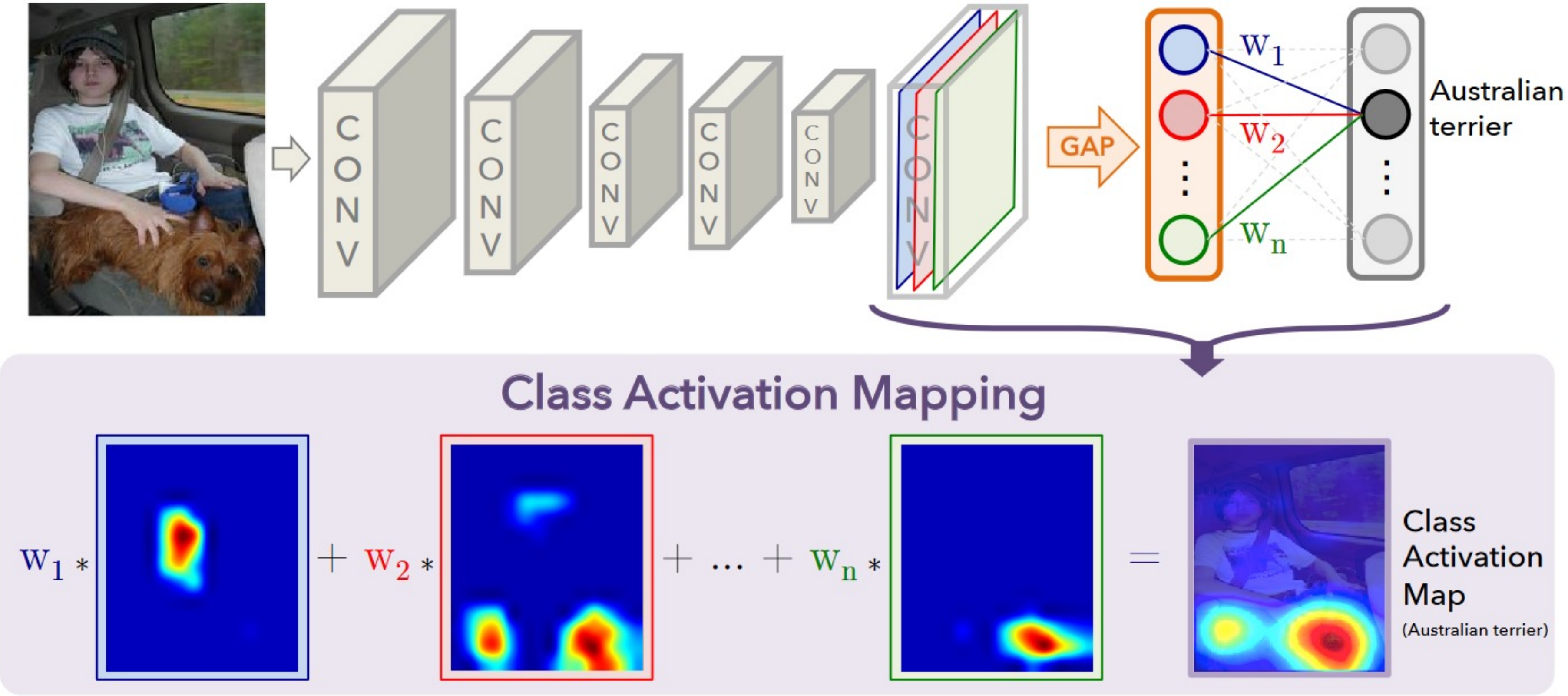
We need to guide the model to **focus more on important pixels** during training.

↓

***Class activation mapping* (CAM)** can help.

Class activation mapping (CAM)

- **CAM:** the predicted class score is mapped back to the previous convolutional layer to generate the class activation maps. The CAM **highlights the regions that influence the classification result the most**.



Learning deep features for discriminative localization. CVPR 2016.

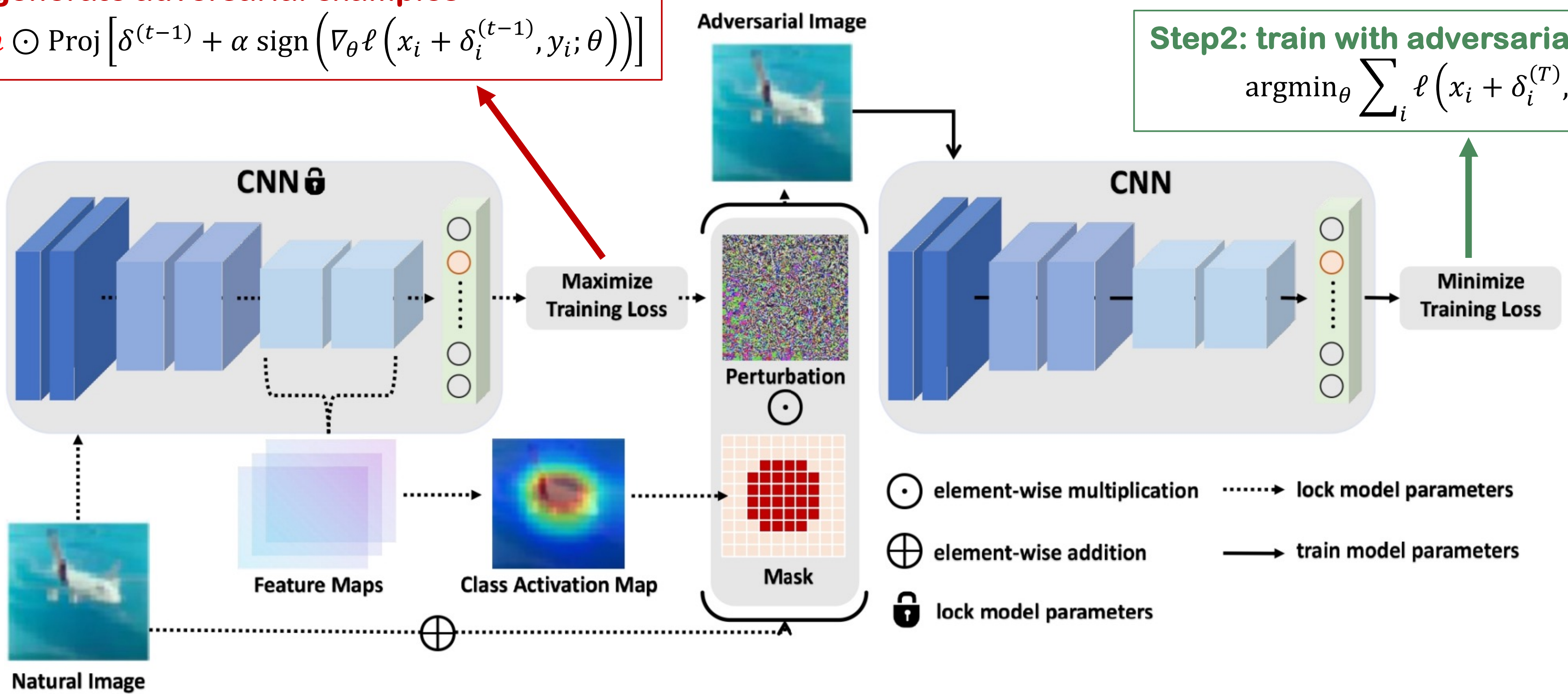
Pixel-reweighted AdversaRial Training (PART)

Step1: generate adversarial examples

$$\delta^{(t)} \leftarrow \boldsymbol{m} \odot \mathrm{Proj}\left[\delta^{(t-1)} + \alpha\, \mathrm{sign}\left(\nabla_\theta \ell\left(x_i + \delta_i^{(t-1)}, y_i; \theta\right)\right)\right]$$

Step2: train with adversarial examples

$$\mathrm{argmin}_\theta \sum_i \ell\left(x_i + \delta_i^{(T)}, y_i; \theta\right)$$



Improving Accuracy-robustness Trade-off via Pixel Reweighted Adversarial Training. ICML 2024.

Pixel-reweighted AdversaRial Training (PART)

Q: Why can PART improve natural accuracy?

A: Because PART partially reduces ϵ.

Q: Why can PART maintain robust accuracy?

A: Because PART explicitly guides the model to focus more on important pixel regions during training, making sure these regions stay robust to adversarial attacks.

Strength of PART

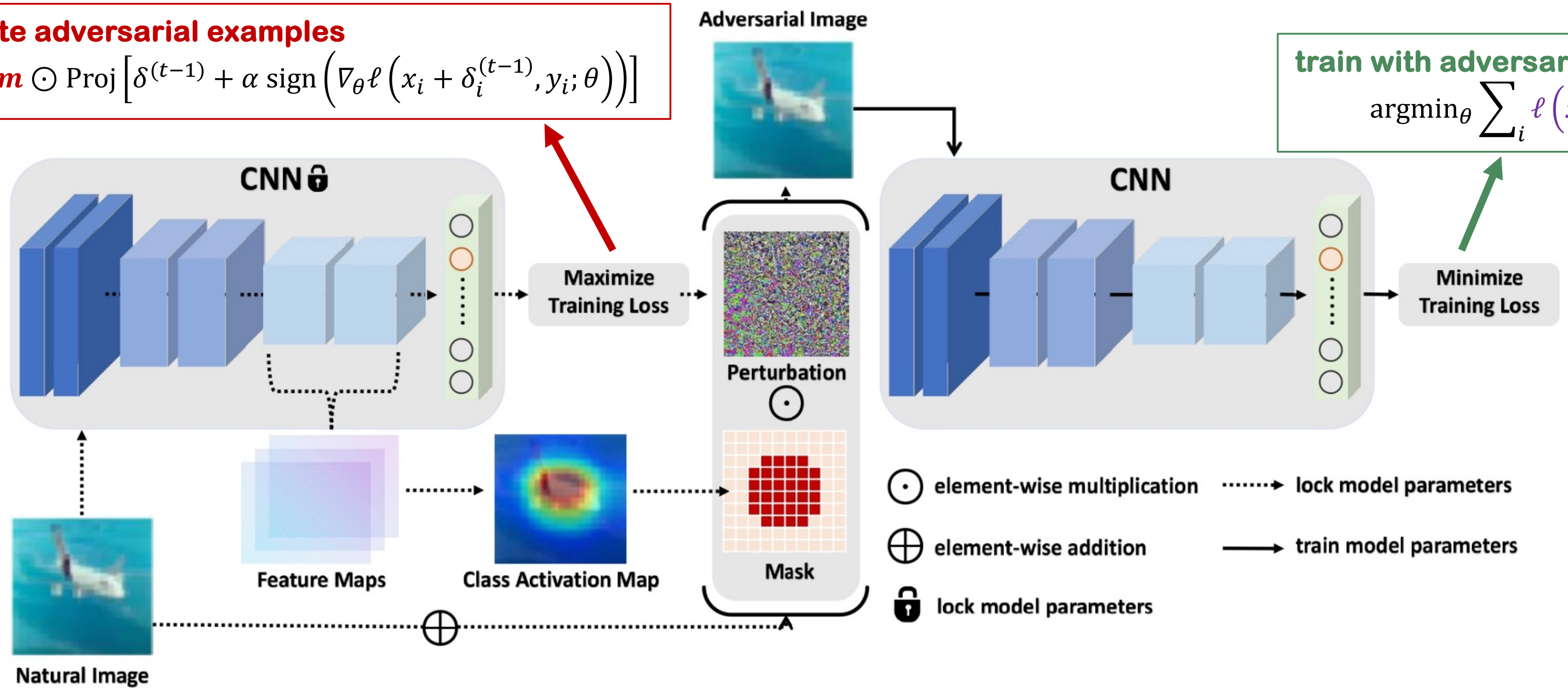
- PART is **orthogonal** to many SOTA AT methods (e.g., TRADES, MART).
- PART can be easily integrated into those methods.

generate adversarial examples

$$\delta^{(t)} \leftarrow \boldsymbol{m} \odot \mathrm{Proj}\left[\delta^{(t-1)} + \alpha \,\mathrm{sign}\left(\nabla_\theta \ell\left(x_i + \delta_i^{(t-1)}, y_i; \theta\right)\right)\right]$$

train with adversarial examples

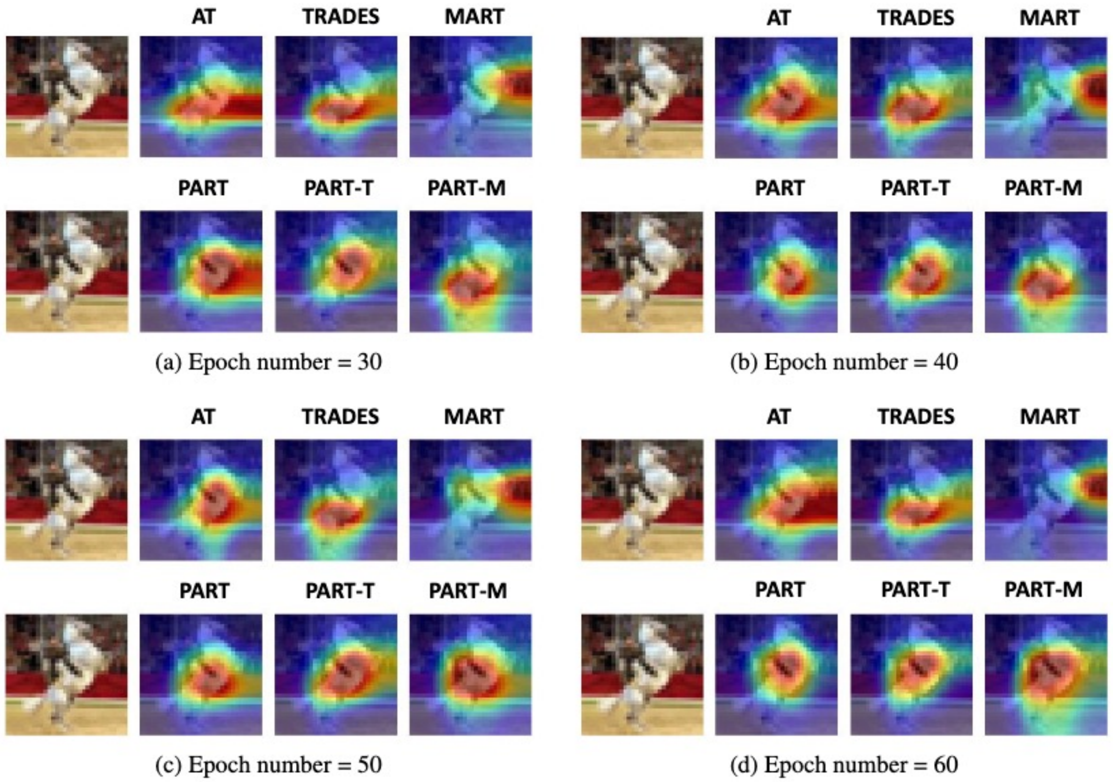
$$\mathrm{argmin}_\theta \sum_i \ell\left(\boldsymbol{x}_i + \boldsymbol{\delta}_i^{(T)}, \boldsymbol{y}_i; \boldsymbol{\theta}\right)$$

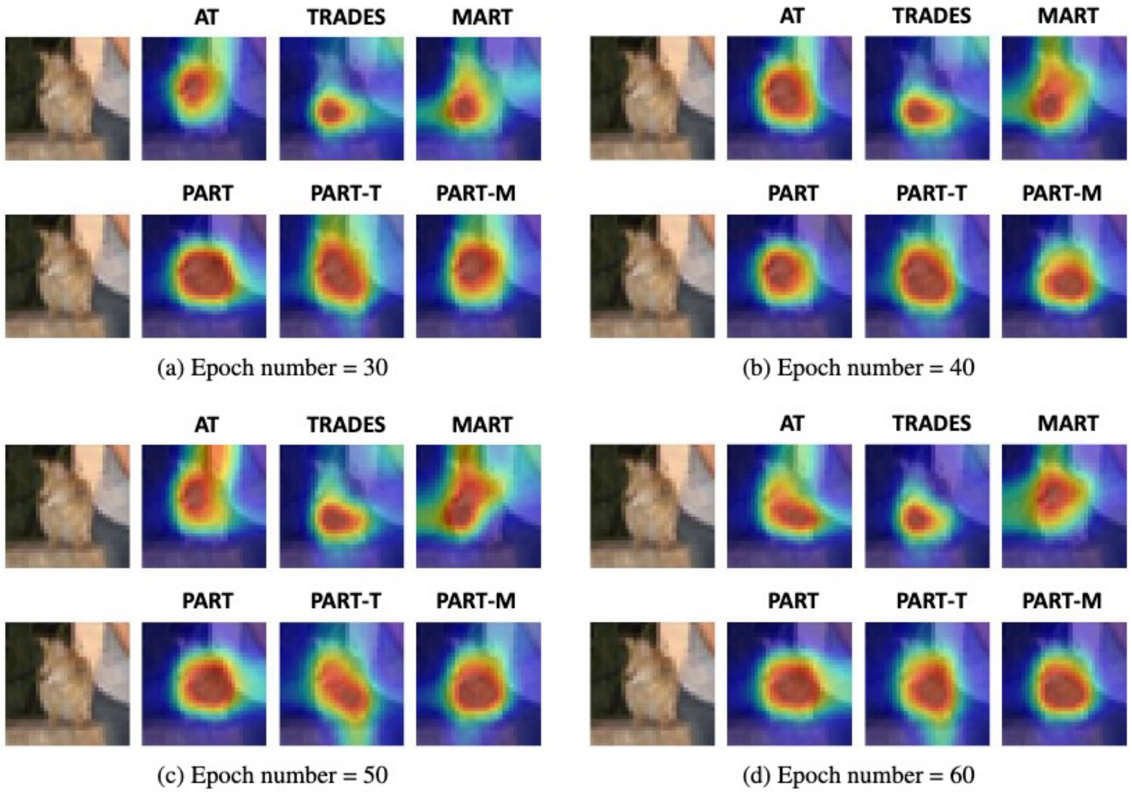


- TRADES and MART focus on improving loss functions.

Strength of PART

- PART-based methods align better with semantic information.





Improving Accuracy-robustness Trade-off via Pixel Reweighted Adversarial Training. ICML 2024.

Main results

Dataset	Method	Natural	PGD-20	MMA	AA
		ResNet-18			
CIFAR-10	AT	82.58 ± 0.14	**43.69 ± 0.28**	41.80 ± 0.10	41.63 ± 0.22
	PART ($s = 1$)	83.42 ± 0.26 (+ 0.84)	43.65 ± 0.16 (- 0.04)	**41.98 ± 0.03 (+ 0.18)**	**41.74 ± 0.04 (+ 0.11)**
	PART ($s = 10$)	**83.77 ± 0.15 (+ 1.19)**	43.36 ± 0.21 (- 0.33)	41.83 ± 0.07 (+ 0.03)	41.41 ± 0.14 (- 0.22)
	TRADES	78.16 ± 0.15	48.28 ± 0.05	45.00 ± 0.08	45.05 ± 0.12
	PART-T ($s = 1$)	79.36 ± 0.31 (+ 1.20)	**48.90 ± 0.14 (+ 0.62)**	**45.90 ± 0.07 (+ 0.90)**	**45.97 ± 0.06 (+ 0.92)**
	PART-T ($s = 10$)	**80.13 ± 0.16 (+ 1.97)**	48.72 ± 0.11 (+ 0.44)	45.59 ± 0.09 (+ 0.59)	45.60 ± 0.04 (+ 0.55)
	MART	76.82 ± 0.28	49.86 ± 0.32	45.42 ± 0.04	45.10 ± 0.06
	PART-M ($s = 1$)	78.67 ± 0.10 (+ 1.85)	**50.26 ± 0.17 (+ 0.40)**	**45.53 ± 0.05 (+ 0.11)**	**45.19 ± 0.04 (+ 0.09)**
	PART-M ($s = 10$)	**80.00 ± 0.15 (+ 3.18)**	49.71 ± 0.12 (- 0.15)	45.14 ± 0.10 (- 0.28)	44.61 ± 0.24 (- 0.49)
		ResNet-18			
SVHN	AT	91.06 ± 0.24	49.83 ± 0.13	47.68 ± 0.06	45.48 ± 0.05
	PART ($s = 1$)	93.14 ± 0.05 (+ 2.08)	**50.34 ± 0.14 (+ 0.51)**	**48.08 ± 0.09 (+ 0.40)**	**45.67 ± 0.13 (+ 0.19)**
	PART ($s = 10$)	**93.75 ± 0.07 (+ 2.69)**	50.21 ± 0.10 (+ 0.38)	48.00 ± 0.14 (+ 0.32)	45.61 ± 0.08 (+ 0.13)
	TRADES	88.91 ± 0.28	58.74 ± 0.53	53.29 ± 0.56	52.21 ± 0.47
	PART-T ($s = 1$)	91.35 ± 0.11 (+ 2.44)	**59.33 ± 0.22 (+ 0.59)**	**54.04 ± 0.16 (+ 0.75)**	**53.07 ± 0.67 (+ 0.86)**
	PART-T ($s = 10$)	**91.94 ± 0.18 (+ 3.03)**	59.01 ± 0.13 (+ 0.27)	53.80 ± 0.20 (+ 0.51)	52.61 ± 0.24 (+ 0.40)
	MART	89.76 ± 0.08	58.52 ± 0.53	52.42 ± 0.34	49.10 ± 0.23
	PART-M ($s = 1$)	91.42 ± 0.36 (+ 1.66)	**58.85 ± 0.29 (+ 0.33)**	**52.45 ± 0.03 (+ 0.03)**	**49.92 ± 0.10 (+ 0.82)**
	PART-M ($s = 10$)	**93.20 ± 0.22 (+ 3.44)**	58.41 ± 0.20 (- 0.11)	52.18 ± 0.14 (- 0.24)	49.25 ± 0.13 (+ 0.15)
		WideResNet-34-10			
TinyImagenet-200	AT	43.51 ± 0.13	11.70 ± 0.08	10.66 ± 0.11	10.53 ± 0.14
	PART ($s = 1$)	44.87 ± 0.21 (+ 1.36)	**11.93 ± 0.16 (+ 0.23)**	**10.96 ± 0.12 (+ 0.30)**	**10.76 ± 0.06 (+ 0.23)**
	PART ($s = 10$)	**45.59 ± 0.14 (+ 2.08)**	11.81 ± 0.10 (+ 0.11)	10.91 ± 0.08 (+ 0.25)	10.68 ± 0.10 (+ 0.15)
	TRADES	43.05 ± 0.15	13.86 ± 0.10	12.62 ± 0.16	12.55 ± 0.09
	PART-T ($s = 1$)	44.31 ± 0.12 (+ 1.26)	**14.08 ± 0.22 (+ 0.22)**	**13.01 ± 0.09 (+ 0.39)**	**12.84 ± 0.14 (+ 0.29)**
	PART-T ($s = 10$)	**45.16 ± 0.10 (+ 2.11)**	13.98 ± 0.15 (+ 0.12)	12.88 ± 0.12 (+ 0.26)	12.72 ± 0.08 (+ 0.17)
	MART	42.68 ± 0.22	14.77 ± 0.18	13.58 ± 0.13	13.42 ± 0.16
	PART-M ($s = 1$)	43.75 ± 0.24 (+ 1.07)	**14.93 ± 0.15 (+ 0.16)**	**13.76 ± 0.06 (+ 0.18)**	**13.68 ± 0.13 (+ 0.24)**
	PART-M ($s = 10$)	**45.02 ± 0.16 (+ 2.34)**	14.65 ± 0.14 (- 0.12)	13.41 ± 0.11 (- 0.17)	13.37 ± 0.15 (- 0.05)

- ❑ Our method can achieve a notable improvement in accuracy-robustness trade-off.
- ❑ In most cases, our method can improve natural accuracy by a notable margin without sacrificing robustness.

Main results

ResNet-18						
Corruption	AT	PART ($s = 10$)	TRADES	PART-T ($s = 10$)	MART	PART-M ($s = 10$)
Gaussian Noise	81.05	**82.46**	76.05	**79.42**	77.57	**79.51**
Shot Noise	81.11	**82.83**	75.91	**79.70**	77.66	**79.74**
Impulse Noise	79.42	**80.76**	74.59	**78.03**	76.12	**78.16**
Speckle Noise	81.42	**82.68**	75.97	**79.68**	77.63	**79.79**
Defocus Blur	81.07	**82.48**	76.06	**79.42**	77.59	**79.50**
Glass Blur	77.82	**78.20**	72.60	**76.26**	74.17	**76.57**
Motion Blur	79.30	**80.13**	74.28	**77.43**	75.35	**77.45**
Zoom Blur	78.87	**79.30**	73.27	**76.74**	74.10	**76.74**
Gaussian Blur	81.05	**82.46**	76.05	**79.42**	77.57	**79.51**
Snow	81.09	**82.01**	76.13	**79.43**	77.63	**79.34**
Frost	78.96	**80.04**	73.90	**76.60**	75.01	**75.80**
Fog	79.34	**80.18**	72.95	**77.15**	75.29	**78.26**
Brightness	81.89	**83.22**	76.87	**80.16**	78.60	**80.26**
Spatter	81.03	**82.03**	75.81	**79.31**	77.62	**79.39**
Contrast	77.09	**77.67**	70.08	**75.00**	71.90	**75.81**
Elastic Transform	77.32	**78.16**	71.99	**75.39**	73.13	**75.68**
Pixelate	81.09	**82.63**	76.05	**79.48**	77.45	**79.43**
JPEG Compression	80.50	**81.91**	75.71	**79.09**	77.15	**79.29**
Saturate	78.01	**79.34**	73.59	**76.34**	74.70	**75.52**

- On corrupted dataset (e.g., CIFAR-10-C), our method can outperform baseline methods by a notable margin.
- The main reason is that the model can still figure out important parts of an image even if an image is corrupted.

Improving Accuracy-robustness Trade-off via Pixel Reweighted Adversarial Training. ICML 2024.

Conclusion and future work

- **Key message we want to deliver in this paper:** Guiding the model to focus more on essential pixel regions during training can help improve the generalizability (e.g., accuracy-robustness trade-off) of vision models.

Questions?

Thank you