ETH zürich
Universität Zürich[UZH]
Department of
Computer Science

Towards Compositional Adversarial Robustness: Generalizing Adversarial Training to Composite Semantic Perturbations

Alexandre Elsig
11. March 2025, Zürich

What is the goal of adversarial attacks?

To fool a model into making a mistake

Type of adversarial attacks

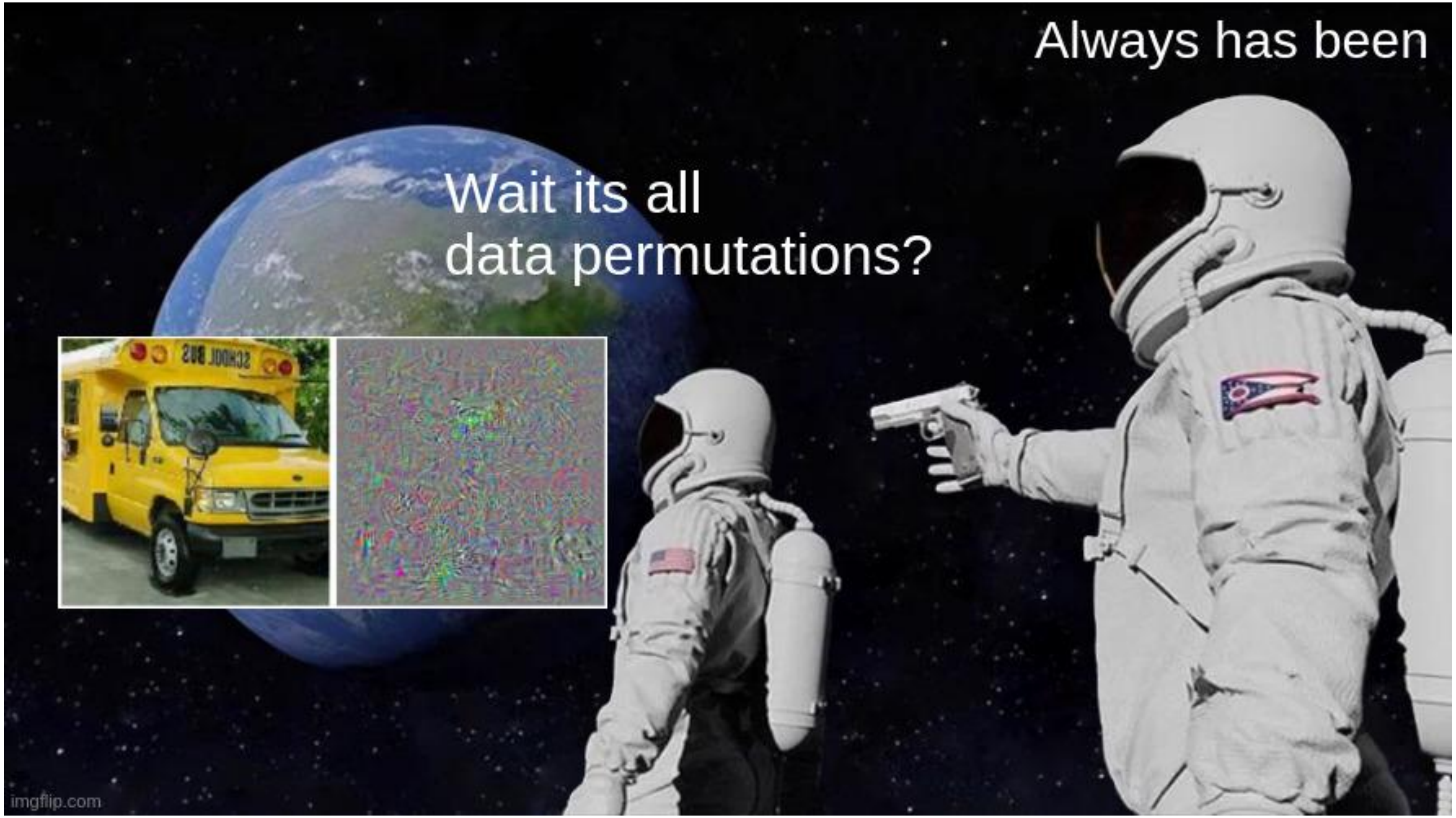
- Black-box vs. White-box
- Poisoning Attacks
- **Evasion Attacks**

Sparse vs Dense attacks

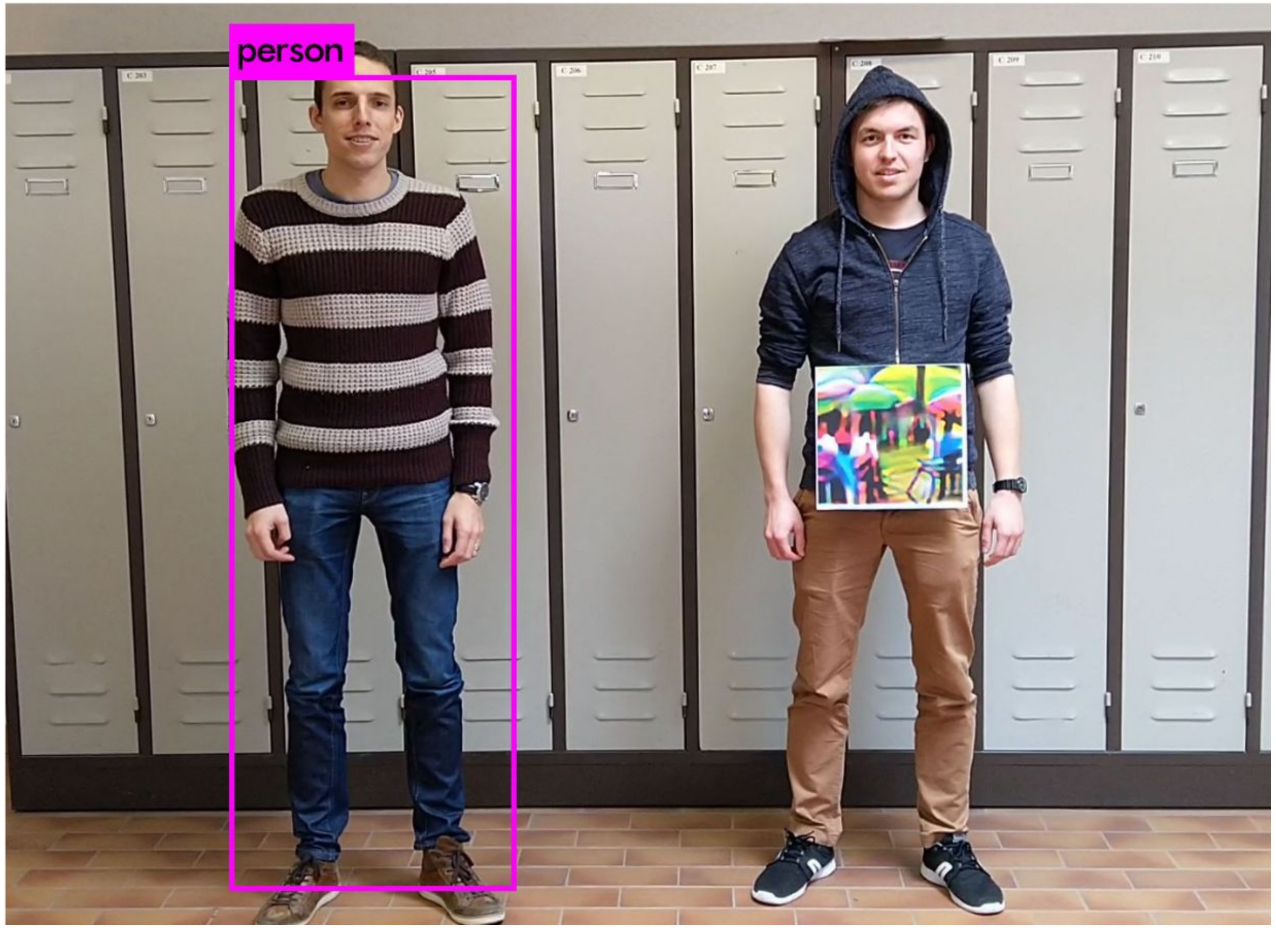
- Modas, A., Moosavi-Dezfooli, S.-M., and Frossard, P. Sparsefool: a few pixels make a big difference. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (2019), pp. 9087–9096.

- Szegedy, C. Intriguing properties of neural networks. arXiv preprint arXiv:1312.6199 (2013).

Is it really that simple?

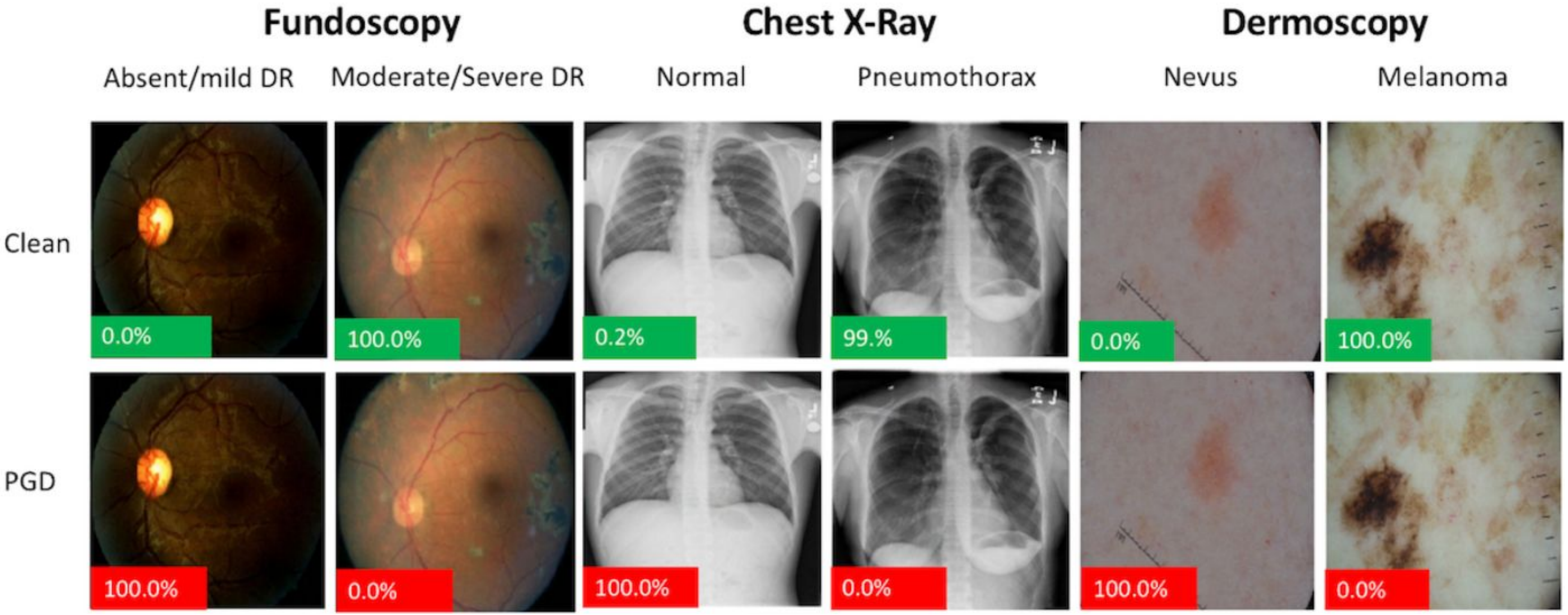


Adversarial Attacks



- Thys, S., Van Ranst, W., and Goedemé, T. Fooling automated surveillance cameras: adversarial patches to attack person detection. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops (2019), pp. 0–0.

Medical Adversarial attack



- Finlayson, S. G., Bowers, J. D., Ito, J., Zittrain, J. L., Beam, A. L., and Kohane, I. S. Adversarial attacks on medical machine learning. Science 363, 6433 (2019), 1287–1289

Adversarial Attacks

- Eykholt, K., Evtimov, I., Fernandes, E., Li, B., Rahmati, A., Xiao, C., Prakash, A., Kohno, T., and Song, D. Robust physical-world attacks on deep learning visual classification. In Proceedings of the IEEE conference on computer vision and pattern recognition (2018), pp. 1625–1634.

Other Attacks

- Malware detection
- Phishing detection
- Fraud detection
- Automated voice authentication for e.g. banks

So what can be done?

- Gradient Masking
- Adversarial Detection
- **Adversarial Training**

Previous seminal Work Madry et. al.

What had been done so far?

- Single attacks
- focus on Evasion Attacks

Idea behind PGD attacks

Idea: find (smallest) perturbation through Projected Gradient Descent (PGD) that makes model misclassify an example.

Projected Gradient Descent (PGD)

- Iterative Nature
- Fine-grained Control
- Evasion of Defenses
- Flexibility

How do we balance Robustness and Performance?

- Saddle Point Problem

$$\min_{\theta} \rho(\theta),$$

D: data dist.
S: Perturbation dist.

*Madry et. al. Towards Deep Learning Models Resistant to Adversarial Attacks

Previous achievements

- saddle-point model
- good robustness against adversaries
  - 86% accuracy on MNIST FGSM (1 gradient step)
  - 46% accuracy against PGD (~10 smaller gradient steps) on CIFAR-10

Issues with the current approaches

- Only norm-bounded attacks (PGD) considered
- Need Gradients
- Real world is complex
- Combined attacks not considered
- (Computationally expensive)

Generalized Adversarial Training

Generalized Adversarial Training (GAT)

Motivation:

- model Real-world adversarial threats

Core Ideas:

- Composite Adversarial Attacks (CAA)

- Train on CAAs to improve generalizability

How is GAT Implemened?

Attacks:
- Hue
- Saturation
- Rotation
- Brightness
- Contrast

GAT: Perturbations

All perturbations operate in a range ϵ specific to the type of perturbation

Perturb the current image by applying some attack

Modify Image by perturbation $\boldsymbol{\delta}$ ~U(ϵ)

GAT: Hue

GAT: Saturation

Saturation

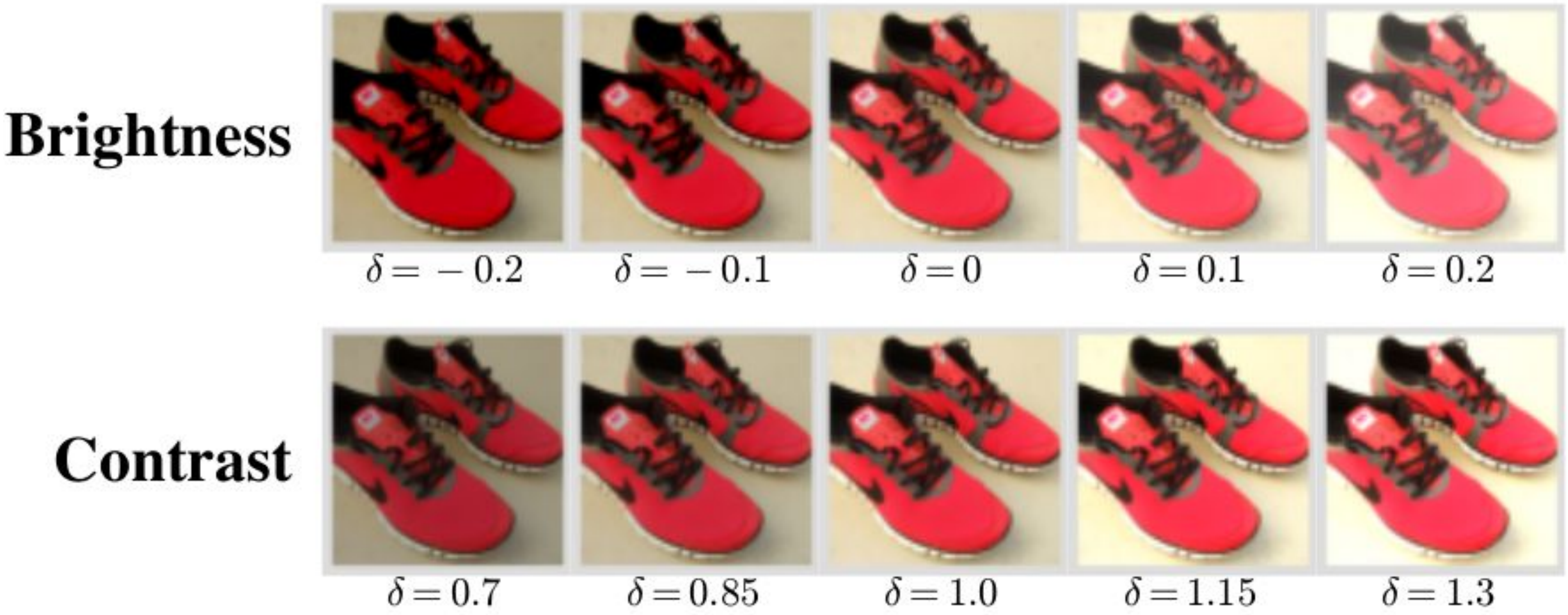
$\delta = 0.7$ $\delta = 0.85$ $\delta = 1.0$ $\delta = 1.15$ $\delta = 1.3$

GAT: Rotation

Rotation

$\delta = -10°$ $\delta = -5°$ $\delta = 0°$ $\delta = 5°$ $\delta = 10°$

GAT: Brightness & Contrast

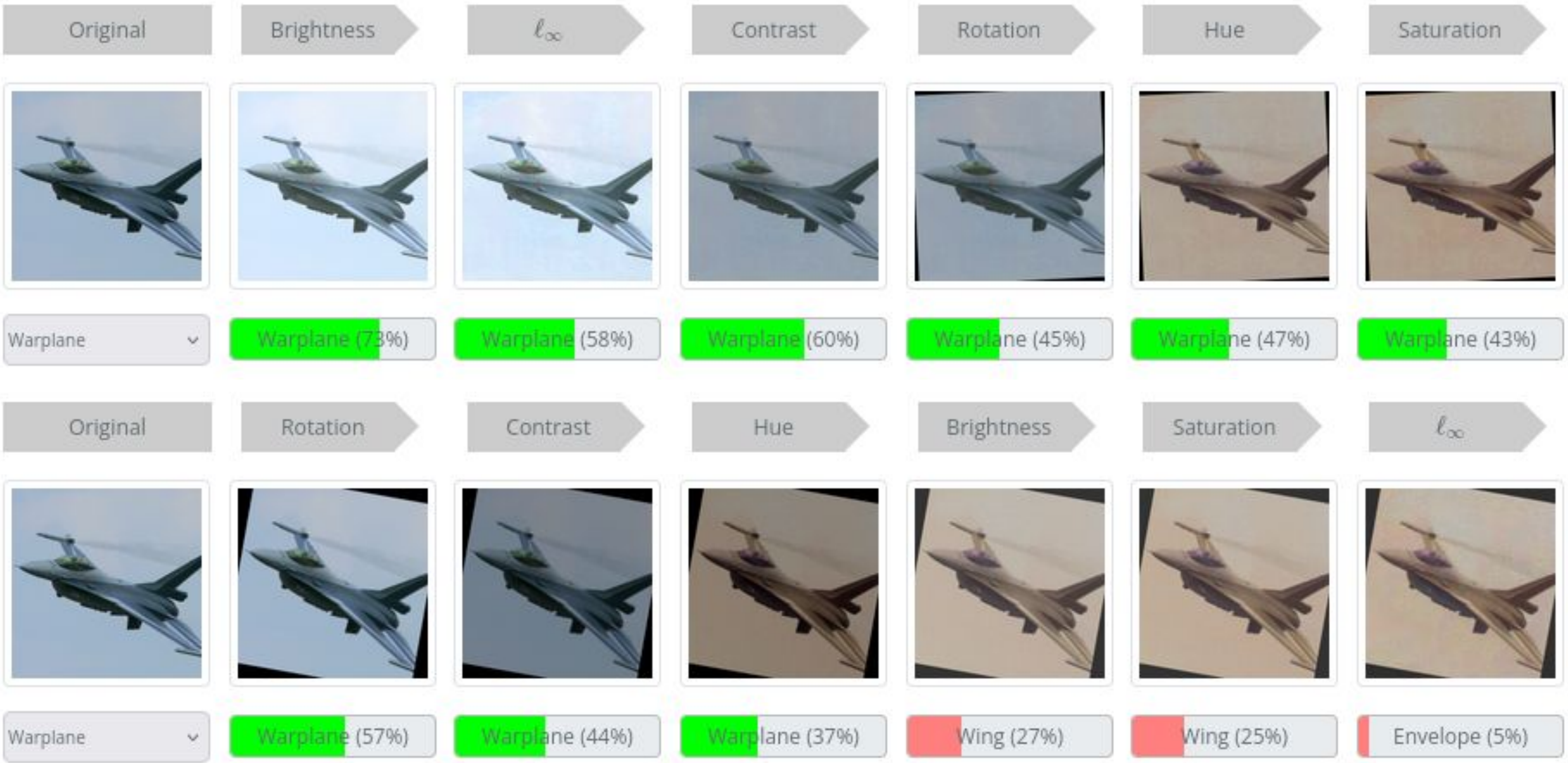


How is GAT Implemened?

Attacks:
- Hue
- Saturation
- Rotation
- Brightness
- Contrast

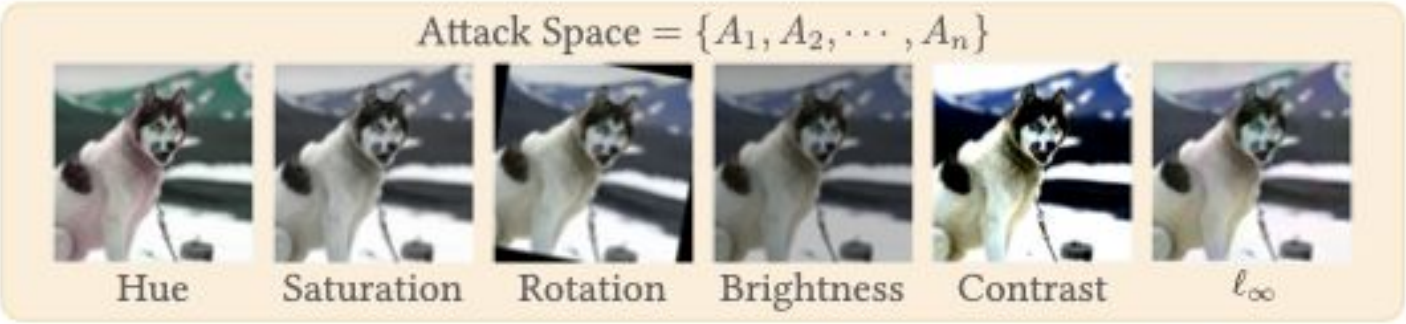
\+ **Attack Scheduling**

Why does Attack Scheduling matter?



- Lei Hsiung and Yun-Yun Tsai and Pin-Yu Chen and Tsung-Yi Ho,Towards Compositional Adversarial Robustness: Generalizing Adversarial Training to Composite Semantic Perturbations

Perturbation Pipeline



- Lei Hsiung and Yun-Yun Tsai and Pin-Yu Chen and Tsung-Yi Ho,Towards Compositional Adversarial Robustness: Generalizing Adversarial Training to Composite Semantic Perturbations

Updated Saddle Point Problem

Loss function

Ground Truth

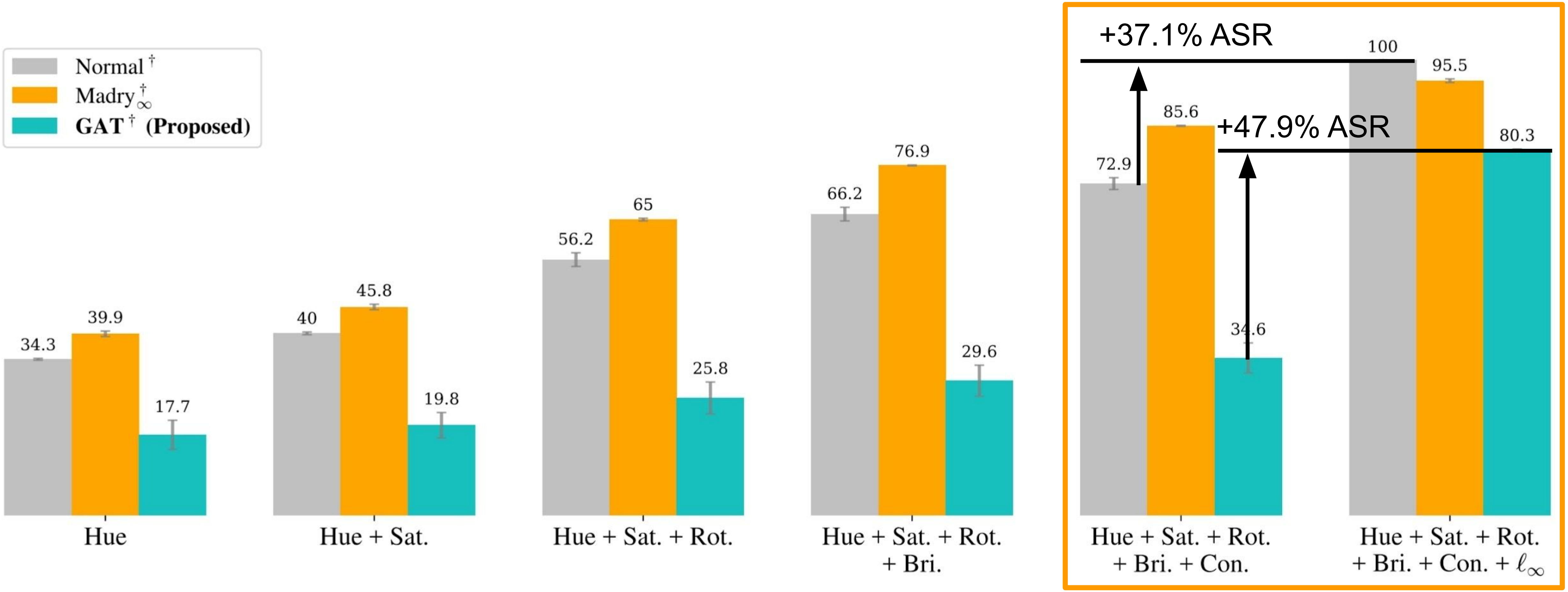
$$\boxed{\min_{\theta_{\mathcal{F}}}} \mathbb{E}_{(x,y)\sim\mathcal{D}} \boxed{\left[\max_{x_{\text{c-adv}} \in \mathcal{B}(x;\Omega;E)} \boxed{\mathcal{L}(}\boxed{\mathcal{F}(x_{\text{c-adv}})},\boxed{y)} \right]}.$$

Model Prediction

Optimize model parameters to minimize the expected (worst case) loss function

Loss function worst case using the optimized adversarial images

Results Attack Success Rate (ASR%)



Results on Cifar-10

f = all attacks seen in training   fs = f + optimal attack scheduling

	Training	Clean	Three attacks CAA_{3a}	Three attacks CAA_{3b}	Three attacks CAA_{3c}	Semantic attacks Rand.	Semantic attacks Sched.	Full attacks Rand.	Full attacks Sched.
ResNet-50	Normal†	95.2	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	59.7 ± 0.2	44.2 ± 0.5	0.0 ± 0.0	0.0 ± 0.0
	Madry$^\dagger_\infty$	87.0	30.8 ± 0.2	18.8 ± 0.5	19.1 ± 0.3	31.5 ± 0.2	21.3 ± 0.2	10.8 ± 0.2	3.7 ± 0.2
	PAT$^\dagger_{self}$	82.4	20.9 ± 0.1	11.9 ± 0.5	17.9 ± 0.3	28.9 ± 0.3	17.5 ± 0.3	9.1 ± 0.3	2.5 ± 0.3
	PAT$^\dagger_{alex}$	71.6	20.7 ± 0.3	12.5 ± 0.2	16.5 ± 0.4	23.4 ± 0.3	12.2 ± 0.4	10.3 ± 0.1	2.5 ± 0.2
	GAT-f†	**82.3**	$\mathbf{39.9 \pm 0.1}$	$\mathbf{33.3 \pm 0.1}$	$\mathbf{28.9 \pm 0.2}$	$\mathbf{69.9 \pm 0.1}$	$\mathbf{66.0 \pm 0.1}$	$\mathbf{30.0 \pm 0.4}$	$\mathbf{18.8 \pm 0.3}$
	GAT-fs†	**82.1**	$\mathbf{43.5 \pm 0.1}$	$\mathbf{36.6 \pm 0.1}$	$\mathbf{32.5 \pm 0.1}$	$\mathbf{69.9 \pm 0.2}$	$\mathbf{66.6 \pm 0.1}$	$\mathbf{32.3 \pm 0.8}$	$\mathbf{21.8 \pm 0.3}$
WideResNet-34	Normal*	94.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	46.0 ± 0.4	29.9 ± 0.5	0.0 ± 0.0	0.0 ± 0.0
	Trades$^*_\infty$	84.9	30.0 ± 0.3	19.8 ± 0.6	10.1 ± 0.5	16.6 ± 0.2	8.1 ± 0.5	5.8 ± 0.3	1.5 ± 0.2
	FAT$^*_\infty$	88.1	29.8 ± 0.4	17.1 ± 0.4	12.8 ± 0.6	18.7 ± 0.2	9.8 ± 0.5	6.1 ± 0.1	1.5 ± 0.1
	AWP$^*_\infty$	85.4	34.2 ± 0.2	23.2 ± 0.2	11.1 ± 0.4	15.6 ± 0.2	7.9 ± 0.2	5.9 ± 0.0	1.7 ± 0.2
	GAT-f*	**83.4**	$\mathbf{40.2 \pm 0.1}$	$\mathbf{34.0 \pm 0.1}$	$\mathbf{30.7 \pm 0.4}$	$\mathbf{71.6 \pm 0.1}$	$\mathbf{67.8 \pm 0.2}$	$\mathbf{31.2 \pm 0.4}$	$\mathbf{20.1 \pm 0.3}$
	GAT-fs*	**83.2**	$\mathbf{43.5 \pm 0.1}$	$\mathbf{36.3 \pm 0.1}$	$\mathbf{32.9 \pm 0.4}$	$\mathbf{70.5 \pm 0.1}$	$\mathbf{66.7 \pm 0.3}$	$\mathbf{32.2 \pm 0.7}$	$\mathbf{21.9 \pm 0.7}$

Table 1. Comparison of accuracy (%) on CIFAR-10. We combine different types of three attacks (CAA_3) with scheduled ordering: CAA_{3a}: (Hue, Saturation, ℓ_∞), CAA_{3b}: (Hue, Rotation, ℓ_∞), CAA_{3c}: (Brightness, Contrast, ℓ_∞), on CIFAR-10

So is the problem solved?

Evaluation & critique

Strengths:
- Addressed an important gap in Adversarial Robustness
- Expanded beyond simple pixel & norm perturbations

Weaknesses:
- Computationally expensive (multiple perturbed versions of each image!)
- Might not generalize to unseen perturbations (my experiment)
- No theoretical guarantees on robustness improvement.

Critique of the claims

- Do their models really generalize better?
- What about their comparisons?

Single perturbation Examples

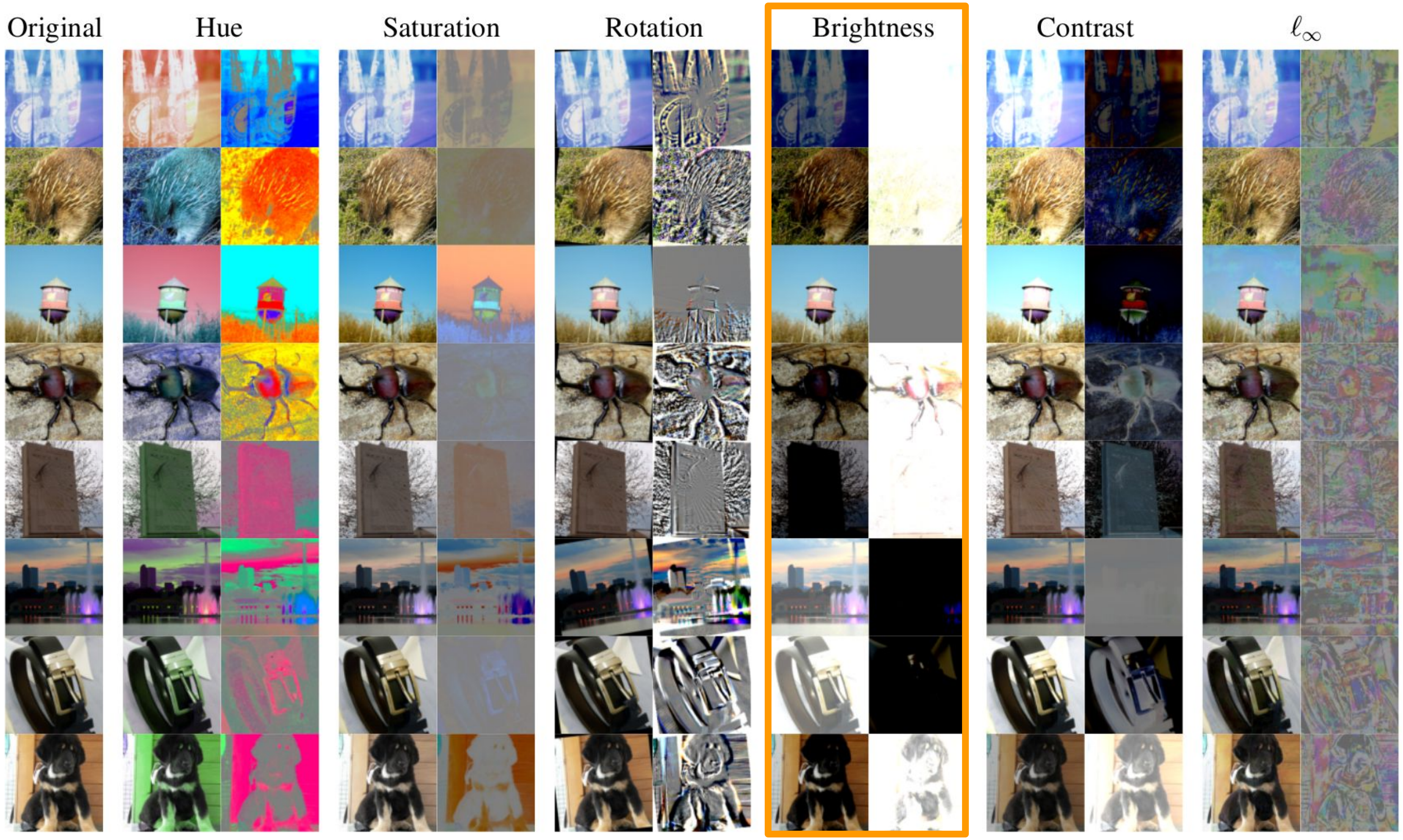


Figure A6. Adversarial examples generated under **single semantic** attacks or ℓ_∞ attack.

Dual Perturbation Examples

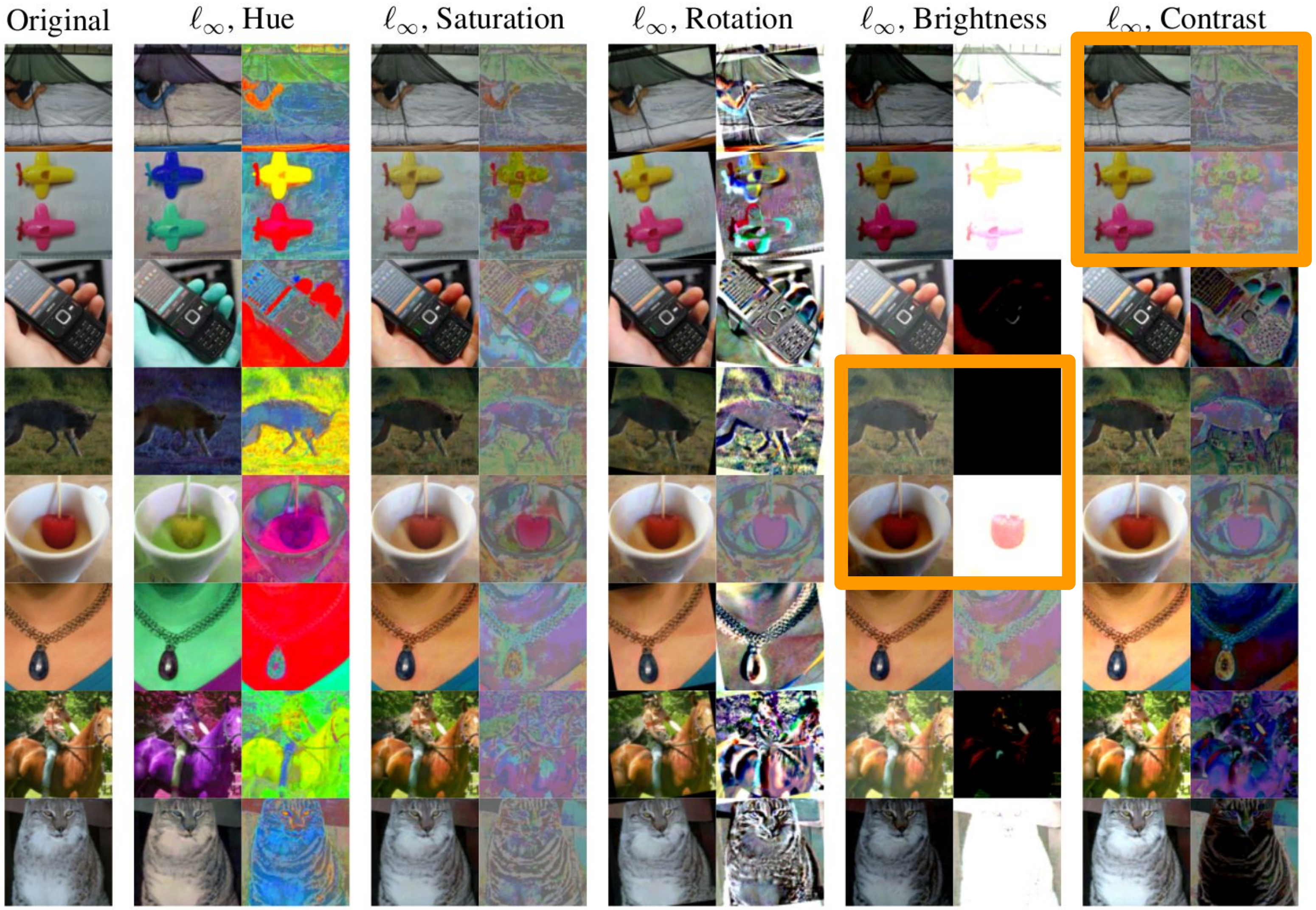


Figure A7. Composite adversarial examples generated under **two attacks** (composed of one semantic attack and the ℓ_∞ attack).

My Experiment

Comparison on PGD Attack

- Madry:
  - Robust Accuracy: 43%
- GAT:
  - Robust Accuracy: 0.076%
  -

Discussion

Comparison of the Min-Max Functions

$$\min_{\theta} \rho(\theta), \quad \text{where} \quad \rho(\theta) = \mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\max_{\delta\in\mathcal{S}} L(\theta, x+\delta, y)\right] .$$

$$\min_{\theta_{\mathcal{F}}} \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}\left[\max_{x_{\text{c-adv}}\in\mathcal{B}(x;\Omega;E)} \mathcal{L}(\mathcal{F}(x_{\text{c-adv}}), y)\right].$$

Experiments

Remove 1 of the perturbations e.g. hue / rotation from the training set-up

See how robust it is towards these attacks generate examples with imagenet images

Change optimizer

Plot training batch to double check images

Cifar10 150 epochs resnet50 Gat-FS

C.a. 1 min per slide

BACKUP SLIDES!

GAT: Hue

$$\epsilon_H = [-\pi, \pi],\ x_H^t = \mathrm{Hue}(x_{c-adv}^t) = \mathrm{clip}_{[0,2\pi]}(x_H + \delta_H^t)$$

GAT: Saturation

Saturation

$\delta = 0.7$ $\delta = 0.85$ $\delta = 1.0$ $\delta = 1.15$ $\delta = 1.3$

$$\epsilon_S = [0, \infty],\ x_S^t = \mathrm{Sat}(x_{c-adv}^t) = \mathrm{clip}_{[0,1]}(x_S + \delta_S^t)$$

GAT: Rotation

Rotation

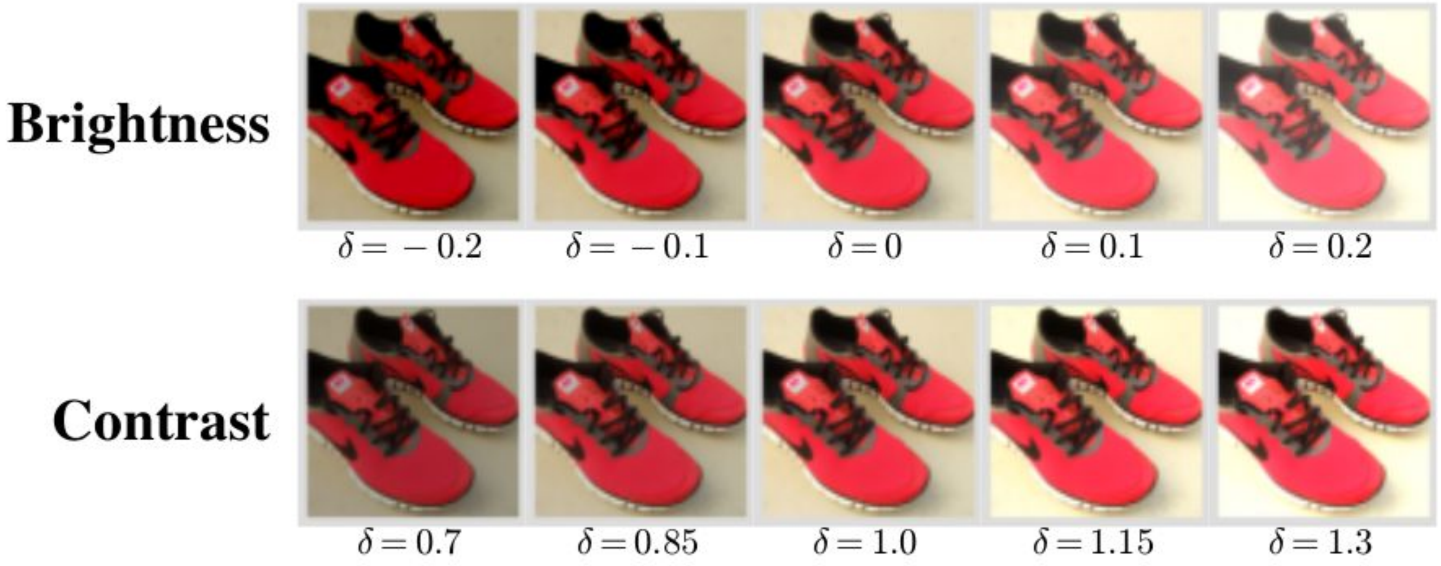
$\delta = -10°$ $\delta = -5°$ $\delta = 0°$ $\delta = 5°$ $\delta = 10°$

Rotate around center

$$\epsilon_R = [\alpha_R, \beta_R],\ \alpha_R \leq \beta_R \in \mathbb{R}$$

$$\begin{bmatrix} i' \\ j' \end{bmatrix} = \begin{bmatrix} \cos\theta \cdot i + \sin\theta \cdot j + (1 - \cos\theta) \cdot c - \sin\theta \cdot c \\ -\sin\theta \cdot i + \cos\theta \cdot j + \sin\theta \cdot c + (1 - \cos\theta) \cdot c \end{bmatrix}.$$

GAT: Brightness & Contrast



$$\epsilon_B = [\alpha_B, \beta_B], -1 \leq \alpha_B \leq \beta_B \leq 1 \text{ and } \epsilon_C = [\alpha_C, \beta_C], 0 \leq \alpha_C \leq \beta_C < \infty,$$

$$x^t_{\text{c-adv}} = \text{clip}_{[0,1]}(x + \delta^t_B) \;\text{ and }\; x^t_{\text{c-adv}} = \text{clip}_{[0,1]}(x \cdot \delta^t_C).$$

Finding the optimal Schedule

Schedule Attacks:

$$x_{\text{c-adv}} = A_{\pi_i(n)}(A_{\pi_i(n-1)}(\cdots A_{\pi_i(1)}(x))).$$

Optimizing the attack schedule

$$x_{\text{c-adv}} = A_{\pi_i(n)}(A_{\pi_i(n-1)}(\cdots A_{\pi_i(1)}(x))).$$

$$\arg\max_{\delta_k \in \epsilon_k} \mathcal{L}(\mathcal{F}(A_k(x;\delta_k)), y),$$

$$\max_{\pi} \mathcal{L}(\mathcal{F}(A_{\pi(n)}(\cdots A_{\pi(1)}(x;\delta_{\pi(1)})\cdots;\delta_{\pi(n)})), y).$$

Scheduling as a Scheduling Matrix

$\pi_i\ (\cdot)$ is essentially a permutation matrix and can optimize it by treating it as a (relaxed) scheduling matrix $\mathbf{Z^i}$

$$Z^i = \left[\mathbf{z}_1, \ldots, \mathbf{z}_n\right]^\top$$

Where $\mathbf{Z^i}$ is doubly stochastic:

$$\mathbf{z}_j \in \mathbb{R}^n, \sum_i z_{ij} = \sum_j z_{ij} = 1, \forall\, i, j \in \{1, \ldots, n\}.$$

Surrogate Image for scheduling optimization

- 1 Permutation per Iteration
- Get a surrogate image to compute the loss of each update

Optimize the scheduling matrix using:

$$\mathcal{L}(\mathcal{F}(x_{\text{surr}}), y).$$

Surrogate Image for scheduling optimization

- 1 Permutation per Iteration
- Get a surrogate image to compute the loss of each update

Optimize the scheduling matrix using:

$$\mathcal{L}(\mathcal{F}(x_{\text{surr}}), y).$$

Surrogate image at each iteration is defined as:

$$x_{\text{surr}}^{i} = \sum_{j=1}^{n} z_{ij} \cdot A_j(x_{\text{surr}}^{i-1}; \delta_j)), \forall i \in \{1, \ldots, n\},$$

Optimizing the attack schedule

$$\max_{\pi} \mathcal{L}(\mathcal{F}(A_{\pi(n)}(\cdots A_{\pi(1)}(x; \delta_{\pi(1)}) \cdots ; \delta_{\pi(n)})), y).$$

$$Z^t = \mathcal{S}\Big(\exp(Z^{t-1} + \frac{\partial \mathcal{L}(\mathcal{F}(x_{\text{surr}}), y)}{\partial Z^{t-1}})\Big),$$

Optimizing the attack schedule

$$\max_{\pi} \mathcal{L}(\mathcal{F}(A_{\pi(n)}(\cdots A_{\pi(1)}(x;\delta_{\pi(1)})\cdots;\delta_{\pi(n)})),y).$$

$$\pi_t(j) := \arg\max \mathbf{z}_j, \forall j \in \{1,\ldots,n\}.$$

GAT: Component-wise Projected Gradient Descent (Comp-PGD)

$$\delta_k^{t+1} = \text{clip}_{\epsilon_k}\left(\delta_k^t + \alpha \cdot \text{sign}(\nabla_{\delta_k^t}\mathcal{L}(\mathcal{F}(A_k(x;\delta_k^t)), y))\right),$$

Hier steht der Folientitel

- Textplatzhalter haben als Standard Bulletpoints. Um diese zu entfernen drücken Sie vor der Texteingabe die Backspace Taste.