

DiffHammer

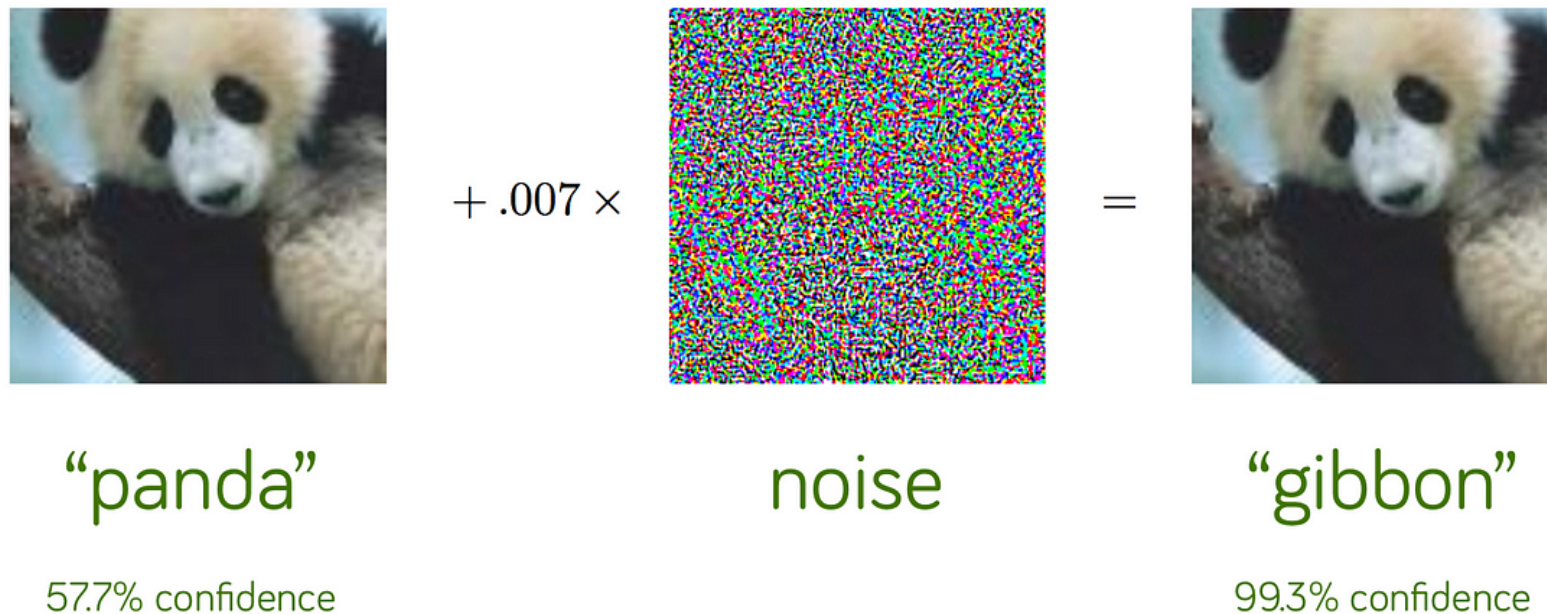
Rethinking the Robustness of Diffusion-Based Adversarial Purification

Kaibo Wang¹, Xiaowen Fu¹, Yuxuan Han¹, Yang Xiang^{1,2}

1. Department of Mathematics, The Hong Kong University of Science and Technology

2. HKUST Shenzhen-Hong Kong Collaborative Innovation Research Institute

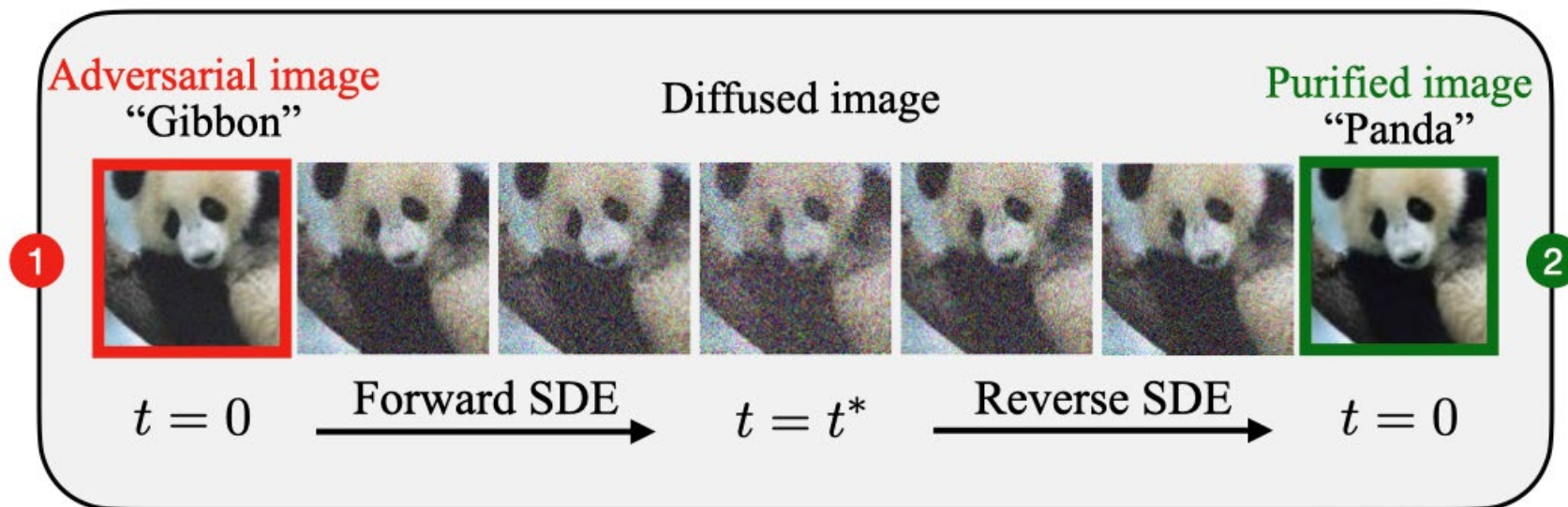
Adversarial attack in DNNs



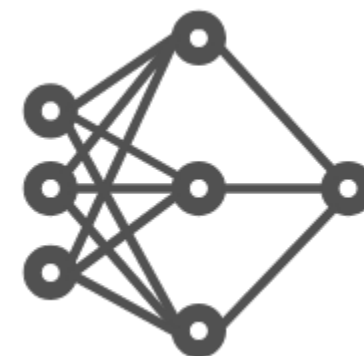
Adversarial samples hinder the application of DNNs in the security-critical domain

Diffusion-based purification

Purification



Classifier



Diffusion-based purification demonstrated impressive robustness

Challenges in evaluation

Diffusion: Iterative Complex

Stochastic

$$grad = \frac{1}{N} (grad_1 + \cdots + grad_N) \rightarrow x_{adv} \rightarrow \checkmark \checkmark \checkmark \times \checkmark$$

Time-consuming Gradient dilemma

Resubmit risk

EOT Attack with 1-evaluation

Inherent robustness or insufficient evaluation?

Wish list in DiffHammer

❖ Effective



Selective attack

Avoid gradient dilemma

❖ Efficient



Gradient grafting

Efficient gradient aggregation

❖ Comprehensive



N-evaluation

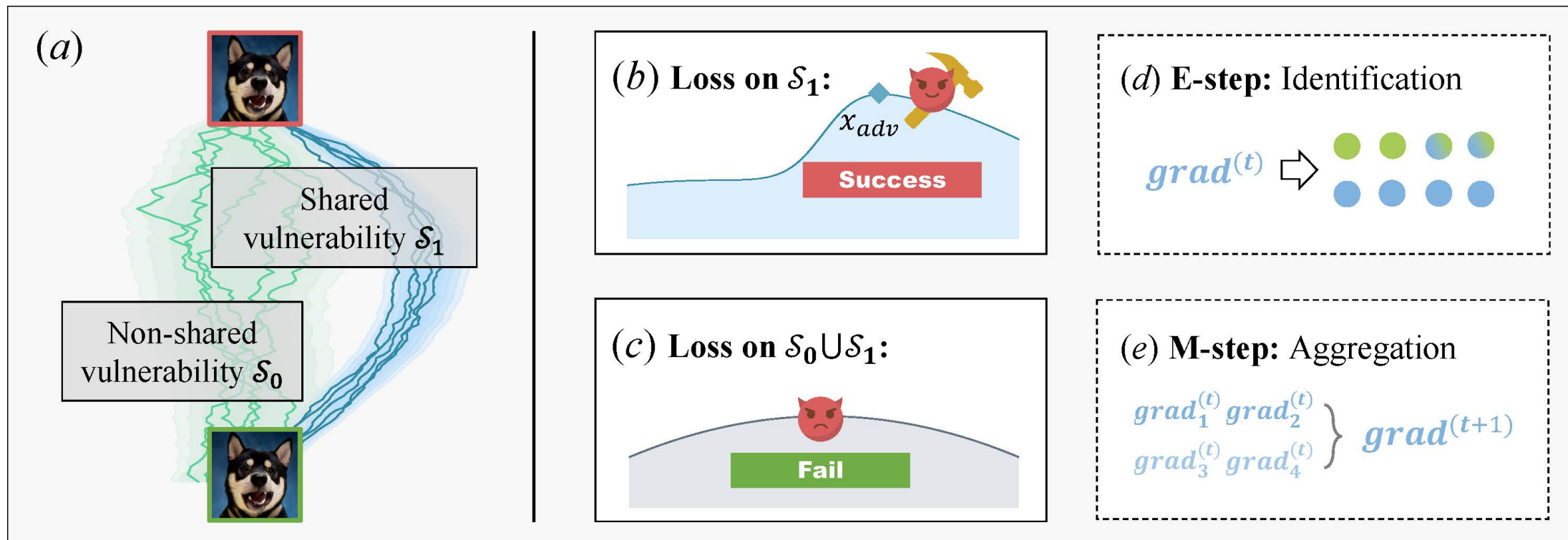
Diagnosis of resubmit risk

Attack side

Evaluation side

DiffHammer: Adaptive attack for diffusion based purification

Selective attack



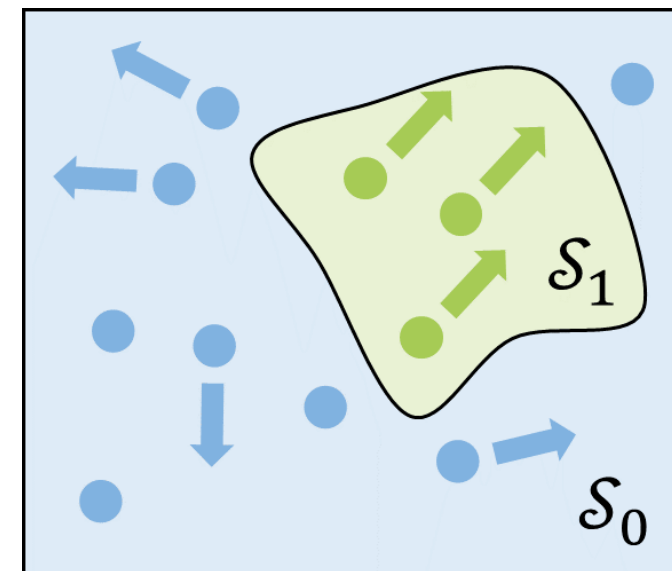
Attacks toward $\mathcal{S}_0$ are unhelpful and even detrimental

Attack on vulnerable set

Target $\mathcal{S}_1$: **Largest** set of purifications attacked by a **same** r^* .

Coupled optimization problem

- Identify z_i (whether ϕ_i in $\mathcal{S}_1$) based on r
- Optimize r for purification with $z_i = 1$



● purification ϕ
➔ adversarial noise r

We adopt the EM algorithm as the optimizer

E-Step: Set $q(z_i) = p(z = 1|\mathcal{A}, r)$ given r

- Linear optimization

$$r^* = r + \Delta r = r + \arg \max \sum_{i=1}^N \sigma(\mathcal{L}_{\phi_i}(x + r) + \Delta r^T \nabla_x \mathcal{L}_{\phi_i} \big|_{x+r})$$

Attack as many purifications as possible

Linear approximated loss

- Loss-to-probability mapping σ

$$q(z_i) = \sigma(\mathcal{L}_{\phi_i}(x + r) + (r^* - r)^T \nabla_x \mathcal{L}_{\phi_i} \big|_{x+r})$$

E-Step: probability of purification attacked by optimized r^*

M-Step: optimize r given $q(z_i)$

- Weighted gradient aggregation **Time-consuming**

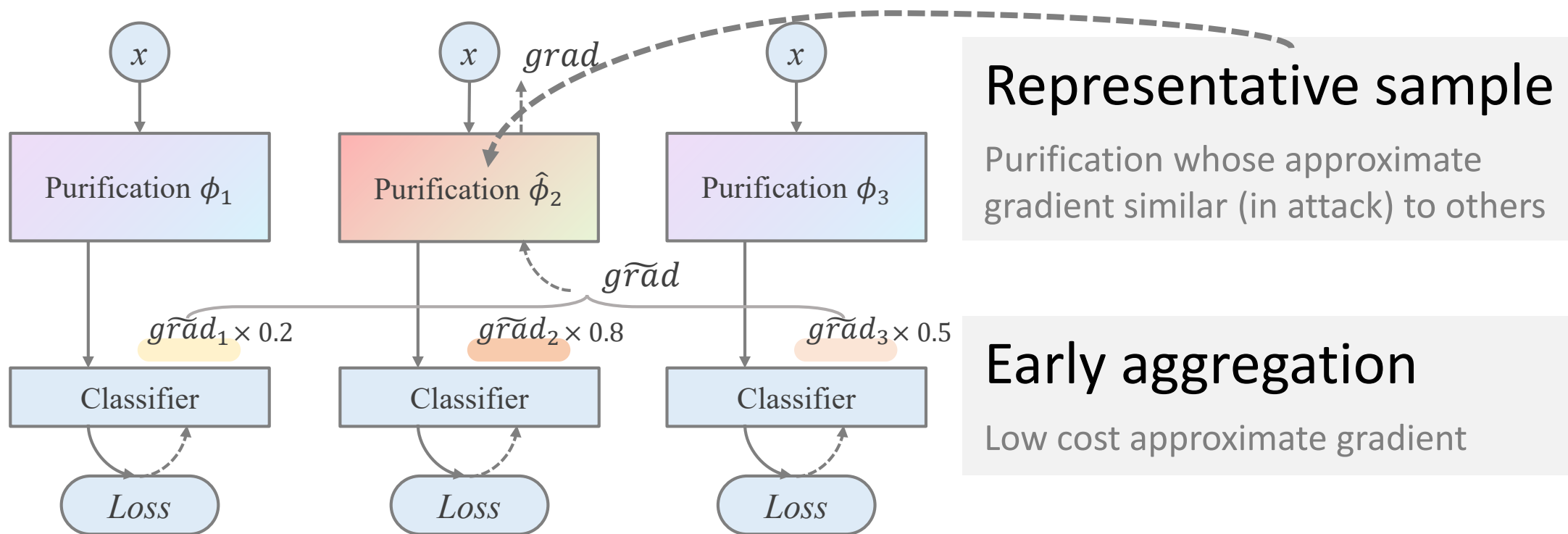
$$\nabla_r \mathcal{L} = \frac{1}{N} \sum_{i=1}^N q(z_i) \nabla_x \mathcal{L}_{\phi_i}$$

- Plug-in for off-the-shelf attack algorithms

$$r^{(t+1)} = \Pi_{\|r\|_{\infty} \leq \epsilon} [r^{(t)} + \alpha \text{sign}(\nabla_r \mathcal{L})]$$

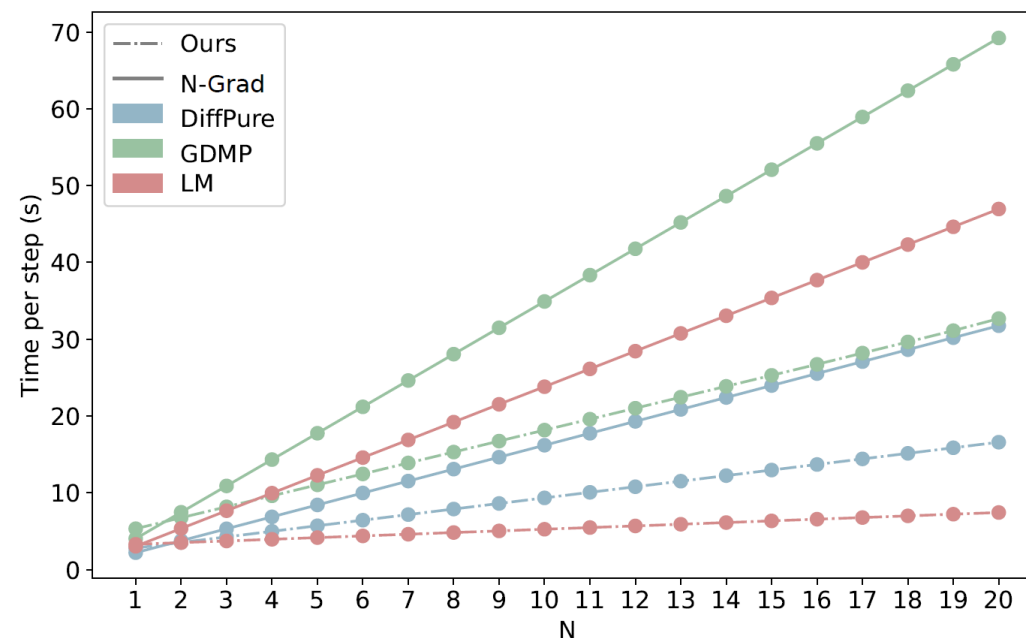
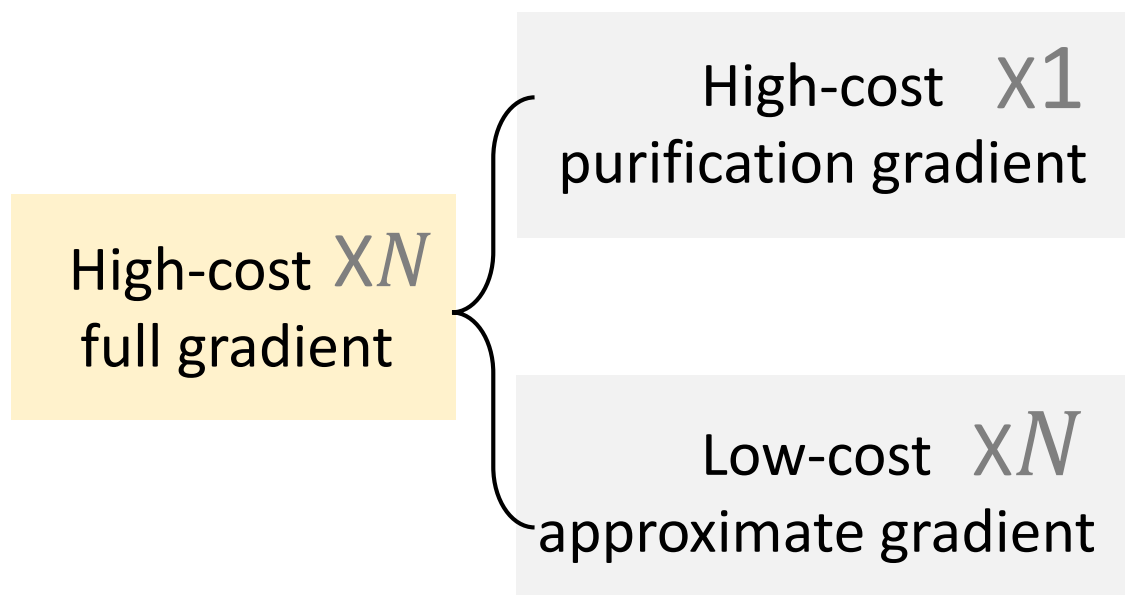
M-Step: aggregate gradients of purification from $\mathcal{S}_1$

Gradient grafting



Aggregate early and graft on representative sample to backpropagate

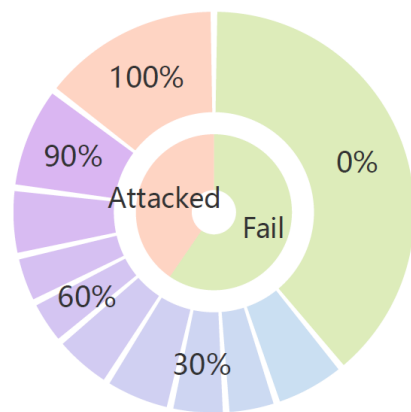
Gradient grafting



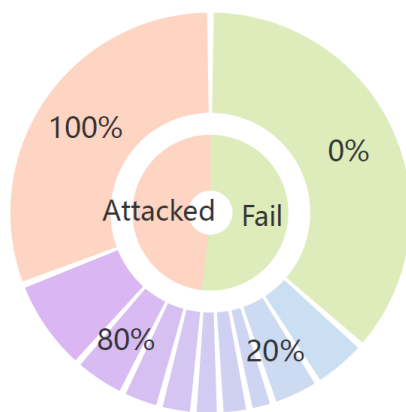
Gradient grafting significantly improves attack efficiency

Estimation of p is important for M resubmit risk: $(1 - p)^M$ ✓✓✓✗✓ ($p=0.2$)

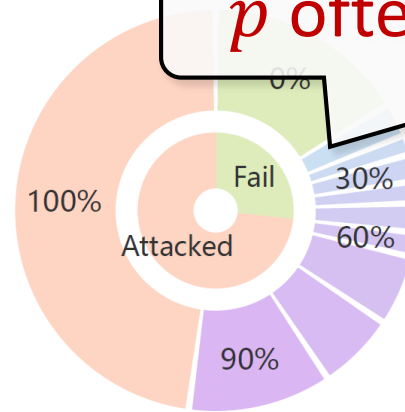
1-Evaluation only suitable for $p = 0,1$



(a) DiffPure



(b) GDMP



(c) LM

p often lies between 0 and 1

Statistically, 1-evaluation underestimates p and resubmit risk

N -Evaluation: ✓✓✓✗✓✓✗✓✓✓

1. More accurate estimate in $t - 1$:

$$p^{(t-1)} \approx 0.2$$

2. Byproducts for attack stage t :

$$\mathcal{L}_1^{(t)}, \dots, \mathcal{L}_{10}^{(t)}, \quad \tilde{g}_1^{(t)}, \dots, \tilde{g}_{10}^{(t)}$$

for $t \leftarrow 1$ **to** T **do**

Evaluation for $t - 1$ iteration and input for t iteration;

Sample $\phi_i, i = 1, \dots, N$;

$Rob^{(t-1)} = \text{Eval}(r^{(t-1)}, M)$ // for evaluation;

Record $\mathcal{L}_{\phi_i}^{(t)}, \nabla_{\phi_i}^{(t)} \mathcal{L}_{\phi_i}$ // for attack;

E-step: identify the set with shared vulnerability;

$\Delta \tilde{r}^{(t)} = \arg \max \sum_i \sigma(\mathcal{L}_{\phi_i}^{(t)} + \Delta \tilde{r}^T \nabla_{\phi_i}^{(t)} \mathcal{L}_{\phi_i})$;

$q_i^{(t)} = \sigma(\mathcal{L}_{\phi_i}^{(t)} + \Delta \tilde{r}^T \nabla_{\phi_i}^{(t)} \mathcal{L}_{\phi_i})$ // probability of affiliation;

M-step: estimate the full gradients' aggregation;

$\tilde{g}^{(t)} = \sum_i q_i^{(t)} \nabla_{\phi_i}^{(t)} \mathcal{L}_{\phi_i} / N$ // aggregation in ϕ stage;

$g^{(t)} = \text{Backward}(\hat{\phi}(x + r^{(t-1)})^T \tilde{g}^{(t)})$ // gradient grafting;

$r^{(t)} = \text{AttackAlgorithm}(r^{(t-1)}, g^{(t)})$;

Better risk assessment is almost free

Effectiveness and efficiency

Defense Metrics	DiffPure		GDMP		LM	
	Avg.Rob (it.)↓	Wor.Rob (it.)↓	Avg.Rob (it.)↓	Wor.Rob (it.)↓	Avg.Rob (it.)↓	Wor.Rob (it.)↓
Clean	90.98	76.56	93.26	83.79	87.77	74.61
BPD	70.74 (N/A)	36.72 (N/A)	80.57 (N/A)	51.95 (N/A)	55.27 (N/A)	27.54 (N/A)
DA/A	57.60 (N/A)	33.79 (N/A)	52.83 (N/A)	37.70 (N/A)	32.56 (N/A)	17.97 (N/A)
PGD	52.73 (112)	31.05 (112)	49.41 (N/A)	36.91 (N/A)	17.99 (31)	9.38 (31)
DH	42.54 (20)	22.66 (17)	41.64 (17)	27.54 (13)	16.15 (17)	8.01 (14)
DMI [†]	45.64 (41)	25.20 (35)	43.40 (31)	32.42 (27)	38.81 (N/A)	23.83 (N/A)
TMI [†]	45.04 (39)	25.20 (38)	45.43 (37)	34.77 (30)	41.13 (N/A)	25.59 (N/A)
VMI [†]	50.55 (N/A)	28.71 (44)	50.76 (N/A)	37.11 (44)	21.97 (39)	11.72 (32)
SVRE [†]	59.12 (N/A)	32.81 (N/A)	60.37 (N/A)	42.77 (N/A)	36.11 (N/A)	19.53 (136)

Iterations taken to reach 90% best performance

$\ell_\infty : 8/255$

DiffHammer (DH) achieves effective results within 10-30 iterations

Critical resubmit risk

Defense Metrics		DiffPure		GDMP		LM	
		Avg.Rob (it.)↓	Wor.Rob (it.)↓	Avg.Rob (it.)↓	Wor.Rob (it.)↓	Avg.Rob (it.)↓	Wor.Rob (it.)↓
Clean		90.98	76.56	93.26	83.79	87.77	74.61
$\ell_\infty : 8/255$	BPDA	70.74 (N/A)	36.72 (N/A)	80.57 (N/A)	51.95 (N/A)	55.27 (N/A)	27.54 (N/A)
	DA/AA	57.60 (N/A)	33.79 (N/A)	52.83 (N/A)	37.70 (N/A)	32.56 (N/A)	17.97 (N/A)
	PGD	52.73 (N/A)	31.05 (112)	49.41 (N/A)	36.91 (N/A)	17.99 (31)	9.38 (31)
	DH	42.54 (20)	22.66 (17)	41.64 (17)	27.54 (13)	16.15 (17)	8.01 (14)
	DMI [†]	45.64 (41)	25.20 (35)	43.40 (31)	32.42 (27)	38.81 (N/A)	23.83 (N/A)
	TMI [†]	45.04 (39)	25.20 (38)	45.43 (37)	34.77 (30)	41.13 (N/A)	25.59 (N/A)
	VMI [†]	50.55 (N/A)	28.71 (44)	50.76 (N/A)	37.11 (44)	21.97 (39)	11.72 (32)
	SVRE [†]	59.12 (N/A)	32.81 (N/A)	60.37 (N/A)	42.77 (N/A)	36.11 (N/A)	19.53 (136)

Defenses show robustness below 30% with 10-evaluation

Defense Metrics	DiffPure		GDMP		LM	
	Avg.Rob (it.)↓	Wor.Rob (it.)↓	Avg.Rob (it.)↓	Wor.Rob (it.)↓	Avg.Rob (it.)↓	Wor.Rob (it.)↓
Clean	90.98	76.56	93.26	83.79	87.77	74.61
BPDA	70.74 (N/A)	36.72 (N/A)	80.57 (N/A)	51.95 (N/A)	55.27 (N/A)	27.54 (N/A)
DA/AA	57.60 (N/A)	33.79 (N/A)	52.83 (N/A)	37.70 (N/A)	32.56 (N/A)	17.97 (N/A)
PGD	52.73 (N/A)	31.05 (112)	49.41 (N/A)	36.91 (N/A)	17.99 (31)	9.38 (31)
$\ell_\infty : 8/255$ DH	42.54 (20)	22.66 (17)	41.64 (17)	27.54 (13)	16.15 (17)	8.01 (14)
DMI [†]	45.64 (41)	25.20 (35)	43.40 (31)	32.42 (27)	38.81 (N/A)	23.83 (N/A)
TMI [†]	45.04 (39)	25.20 (38)	45.43 (37)	34.77 (30)	41.13 (N/A)	25.59 (N/A)
VMI [†]	50.55 (N/A)	28.71 (44)	50.76 (N/A)	37.11 (44)	21.97 (39)	11.72 (32)
SVRE [†]	59.12 (N/A)	32.81 (N/A)	60.37 (N/A)	42.77 (N/A)	36.11 (N/A)	19.53 (136)

Transfer-based attacks that generalize to all purifications

Generalization in $\mathcal{S}_1$ is beneficial (DiffHammer) while in $\mathcal{S}_0$ is harmful

- ❖ We propose a selective attack strategy that targets vulnerable purifications, enhancing evaluation efficiency through gradient grafting.
- ❖ We incorporate N -evaluation within the loop to quantify the resubmit risk of achieving at least one successful attack in practice.



香港科技大學
THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

DiffHammer

Rethinking the Robustness of Diffusion-Based Adversarial Purification

Kaibo Wang, Xiaowen Fu, Yuxuan Han, Yang Xiang



Website: <https://github.com/Ka1b0/DiffHammer>