MADS&P
Security and Privacy Research Group
at UW-Madison

The Interplay Between Vulnerabilities in Machine Learning Systems

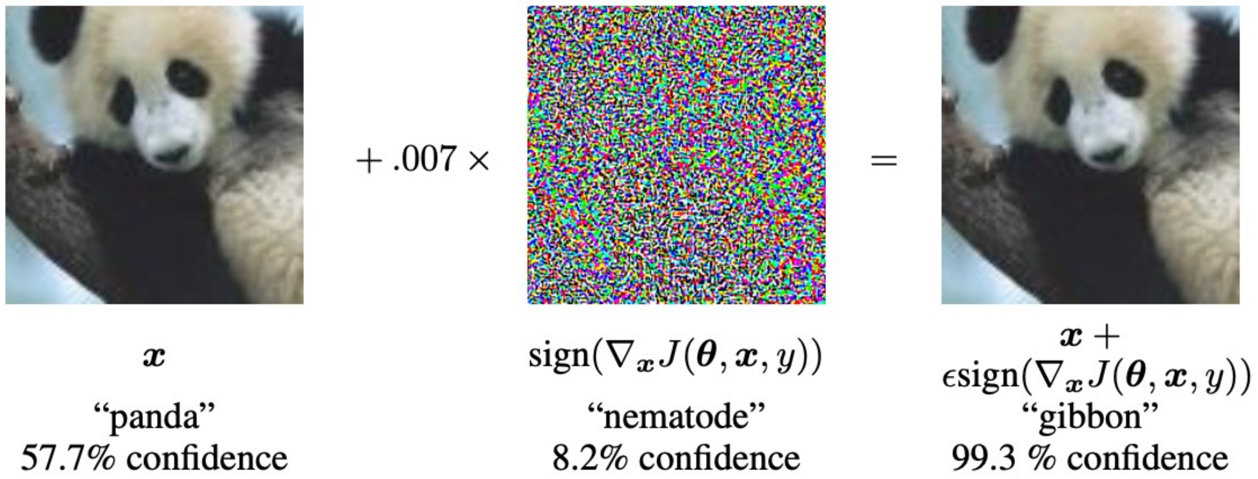
Yue Gao, Ilia Shumailov, Kassem Fawaz
University of Wisconsin–Madison

The 39th International Conference on Machine Learning, ICML 2022
Baltimore, USA

Motivation

Adversarial robustness of real-world ML systems?

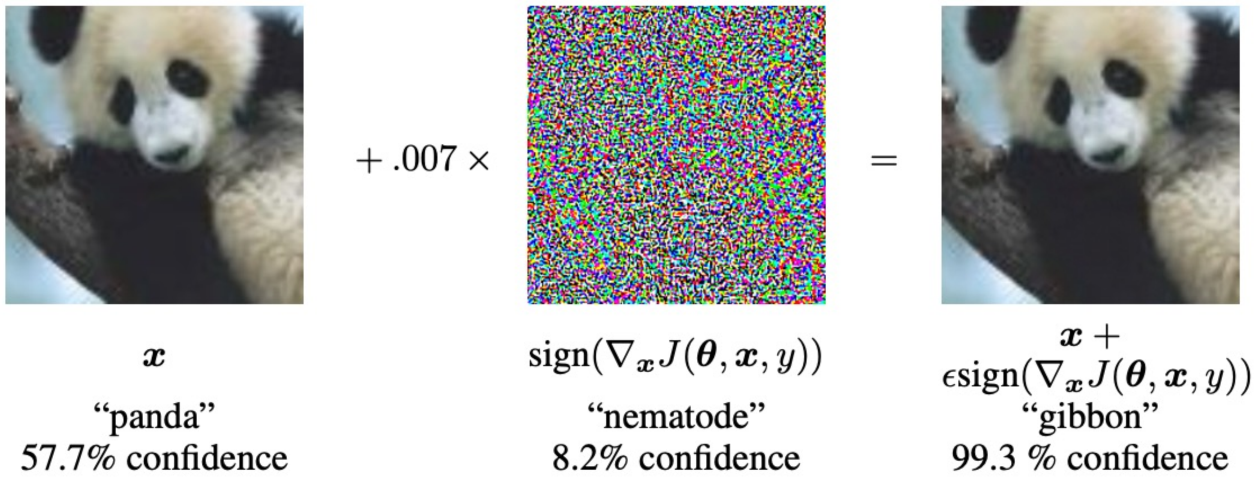
ML Model Attacks & Defenses



(Szegedy et al. 2013, Goodfellow et al. 2015)

- Adversarial Training
- Randomized Smoothing
- Pre-processing
- Post-processing
- Detection
- …

ML Model Attacks & Defenses

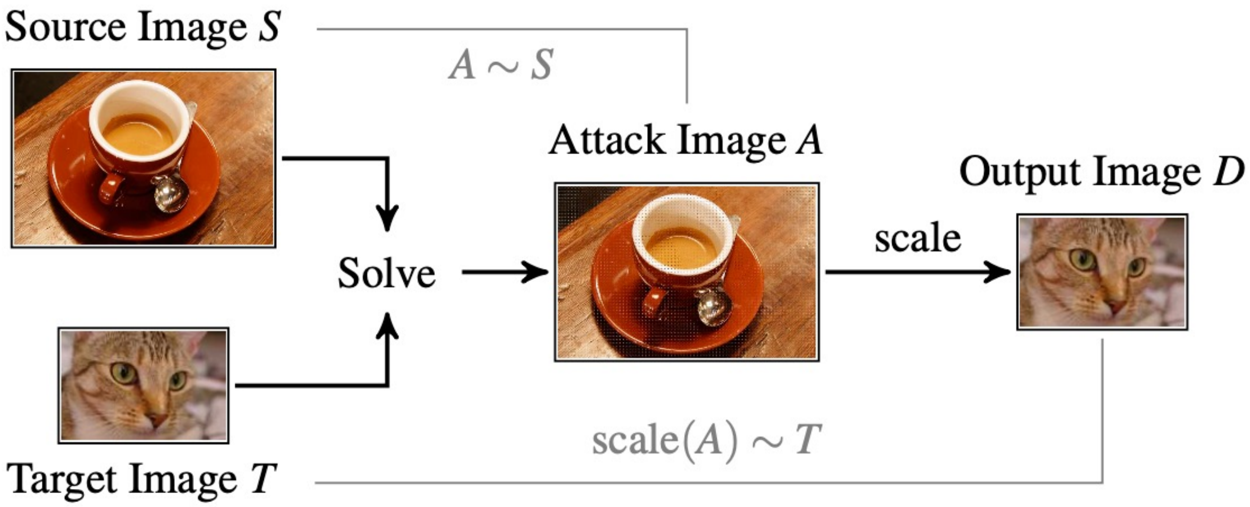


(Szegedy et al. 2013, Goodfellow et al. 2015)

- Adversarial Training
- Randomized Smoothing
- Pre-processing
- Post-processing
- Detection
- …

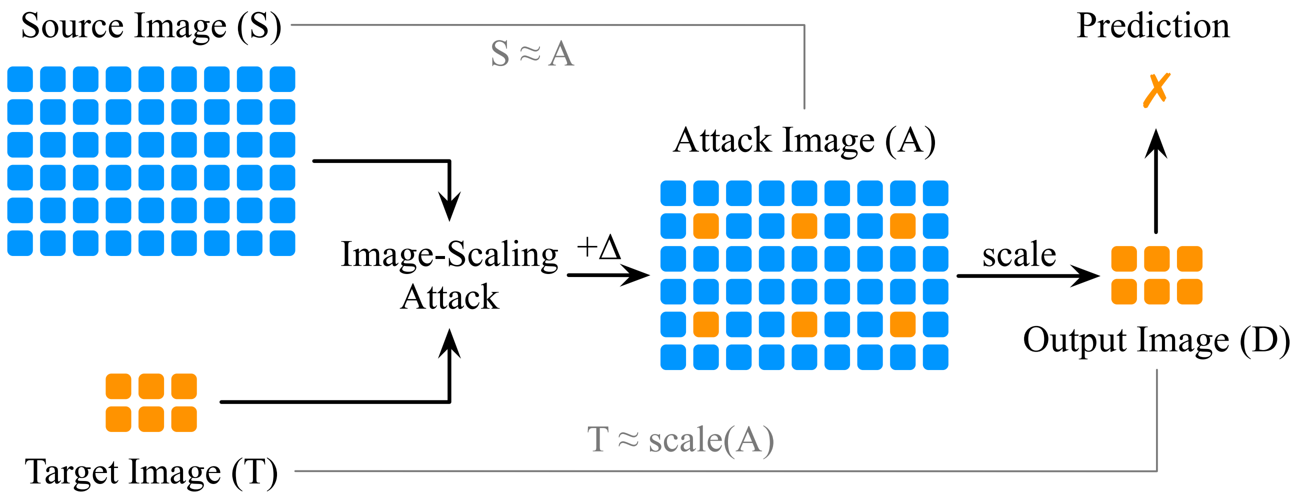
ML System = ML Model + Pre-processing + ...

Image-Scaling Attacks & Defenses



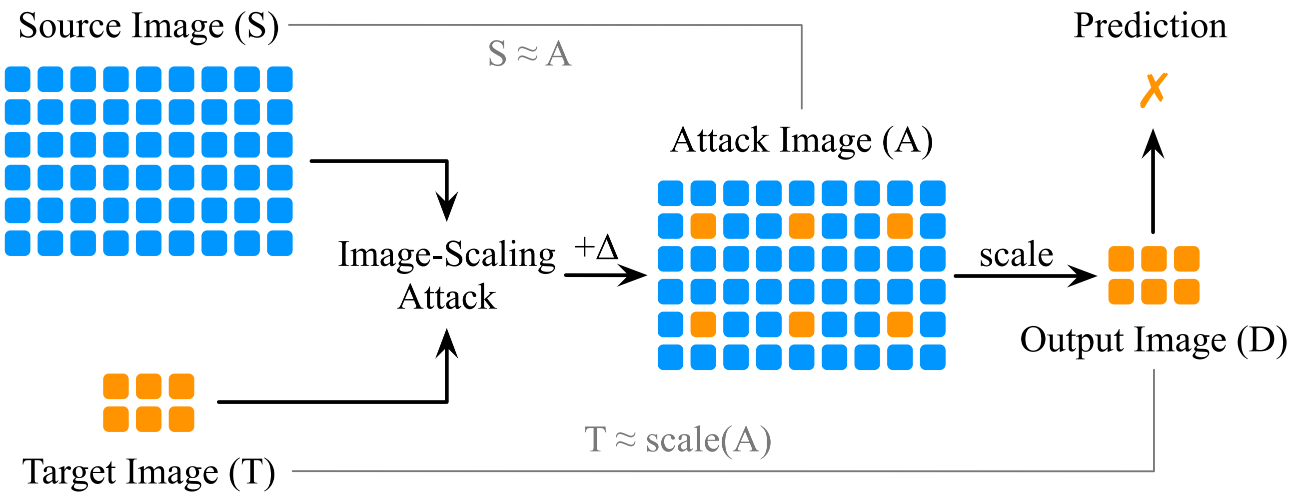
(Xiao et al. 2019, Quiring et al. 2020)

Image-Scaling Attacks & Defenses



A Simplified Demonstration

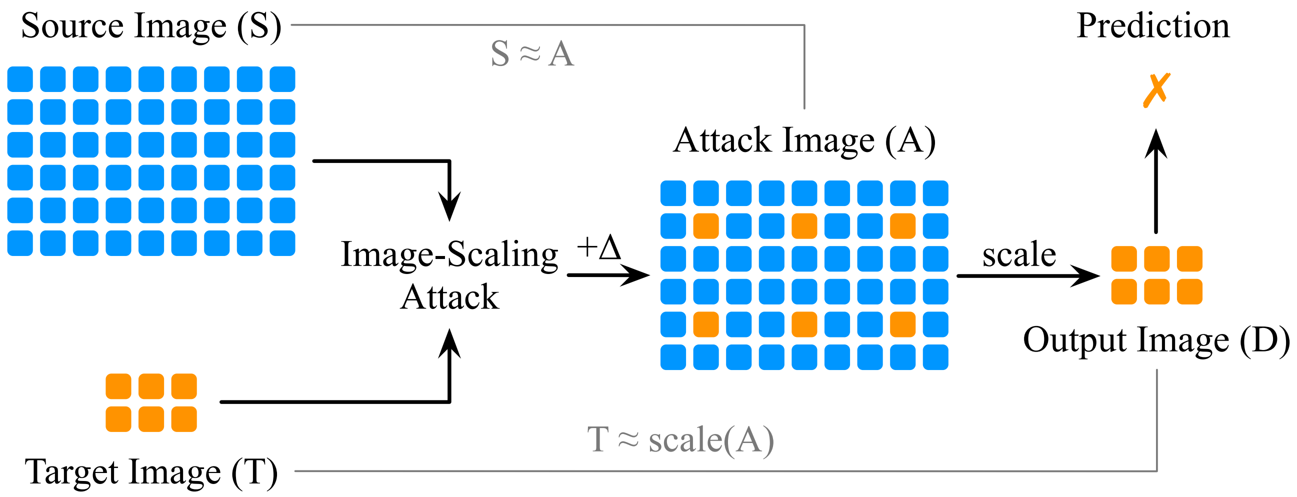
Image-Scaling Attacks & Defenses



A Simplified Demonstration

Practical: Infer the scaling function with black-box queries

Image-Scaling Attacks & Defenses

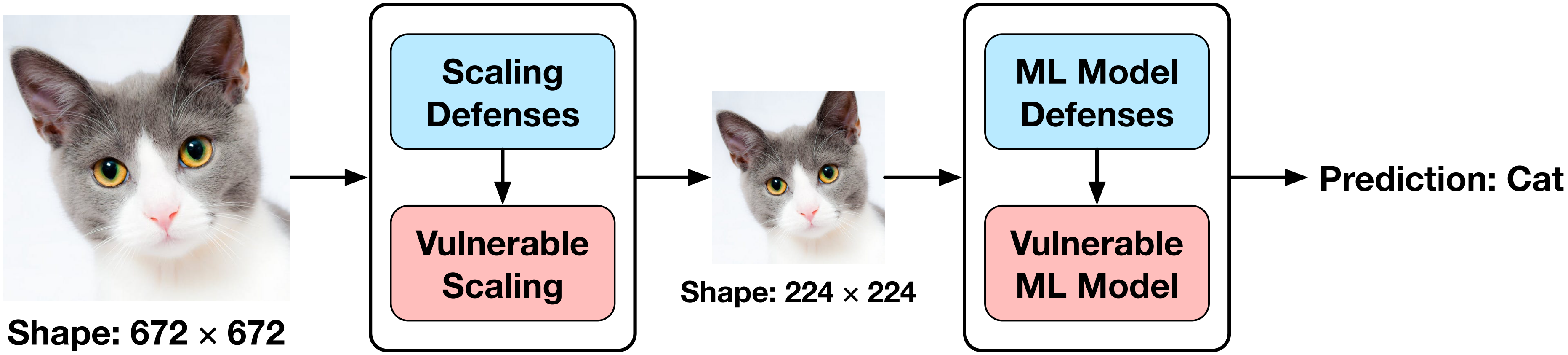


A Simplified Demonstration

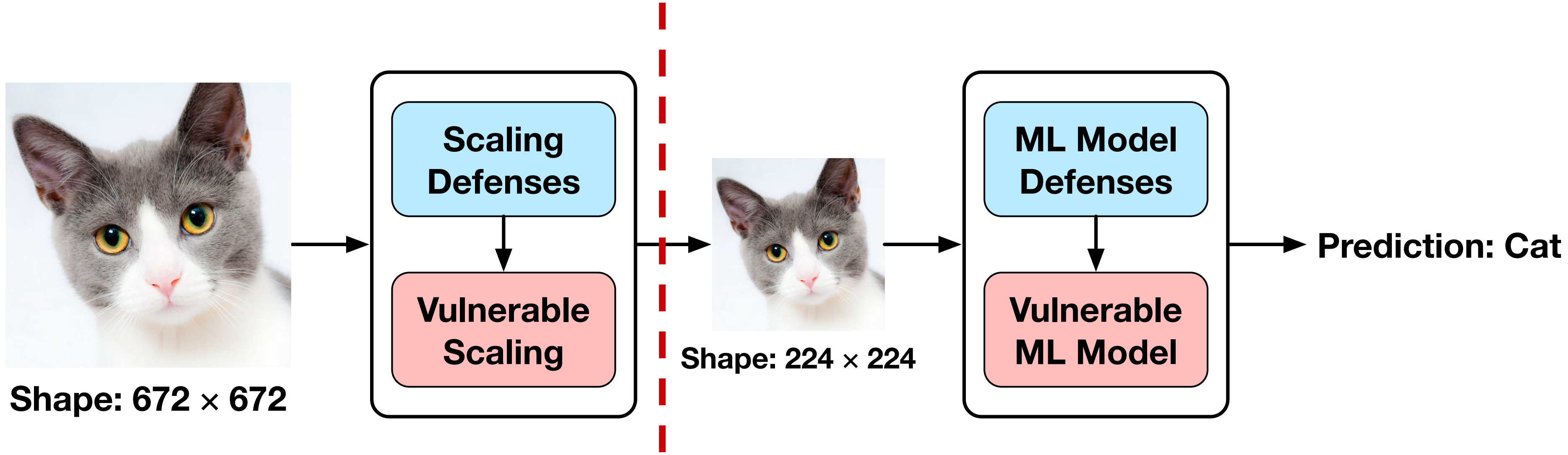
- Median Filtering
- Randomized Filtering
- Down-scaling + Up-scaling
- Spectrum Detection
- Statistical Test
- …

Practical: Infer the scaling function with black-box queries

A Broader View of the Entire ML Pipeline

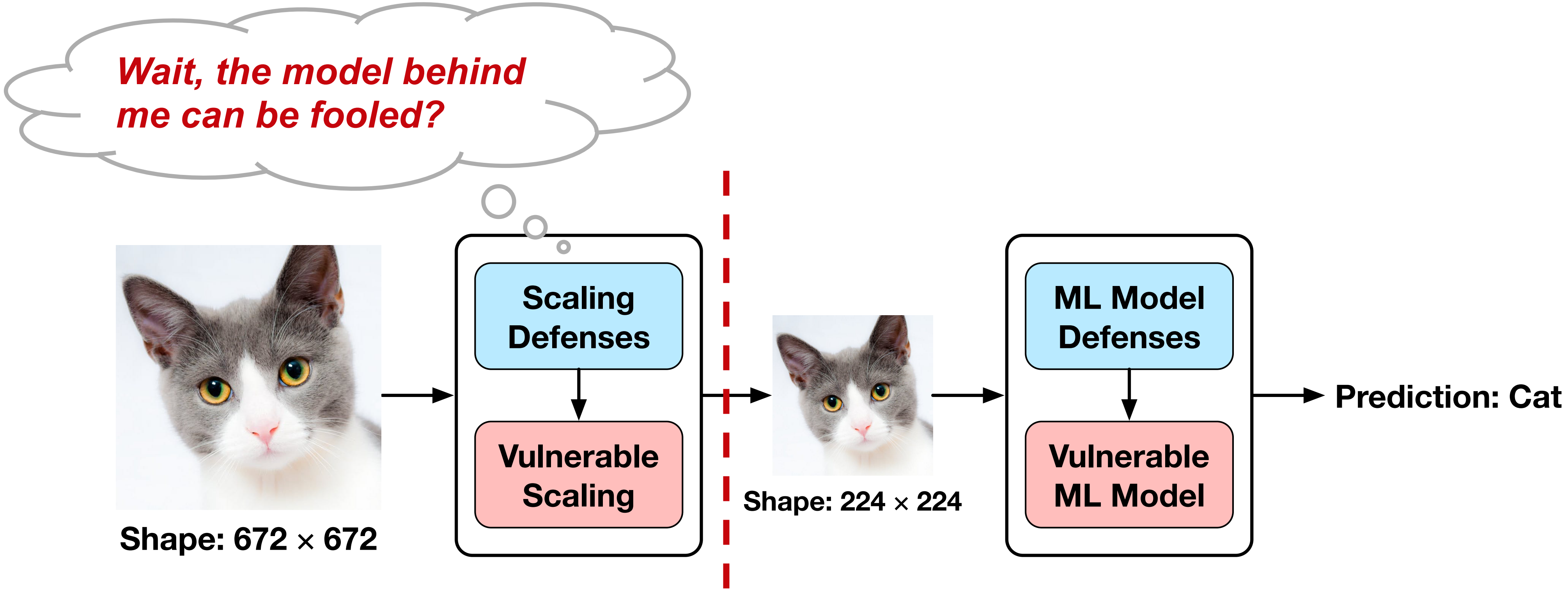


A Broader View of the Entire ML Pipeline



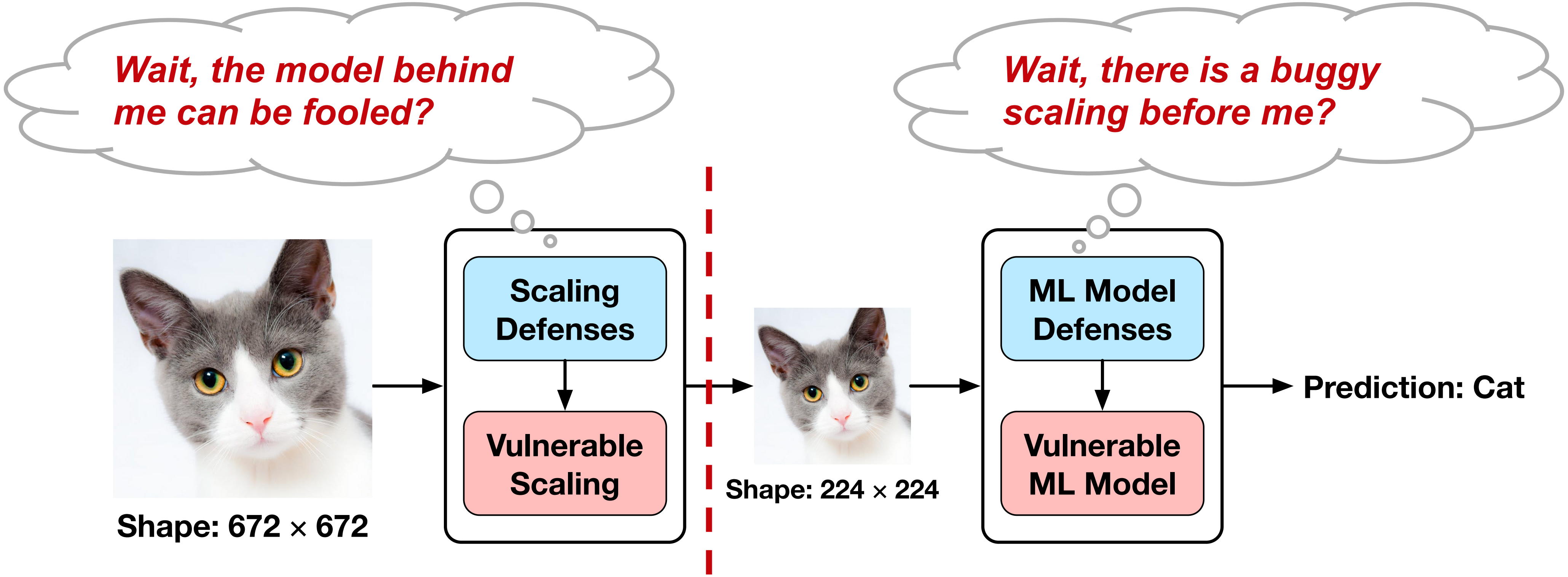
Defenses are tailored to each component.

A Broader View of the Entire ML Pipeline



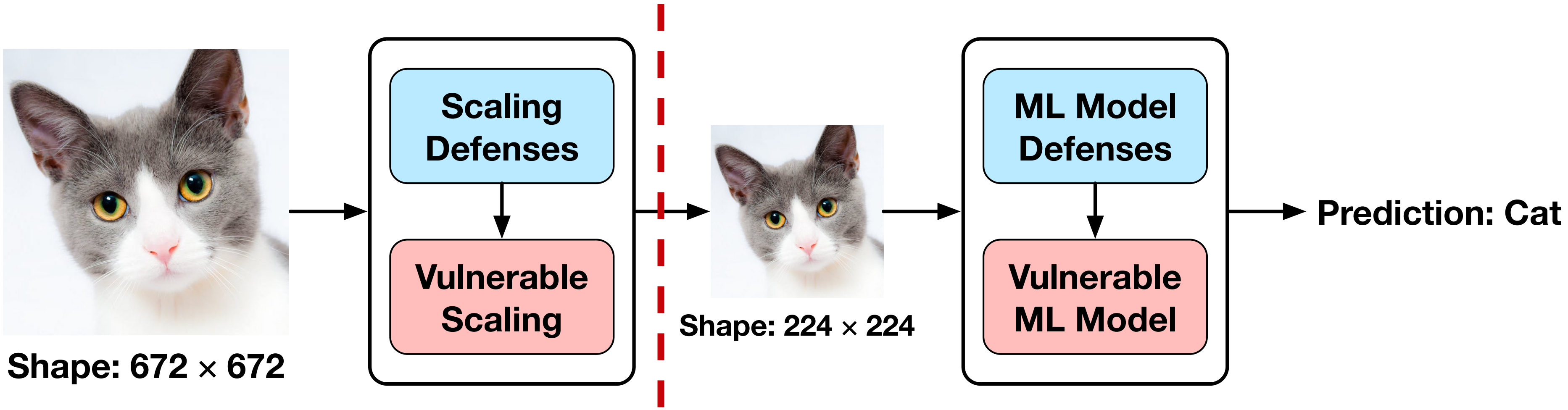
Defenses are tailored to each component.

A Broader View of the Entire ML Pipeline

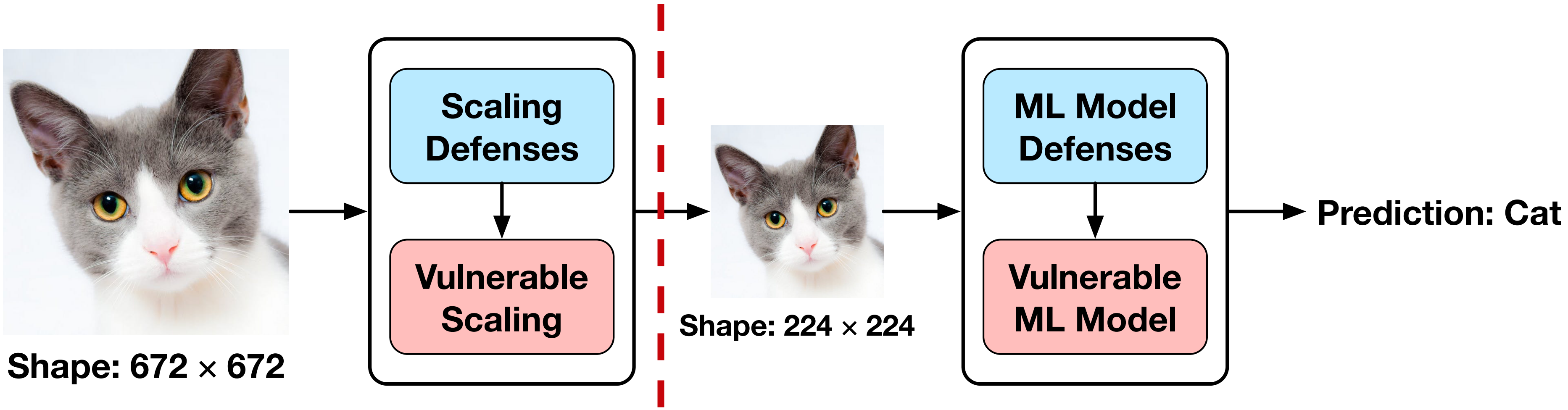


Defenses are tailored to each component.

Defenses Hold (Unnecessary) Strong Assumptions



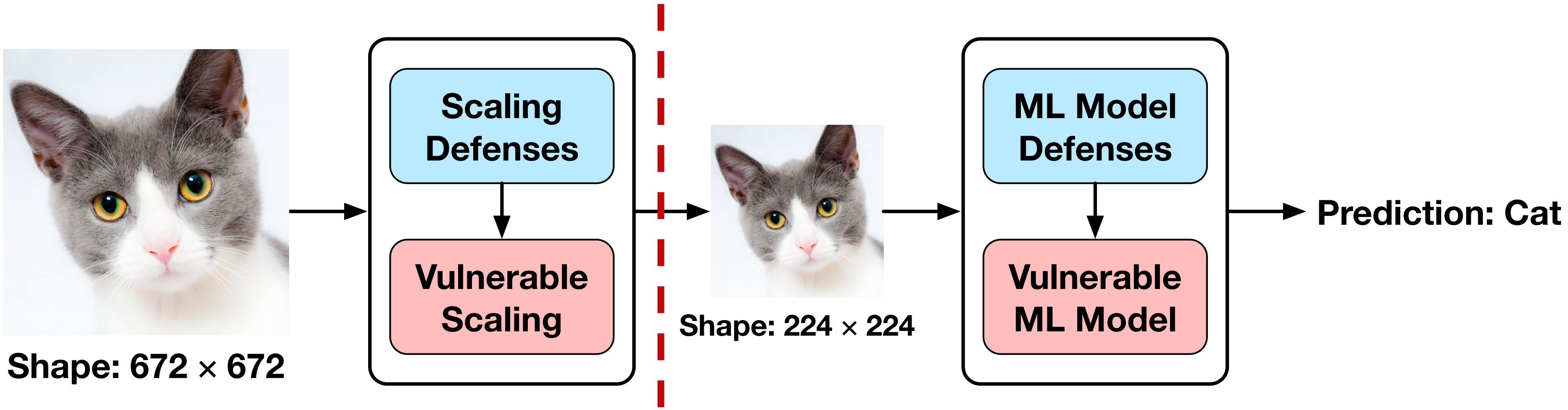
Defenses Hold (Unnecessary) Strong Assumptions



 "I inject clean images."

 "OK, you only inject clean images."

Defenses Hold (Unnecessary) Strong Assumptions



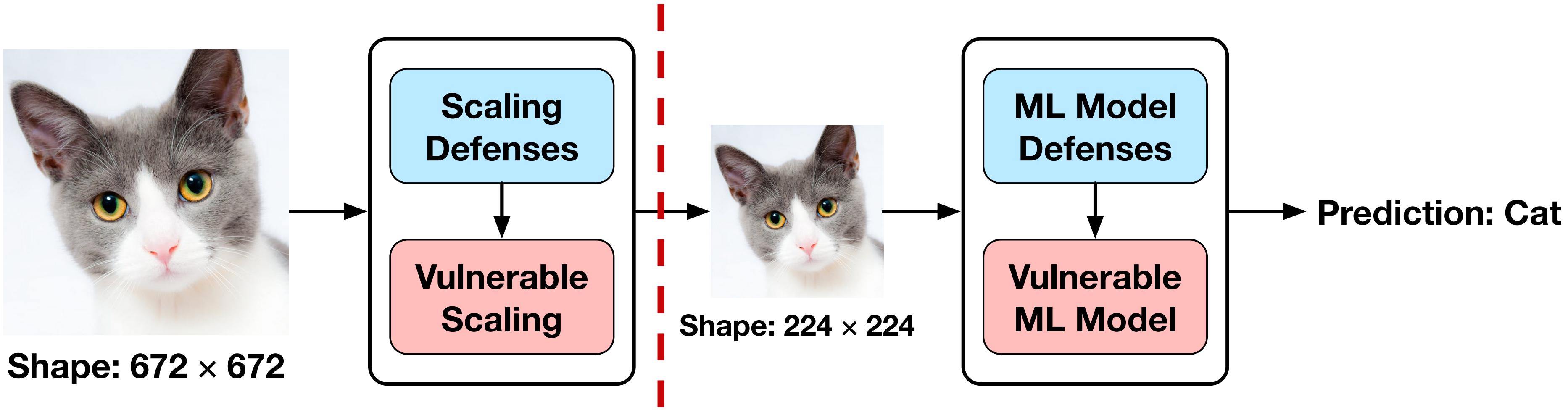
"I inject clean images."

"OK, you only inject clean images."

"I perturb the model's exact input."

"OK, you only perturb the exact input."

Defenses Hold (Unnecessary) Strong Assumptions



"I inject clean images."

"OK, you only inject clean images."

"I perturb the model's exact input."

"OK, you only perturb the exact input."

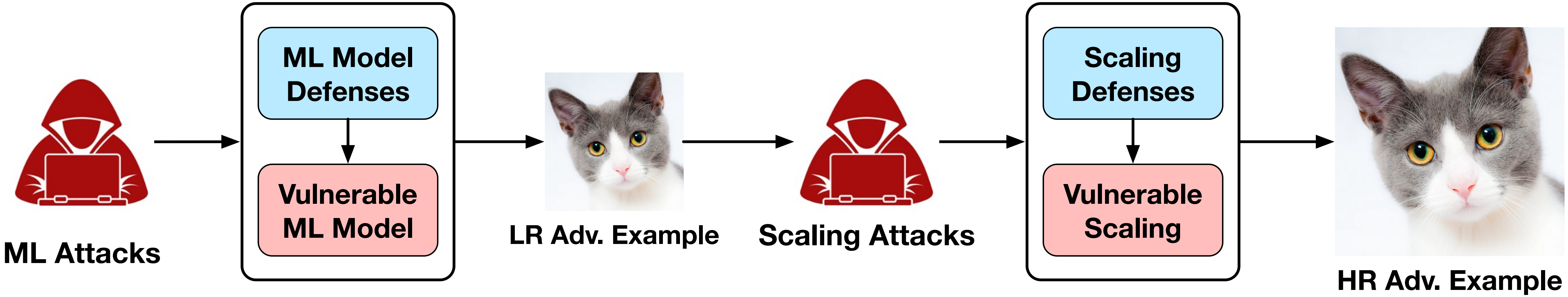
What if the adversary is aware of multiple vulnerabilities?

Scaling-aware Evasion Attacks

A black-box adversary targeting the entire ML pipeline.

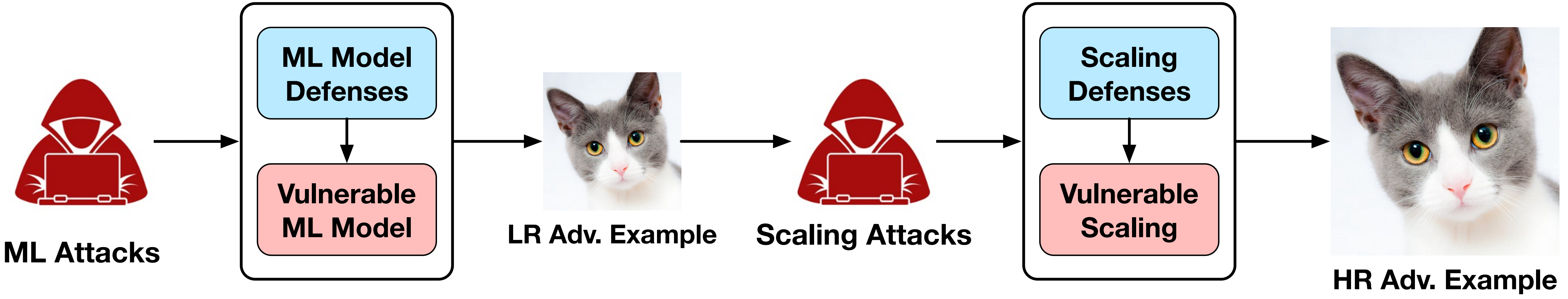
How to Make Attacks “Scaling-aware”?

- Strategy 1: Naively combine two attacks.



How to Make Attacks "Scaling-aware"?

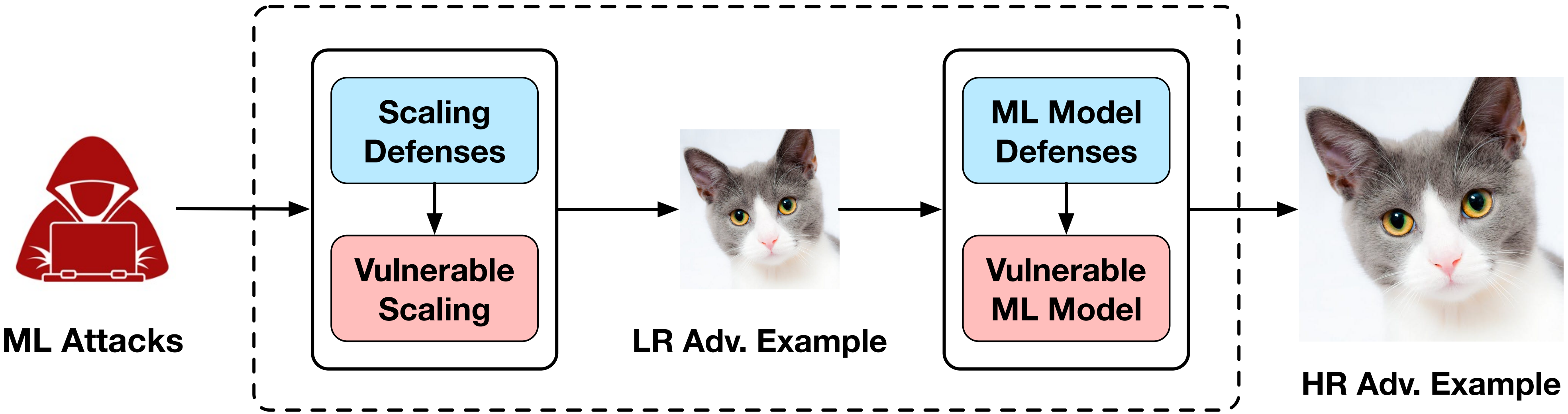
- Strategy 1: Naively combine two attacks.



✗ hard to remain adversarial

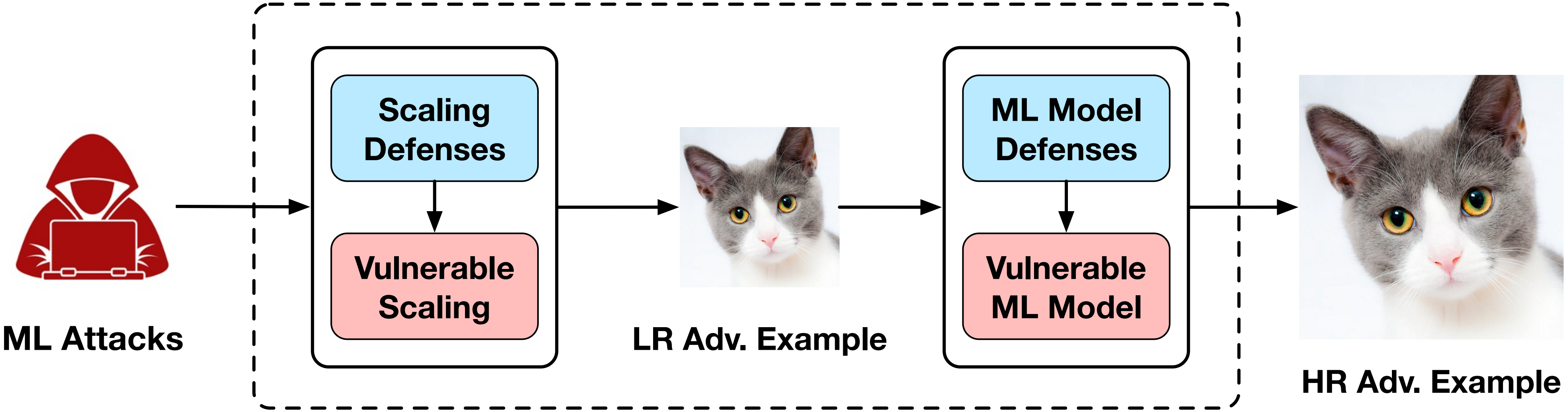
How to Make Attacks "Scaling-aware"?

- ~~Strategy 1: Naively combine two attacks.~~
- Strategy 2: Adapt existing black-box attacks to the entire pipeline.



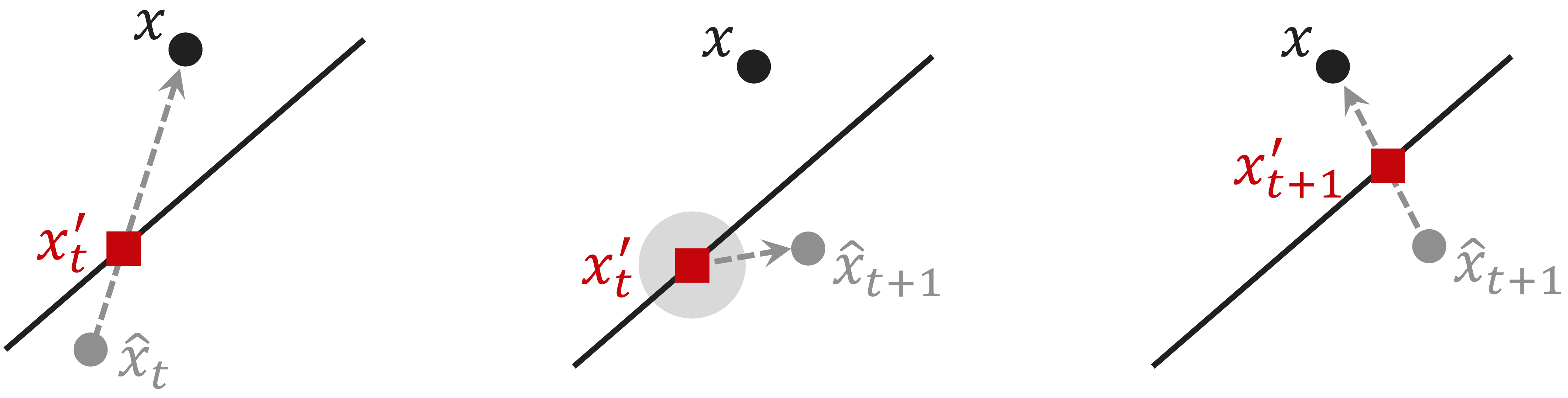
How to Make Attacks “Scaling-aware”?

- ~~Strategy 1: Naively combine two attacks.~~
- Strategy 2: Adapt existing black-box attacks to the entire pipeline.



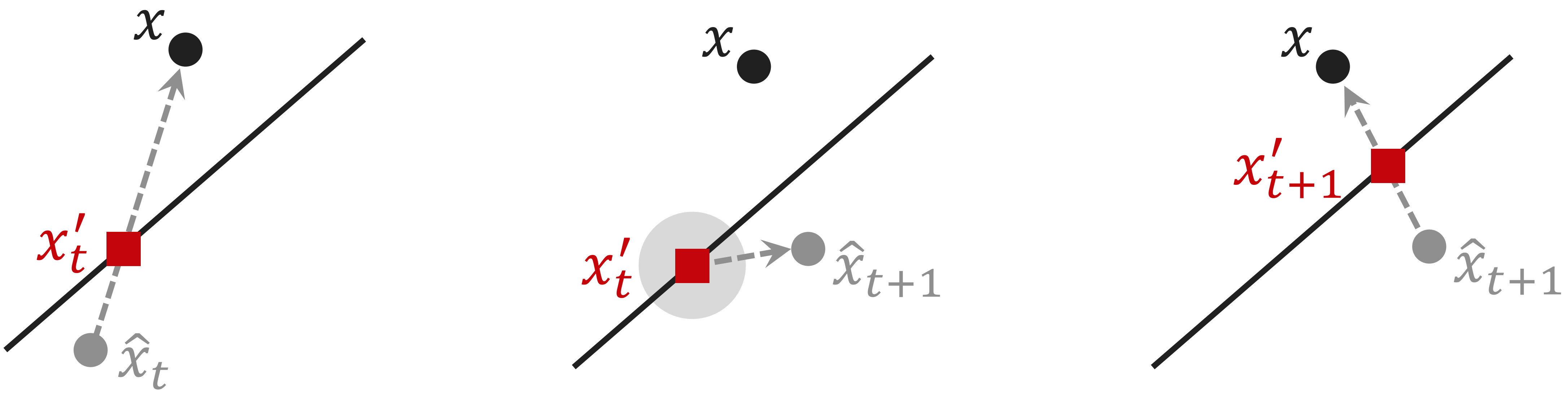
✗ *cannot exploit scaling by itself*

Typical Decision-based Black-box Attacks



1. Find a point near the boundary
2. Sample noise to estimate gradient
3. Find a better point

Typical Decision-based Black-box Attacks

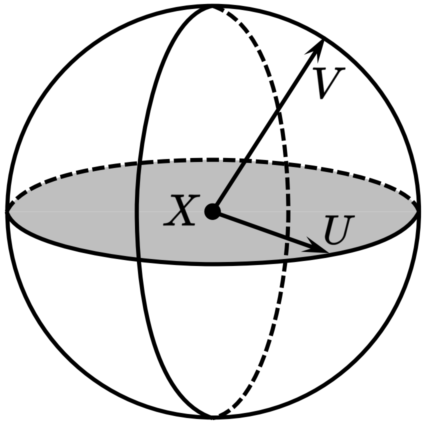


1. Find a point near the boundary
2. Sample noise to estimate gradient
3. Find a better point

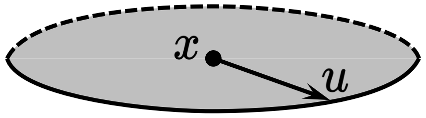
↑ ***incorporate the vulnerability here***

Main Technique: Scaling-aware Noise Sampling

- Vulnerability lies in the LR space (gray).
- We need noise in the HR space (ball).
- How likely a uniform noise satisfies that? Zero.



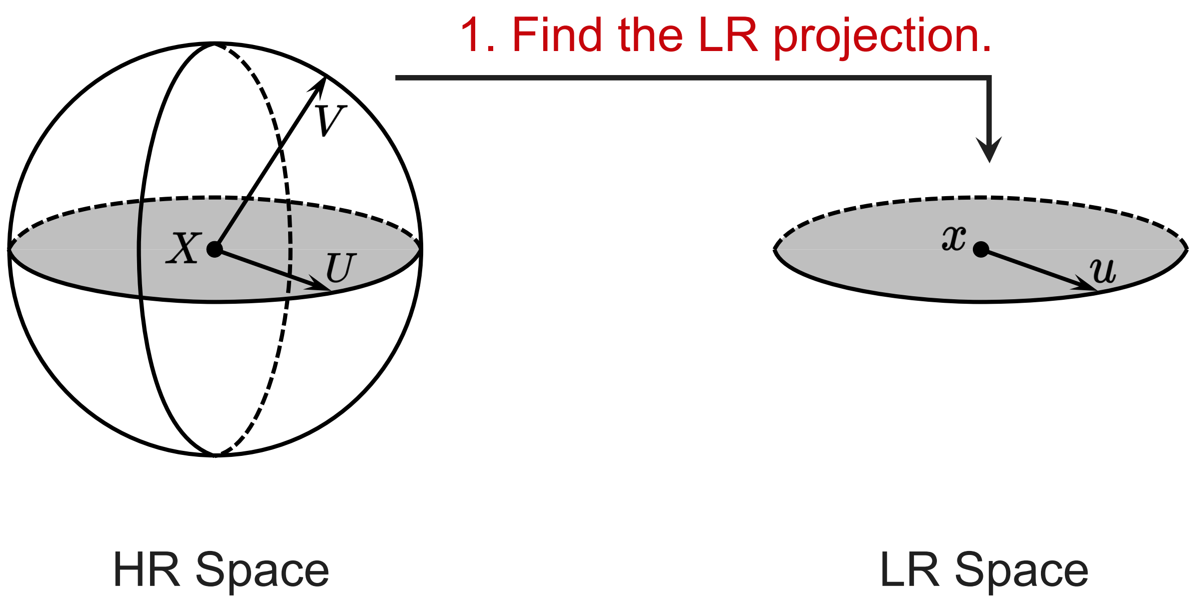
HR Space



LR Space

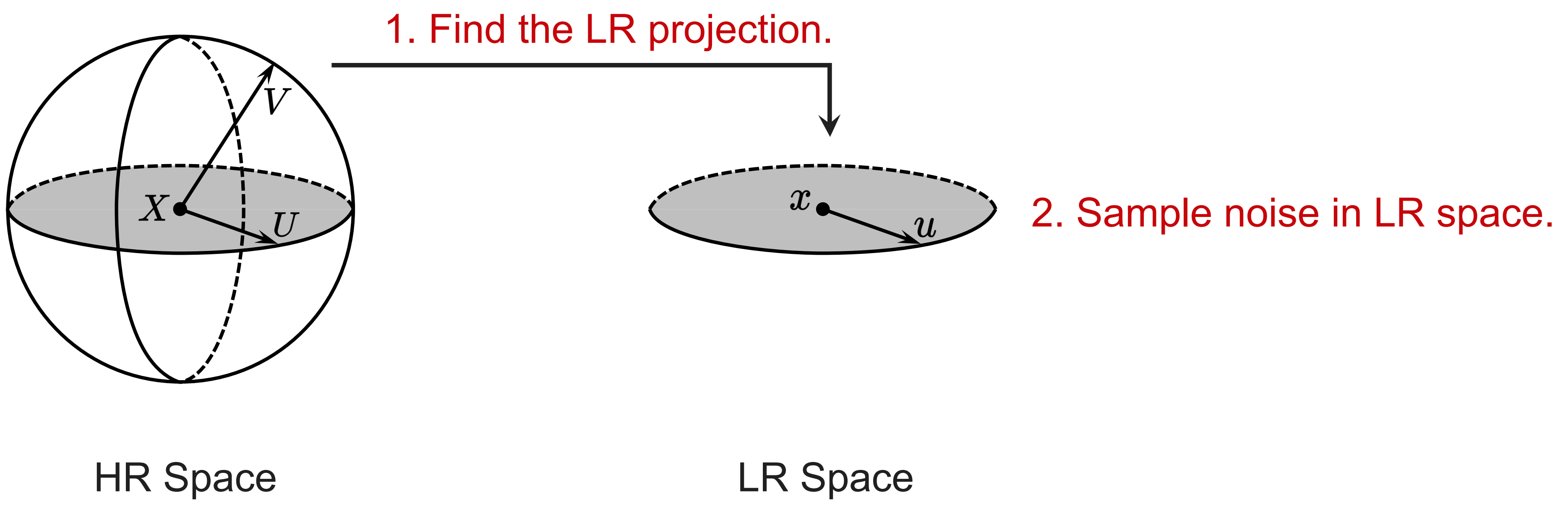
Main Technique: Scaling-aware Noise Sampling

- Vulnerability lies in the LR space (gray).
- We need noise in the HR space (ball).
- How likely a uniform noise satisfies that? Zero.



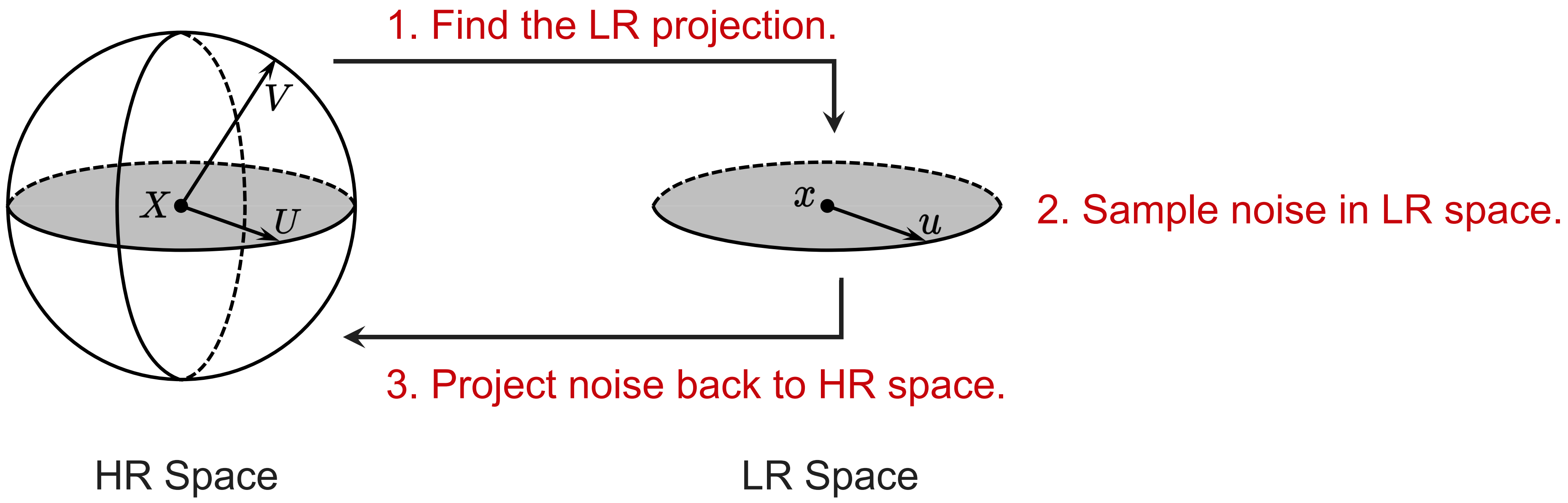
Main Technique: Scaling-aware Noise Sampling

- Vulnerability lies in the LR space (gray).
- We need noise in the HR space (ball).
- How likely a uniform noise satisfies that? Zero.

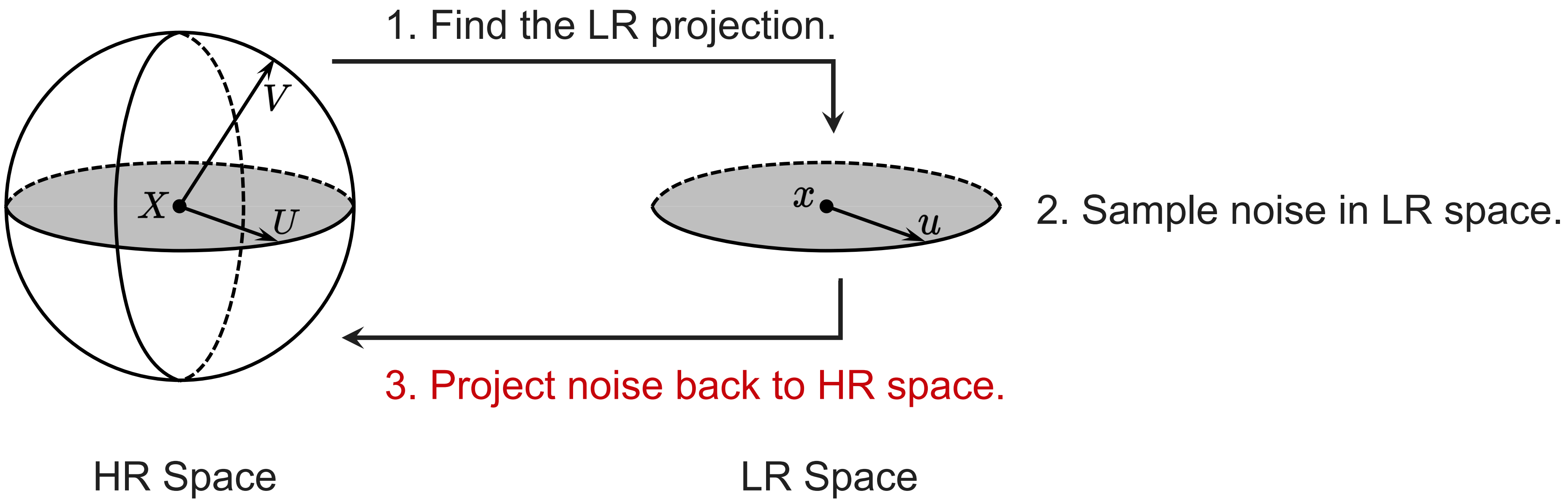


Main Technique: Scaling-aware Noise Sampling

- Vulnerability lies in the LR space (gray).
- We need noise in the HR space (ball).
- How likely a uniform noise satisfies that? Zero.



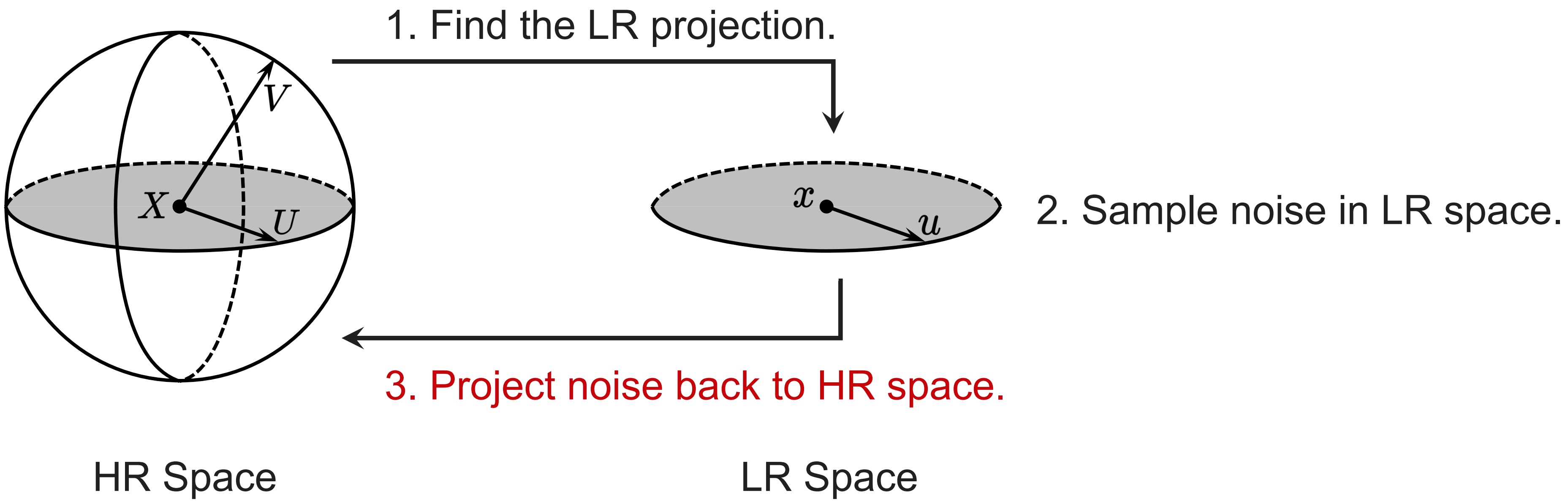
How to Inverse the Projection?



How to Inverse the Projection?

- Straightforward inversion.

$$U^* := \arg\min_{U \in \mathbb{H}} \|\text{scale}(X+U) - (\text{scale}(X)+u)\|_2^2$$



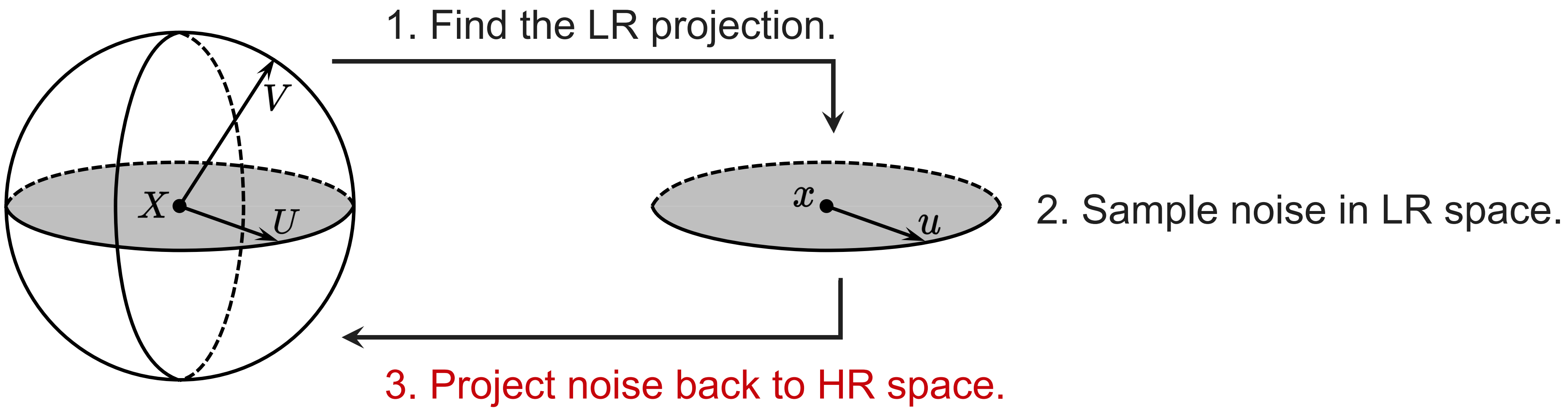
How to Inverse the Projection?

- Straightforward inversion.

LR Noise (sampled)

$$U^* := \arg\min_{U \in \mathbb{H}} \|\mathrm{scale}(X + U) - (\mathrm{scale}(X) + u)\|_2^2$$

HR Noise (unknown)



HR Space

LR Space

How to Inverse the Projection?

- Straightforward inversion.

LR Noise (sampled)

$$U^* := \arg\min_{U \in \mathbb{H}} \|\text{scale}(X + U) - (\text{scale}(X) + u)\|_2^2$$

HR Noise (unknown)

Cost: 1K step SGD for ~1K noise per attack step.

How to Inverse the Projection?

- Straightforward inversion.

LR Noise (sampled)

$$U^* := \arg\min_{U \in \mathbb{H}} \|\text{scale}(X + U) - (\text{scale}(X) + u)\|_2^2$$

HR Noise (unknown)

Cost: 1K step SGD for ~1K noise per attack step.

Insight: We do not need a precise solution for a noise.

How to Inverse the Projection?

- Straightforward inversion.

LR Noise (sampled)

$$U^* := \arg\min_{U \in \mathbb{H}} \|\mathrm{scale}(X + U) - (\mathrm{scale}(X) + u)\|_2^2$$

HR Noise (unknown)

- Efficient inversion.

$$\hat{U} := \nabla_U \|\mathrm{scale}(X + U) - (\mathrm{scale}(X) + u)\|_2^2$$

How to Inverse the Projection?

- Straightforward inversion.

LR Noise (sampled)

$$U^* := \arg\min_{U \in \mathbb{H}} \|\text{scale}(X + U) - (\text{scale}(X) + u)\|_2^2$$

HR Noise (unknown)

- Efficient inversion.

Vulnerable Direction

$$\hat{U} := \nabla_U \|\text{scale}(X + U) - (\text{scale}(X) + u)\|_2^2$$

Encode Vulnerability

How to Inverse the Projection?

- Straightforward inversion.

LR Noise (sampled)

$$U^* := \arg\min_{U \in \mathbb{H}} \|\mathrm{scale}(X+U) - (\mathrm{scale}(X)+u)\|_2^2$$

HR Noise (unknown)

- Efficient inversion.

Vulnerable Direction

$$\hat{U} := \nabla_U \|\mathrm{scale}(X+U) - (\mathrm{scale}(X)+u)\|_2^2$$

Encode Vulnerability

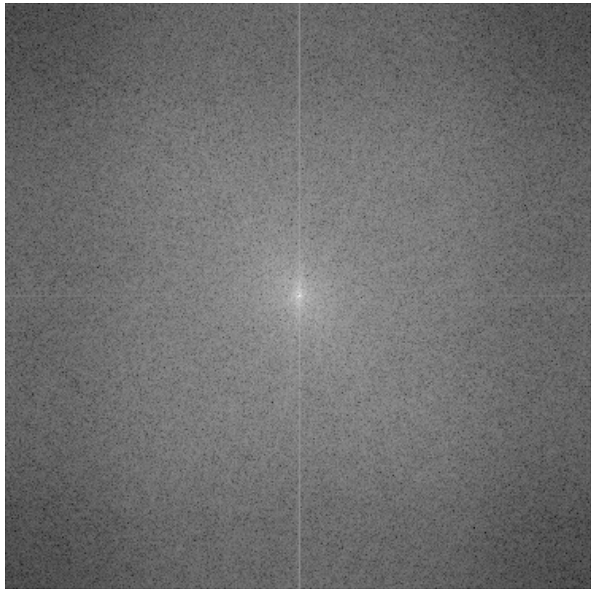
Cost: 1K step SGD ➔ 1 Backward Pass

Amplified Threats

From the interplay between vulnerabilities.

Evade Scaling Defenses

- Evade 4 out of 5 scaling defenses.
- E.g., no artifacts in the spectrum image.



Original Image

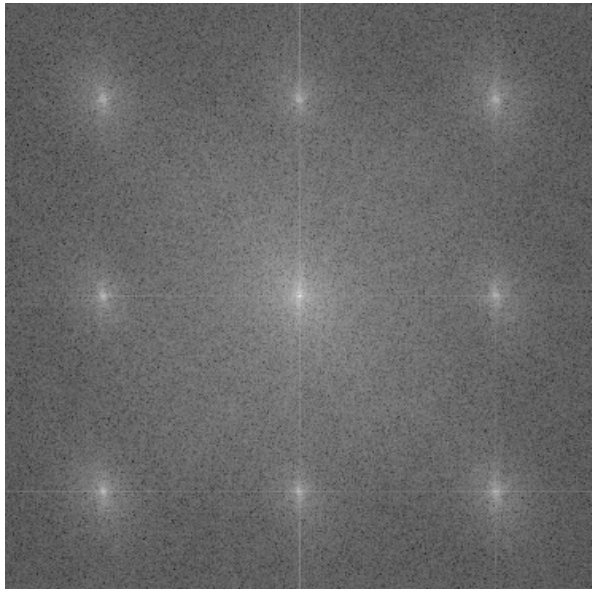
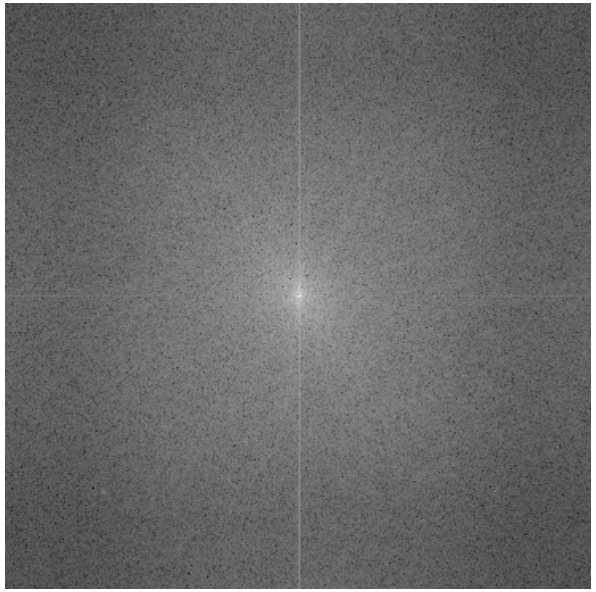


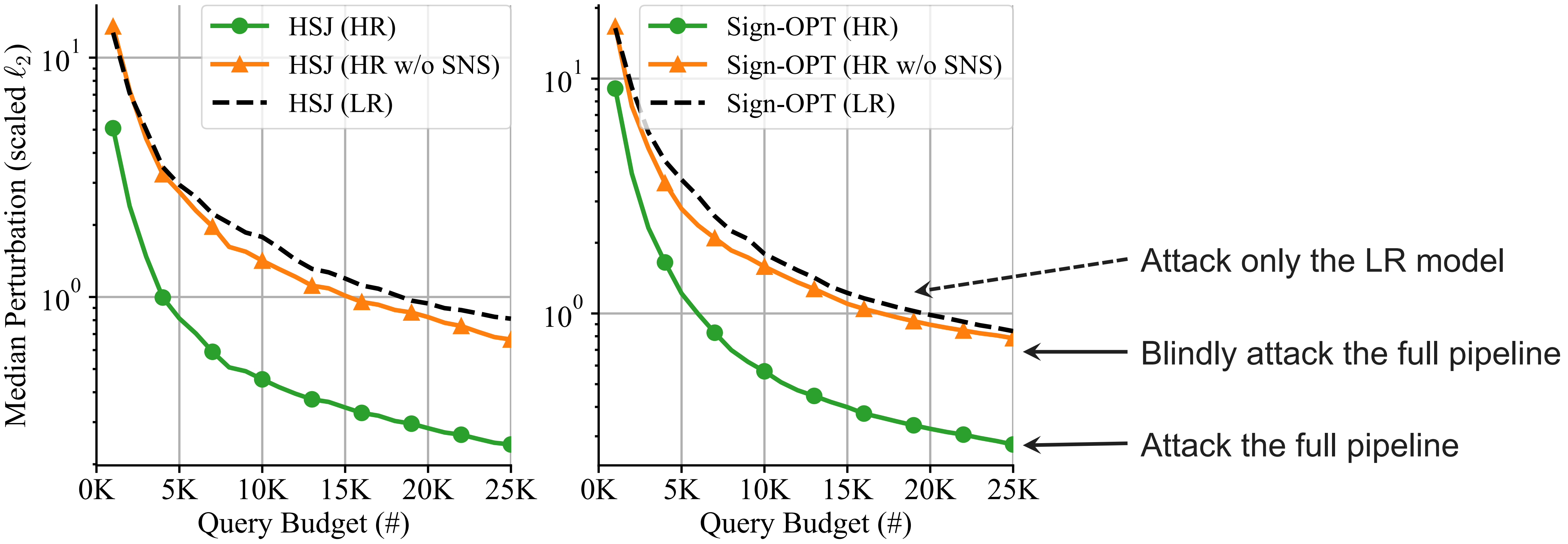
Image-Scaling Attack



Scaling-aware Attack

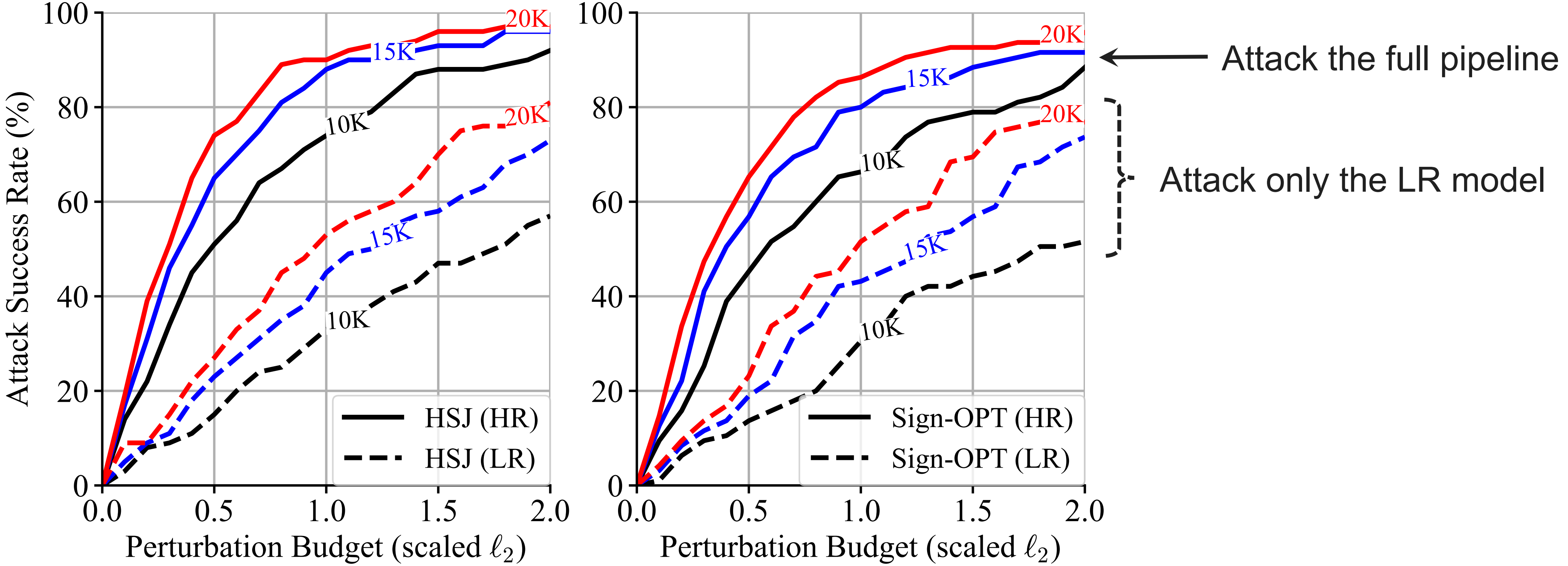
Black-box Attacks: More Query Efficient

- Same query budget, less perturbation.



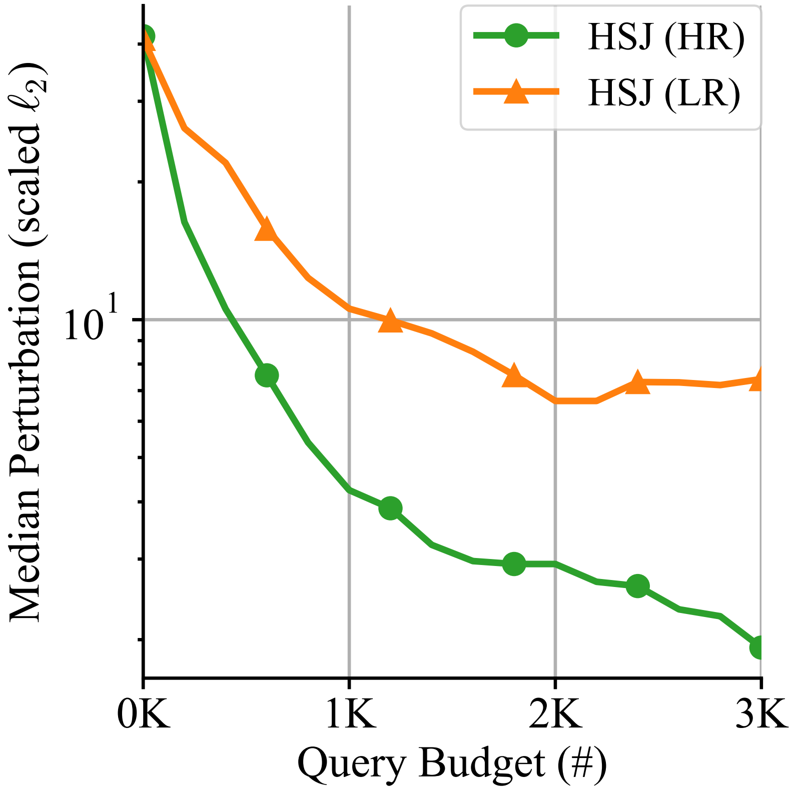
Black-box Attacks: More Effective

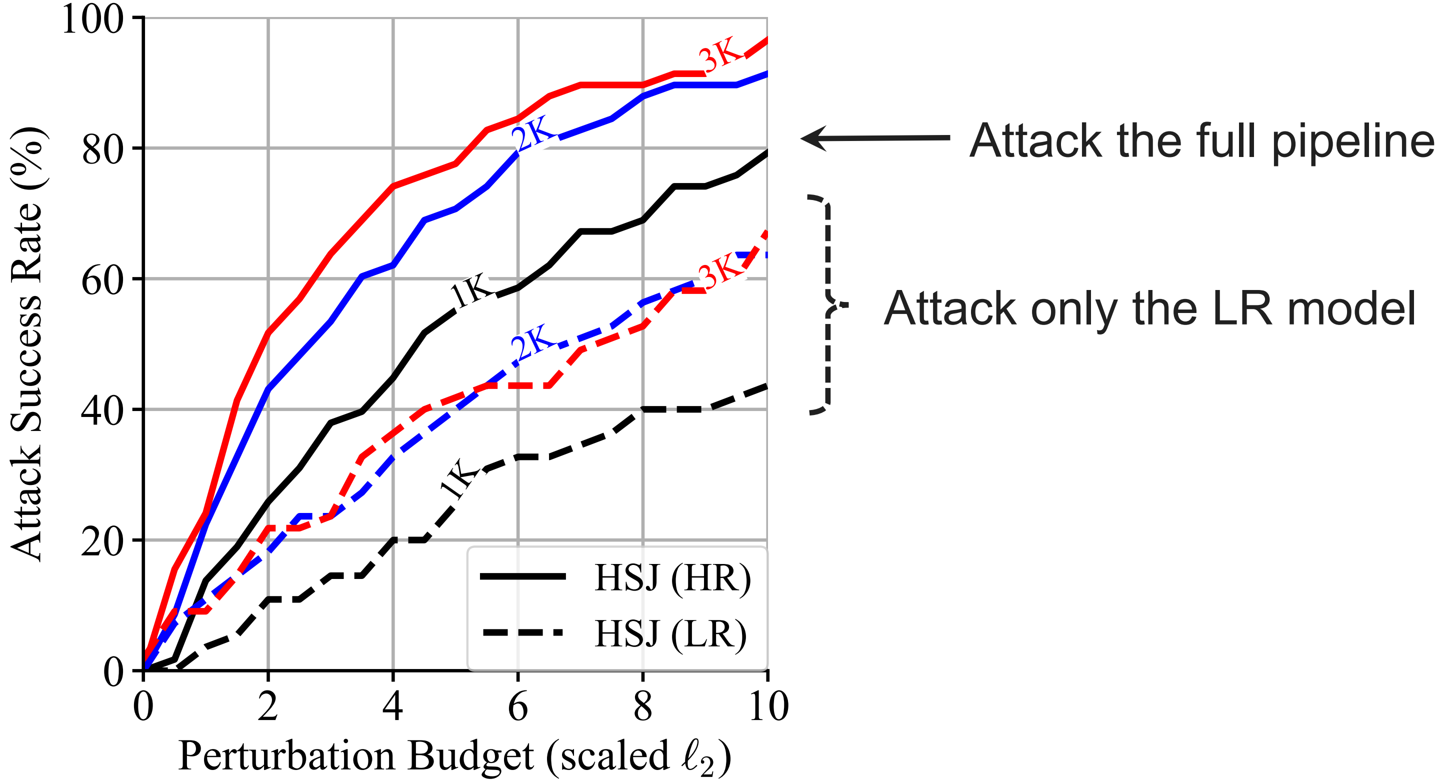
- Same perturbation budget, higher attack success rate.



Black-box Attacks: More Practical

- Same improvements on Tencent Image Analysis API





Conclusions

Implications for trustworthy machine learning.

Be cautious about unnecessary assumptions.

- Assumptions that make attacks stronger …

  “I inject clean images.”

  “I perturb the model’s exact input.”

  Good Attack ☺

- … can make defenses weaker.

  “OK, you only inject clean images.”

  “OK, you only perturb the exact input.”

  Bad Defense ☹

- Always consider the strongest adversary in your threat model.

Fix bugs, not attacks.

- Attacks are *potentially weak* exploits of a bug.

Buggy Scaling

- Weak: inject clean images
- Strong: inject adversarial perturbation

Fool black-box models

- Fixing weak exploits gives a false sense of security.
- How about adversarial examples?
  - Yes, we are still fixing attacks.
  - Preventing adversarial examples remain open.

Poster
Tue 19 Jul 6:30 p.m. — 8:30 p.m.
Hall E #1014

MADS&P
Security and Privacy Research Group
at UW-Madison

Thank You

Yue Gao

Ph.D. Student, University of Wisconsin–Madison

Research Interests: Trustworthy Machine Learning, Security and Privacy

Contact: gy@cs.wisc.edu

Homepage: https://pages.cs.wisc.edu/~gy

arXiv