NYU TANDON SCHOOL OF ENGINEERING

Robot Perception

Adversarial Robustness

Dr. Felix Juefei Xu

juefei.xu@nyu.edu

ROB-GY 6203, Fall 2022

Overview

- Adversarial robustness in perception
  - + White-box attack
  - + Black-box attack
  - + Universal adversarial perturbation (UAP)
  - + Physical adversarial examples
  - + Adversarial training and defense strategies
  - + Un-adversarial examples
  - + Beyond adversarial noise perturbation

+: know the concept

References

- Goodfellow, Ian J., Jonathon Shlens, and Christian Szegedy. "Explaining and harnessing adversarial examples." *arXiv preprint arXiv:1412.6572* (2014).
- Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., & Fergus, R. (2013). Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.
- Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z. B., & Swami, A. (2017, April). Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia conference on computer and communications security* (pp. 506-519).
- Kurakin, A., Goodfellow, I. J., & Bengio, S. (2018). Adversarial examples in the physical world. In *Artificial intelligence safety and security* (pp. 99-112). Chapman and Hall/CRC.
- Moosavi-Dezfooli, S. M., Fawzi, A., Fawzi, O., & Frossard, P. (2017). Universal adversarial perturbations. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1765-1773).
- Madry, A., Makelov, A., Schmidt, L., Tsipras, D., & Vladu, A. (2017). Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*.
- Salman, H., Ilyas, A., Engstrom, L., Vemprala, S., Madry, A., & Kapoor, A. (2021). Unadversarial examples: Designing objects for robust vision. *Advances in Neural Information Processing Systems*, *34*, 15270-15284.
- Guo, Q., et al. (2020). Watch out! motion is blurring the vision of your deep neural networks. *Advances in Neural Information Processing Systems*, *33*, 975-985.
- Li, Yiming, et al. "Fooling lidar perception via adversarial trajectory perturbation." *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021.

Intriguing Properties of (Deep) Neural Network

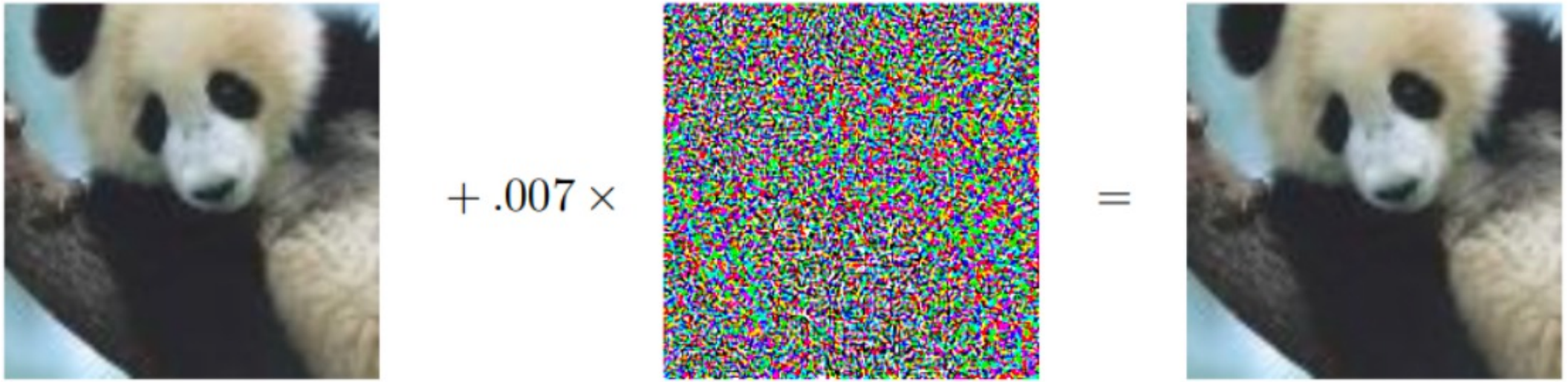
If we add deliberately optimized noise to the image, we can easily alter the classification output.

(a) (b)

Figure 5: Adversarial examples generated for AlexNet [9].(Left) is a correctly predicted sample, (center) difference between correct image, and image predicted incorrectly magnified by 10x (values shifted by 128 and clamped), (right) adversarial example. All images in the right column are predicted to be an *"ostrich, Struthio camelus"*. Average distortion based on 64 examples is 0.006508. Plase refer to `http://goo.gl/huaGPb` for full resolution images. The examples are strictly randomly chosen. There is not any postselection involved.

White-Box Adversarial Attack

- Adversarial additive noise perturbation
  - Optimize for the noise perturbation that leads to maximally erroneous classification (maximize the error loss function)
  - We freeze the model parameters and update the additive noise using back-propagation.

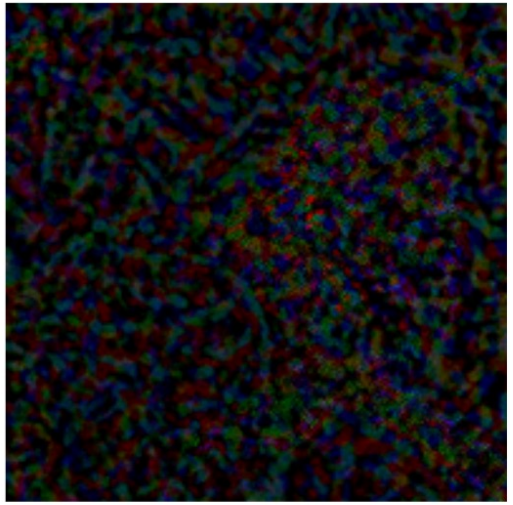


White-Box Adversarial Attack

- Adversarial additive noise perturbation
  - Have full access to the model being attacked (model parameters)
  - Gradient back-propagation is possible

Inception v3: Bird $\mathbf{X}^{\text{real}}$

+



δ

=

Inception v3: Dog

$$\arg\max_{\delta} J(\mathbf{X}^{\text{real}} + \delta, y) \text{ subject to } \|\delta\|_{\text{p}} \leq \epsilon$$

White-Box Adversarial Attack

- Adversarial additive noise perturbation
  - Targeted attack vs. untargeted attack
    - Targeted attack: mislead the model to provide the target prediction $y^* \neq y$ specified by the adversary
    - x: the original input, y: ground truth label; x*: adversarial example

$$\min_{x^*} \ell(f_\theta(x^*), y^*)$$
$$\text{s.t.} \quad d(x, x^*) \leq B$$

    - Untargeted attack: mislead the model to provide any wrong prediction (more common, easier to achieve)

$$\max_{x^*} \ell(f_\theta(x^*), y)$$
$$\text{s.t.} \quad d(x, x^*) \leq B$$

White-Box Adversarial Attack

- Adversarial additive noise perturbation
  - Many types of white-box attacks
    - Fast gradient sign method (FGSM)

Fast Gradient Sign Method (FGSM). FGSM perturbs normal examples $\mathbf{x}$ for one step by the amount of ϵ along the input gradient direction (Goodfellow et al., 2015):

$$\mathbf{x}_{adv} = \mathbf{x} + \epsilon \cdot \text{sign}(\nabla_{\mathbf{x}} \ell(h(\mathbf{x}), y)). \quad (4)$$

    - Basic iterative method (BIM) = Iterative FGSM (I-FGSM)

Basic Iterative Method (BIM). BIM (Kurakin et al., 2017) is an iterative version of FGSM. Different to FGSM, BIM iteratively perturbs the input with smaller step size,

$$\mathbf{x}^t = \left(\mathbf{x}^{t-1} + \alpha \cdot \text{sign}(\nabla_{\mathbf{x}} \ell(h(\mathbf{x}^{t-1}), y)\right), \quad (5)$$

where α is the step size, and $\mathbf{x}^t$ is the adversarial example at the t-th step ($\mathbf{x}^0 = \mathbf{x}$). The step size is usually set to $\epsilon/T \leq \alpha < \epsilon$ for overall T steps of perturbation.

Ma, X et al. (2021). Understanding adversarial attacks on deep learning based medical image analysis systems. *Pattern Recognition*, *110*, 107332.

White-Box Adversarial Attack

- Adversarial additive noise perturbation
  - Many types of white-box attacks
    - Projected gradient descent (PGD)

Projected Gradient Descent (PGD). PGD (Madry et al., 2018) perturbs a normal example $\mathbf{x}$ for a number of T steps with smaller step size. After each step of perturbation, PGD projects the adversarial example back onto the ϵ-ball of $\mathbf{x}$, if it goes beyond:

$$\mathbf{x}^t = \Pi_\epsilon\big(\mathbf{x}^{t-1} + \alpha \cdot \text{sign}(\nabla_{\mathbf{x}} \ell(h(\mathbf{x}^{t-1}), y))\big), \quad (6)$$

where α is the step size, $\Pi(\cdot)$ is the projection function, and $\mathbf{x}^t$ is the adversarial example at the t-th step ($\mathbf{x}^0 = \mathbf{x}$). Different from BIM, PGD uses random start for $\mathbf{x}^0 = \mathbf{x} + \mathcal{U}^d(-\epsilon, \epsilon)$, where $\mathcal{U}^d(-\epsilon, \epsilon)$ is the uniform distribution between $-\epsilon$ and ϵ, and of the same d dimensions as $\mathbf{x}$. PGD is normally regarded as the strongest first-order attack.

Ma, X et al. (2021). Understanding adversarial attacks on deep learning based medical image analysis systems. *Pattern Recognition*, *110*, 107332.

White-Box Adversarial Attack

- Adversarial additive noise perturbation
  - Many types of white-box attacks
    - Carlini and Wagner (CW)

Carlini and Wagner (CW) Attack. The CW attack is a state-of-the-art optimization-based attack (Carlini & Wagner, 2017). There are two versions of the CW attack: L_2 and L_∞, here we focus on the L_∞ version. According to (Madry et al., 2018), the L_∞ version of targeted CW attack can be solved by the PGD algorithm iteratively as following

$$\mathbf{x}^t = \Pi_\epsilon\left(\mathbf{x}^{t-1} - \alpha \cdot \text{sign}(\nabla_{\mathbf{x}} \hat{f}(\mathbf{x}^{t-1}))\right) \tag{7}$$

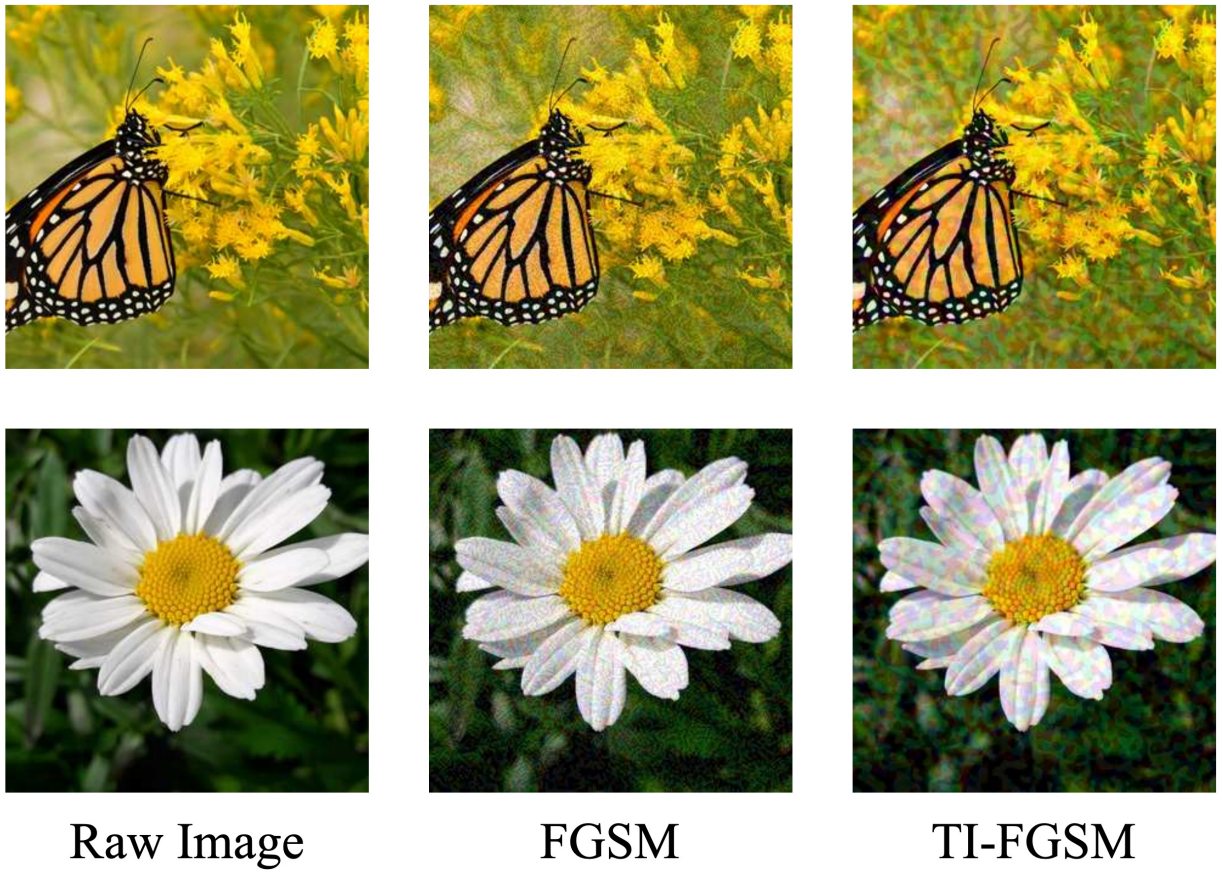
$$\hat{f}(\mathbf{x}^{t-1}) = \max\left(\mathbf{z}_y(\mathbf{x}^{t-1}, \boldsymbol{\theta}) - \mathbf{z}_{y_{max} \neq y}(\mathbf{x}^{t-1}, \boldsymbol{\theta}), -\kappa\right), \tag{8}$$

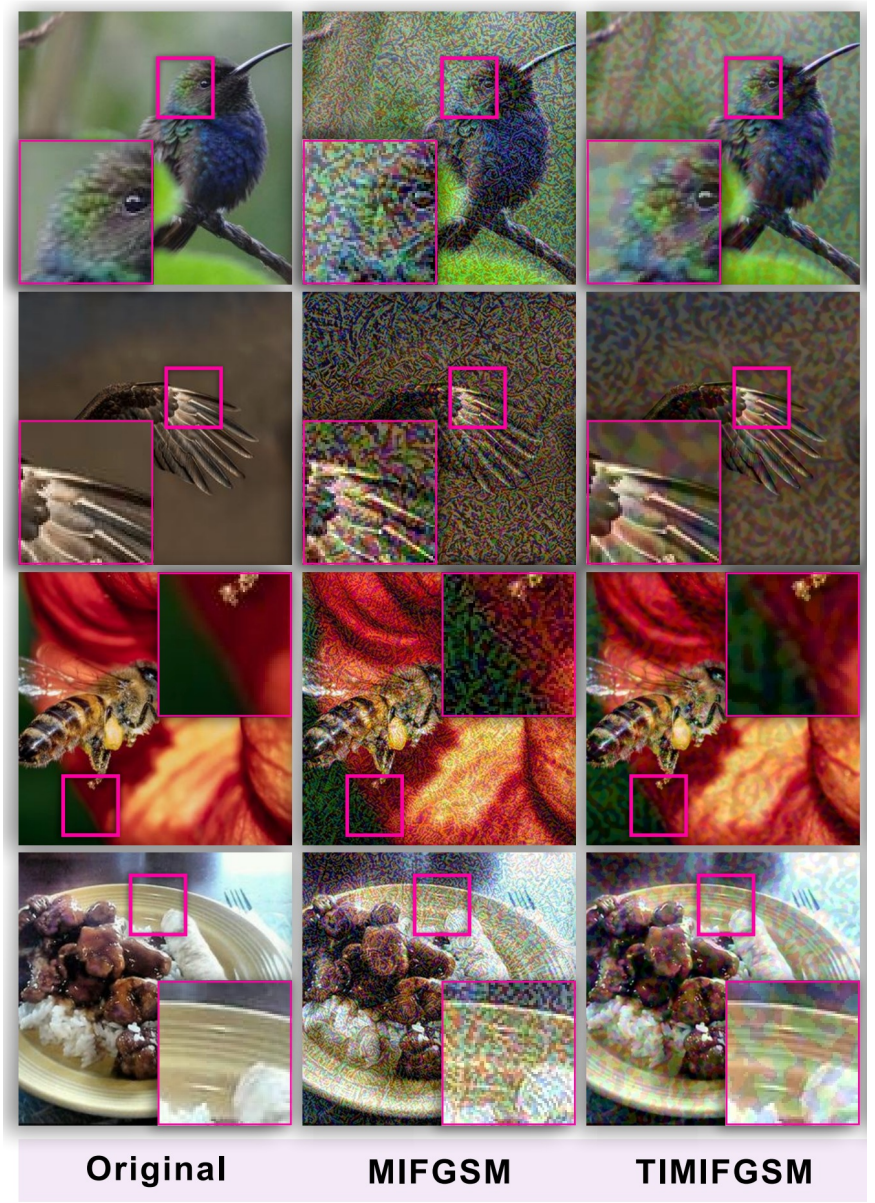
where $\hat{f}(\cdot)$ is the surrogate loss for the constrained optimization problem defined in Eqn. (3), $\mathbf{z}_y$ is the logits with respect to class y, $\mathbf{z}_{y_{max} \neq y}$ is the maximum logits of other classes, and κ is a parameter controls the confidence of the attack.

Ma, X et al. (2021). Understanding adversarial attacks on deep learning based medical image analysis systems. *Pattern Recognition*, *110*, 107332.

White-Box Adversarial Attack

- Visualization of various white-box attacks

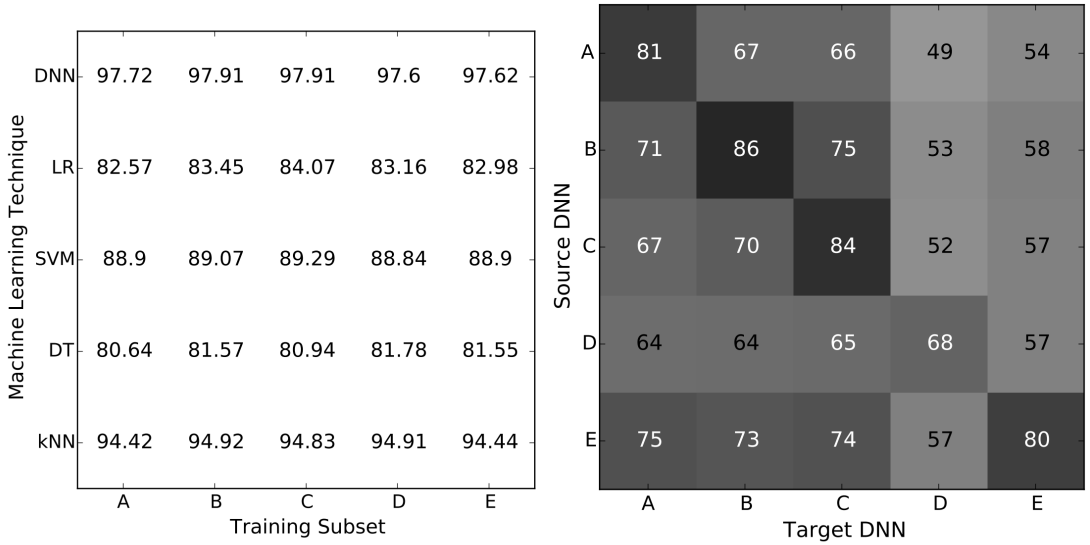




Ma, X et al. (2021). Understanding adversarial attacks on deep learning based medical image analysis systems. *Pattern Recognition*, *110*, 107332.
Guo, Qing, et al. "Watch out! motion is blurring the vision of your deep neural networks." Advances in *Neural Information Processing Systems 33 (2020)*: 975-985.

Black-Box Adversarial Attack

- White-box attacks are relatively easy
- Black-box attacks are much harder
  - Adversary cannot have access to model weights
  - Adversary cannot query the target model
- Observation: adversarial examples generated for one model may transfer to another model.



(a) Model Accuracies (b) DNN models

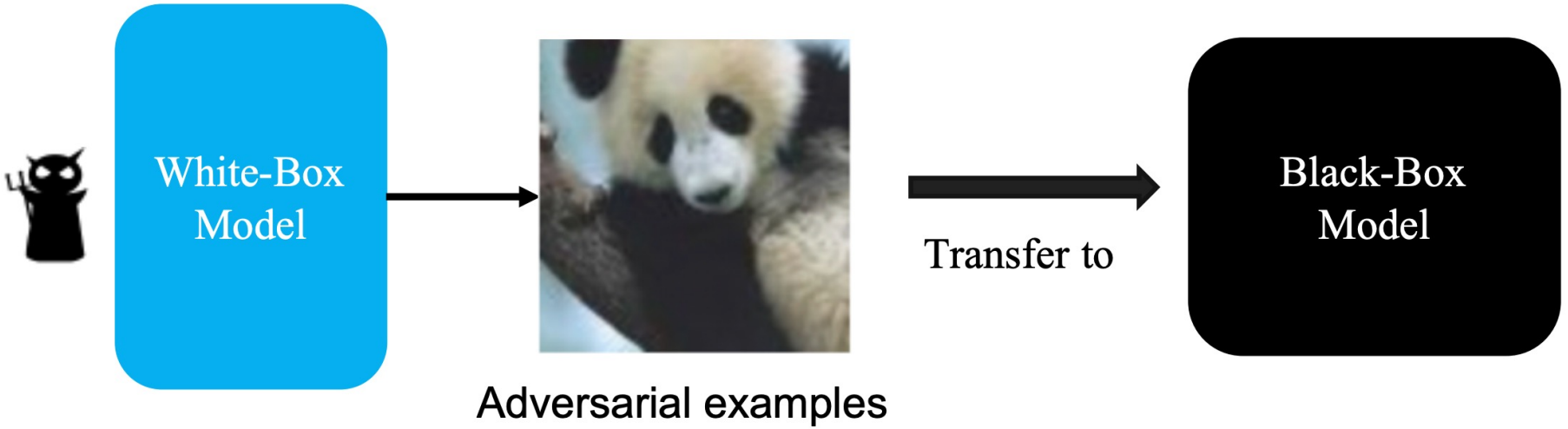
Figure (a) reports the accuracy rates of the 25 models used, computed on the MNIST test set.

Figures (b) are such that cell (i, j) reports the intra-technique transferability between models i and j, i.e. the percentage of adversarial samples produced using model i misclassified by model j.

Papernot et al. Transferability in Machine Learning: from Phenomena to Black-box Attacks using Adversarial Examples.

Black-Box Adversarial Attack

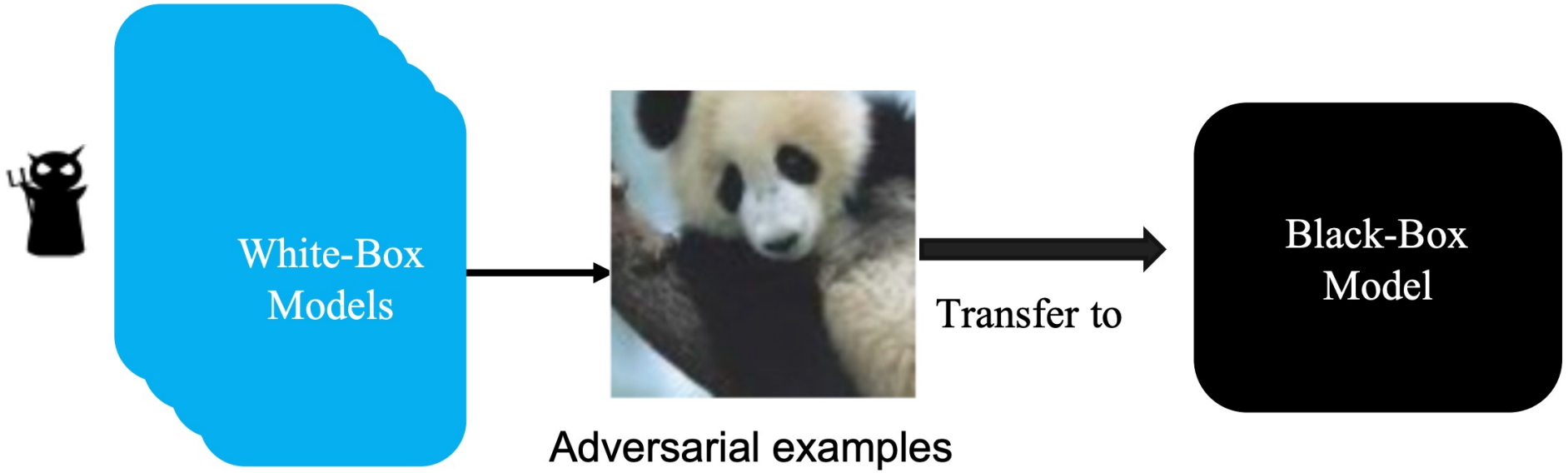
- Observation: adversarial examples generated for one model may transfer to another model.
  - Solution 1: Black-box attack based on transferability.



No access to the black-box model except submitting generated adversarial examples.

Black-Box Adversarial Attack

- Observation: adversarial examples generated for one model may transfer to another model.
  - Solution 2: Ensemble Black-box attack based on transferability.



Intuition: If an adversarial example can fool N-1 white-box models, it might transfer better to the N-th black-box model.

https://advmlincv.github.io/cvpr21-tutorial/slides/lecture1.pdf
Liu, Chen, Liu, Song. Delving into Transferable Adversarial Examples and Black-box Attacks, ICLR 2017.

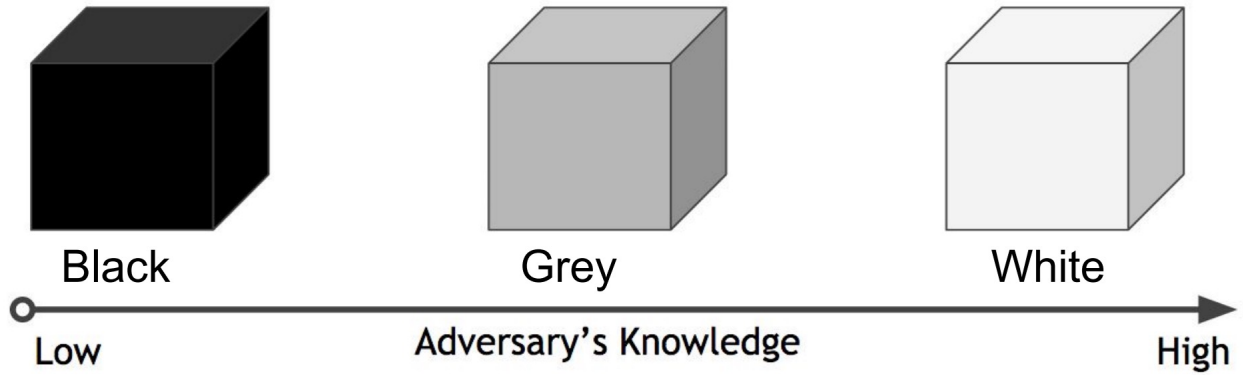
Black-Box Adversarial Attack

- Active research area: designing highly transferrable adversarial examples for black-box attacks.

- Can one incorporate “transferability” into the loss / optimization?
- Can model interpretation / explainability help create highly transferable adversarial examples?
- Can one look beyond additive noise perturbations to craft highly transferable adversarial examples?

Grey-Box Adversarial Attack

- Grey-box attack is somewhere in between black-box and white-box attack
- Adversary's knowledge may include:
  - Network details: architecture, weights, layers, etc.
  - Training details: optimizer, learning scheduling and curriculum, checkpoints, etc.
  - Data details: sometimes even actual data and/or data distribution, etc.
- The adversary utilizes available information to identify the feature space where the model may be vulnerable, i.e, for which the model has a high error rate.
  - Then the adversary can make full use of the network information to carefully craft adversarial examples.



https://phibenz.github.io/talk/2d3d.ai/2d3dai_adversarial_transferability_and_beyond.pdf

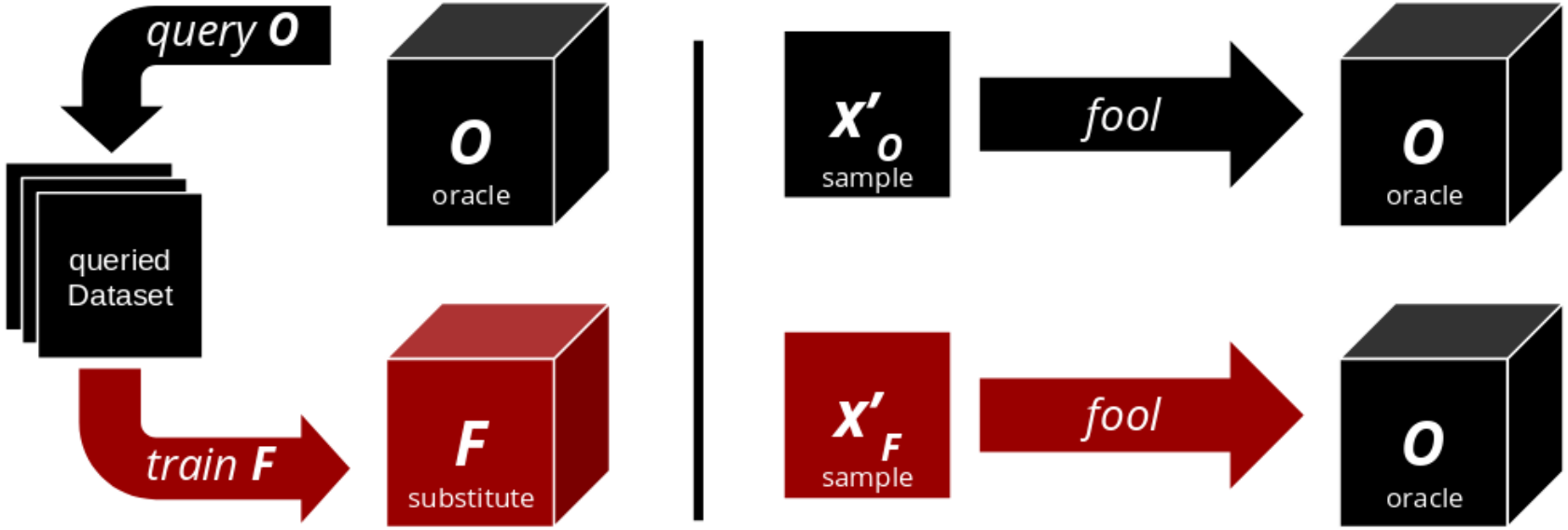
Grey-Box Adversarial Attack

- Grey-box attack is somewhere in between black-box and white-box attack
- Adversary may have partial access to the model, e.g.:
  - Knows the architecture: how many layers, activation, normalization layers, etc.
  - Knows the model family: ResNet, MobileNet, MobileNet-v2, WideResNet, ViT, etc.
  - But not the actual weights

- Adversary may be able to query the target model, i.e.:
  - Knows the target model prediction output, i.e., adversary can gather (x,y) pairs.

Grey-Box Adversarial Attack

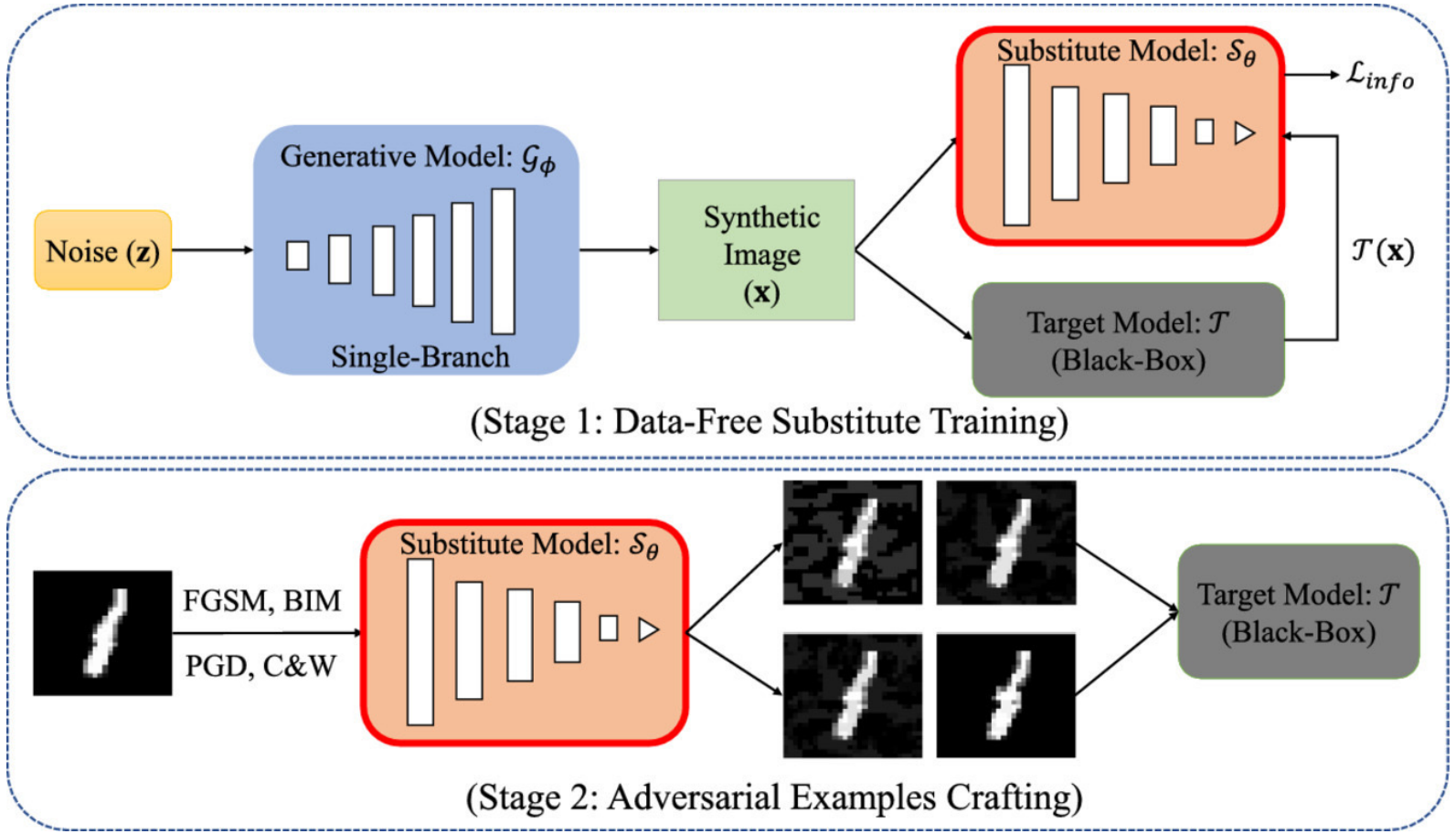
- Query-based surrogate (substitute) model attack
  - **With actual data**
    - Train a surrogate model F to mimic the behavior of target model O
    - Craft adversarial examples x using F
    - Transfer attack O using x
  - Without actual data



https://ai.hdm-stuttgart.de/news/2020/selected-topics-1-adversarial-attacks-3-black-box-attacks/

Grey-Box Adversarial Attack

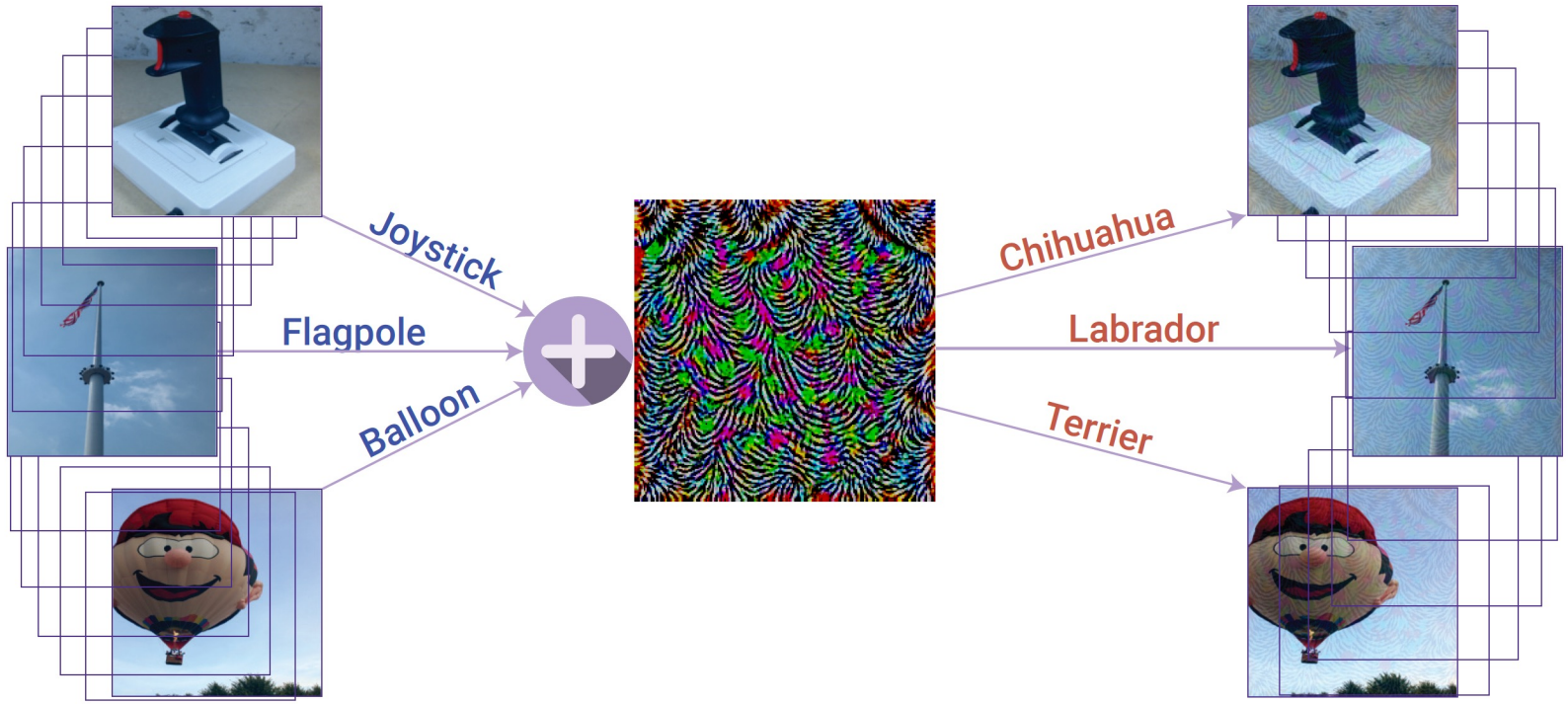
- Query-based surrogate (substitute) model attack
  - With actual data
  - **Without actual data**
    - GAN for data generation

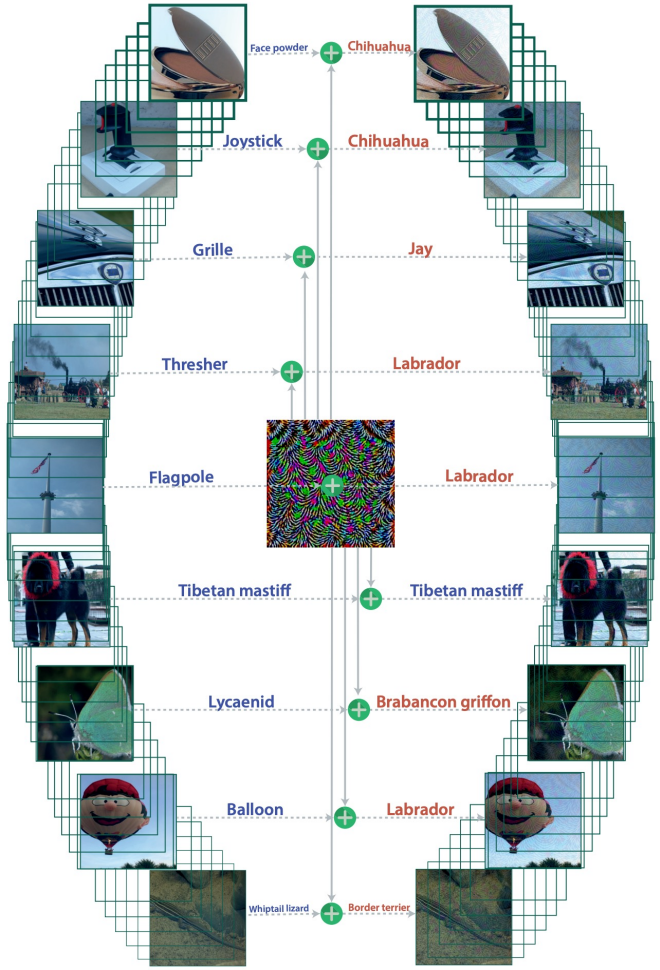


Yu, Mengran, and Shiliang Sun. "FE-DaST: Fast and effective data-free substitute training for black-box adversarial attacks." *Computers & Security* 113 (2022): 102555.

Universal Adversarial Perturbation (UAP)

- A universal perturbation (noise pattern) that can used to make a wide set of images adversarial (for a given classifier)



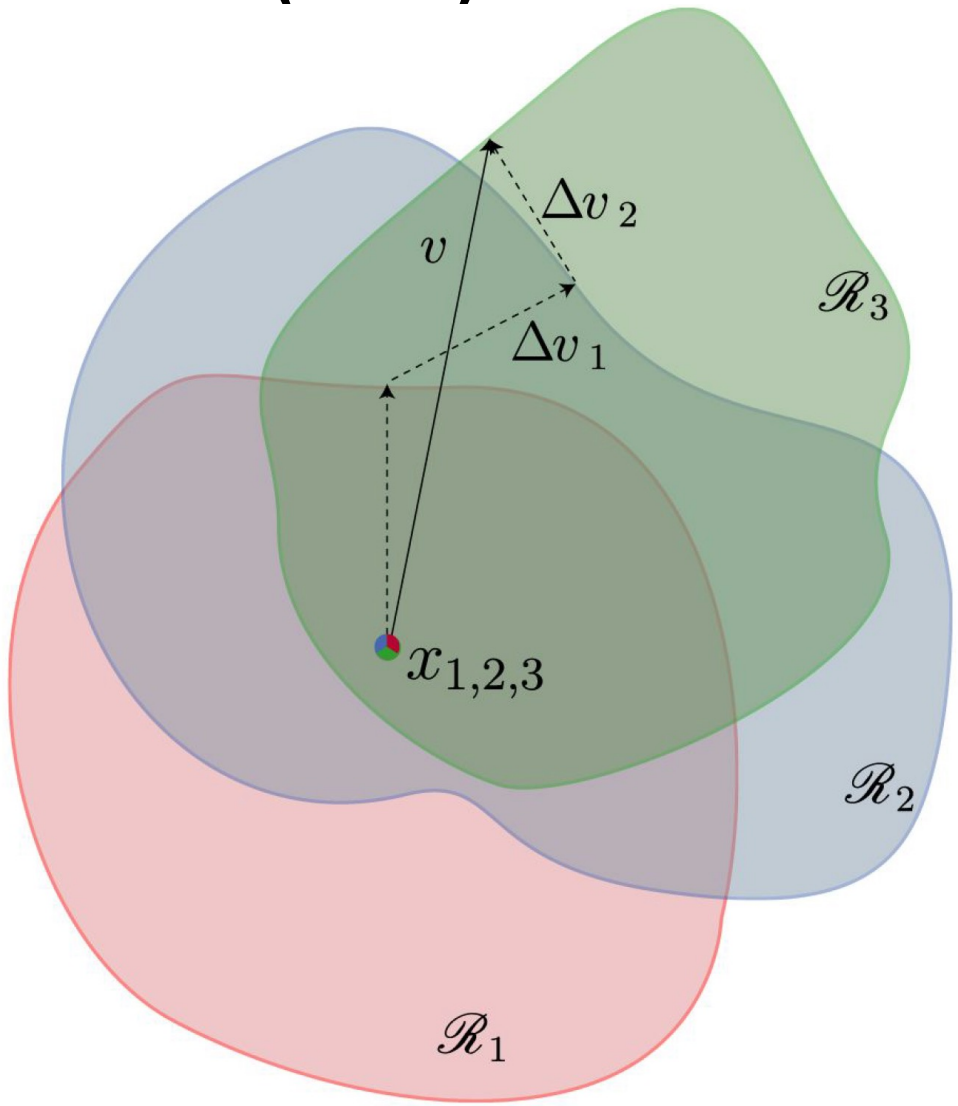


Moosavi-Dezfooli, Seyed-Mohsen, et al. "Universal adversarial perturbations." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017.

Universal Adversarial Perturbation (UAP)

- Finding universal perturbations

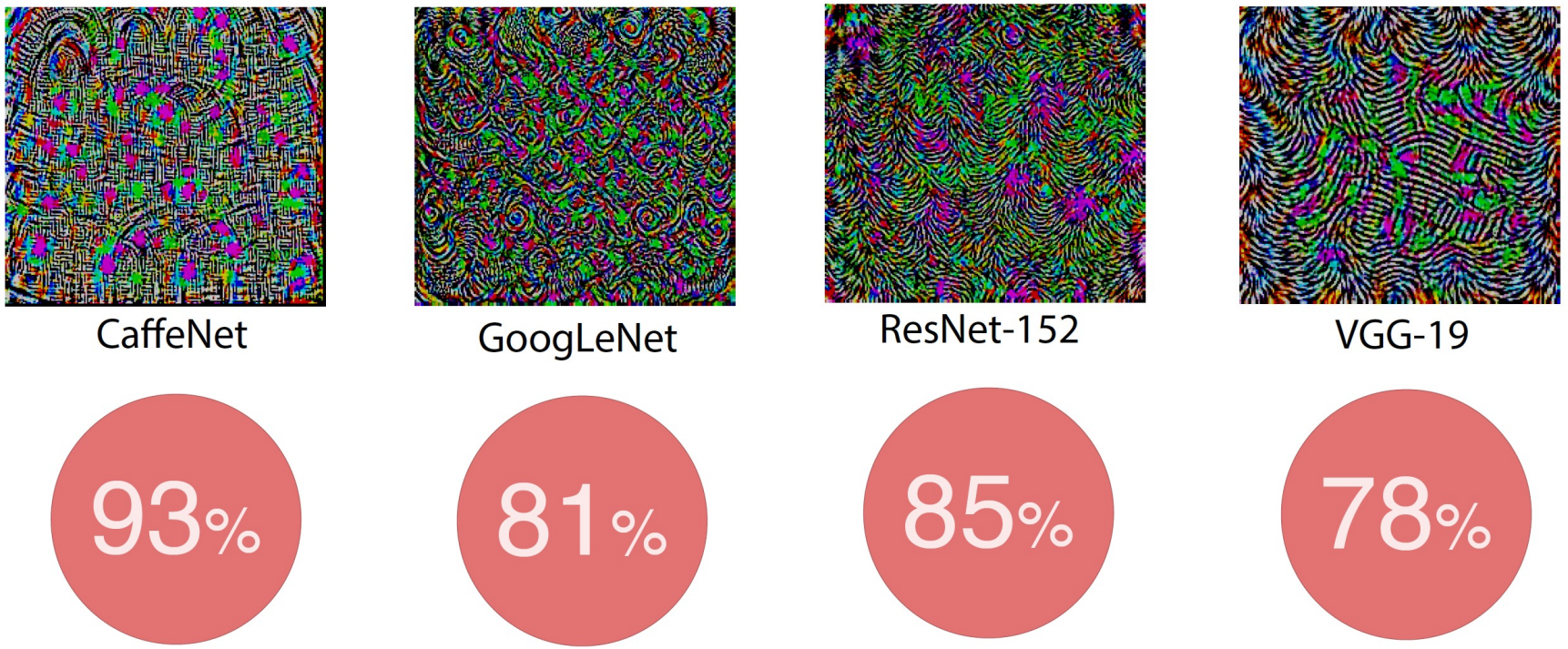
Let $X = \{x_1, \ldots, x_m\}$ be training points.



Moosavi-Dezfooli, Seyed-Mohsen, et al. "Universal adversarial perturbations." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017.

Universal Adversarial Perturbation (UAP)

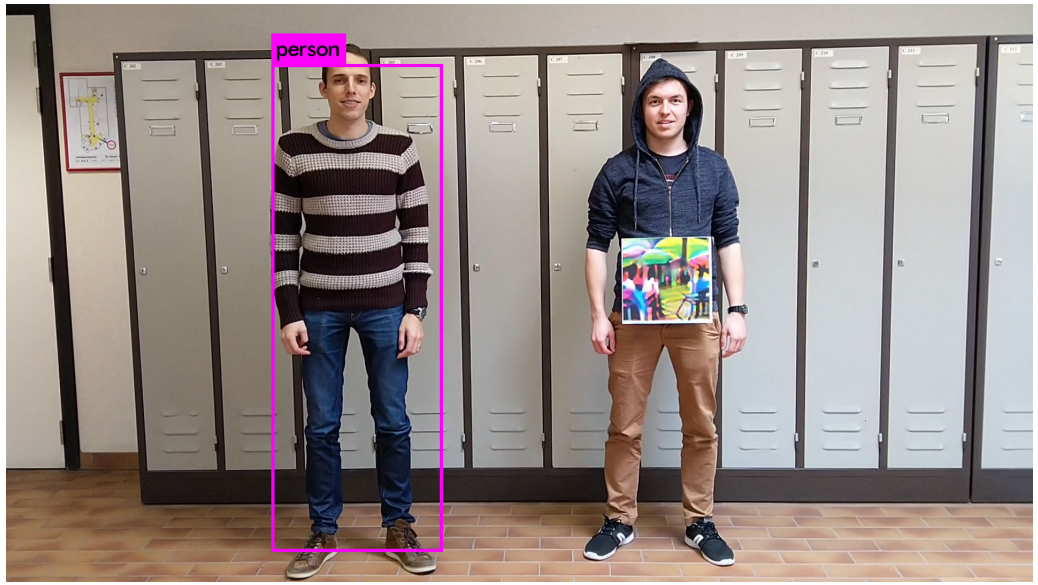
- Very high fooling rate
  - One DNN model: one universal perturbation pattern.
  - Doubly universal perturbations: Generalization of universal perturbations across networks

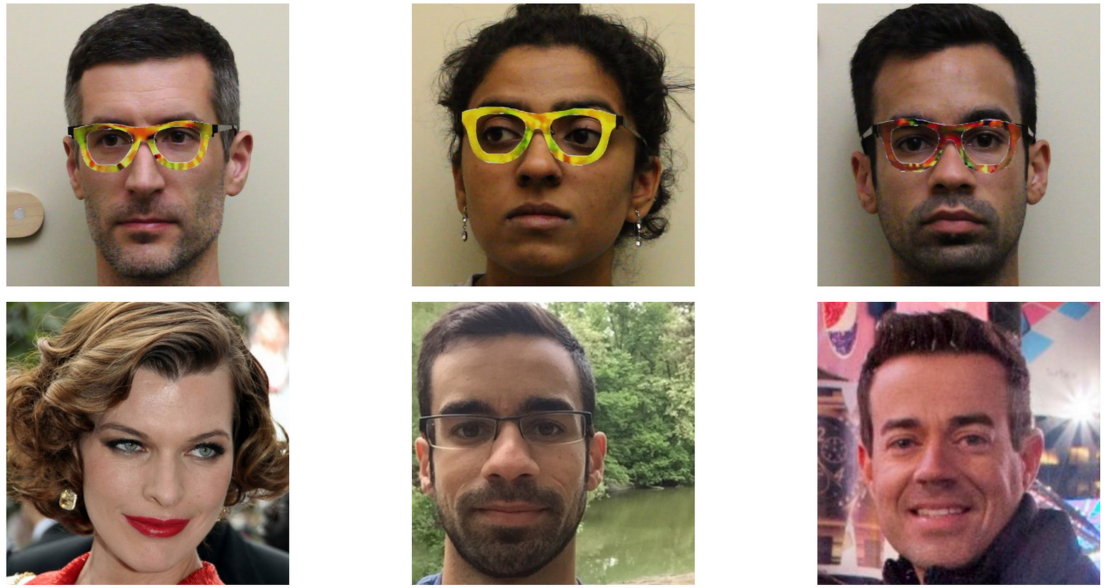


Moosavi-Dezfooli, Seyed-Mohsen, et al. "Universal adversarial perturbations." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017.

Physical Adversarial Examples

- Physically realizable perturbations for RGB perception





Physical Adversarial Examples

- Physically realizable perturbations for RGB perception

Duan, Ranjie, et al. "Adversarial laser beam: Effective physical-world attack to dnns in a blink." *Proceedings of the CVPR*. 2021.

Physical Adversarial Examples

- Physically realizable perturbations for LiDAR perception

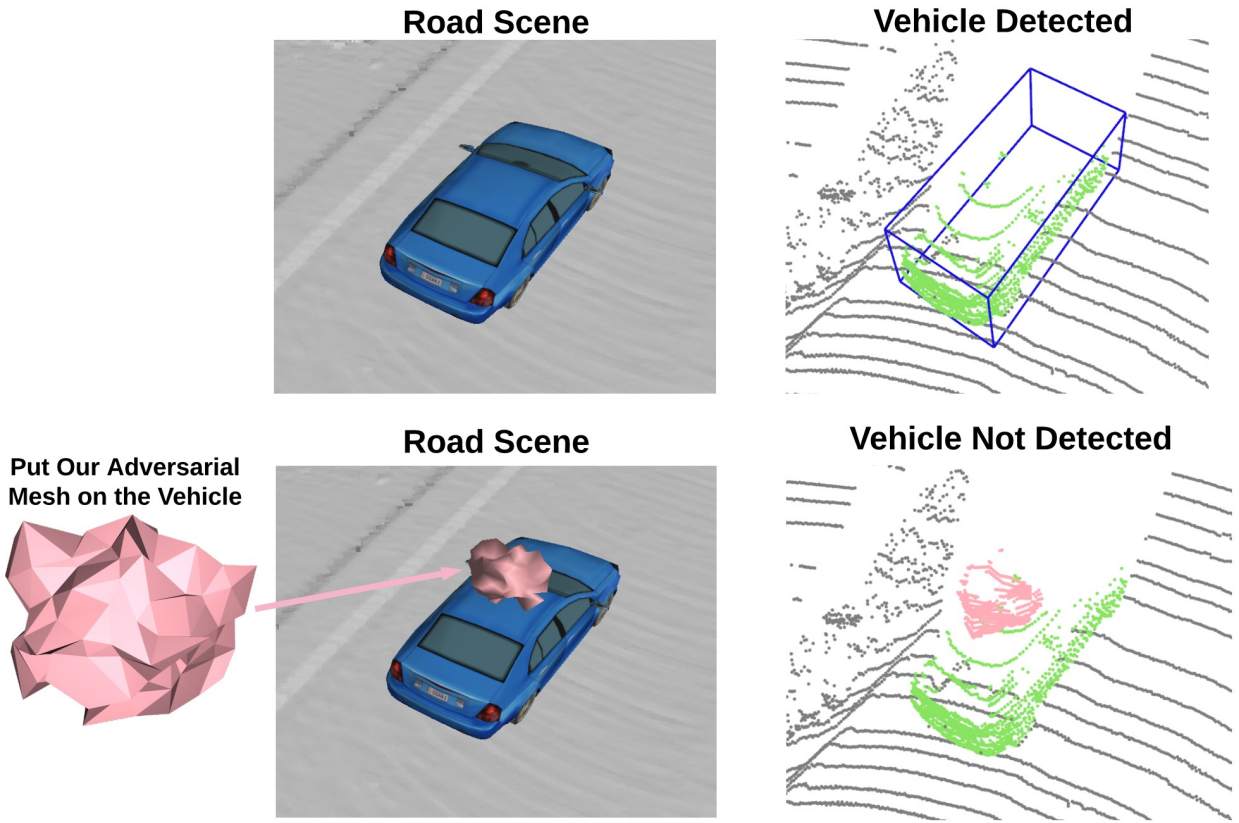


Figure 1: In this work we produce a physically realizable adversarial object that can make vehicles "invisible". After placing the object on the rooftop of a target vehicle, the vehicle will no longer be detected by a LiDAR detector.

Tu, James, et al. "Physically realizable adversarial examples for lidar object detection." *Proceedings of the CVPR*. 2020.

Defense Strategies against Adversarial Attacks

- Data purification (denoising) of the adversarial image x

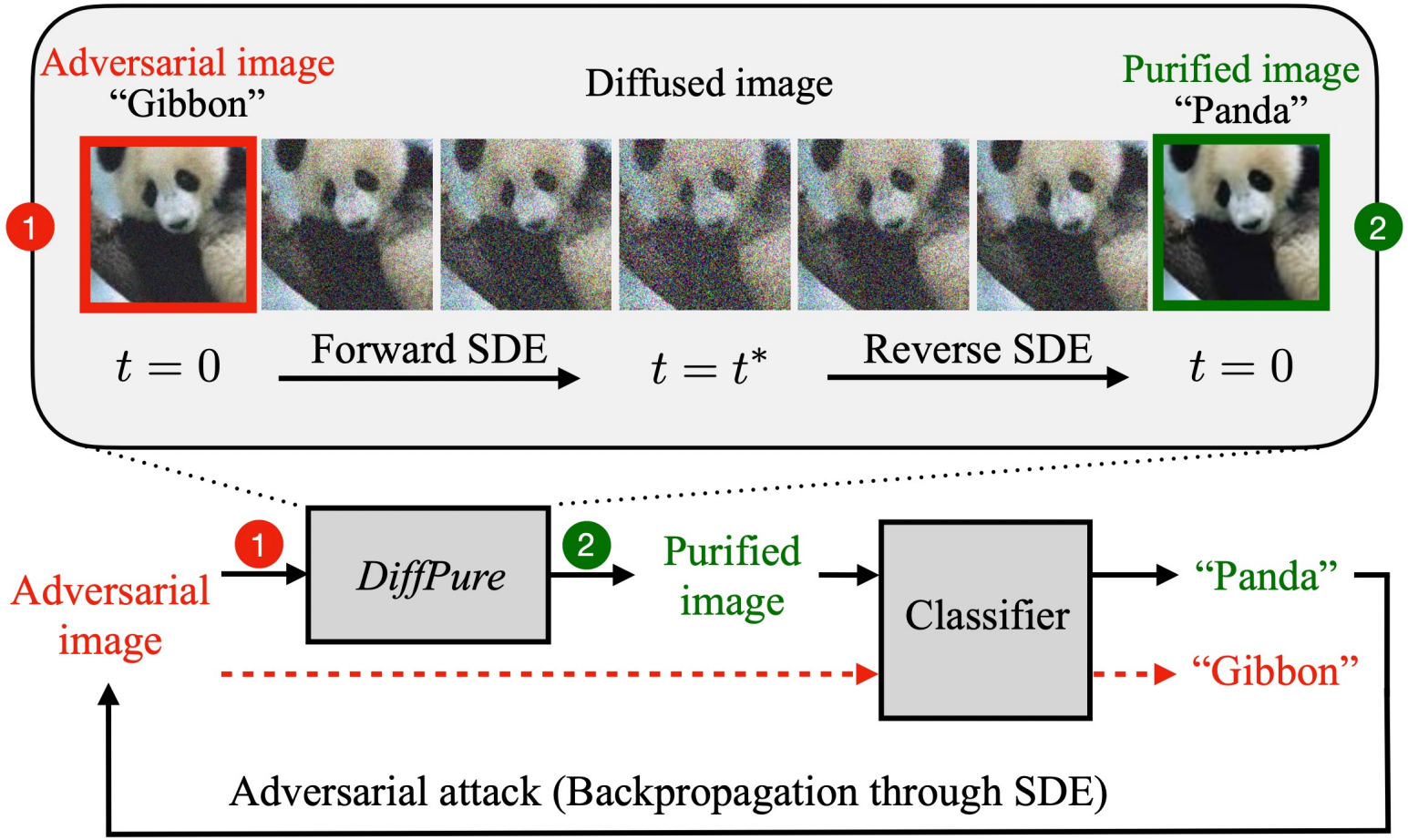


Figure 1. An illustration of *DiffPure*. Given a pre-trained diffusion model, we add noise to adversarial images following the forward diffusion process with a small diffusion timestep t^* to get diffused images, from which we recover clean images through the reverse denoising process before classification. Adaptive attacks backpropagate through the SDE to get full gradients of our defense system.

Nie, Weili, et al. "Diffusion Models for Adversarial Purification." *arXiv preprint arXiv:2205.07460* (2022).

Defense Strategies against Adversarial Attacks

- Data purification (denoising) of the adversarial image x

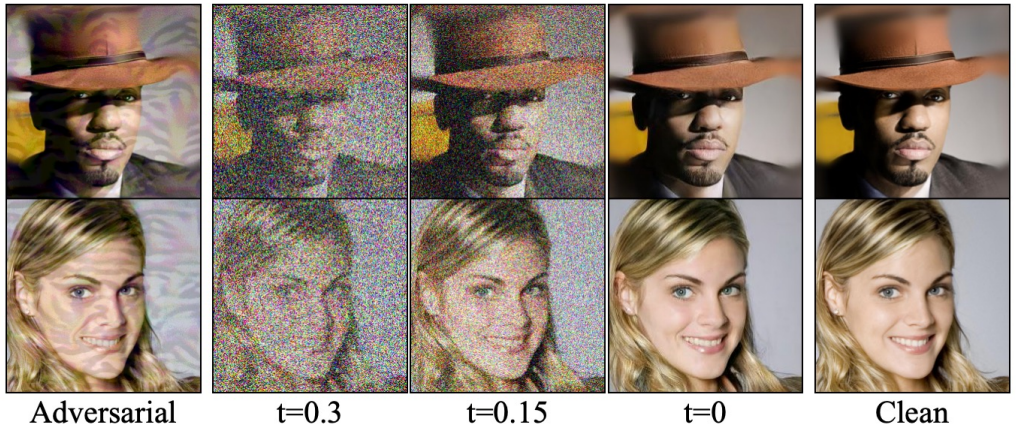


Figure 2. Our method purifies adversarial examples (first column) produced by attacking attribute classifiers using PGD ℓ_∞ ($\epsilon = 16/255$), where $t^* = 0.3$. The middle three columns show the results of the SDE in Eq. (4) at different timesteps, and we observe the purified images at t=0 match the clean images (last column). Better zoom in to see how we remove adversarial perturbations.

Nie, Weili, et al. "Diffusion Models for Adversarial Purification." *arXiv preprint arXiv:2205.07460* (2022).

Defense Strategies against Adversarial Attacks

- Data purification (denoising) of the adversarial image x

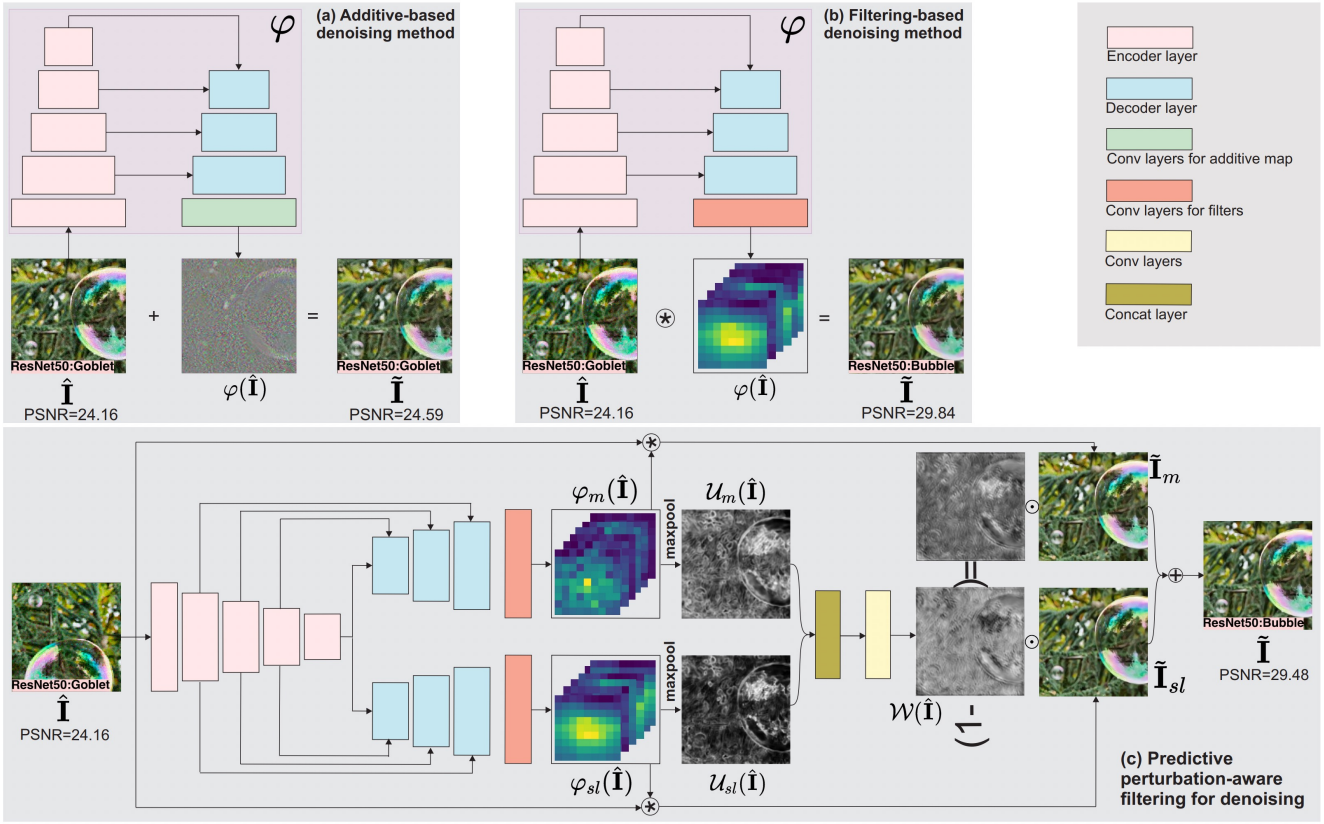


Figure 1: Architectures of additive-based pixel denoising (*i.e.*, (a)), filtering-based pixel denoising (*i.e.*, (b)), and the proposed predictive perturbation-aware filtering (*i.e.*, (c)) for adversarial robustness enhancement.

Huang, Yihao, et al. "AdvFilter: Predictive Perturbation-aware Filtering against Adversarial Attack via Multi-domain Learning." *Proceedings of the 29th ACM MM*. 2021.

Defense Strategies against Adversarial Attacks

- Data purification (denoising) of the adversarial image x

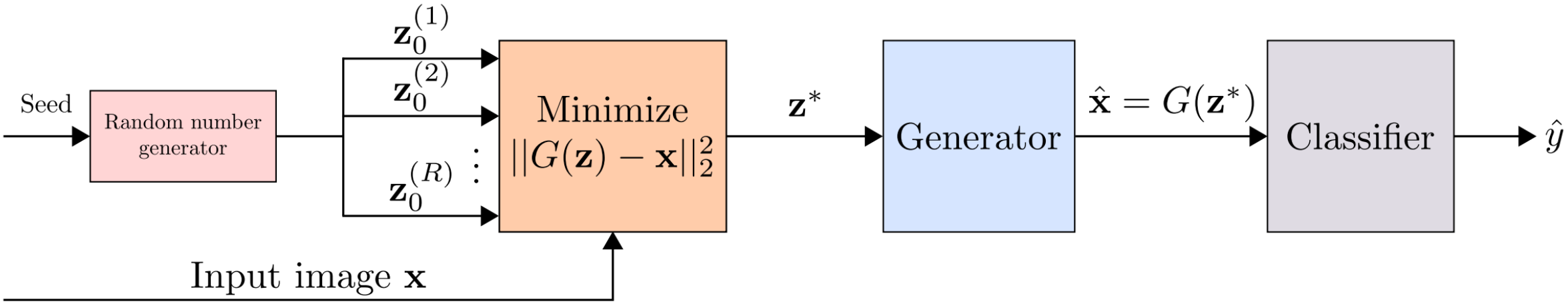


Figure 1: Overview of the Defense-GAN algorithm.

Samangouei, Pouya, et al "Defense-gan: Protecting classifiers against adversarial attacks using generative models." *arXiv preprint arXiv:1805.06605* (2018).

Defense Strategies against Adversarial Attacks

- Defense: one of the strongest defense mechanism is Adversarial Training (AT), with a min-max game.
  - Inner max loop: on the fly, generate fresh batches of additive adversarial examples (x+ δ) that maximizes the loss, with model parameters θ fixed.
  - Outer min loop: update the model parameters θ to minimize the the loss by training with (optional) clean and adversarially generated data batches.

$$\theta^*_{robust} = \arg\min_{\theta} \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\max_{\|\delta\|_2 \leq \varepsilon} \mathcal{L}_\theta(x+\delta, y) \right]$$

- Can we go beyond additive perturbation to obtain the adversarial examples (AE)?

Un-adversarial Examples

- Can we use the “adversarial optimization” idea to generate perturbation that boosts perception performance?

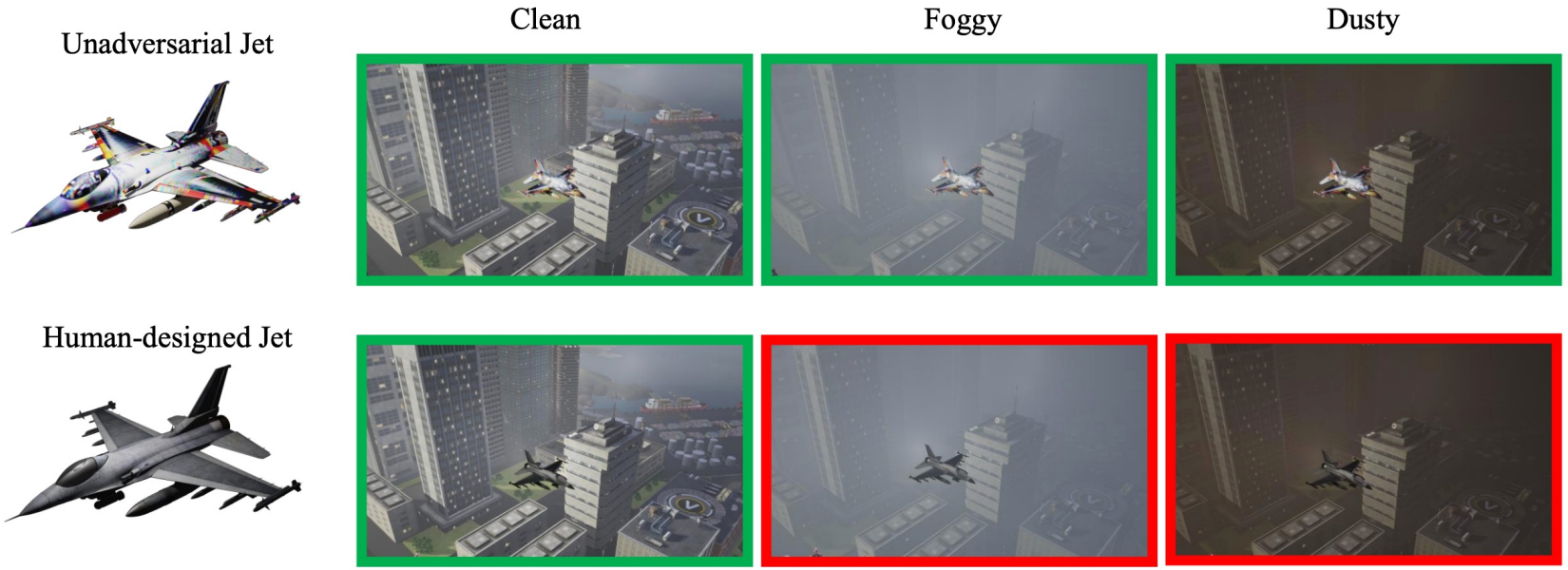


Figure 1: We demonstrate that optimizing objects (e.g., the pictured jet) for pre-trained neural networks can boost performance and robustness on computer vision tasks. Here, we show an example of classifying an unadversarial jet and a standard jet using a pretrained ImageNet model. The model correctly classifies the unadversarial jet even under bad weather conditions (e.g., foggy or dusty), whereas it fails to correctly classify the standard jet.

Salman, Hadi, et al. "Unadversarial examples: Designing objects for robust vision." *Advances in Neural Information Processing Systems* 34 (2021): 15270-15284.

Un-adversarial Examples

- Can we use the “adversarial optimization” idea to generate perturbation that boosts perception performance?

Unadv patch

(a) An example unadversarial patch designed for the “tiger” class.

Unadv texture

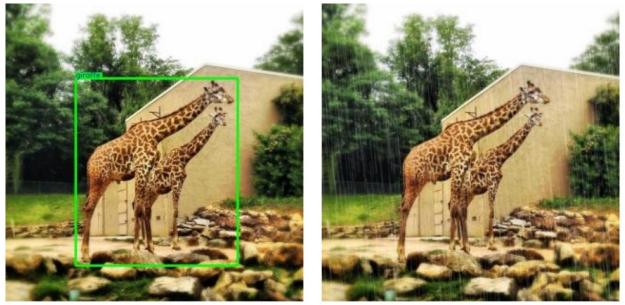
(b) An example unadversarial texture designed for a jet 3D mesh (class “warplane”) and applied to rendered city backgrounds.

Figure 2: Examples of the two considered methods for constructing unadversarial objects.

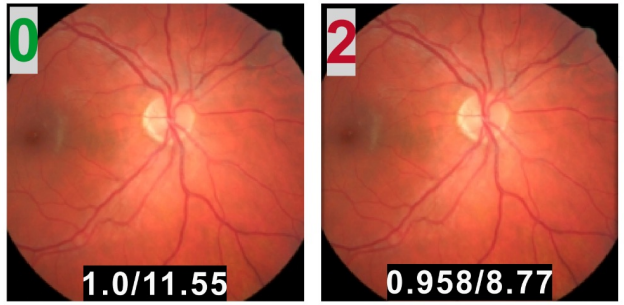
Salman, Hadi, et al. "Unadversarial examples: Designing objects for robust vision." *Advances in Neural Information Processing Systems* 34 (2021): 15270-15284.

Beyond Adversarial Noise Perturbation

adversarial motion blur



adversarial rain



adversarial exposure

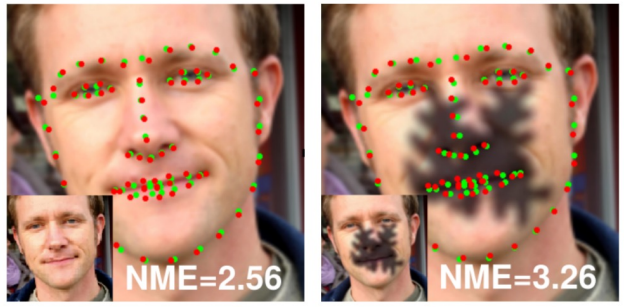
adversarial defocus blur

adversarial haze

joint adversarial noise & exposure

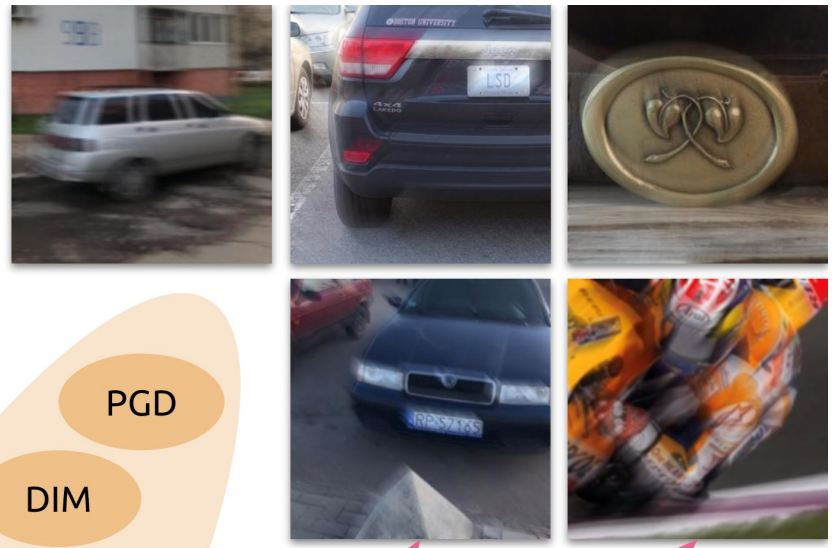
adversarial vignetting

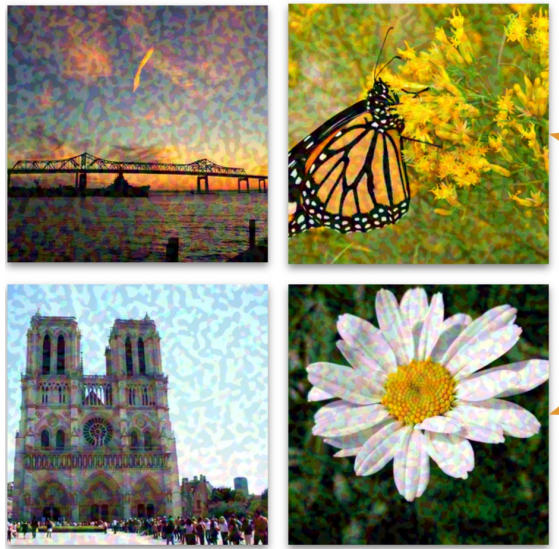
adversarial relighting



adversarial shadow

Adversarial Motion Blur Attack





PGD

DIM

Adv. Motion Blur

Clean

FGSM

TIFGSM

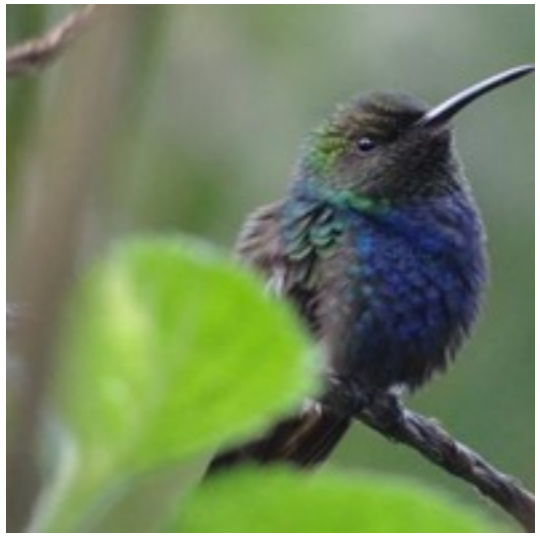
C&W

Motion Blurred
$|| \delta ||_p \leq \epsilon$

Adversarially Perturbed
$|| \delta ||_p \leq \epsilon$

Adversarial Motion Blur Attack

- Motion blur commonly occurs in practical image perception systems.
  - Caused by motion of objects or camera
  - Pretty stealthy: even large blur perturbations cannot be easily recognized as attacks.



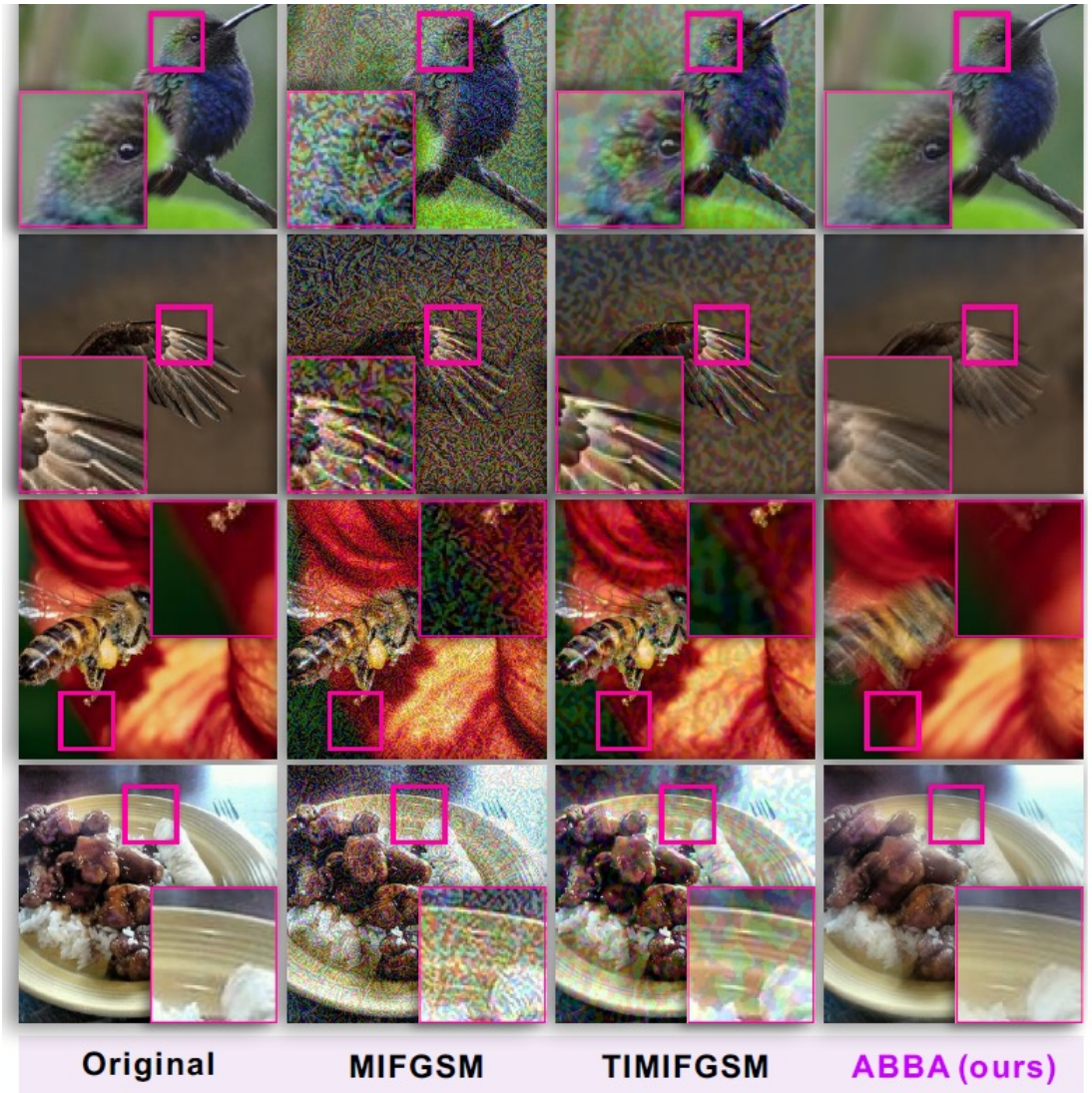
Real
Inception v3: Bird

Adversarial
Inception v3: Car

Difference GIF

Guo, Qing, et al. "Watch out! motion is blurring the vision of your deep neural networks." NeurIPS (2020): 975-985.

Adversarial Motion Blur Attack



Guo, Qing, et al. "Watch out! motion is blurring the vision of your deep neural networks." NeurIPS (2020): 975-985.

Adversarial Motion Blur Attack

Kernel-prediction-based adversarial attack: $\mathrm{ABBA}_{\mathrm{pixel}}$

- Given a real example, we aim to generate an adversarial example via kernels: $\mathcal{K} = \{\mathbf{k}_p | \forall p \text{ in } \mathbf{X}^{\mathrm{real}}\}$, each pixel has its exclusive kernel.

real example adversarial example

$$\mathbf{X}_p^{\mathrm{adv}} = \mathrm{g}(\mathbf{X}_p^{\mathrm{real}}, \mathbf{k}_p, \mathcal{N}(p)) = \sum_{q \in \mathcal{N}(p)} \mathbf{X}_q^{\mathrm{real}} k_{pq},$$

- Predict kernels by solving:

$$\arg\max_{\mathcal{K}} J(\{\sum_{q \in \mathcal{N}(p)} \mathbf{X}_q^{\mathrm{real}} k_{pq}\}, y)$$

$$\text{subject to } \forall p, \|\mathbf{k}_p\|_0 \leq \epsilon,$$

$$\max(\mathbf{k}_p) = k_{pp}, \sum_{q \in \mathcal{N}(p)} k_{pq} = 1,$$

Loss function for image classification

The number of valid kernel elements

[1, N], controls the upper bound

The main information is well preserved

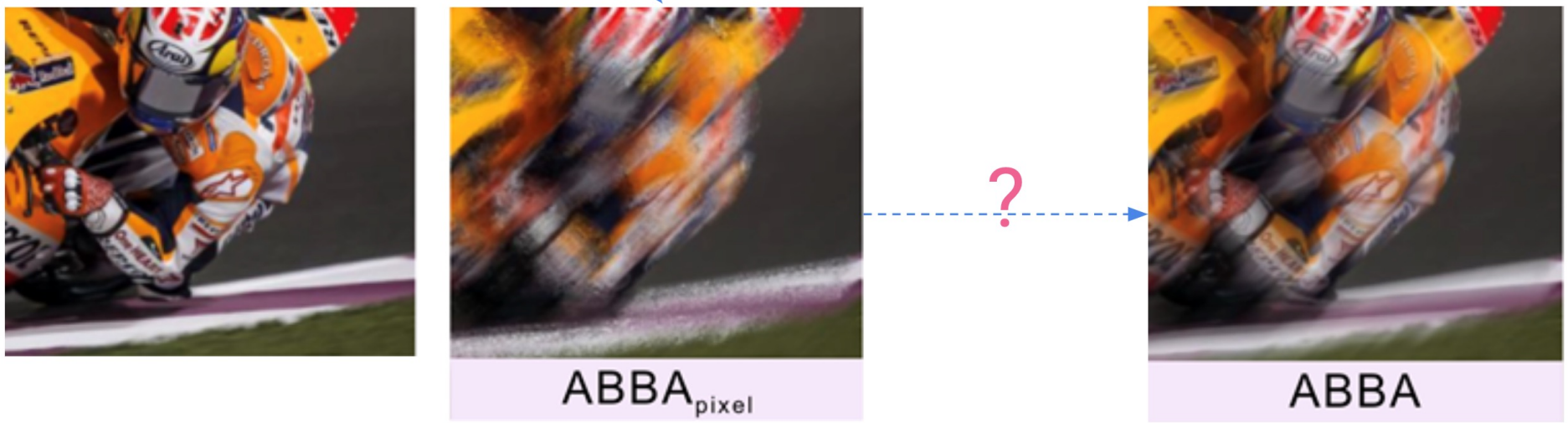
Guo, Qing, et al. "Watch out! motion is blurring the vision of your deep neural networks." NeurIPS (2020): 975-985.

Adversarial Motion Blur Attack

Kernel-prediction-based adversarial attack: $\mathrm{ABBA}_{\mathrm{pixel}}$

- New challenge:

  Tuning each pixel's kernel independently leads to noise-like results

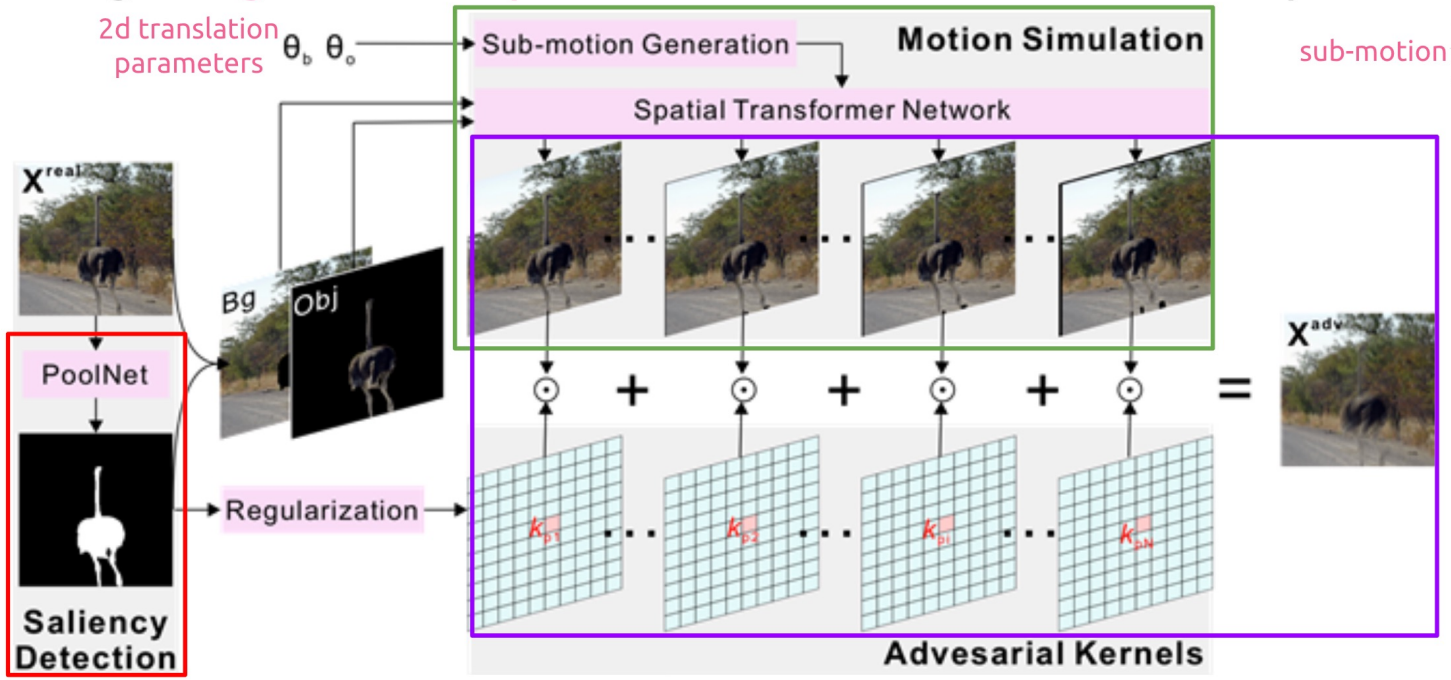


Guo, Qing, et al. "Watch out! motion is blurring the vision of your deep neural networks." NeurIPS (2020): 975-985.

Adversarial Motion Blur Attack

Motion-based adversarial blur attack: ABBA

- Simulating the generation process of motion blur via kernel prediction



sub-motion $\{i\Delta\theta_{\mathrm{o}}, |i \in [1, N]\}$
$\{i\Delta\theta_{\mathrm{b}}, |i \in [1, N]\}$
$\Delta\theta_{\mathrm{o}} = \theta_{\mathrm{o}}/N$
$\Delta\theta_{\mathrm{b}} = \theta_{\mathrm{b}}/N$

The set of kernels of all pixels
$\mathcal{K} = \{\mathbf{k}_p | \forall p \text{ in } \mathbf{X}^{\mathrm{real}}\}$

$$\mathbf{X}_p^{\mathrm{adv}} = \mathrm{g}(\mathbf{X}_p^{\mathrm{real}}, \mathbf{S}, \mathbf{k}_p, \mathcal{N}(p)) = \sum_{q=\mathcal{N}(p,i), i\in[1,N]} (\mathbf{X}_q^{S,i\Delta\theta_{\mathrm{o}}} + \mathbf{X}_q^{1-S,i\Delta\theta_{\mathrm{b}}}) k_{pq}$$

$$\arg\max_{\mathcal{K},\theta_{\mathrm{o}},\theta_{\mathrm{b}}} J(\{\sum_{\substack{q=\mathcal{N}(p,i)\\ i\in[1,N]}} (\mathbf{X}_q^{S,i\Delta\theta_{\mathrm{o}}} + \mathbf{X}_q^{1-S,i\Delta\theta_{\mathrm{b}}}) k_{pq}\}, y)$$

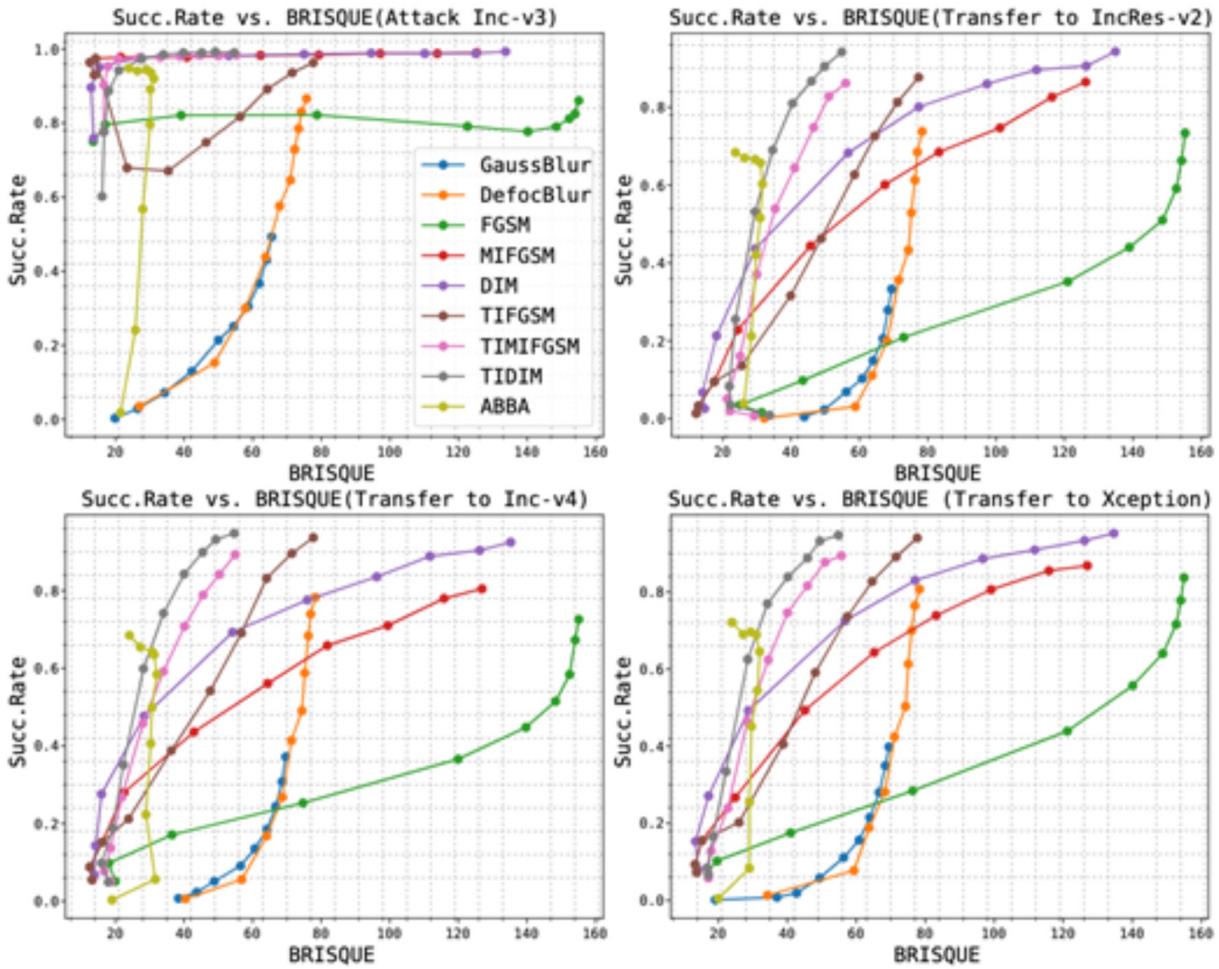
$$\text{subject to } \forall p, \|\mathbf{k}_p\|_0 \le \epsilon, \max(\mathbf{k}_p) = k_{pp}, \sum_{q\in\mathcal{N}(p)} k_{pq} = 1$$

$$\forall p, q, \mathbf{k}_p = \mathbf{k}_q, \text{ if } \mathbf{S}(p) = \mathbf{S}(q), \|\theta_{\mathrm{o}}\|_\infty \le \epsilon_\theta, \|\theta_{\mathrm{b}}\|_\infty \le \epsilon_\theta.$$

$\mathrm{T}(\mathbf{X}^{\mathrm{real}} \odot \mathbf{S}, \theta_{\mathrm{o}}) \rightarrow \mathbf{X}^{S,\theta_{\mathrm{o}}}$ $\mathrm{T}(\mathbf{X}^{\mathrm{real}} \odot (1-\mathbf{S}), \theta_{\mathrm{b}}) \rightarrow \mathbf{X}^{1-S,\theta_{\mathrm{b}}}$

Adversarial Motion Blur Attack

Comparison with baselines on image quality:

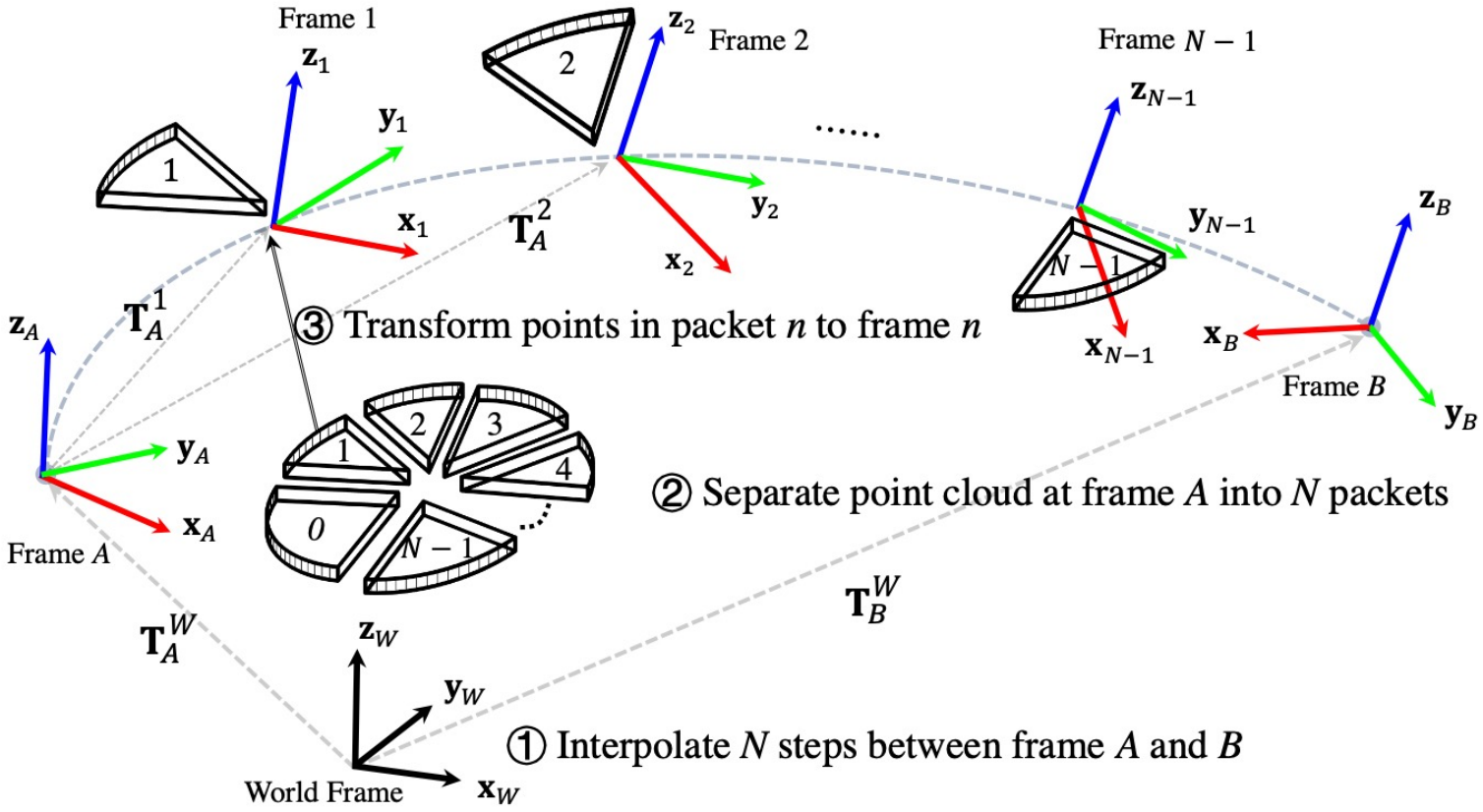


- We analyze BRISQUE[1] and success rate of attacks. Smaller BRISQUE corresponds to more natural images.
- In general, the quality of images generated by all baseline methods gradually gets worse as their success rate becomes larger. In contrast, ABBA can produce visually natural adversarial examples with high attack success rate.
- When transferring the adversarial examples to other models, success rate of additive-perturbation-based attacks decrease sharply, while the blur-based attacks are not impacted so largely.

Guo, Qing, et al. "Watch out! motion is blurring the vision of your deep neural networks." NeurIPS (2020): 975-985.

Fooling LiDAR Perception via Adversarial Trajectory Perturbation

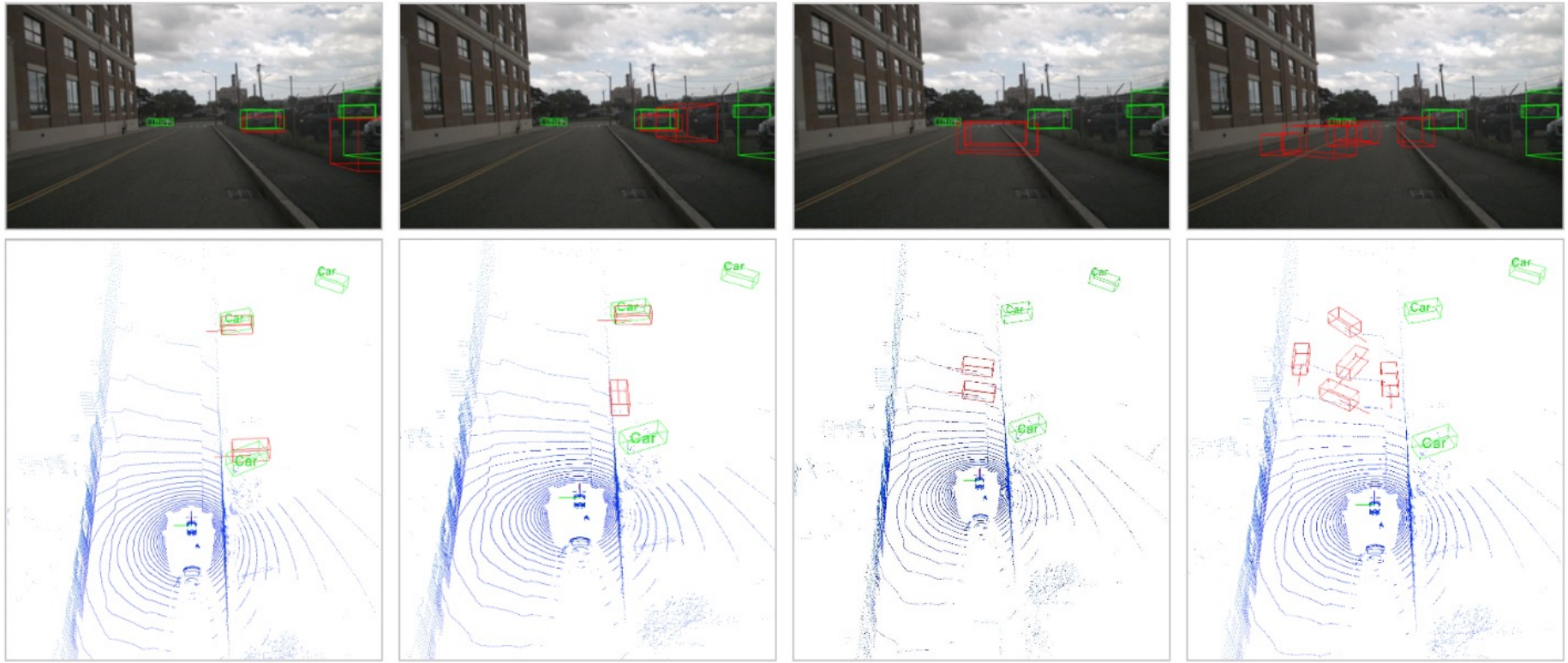
- LiDAR measurement are obtained along with the rotation of its beams
  - The measurements in a full sweep are captured at different timestamps, introducing motion distortion which jeopardizes the vehicle perception.



Li, Yiming, et al. "Fooling lidar perception via adversarial trajectory perturbation." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021.

Fooling LiDAR Perception via Adversarial Trajectory Perturbation

- We explore the adversarial vulnerability of this setting: attacking 3D point cloud perception of the distorted point cloud by adversarially perturbing the vehicle trajectory.

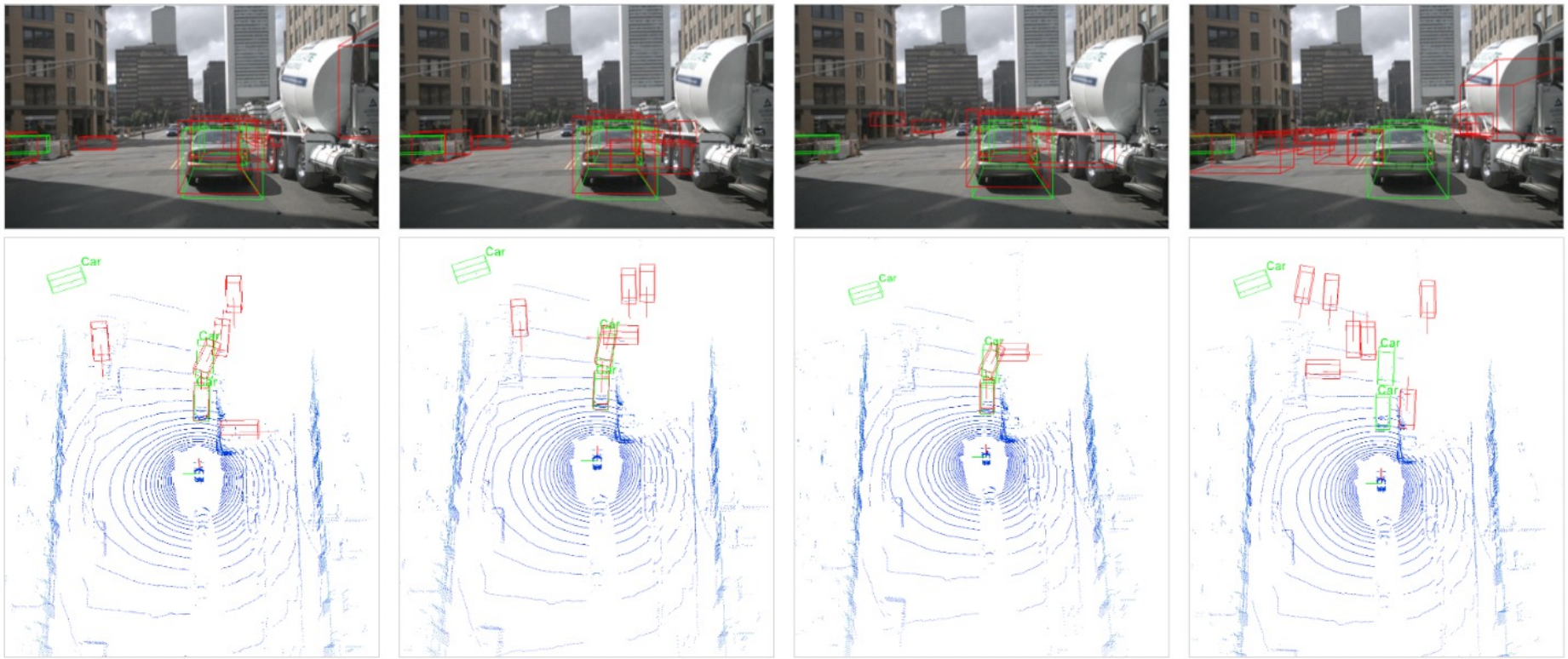


a) PointRCNN b) FLAT (Translation) c) FLAT (Rotation) d) FLAT (Full)

Li, Yiming, et al. "Fooling lidar perception via adversarial trajectory perturbation." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021.

Fooling LiDAR Perception via Adversarial Trajectory Perturbation

- We explore the adversarial vulnerability of this setting: attacking 3D point cloud perception of the distorted point cloud by adversarially perturbing the vehicle trajectory.



a) PointRCNN b) FLAT (Translation) c) FLAT (Rotation) d) FLAT (Full)

Li, Yiming, et al. "Fooling lidar perception via adversarial trajectory perturbation." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021.

Fooling LiDAR Perception via Adversarial Trajectory Perturbation

- We explore the adversarial vulnerability of this setting: attacking 3D point cloud perception of the distorted point cloud by adversarially perturbing the vehicle trajectory.

a) Original Detections.

b) Detections after the attack.

Green / red boxes denote ground truth / predictions respectively.

Li, Yiming, et al. "Fooling lidar perception via adversarial trajectory perturbation." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021.

Guest Lectures 1/3 (Today)

Speaker: Dr. Yunzhu Li

Title: Learning Structured World Models From and For Physical Interactions

Abstract: Humans have a strong intuitive understanding of the physical world. We observe and interact with the environment through multiple sensory modalities and build a mental model that predicts how the world would change if we applied a specific action (i.e., intuitive physics). My research draws on insights from humans and develops model-based reinforcement learning (RL) agents that learn from their interactions and build predictive models of the environment that generalize widely across a range of objects made with different materials. The core idea behind my research is to introduce novel representations and integrate structural priors into the learning systems to model the dynamics at different levels of abstraction. I will discuss how such structures can make model-based planning algorithms more effective and help robots to accomplish complicated manipulation tasks (e.g., manipulating an object pile, pouring a cup of water, and shaping deformable foam into a target configuration). Beyond visual perception, I will also discuss how we built multi-modal sensing platforms with dense tactile sensors in various forms (e.g., gloves, socks, vests, and robot sleeves) and how they can lead to more structured and physically grounded models of the world.

Guest Lectures 2/3 (Dec. 14)

Speaker: Dr. Sharon Yixuan Li

Title: How to Handle Data Shifts? Challenges, Research Progress and Path Forward

Abstract: The real world is open and full of unknowns, presenting significant challenges for machine learning systems that must reliably handle diverse, and sometimes anomalous inputs. Out-of-distribution (OOD) uncertainty arises when a machine learning model sees a test-time input that differs from its training data, and thus should not be predicted by the model. As machine learning is used for more safety-critical domains, the ability to handle out-of-distribution data is central in building open-world learning systems. In this talk, I will talk about challenges, research progress, and future opportunities in detecting OOD samples for safe and reliable predictions in an open world.

Guest Lectures 3/3 (Dec. 14)

Speaker: Dr. Krishna Murthy

Title: Differentiable programs for physical understanding – Modeling and Inference

Abstract: Modern machine learning has created exciting new opportunities for the design of intelligent scene understanding systems. In particular, gradient-based learning methods have tremendously improved 3D scene understanding in terms of perception, reasoning, and action. However these advancements have undermined many "classical" techniques developed over the last few decades. I postulate that a flexible blend of "classical" and learned methods is the most promising path to developing flexible, interpretable, and actionable models of the world: a necessity for intelligent embodied agents.

While modern learning-based scene understanding systems have produced remarkable results on learning from large volumes of data and/or in simulated scenarios, they fail in unpredictable and unintuitive ways when deployed in real-world applications. Classical systems, on the other hand, offer guarantees and bounds on performance and generalization, but often require heavy handcrafting and oversight. My research aims to deeply integrate classical and learning-based techniques to bring the best of both worlds, by building "differentiable models of the 3D world". In this talk, I will share some recent efforts (by me and collaborators) on building world models and inference techniques geared towards spatial and physical understanding.