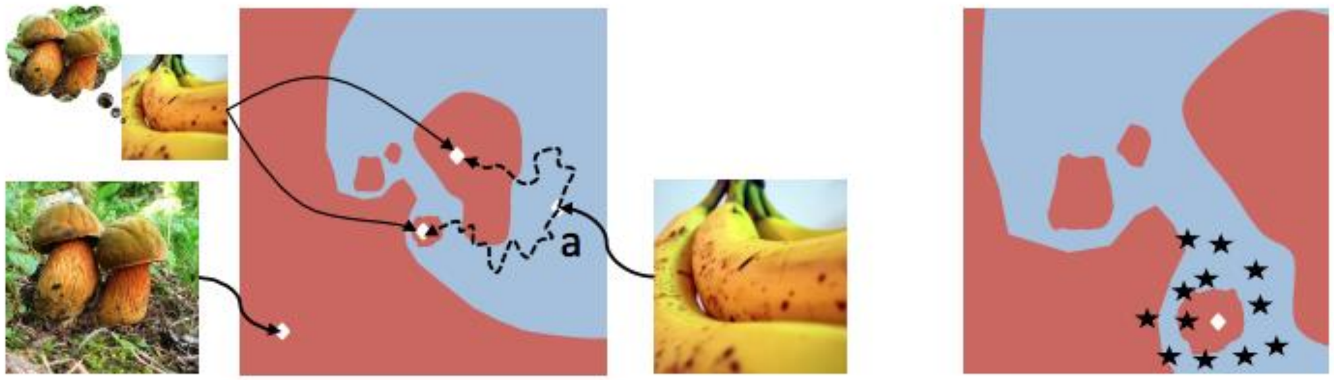
Detection Against Adversarial Attacks

Recall: Blackbox Attack

- No-query blackbox attack (transferability)
- Query based attack (score based, decision boundary based)
- Analysis for adversarial transferability

Exploring the space of adversarial images

- Adversarial examples in both linear and deep classifiers
- Probe the pixel space of adversarial images using noise of varying intensity and distribution
- Adversarial examples are isolated? Or do they form large, compact regions?



Banana (blue); mushroom (red)

Exploring the space of adversarial images

- Adversarial instance generation

$$\begin{aligned} \underset{D}{\text{minimize}} \quad & \|D\| \\ \text{subject to} \quad & L \leq X + D \leq U \\ & p = f(X + D) \\ & \max(p_1 - p_c, ..., p_n - p_c) > 0 \end{aligned}$$

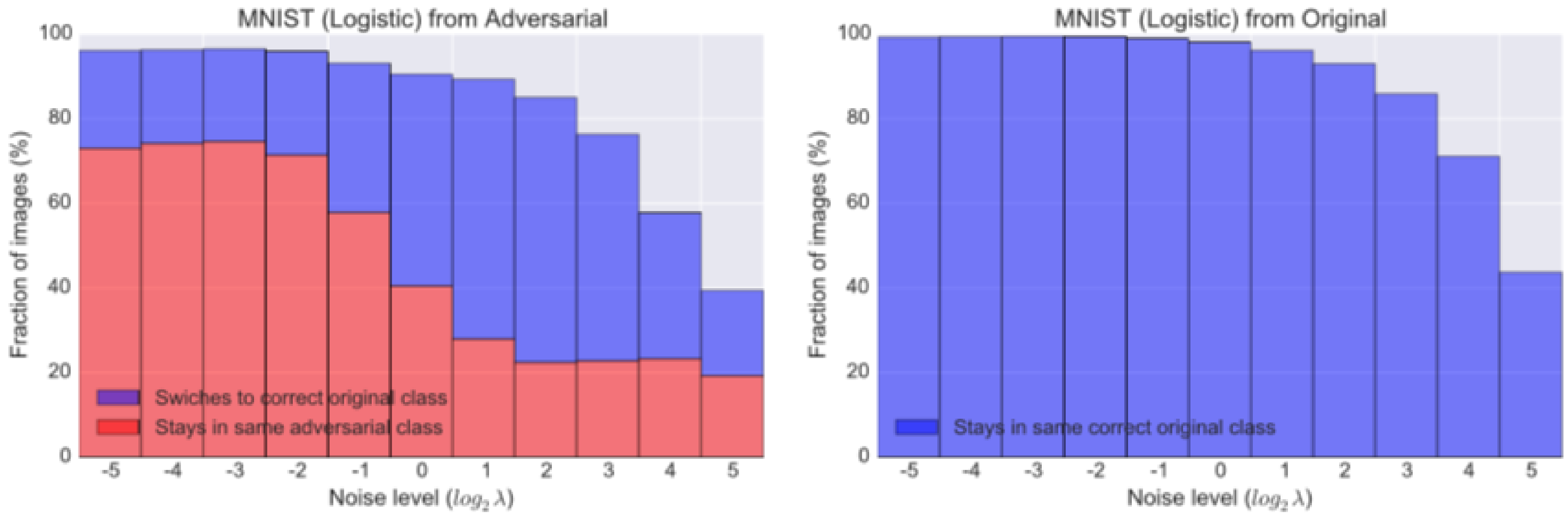
$$\begin{aligned} \underset{D}{\text{minimize}} \quad & \|D\| + C \cdot \mathrm{H}(p, p^A) \\ \text{subject to} \quad & L \leq X + D \leq U \\ & p = f(X + D) \end{aligned}$$

L-BFGS-B to solve the opt, and bisection search for C

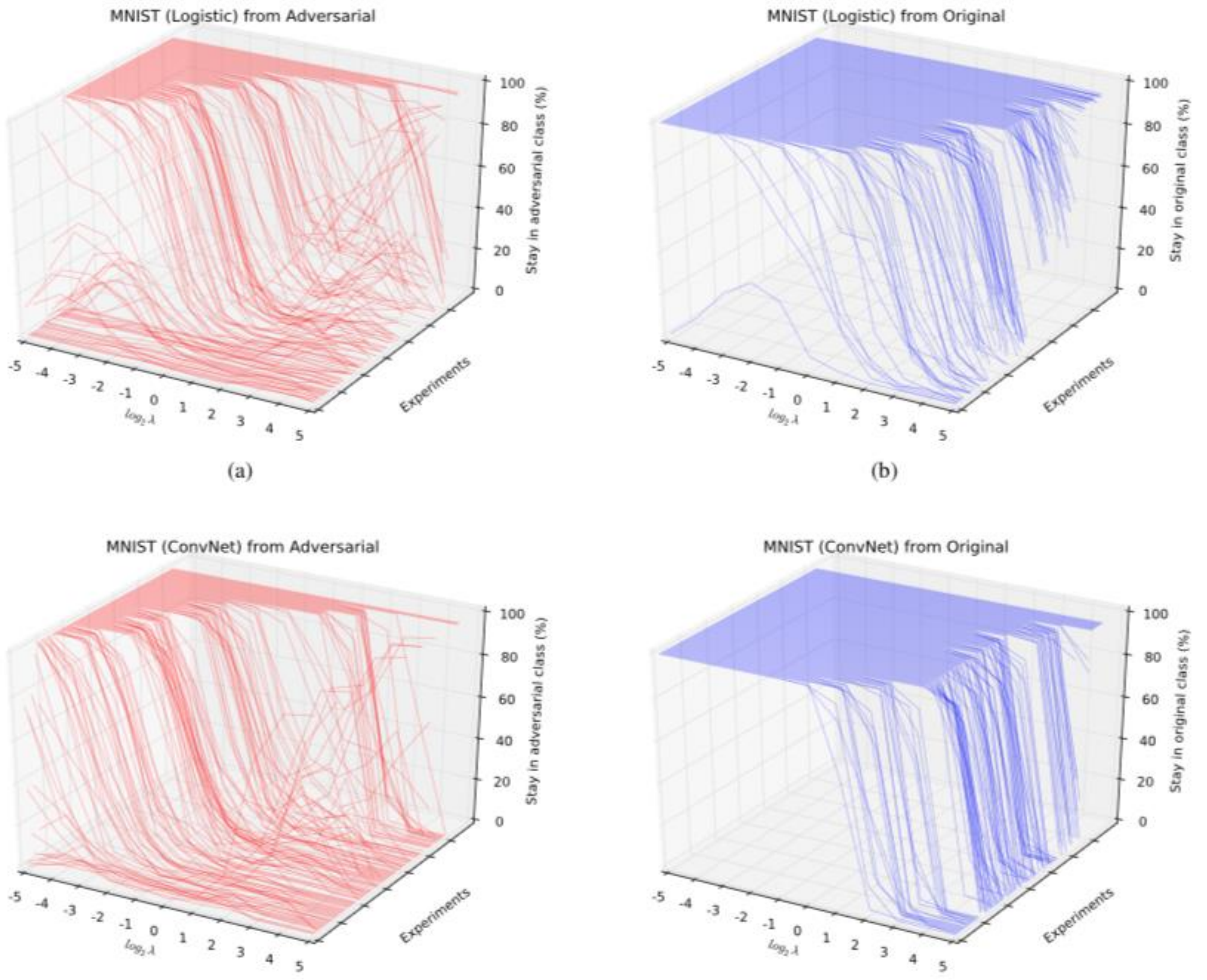
- Adversarial space exploration
  - Probe the space around the images with small random perturbation
    - Round, compact regions: classifier will be consistent
    - Sparse, discontinuous regions: classifier will be erratic

Exploring the space of adversarial images

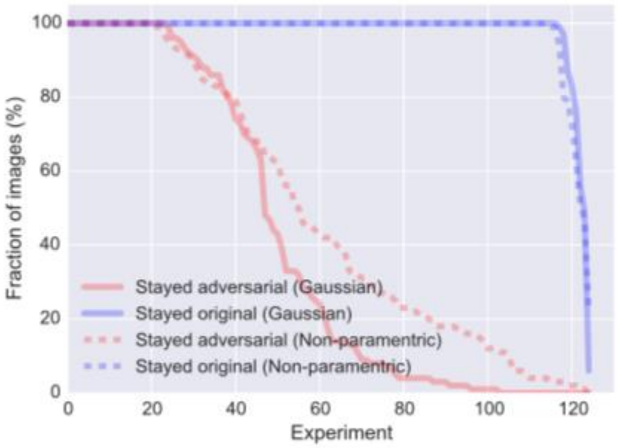
- Add noise to instance x and calculate the fraction that keep or switch labels
  - Gaussian noise $\epsilon \sim \mathcal{N}(\mu, \lambda\sigma^2)$
  - Sample from empirical distribution from a non-parametric observation $\epsilon \sim M$



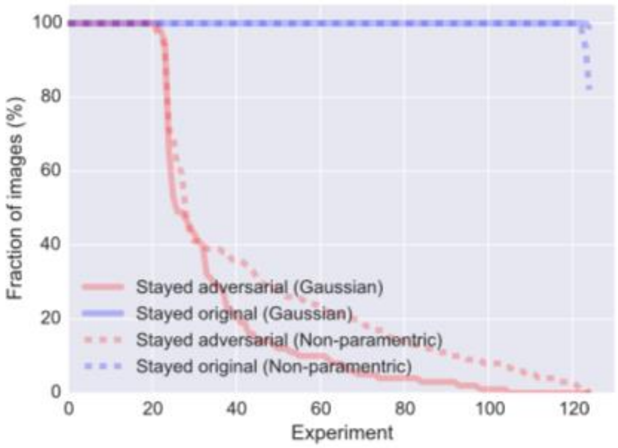
Exploring the space of adversarial images



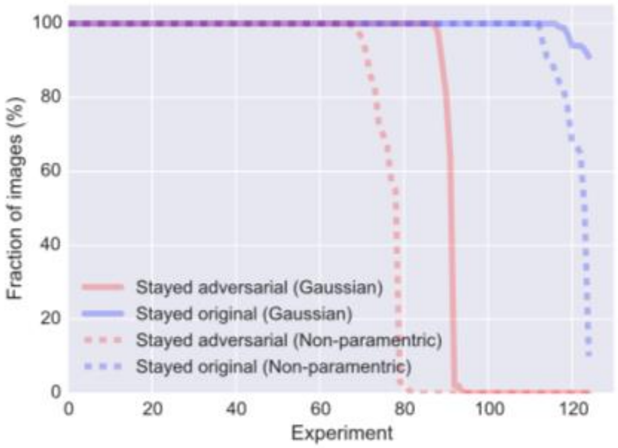
Adding non-parametric empirical noise



(a) MNIST / logistic regression



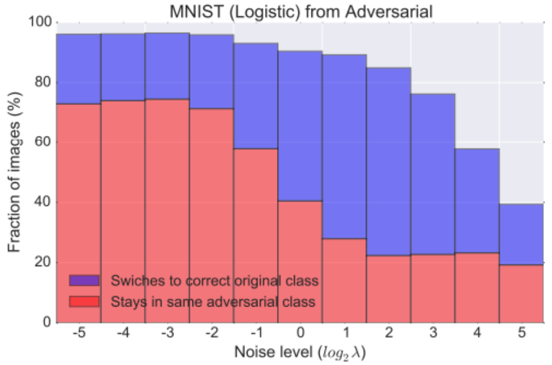
(b) MNIST / convolutional network



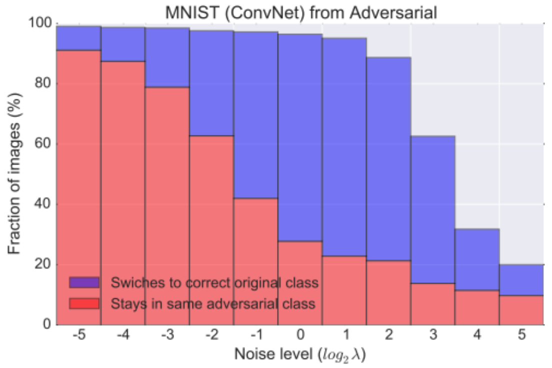
(c) ImageNet / OverFeat

Exploring the space of adversarial images

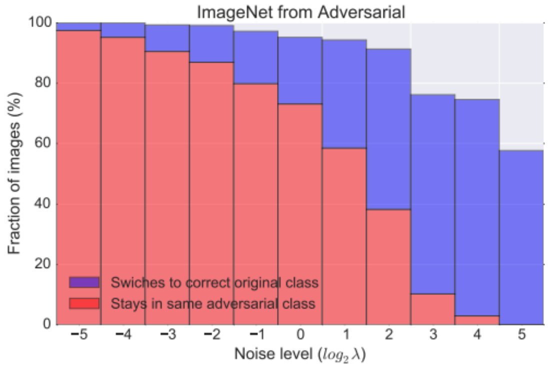
- Classifiers for MNIST are more resilient against adversarial images than ImageNet
- MNIST/logistic behaves differently than the deep MNIST/ConvNet
- i.i.d. Gaussian noise has spatial correlations, and no important higher-order momenta; therefore we also sample noise from nonparametric empirical distribution
  - For imageNet, the curves for non-parametric noise fall before that of Gaussian noise
  - The behavior tailed noise affects the images more even without the spatial correlation



(a)



(c)



Takeaways

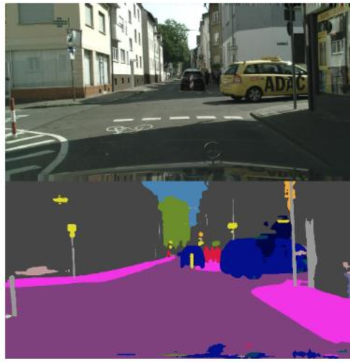
- Adversarial images are not necessarily isolated, spurious points: many of them inhabit relatively dense regions of the pixel space
- This may helps to explain the transferability
- An important next step: understand the spatial natural of the adversarial distortion
- Susceptibility of adversarial attacks is attributed to the linearity in the network but here it shows the phenomenon may be more complex
  - A relatively more linear classifier seems no more susceptible to adversarial images than a strong, deep classifier

Beyond the Min-max Game

- What if we have more knowledge about our learning tasks?
  - Properties of learning tasks and data
  - General understanding about ML models

Characterize Adversarial Examples Based on Spatial Consistency Information for Semantic Segmentation

- Attacks against semantic segmentation
  - State-of-the-art attacks against segmentation: Houdini [NIPS2017], DAG [ICCV 2017]
  - We design diverse adversarial targets: hello kitty, pure color, a real scene, ECCV, color shift, strips of even color of classes
  - Cityscapes and BDD datasets



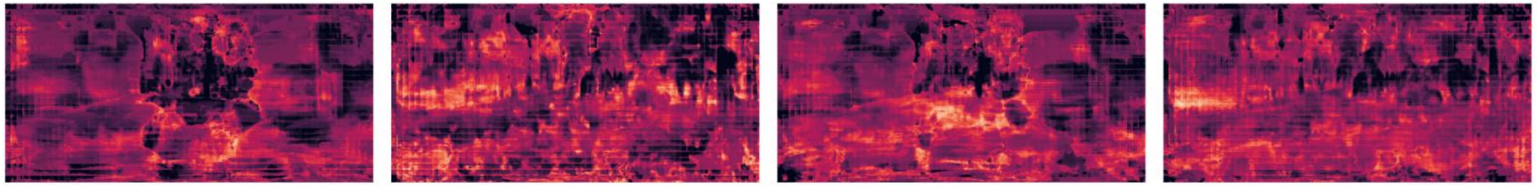
Benign

Adversarial Examples

Spatial Context Information

- Spatial consistency is a distinct property of image segmentation
- Perturbation at one pixel will potentially affect the prediction of surrounding pixels $\mathcal{H}(m) = -\sum_{j} \mathcal{V}_m[j] \log \mathcal{V}_m[j]$

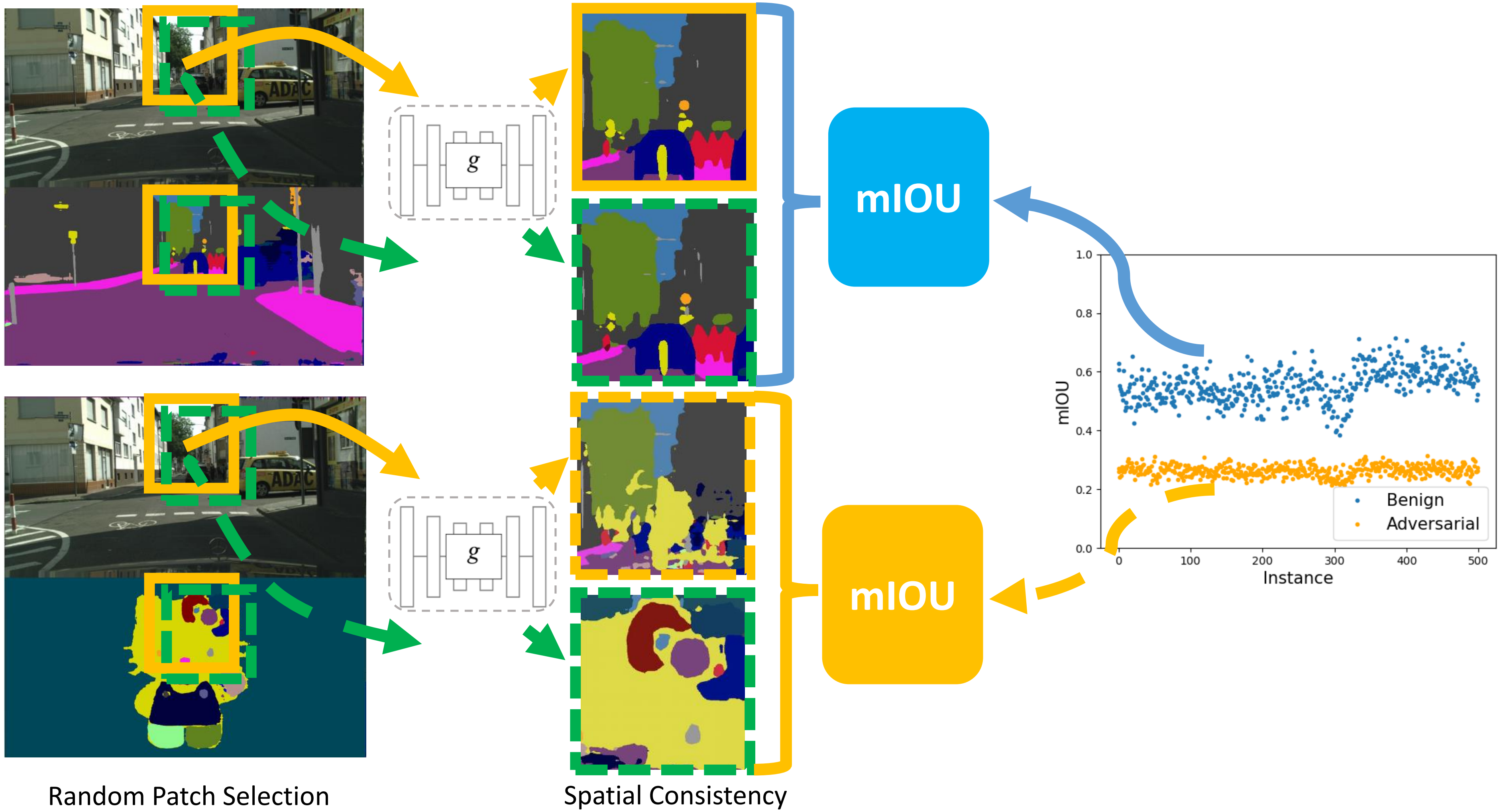
(a) Benign example (b) Heatmap of benign image



(c) DAG | Kitty (d) DAG | Pure (e) Houdini | Kitty (f) Houdini | Pure

Perturbation on single patch may loss its adversarial effect

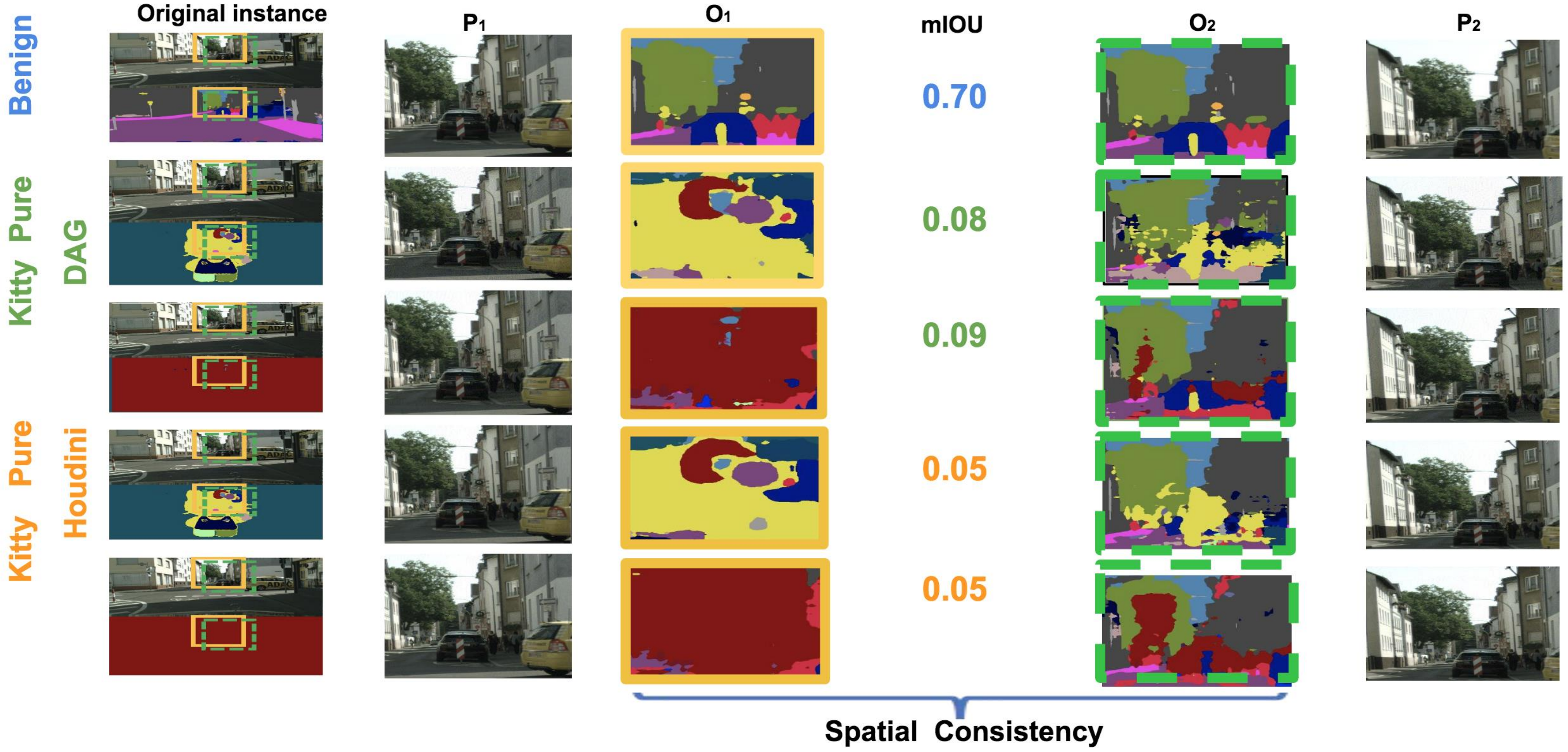
- Spatial consistency: the consistency of segmentation results for randomly selected patches from an image
- Such spatial consistency information from benign and adversarial instances are distinguishable
- We apply mIOU to compare the segmentation results between patches
  - For each class, Intersection over Union (IOU) is calculated as TP/(TP+FP+FN). Here we calculate the relative mIOU for each pair of patches



Pipeline of spatial consistency based detection for adversarial examples on semantic segmentation

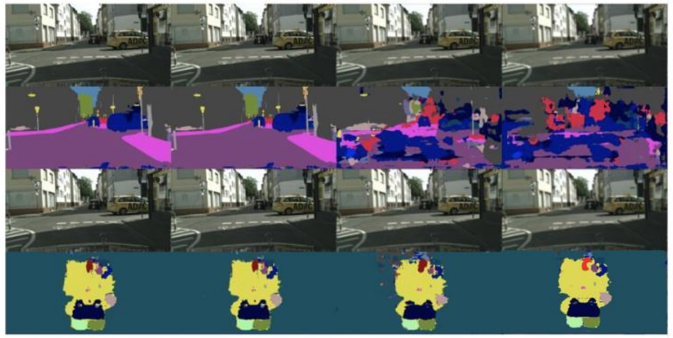
We apply mIOU to evaluate the consistency information for patches from benign and adversarial instances quantitatively

- Detection

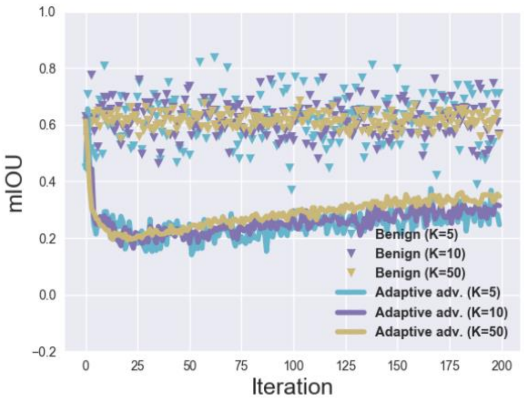


Adaptive Attack Against Spatial Consistency Based Detection

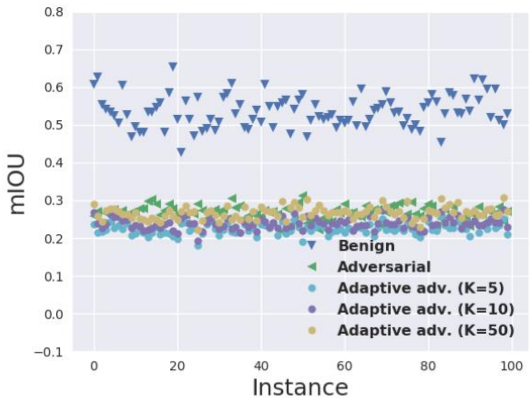
- Adaptive attack:
  - Assume the attacker has perfect knowledge of #selected patches: K
  - We generate perturbation that the selected k patches can all be mis-segmented to the corresponding regions within adversarial target



(a) Adaptive attack against image scaling



(b) Convergence analysis of adaptive attack



(c) Adaptive attack against spatial consistency

Detecting adversarial instances based on spatial consistency information

- Both the spatial consistency based detection and the scaling based baseline achieve promising detection rate on different attacks
- The scaling based baseline fails to detect strong adaptive attacks while the spatial based method can

Method		Model	mIOU	Detection DAG Pure	Detection DAG Kitty	Detection Houdini Pure	Detection Houdini Kitty	Detection Adap DAG Pure	Detection Adap DAG Kitty	Detection Adap Houdini Pure	Detection Adap Houdini Kitty
Scale (std)	0.5	DRN (16.4M)	66.7	100%	95%	100%	99%	100%	67%	100%	78%
	3.0			100%	100%	100%	100%	100%	0%	97%	0%
	5.0			100%	100%	100%	100%	100%	0%	71%	0%
Spatial (K)	1	DRN (16.4M)	66.7	91%	91%	94%	92%	98%	94%	92%	94%
	5			100%	100%	100%	100%	100%	100%	100%	100%
	10			100%	100%	100%	100%	100%	100%	100%	100%
	50			100%	100%	100%	100%	100%	100%	100%	100%

Takeaways

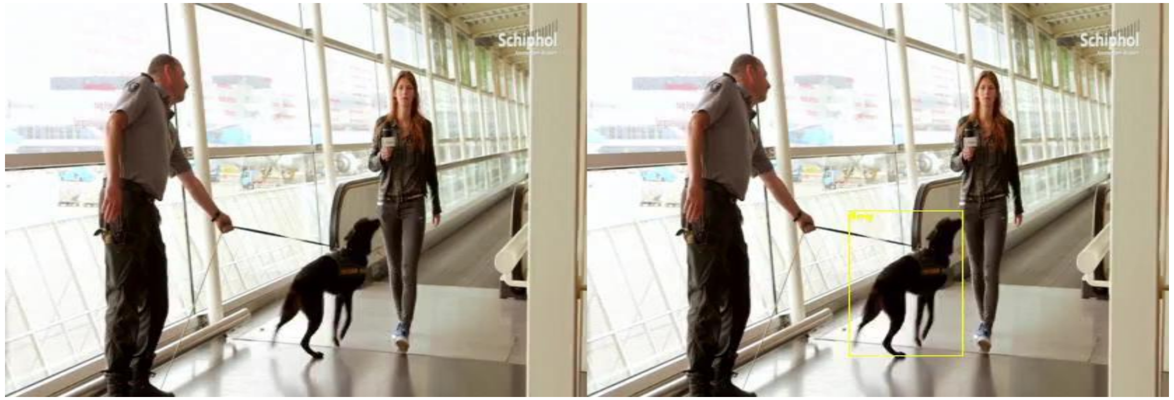
- Spatial consistency information can be potentially applied to help distinguish benign and adversarial instances against segmentation models.
- Strong adaptive attacker can hardly succeed when large randomness is incorporated into the model

Adversarial Frames In Videos

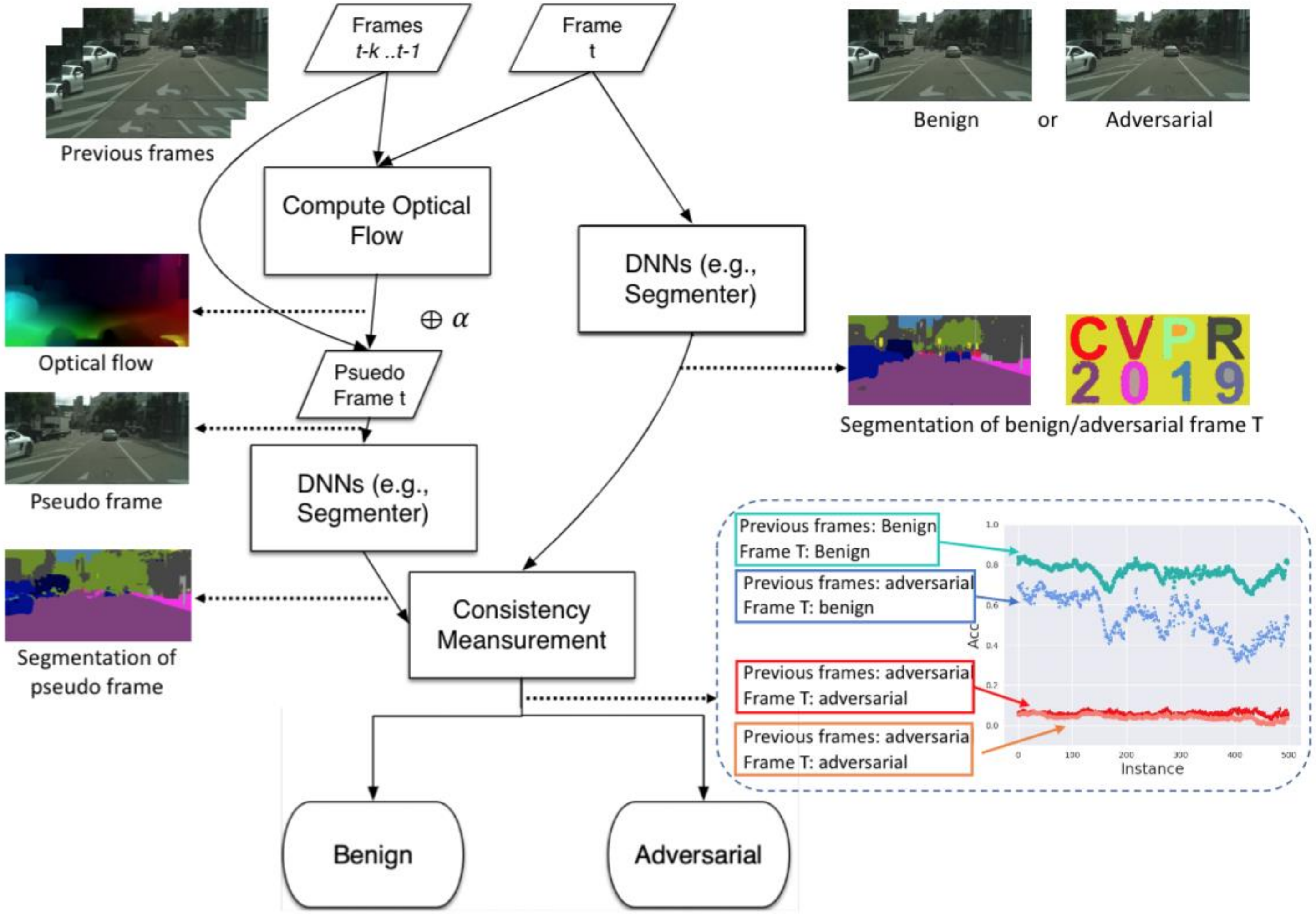
Attacks on segmentation

Attacks on pose estimation

Attacks on object detection

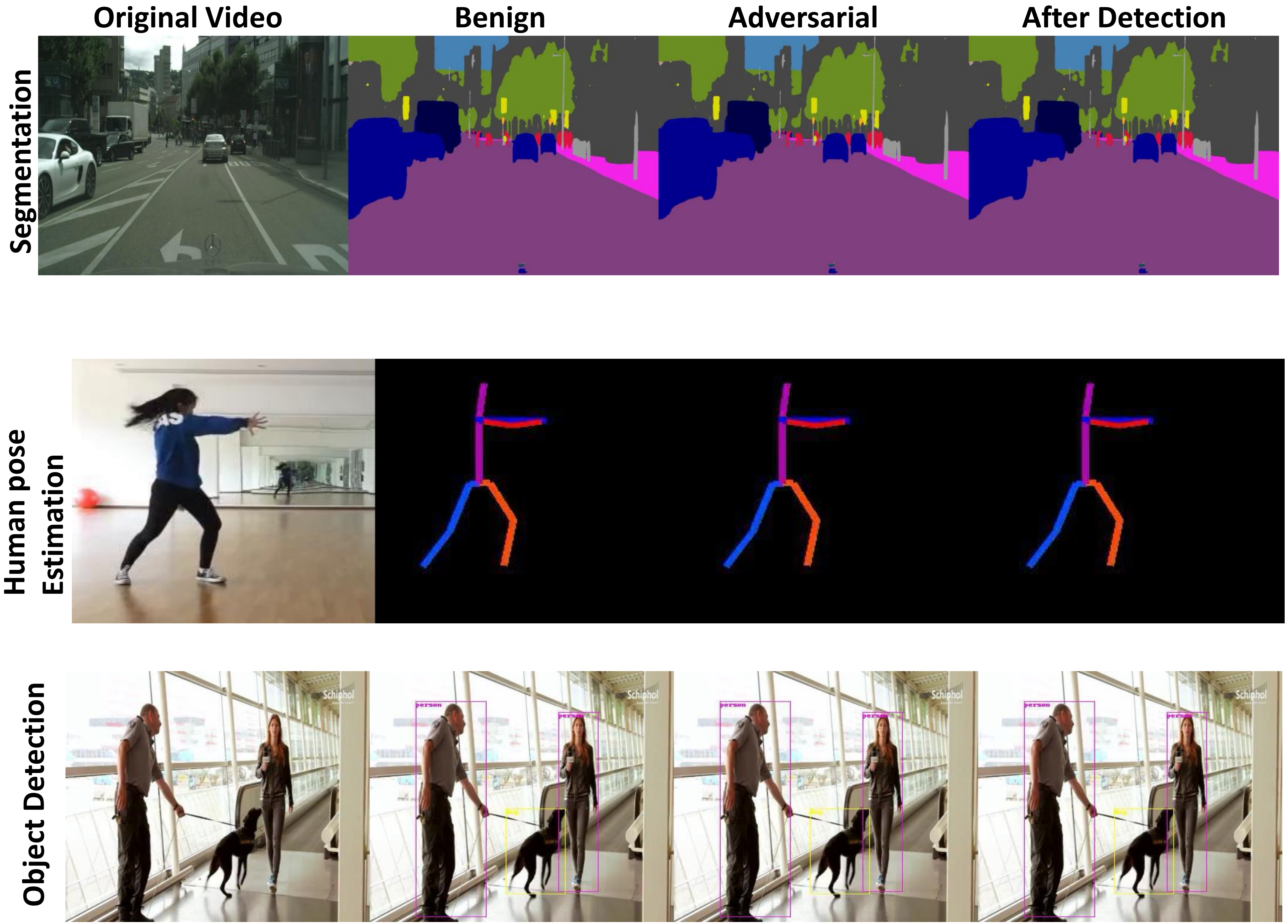


Defensing Adversarial behaviors in Videos – Temporal Dependency



Task	Attack Method	Target	Previous Frames	Detection 1	Detection 3	Detection 5	Detection Adap 1	Detection Adap 3	Detection Adap 5
Semantic Segmentation	Houdini	CVPR	Benign	100%	100%	100%	100%	100%	100%
			Adversarial	100%	100%	100%	100%	100%	100%
		Remapping	Benign	100%	100%	100%	100%	100%	100%
			Adversarial	100%	100%	100%	100%	100%	100%
		Stripe	Benign	100%	100%	100%	100%	100%	100%
			Adversarial	100%	100%	100%	99%	100%	100%
	DAG	CVPR	Benign	100%	100%	100%	100%	100%	100%
			Adversarial	100%	100%	100%	100%	100%	100%
		Remapping	Benign	100%	100%	100%	100%	100%	100%
			Adversarial	100%	100%	100%	100%	100%	100%
		Stripe	Benign	100%	100%	100%	100%	100%	100%
			Adversarial	100%	100%	100%	100%	100%	100%
Human Pose Estimation	Houdini	shuffle	Benign	100%	100%	100%	100%	100%	100%
			Adversarial	100%	100%	100%	99%	100%	100%
		Transpose	Benign	100%	100%	100%	98%	100%	100%
			Adversarial	98%	99%	100%	98 %	99%	100%
Object Detection	DAG	all	Benign	100%	100%	100%	100%	100%	100%
			Adversarial	100%	100%	100%	98%	100%	100%
		person	Benign	99%	100%	100 %	100%	100%	100%
			Adversarial	97%	98%	100%	96 %	97%	100%

- The results show that choosing more random patches can improve detection rate while k=5 is enough to achieve AUC 100%
- The spatial consistency based detection is robust against strong adaptive attackers due to the randomness in patch selection

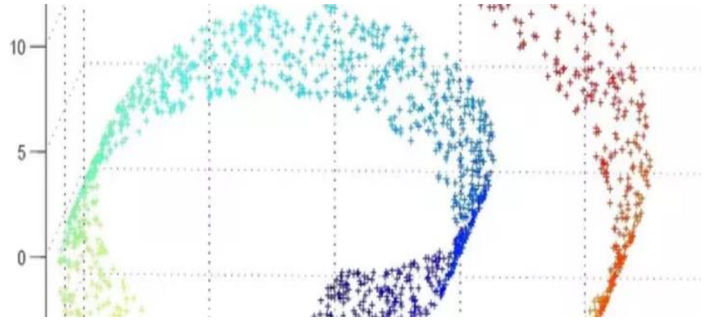

Original Video
Benign
Adversarial
After Detection
Segmentation
Human pose
Estimation
Object Detection

Beyond the Min-max Game

- What if we have more knowledge about our learning tasks?
  - Properties of learning tasks and data
  - General understanding about ML models

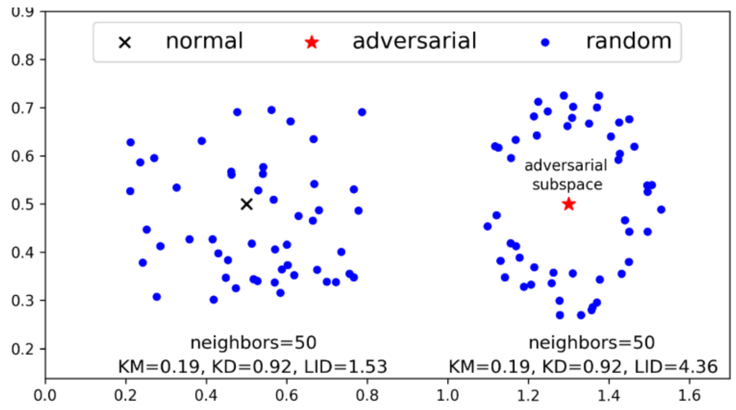
Important Concept: data manifold

- Data Manifold theory:
  - Manifold: the subspace that has local Euclidean space properties
  - The data we observed were actually mapped from a low-dimensional space
  - We use PCA/autoencoders etc. to “unwrap” the manifold
  - We assume the data points from testset and trainset are all from a same manifold
  - Not the case if we consider adversaries



[ICLR 2018]

Previous Measures



- K-means distance
  - Distance to k nearest neighbors
- Kernel density
  - non-parametric
  - estimate the pdf (probability density function) of a random variable

$$\hat{f}(x) = \frac{1}{nh} \sum_{i=1}^{n} k(\frac{x - x_i}{h})$$

- Can fail to distinguish the sub-manifold that a test case lies in

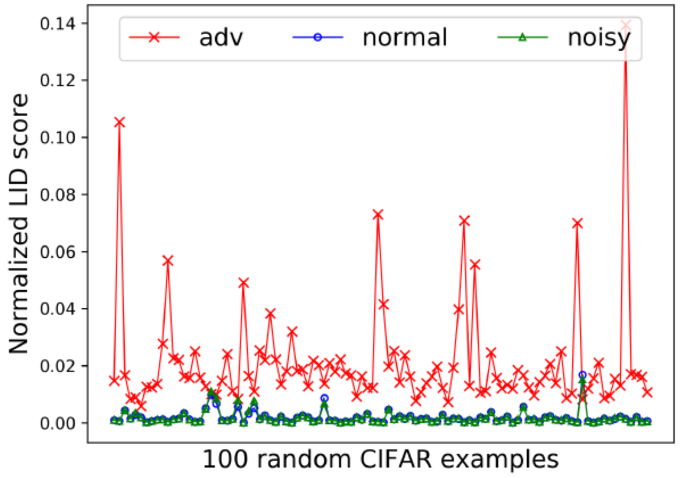
Estimation of Local Intrinsic Dimensionality (LID)

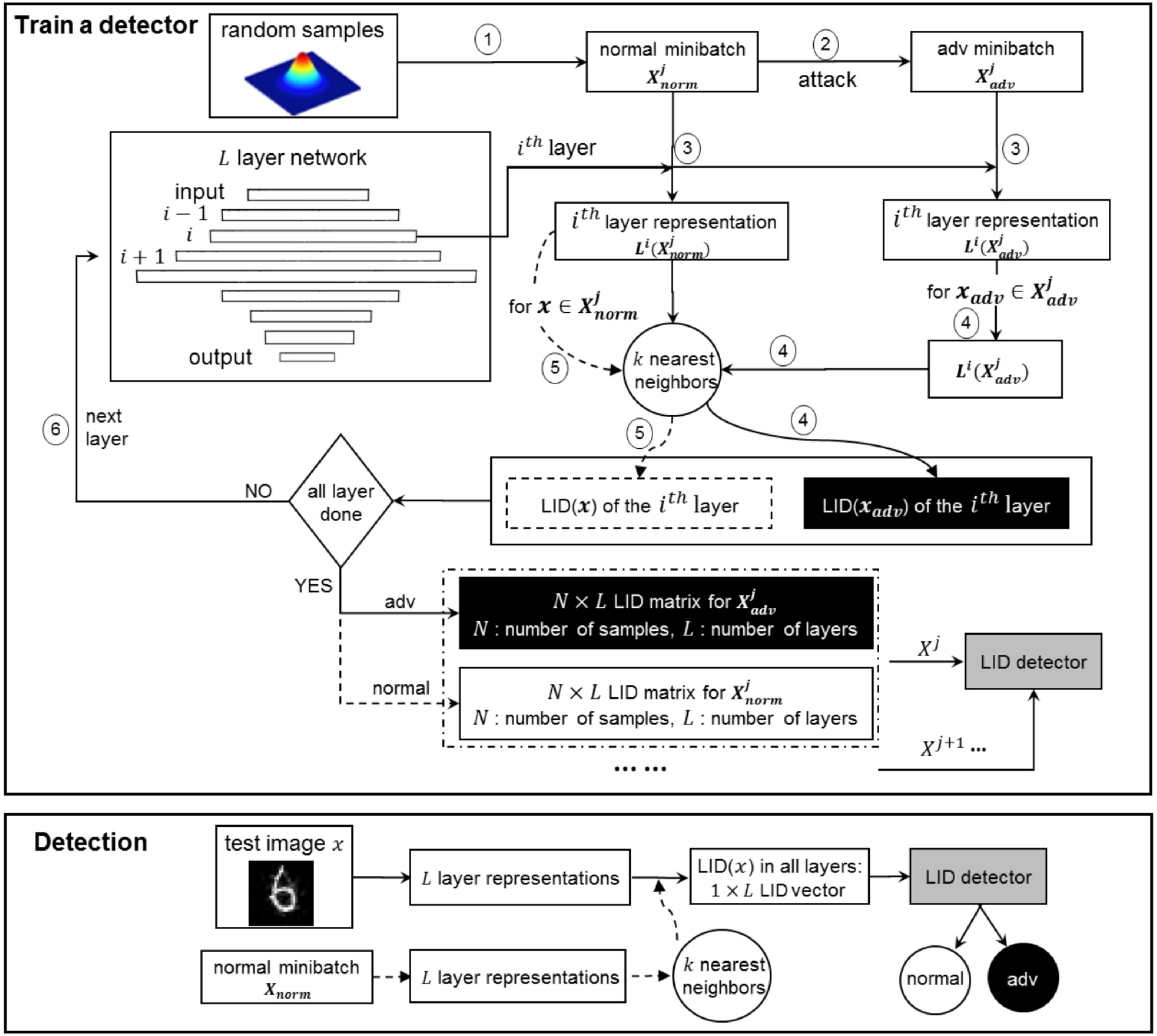
- The sub-manifolds are not parametric
  - given by data points instead
- We use estimation
  - Sample a small set of size larger than k
  - compute their distance to x, take closest k
  - $r_k(x)$ is the maximum of the neighbor distances

$$\widehat{\mathrm{LID}}(x) = -\left(\frac{1}{k}\sum_{i=1}^{k}\log\frac{r_i(x)}{r_k(x)}\right)^{-1}$$

Use LID to characterize the sub-manifold

- LID of benign x
  - The dimension of S (the sub-manifold x lies in)
  - Should be small since S is under some intrinsic constraints
- LID of adversarial x':
  - Full degrees of freedom afforded by the representational dimension of the data domain
  - Attacks generally allow modification of all pixels




Train a detector
random samples
1
normal minibatch X^j_{norm}
2
attack
adv minibatch X^j_{adv}
L layer network
input
$i-1$
i
$i+1$
output
i^{th} layer
3
3
i^{th} layer representation $L^i(X^j_{norm})$
i^{th} layer representation $L^i(X^j_{adv})$
for $\boldsymbol{x} \in X^j_{norm}$
for $\boldsymbol{x}_{adv} \in X^j_{adv}$
4
4
k nearest neighbors
$L^i(X^j_{adv})$
5
5
4
6
next layer
NO
all layer done
LID($\boldsymbol{x}$) of the i^{th} layer
LID($\boldsymbol{x}_{adv}$) of the i^{th} layer
YES
adv
normal
$N \times L$ LID matrix for X^j_{adv}
N : number of samples, L : number of layers
$N \times L$ LID matrix for X^j_{norm}
N : number of samples, L : number of layers
X^j
LID detector
X^{j+1} ...
... ...
Detection
test image x
L layer representations
LID(x) in all layers: $1 \times L$ LID vector
LID detector
normal minibatch X_{norm}
L layer representations
k nearest neighbors
normal
adv

Characterizing Adversarial Examples

AUC of different detection methods against various attacks

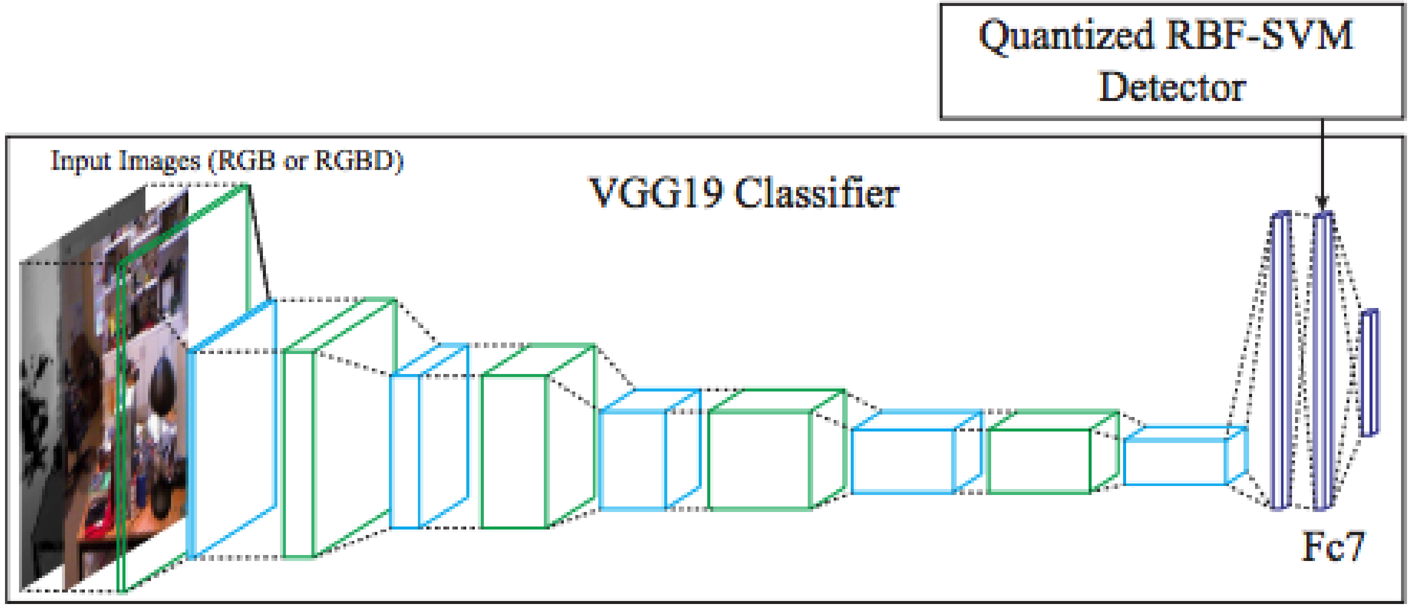
Dataset	Feature	FGM	BIM-a	BIM-b	JSMA	Opt
MNIST	KD	78.12%	99.14%	98.61%	68.77%	95.15%
	BU	32.37%	91.55%	25.46%	88.74%	71.29%
	KD+BU	82.43%	99.20%	98.81%	90.12%	95.35%
	LID	**96.89%**	**99.60%**	**99.83%**	**92.24%**	**99.24%**
CIFAR-10	KD	64.92%	68.38%	98.70%	85.77%	91.35%
	BU	70.53%	81.60%	97.32%	87.36%	91.39%
	KD+BU	70.40%	81.33%	98.90%	88.91%	93.77%
	LID	**82.38%**	**82.51%**	**99.78%**	**95.87%**	**98.93%**
SVHN	KD	70.39%	77.18%	99.57%	86.46%	87.41%
	BU	86.78%	84.07%	86.93%	91.33%	87.13%
	KD+BU	86.86%	83.63%	99.52%	93.19%	90.66%
	LID	**97.61%**	**87.55%**	**99.72%**	**95.07%**	**97.60%**

Attack Failure Rate of Strong Adaptive Attacks Against LID Detector

	MNIST	CIFAR-10	SVHN
Attack Failure Rate (one-layer)	100%	95.7%	97.2%
Attack Failure Rate (all-layer)	100%	100%	100%

SaftyNet: Detecting and Rejecting Adversarial Examples Robustly

- Use RBF-SVM to perform classification based on the discrete codes computed from late stage ReLUs



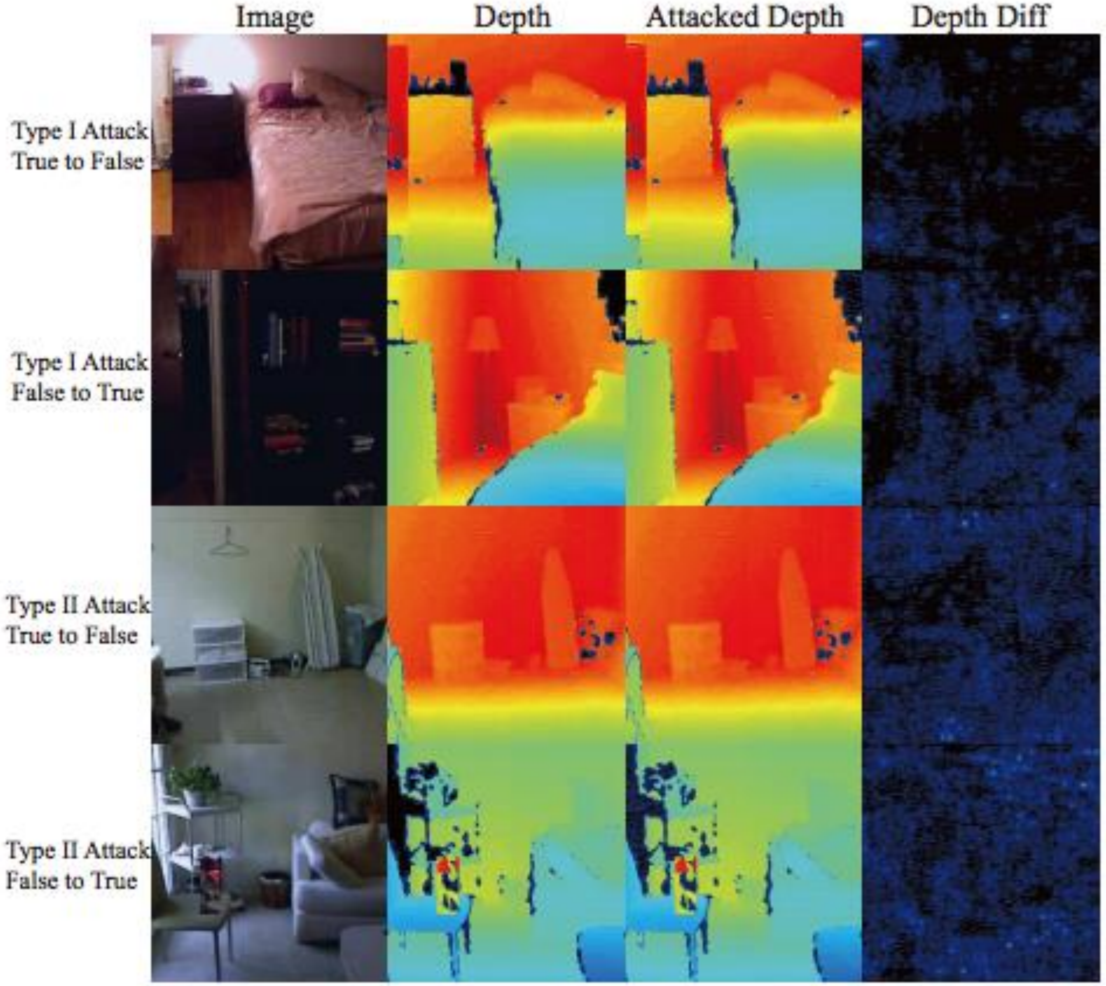
SaftyNet: Detecting and Rejecting Adversarial Examples Robustly

- Quantize each ReLU at some set of thresholds to generate a discrete code (binarized code in the case of one threshold)

$$f(\mathbf{c}) = \sum_{i}^{N} \alpha_i y_i \exp(-||\mathbf{c} - \mathbf{c}_i||^2 / 2\sigma^2) + b$$

- Hypothesis: Adversarial attacks work by producing different patterns of activation in late stage ReLUs to those produced by natural examples

Application: SceneProof



Takeaways

- Leverage a network to make adversarial attacks harder
- In the SceneProof application it is possible to check whether a pair of image and depth map is consistent or not
  - Robust to specific types of applications