Robustness of classifiers: from adversarial to universal perturbations

Pascal Frossard, EPFL

Google, Zurich

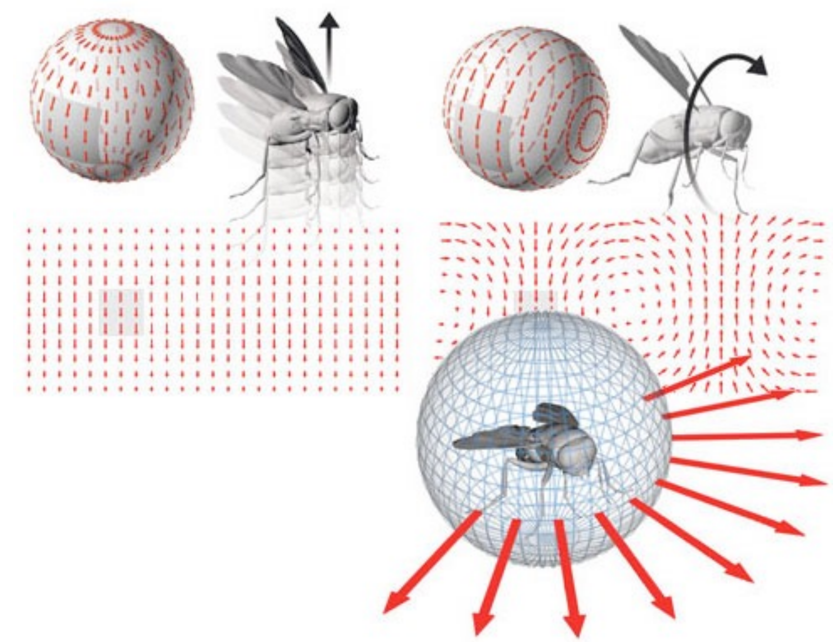
Jan 19th, 2017

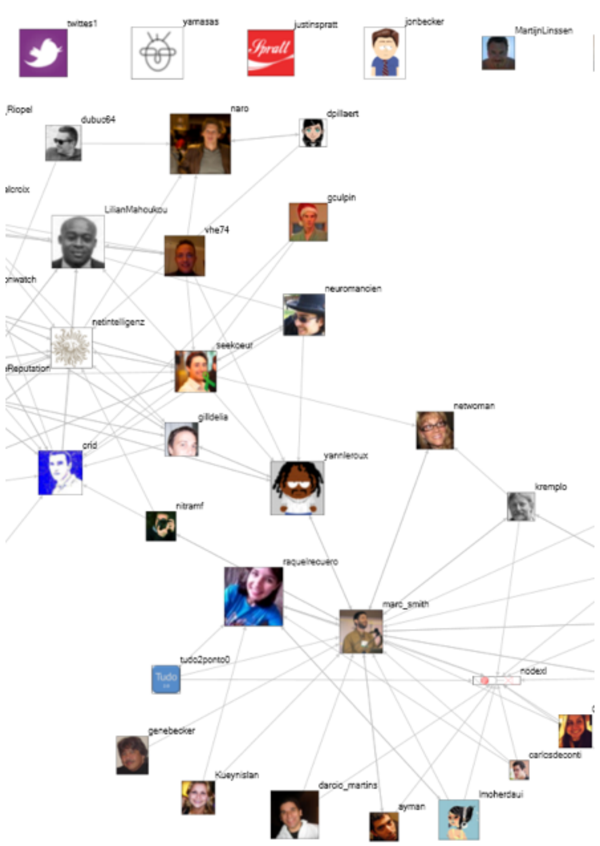
ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Main research topics

- Image/video processing
  - Representation learning
  - Image analysis and classification
  - Immersive communications

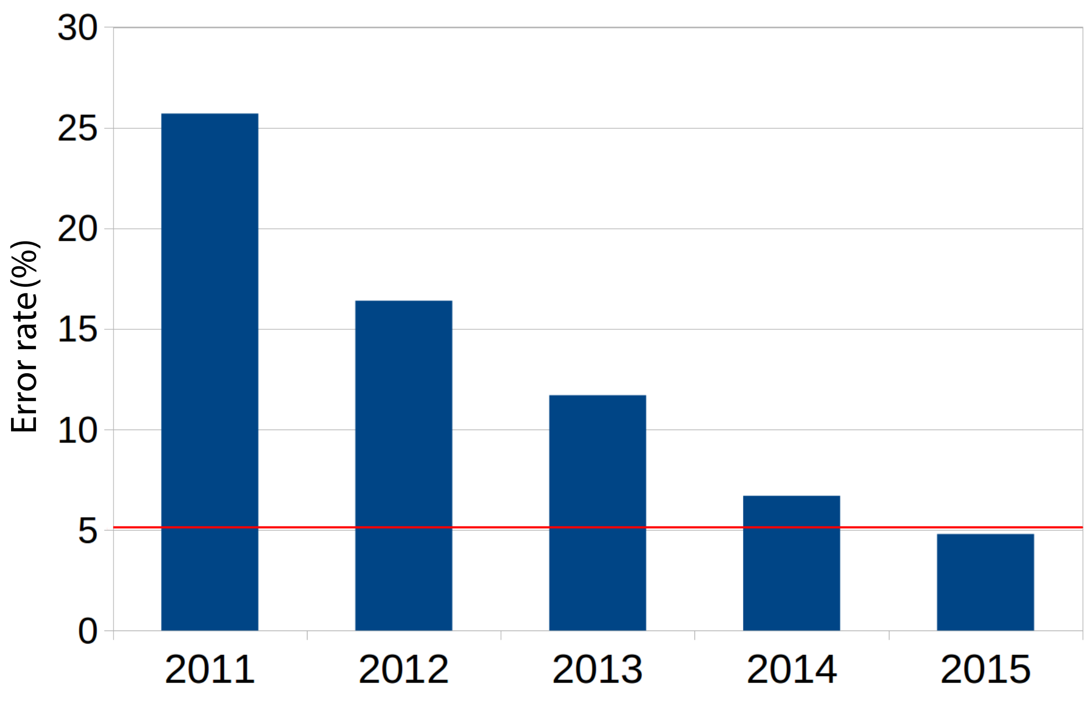
- Graph Signal Processing
  - Representation of structured data
  - Analysis of network data (computer, social, traffic, brain networks…)



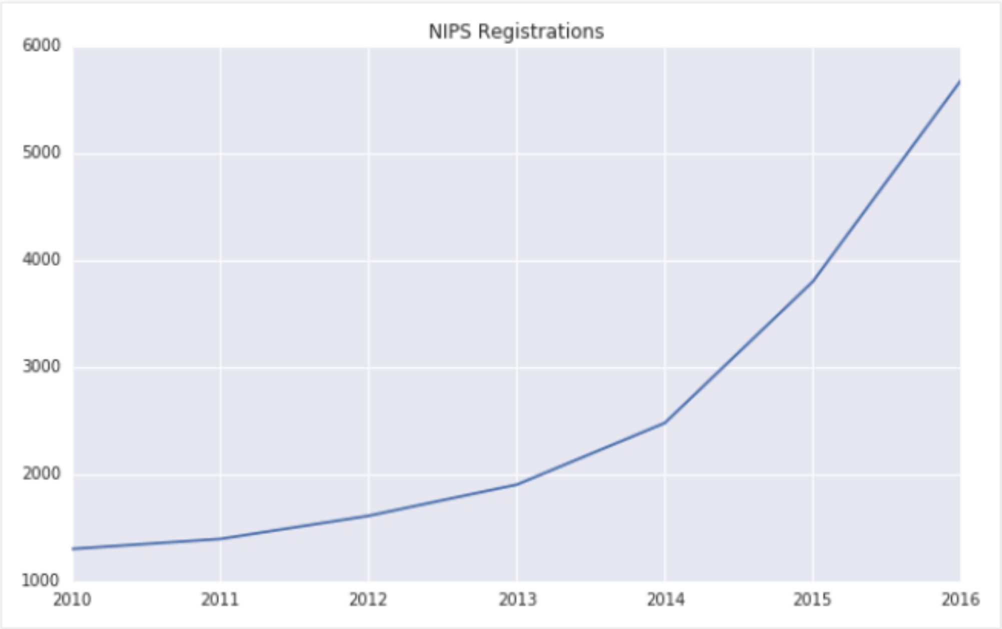


The rise of Deep Learning

- State-of-the-art classifiers achieve a surprisingly good accuracy on very challenging datasets.



ImageNet Large Scale Visual Recognition Challenge, IJCV 2015



http://blog.evjang.com/2017/01/nips2016.html

Sample architecture: CNNs

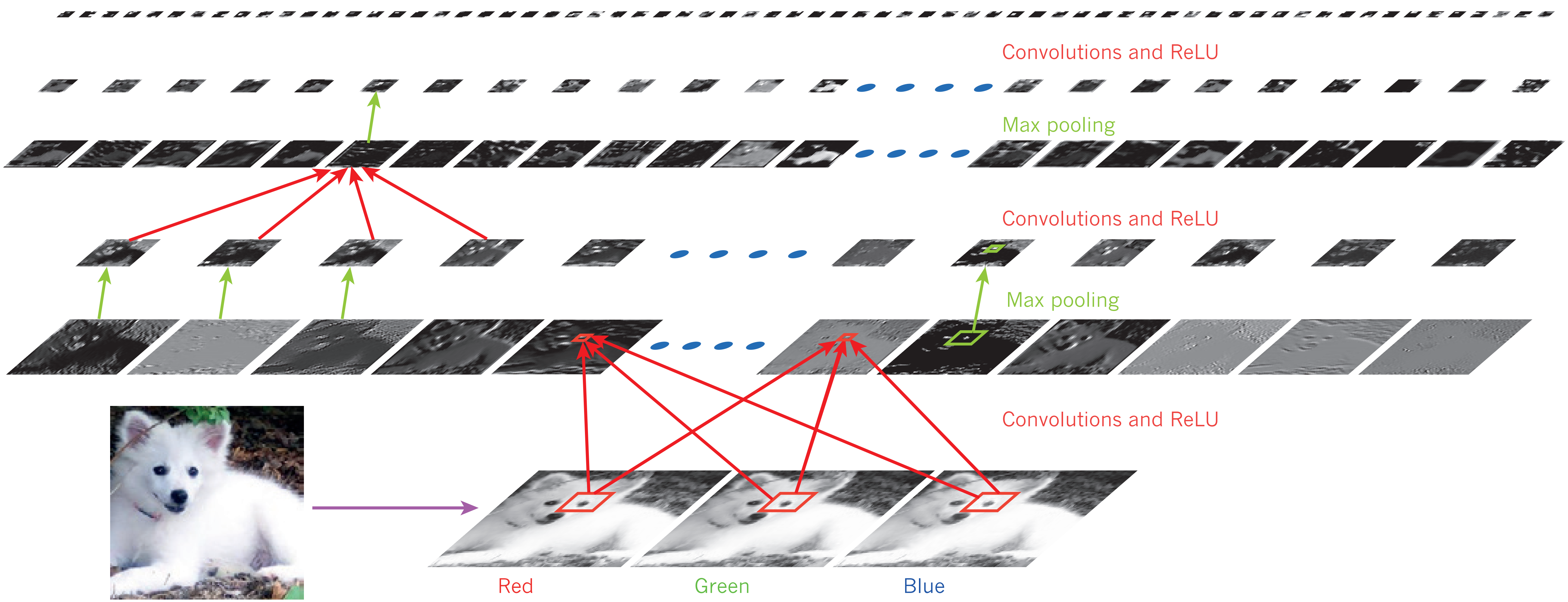


Figure 2 | Inside a convolutional network. The outputs (not the filters) of each layer (horizontally) of a typical convolutional network architecture applied to the image of a Samoyed dog (bottom left; and RGB (red, green, blue) inputs, bottom right). Each rectangular image is a feature map corresponding to the output for one of the learned features, detected at each of the image positions. Information flows bottom up, with lower-level features acting as oriented edge detectors, and a score is computed for each image class in output. ReLU, rectified linear unit.

Figure from: Deep learning, *Yann LeCun, Yoshua Bengio and Geoffrey Hinton, Nature, May 2015*

Are we done?

- Deep Learning is very popular and very successful
  - state-the-art-results in several tasks (speech, vision)

- But deep learning $\neq$ deep understanding
  - proper design is often an art…
  - training data / computing power is not always available

Need for in-depth study of classifiers' performance!

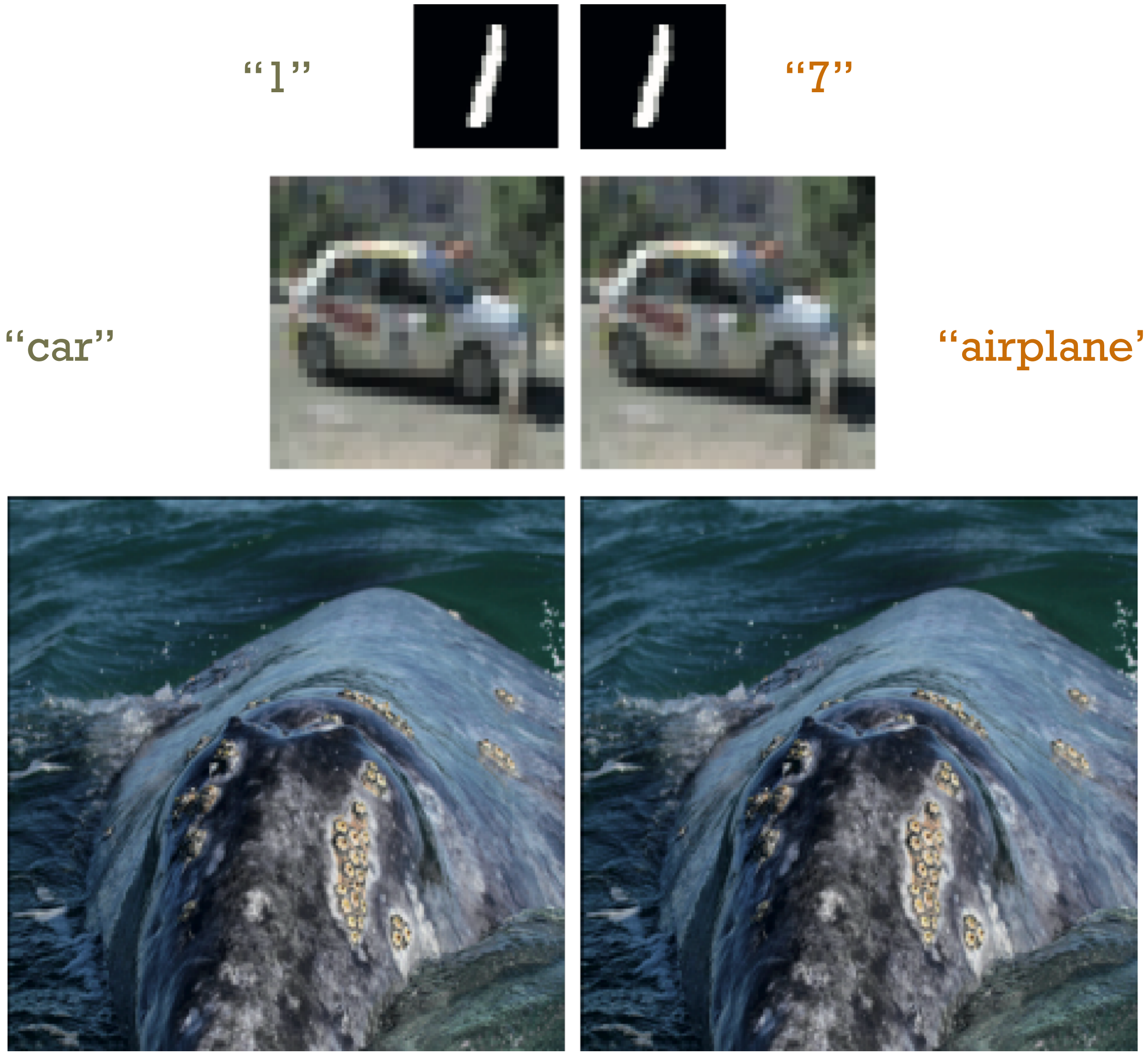
- better understanding of current classifiers
- design of better systems (?)

Motivating examples

"1" "7"

"car" "airplane"

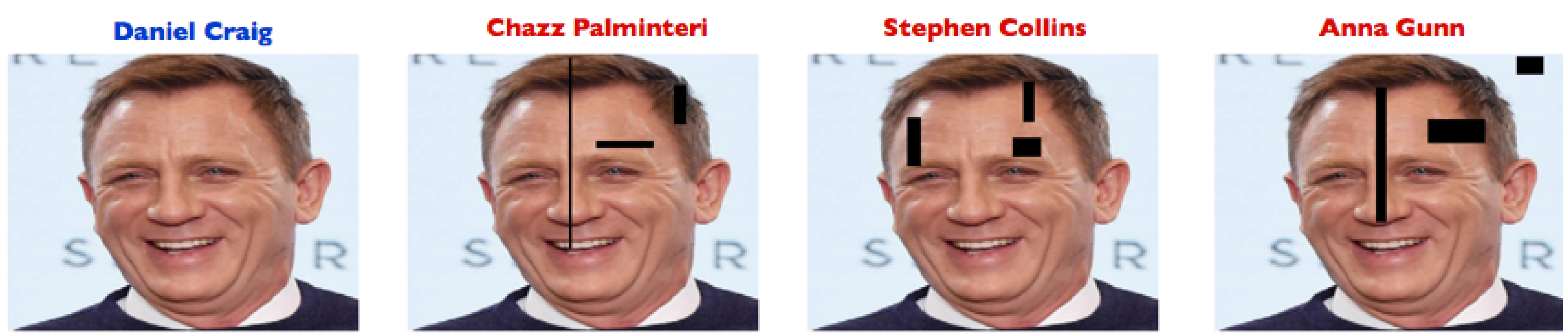
"whale" "turtle"



Any visible difference between the left and right columns?

Further examples

Locally affine transformations



Small occlusions

Agenda

- Intriguing properties of adversarial noise (recall)
- Robustness to random and semi-random noise
- Vulnerability to *universal* perturbations

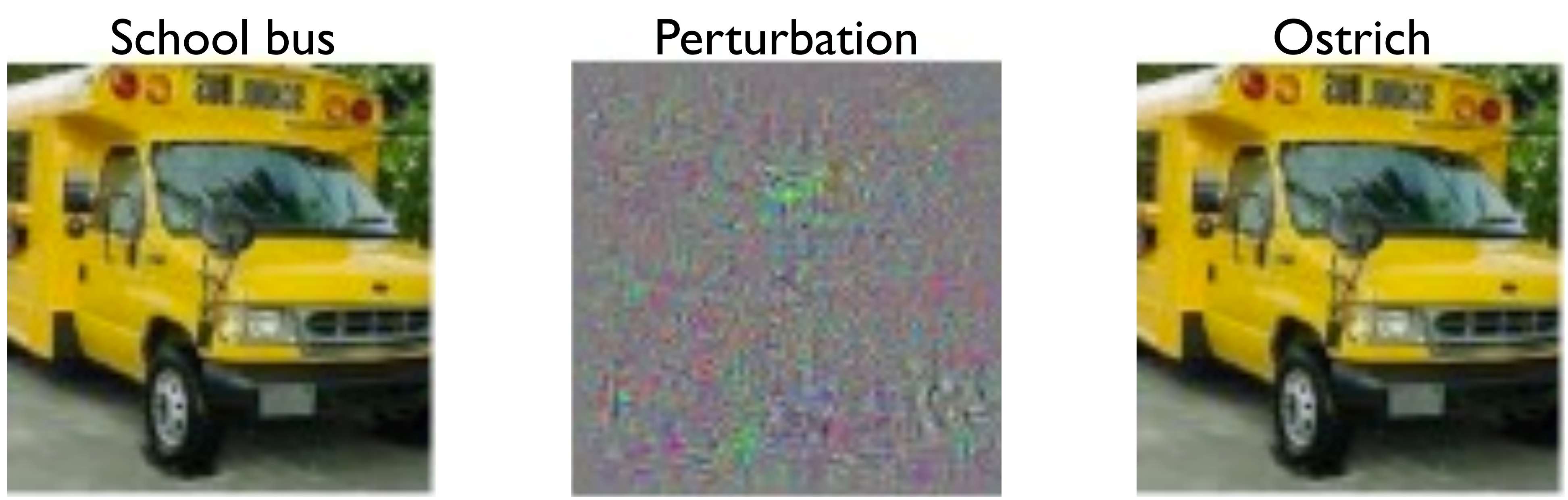
Alhussein Fawzi
EPFL/UCLA

Seyed-Mohsen Moosavi-Dezfooli
EPFL

Omar Fawzi
ENS Lyon

Noise 1: Adversarial

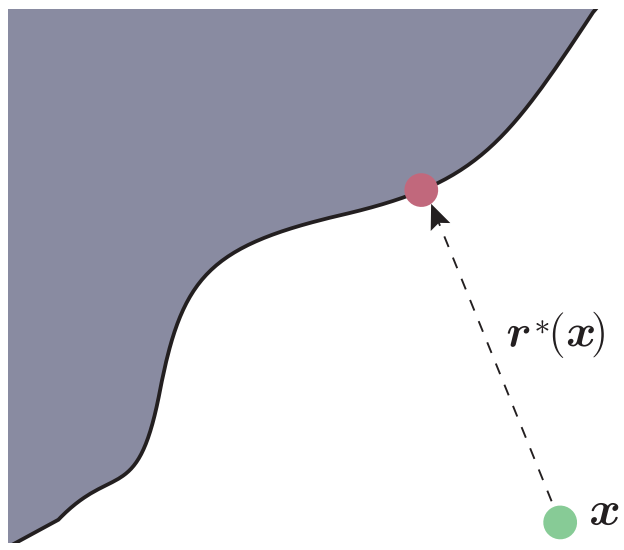
- *Adversarial noise*: smallest additive perturbation that changes the classifier's label
  - State-of-the-art deep nets have been shown to be surprisingly unstable to such data-specific perturbations



Szegedy et. al., Intriguing properties of neural networks, ICLR 2014.

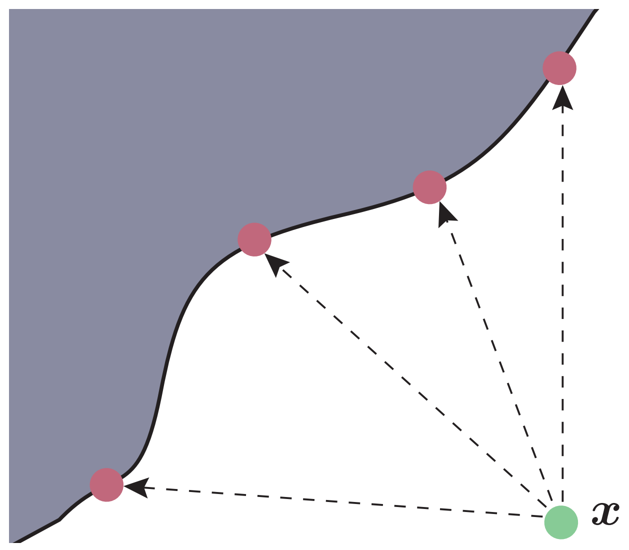
Adversarial robustness

Adversarial noise



$$\min_r \|r\|_2 \text{ subject to } \hat{k}(x+r) \neq \hat{k}(x)$$

Random noise

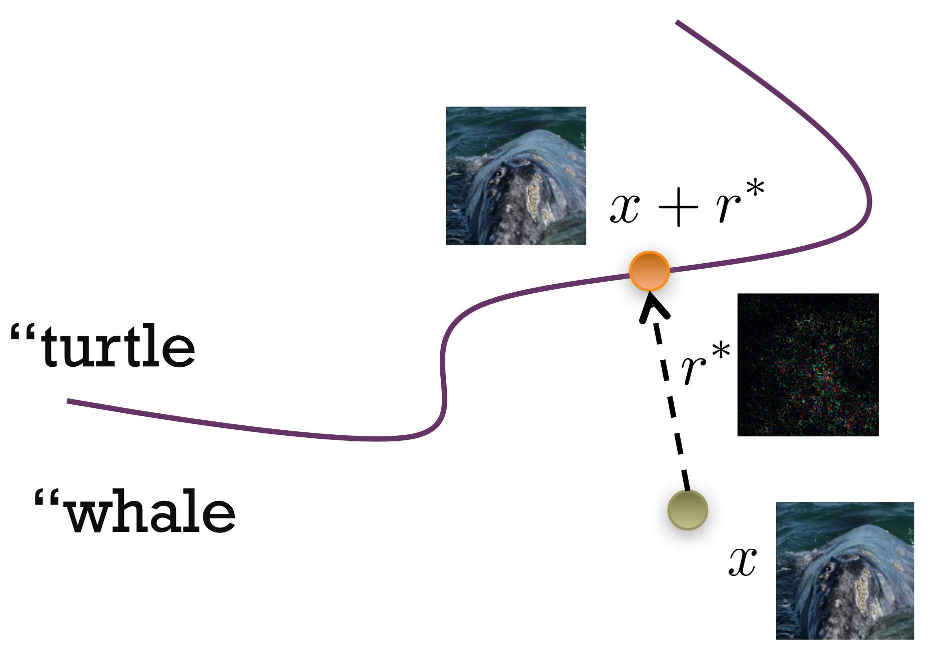


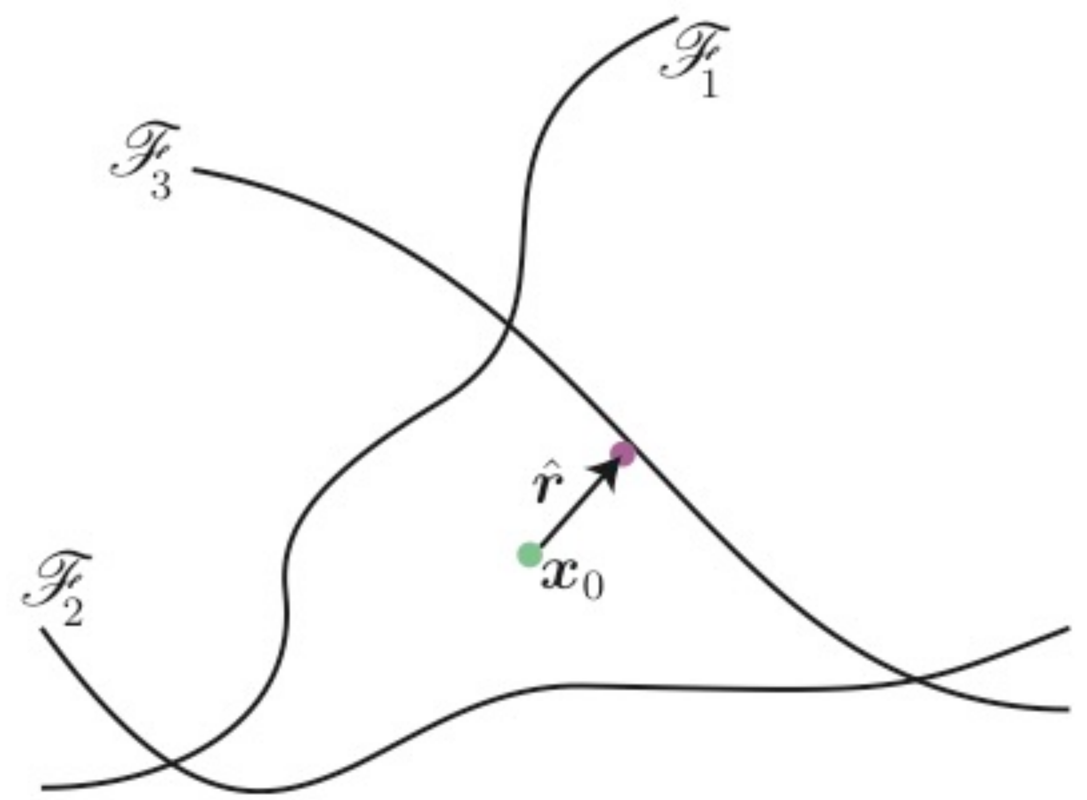
$$\min_t |t| \text{ subject to } \hat{k}(x+t[v]) \neq \hat{k}(x)$$

$$[v] \text{ uniformly sampled from } \mathbb{S}^{d-1}$$

DeepFool algorithm

- Effective computation of adversarial robustness
  - Simple idea: iterative linearization of the decision boundaries

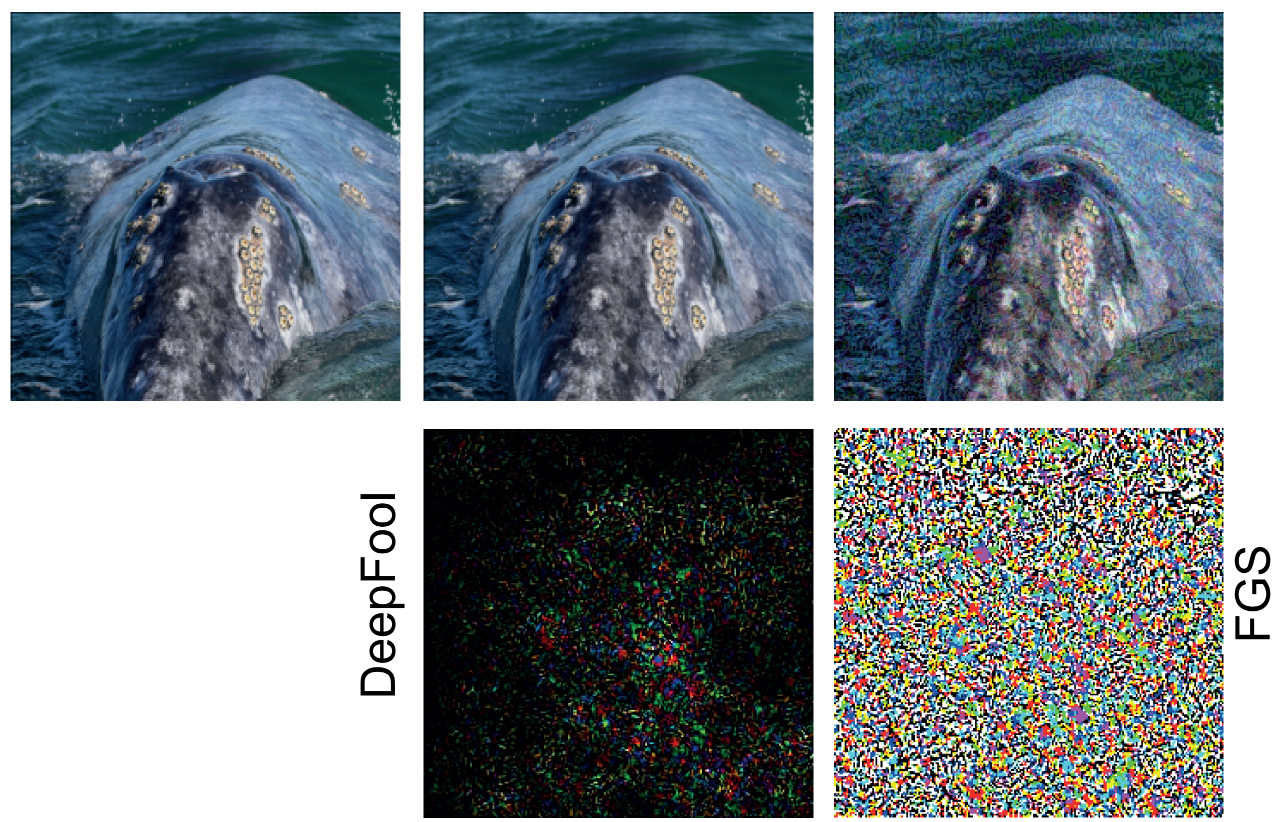




DeepFool: a simple and accurate method to fool deep neural networks
Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi and Pascal Frossard
IEEE CVPR, Las Vegas, Nevada, June 2016.

Classifiers are really not robust!

Classifier	**Test error**	$\hat{\rho}_{\text{adv}}$ [DeepFool]	time	$\hat{\rho}_{\text{adv}}$ [4]	time	$\hat{\rho}_{\text{adv}}$ [18]	time
LeNet (MNIST)	1%	2.0×10^{-1}	110 ms	1.0	20 ms	2.5×10^{-1}	> 4 s
FC500-150-10 (MNIST)	1.7%	1.1×10^{-1}	50 ms	3.9×10^{-1}	10 ms	1.2×10^{-1}	> 2 s
NIN (CIFAR-10)	11.5%	2.3×10^{-2}	1100 ms	1.2×10^{-1}	180 ms	2.4×10^{-2}	>50 s
LeNet (CIFAR-10)	22.6%	3.0×10^{-2}	220 ms	1.3×10^{-1}	50 ms	3.9×10^{-2}	>7 s
CaffeNet (ILSVRC2012)	42.6%	2.7×10^{-3}	510 ms*	3.5×10^{-2}	50 ms*	-	-
GoogLeNet (ILSVRC2012)	31.3%	1.9×10^{-3}	800 ms*	4.7×10^{-2}	80 ms*	-	-



[4] Goodfellow:ICLR 2015

[18] Szegedy:ICLR2014

Noise 2: semi-random

- We introduce the *semi-random noise* regime

$$\min_{r \in \mathcal{S}} \|r\|_2 \text{ subject to } \hat{k}(x+r) \neq \hat{k}(x)$$

- where $\mathcal{S}$ is a randomly chosen subspace of dimension $m \leq d$

- Semi-random noise interpolates between random and adversarial noise.

Robustness of classifiers to random and semi-random noise
Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi and Pascal Frossard
NIPS, December 2016.

Robustness to semi-random noise

Let $\mathcal{S}$ be a random subspace of dimension m. For affine classifiers, we have

$$\|r^*_{\mathcal{S}}\|_2 = \Theta\left(\sqrt{\frac{d}{m}}\|r^*\|_2\right)$$

with high probability.

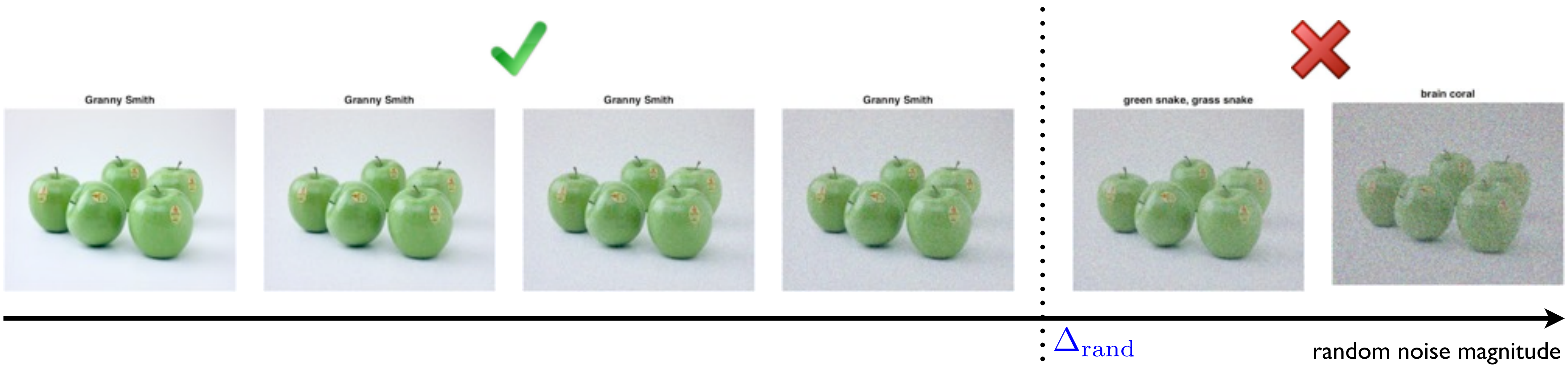
Theorem 1

Provided the curvature of the decision boundary of the classifier is sufficiently small, the above result holds for non-linear classifiers too.

Theorem 2

Theorem's implication

- Special case A: $m = 1$
  - Robustness to random noise $\approx$ Robustness to adversarial noise $\times \sqrt{d}$



- Special case B: $m = \epsilon d$
  - Robustness to semi- random noise $\approx$ Rob. to adversarial noise $\times \epsilon^{-1/2}$

CNNs are also vulnerable to semi-random noise!

Experimental validation

We measure robustness with DeepFool, and compute a normalised metric:

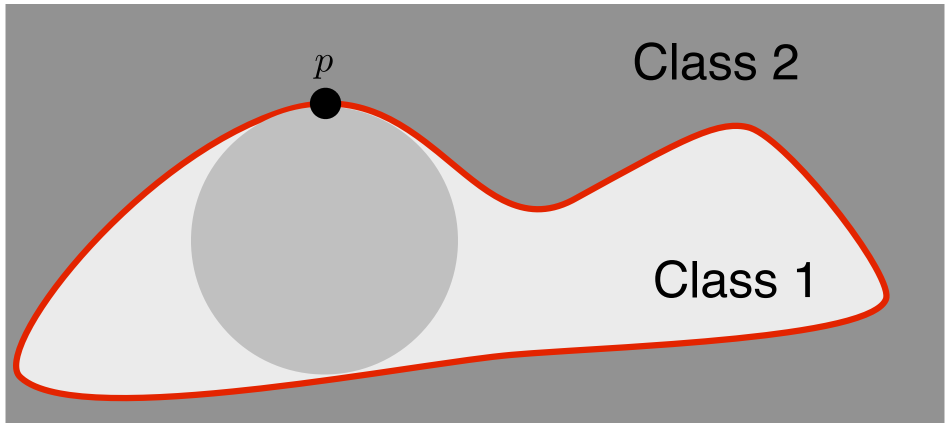
$$\beta(f;m) = \sqrt{m/d}\frac{1}{|\mathscr{D}|}\sum_{\boldsymbol{x}\in\mathscr{D}}\frac{\|\boldsymbol{r}_S^*(\boldsymbol{x})\|_2}{\|\boldsymbol{r}^*(\boldsymbol{x}))\|_2}$$

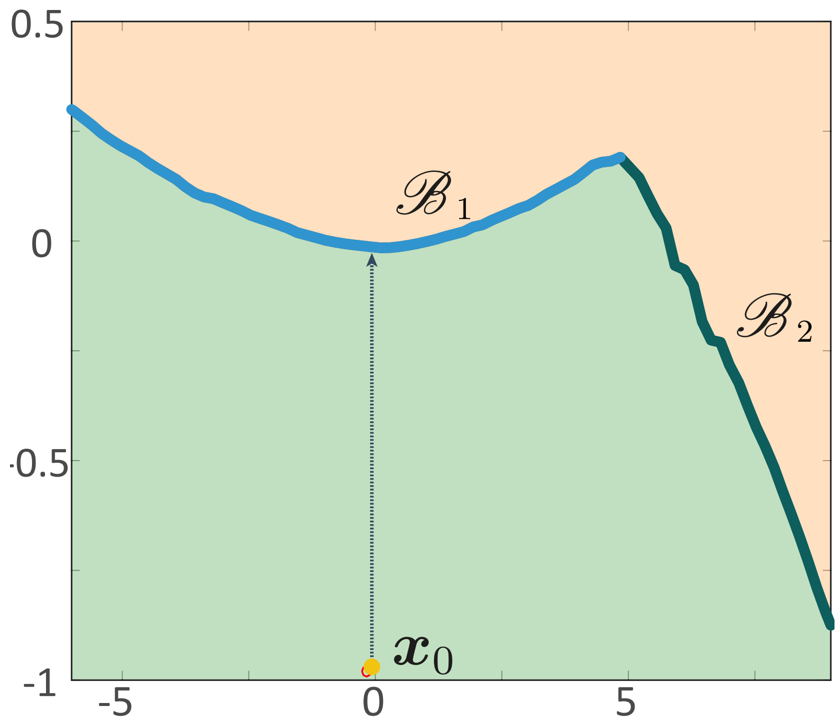
	m/d					
Classifier	**1**	**1/4**	**1/16**	**1/36**	**1/64**	**1/100**
LeNet (MNIST)	1.00	1.00 ± 0.06	1.01 ± 0.12	1.03 ± 0.20	1.01 ± 0.26	1.05 ± 0.34
LeNet (CIFAR-10)	1.00	1.01 ± 0.03	1.02 ± 0.07	1.04 ± 0.10	1.06 ± 0.14	1.10 ± 0.19
VGG-F (ImageNet)	1.00	1.00 ± 0.01	1.02 ± 0.02	1.03 ± 0.04	1.03 ± 0.05	1.04 ± 0.06
VGG-19 (ImageNet)	1.00	1.00 ± 0.01	1.02 ± 0.03	1.02 ± 0.05	1.03 ± 0.06	1.04 ± 0.08

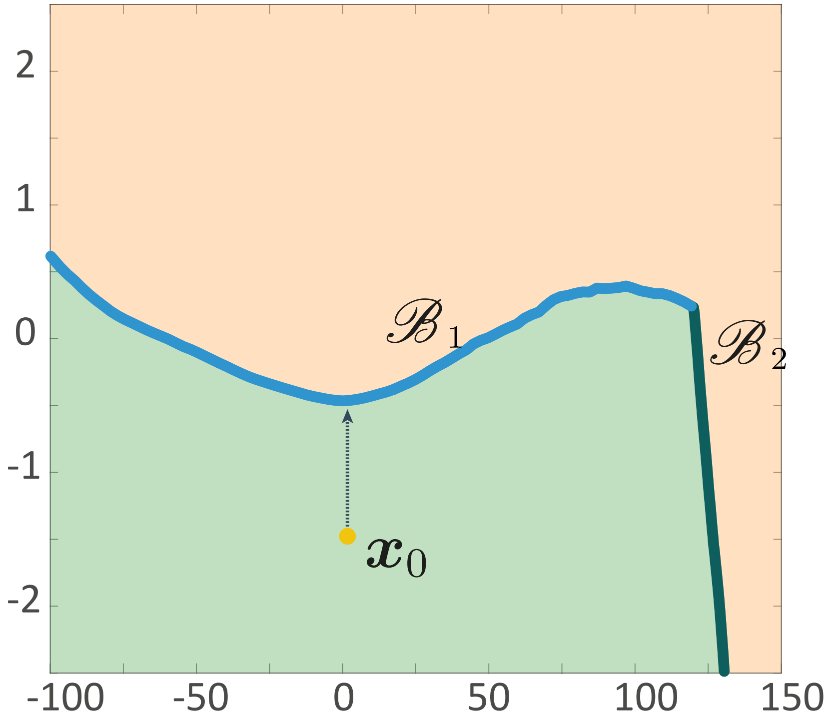
Our quantitative results provide a very accurate estimate of the robustness to semi-random noise!

Why does it work so well?

- Robustness results hold for small curvature
- Curvature seems indeed small in CNNs
  - Two-dimensional cross-sections of classifiers' boundary:







Visual examples

Original

Cauliflower

Random
(**perceptible** noise)

Artichoke

Semi-random (m=10)
(**imperceptible** noise)

Artichoke

Adversarial
(**imperceptible** noise)

Artichoke

Application (?): Hiding messages

- Random positions and scales of “NIPS”, “SPAIN” and “2016”
  - $\mathcal{S}$ = span{random positions and scales of “NIPS”, “SPAIN”, “2016”}.
  - Colors determined in an adversarial way.

Flowerplant

Structured noise

Pineapple

What did we learn so far?

- A semi-random noise regime can interpolate between random and adversarial noise
  - Blessing of dimensionality for robustness to random noise

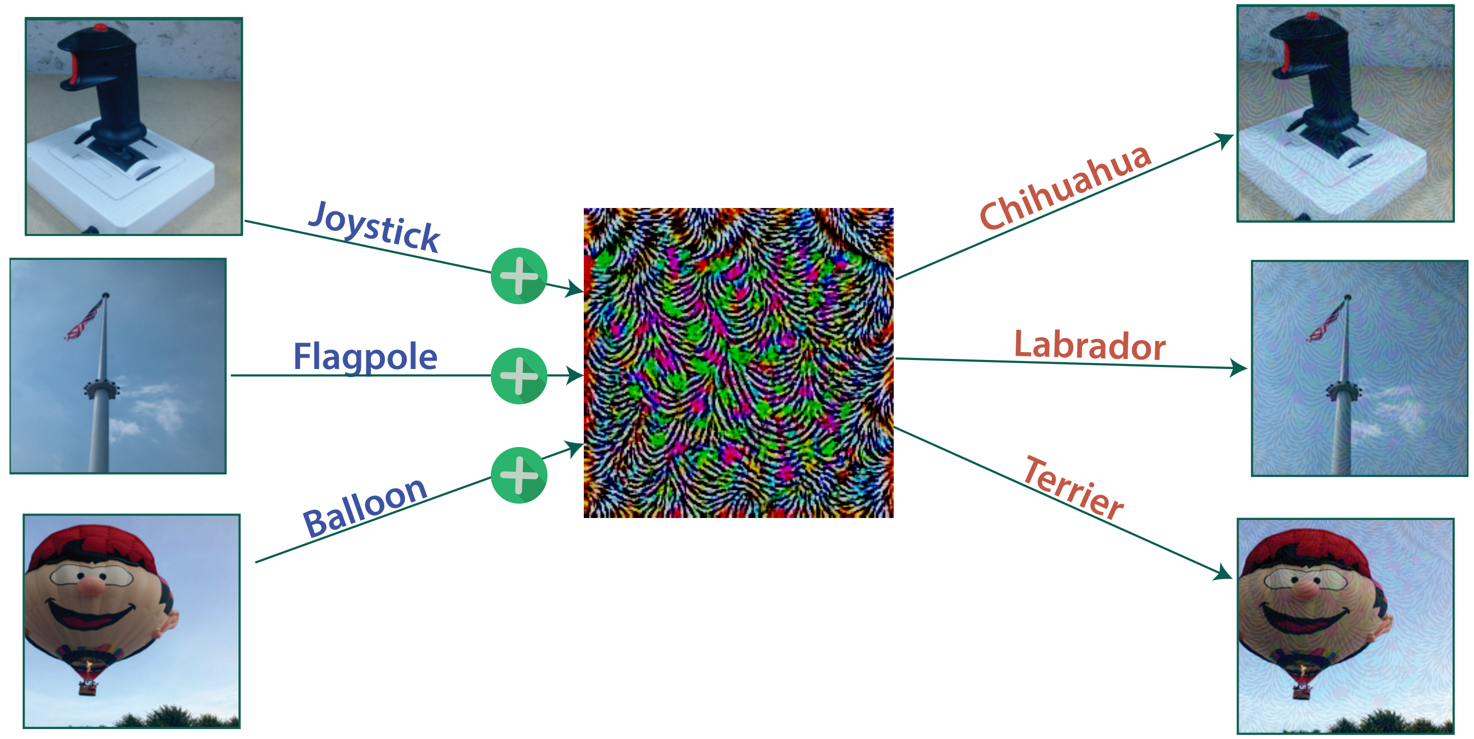
- Even for a very small m, state-of-the-art classifiers are not robust.
  - Experimental results suggest that such classifiers have very flat decision boundaries.
  - We only need to know the classifier in the low-dim. subspace!!

Could it be worse?

Noise 3: Universal

Is there any single (universal) *quasi-imperceptible perturbation that leads to misclassify all images w.h.p?*

Yes! (suprisingly enough!)



Universal adversarial perturbations
Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi and Pascal Frossard
Submitted to IEEE CVPR, December 2017.

Universal perturbations

- Our objective: Given μ the distribution of natural images in $\mathbb{R}^d$ and $\hat{k}$ the classification function, find a *small* v such that $\hat{k}(x+v) \neq \hat{k}(x)$ for *most* natural images.

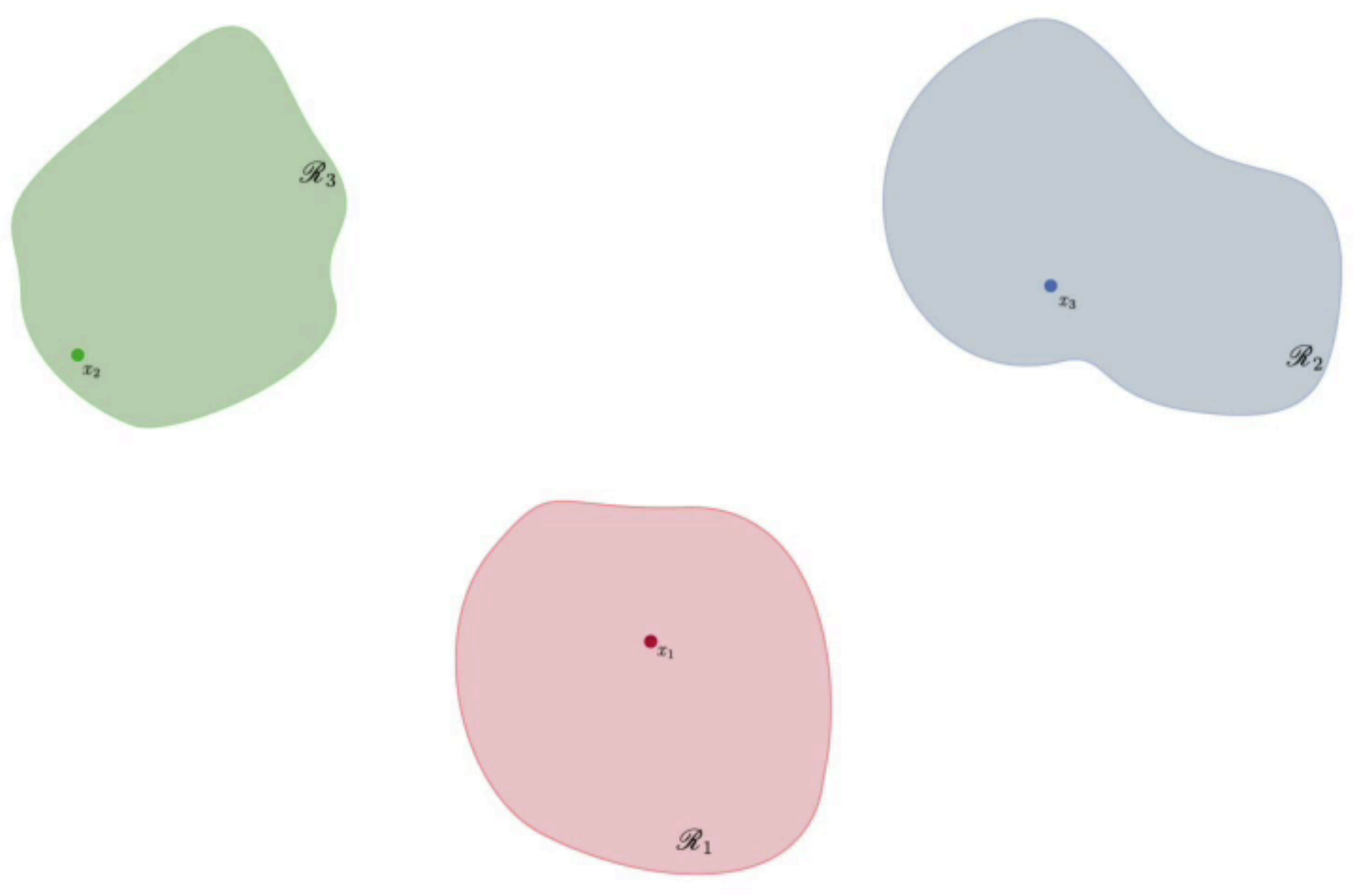
- More formally:

Find v such that:

1. $\|v\|_p \leq \xi$
2. $\underset{x \sim \mu}{\mathbb{P}} \left(\hat{k}(x+v) \neq \hat{k}(x) \right) \geq 1 - \delta$

Computing universal perturbations

- We compute perturbations by summing up perturbations for a subset of training samples $X = \{x_1, \ldots, x_m\}$



Iterative algorithm

Initialize $v \leftarrow 0$.

while the proportion of fooled images in X is $\leq 1 - \delta$ **do**

  for each datapoint $x_i \in X$ **do**

    if v does not fool x_i **then**

      Step 1. Compute perturbation increment.

$$\Delta v_i \leftarrow \arg\min_r \|r\|_2 \text{ s.t. } \hat{k}(x_i + v + r) \neq \hat{k}(x_i).$$

DeepFool

      Step 2. Project the updated perturbation

$$v \leftarrow \text{Projection of } v + \Delta v_i \text{ on the } \ell_p \text{ ball of radius } \xi.$$

    end if

  end for

end while

Robustness of Deep Nets

- Experiments on state-of-the-art deep nets
  - We set $X = 10,000$ training images from the ILSVRC 2012 data set.
  - We pick ξ to guarantee that the perturbation is quasi-imperceptible, when added to the image.
  - We then evaluate the perturbation v on the validation set (images not in the set X).

	CaffeNet	VGG-F	VGG-16	VGG-19	GoogLeNet	ResNet-152
Val.	93.3%	93.7%	78.3%	77.8%	78.9%	84.0%

Rate of images that are fooled, for different networks.

Hard to convince?

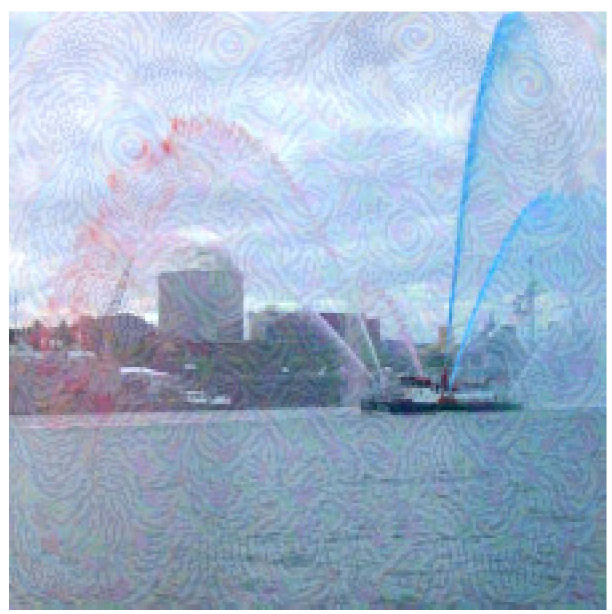
wool

Indian elephant

African grey

triceratops

Indian elephant



hippopotamus

running shoe

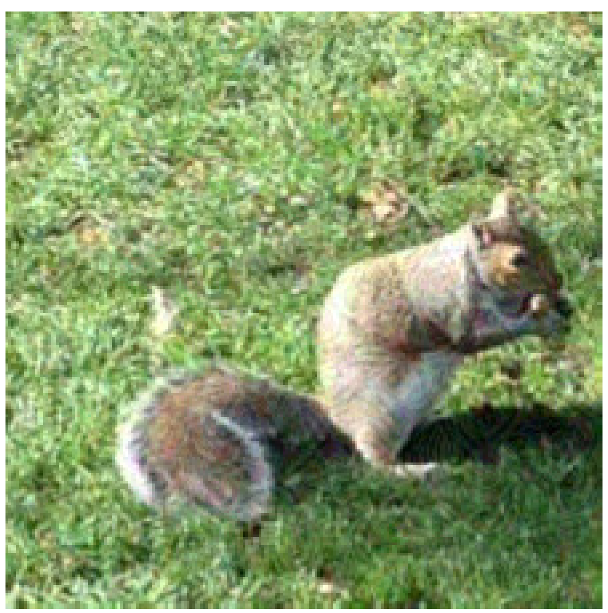
pillow

More examples…

fox squirrel

pot

Arabian camel

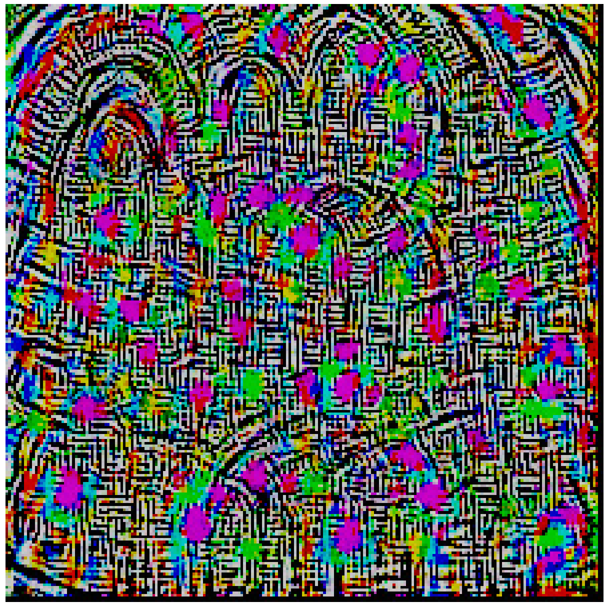


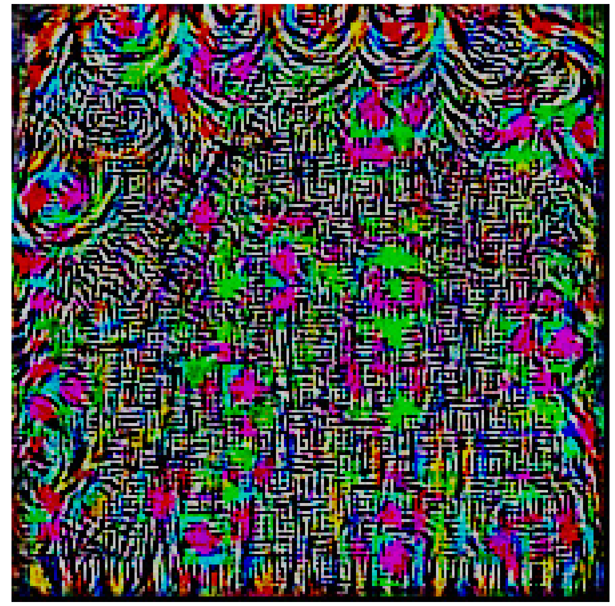
grey fox

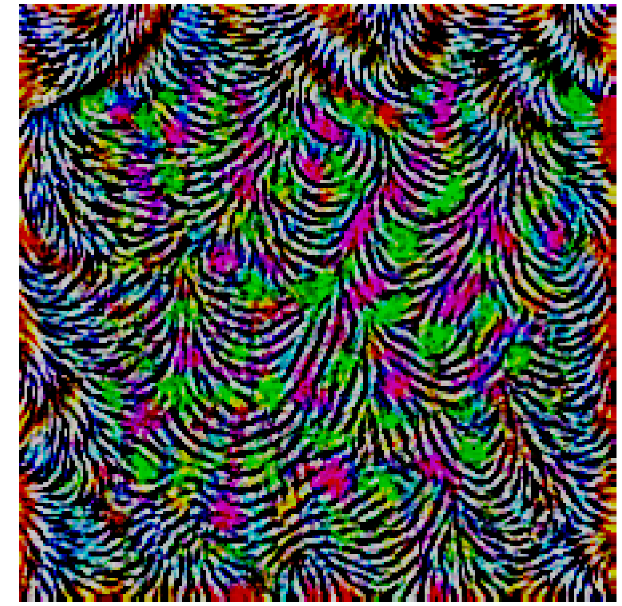
macaw

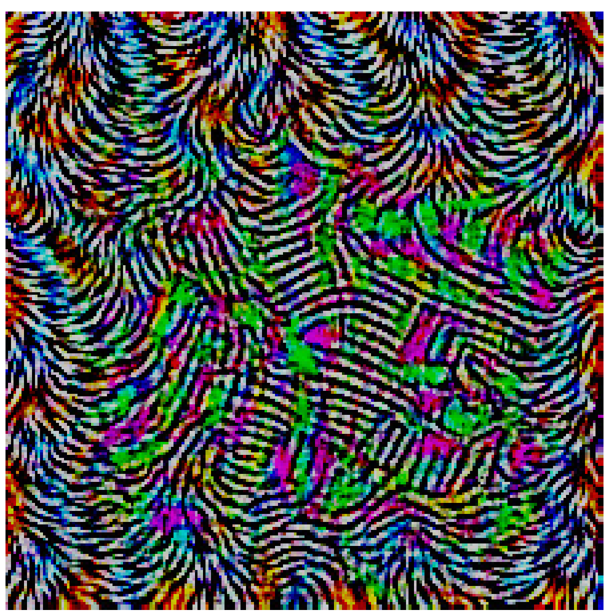
three-toed sloth

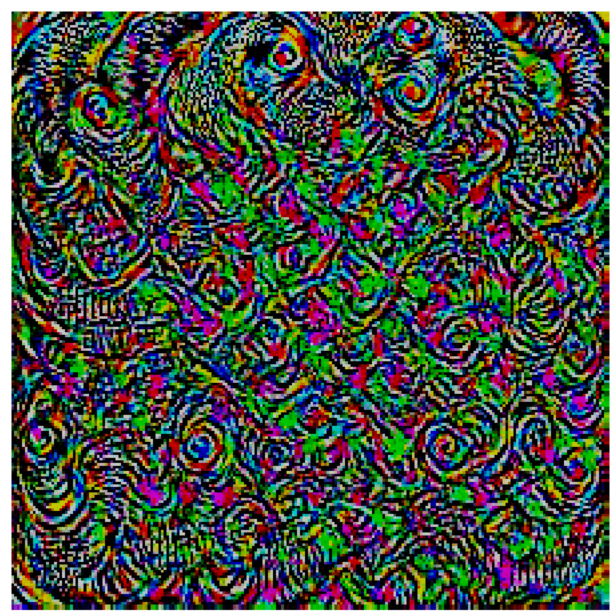
Sample perturbations

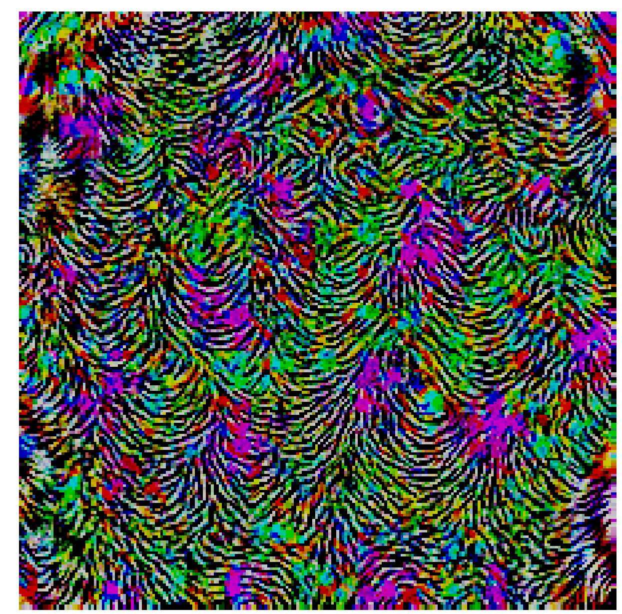

CaffeNet


VGG-F


VGG-16


VGG-19


GoogLeNet


ResNet-152

Doubly universal perturbations

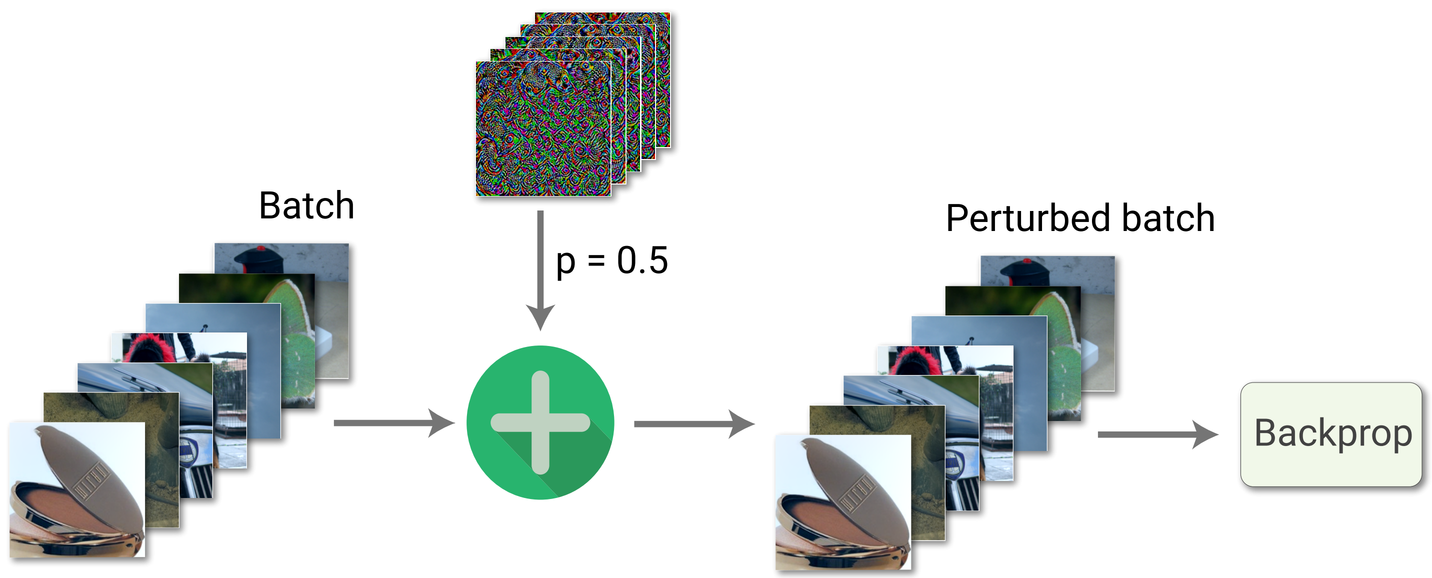
Universal perturbations actually generalize surprisingly well across different neural networks!

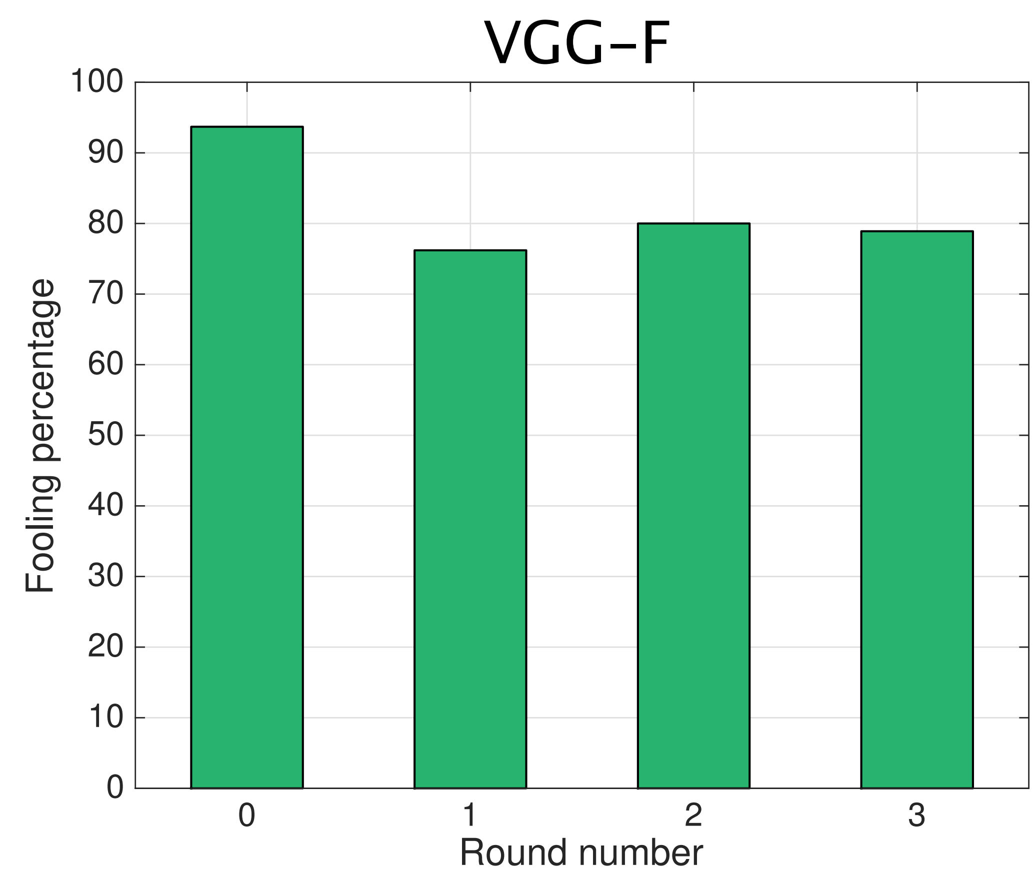
	VGG-F	CaffeNet	GoogLeNet	VGG-16	VGG-19	ResNet-152
VGG-F	**93.7%**	71.8%	48.4%	42.1%	42.1%	47.4 %
CaffeNet	74.0%	**93.3%**	47.7%	39.9%	39.9%	48.0%
GoogLeNet	46.2%	43.8%	**78.9%**	39.2%	39.8%	45.5%
VGG-16	63.4%	55.8%	56.5%	**78.3%**	73.1%	63.4%
VGG-19	64.0%	57.2%	53.6%	73.5%	**77.8%**	58.0%
ResNet-152	46.3%	46.3%	50.5%	47.0%	45.5%	**84.0%**

They are *doubly* universal (wrt data and network)…

Feedbacking

- One can try to improve robustness by feedbacking

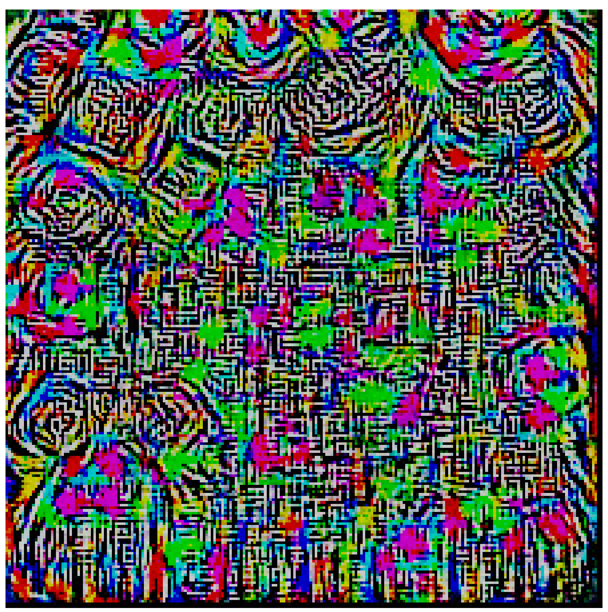




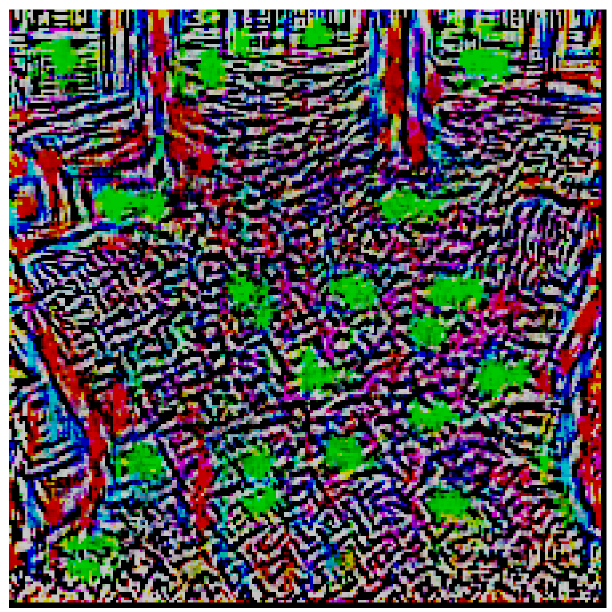
- Only mild improvement in robustness :(

No unique solution

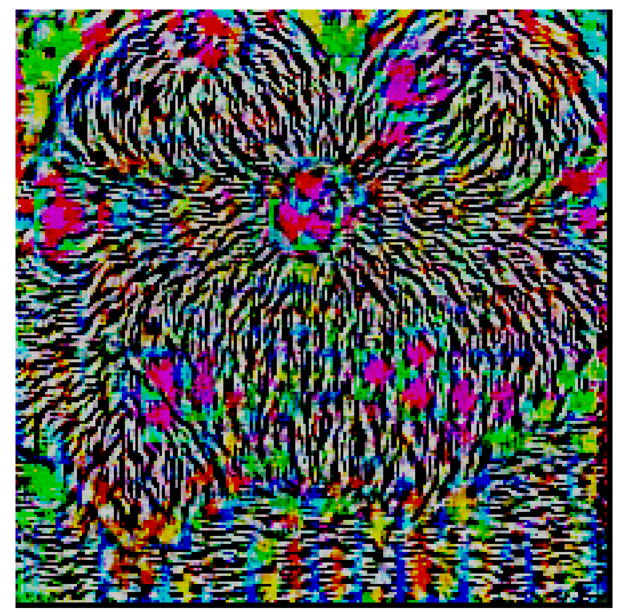
Universal perturbations are far from unique: there exist *many* directions that cause classifier to misclassify.



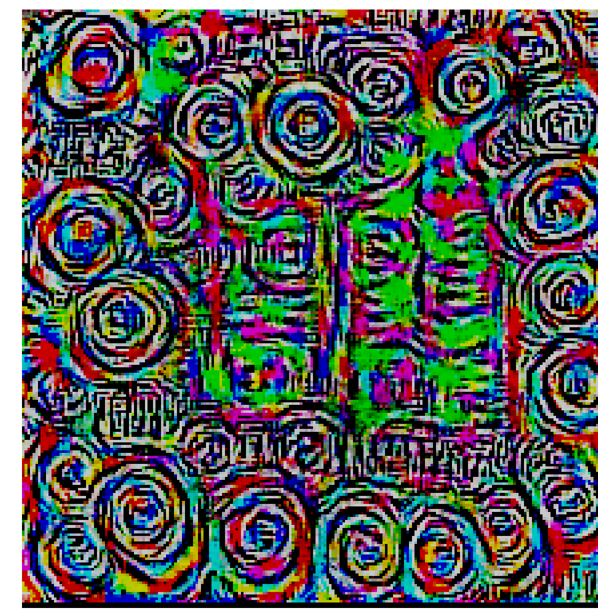
Round 0



Round 1



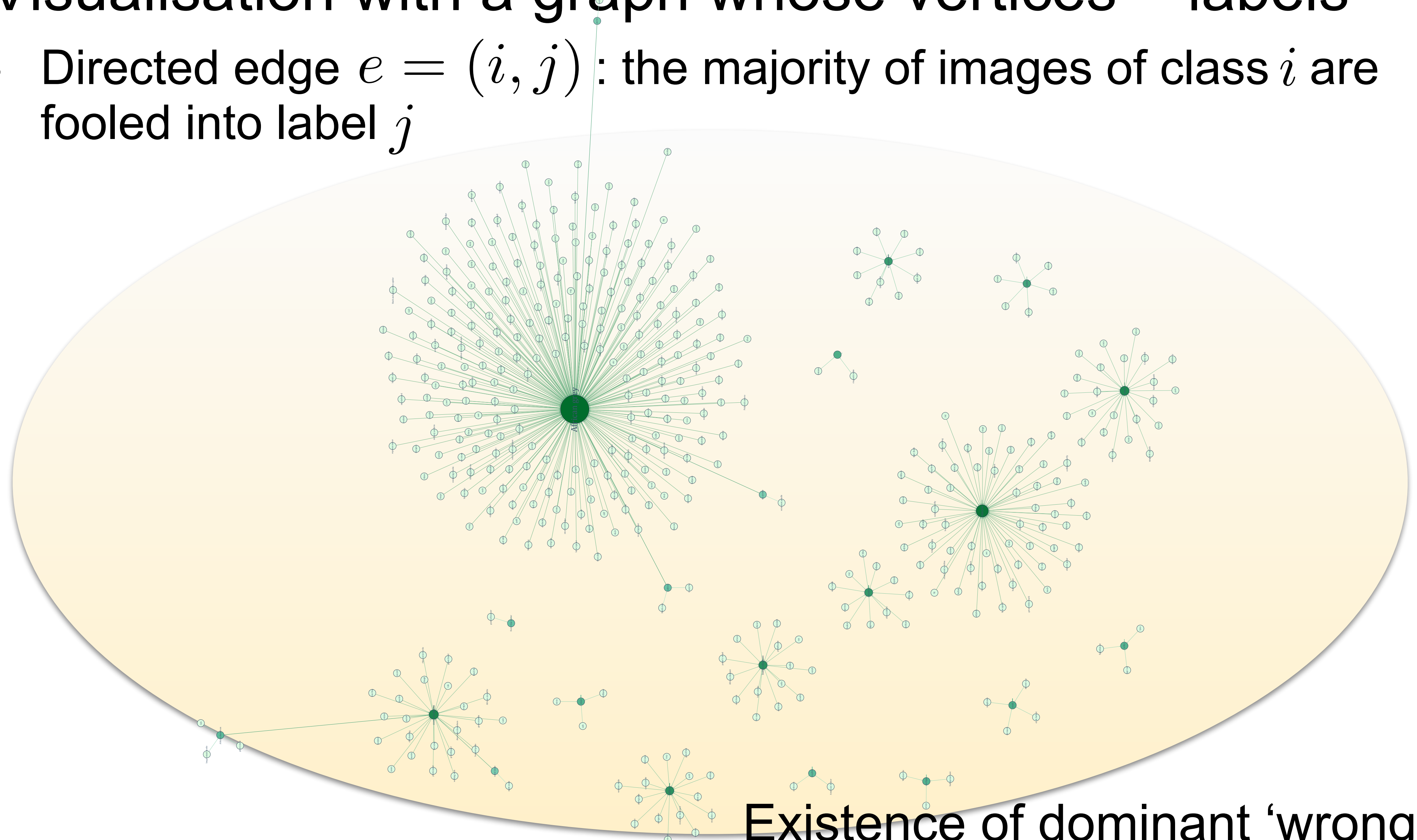
Round 2



Round 3

Effect of perturbations

- Visualisation with a graph whose vertices = labels
  - Directed edge $e = (i, j)$: the majority of images of class i are fooled into label j

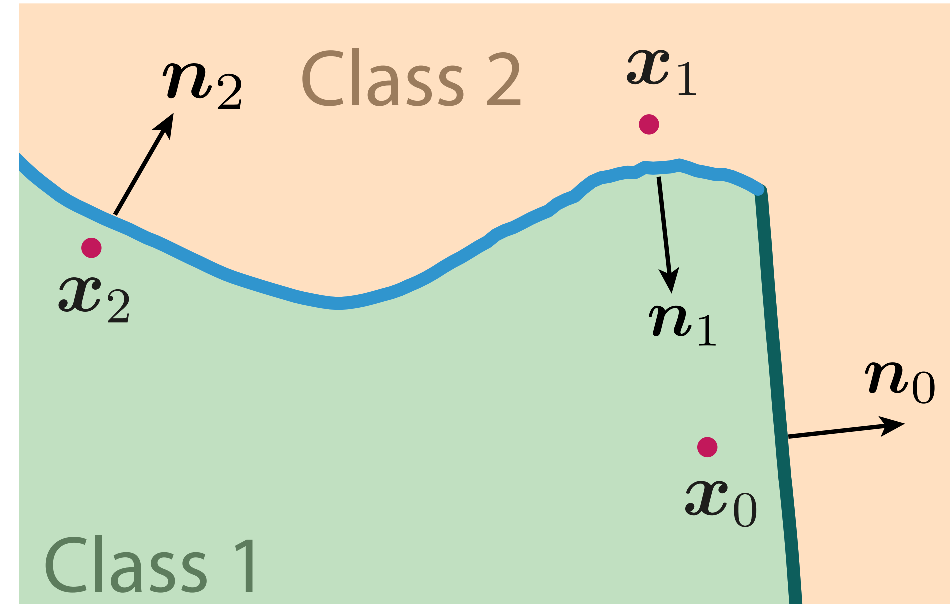


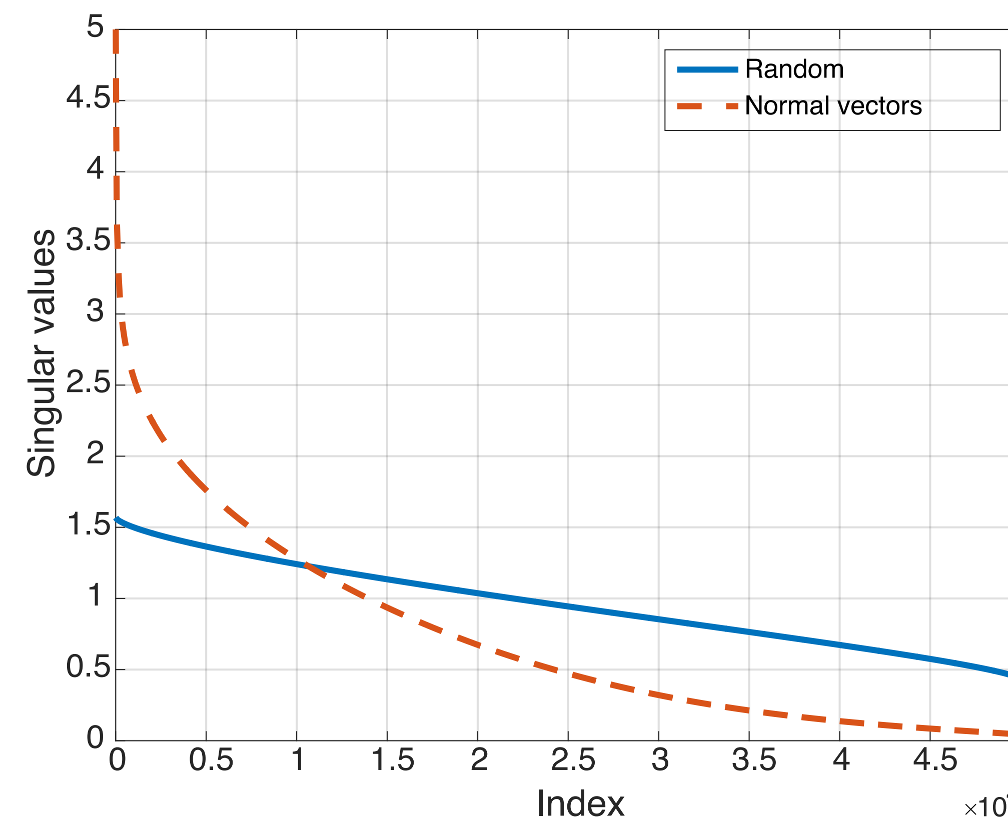
Existence of dominant 'wrong' labels!

First explanations…

- Geometric correlations between regions of the decision boundary
  - Define the matrix of normal vectors to the decision boundary in the vicinity of k natural images.

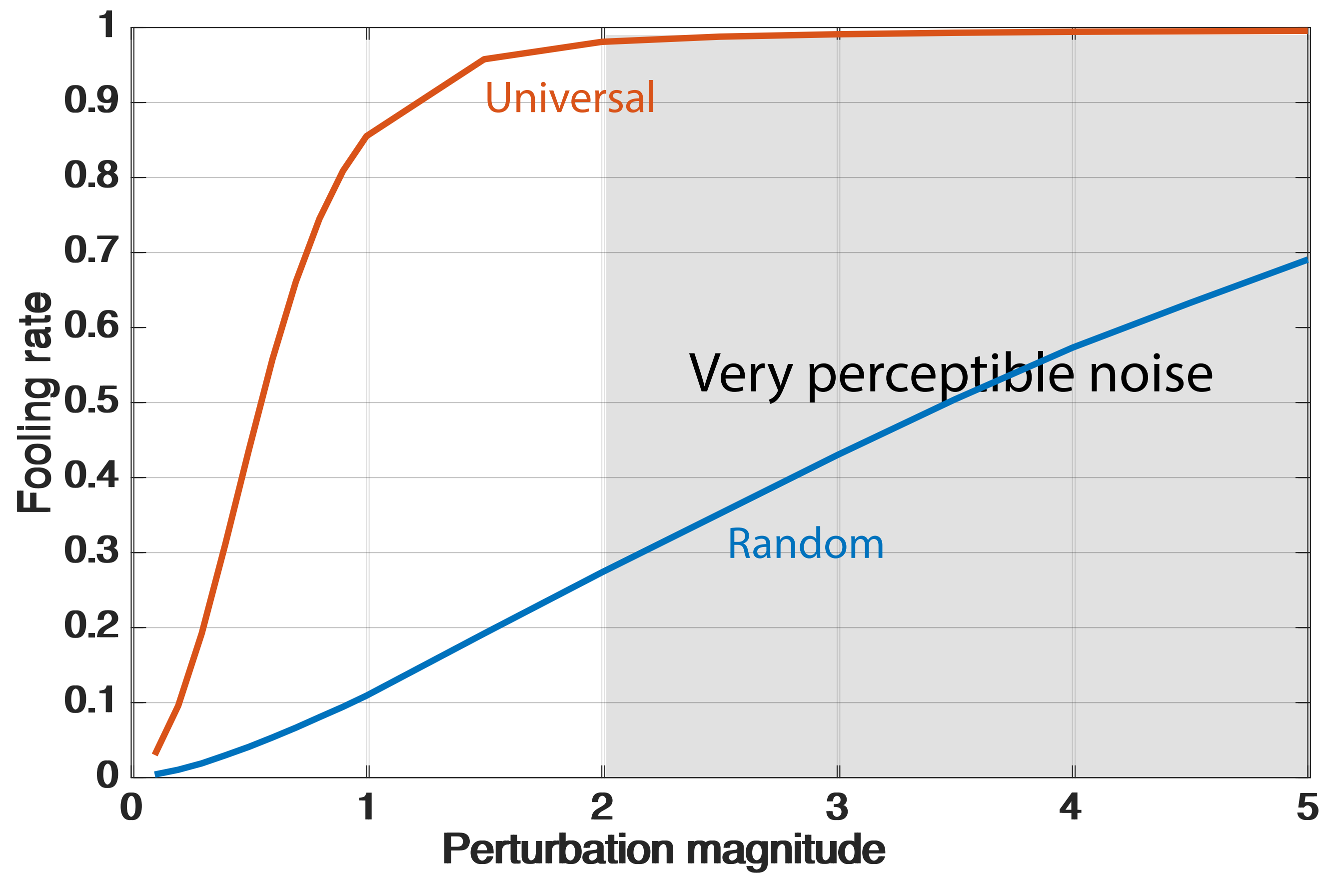
$$\mathbf{N} = [\boldsymbol{n}_0 | \dots | \boldsymbol{n}_{k-1}]$$





- Existence of a low-dimensional subspace containing most normal vectors
- Universal perturbations belong to this subspace of normal vectors

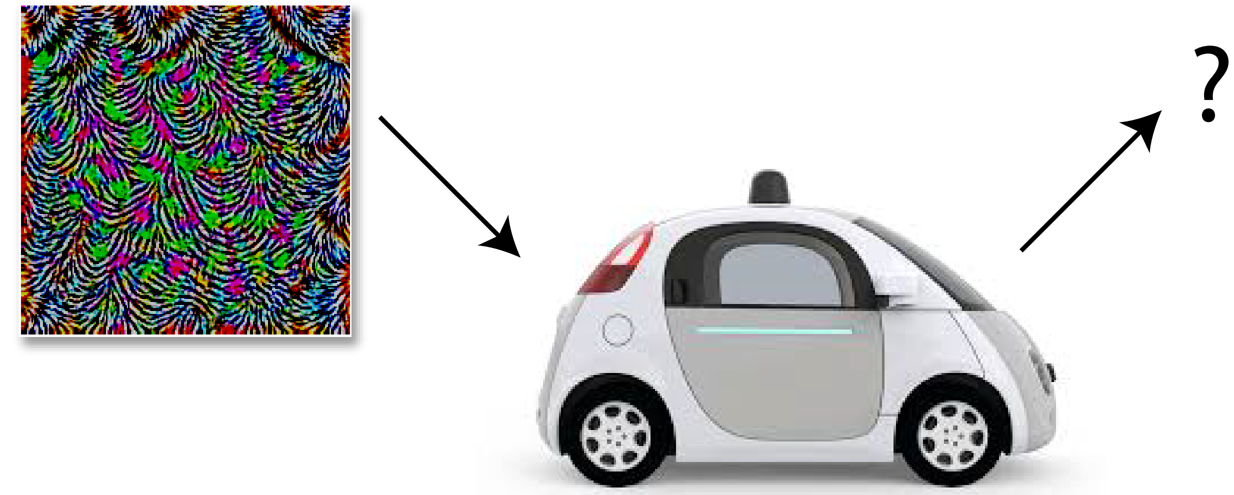
Insights from universal noise



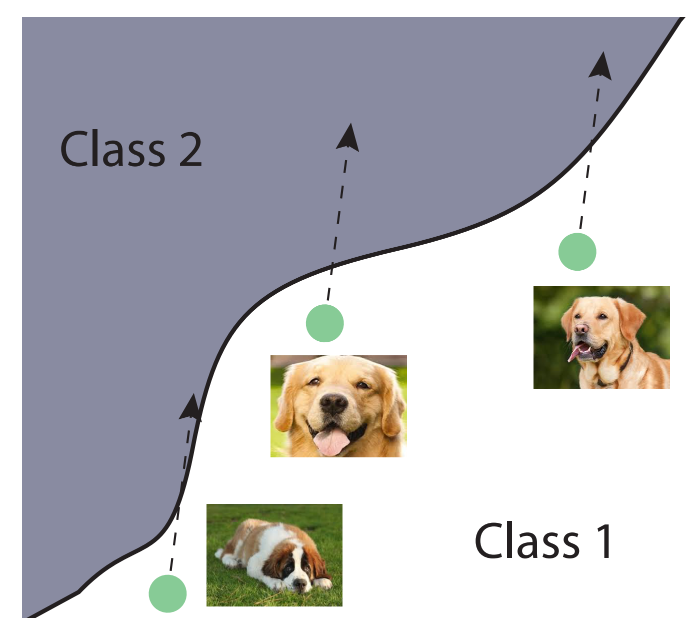
- State-of-the-art deep nets are not robust to universal (image-agnostic) perturbations.
- These perturbations are doubly-universal, to some extent.
- This suggests the existence of high correlation between different regions in the decision boundary of the classifier

Why should we care?

- Such perturbations can be relatively straightforward to implement by adversaries



- They may lead to a better understanding of the geometry of state-of-the-art classifiers.

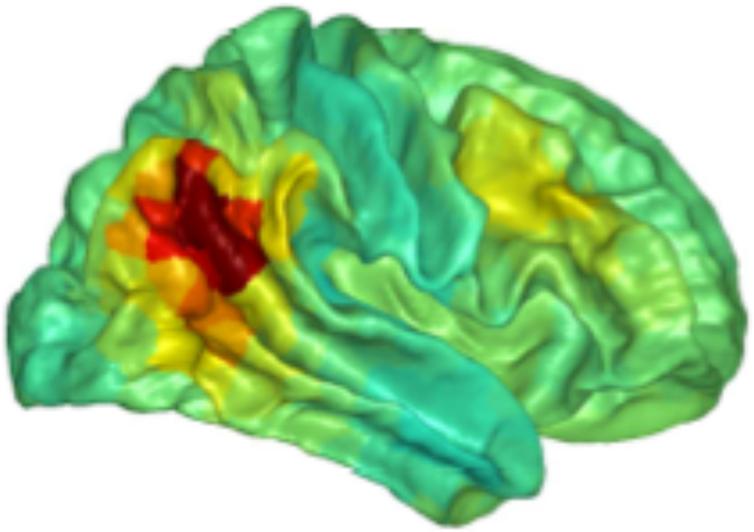


Demo

Conclusions

- Image analysis systems may have important limitations
  - Lack of robustness to perturbations (empirically and theoretically)

- Future works
  - Fundamental limits on the robustness of deep nets
  - Methods to find adversarial perturbations using only limited knowledge of the classifier
  - Visualization of high dimensional decision boundaries
  - New architectures with improved robustness



- More promising paths?
  - Better representation models?
  - Back to HVS inspirations?

References

- S. Moosavi, A. Fawzi, O. Fawzi, P. Frossard, *Universal adversarial perturbations*, arXiv preprint arXiv:1610.08401, 2016.
- A. Fawzi, S. Moosavi, P. Frossard, *Robustness of classifiers: from adversarial to random noise*, NIPS 2016.
- S. Moosavi, A. Fawzi, P. Frossard, *DeepFool: a simple and accurate method for fooling deep neural networks*, CVPR 2016.
- A. Fawzi, O. Fawzi, P. Frossard, *Analysis of classifiers' robustness to adversarial perturbations*, accepted for publication, Machine Learning Journal, 2016.
- A. Fawzi, P. Frossard, *Measuring the effect of nuisance variables on classifiers*, BMVC 2016.
- A. Fawzi, P. Frossard, *Manitest: are classifiers really invariant?*, BMVC 2015.