Informative Dropout for Robust Representation Learning: A Shape-bias Perspective

Baifeng Shi*, Dinghuai Zhang*, Qi Dai, Zhanxing Zhu, Yadong Mu, Jingdong Wang

Contents

CNN is not robust

CNN is not robust

VS.

VS.

😀

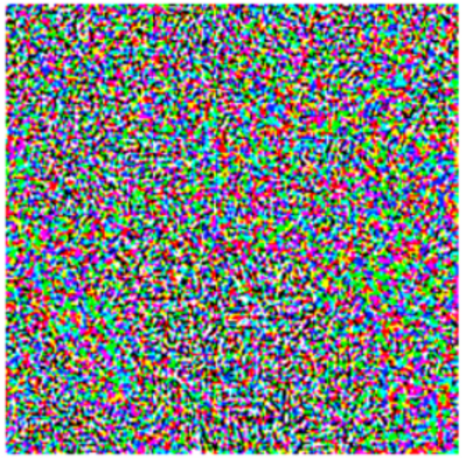
😵

CNN is not robust

😀 😵

CNN is not robust

$+ .007 \times$



$=$

“panda” noise “gibbon”

😀 😵

CNN is biased towards texture

(a) Texture image
81.4% **Indian elephant**
10.3% indri
8.2% black swan

(b) Content image
71.1% **tabby cat**
17.3% grey fox
3.3% Siamese cat

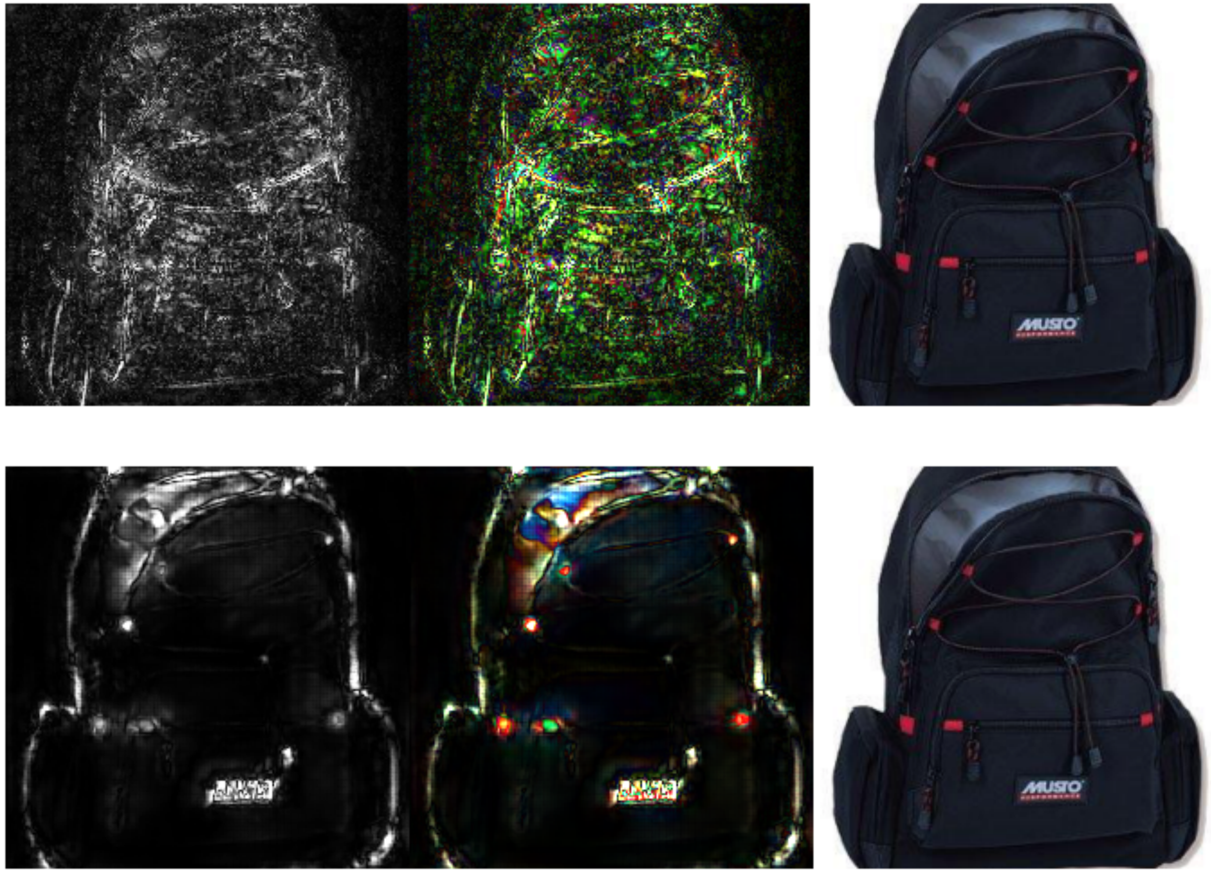
(c) Texture-shape cue conflict
63.9% **Indian elephant**
26.4% indri
9.6% black swan

ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness, R Geirhos et al., ICLR' 19

Robustness -> shape-bias

Regular CNN

Adversarially-trained CNN



Interpreting Adversarially Trained Convolutional Neural Networks, Tianyuan Zhang et al., ICML'

Is texture-bias a common reason for CNN’s non-robustness?

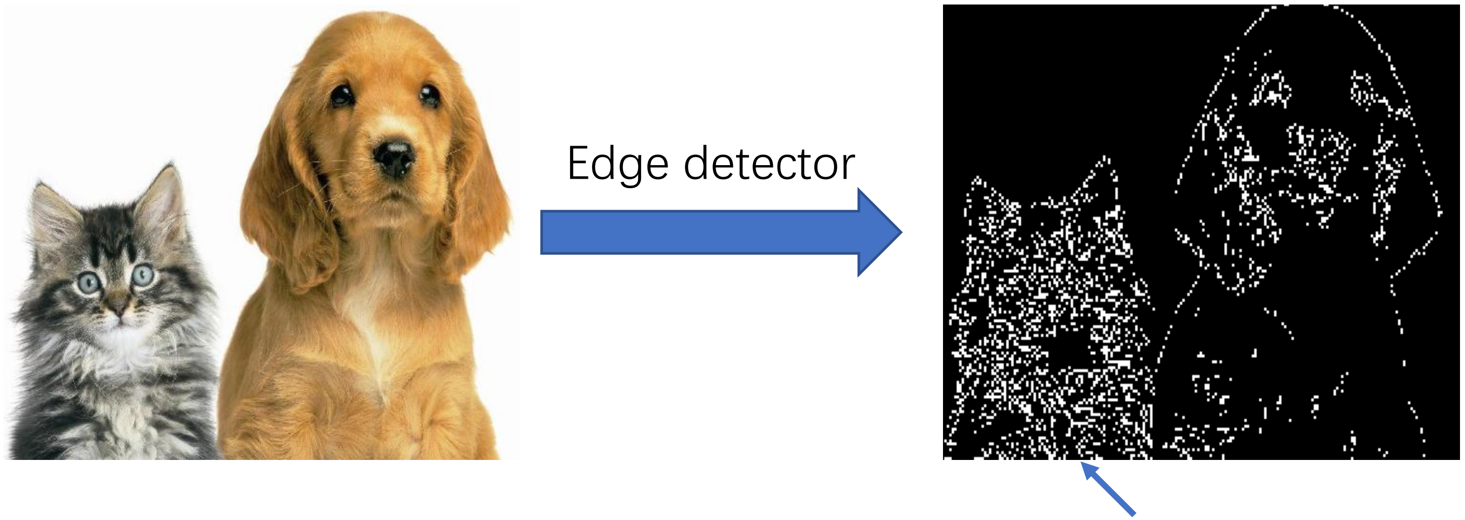
Overview

- Our motivation: Improve robustness by training a shape-biased model
- Methodology:
  - Design an algorithm to automatically detect shape/texture
  - Train a model to be insensitive to texture
- Experiments:
  - Is our model more shape-biased?
  - Is our model more robust?
    - domain generalization, few-shot learning, random corruption, adversarial perturbation

Methodology

How to detect shape/texture?

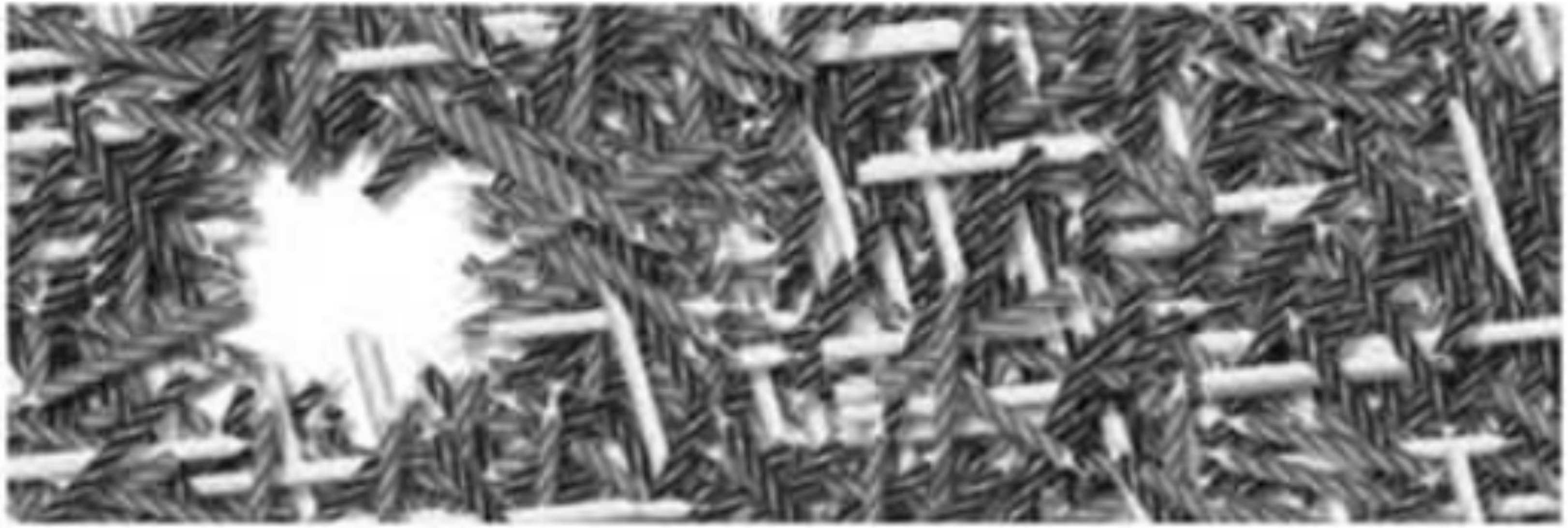
- Edge detection?
  - not robust to complex texture



Poor little cat…

Eye fixation and saliency detection

- Humans tend to look at regions with **high self-information（“surprise”）**



Saliency Based on Information Maximization, N Bruce et al., NIPS’ 06

Information-based detector

- Shannon self-information of event x:

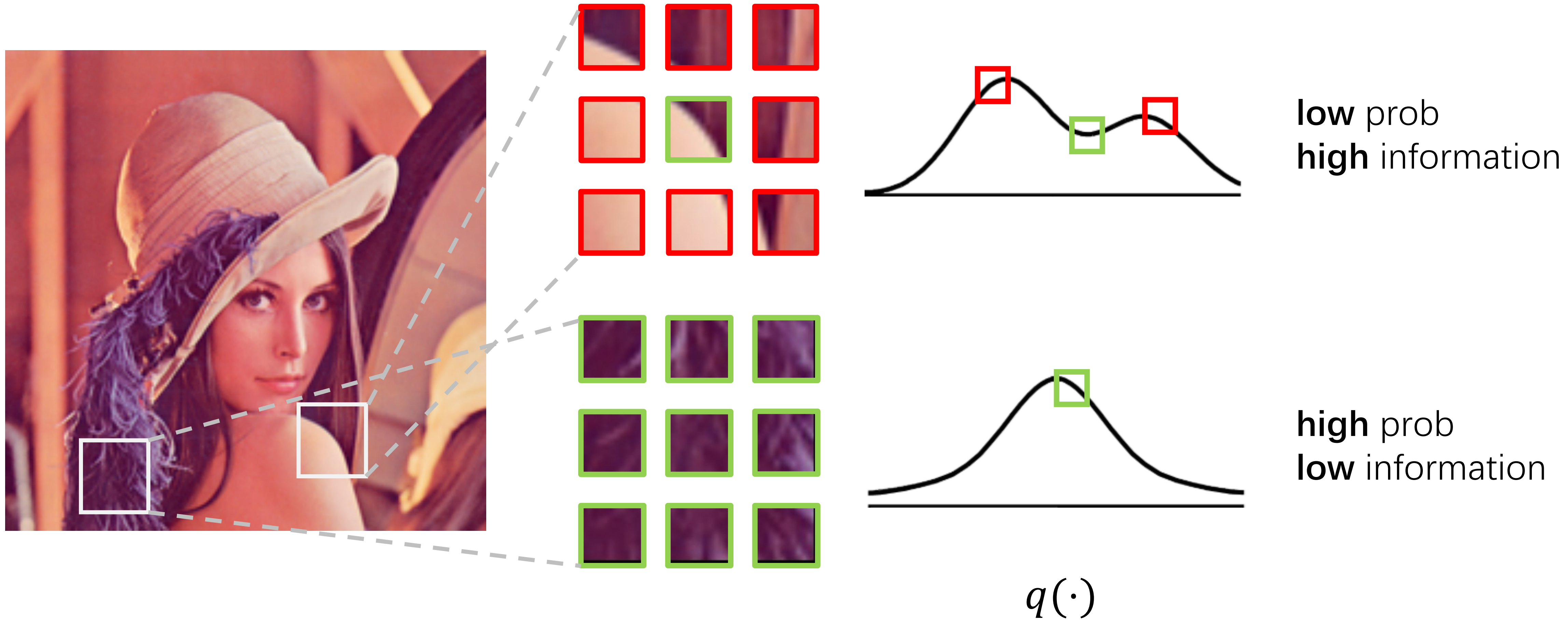
$$I(x) = -\log q(x).$$

- For each patch p in an image, it contains self-information of

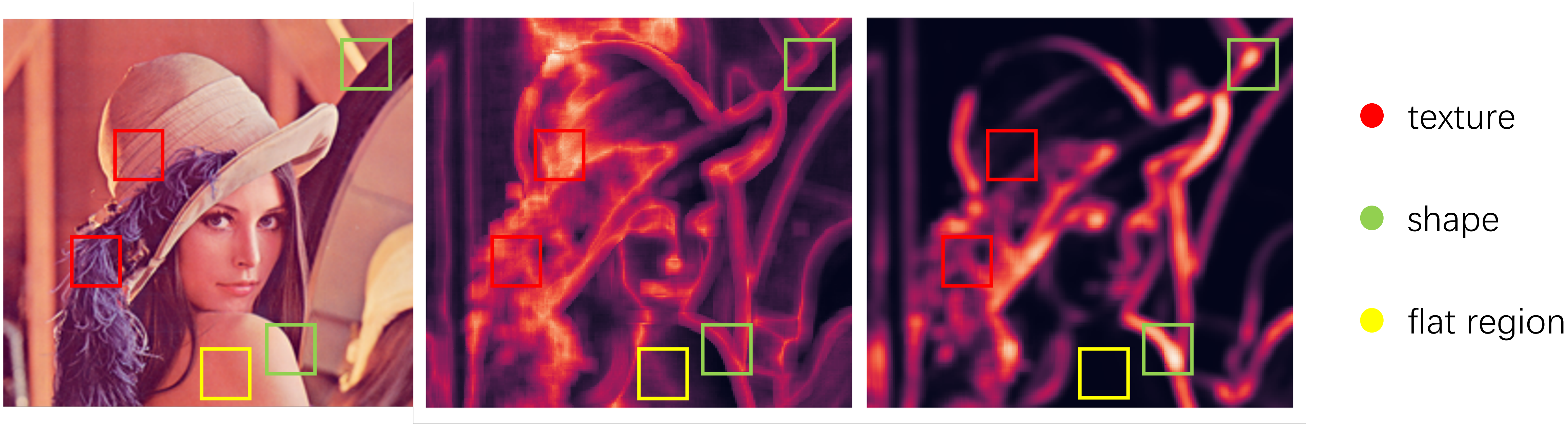
$$I(p) = -\log q(p),$$

where $q(\cdot)$ is the patch distribution in the neighborhood of p.

Information-based detector



An intuitive explanation



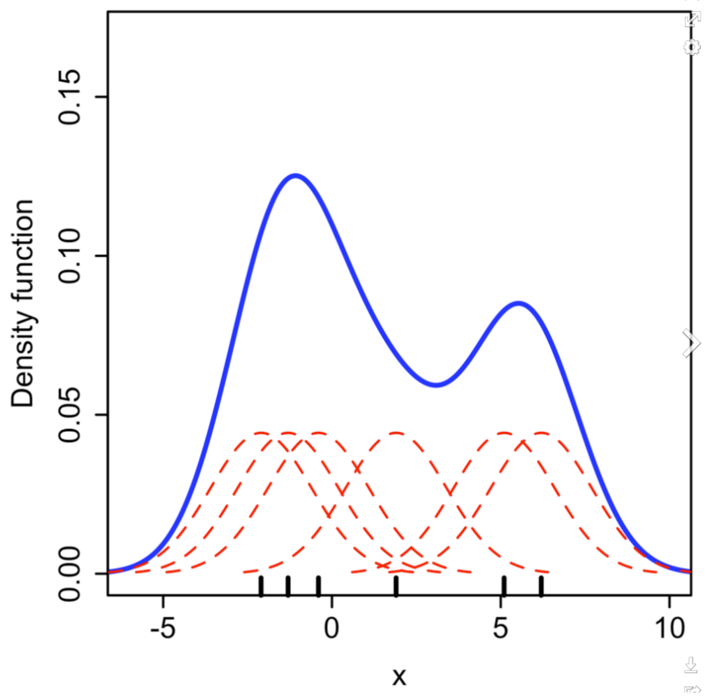
(a) original image (b) frequency map (c) self-information map

How to approximate $q(p)$

- With the patches in the neighborhood $N(p)$ as samples, we use the kernel density estimator $\hat{q}(p)$ to approximate $q(p)$:

$$\hat{q}(p) = \frac{1}{|N(p)|} \sum_{p' \in N(p)} K(p, p'),$$

where K is the kernel (e.g. Gaussian).

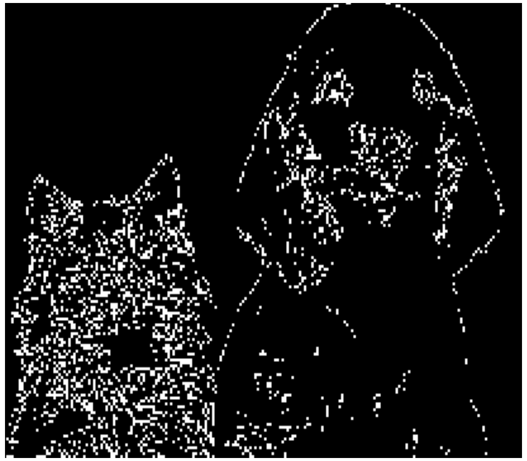


Information-based detector

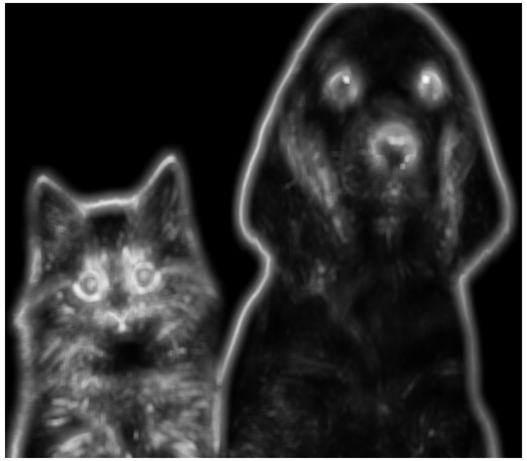
- Now we can estimate the self-information of p through:

$$I(p) = -\log \hat{q}(p) = -\log \frac{1}{|N(p)|} \sum_{p' \in N(p)} K(p, p').$$

(a) Original image



(b) Edge detection



(c) Information-guided

From images to feature maps

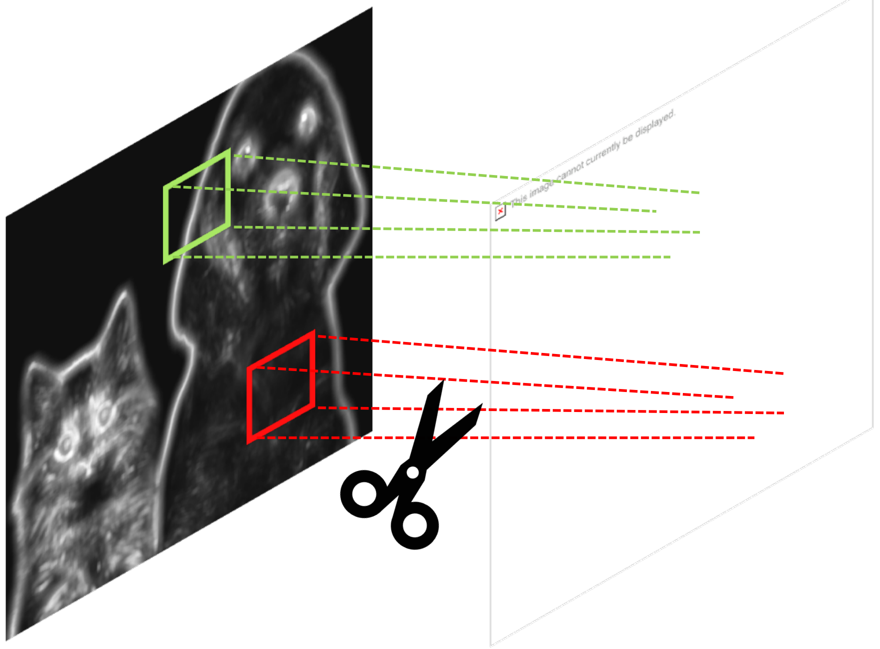
- We can also estimate the self-information of patches in a feature map.

- We find it the best practice to use our method on input image AND feature maps in CNN’ s early layers.

Towards a shape-biased model

- Objective: make the model **insensitive** to low-information regions (texture)

- Our approach: a dropout-like algorithm



Lower information -> higher drop rate

Informative Dropout (InfoDrop)

- If a neuron $z = \sigma(k \cdot p + b)$ is the output from an input patch, where k is the convolution kernel, b is the bias and σ is the activation function, then the drop rate of z is

$$r(z) \propto e^{-\frac{I(p)}{T}},$$

where T is temperature.

“Internal” shape-bias

During inference:

Use InfoDrop to “intentionally” remove texture

The convolution kernels can automatically filter out texture

“Internal” shape-bias

- We want to throw away InfoDrop during inference

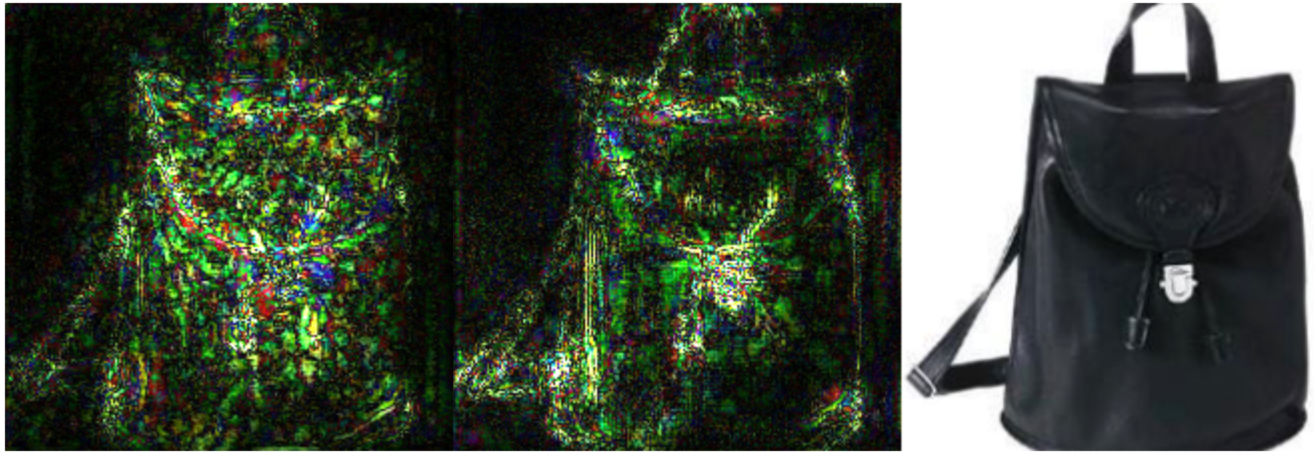
- Directly removing it may cause troubles
  - e.g. statistical mismatch in BatchNorm

- We first train with InfoDrop on, and then **remove InfoDrop and finetune** on the training data.

Experiments

Is our model more shape-biased now?

- Gradient-based saliency
- For input image x, the saliency $S(x) = \frac{1}{n}\sum_{i=1}^{n} \frac{\partial f(x+\delta_i)}{\partial x}$, where f is the network and δ_i is random noise.



regular CNN w/ InfoDrop input image

Is our model more shape-biased now?

- Style Transfer
- Add InfoDrop to extract and transfer only shape feature

Is our model more robust now?

- Domain generalization
  - **distribution shift between training/test images**
  - PACS dataset: 4 domains (photo, art, cartoon, sketch)
- After applying InfoDrop:

Source \ Target	PHOTO	ART	CARTOON	SKETCH
PHOTO	*-0.06*	**+2.49**	**+6.52**	**+14.76**
ART	**+0.12**	*+0.20*	**+2.30**	**+0.81**
CARTOON	-0.84	-0.44	*+0.04*	**+4.81**
SKETCH	**+11.91**	**+4.23**	**+6.19**	*+0.15*

Is our model more robust now?

- Few-shot Classification
  - **class-wise distribution shift**
  - CUB dataset
    - finegrained classification
  - Various baselines
    - ProtoNet, MatchingNet, RelationNet

	5-SHOT	1-SHOT
MATCHINGNET	71.18 +- 0.70	57.81 +- 0.88
+ INFODROP	**71.86 +- 0.72**	**58.06 +- 0.92**
PROTONET	67.13 +- 0.74	51.62 +- 0.90
+ INFODROP	**70.18 +- 0.73**	**52.70 +- 0.86**
RELATIONNET	69.85 +- 0.75	56.71 +- 1.01
+ INFODROP	**73.27 +- 0.69**	**60.74 +- 0.97**

Is our model more robust now?

- Random image corruption
  - Caltech-256 dataset
  - Corruption function from Imagenet-C

Table 6. Classification accuracy on clean and randomly corrupted images. 'A' and 'I' means usage of adversarial training and InfoDrop, respectively. All corruptions are generated under severity of level 1 (Hendrycks & Dietterich, 2019).

A	I	CLEAN	NOISE			BLUR			WEATHER			DIGITAL		
			GAUSSIAN	SHOT	IMPULSE	DEFOCUS	MOTION	GAUSSIAN	SNOW	FROST	FOG	ELASTIC	JPEG	SATURATE
✗	✗	**82.98**	66.38	62.85	49.97	**65.97**	**74.79**	**78.75**	53.10	67.09	72.42	**76.58**	79.77	77.15
✗	✓	**83.14**	69.58	66.83	53.00	62.52	71.76	77.03	56.44	**69.80**	**72.75**	74.54	**80.49**	**77.77**
✓	✗	79.69	75.30	73.80	70.71	61.53	71.68	73.77	61.11	69.06	54.52	71.69	79.31	72.62
✓	✓	78.59	**76.17**	**74.90**	**72.26**	62.32	71.32	74.04	**61.69**	**69.83**	55.00	70.26	78.10	71.26

Is our model more robust now?

- Adversarial perturbation
  - CIFAR-10 dataset
  - 20 runs of PGD, $l_{inf} = \frac{8}{255}$
  - Adversarial training w/ InfoDrop

	CLEAN ACC	ADV ACC
ADV TRAINING	86.62	42.05
+ INFODROP	86.59	43.07

Take home messages

- Enhancing shape-bias can improve various kinds of robustness.
- We can discriminate shape from texture based on self-information.
- We can alleviate texture-bias through InfoDrop, an information-based add-on during training only.
- With InfoDrop applied, CNN is more robust against distribution shift (domain generalization, few-shot learning), image corruption and adversarial perturbation.

Many thanks to all the collaborators!

Code will be available on GitHub:
https://github.com/bfshi/InfoDrop

Contact: Baifeng Shi

- https://bfshi.github.io/
- bfshi@pku.edu.cn