CutMix: Regularization Strategy to Train Strong Classifiers with Localizable Features

Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, Youngjoon Yoo

ICCV, 2019, Naver Clova AI

Wonhee Cho

Vision and Learning Laboratory
School of Computer Science and Engineering, Chung-Ang University, Seoul, Republic of Korea
Emails : wonhee4274@cau.ac.kr

Index

Sangdoo Yun, Clova AI Naver

Dongyoon Han, Clova AI Naver

Seong Joon Oh, Clova AI LINE+

Sanghyuk Chun, Clova AI Naver

Junsuk Choe*, Yonsei University

Youngjoon Yoo, Clova AI Naver

Introduction

How to get a model with high-performance??

Accuracy

Generalization

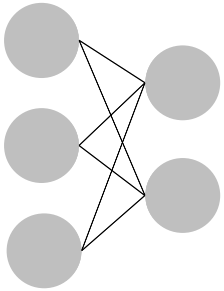
☞ All we need is Augmentation!

Introduction

In the early image classification

Input Image

Model



I'm shallow and weak,
but I'll try to learn.

Target

Bear = 1.0

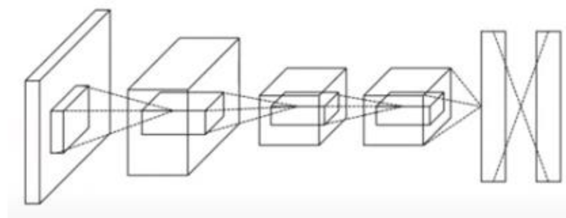
Introduction

Nowadays

I'm deep and complex.
It's easy, maybe I can remember them!

Input Image

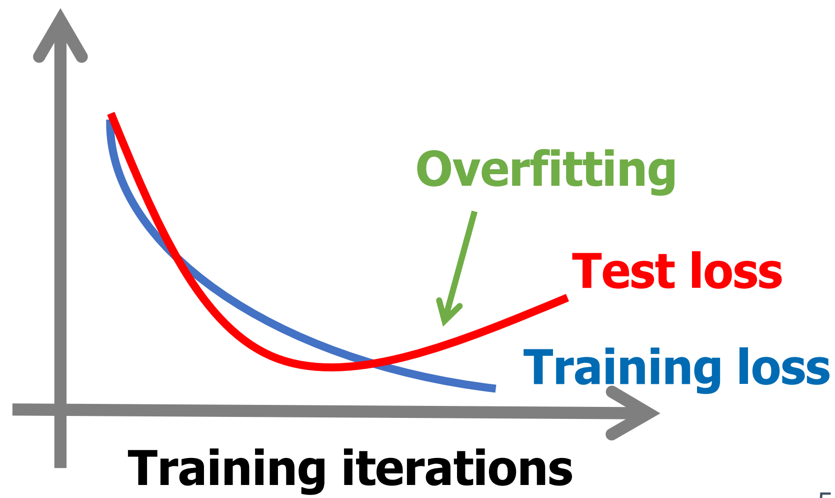
Model



e.g) ResNet, DenseNet, EfficientNet, etc.

Target

Bear = 1.0



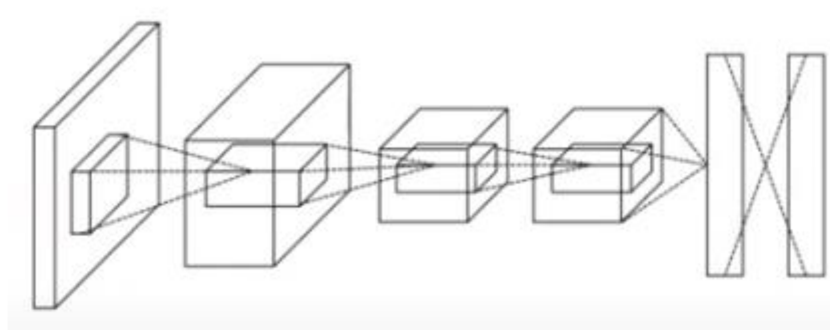
Goal

- **Better generalization and robustness**
  - Improve image classification accuracy
  - Improve transfer learning performance
- **Simple and easy to use**
  - No modification of network architecture (e.g., attention module)
  - No additional training cost (e.g., adversarial training)

Related Works

Data Augmentation

- Decrease the gap between training and test data by **transforming** training data.



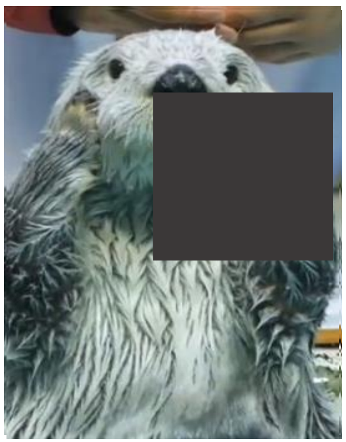
Tiger

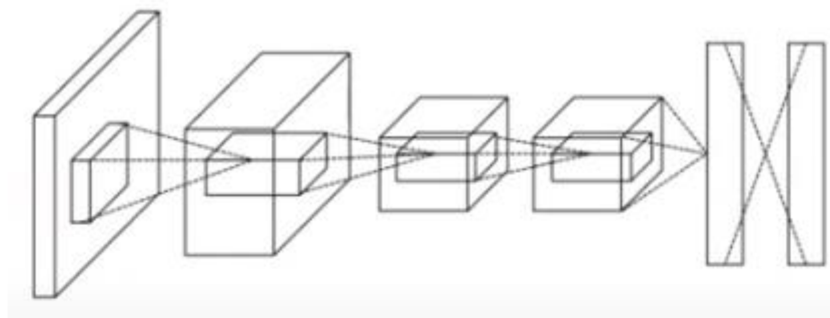
Related Works

Regularization

- Regional Dropout[1,2]: randomly remove image regions.

☞ Make "occlusion-robust" model





Sea Otter

Good generalization ability

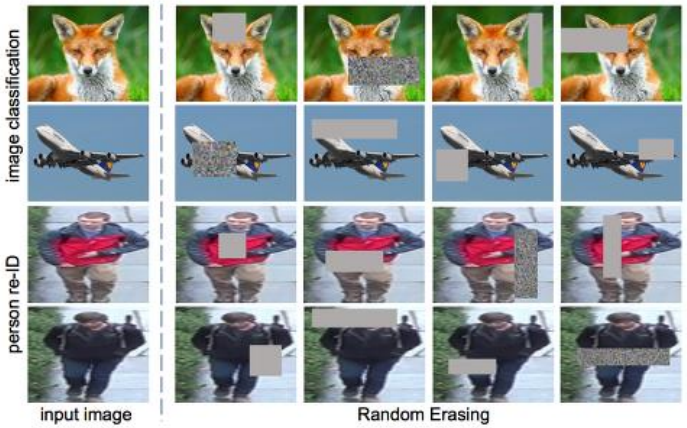
Can't utilize full image regions

[1] Improved regularization of convolutional neural networks with cutout, Devries et al, 2017, arXiv
[2] Random erasing data augmentation, Zhong et al, 2017, arXiv

Related Works

Regional Dropout

- Random erasing[2]



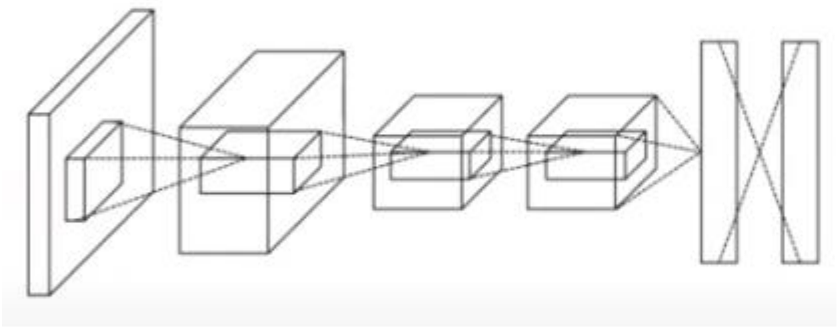
- Cutout[3]

[1] Improved regularization of convolutional neural networks with cutout, Devries et al, 2017, arXiv
[2] Random erasing data augmentation, Zhong et al, 2017, arXiv

Related Works

Regularization: ***Mixup*** [1]

☞ Make model robust to uncertain samples



Panda 50%
Koala 50%

- Good generalization ability
- Use full image region
- Locally unrealistic image

$\tilde{x} = \lambda x_i + (1 - \lambda) x_j$, where x_i, x_j are raw input vectors

$\tilde{y} = \lambda y_i + (1 - \lambda) y_j$, where y_i, y_j are one-hot label encodings

[1] Mixup: Beyond empirical risk minimization, Zhang et al, 2018, ICLR

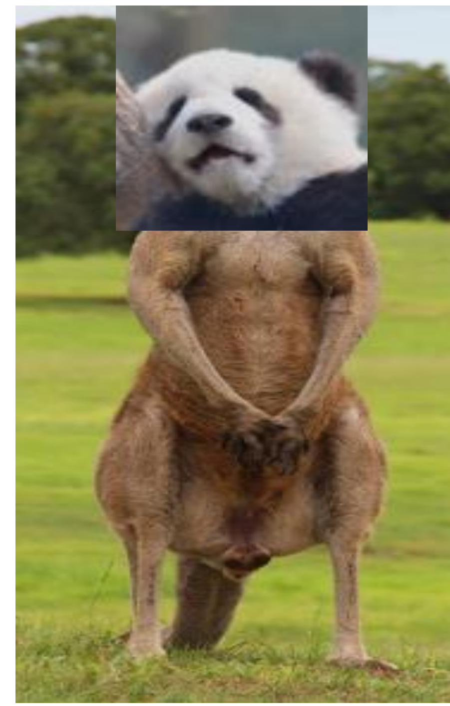
CutMix

Patch

Paste

Cut

CutMix



Kangaroo

Panda

What is this?

Pangaroo?!

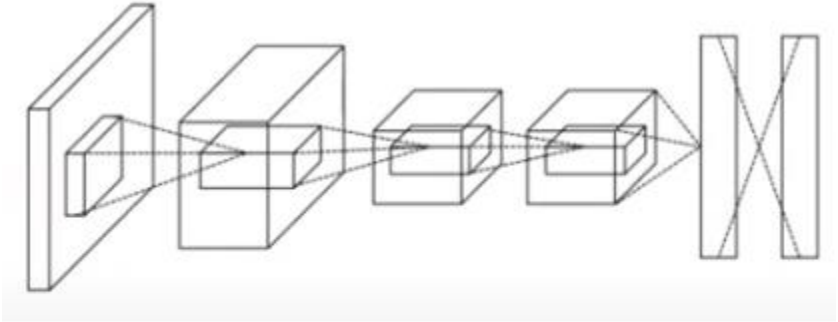
CutMix

Cut

Cat

Paste

Label is decided by the pixel ratio of each image



Cat 50%
Dog 50%

CutMix_Algorithm

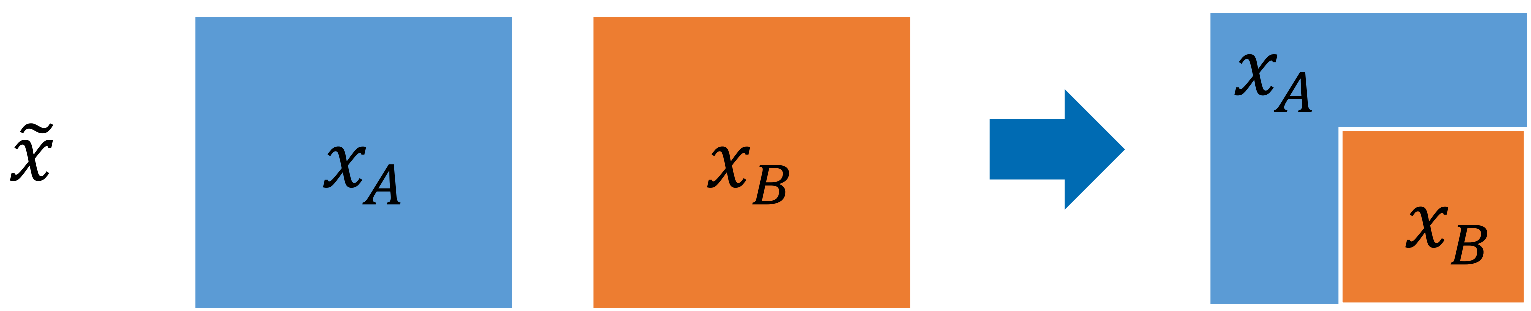
Generate new training sample: $(\tilde{x}, \tilde{y})$
Combining two training samples: (x_A, x_B) and (y_A, y_B)

Binary mask indication where to drop out and fill in from two images

$$\tilde{x} = M \odot x_A + (1-\lambda) \odot x_B \qquad M \in \{0,1\}^{W \times H}$$

$$\tilde{y} = \lambda y_A + (1-\lambda) y_B$$

element-wise multiplication

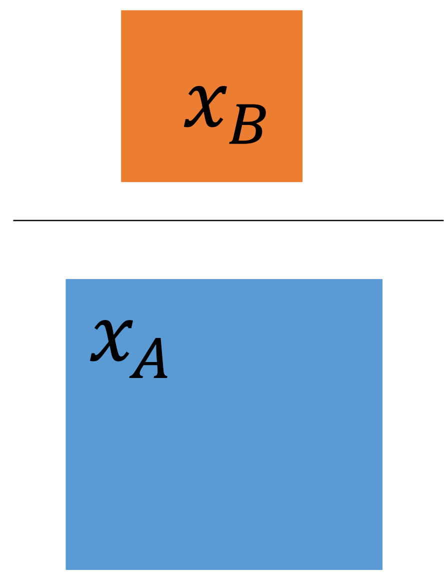


CutMix_Algorithm

Generate new training sample: $(\tilde{x}, \tilde{y})$
Combining two training samples: $(x_{A,}\, x_B)$ and $(y_{A,}\, y_B)$

$$\tilde{x} = M \odot x_A + (1 - \lambda) \odot x_B \qquad \mathbf{M} \in \{0,1\}^{W \times H}$$

$$\tilde{y} = \lambda y_A + \boxed{(1 - \lambda)} y_B$$



CutMix_Algorithm

λ : combination ratio

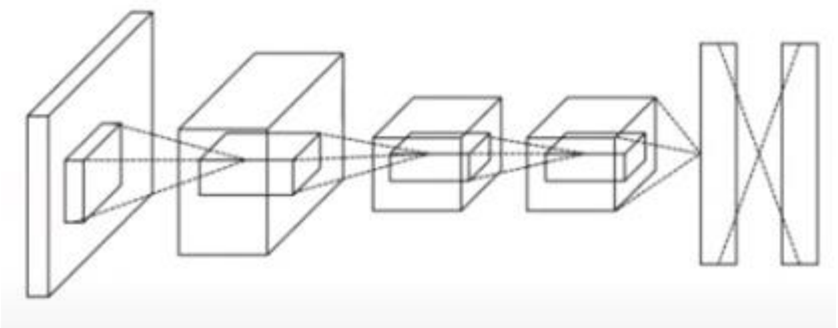
It is sampling from the beta distribution $\mathrm{Beta}(\alpha, \alpha)$.

To sample the binary mask M,

sample the bounding box coordinates $\mathbf{B} = (r_x, r_y, r_w, r_h)$

$$r_x \sim \text{Unif}\ (0, W),\quad r_w = W\sqrt{1-\lambda},$$
$$r_y \sim \text{Unif}\ (0, H),\quad r_h = H\sqrt{1-\lambda}$$

CutMix



Cat 40%
Dog 60%

Image

Target Label

☞ Make model robust to both occlusion and uncertain samples

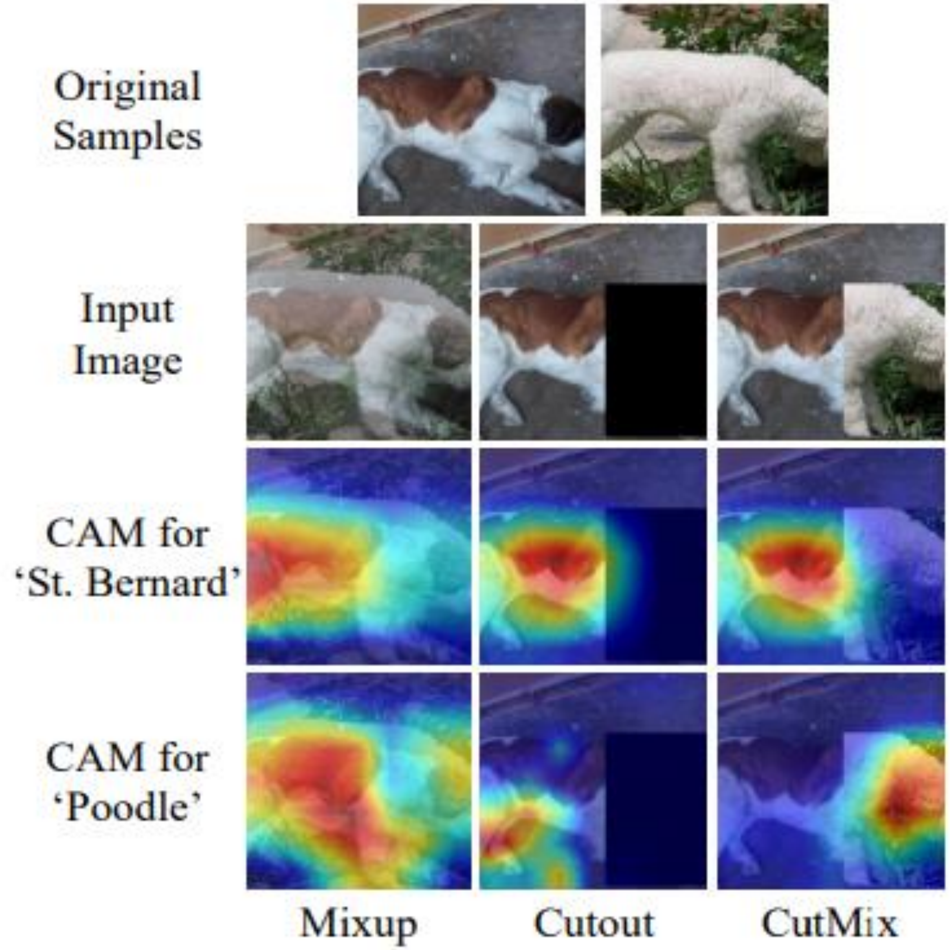
CutMix

	Original	Mixup	Cutout	CutMix
Image				
Target Label	Dog 1.0	Dog 0.5 Cat 0.5	Dog 1.0	Dog 0.6 Cat 0.4

- Unlike Cutout, CutMix uses full image region.
- Unlike Mixup, CutMix makes realistic local image patches.

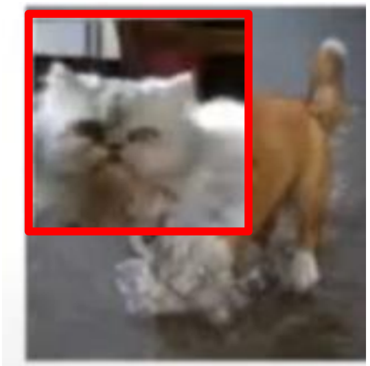
CutMix

Class Activation Map(CAM)

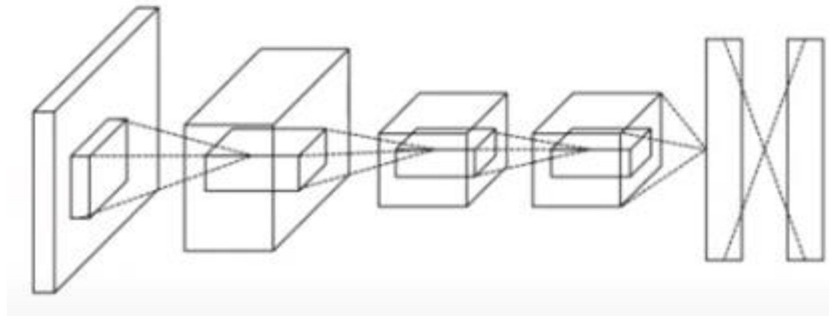


	Mixup	Cutout	CutMix
Usage of full image region	✔	✘	✔
Regional dropout	✘	✔	✔
Mixed image & label	✔	✘	✔

CutMix



Image



Cat 40%
Dog 60%

☞ Finding "**what**", "**where**" and "**how large**" the objects are in the image

Cat	**Upper-left**	**40%**
Dog	**Remaining region**	**60%**

Experiments

- ImageNet classification

Model	# Params	Top-1 Err (%)	Top-5 Err (%)
ResNet-50 (Baseline)	25.6 M	23.68	7.05
ResNet-50 + Cutout [3]	25.6 M	22.93	6.66
ResNet-50 + StochDepth [17]	25.6 M	22.46	6.27
ResNet-50 + Mixup [47]	25.6 M	22.58	6.40
ResNet-50 + Manifold Mixup [41]	25.6 M	22.50	6.21
ResNet-50 + DropBlock* [8]	25.6 M	21.87	5.98
ResNet-50 + Feature CutMix	25.6 M	21.80	6.06
ResNet-50 + CutMix	25.6 M	**21.40**	**5.92**

Model	# Params	Top-1 Err (%)	Top-5 Err (%)
ResNet-101 (Baseline) [12]	44.6 M	21.87	6.29
ResNet-101 + Cutout [3]	44.6 M	20.72	5.51
ResNet-101 + Mixup [47]	44.6 M	20.52	5.28
ResNet-101 + CutMix	44.6 M	**20.17**	**5.24**
ResNeXt-101 (Baseline) [44]	44.1 M	21.18	5.57
ResNeXt-101 + CutMix	44.1 M	**19.47**	**5.03**

😆 Great improvement over baseline (+2%p)

😆 Outperforming existing methods

Experiments

PyramidNet-200 ($\tilde{\alpha}$=240) (# params: 26.8 M)	Top-1 Err (%)	Top-5 Err (%)
Baseline	16.45	3.69
+ StochDepth [17]	15.86	3.33
+ Label smoothing (ϵ=0.1) [37]	16.73	3.37
+ Cutout [3]	16.53	3.65
+ Cutout + Label smoothing (ϵ=0.1)	15.61	3.88
+ DropBlock [8]	15.73	3.26
+ DropBlock + Label smoothing (ϵ=0.1)	15.16	3.86
+ Mixup (α=0.5) [47]	15.78	4.04
+ Mixup (α=1.0) [47]	15.63	3.99
+ Manifold Mixup (α=1.0) [41]	16.14	4.07
+ Cutout + Mixup (α=1.0)	15.46	3.42
+ Cutout + Manifold Mixup (α=1.0)	15.09	3.35
+ ShakeDrop [45]	15.08	2.72
+ CutMix	14.47	2.97
+ CutMix + ShakeDrop [45]	**13.81**	**2.29**

Table 5: Comparison of state-of-the-art regularization methods on CIFAR-100.

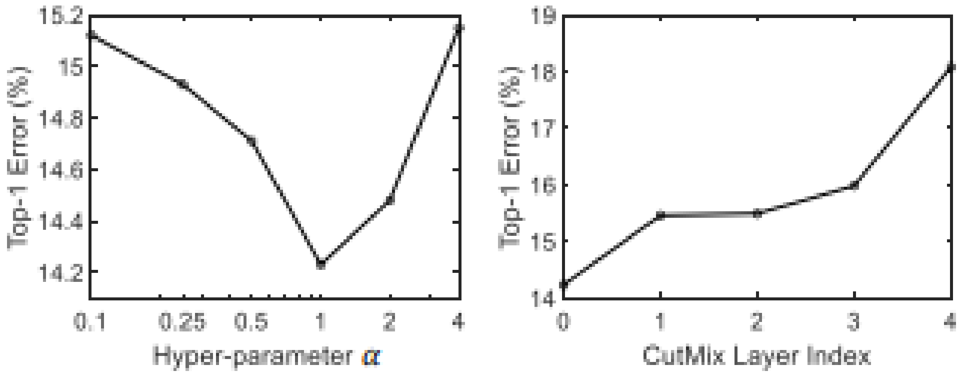


Figure 3: Impact of α and CutMix layer depth on CIFAR-100 top-1 error.

PyramidNet-200 ($\tilde{\alpha}$=240)	Top-1 Error (%)
Baseline	3.85
+ Cutout	3.10
+ Mixup (α=1.0)	3.09
+ Manifold Mixup (α=1.0)	3.15
+ CutMix	**2.88**

Table 7: Impact of CutMix on CIFAR-10.

Experiments

- Transfer learning to object detection and image captioning

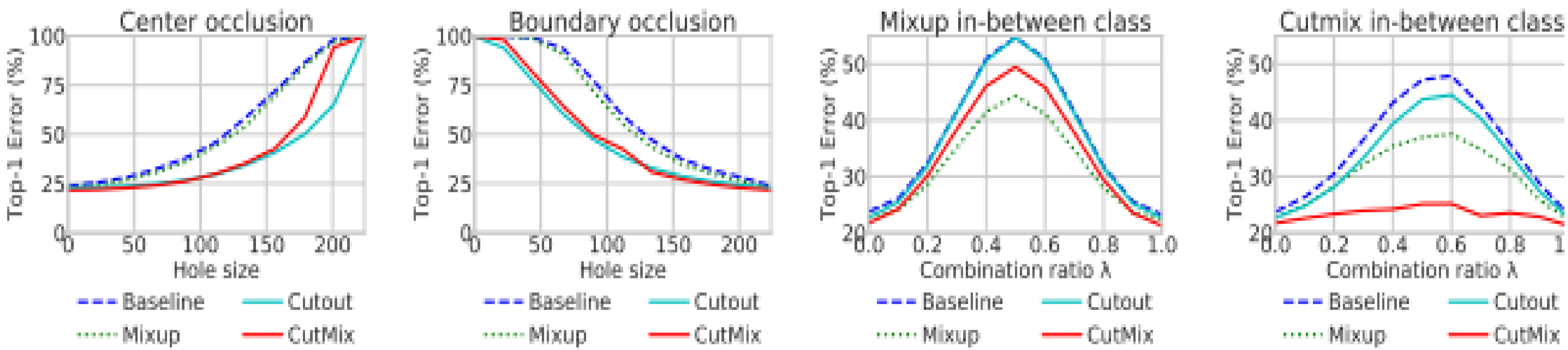
Backbone Network	ImageNet Cls Top-1 Error (%)	Detection		Image Captioning	
		SSD [23] (mAP)	Faster-RCNN [29] (mAP)	NIC [42] (BLEU-1)	NIC [42] (BLEU-4)
ResNet-50 (Baseline)	23.68	76.7 (+0.0)	75.6 (+0.0)	61.4 (+0.0)	22.9 (+0.0)
Mixup-trained	22.58	76.6 (-0.1)	73.9 (-1.7)	61.6 (+0.2)	23.2 (+0.3)
Cutout-trained	22.93	76.8 (+0.1)	75.0 (-0.6)	63.0 (+1.6)	24.0 (+1.1)
CutMix-trained	21.40	**77.6 (+0.9)**	**76.7 (+1.1)**	**64.2 (+2.8)**	**24.9 (+2.0)**

😆 +2%p improvements on MS-COCO

😆 CutMix-pretrained model brings great performance improvement

Experiments

- Robustness



(a) Analysis for occluded samples

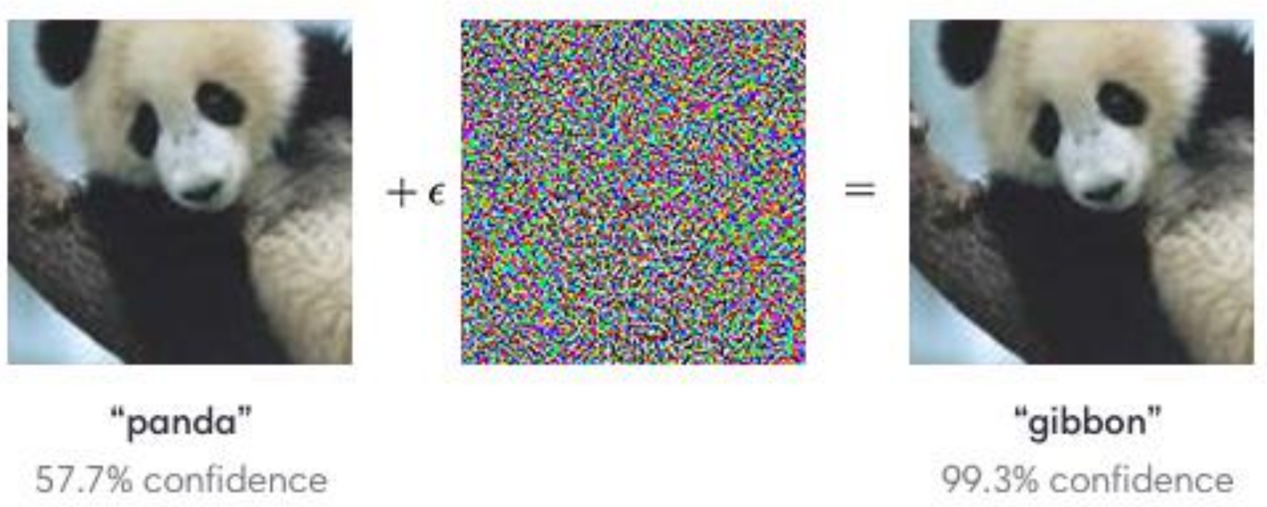
(b) Analysis for in-between class samples

	Baseline	Mixup	Cutout	CutMix
Top-1 Acc (%)	8.2	24.4	11.5	**31.0**

Table 11: Top-1 accuracy after FGSM white-box attack on ImageNet validation set.

Experiments

- Adversarial attacks
  - adversarial examples are perturbed inputs designed to fool machine learning models



Experiments

- Uncertainty

Method	TNR at TPR 95%	AUROC	Detection Acc.
Baseline	26.3 (+0)	87.3 (+0)	82.0 (+0)
Mixup	11.8 (-14.5)	49.3 (-38.0)	60.9 (-21.0)
Cutout	18.8 (-7.5)	68.7 (-18.6)	71.3 (-10.7)
CutMix	**69.0** (+42.7)	**94.4** (+7.1)	**89.1** (+7.1)

Table 12: Out-of-distribution (OOD) detection results with CIFAR-100 trained models. Results are averaged on seven datasets. All numbers are in percents; higher is better.

Conclusions

- Easy to use and has no computational overhead, while being surprisingly effective on various tasks

- Strong classification and localization ability
  - Image classification
  - Weakly supervised object localization
  - Transfer learning of pre-trained model
  - Robustness and Uncertainty