Random Erasing Data Augmentation

Abstract

rectangle region in an image and erases its pixels with random values
various levels of occlusion are generated, which reduces the risk of over-fitting and makes the model robust to occlusion
parameter learning free, easy to implement, and can be integrated with most of the CNN-based recognition models

Introduction

The ability to generalize is a research focus
having too many parameters compared to the number of training samples
over-fitting might happen
weaken its generalization ability

Occlusion
influencing factor on the generalization ability of CNNs
may fail to recognize objects which are partially occluded

random flipping and random cropping, also work
In comparison with Random Erasing, random flipping does not incur information loss during augmentation.

Different from random cropping, in Random Erasing,
1) only part of the object is occluded and the overall object structure is preserved
2) pixels of the erased region are re-assigned with random values, which can be viewed as adding block noise to the image

performing Dropout on the image level
difference
1) we operate on a continuous rectangular region
2) no pixels (units) are discarded
3) we focus on making the model more robust to noise and occlusion.

A-Fast-RCNN
adversarial network that generates examples with occlusion
difference
no parameter learning
can be easily applied to other CNN-based recognition tasks

advantages
- A lightweight method that does not require any extra parameter learning or memory consumption. It can be integrated with various CNN models without changing the learning strategy.
- A complementary method to existing data augmentation and regularization approaches. When combined, Random Erasing further improves the recognition performance.
- Consistently improving the performance of recent state-of-the-art deep models on image classification, object detection, and person re-ID.
- Improving the robustness of CNNs to partially occluded samples. When we randomly adding occlusion to the CIFAR-10 testing dataset, Random Erasing significantly outperforms the baseline model.

Related Work

Dropout
DropConnect
Stochastic Pooling

DisturbLabel
PatchShuffle

two most popular and effective data augmentation methods in training of deep CNN are random flipping and random cropping
For random cropping, it may crop off the corners of the object, while Random Erasing may occlude some parts of the object.

Fast-RCNN

computationally free and does not require any extra parameters learning

Datasets

Classification
CIFAR-10 and CIFAR-100 [13], and a new dataset Fashion-MNIST

Object Detection
PASCAL VOC 2007

Person re-identification
Market-1501
DukeMTMC-reID
CUHK03

Our Approach

detailed procedure of Random Erasing
implementation of Random Erasing in different tasks
analyze the differences between Random Erasing and random cropping

Random Erasing

Algorithm 1: Random Erasing Procedure

Input : Input image I;
Image size W and H;
Area of image S;
Erasing probability p;
Erasing area ratio range s_l and s_h;
Erasing aspect ratio range r_1 and r_2.
Output: Erased image I^*.
Initialization: $p_1 \leftarrow$ Rand (0, 1).
if $p_1 \geq p$ **then**
    $I^* \leftarrow I$;
    return I^*.
else
    while *True* **do**
        $S_e \leftarrow$ Rand $(s_l, s_h) \times S$;

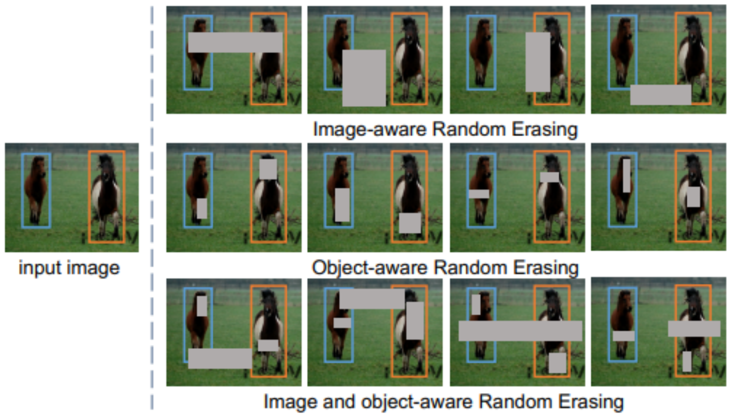
$r_e \leftarrow$ Rand (r_1, r_2);
$H_e \leftarrow \sqrt{S_e \times r_e}$, $W_e \leftarrow \sqrt{\frac{S_e}{r_e}}$;
$x_e \leftarrow$ Rand $(0, W)$, $y_e \leftarrow$ Rand $(0, H)$;
if $x_e + W_e \leq W$ *and* $y_e + H_e \leq H$ **then**
    $I_e \leftarrow (x_e, y_e, x_e + W_e, y_e + H_e)$;
    $I(I_e) \leftarrow$ Rand (0, 255);
    $I^* \leftarrow I$;
    return I^*.
  end
 end
end

Random Erasing for Image Classification and Person Re-identification

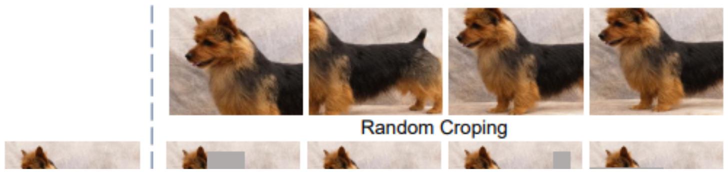
perform Random Erasing on the whole image

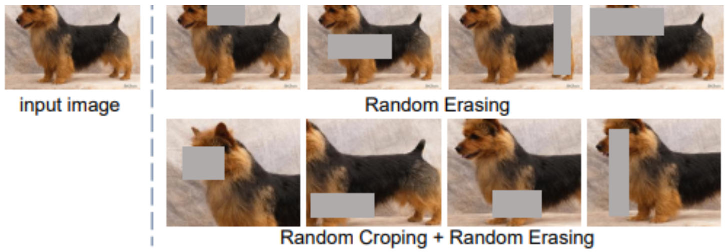
Random Erasing for Object Detection

Since the location of each object in the training image is known, we implement Random Erasing with three schemes



Comparison with Random Cropping





Experiment

Image Classification

Experiment Settings

Four architectures

ResNet [8], pre-activation ResNet [9], ResNeXt [31], and Wide Residual Networks

CIFAR-10, CIFAR-100 and Fashion-MNIST

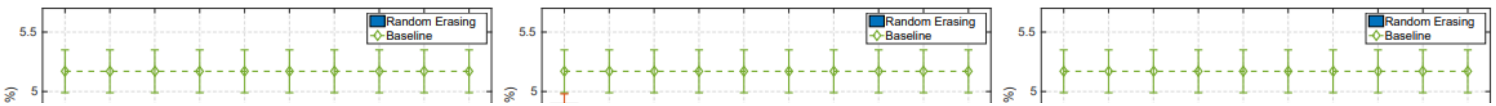
Classification Evaluation

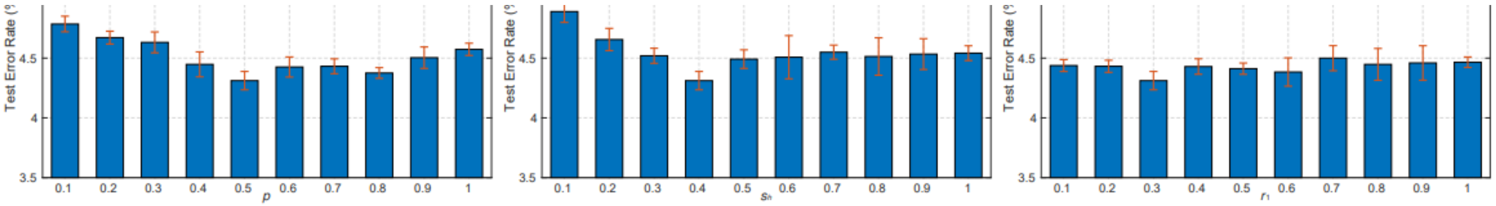
Classification accuracy on different datasets

Model	CIFAR-10		CIFAR-100		Fashion-MNIST	
	Baseline	Random Erasing	Baseline	Random Erasing	Baseline	Random Erasing
ResNet-20	7.21 ± 0.17	6.73 ± 0.09	30.84 ± 0.19	29.97 ± 0.11	4.39 ± 0.08	4.02 ± 0.07
ResNet-32	6.41 ± 0.06	5.66 ± 0.10	28.50 ± 0.37	27.18 ± 0.32	4.16 ± 0.13	3.80 ± 0.05
ResNet-44	5.53 ± 0.08	5.13 ± 0.09	25.27 ± 0.21	24.29 ± 0.16	4.41 ± 0.09	4.01 ± 0.14
ResNet-56	5.31 ± 0.07	4.89 ± 0.07	24.82 ± 0.27	23.69 ± 0.33	4.39 ± 0.10	4.13 ± 0.42
ResNet-110	5.10 ± 0.07	4.61 ± 0.06	23.73 ± 0.37	22.10 ± 0.41	4.40 ± 0.10	4.01 ± 0.13
ResNet-20-PreAct	7.36 ± 0.11	6.78 ± 0.06	30.58 ± 0.16	30.18 ± 0.13	4.43 ± 0.19	4.02 ± 0.09
ResNet-32-PreAct	6.42 ± 0.11	5.79 ± 0.10	29.04 ± 0.25	27.82 ± 0.28	4.36 ± 0.02	4.00 ± 0.05
ResNet-44-PreAct	5.54 ± 0.16	5.09 ± 0.10	25.22 ± 0.19	24.10 ± 0.26	4.92 ± 0.30	4.23 ± 0.15
ResNet-56-PreAct	5.28 ± 0.12	4.84 ± 0.09	24.14 ± 0.25	22.93 ± 0.27	4.55 ± 0.30	3.99 ± 0.08
ResNet-110-PreAct	4.80 ± 0.09	4.47 ± 0.11	22.11 ± 0.20	20.99 ± 0.11	5.11 ± 0.55	4.19 ± 0.15
ResNet-18-PreAct	5.17 ± 0.18	4.31 ± 0.07	24.50 ± 0.29	24.03 ± 0.19	4.31 ± 0.06	3.90 ± 0.06
WRN-28-10	3.80 ± 0.07	$\mathbf{3.08 \pm 0.05}$	$\mathbf{18.49 \pm 0.11}$	$\mathbf{17.73 \pm 0.15}$	$\mathbf{4.01 \pm 0.10}$	$\mathbf{3.65 \pm 0.03}$
ResNeXt-8-64	$\mathbf{3.54 \pm 0.04}$	3.24 ± 0.03	19.27 ± 0.30	18.84 ± 0.18	4.02 ± 0.05	3.79 ± 0.06

Table 1. Test errors (%) with different architectures on CIFAR-10, CIFAR-100 and Fashion-MNIST.

The impact of hyper-parameters





(a) probability p (b) area ratio s_h (c) aspect ratio r_1

Four types of random values for erasing

Types of Erasing Value	Baseline	RE-R	RE-M	RE-0	RE-255
Test error rate (%)	5.17 ± 0.18	$\mathbf{4.31 \pm 0.07}$	4.35 ± 0.12	4.62 ± 0.09	4.85 ± 0.13

Table 2. Test errors (%) on CIFAR-10 based on ResNet18 (pre-act) with four types of erasing value. **Baseline:** Baseline model, **RE-R:** Random Erasing model with random value, **RE-M:** Random Erasing model with mean value of ImageNet 2012, **RE-0:** Random Erasing model with 0, **RE-255:** Random Erasing model with 255.

1) all erasing schemes outperform the baseline

2) RE-R achieves approximately equal performance to RE-M

3) both RE-R and RE-M are superior to RE-0 and RE-255

Comparison with Dropout and random noise

Method	Test error (%)	Method	Test error (%)
Baseline	5.17 ± 0.18	Baseline	5.17 ± 0.18
Random Erasing	$\mathbf{4.31 \pm 0.07}$	Random Erasing	$\mathbf{4.31 \pm 0.07}$
Dropout	Test error (%)	Random Noise	Test error (%)
$\lambda_1 = 0.001$	5.37 ± 0.12	$\lambda_2 = 0.01$	5.38 ± 0.07
$\lambda_1 = 0.005$	5.48 ± 0.15	$\lambda_2 = 0.05$	5.79 ± 0.14
$\lambda_1 = 0.01$	5.89 ± 0.14	$\lambda_2 = 0.1$	6.13 ± 0.12
$\lambda_1 = 0.05$	6.23 ± 0.11	$\lambda_2 = 0.2$	6.25 ± 0.09
$\lambda_1 = 0.1$	6.38 ± 0.18	$\lambda_2 = 0.4$	6.52 ± 0.12

Table 3. Comparing Random Erasing with dropout and random noise on CIFAR-10 with using ResNet18 (pre-act).

Comparing with data augmentation methods

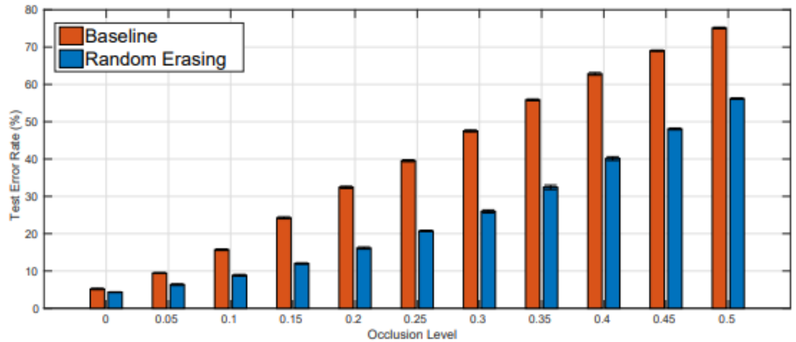
Method	RF	RC	RE	Test errors (%)
				11.31 ± 0.18

Baseline				11.31 ± 0.18
	✓			8.30 ± 0.17
		✓		6.33 ± 0.15
			✓	10.13 ± 0.14
	✓	✓		5.17 ± 0.18
	✓		✓	7.19 ± 0.10
		✓	✓	5.21 ± 0.14
	✓	✓	✓	**4.31** ± **0.07**

Table 4. Test errors (%) with different data augmentation methods on CIFAR-10 based on ResNet18 (pre-act). **RF**: Random flipping, **RC**: Random cropping, **RE**: Random Erasing.

Random Erasing and the two competing techniques are complementary

Robustness to occlusion



Object Detection

Experiment Settings

Fast-RCNN

initialized - ImageNet

fine-tuned

VGG16

A-Fast-RCNN

Detection Evaluation

Ours (I+ORE)	07	**71.5**	76.1	81.6	69.5	60.1	45.6	82.2	79.2	84.5	52.5	78.7	71.6	80.4	83.3	76.7	73.9	39.4	68.9	69.8	79.2	77.4
FRCN [7]	07+12	70.0	77.0	78.1	69.3	59.4	38.3	81.6	78.6	86.7	42.8	78.8	68.9	84.7	82.0	76.6	69.9	31.8	70.1	74.8	80.4	70.4
FRCN⋆ [27]	07+12	74.8	78.5	81.0	74.7	67.9	53.4	85.6	84.4	86.2	57.4	80.1	72.2	85.2	84.2	77.6	76.1	45.3	75.7	72.3	81.8	77.3
Ours (IRE)	07+12	75.6	79.0	84.1	76.3	66.9	52.7	84.5	84.4	88.7	58.0	82.9	71.1	84.8	84.4	78.6	76.7	45.5	77.1	76.3	82.5	76.8
Ours (ORE)	07+12	75.8	79.4	81.6	75.6	66.5	52.7	85.5	84.7	88.3	58.7	82.9	72.8	85.0	84.3	79.3	76.3	46.3	76.3	74.9	86.0	78.2

Ours (I+ORE)	07+12	**76.2**	79.6	82.5	75.7	70.5	55.1	85.2	84.4	88.4	58.6	82.6	73.9	84.2	84.7	78.8	76.3	46.7	77.9	75.9	83.3	79.3

Table 5. **VOC 2007 test** detection average precision (%). FRCN⋆ refers to FRCN with training schedule in [27].

Person Re-identification

Experiment Settings

the IDdiscriminative Embedding (IDE) [39], TriNet [10], and SVDNet

Person Re-identification Performance

Random Erasing improves different baseline models

reduce the risk of over-fitting and improves the re-ID performance

Comparison with the state-of-the-art methods

Method	Rank-1	mAP
BOW [38]	34.40	14.09
LOMO+XQDA [18]	43.79	22.22
DNS [36]	61.02	35.68
Gated [25]	65.88	39.55
IDE [39]	72.54	46.00
MSCAN [15]	80.31	57.53
DF [37]	81.0	63.4
SSM [3]	82.21	68.80
SVDNet [24]	82.3	62.1
GAN [40]	83.97	66.07
PDF [23]	84.14	63.41
TriNet [10]	84.92	69.14
DJL [17]	85.1	65.5
SVDNet+Ours	87.08	71.31
SVDNet+Ours+re [41]	89.13	83.93

Table 7. Comparison of our method with state-of-the-art methods on the Market-1501 dataset. We use ResNet-50 as backbone. The best, second and third highest results are in red, blue and green, respectively.

Method	Labeled		Detected	
	Rank-1	mAP	Rank-1	mAP
BOW+XQDA [38]	7.93	7.29	6.36	6.39
LOMO+XQDA [18]	14.8	13.6	12.8	11.5
IDE [39]	22.2	21.0	21.3	19.7
IDE+DaF [32]	27.5	31.5	26.4	30.0
SVDNet [24]	40.9	37.8	41.5	37.2
DPFL [4]	43.0	40.5	40.7	37.0

Method	Rank-1	mAP
BOW+kissme [38]	25.13	12.17
LOMO+XQDA [18]	30.75	17.04
IDE [39]	65.22	44.99
GAN [40]	67.68	47.13
OIM [29]	68.1	47.4
TriNet [10]	72.44	53.50
ACRN [20]	72.58	51.96
SVDNet [24]	76.7	56.8
SVDNet+Ours	**79.31**	**62.44**
SVDNet+Ours+re [41]	**84.02**	**78.28**

Table 8. Comparison of our method with state-of-the-art methods on the DukeMTMC-reID dataset. We use ResNet-50 as backbone.

DPFL [4]	43.0	40.5	40.7	37.0
TriNet [10]	49.86	46.74	50.50	46.47
TriNet+Ours	**58.14**	**53.83**	**55.50**	**50.74**
TriNet+Ours+re [41]	**63.93**	**65.05**	**64.43**	**64.75**

Table 9. Comparison of our method with state-of-the-art methods on the CUHK03 dataset using the new evaluation protocol in [41]. We use ResNet-50 as backbone.

Conclusion

It is easy to implemented

randomly occludes an arbitrary region of the input image during each training iteration

future work, we will apply our approach to other CNN recognition tasks, such as, image retrieval and face recognition