#3 CutMix

- Classification, Localization(분류+박스, 단일 인식), Object Detection(박스, 복합 인식) : Task의 차이 있음. → 구분하기

🔎Abstraction

기존 방법에는 regional dropout(트레이닝 이미지의 픽셀 정보를 없애거나 블랙 픽셀 혹은 랜덤 노이즈로 덮어 씌우는 경우)를 사용했다 하지만 이는 정보 손실과 비효율성으로 이어졌음.

→ 따라서 트레이닝 이미지에 대한 *patch*를 자르고 붙이는 방법을 제안함, 이에 대한 ground truth label은 트레이닝 이미지가 mixed(혼합된)만큼 변형

📖 Introduction

CNN이 특정 set이나 특정 이미지의 일부분만 보고 너무 집중하는 (inductive bias)를 막기 위해 여러 방안(Regional Dropout)이 제시되어 왔음

이는 모델이 전체 객체에 대해 집중할 수 있도록 하여 실제로 일반화와 위치화 성능을 올릴 수 있다고 증명되어 왔음.

→ Regional Dropout: 인풋 이미지에 대한 랜덤한 지역을 삭제하는 방식

하지만 삭제된 영역은 보통 zeroed-out되거나 랜덤 노이즈로 채워지는 경우가 많았다

→ 데이터 양에 취약한 CNN에 있어 아쉬운 점, 그래서 이 삭제된 영역을 최대한 일반화와 위치화에 대한 이점으로 활용하고 싶었음

그래서 **CutMix**라는 개념을 제안, 삭제된 지역은 다른 이미지의 패치로 대체하고, 이미지의 ground truth 라벨도 섞인 만큼 비율에 따라 수정한다.

CutMix랑 Mixup이랑 비슷하게 생각 가능, 둘다 두 개의 이미지와 라벨을 보간하는 방법이기 때문, Mixup도 퍼포먼스 올려주긴 하는데, 얘네 샘플은 좀 unnatural, 부자연스러워 보이는 경향이 있음.

→ CutMix는 이러한 문제를 단지 대체하는 방법으로 극복함.

그리고 최근 컴퓨터비전은 방대한 양의 컴퓨팅과 데이터를 필요로 하여 Weight Decay, Dropout, Batch Normalization 등과 같은 것들이 주로 쓰이고, feature map에 노이즈를 가한다 던가, 아키텍처에 모듈을 붙인다던가 시도를 하고 있는데, CutMix는 데이터 레벨에서 진행되기 때문에 이론 것들과 상호 보완적이라고 볼 수 있음.

📄Method

Algorithm

M이라고 하는 인풋 이미지 전체의 Width와 Height에 대한 0,1로 이루어진 Matrix에 대해서, 0,1의 binary masking을 통해서 한 이미지에서 어떤 부분을 버릴지 어떤 부분을 유지할 지를 선택

1이면 유지한다는 거임, 나머지는 (1-M)에 의해서 다른 이미지가 채움, 이때 이미지와 M의 연산은 element-wise로 행렬 곱셈 해준다.

$$\tilde{x} = \mathbf{M} \odot x_A + (\mathbf{1} - \mathbf{M}) \odot x_B$$
$$\tilde{y} = \lambda y_A + (1 - \lambda) y_B,$$

또한, Mixup에서 섞인 데이터의 라벨은 combination ratio인 감마에 의해 결정되는데, 감마는 두 이미지의 결합 비율로 0과 1사이의 값을 갖게끔 Beta Distribution을 통해 정해졌음

- 베타(알파,알파) = Uniform Distribution, 여기서 알파는 하이퍼 파라미터로 1이 아니면 Beta Distribution을 따름
- 베타 분포에서 두 인자가 같으면 평평해지는데 1이면 완전 평평, 같은 값이 높아질 수록 정규분포에 가까워짐

CutMix에서는 샘플링(데이터 섞을)시에 uniform distribution에서 감마값을 결정함.

→ 데이터가 섞이는 비율이 랜덤으로 일정(in mixup), 라벨 성분이 섞이는 비율이 랜덤

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

M을 만들 때에는, 처음으로는 직사각형의 bounding box 좌표를 먼저 만들어야 함. 각 이미지에서 crop될 영역.

→ x축으로 0부터 최대 너비까지, y축으로 0부터 최대 높이까지 모두 동일한 확률로 크롭 영역을 선정 (r_x, r_y) 나머지 두 좌표는 (1-감마) 값에 의해 정해짐

- 바운딩 박스 B가 정해지면 여기는 다 0으로 채워지고 나머지는 다 1로해서 기존 이미지랑 곱하면 되겠다.
- Cutmix된 이미지가 만들어질 때는 한 미니 배치에서 두 개의 훈련 데이터를 랜덤하게 선택하여 만든다. → 무시할만한 컴퓨팅 오버헤드이며 어떤 네트워크 구조에도 효율적

🔬Experiments

Image Classification

CutMix의 하이퍼 파라미터 알파 값에 대한 Ablation

: **1.0일때 가장 베스트였음** 또한 CutMix가 이루어 지는 레벨도 했는데, **인풋 이미지에 수행하는 게 가장 베스트**였음 (0:image, 1:conv-bn, 2:layer2 …)

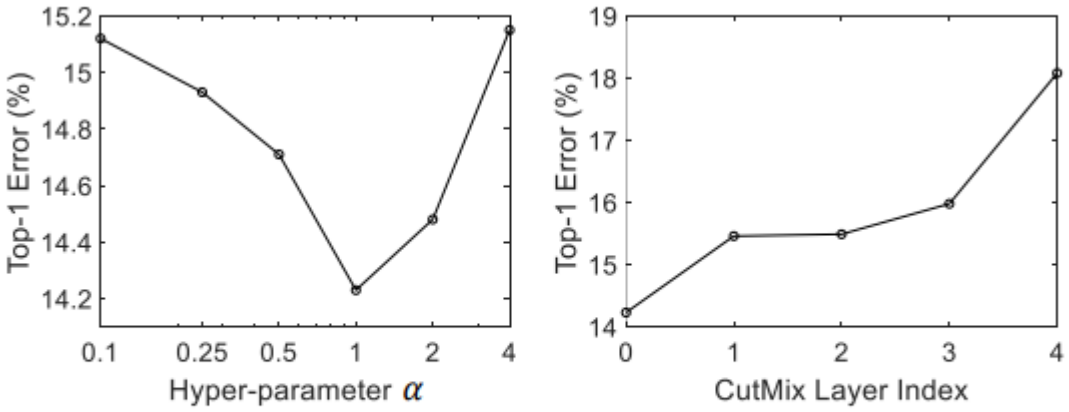


Figure 3: Impact of α and CutMix layer depth on CIFAR-100 top-1 error.

다양한 CutMix 변형 방법

- Center Gaussian Cutmix: r_x, r_y를 이미지 중간의 mean값에 따라 Gaussian Distribution에 따라 선정하는 방식
- Fixed-size CutMix: 크롭할 영역을 16x16, 감마는 0.75로 고정
- Scheduled CutMix: 트레이닝 하는 동안 CutMix의 적용 확률을 선형적으로 늘리는 방법
- One-hot CutMix: 더 많은 양의 이미지를 가진 이미지의 라벨이 새로 Cut-mix된 이미지의 라벨을 결정하는 방식

- Complete-label CutMix: Mix된 이미지의 타겟 라벨을 감마 값에 관계없이 항상 0.5 0.5로 나눠서 결정하는 방식

⇒ Proposed된 방식이 제일 높은 성능을 보임

PyramidNet-200 ($\tilde{\alpha}$=240) (# params: 26.8 M)	Top-1 Error (%)	Top-5 Error (%)
Baseline	16.45	3.69
Proposed (CutMix)	14.47	2.97
Center Gaussian CutMix	15.95	3.40
Fixed-size CutMix	14.97	3.15
One-hot CutMix	15.89	3.32
Scheduled CutMix	14.72	3.17
Complete-label CutMix	15.17	3.10

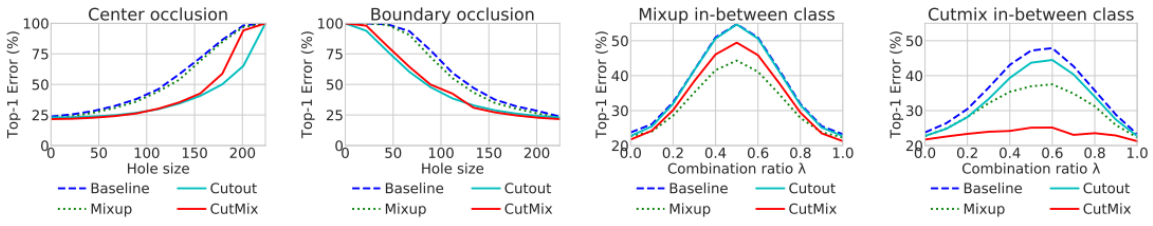
Table 8: Performance of CutMix variants on CIFAR-100.

CIFAR-10, CutMix also enhances the classification performances by +0.97%, outperforming Mixup and Cutout performances.

Model Robustness and model over-confidence

딥러닝 입력 데이터에 어떤 변화를 가했을 때, 혹은 Adversarial한 sample에 대해 학습할 때, 모델의 출력이 크게 바뀌지 않거나 덜 영향을 받게 하는 실험도 진행되었다.

→ 여러 Augmentation 기법을 Robustness 관점에서 평가함



(a) Analysis for occluded samples (b) Analysis for in-between class samples

Figure 5: Robustness experiments on the ImageNet validation set.