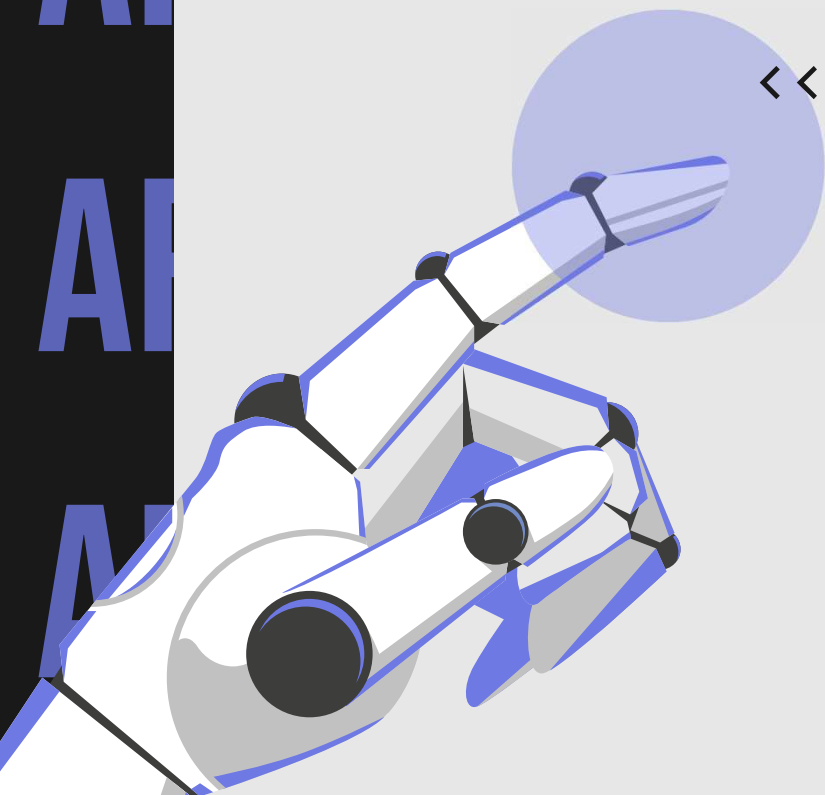


ARTIFICIAL INTELLIGENCE (AI)

ARTIFICIAL INTELLIGENCE (AI)

ARTIFICIAL INTELLIGENCE (AI)

ARTIFICIAL INTELLIGENCE (AI)



<<<<

YOLO V4

학부연구생 2단계 과정 세미나

18011816 김동영

>>>>

TABLE OF CONTENTS

01.

OBJECT DETECTION
MODELS

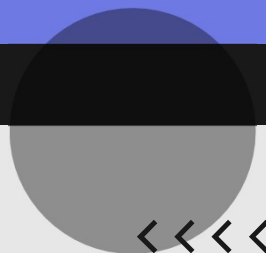
02.

BAG OF FREEBIES

03.

BAG OF SPECIALS

ARTIFICIAL INTELLIGENCE (AI)



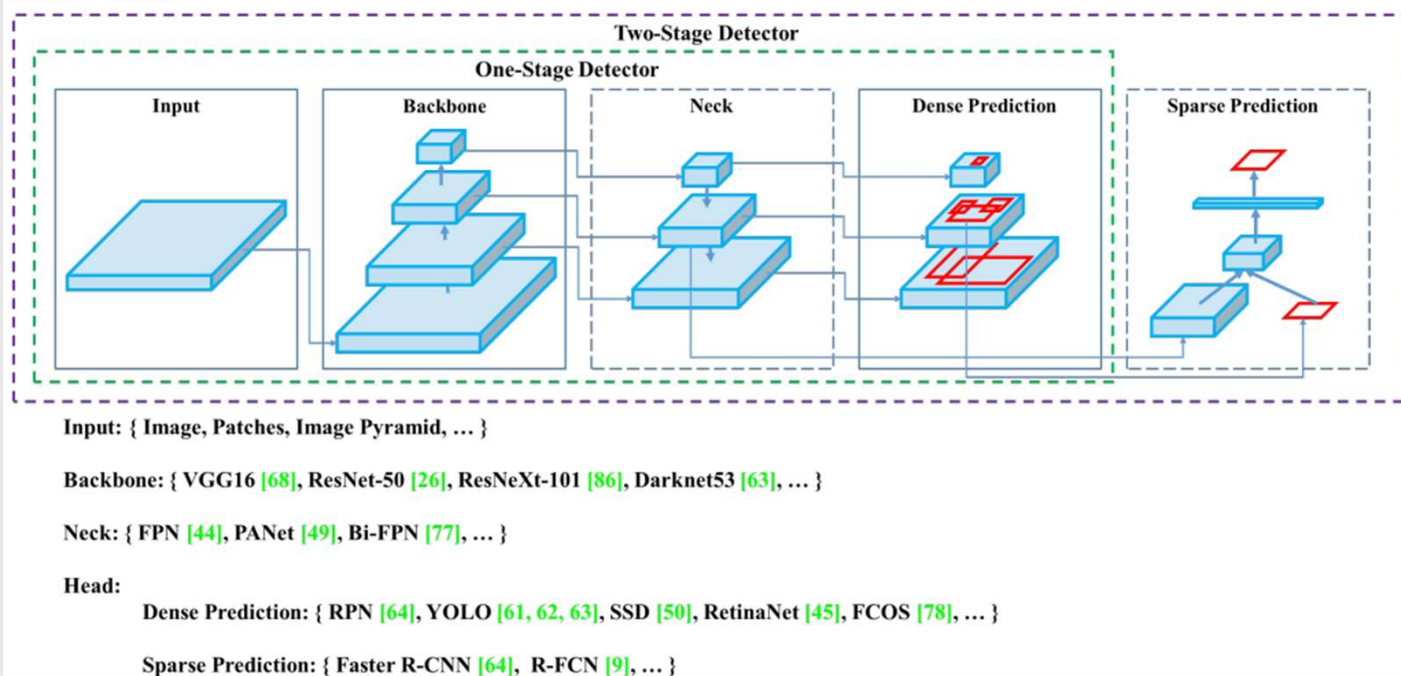
<<<<

01.

OBJECT DETECTION MODELS



OBJECT DETECTION MODELS

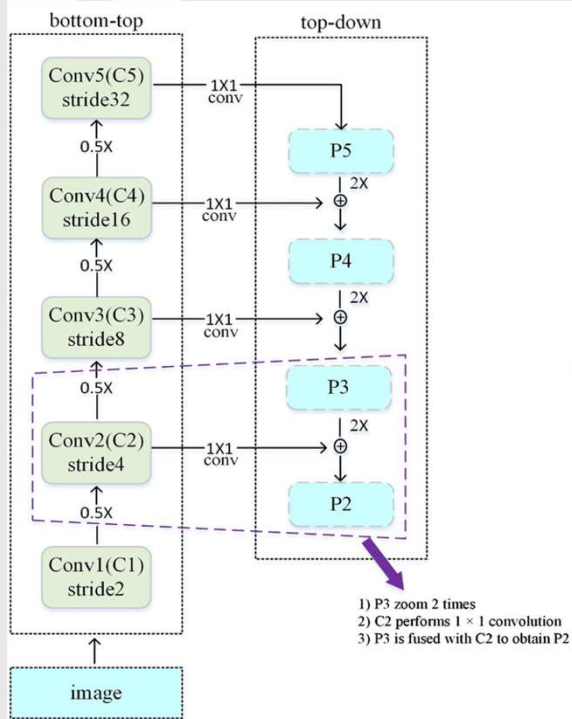


1. Backbone: ImageNet Dataset으로 Pre-trained됨

2. Head : Object에 대한 class와 bboxes를 예측

-One-stage or Two-stage object detectors -> anchor-based or anchor-free 두개로 나뉨

OBJECT DETECTION MODELS



3. Neck : Backbone과 Head 사이에 서로 다른 stages(ex. small,medium,big)들로부터 온 feature maps들을 모으기 위해 Layer를 추가하여 생긴 부분

-보통, 몇몇의 bottom-up paths와 top-down paths들로 구성됨

bottom-up path는 feature를 추출하기 위한 과정이고 top-down path의 경우 feature를 up-sampling함

upper sampling map과 이에 해당하는 bottom-up map이 1x1 conv에서 합쳐 짐

마지막으로 3x3 conv가 다시 붙어 최종map 생성

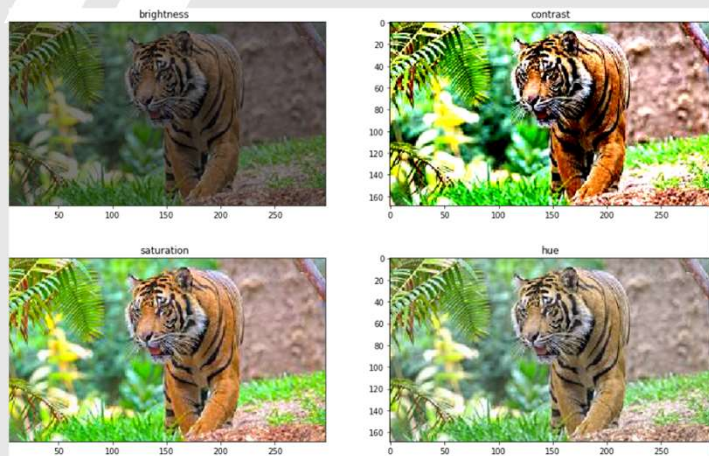


02.

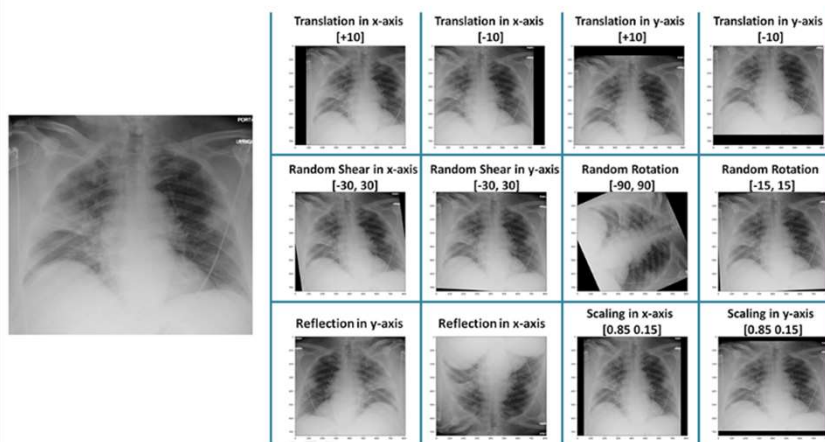
BAG OF FREEBIES

Inference(추론) cost를 높이지 않는 한에서 Training 전략만 바꾸거나 Training에 소요되는 비용만 증가시키는 방법

DATA AUGMENTATION



Geometric Augmentations



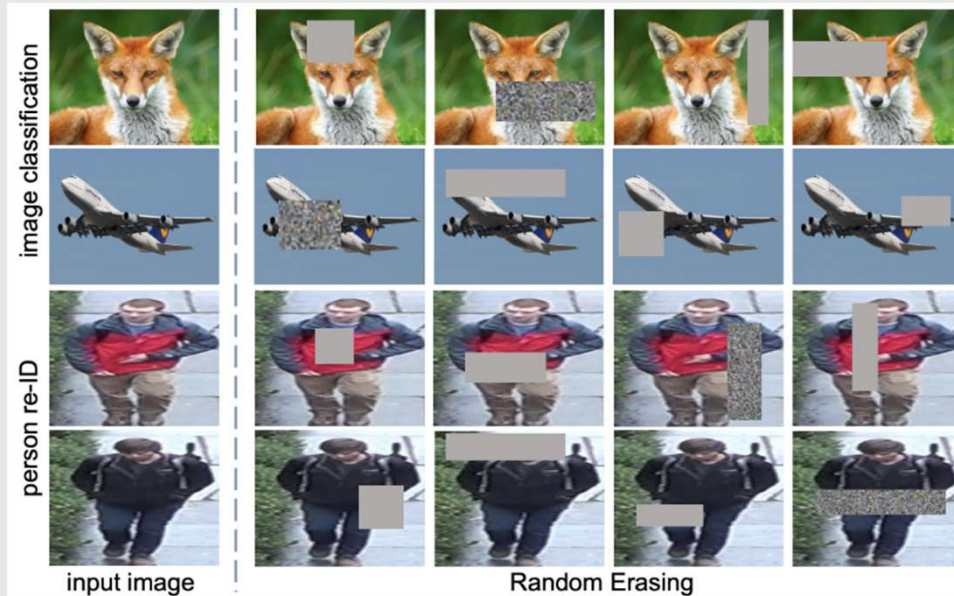
1) 광학적, 기하학적 augmentation

- 광학적 왜곡 : 이미지의 brightness, contrast, hue, saturation, noise 등을 조절

- 기하학적 왜곡: random scaling, cropping, flipping, rotating 등을 추가

- pixel-wise한 조정으로, 조정된 영역 내 원래 pixel 정보는 유지가 됨

DATA AUGMENTATION



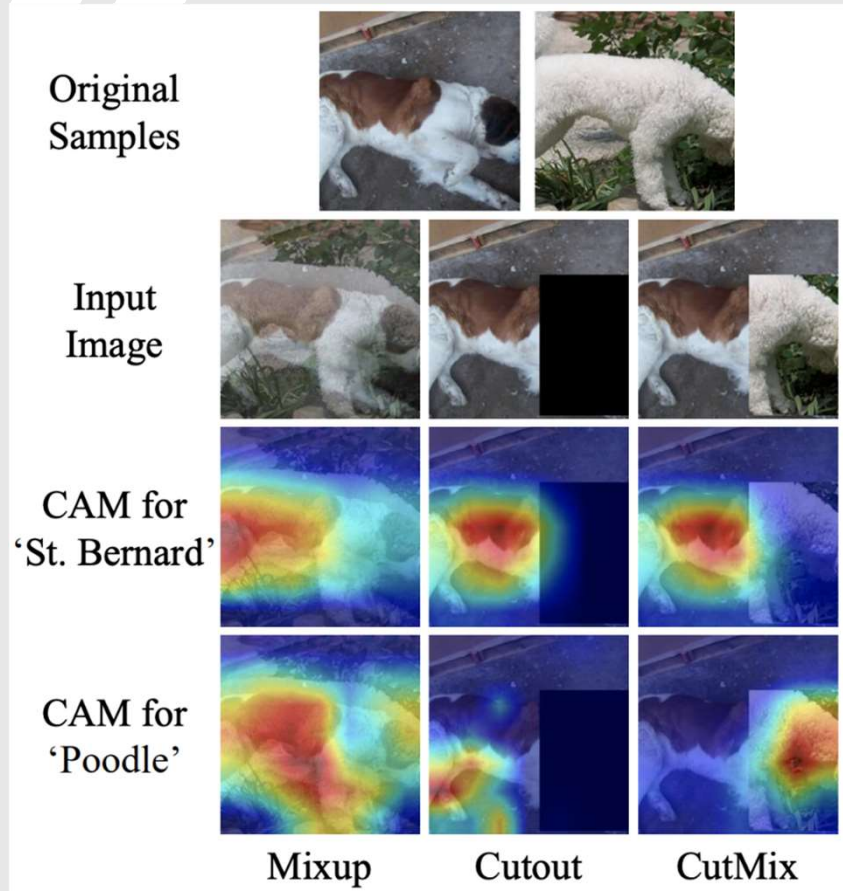
2) Object occlusion을 활용한 방법

- random erase, CutOut, hide-and-seek, grid mask

- 이미지 내 사각형 영역을 random하게 선택하여 random하게 채우거나(random erase), 0의 값으로 보완함(CutOut)

- 이미지 내 여러 사각형 영역을 random하게 또는 고르게 선택하여 모두 0으로 바꿈(hide-and-seek, grid mask)

DATA AUGMENTATION



3) 한 개 이상 이미지를 사용한 방법

- MixUp, CutMix

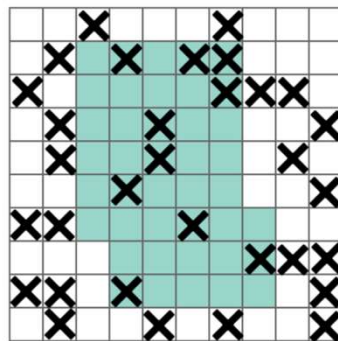
- 두개의 이미지를 사용하여 서로 다른 coefficient ratios를 곱한 후 중첩시킨 다음, 이렇게 중첩된 ratios로 label을 조정함(MixUp)

- 잘라낸 이미지를 다른 이미지의 사각형 영역으로 덮고 mix area의 크기에 따라서 label을 조정함 (CutMix)->실제로 사용

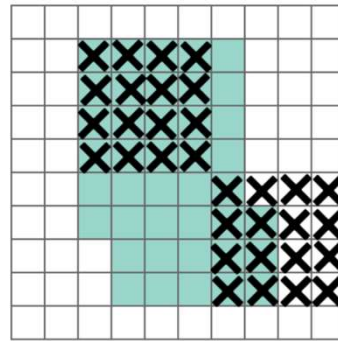
OTHERS



(a)



(b)



(c)

DropBlock (in Regularization)

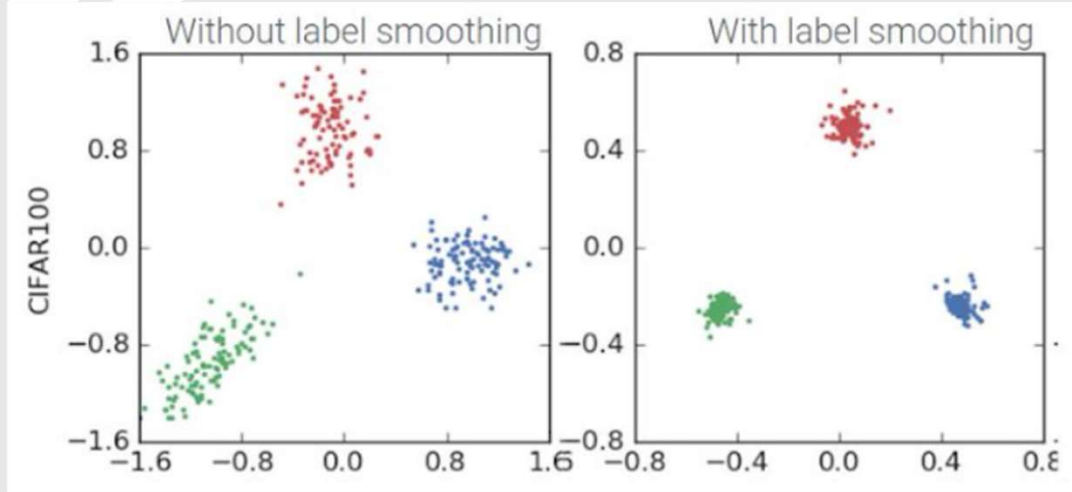
CNN 네트워크를 정규화 하기위한 구조화된 형태의 드롭아웃

(b) 처럼 Random하게 드롭 하는 것은 feature를 가지고 있는 부분(초록색)에 영향을 주므로 좋지 않음 따라서 (c) 처럼 연속적인 부분을 drop하게 되면 강아지의 머리카락 등을 제거 할 수 있으므로 효과적으로 학습이 가능함

(b)의 경우 일반적인 Dropout 방식으로 무작위로 골라 삭제하는 방법 -> Overfitting 방지

DropConnect : Dropout과 비슷한 방식으로 동작하는데 뉴런을 Drop하는 방식과는 달리 DropConnect는 weight 값을 Drop함

OTHERS



Label smoothing (in Regularization)

Label들에 대한 noise들을 줄이기 위해 사용
만약 아주 작은 값 ϵ 에 대해, 어떤 label이 정확할 확률이 $1-\epsilon$ 이고 아닐 확률이 ϵ 일 경우
label smoothing은 softmax를 기반으로 k 개의 output label이 있다면 원래 0과 1로
설정된 target을 $\epsilon/k-1$ 과 $1-\epsilon$ 로 바꿔줌(?) -> Data에 mistake이 있어도 어느정도 감안 가능

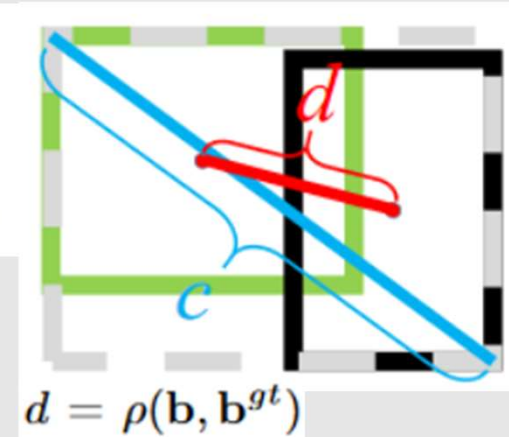
서로 다른 categories 사이의
degree of association(?)을
one-hot hard
representation(?)을 통해
표현하기 어려움

→ 이를 해결하고자 학습할 때
hard label을 soft label로
변환하여 모델을 보다
Robust하게 만드는 label
smoothing을 제안함

OTHERS

$$\mathcal{L}_{CIoU} = 1 - IoU + \underbrace{\frac{\rho^2(\mathbf{b}, \mathbf{b}^{gt})}{c^2}}_{\text{DIoU}} + \alpha v.$$

$$\mathcal{R}_{CIoU} = \frac{\rho^2(\mathbf{b}, \mathbf{b}^{gt})}{c^2} + \alpha v, \quad v = \frac{4}{\pi^2} \left(\arctan \frac{w^{gt}}{h^{gt}} - \arctan \frac{w}{h} \right)^2, \quad \alpha = \frac{v}{(1 - IoU) + v},$$



CIoU Loss (in Objective function)

겹치는 영역, 중심점 사이의 거리, 종횡비 세 가지를 동시에 고려, DIoU의 확장버전

- v 는 두 box의 종횡비의 일치성을 측정하는 역할
- α 는 positive trade-off parameter로 non-overlapping case와 overlapping case의 균형을 조절하는 역할
- DIoU와 αv 가 추가됨 -> 상자 종횡비의 일관성 유지를 위함 (Data unbalance X)

03.

BAG OF SPECIALS

Inference(추정) cost를 조금만 올리면서 눈에 띄는 검출 정확성 변화를 가져다 주는 방법

BAG OF SPECIALS

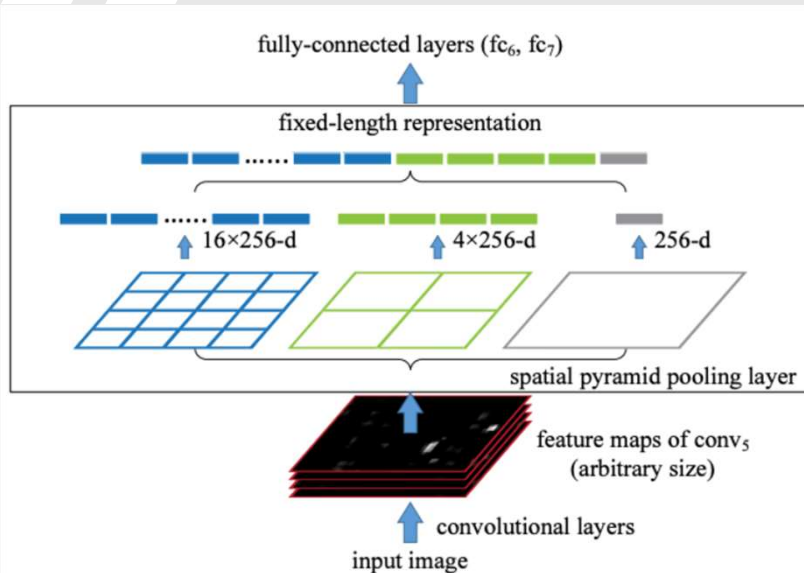


Figure 3: A network structure with a **spatial pyramid pooling layer**. Here 256 is the filter number of the conv₅ layer, and conv₅ is the last convolutional layer.

SPP (Spatial Pyramid Pooling)

Conv Layer 통과 후 feature map을 input으로 받음
미리 정해져 있는 영역으로 나누어 줌
위의 예시에서는 미리 4x4, 2x2, 1x1 세 가지
영역으로 나누어 이를 피라미드라고 부름

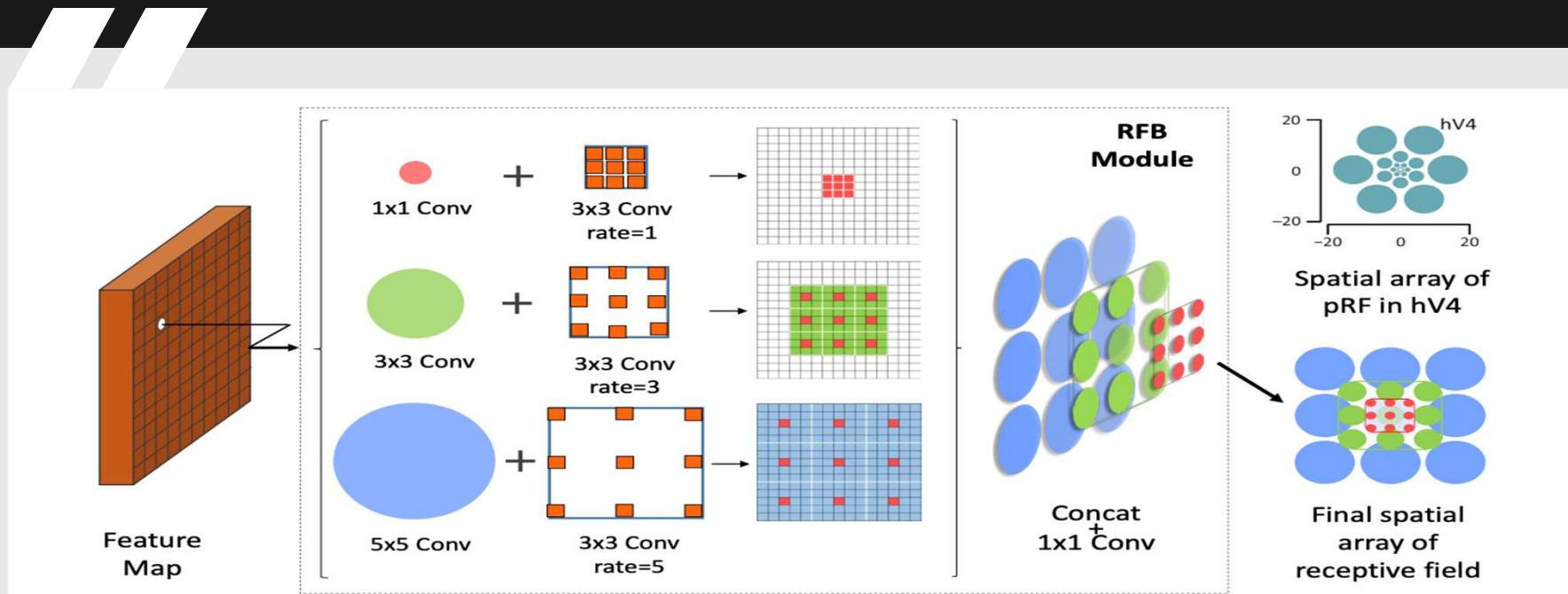
피라미드의 한 칸을 bin 이라고 함

Ex) input이 64 x 64 x 256 크기의
feature map -> 파란색 4x4의 피라미드의 bin의
크기는 16x16이 됨

각 bin에서 max pooling을 통해 가장 큰값을
추출하여 쪽 이어붙여 줌

SPP의 장점은 입력 이미지의 크기와는 상관없이 미리 설정한 bin의 개수와 CNN 채널 값으로
SPP의 출력이 결정되므로, 항상 동일한 크기의 결과가 나옴 -> 고정 크기로 조절할 필요 X

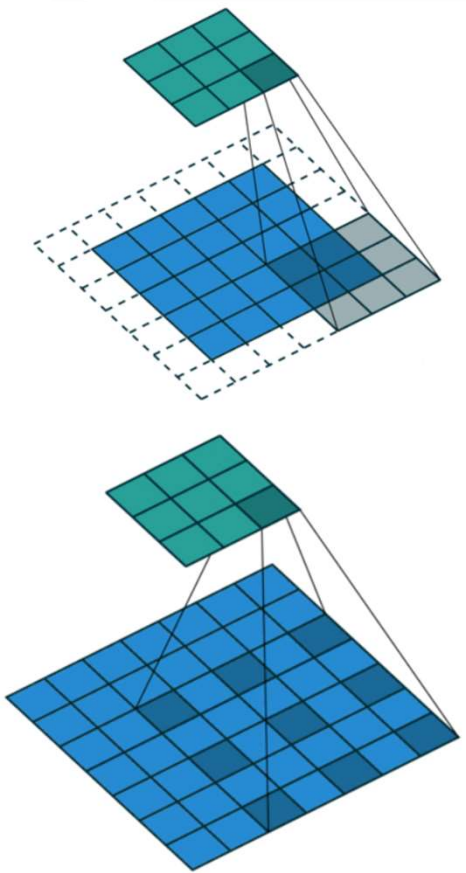
BAG OF SPECIALS



RFB (Receptive Field Block)

lightweight CNN모델에서 학습된 feature를 강화 시켜 빠르고 정확한 detector에 적용시키기 위해 등장. 여러 개의 kernel 사이즈의 conv와 dilated conv를 합하여 적용시킨 후 Concat하여 합치고 이를 1×1 conv를 통과해 feature를 추출함

BAG OF SPECIALS



Dilated Convolution

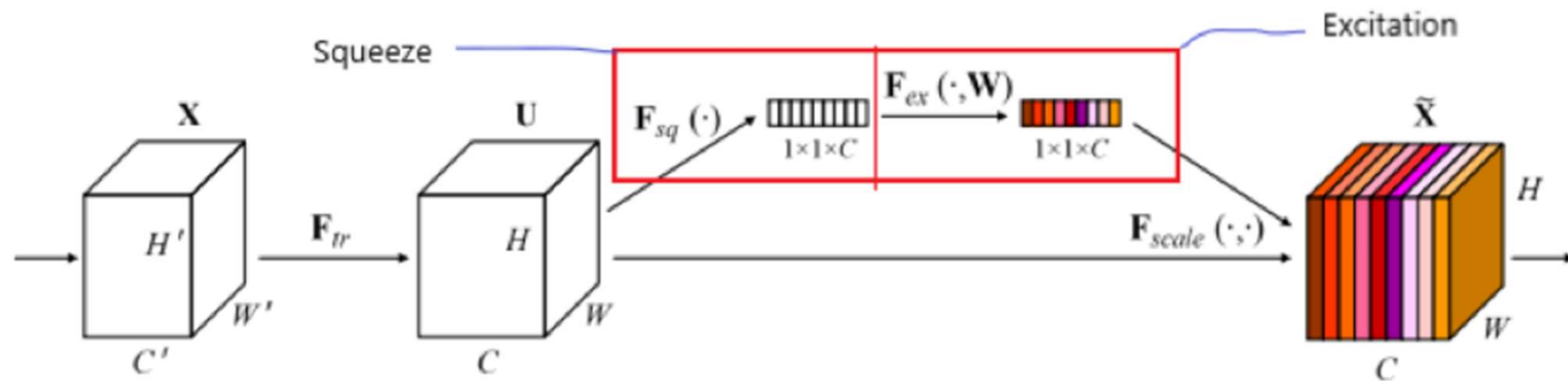
Dilated Convolution은 Convolutional layer에 또 다른 파라미터인 dilation rate가 추가됨

dilation rate은 커널 사이의 간격을 정의

dilation rate가 2인 3x3 커널은 9개의 파라미터를 사용하면서 5x5 커널과 동일한 시야(view)를 가짐

필터 내부에 zero padding을 추가하여 강제로 receptive field를 늘려주어 동일한 계산 비용으로 더 넓은 시야를 제공

BAG OF SPECIALS

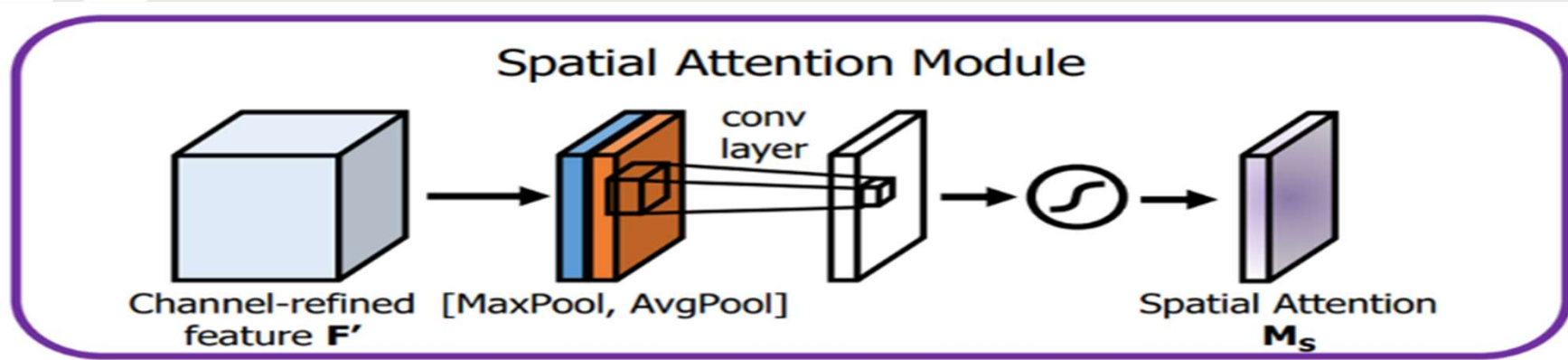


SE Block (Squeeze-and-Excitation Blocks)

Feature map의 각 채널마다 가중치를 부여하여 feature map의 각 채널에 곱함

Squeeze(압축) + Excitation(재조정)을 통해 가중치가 큰 channel은 중요한 특징을 가짐

BAG OF SPECIALS

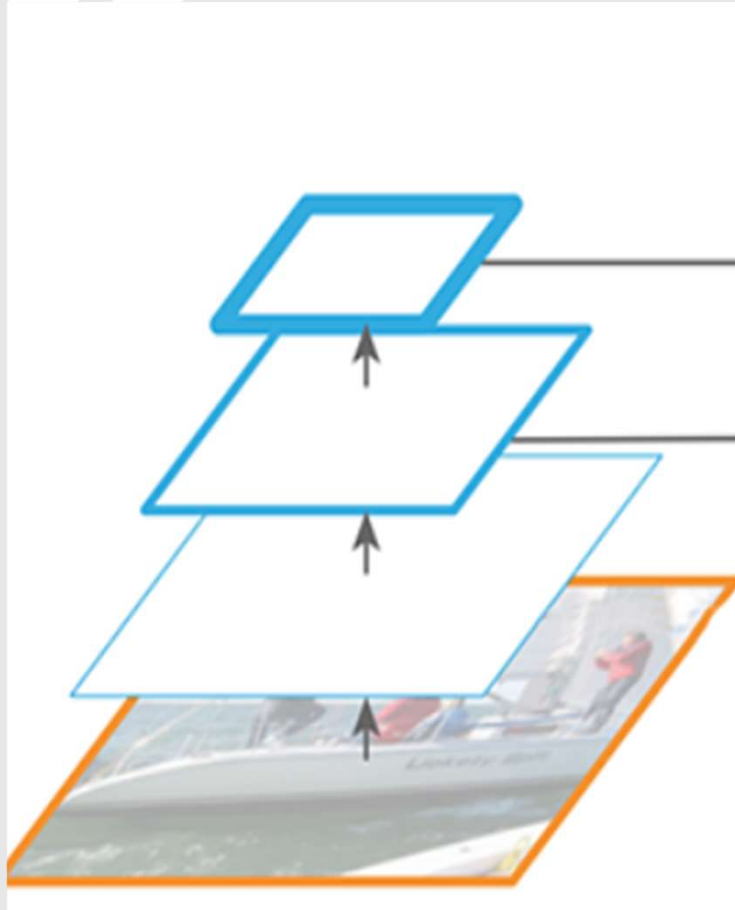


SAM (Spatial Attention Module)

Average Pooling과 Max Pooling을 channel 축(Axis)으로 적용함
 $C(\text{channel 개수}) \times H \times W \rightarrow (1 \times H \times W) \times 2(\text{AP, MP})$ 로 변환됨
Channel 축으로 적용된 pooling은 영역 정보를 효과적으로 강조함
압축된 channel에 7x7 Conv Layer를 적용하여 Spatial Attention map 생성

0.1%의 추가적인 계산만으로도, ImageNet을 이용한 image classification task에서
ResNet50-SE의 top-1 accuracy 성능을 0.5% 향상, GPU 상에서 inference 속도에 영향 X

BAG OF SPECIALS

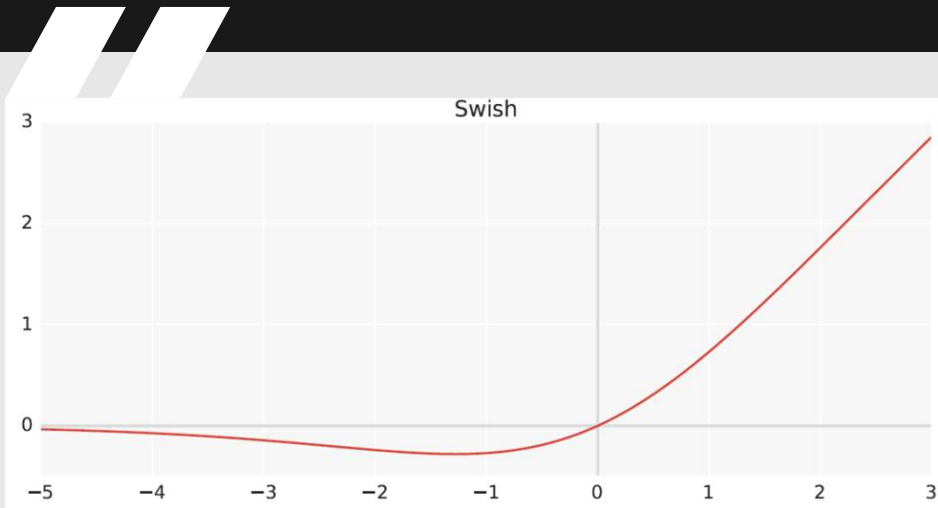


FPN 이후 연구들

multi-scale의 예측 방법들이 대중화되면서, 서로 다른 feature pyramid를 통합하는 많은 경량 module들이 제안됨

1. SFAM : SE module을 사용해 multi scale로 연결 된 feature map에서 channel-wise level의 re-weighting을 함
2. ASFF : point-wise level의 re-weighting을 위해 softmax(0~1 정규화)를 사용한 후 서로 다른 scale들의 feature map들을 추가함
3. BiFPN : scale-wise level의 re-weighting을 수행한 후 서로 다른 scale들의 feature map들을 추가하기 위해 multi-input의 weighted residual connections(과거의 conv를 미래 conv에 연결
->중간의 conv step이 skip됨)이 제안됨

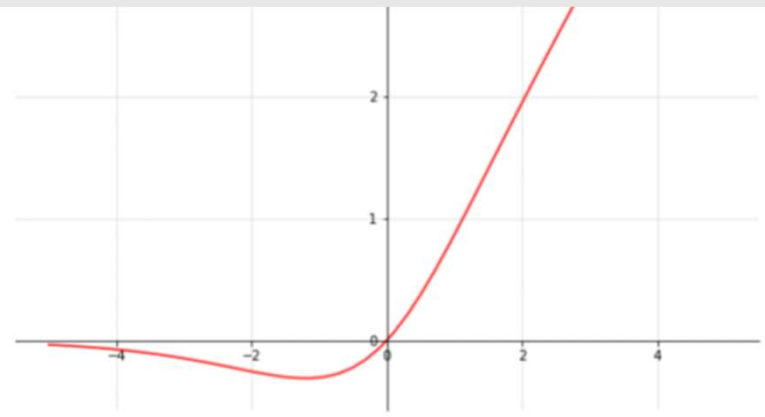
BAG OF SPECIALS



Swish

Swish 는 매우 깊은 신경망에서 ReLU 보다 높은 정확도를 달성

모든 $x < 0$ 에 대해 함수를 감소시키거나 증가시키지 않음



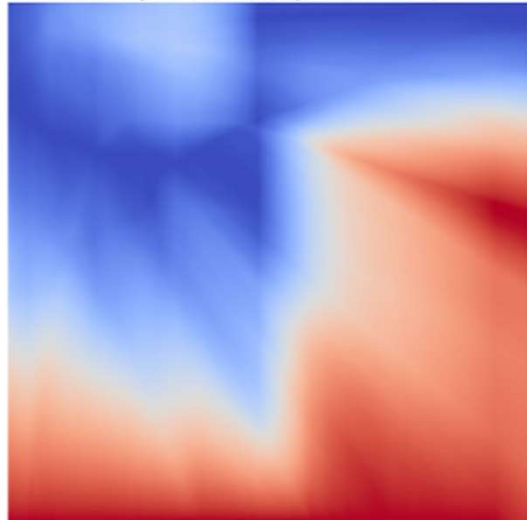
Mish (Swish 기반)

양의 값이 어떤 높이로든 갈 수 있어 포화를 피할 수 있음

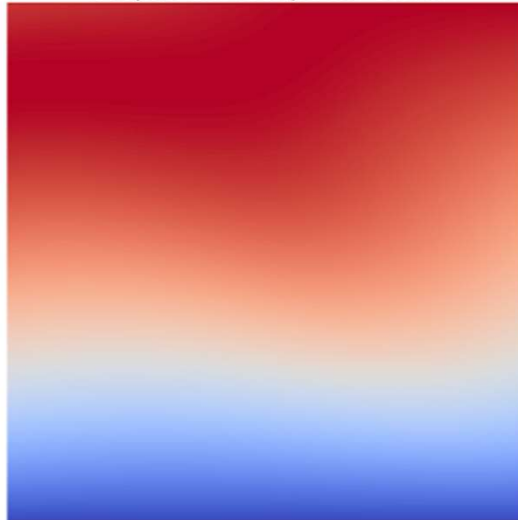
약간의 음수를 허용해 ReLU보다 학습이 더 잘됨

BAG OF SPECIALS

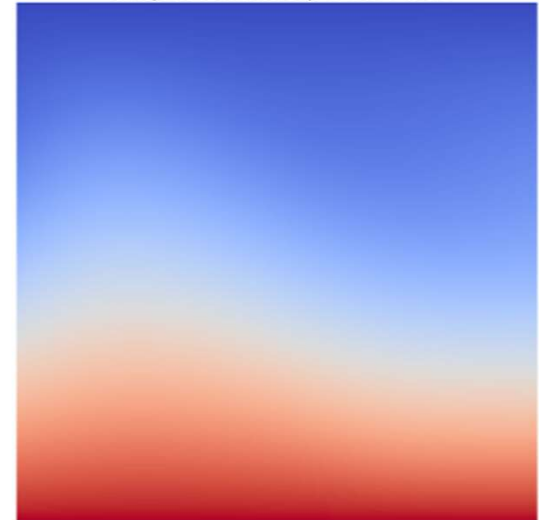
Output Landscape for ReLU



Output Landscape for Swish



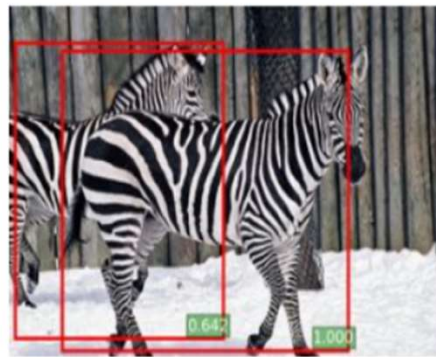
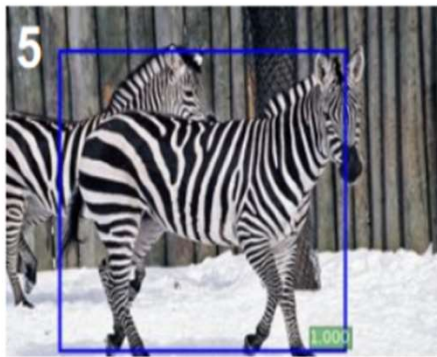
Output Landscape for Mish



ReLU는 Swish 와 Mish에 비해 sharp transition, 즉 날카롭게 바뀌춤

하지만, Swish 와 Mish의 경우 Smoother transition, 즉 보다 부드러운 최적화 기능을 제공하여 손실을 줄이면서 신경망을 일반화 함

BAG OF SPECIALS



왼쪽: NMS, 오른쪽: Soft-NMS

$$s_i = \begin{cases} s_i, & \text{iou}(\mathcal{M}, b_i) < N_t \\ 0, & \text{iou}(\mathcal{M}, b_i) \geq N_t \end{cases}$$

기존 NMS 함수

$$s_i = \begin{cases} s_i, & \text{iou}(\mathcal{M}, b_i) < N_t \\ s_i(1 - \text{iou}(\mathcal{M}, b_i)), & \text{iou}(\mathcal{M}, b_i) \geq N_t \end{cases},$$

Soft-NMS 함수 1

Soft-NMS

Soft-NMS는 기존의 NMS와는 달리 bbox를 억제하는 것이 아니라, confidence를 감소시킴
식을 보면 $0 \rightarrow s_i(1 - \text{iou}(m_i, b_i))$ 로 바뀜

기존 NMS의 경우 0으로 없어지지만 Soft-NMS는 0이 아닌 낮은 confidence로 bbox 표현

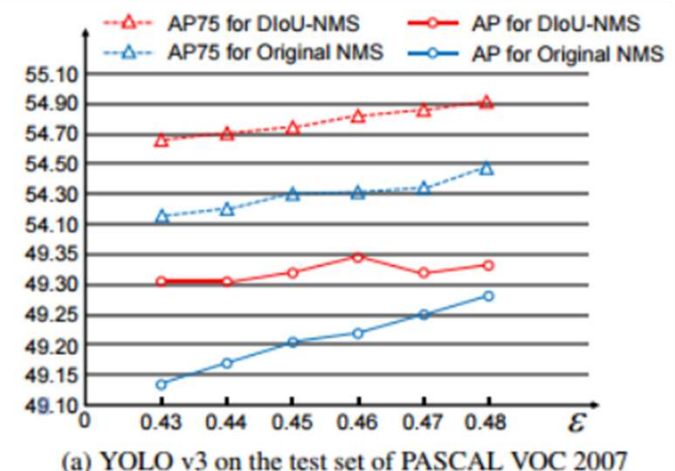
BAG OF SPECIALS



$\mathcal{L}_{CIoU} + NMS$



$\mathcal{L}_{CIoU} + DIoU-NMS$



DIoU-NMS

soft NMS에 기초하여, bbox screening process에 center point 거리에 대한 정보를 추가
위 그림처럼 동일한 class의 GT가 여러 개 겹쳐져 있는 경우 IoU에 비해 robust 함
기존 NMS에서 억제 되었던 bbox들이 DIoU-NMS에서는 나타남

두 box가 겹치지 않을 때, GIoU처럼 Bbox의 영역을 넓히지 않고 중심 좌표를 통해 박스의
거리 차이를 최소화 함으로써 수렴 속도를 향상 -> 성능 향상

ARTIFICIAL

INTELLIGENCE

[AI]

THANKS A LOT
FOR LISTENING



[AI]