

Open DMQA Seminar

Revisiting CNNs

2022. 03. 11

발표자 : 김정원

발표자 소개



❖ 이름 : 김정원

- 고려대 산업경영공학과 석사과정
- 데이터마이닝 및 품질애널리틱스(DMQA) 연구실

❖ 연구분야

- Object detection, Semantic segmentation
- Scene text detection & recognition
- ML for multivariate & time-series data

❖ 연락처

- jwnkim@korea.ac.kr

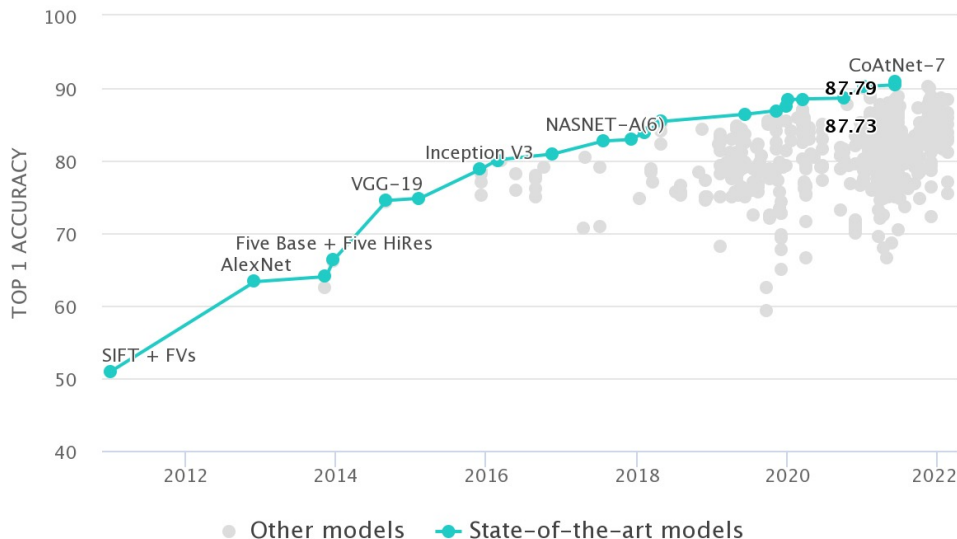
목차

- ❖ Introduction
- ❖ Basic concepts
- ❖ CNN vs Transformer
- ❖ Findings
- ❖ Hybrid CNNs
- ❖ Conclusion

Introduction

❖ Evolution of CNNs

- 딥러닝의 발전은 합성곱 신경망(convolutional neural network; CNN)의 발전이라고 할 수 있음
- 2012년 이미지 분류 대회 ImageNet에서 CNN 기반 AlexNet이 월등한 성능을 보인 후, scaling, skip connection 등 기법을 이용해 구조 변경하며 성능을 개선해 옴

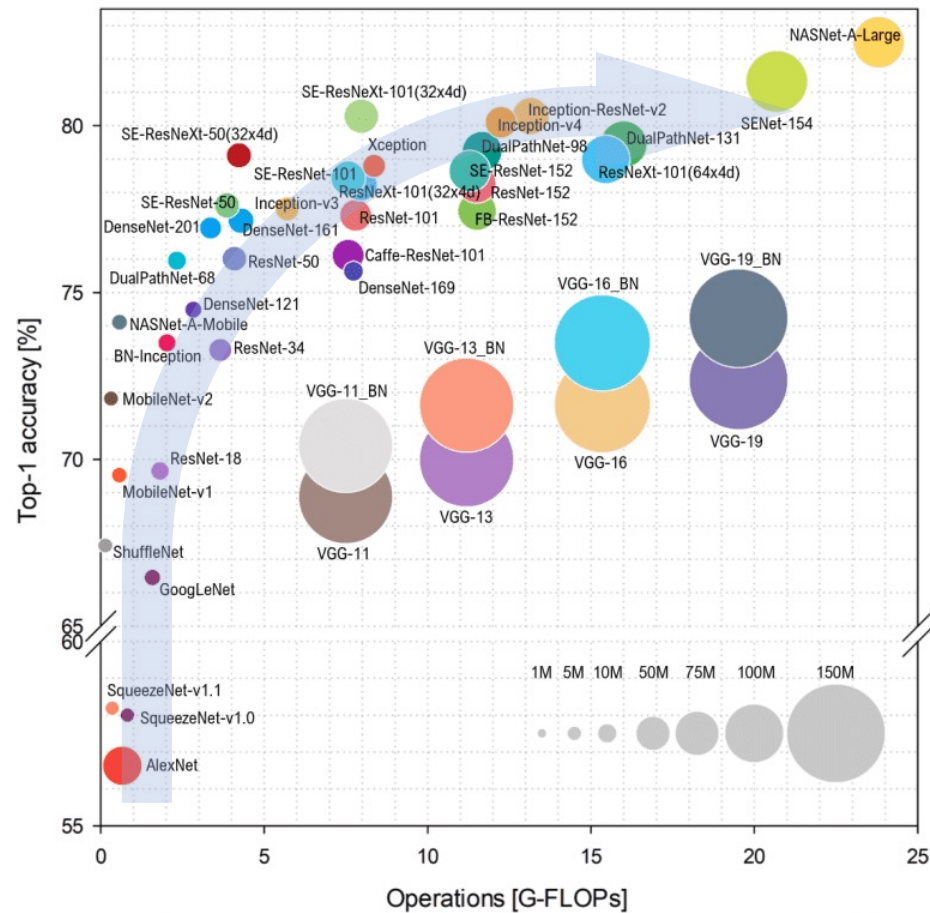


(source: <https://paperswithcode.com/sota/image-classification-on-imagenet>)

year	model
2012	AlexNet
2014	GoogleNet
2014	VGG
2015	Inception V2~V4
2015	ResNet
2015	DenseNet
...	
2019	EfficientNet

Introduction

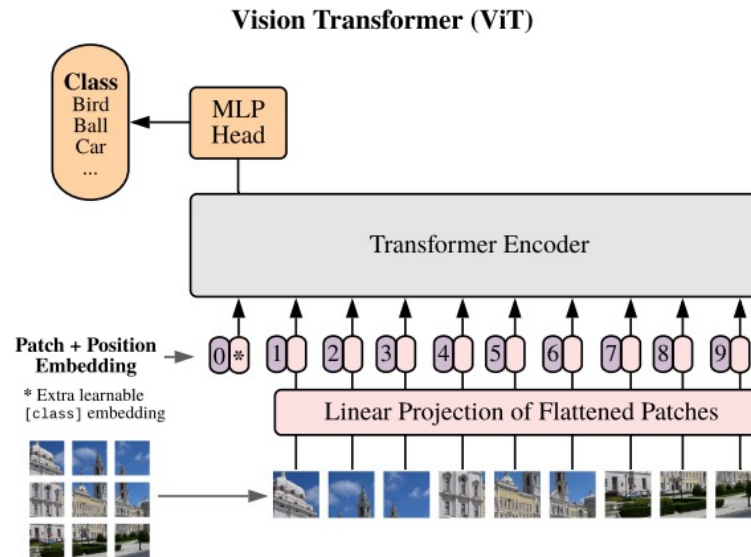
❖ Evolution of CNNs



Introduction

❖ Vision Transformer

- 자연어처리(NLP) 분야에서 좋은 성능을 보인 Transformer 모델이 컴퓨터 비전 분야에서 활용되기 시작함
- Transformer 구조 활용한 Vision Transformer(ViT)가 SOTA급 성능을 보임
- 이후 DeiT, Swin Transformer 등 한계점 개선해가며 활발히 연구되고 있음



An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale(Dosovitskiy et al, 2021)

Introduction

❖ Which one is better, CNN vs Transformer?

- Transformer 구조가 상당한 발전 가능성을 보여주고 있지만, 향후 CNN 구조를 대체할 수 있을지에 대해선 전망이 나뉨
- 이미지 처리에 있어 CNN 구조의 강력한 이점이 있기 때문에 Transformer가 이를 극복할 것인지가 관건



CNN

vs



Transformer

Introduction

❖ Three main questions of this seminar

- 1) 모델의 기본 구조로서 CNN과 Transformer 중 어떤 것이 더 성능이 좋을까?
 - A Battle of Network Structures: An Empirical Study of CNN, Transformer, and MLP(Zhao et al., 2021)
- 2) 비전 과제에서 모델 성능을 높이는 주요 요인은 무엇일까?
 - Revisiting ResNets: Improved Training and Scaling Strategies(Bello et al., 2021)
- 3) Transformer의 이점을 CNN에 적용해 발전시킬 수는 없을까?
 - A ConvNet for the 2020s(Liu et al., 2022)

Basic concepts

❖ CNN

- 합성곱(convolution) 연산을 통해 이미지로부터 특징을 추출하는 방식의 신경망

2	3	7	4	6
6	6	9	8	7
3	4	8	3	8
7	8	3	6	6
4	2	1	8	3

Input

*

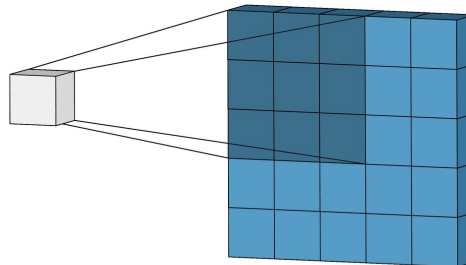
-1	2	1
-1	2	1
-1	1	0

3x3 filter

=

27	39	16
29	30	23
23	18	28

Output

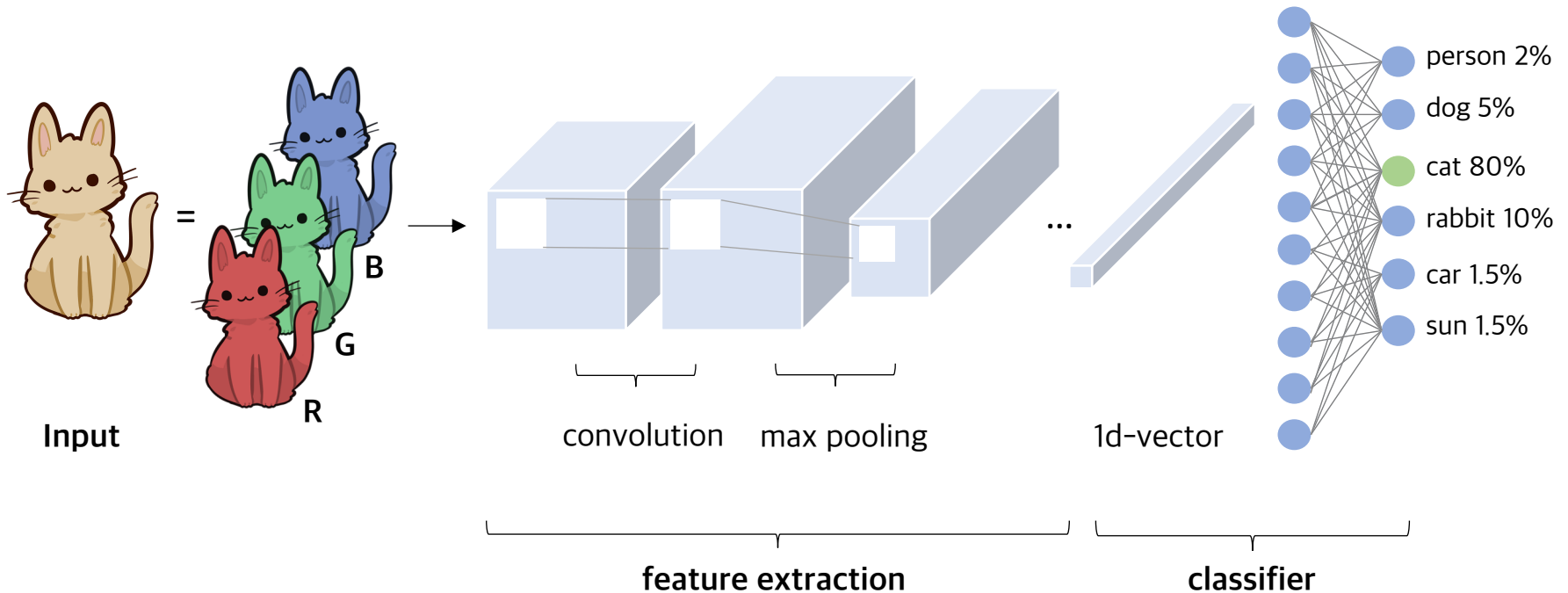


source: <https://towardsdatascience.com/intuitively-understanding-convolutions-for-deep-learning-1f6f42faee1>

Basic concepts

❖ CNN

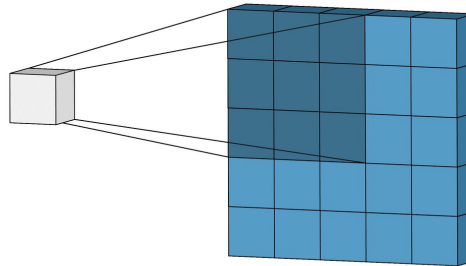
- convolution 층과 pooling 층을 거친 후 fully-connected 층을 통해 이미지를 분류함
- 그외 객체 검출(object detection), 이미지 분할(semantic segmentation) 등 컴퓨터 비전 과제에서도 특징 추출을 위한 백본 역할을 함



Basic concepts

❖ CNN의 장점

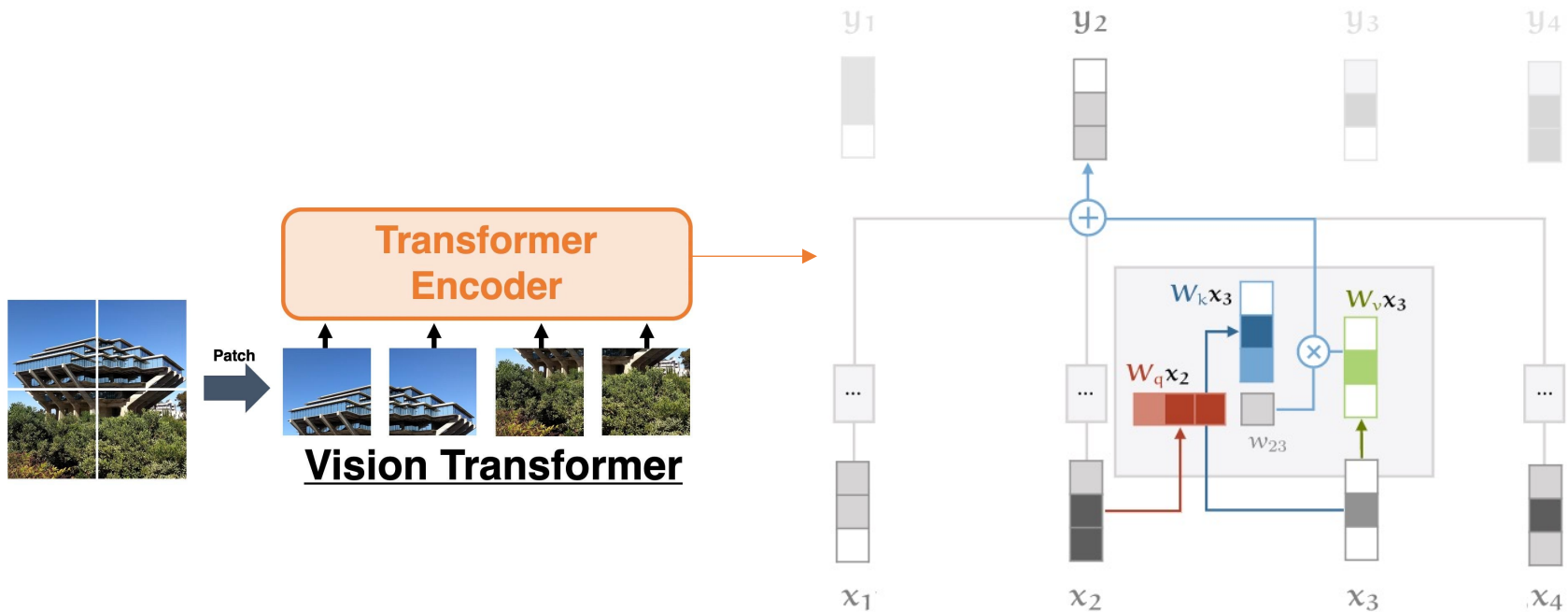
- Locality를 잘 반영할 수 있음 ★★★★★
 - 필터를 통해 local($\leftrightarrow$ global)한 정보를 다양한 층위로 추출함
- 연산이 효율적임 ★★★★★
 - 가중치를 공유하는 하나의 필터를 sliding window 방식으로 작동시켜 연산량 감소
- 객체의 위치가 바뀌면 출력도 바뀐(translation equivariance) ★★★
 - 객체 검출 등 다른 비전 과제에서 중요한 특성



Basic concepts

❖ Vision Transformer

- Self-attention을 이용해 이미지 정보를 임베딩하는 방식의 모델
- 이미지를 16 x 16 크기의 패치로 분리한 다음, 패치들 간 상관관계를 바탕으로 특징을 추출함
- 패치들 가운데 중요한 영역을 강조하도록 임베딩 됨



Basic concepts

❖ Vision Transformer

- Self-attention을 이용해 이미지 정보를 임베딩하는 방식의 모델
- 이미지를 16 x 16 크기의 패치로 분리한 다음, 패치들 간 상관관계를 바탕으로 특징을 추출함
- 패치들 가운데 중요한 영역을 강조하도록 임베딩 됨

종료
Transformer in Computer Vision
Open DMOA Seminar
2021.03.26
조한샘

Transformer in Computer Vision
발표자:  **조한샘**
📅 2021년 3월 26일
🕒 오후 1시 ~
📺 온라인 비디오 시청 (YouTube)

종료
How to train your ViT? Data, Augmentation, and Regularization in Vision Transformers
2022.01.21
Data Mining and Quality Analytics Lab

How to train your ViT? Data, Augmentation, and Regularization in Vision Transformers
발표자:  **박진혁**
📅 2022년 1월 21일
🕒 오후 1시 ~
📺 온라인 비디오 시청 (YouTube)

Basic concepts

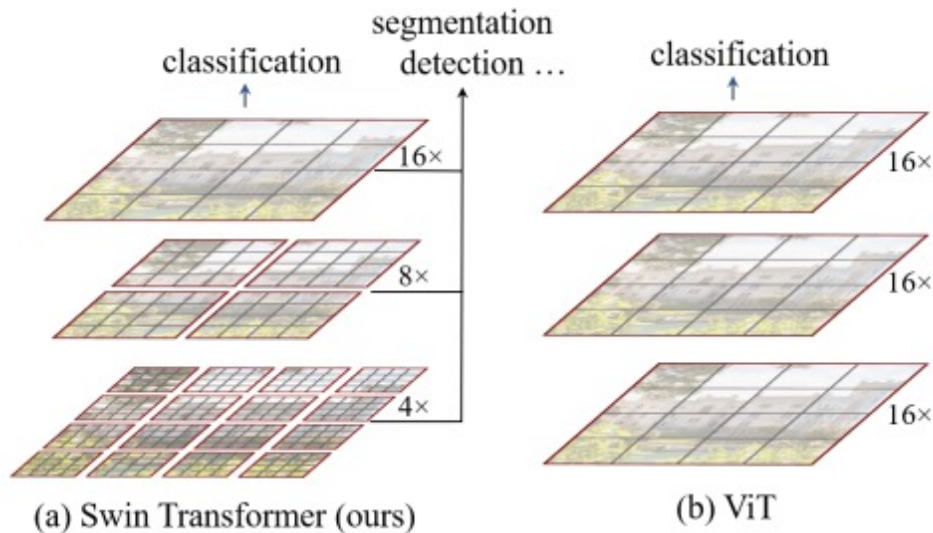
❖ Vision Transformer의 장단점

- Global($\leftrightarrow$ local) attention을 기반으로 해 이미지를 한꺼번에 분석 가능 ★★★★★
- 이미지에 특화된 inductive bias가 없기 때문에 대규모 학습 데이터셋이 필요함
- 이미지 크기가 커지면 연산량이 기하급수적으로 늘어남

Basic concepts

❖ Swin Transformer

- ViT에서 패치화 된 이미지는 하나의 벡터로 취급되기 때문에 어떻게 패치를 나누는지가 중요함
- CNN은 연속 convolution을 통해 local 단위에서 다층적으로 특징을 추출함
- 이를 적용해 이미지를 여러 window로 나눈 뒤, 한 window에 대해서만 self-attention을 수행 → stage를 진행할 수록 패치를 병합해 더 큰 window에서 self-attention 수행하는 모델



CNN vs Transformers

1) 모델의 기본 구조로서 CNN과 Transformer 중 어떤 것이 더 성능이 좋을까?

- 두 모델의 핵심적인 부분을 가져와 동일한 구성으로 만든 뒤 비교 실험
- 결론적으로 “큰 차이가 없다!”

A Battle of Network Structures: An Empirical Study of CNN, Transformer, and MLP

Yucheng Zhao^{*†1} Guangting Wang^{*†1} Chuanxin Tang^{*2} Chong Luo² Wenjun Zeng²

Zheng-Jun Zha¹

University of Science and Technology of China¹

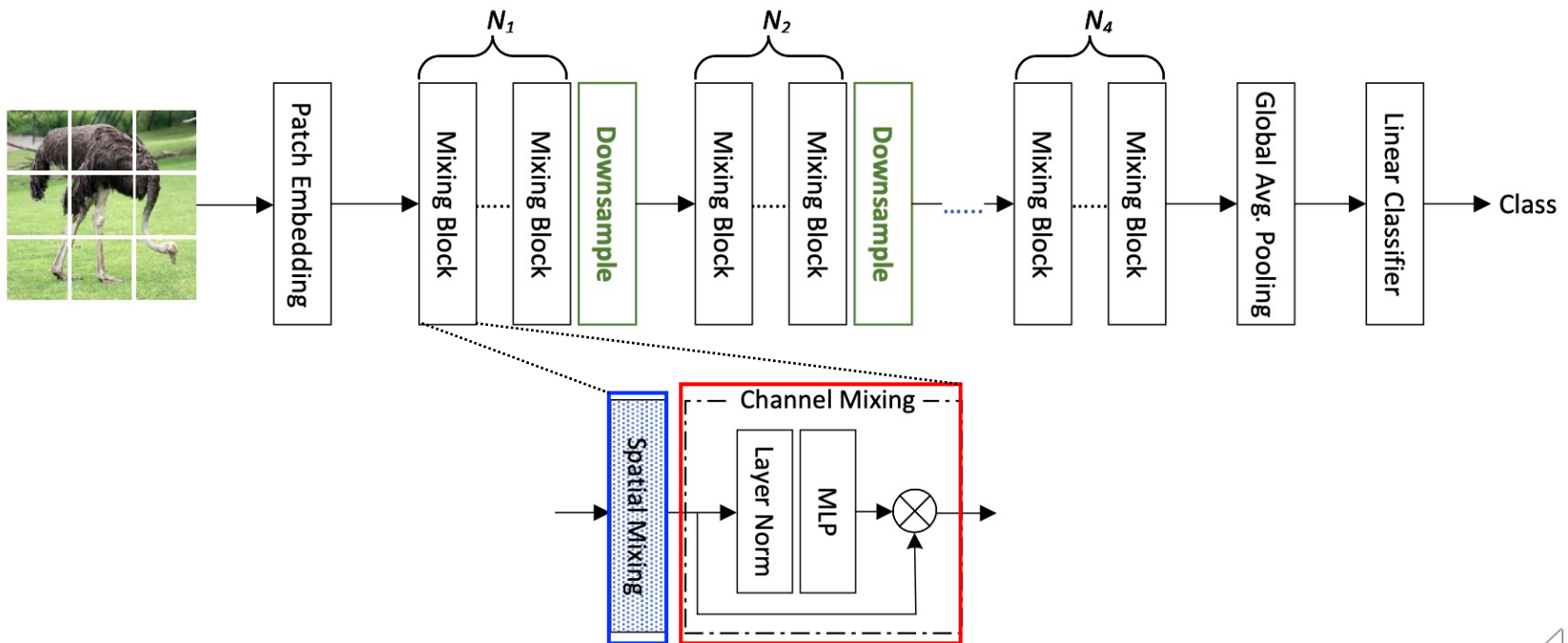
Microsoft Research Asia²

{lnc, flylight}@mail.ustc.edu.cn {chutan, cluo, wezeng}@microsoft.com zhazj@ustc.edu.cn

CNN vs Transformers

❖ 실험 방식

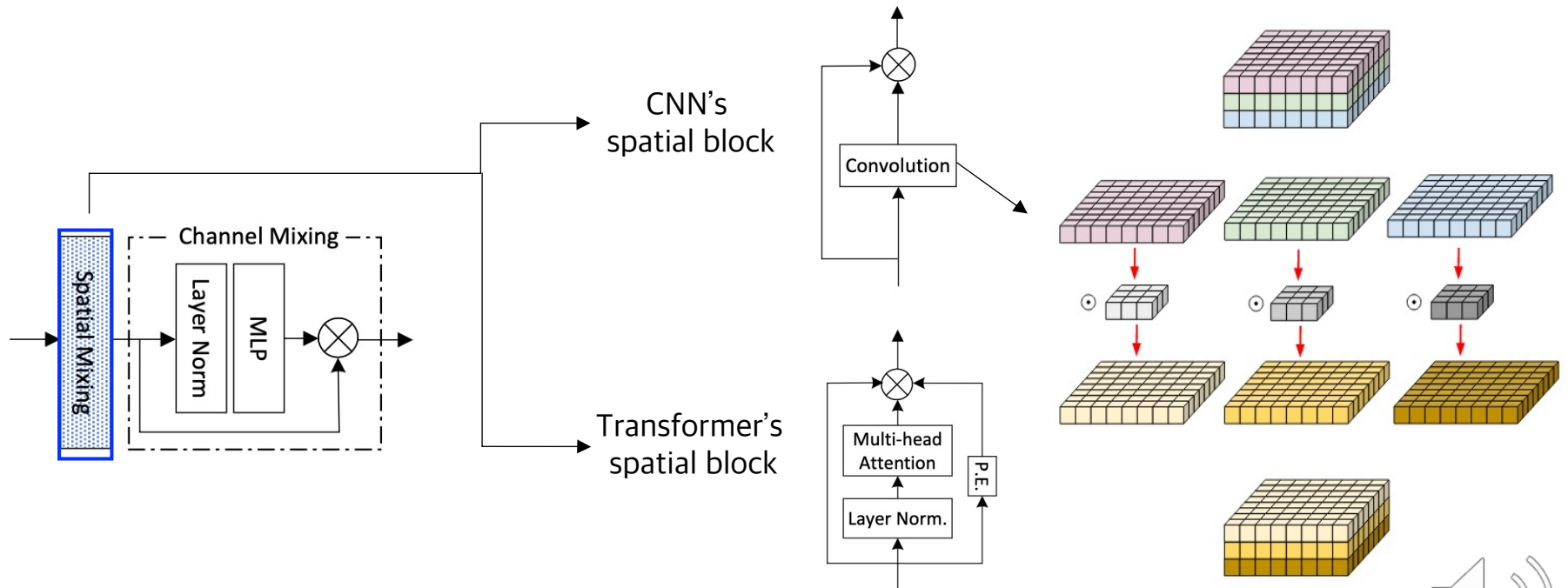
- 크게 2가지 블록으로 이뤄진 Mixing 블록을 쌓아 모델 구성
 - 공간 context 정보를 분석하는 **spatial 블록** : 각 CNN, Transformer 구조로 설계
 - 채널 정보를 분석하는 **channel 블록** : 동일하게 Layer normalization, Multi-layer Perceptron(MLP)으로 구성



CNN vs Transformers

❖ 실험 방식

- spatial 블록 구성
 - CNN : 3 x 3 depth-wise convolution 층으로 구성
 - Transformer : layer normalization, multi-head self-attention 층으로 구성
- : convolutional positional encoding(Chu et al., 2021) 모듈을 추가

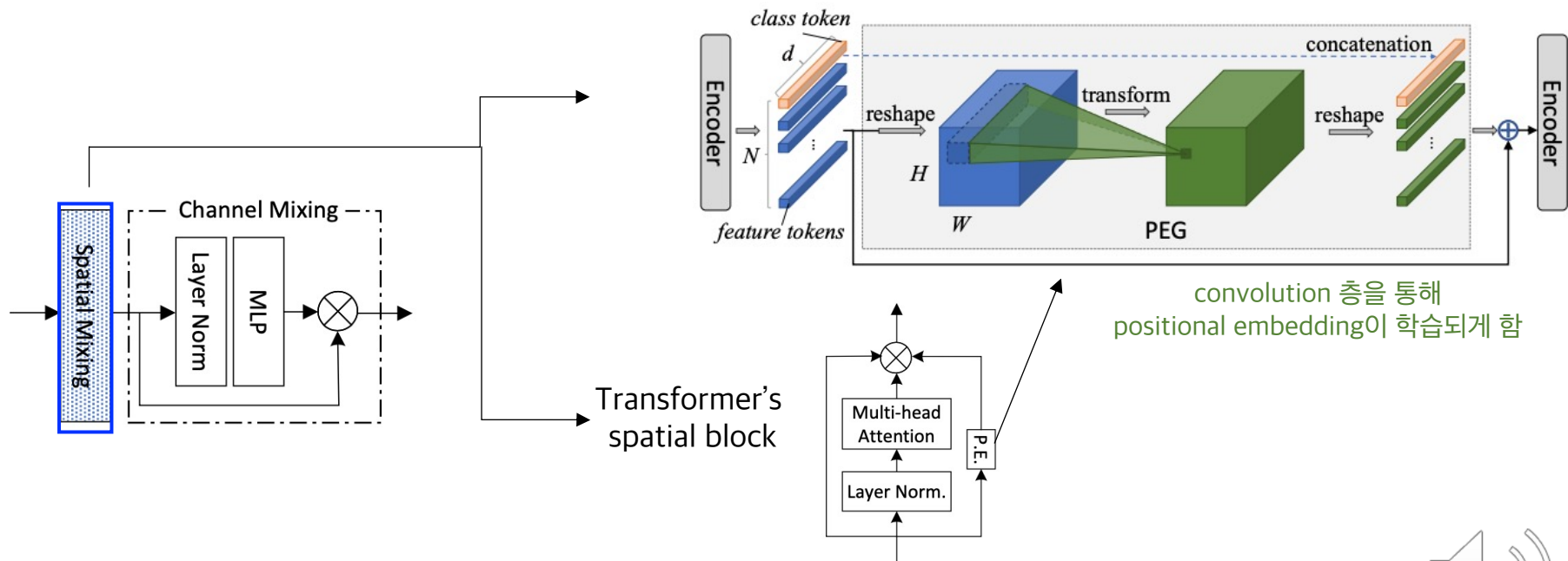


source: <https://medium.com/@zurister/depth-wise-convolution-and-depth-wise-separable-convolution-734676514ec>

CNN vs Transformers

❖ 실험 방식

- spatial 블록 구성
 - CNN : 3 x 3 depth-wise convolution 층으로 구성
 - Transformer : layer normalization, multi-head self-attention 층으로 구성
- : convolutional positional encoding(Chu et al., 2021) 모듈을 추가



CNN vs Transformers

❖ 실험 결과

- 전체 중에서는 Transformer(S)가 가장 좋은 성능을 보이지만, 중간 스케일(XS)에서는 두 성능이 동일하게 나타남
- 가장 가벼운 모델(XXS)에서는 CNN이 월등한 성능을 보여 CNN의 효율성을 입증함

Network scale	Model	variants	Top-1 accuracy(%)
XXS	CNN	$C = 64, R = 2, N_s = \{2, 2, 6, 2\}$	75.3
	Transformer	$C = 32, R = 2, N_s = \{2, 2, 6, 2\}$	65.4
XS	CNN	$C = 96, R = 2, N_s = \{3, 4, 12, 3\}$	80.1
	Transformer	$C = 64, R = 2, N_s = \{3, 4, 12, 3\}$	80.1
S	CNN	$C = 128, R = 3, N_s = \{3, 4, 12, 3\}$	81.6
	Transformer	$C = 96, R = 3, N_s = \{3, 4, 12, 3\}$	82.9

C : feature dimension, R : expansion ratio of MLP in channel block, N_s : number of mixing blocks

CNN vs Transformers

❖ 실험 결과

- **Multi-stage** 구조가 single-stage보다 무조건 유리하다
 - 특징 추출을 위한 블록(mixing block)을 전부 이어붙이는 것보다 여러 번에 나눠 down-sampling하는 것이 성능이 좋음
- 이미지 처리에 있어 **Locality** 모델링은 매우 중요하다
 - CNN 구조는 convolutional 층만으로 XXS, XS 버전에서 Transformer 이상의 성능을 냄
 - Transformer 구조에서 convolutional positional encoding 모듈을 없애고 기존 ViT의 positional embedding 방법론을 썼을 때 성능이 0.6%p 하락함
- 연산 효율성, 일반화 성능이 좋은 CNN 구조와 모델 capacity가 좋은 Transformer의 **융합**할 경우 추가적인 성능 개선을 이룰 수 있을 것

Findings

2) 비전 과제에서 모델 성능을 높이는 주요 요인은 무엇일까?

- CNN 기반 대표적인 모델인 ResNet의 이미지 분류 성능을 최대로 끌어올릴 수 있는 기법 탐색
- 주요 모델 구조의 변경 없이 학습 방식, 규제(regularization) 전략만으로 성능을 상당폭 개선할 수 있음을 지적함

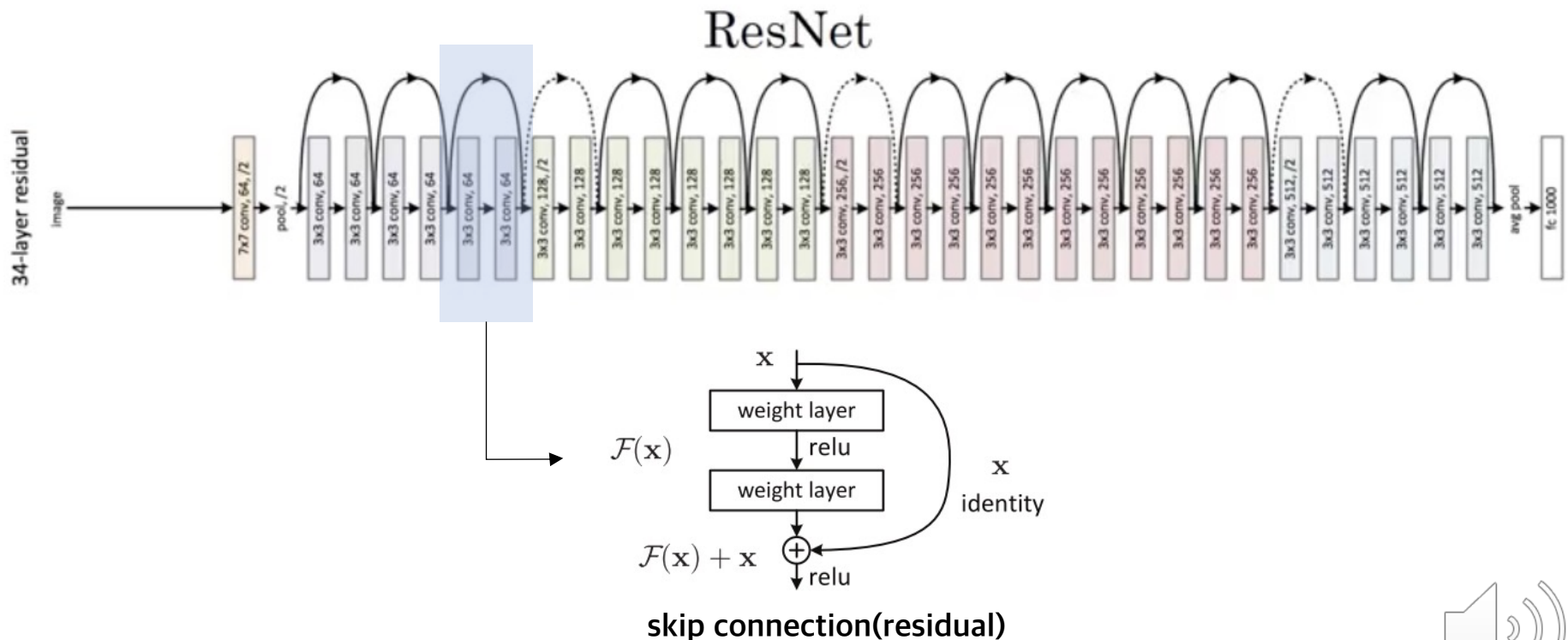
Revisiting ResNets: Improved Training and Scaling Strategies

Irwan Bello¹ William Fedus¹ Xianzhi Du¹ Ekin D. Cubuk¹ Aravind Srinivas² Tsung-Yi Lin¹
Jonathon Shlens¹ Barret Zoph¹

Findings

❖ ResNet 기본 구조

- Deep Residual Learning for Image Recognition(He et al., 2015)
- 딥러닝 모델이 너무 깊어져 생기는 기울기 소실(vanishing gradient) 문제를 skip connection으로 해결한 모델



Findings

❖ 실험 방식

- 학습 방식
 - Cosine learning rate decay
 - Momentum optimizer
- 규제 전략
 - Weight decay
 - Label smoothing
 - Dropout
 - Stochastic depth
- 데이터 증강
 - RandAugment(Cubuk et al., 2019)
- 구조 변경
 - Squeeze-and-excitation
 - ResNet-D

Findings

❖ 규제 전략 ① Weight decay

- 모델 복잡도가 높아짐에 따라 gradient descent 기반 학습 시 가중치가 점차 증가하는 현상이 발생함
- 이로 인한 과적합을 막기 위해 loss에 가중치의 제곱합을 페널티를 부과하는 제약을 걸어줌

• 규제 전략

- Weight decay
- Label smoothing
- Dropout
- Stochastic depth

Loss

$$Loss(w, x) = DataLoss(w, x) + \frac{1}{2} \lambda \|w\|^2$$

미분

$$\begin{aligned} w &\leftarrow w - \eta \left(\frac{\partial DataLoss}{\partial w} + \lambda w \right) \\ &= w(1 - \eta \lambda) - \eta \frac{\partial DataLoss}{\partial w} \quad (\eta : \text{학습율}) \end{aligned}$$

코드

```
optimizier = torch.optim.Adam(model.parameters(),  
                                lr=1e-3,  
                                weight_decay=0.9)
```

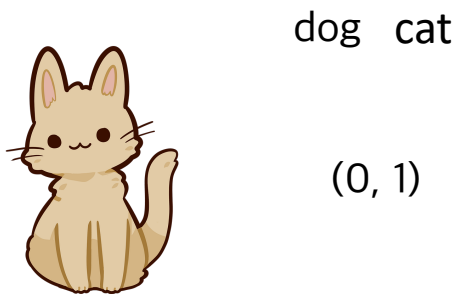
source: <https://sanghyunchoi.github.io/88>

Findings

❖ 규제 전략 ② Label smoothing

- 분류 과제에서 $(1, 0, 0, \dots, 0)$ 으로 표기하는 레이블을 soft하게 바꾸는 기법
- 모델이 smooth하게 예측하게 유도함으로써 과적합을 방지함

- 규제 전략
 - Weight decay
 - [Label smoothing](#)
 - Dropout
 - Stochastic depth



$$y_k^{smooth} = y_k(1 - \alpha) + \alpha/K$$

α : smoothing factor
 K : number of class

dog cat

$(0.1, 0.9)$

$(0.9, 0.1)$

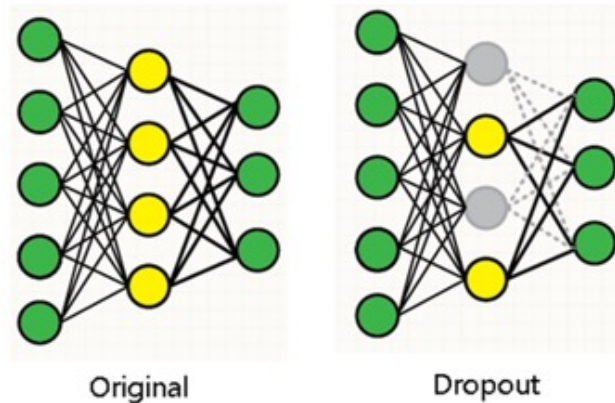
Findings

❖ 규제 전략 ③ Dropout

- 학습 시 Fully connected layer의 노드 중 일부를 랜덤하게 선택해 연결을 끊고 남은 노드로만 학습하는 방법
- 매번 서로 다른 모델이 학습되는 셈이기 때문에 ensemble 효과가 일어남
- 노드마다 중복되지 않고 서로 다른 특성을 학습함

- 규제 전략

- Weight decay
- Label smoothing
- **Dropout**
- Stochastic depth



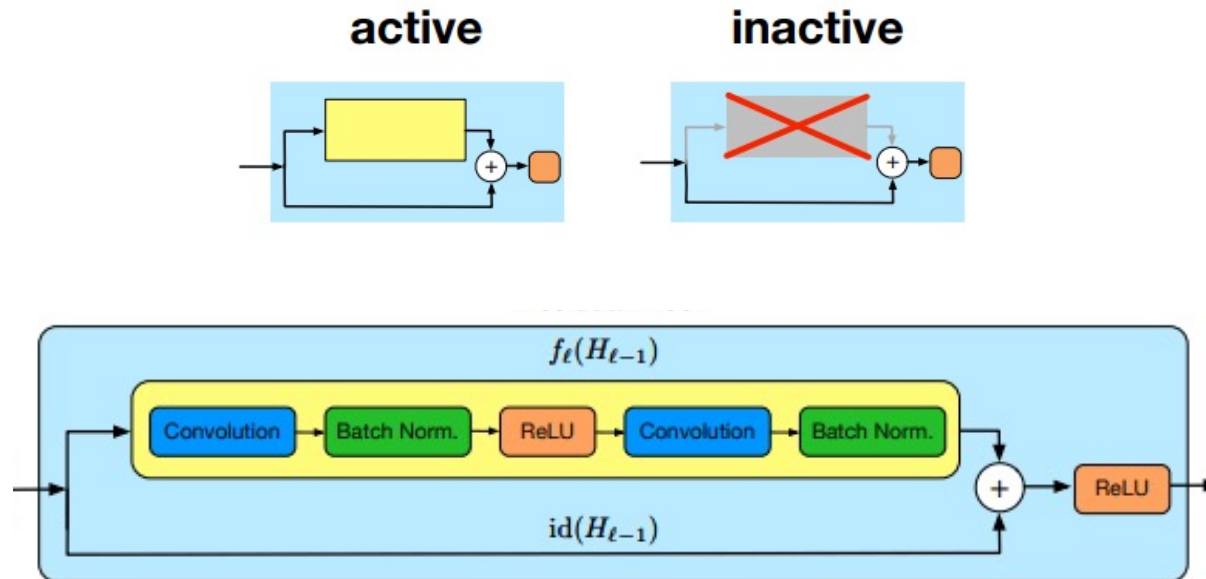
Findings

❖ 규제 전략 ④ Stochastic depth

- Residual block에서 학습되는 convolution 블록을 건너 뛰게 만듦
- 깊은 네트워크를 유지하면서도 dropout과 유사한 효과를 갖게 함
- Deep Networks with Stochastic Depth(Huang et al., 2016)

- 규제 전략

- Weight decay
- Label smoothing
- Dropout
- Stochastic depth



Findings

❖ 구조 변경 ① Squeeze-and-excitation

- Squeeze-and-excitation networks(Hu et al., 2018) : SENet
- Global average pooling(GAP)으로 정보를 축약
 - Fully connected layer로 차원 수 줄였다 원 채널 수로 복원
 - 원래의 특징 지도에 곱해 채널마다 가중치 부여

- 구조 변경
 - Squeeze-and-excitation
 - ResNet-D

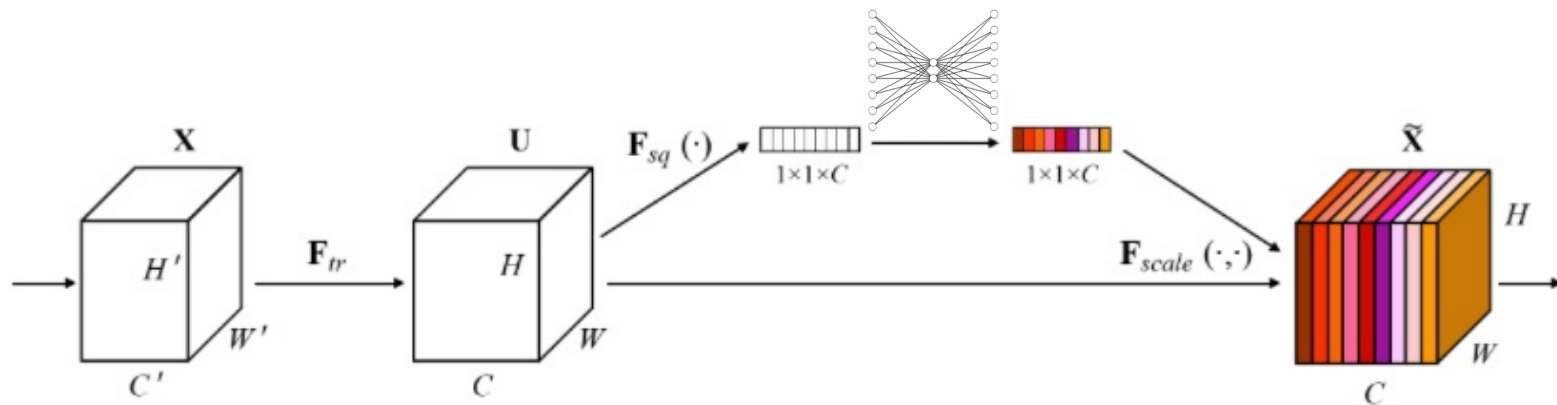


Figure 1: A Squeeze-and-Excitation block.

Findings

❖ 구조 변경 ② ResNet-D

- Bag of tricks for image classification with convolutional neural networks(He et al., 2018)
 - 커널 크기와 stride, max pooling 설정을 조정

- 구조 변경
 - Squeeze-and-excitation
 - ResNet-D

Block Group	Output Size	Convolution Layout
stem	112x112	3x3, 64, s2 3x3, 64 3x3, 64 x1
c2	56x56	1x1, 64 3x3, 64 1x1, 256 x3
c3	28x28	1x1, 128 3x3, 128 1x1, 512 x4
c4	14x14	1x1, 256 3x3, 256 1x1, 1024 x23
c5	7x7	1x1, 512 3x3, 512 1x1, 2048 x3
	1x1	Avg Pool Dropout 1000-d FC x1



Findings

❖ 실험 결과

- 개선 전 ImageNet에 대해 79% 정확도 → 83.4%로 증가(+4.4%p)
- 학습 방식과 규제 전략으로 인한 정확도 향상이 ¾를 차지함(+3.2%p)

Improvements		Top-1	Δ
ResNet-200		79.0	—
학습 방식	+ Cosine LR Decay	79.3	+0.3
	+ Increase training epochs	78.8 †	-0.5
	+ EMA of weights	79.1	+0.3
규제 전략	+ Label Smoothing	80.4	+1.3
	+ Stochastic Depth	80.6	+0.2
	+ RandAugment	81.0	+0.4
	+ Dropout on FC	80.7 ‡	-0.3
	+ Decrease weight decay	82.2	+1.5
구조 변경	+ Squeeze-and-Excitation	82.9	+0.7
	+ ResNet-D	83.4	+0.5

Findings

❖ 실험 결과

- Weight decay의 적절한 사용이 매우 중요하다
 - 다른 규제 전략을 얼마나 쓰는지에 따라 낮춰주지 않으면 과도 규제가 되어 성능 하락
 - Dropout, Stochastic depth 사용할 때 lambda 값 줄여줘야 함

	Improvements	Top-1	Δ
	ResNet-200	79.0	—
학습 방식	+ Cosine LR Decay	79.3	+0.3
	+ Increase training epochs	78.8 [†]	-0.5
규제 전략	+ EMA of weights	79.1	+0.3
	+ Label Smoothing	80.4	+1.3
	+ Stochastic Depth	80.6	+0.2
	+ RandAugment	81.0	+0.4
	+ Dropout on FC	80.7 [‡]	-0.3
	+ Decrease weight decay	82.2	+1.5 V
구조 변경	+ Squeeze-and-Excitation	82.9	+0.7
	+ ResNet-D	83.4	+0.5

Hybrid CNNs

3) Transformer의 이점을 CNN에 적용해 발전시킬 수는 없을까?

- Transformer가 발전하고 있지만 이미지 분류 외 다양한 비전 과제에서 백본 네트워크로 활용되기엔 한계가 많다고 지적
- 반면 CNN은 이미지 처리에 적합한 inductive bias를 갖고 있어 백본 구조로 명확한 이점을 가짐
- CNN(ResNet)에 Transformer에 쓰인 기법들을 적용해 발전시킨 하이브리드 CNN 'ConvNext' 제안

A ConvNet for the 2020s

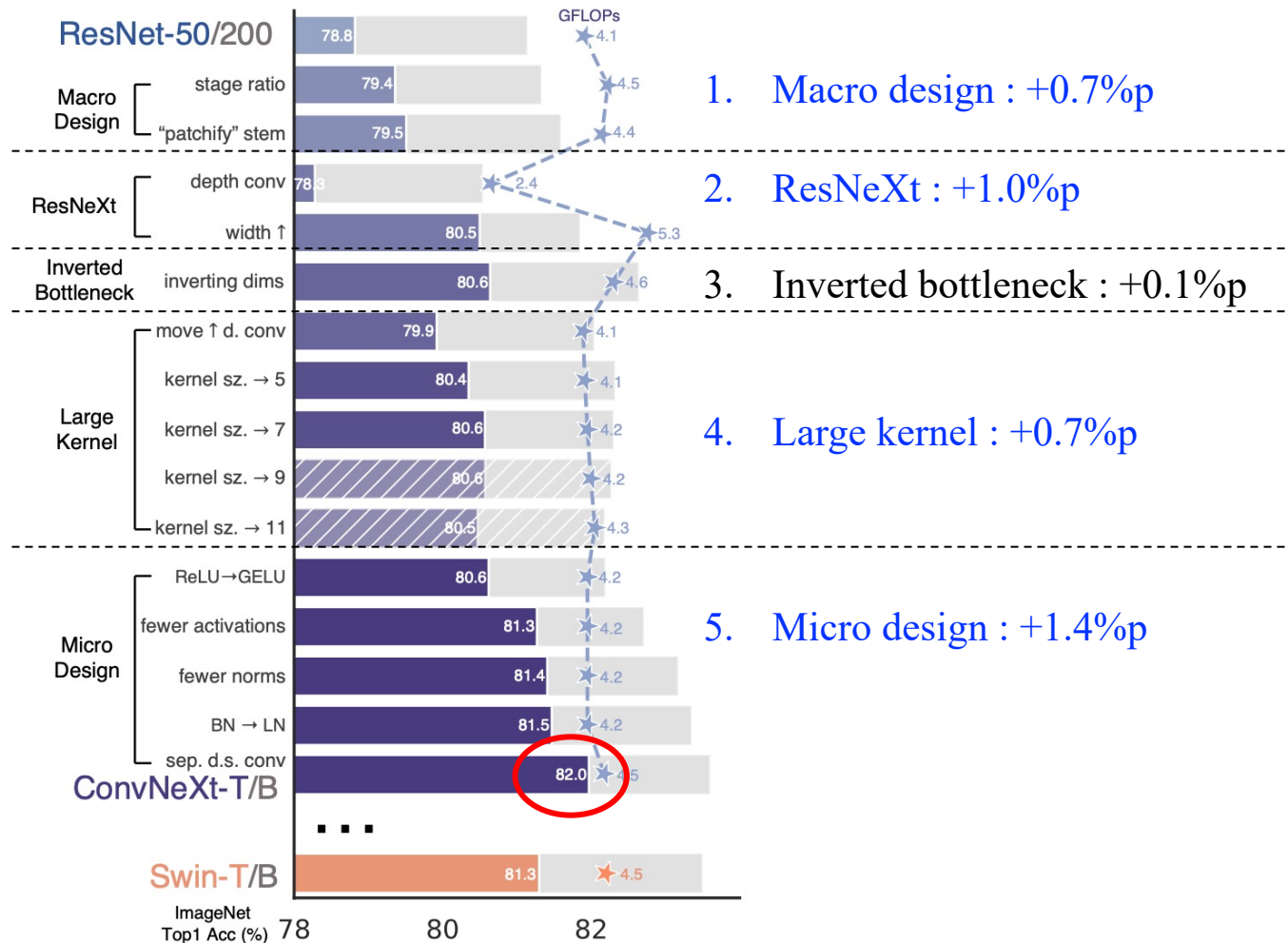
Zhuang Liu^{1,2*} Hanzi Mao¹ Chao-Yuan Wu¹ Christoph Feichtenhofer¹ Trevor Darrell² Saining Xie^{1†}

¹Facebook AI Research (FAIR) ²UC Berkeley

Code: <https://github.com/facebookresearch/ConvNeXt>

Hybrid CNNs

3) Transformer의 이점을 CNN에 적용해 발전시킬 수는 없을까?



Hybrid CNNs

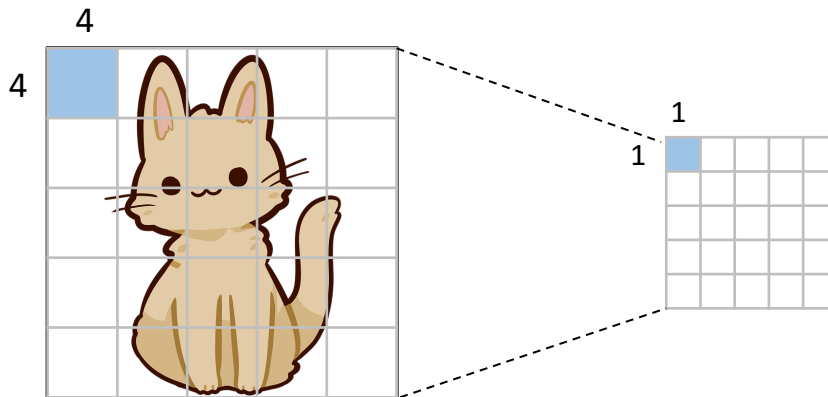
❖ 1. Macro design

① ResNet-50의 블록 수를 Swin-T와 유사한 비율로 조정

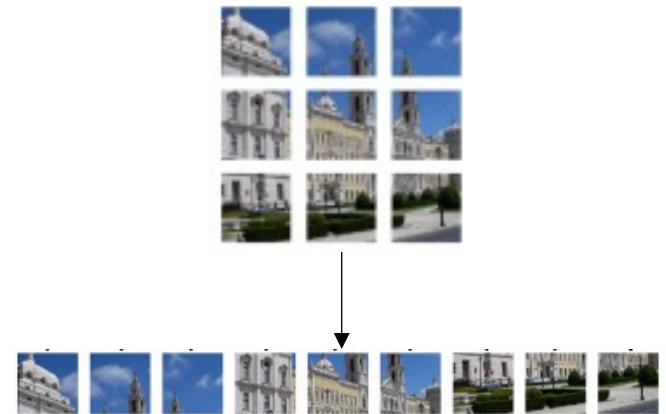
: (3, 4, 6, 3) → (3, 3, 9, 3)

② 이미지 입력 직후 convolution 레이어를 ViT 방식으로 변경

: 7 x 7 convolution (stride 2) → 4 x 4 convolution (stride 4)



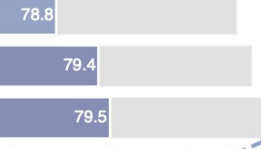
"patchify" stem



ViT

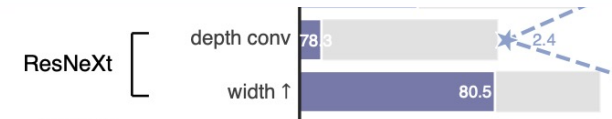
ResNet-50/200

Macro Design [stage ratio
"patchify" stem



Hybrid CNNs

❖ 2. ResNeXt

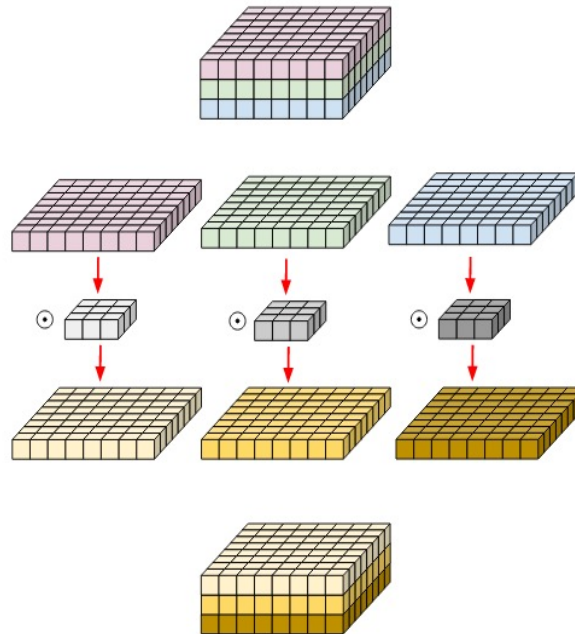


① Bottleneck 블록에 depthwise convolution 적용

: 채널별로 다른 가중치 필터를 사용하기 때문에 self-attention 방식과 유사한 효과 가짐

② 특징 지도 채널 수를 Swin-T와 동일하게 조정

: 64 → 96

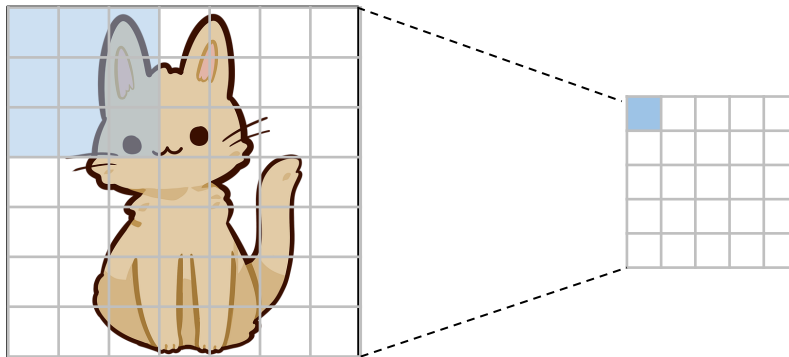
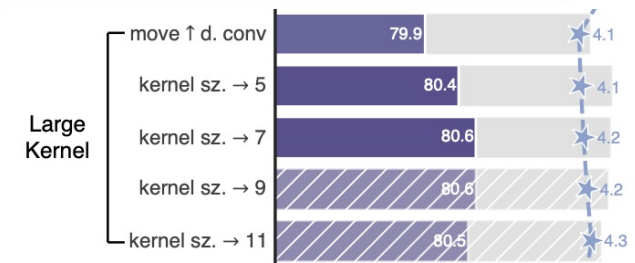


Hybrid CNNs

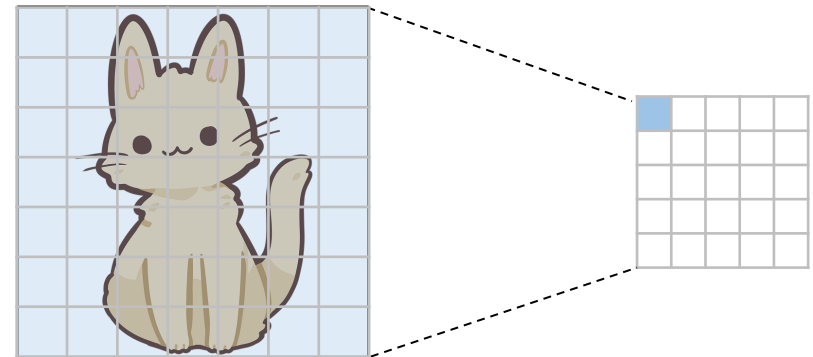
❖ 4. Large kernel

① 모든 블록에서 필터 크기를 키워 ViT와 같이 수용 영역을 넓힘

: 3 x 3 depthwise convolution → 7 x 7 depthwise convolution



3 x 3 convolution

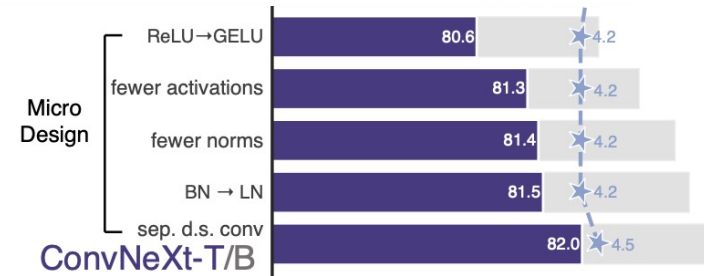


7 x 7 convolution

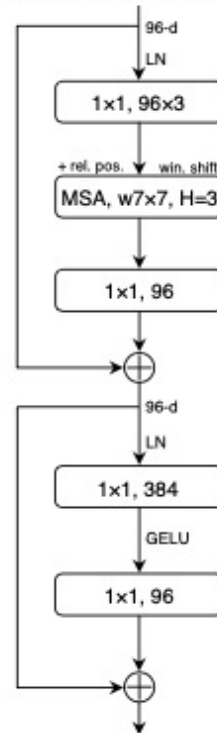
Hybrid CNNs

❖ 5. Micro design

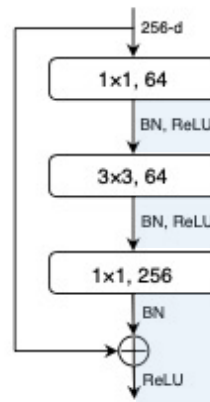
: 활성화 함수(ReLU → GELU), 정규화 방식(Batch norm → Layer norm) 등을 Swin Transformer 방식으로 변경함



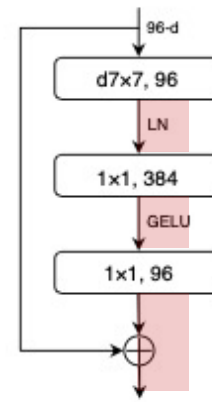
Swin Transformer Block



ResNet Block



ConvNeXt Block



Hybrid CNNs

❖ 학습 방식

- Transformer(DeiT, Swin Transformer) 학습 기법과 유사하게 설정
 - training epoch 300
 - AdamW optimizer
 - data augmentation : Mixup, Cutmix, RandAugment, Random Erasing
 - regularization : Stochastic depth, Label smoothing

Hybrid CNNs

❖ 실험 결과

- 제안한 기본 ConvNeXt(82.1%)로 Swin Transformer 성능(81.3%)을 넘어섬
- 입력 이미지 크기를 늘릴 경우 최고 85.5% 성능 기록함

model	image size	#param.	FLOPs	throughput (image / s)	IN-1K top-1 acc.
ImageNet-1K trained models					
• RegNetY-4G [51]	224 ²	21M	4.0G	1156.7	80.0
• RegNetY-8G [51]	224 ²	39M	8.0G	591.6	81.7
• RegNetY-16G [51]	224 ²	84M	16.0G	334.7	82.9
• EffNet-B3 [67]	300 ²	12M	1.8G	732.1	81.6
• EffNet-B4 [67]	380 ²	19M	4.2G	349.4	82.9
• EffNet-B5 [67]	456 ²	30M	9.9G	169.1	83.6
• EffNet-B6 [67]	528 ²	43M	19.0G	96.9	84.0
• EffNet-B7 [67]	600 ²	66M	37.0G	55.1	84.3
○ DeiT-S [68]	224 ²	22M	4.6G	978.5	79.8
○ DeiT-B [68]	224 ²	87M	17.6G	302.1	81.8
○ Swin-T	224 ²	28M	4.5G	757.9	81.3
• ConvNeXt-T	224 ²	29M	4.5G	774.7	82.1
○ Swin-S	224 ²	50M	8.7G	436.7	83.0
• ConvNeXt-S	224 ²	50M	8.7G	447.1	83.1
○ Swin-B	224 ²	88M	15.4G	286.6	83.5
• ConvNeXt-B	224 ²	89M	15.4G	292.1	83.8
○ Swin-B	384 ²	88M	47.1G	85.1	84.5
• ConvNeXt-B	384 ²	89M	45.0G	95.7	85.1
• ConvNeXt-L	224 ²	198M	34.4G	146.8	84.3
• ConvNeXt-L	384 ²	198M	101.0G	50.4	85.5

Conclusions

- ❖ CNN은 뛰어난 연산 효율성과 이미지 특징 추출에 적합한 inductive bias로 오랜 기간 컴퓨터 비전 과제를 위한 백본 네트워크로 활용됨
- ❖ Vision transformer 구조는 global self-attention을 기반으로 여러 비전 과제에 도입되고 있지만, 대규모 학습 데이터셋을 필요로 하는 등 제약 사항이 존재함
- ❖ Vision transformer 모델 학습에 쓰인 여러 기법을 활용해 CNN 모델의 성능을 높일 수 있음
 - Depthwise convolution
 - Regularization : Stochastic depth, Label smoothing 등

감사합니다.

