Aggregated Residual Transformations for Deep Neural Networks : ResNeXt

Xie, Saining, et al. "Aggregated residual transformations for deep neural networks." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

ISL
안재원

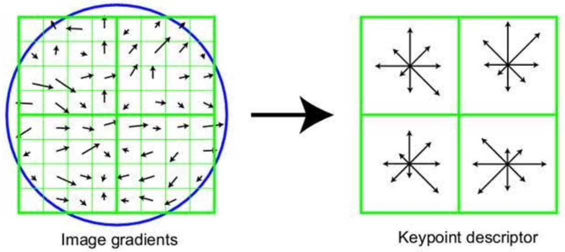
CONTENTS

Introduction

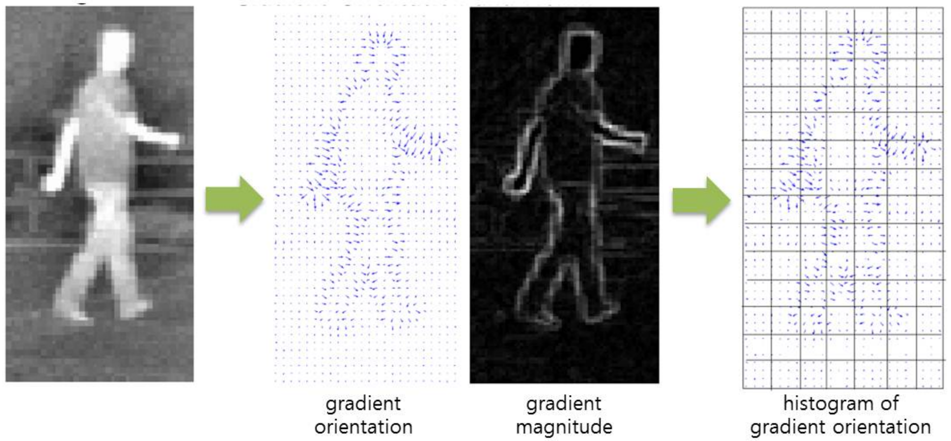
Transition form ‘Feature engineering’ to ‘Network engineering’

- Hand-designed(made) feature.

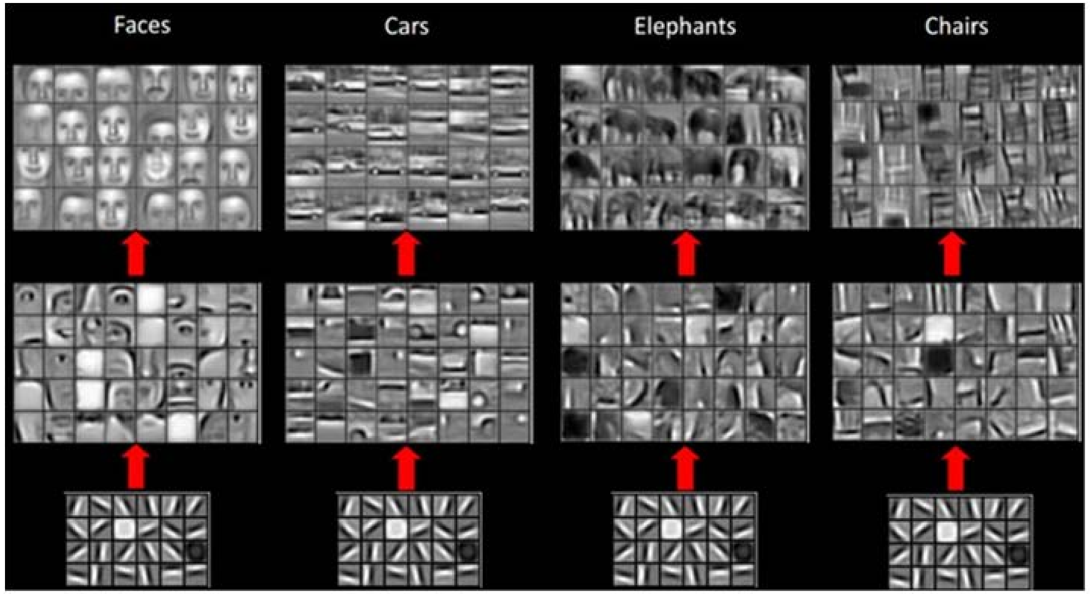
- SIFT(Scale Invariant Feature Transform)



- HOG(Histogram of Oriented Gradient)



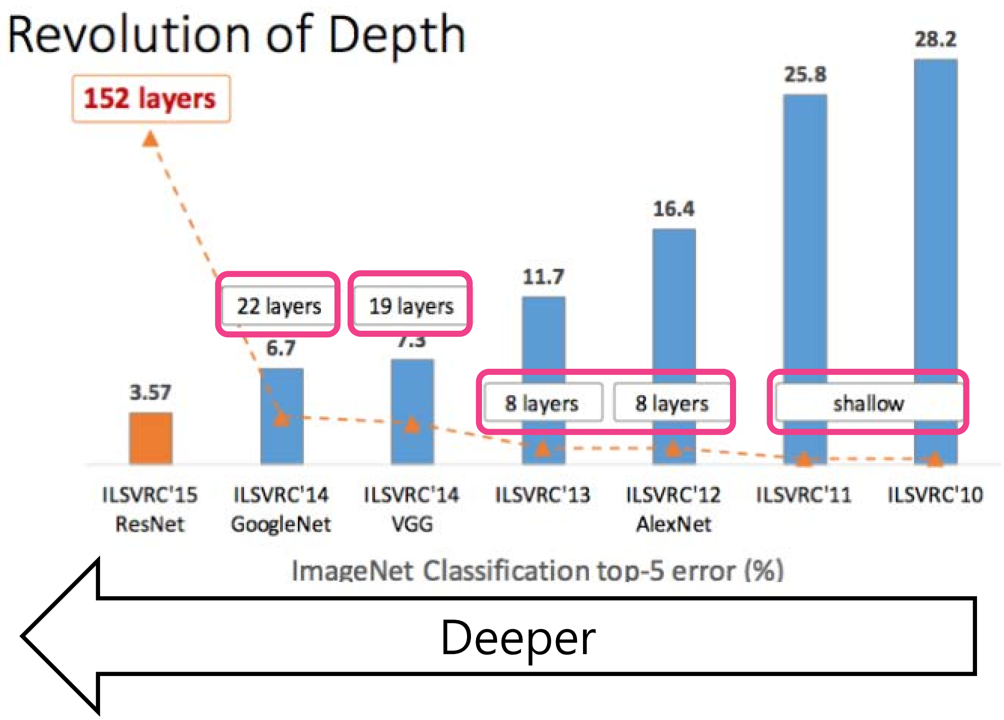
- Features learning by neural networks



Introduction

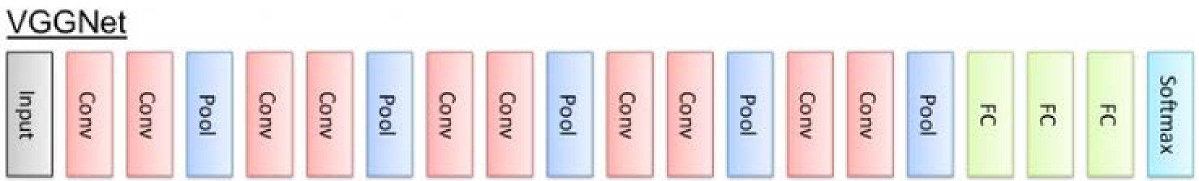
VGG-nets/ResNets & Inception model(GoogLeNets)

- Stacking building blocks



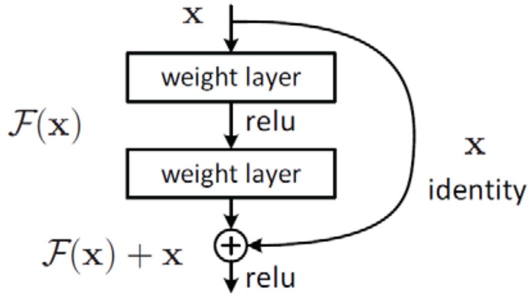
- 깊으면 깊을 수록 더 좋은 성능을 보임

VGGNet



- 무조건 깊을 수록 좋은 것은 아니다.

- ResNets



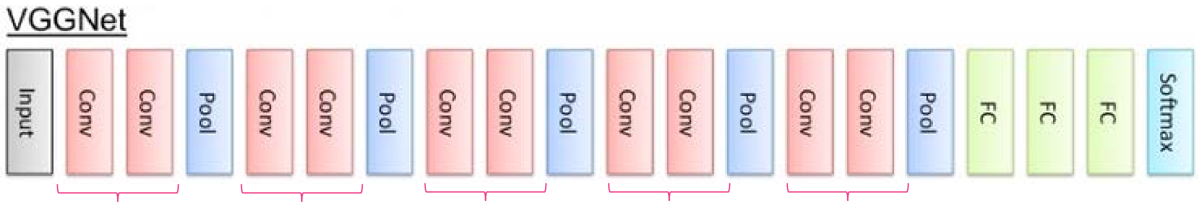
- 어떻게 깊어져야 하는가.

Introduction

VGG-nets/ResNets & Inception model(GoogLeNets)

- Block이 Network 목적에 적합한가?

  - 적절한 Layer가 사용되고 있는가?



1. Conv-Layer 에서 3x3 kernel을 사용했을 때?
2. Conv-Layer 에서 5x5 kernel을 사용했을 때?
3. Conv-Layer 에서 3x3 kernel과 5x5 kernel을 섞어서 사용했을 때?

→ 딥러닝 관련 논문들의 실험결과의 성능이 들쑥날쑥한 이유가 이 부분 때문이지 않을까…

- Overfitting의 문제

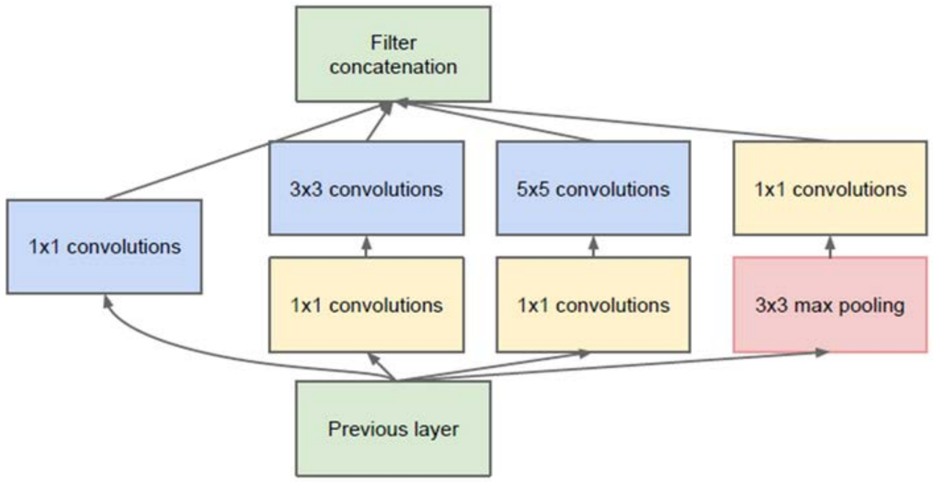
⇨ 간단한 구조의 사용으로 인한 자유로움은 떨어진다.

Introduction

VGG-nets/ResNets & Inception model(GoogLeNets)

- Split - transform - merge

  - Inception model



Split

1. 네트워크가 더 넓어지는(Wide) 효과
2. 네트워크 자유도 상승

Transform

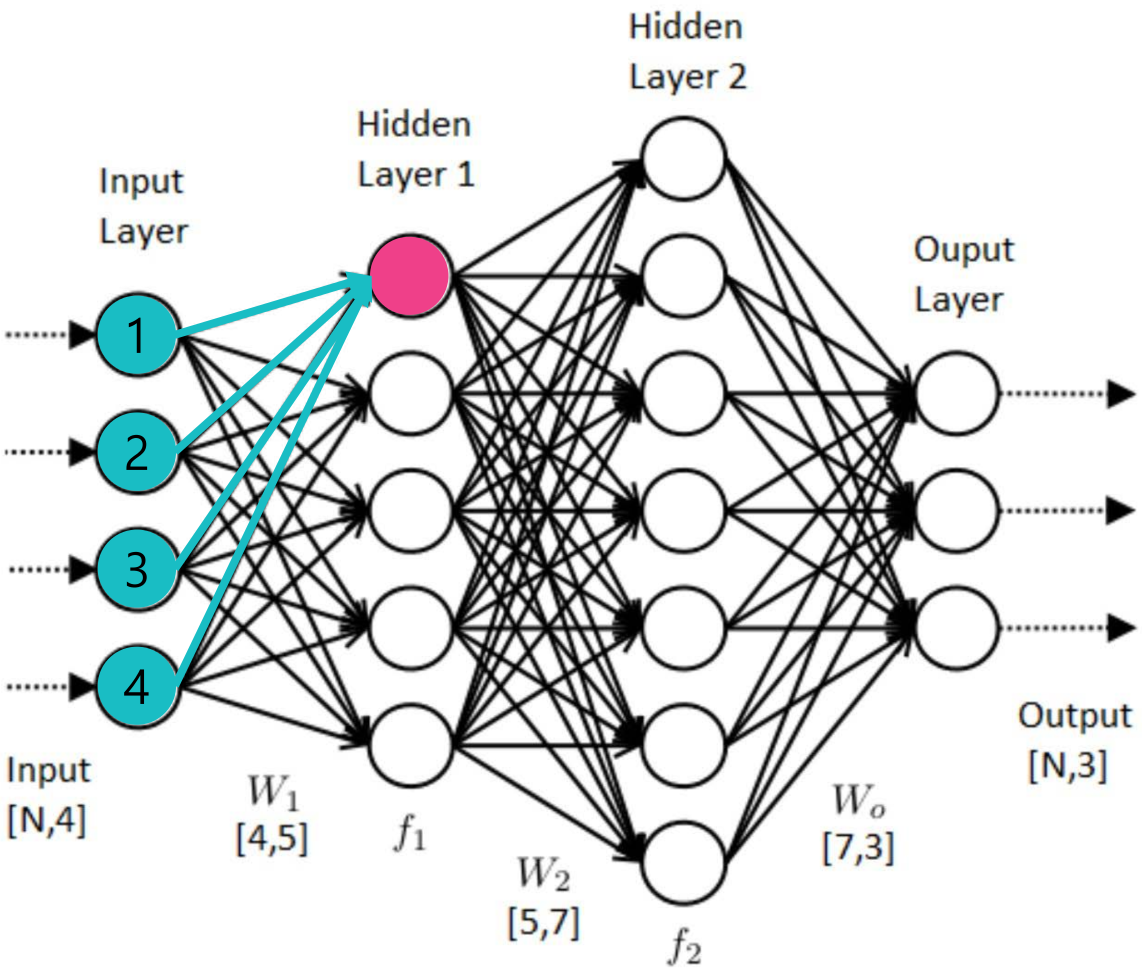
Merge

1. 복잡한 형태(다른 크기의 conv이 진행된다.)
2. 데이터에 따라 요구하는 Transform이 다를 수 있다.
3. 부족한 자유도를 극복하려 했지만, 한계가 있다.

- 자유도 높은(적응력 높은) 네트워크 구성이 필요하다.
- 더 깊은(Deep) 네트워크가 아닌 더 넓은(Wide) 네트워크를 구성.
- 즉, 더 넓고 더 자유로운 네트워크가 필요하다.

ResNeXt

Simple Neurons



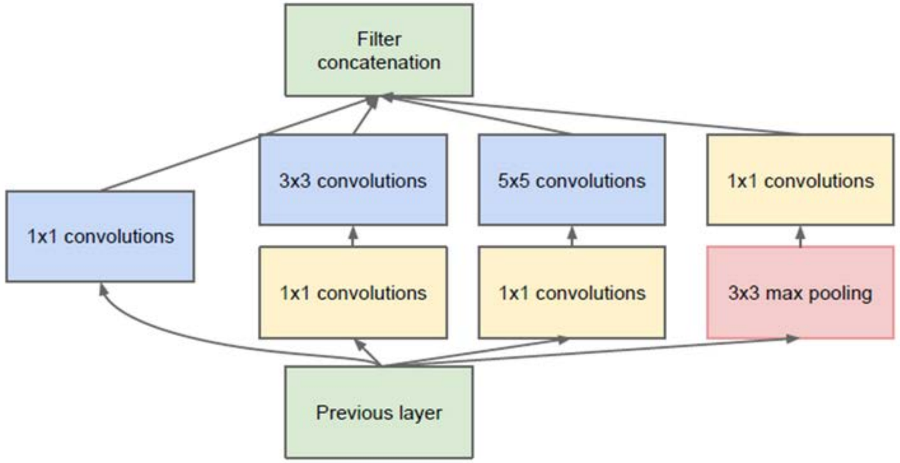
- 한 Neuron에 입력되는 값이 만들어지는 과정

$$1 = w_{1,1}x_1$$

$$i = w_{i,1}x_i$$

$$= \sum_{i=1}^{4} w_{i,1}x_i = F_1(X)$$

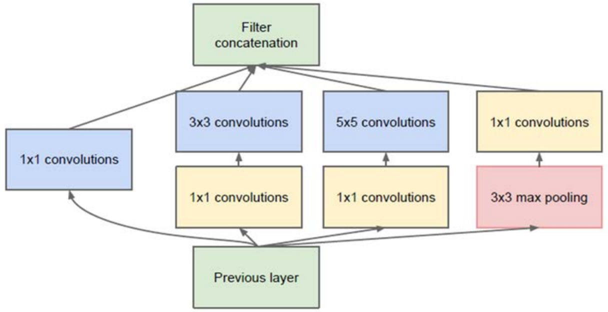
Inception Model과 Neuron의 유사성



ResNeXt

Split – Transform – Merge

- Inception model(Network in Network)



Split

Transform

Merge

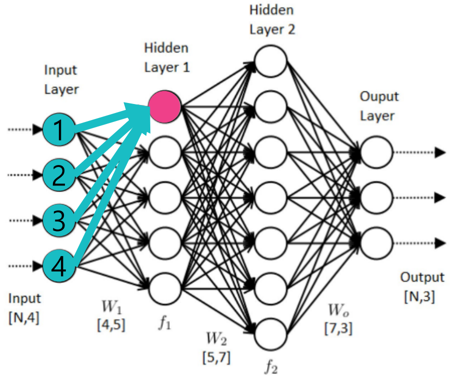
- Neuron

$$F_1(X) = \sum_{i=1}^{D} \underbrace{w_{i,1} x_i}$$

Split $[X => x_i]$

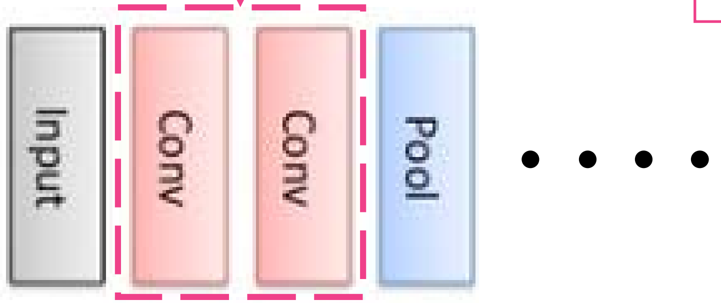
Merge

Transform



임의의 함수로 생각 할 수 있다.($\mathcal{T}_i(X)$)

- Network in 'Neuron'



- Aggregated transformations

$$\mathcal{F}_1(X) = \sum_{i=1}^{C} \mathcal{T}_i(X)$$

Cardinality : The size of set of transformation

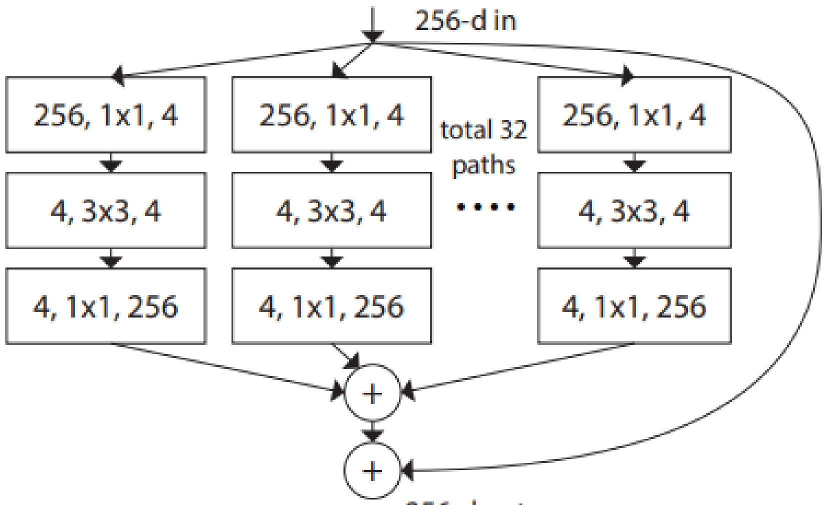
- Aggregated transformations with residual

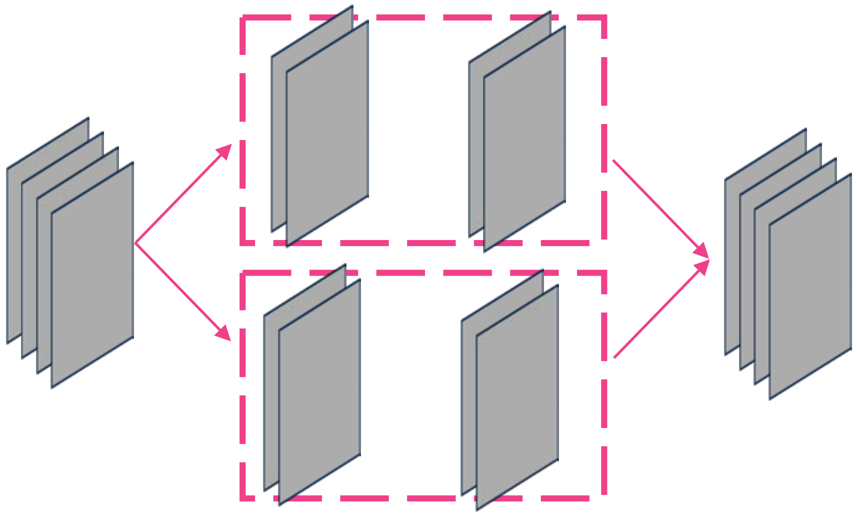
$$y_1(X) = X + \sum_{i=1}^{C} \mathcal{T}_i(X)$$

ResNeXt

ResNeXt

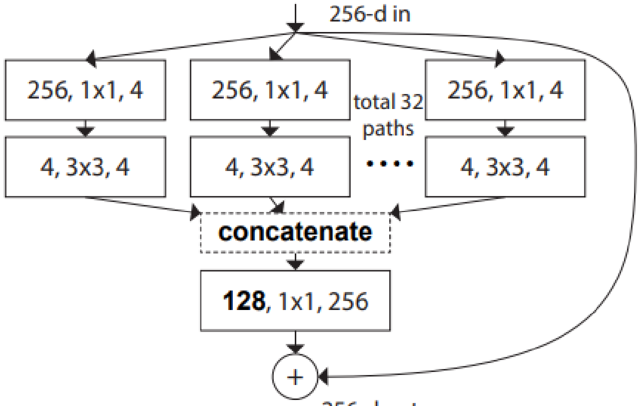
- ResNeXt model

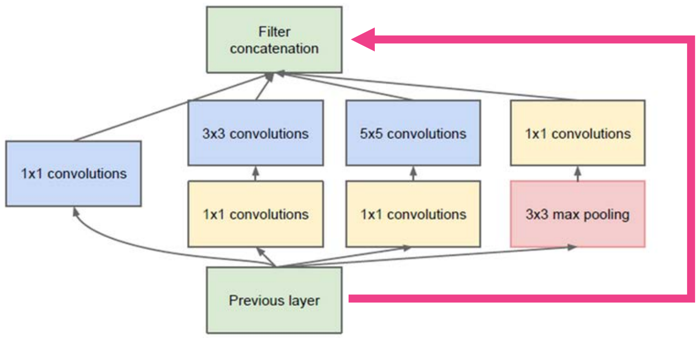




- Equivalent model I.(Similar to the Inception-ResNet)

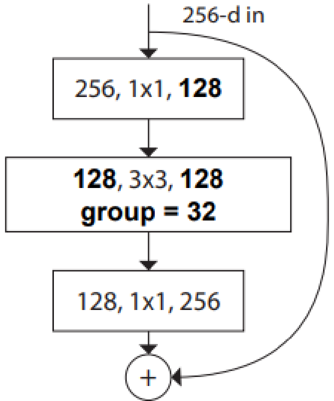
모든 split 에 다하여 같은 꼴의 Transform을 적용하는가?

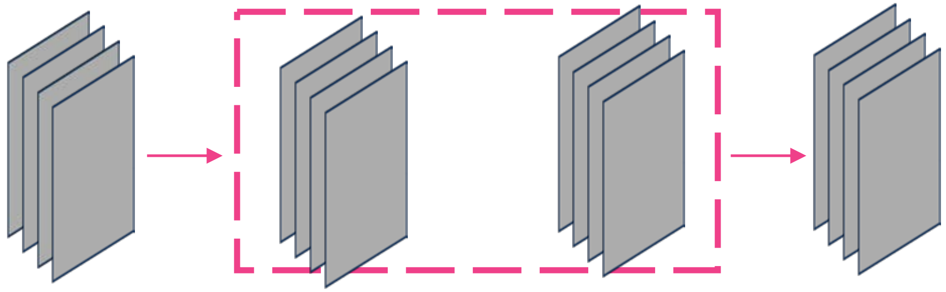




- Equivalent model II.(Similar to the ResNet)

Split간에 영향을 주는가?





Experiments

- ResNet과 ResNeXt 성능 비교.

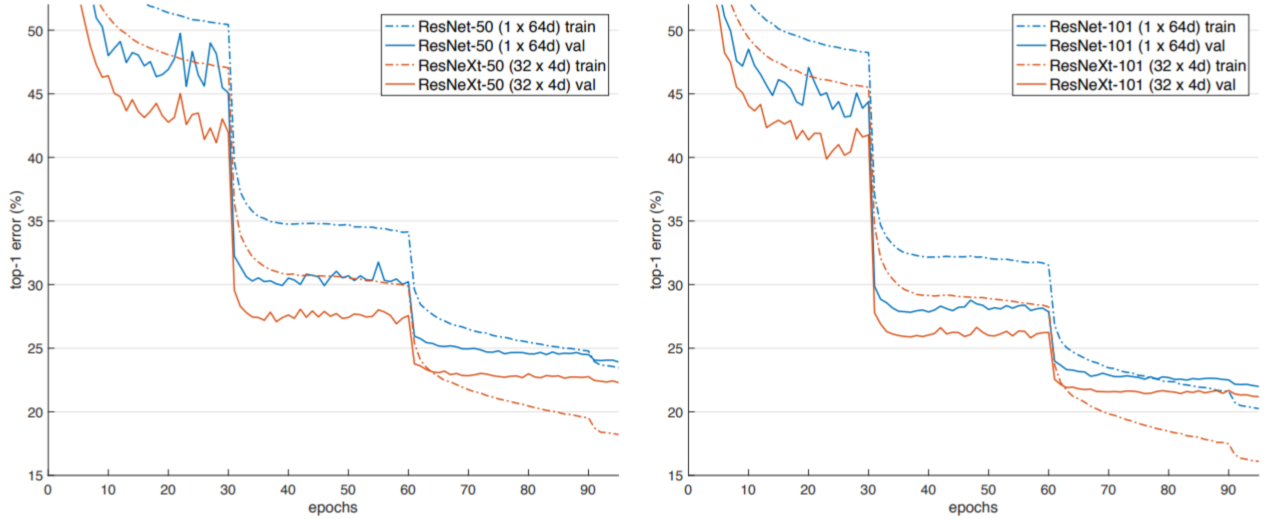


Figure 5. Training curves on ImageNet-1K. (**Left**): ResNet/ResNeXt-50 with preserved complexity (~4.1 billion FLOPs, ~25 million parameters); (**Right**): ResNet/ResNeXt-101 with preserved complexity (~7.8 billion FLOPs, ~44 million parameters).

- ResNeXt의 Cardinality에 따른 성능 비교.

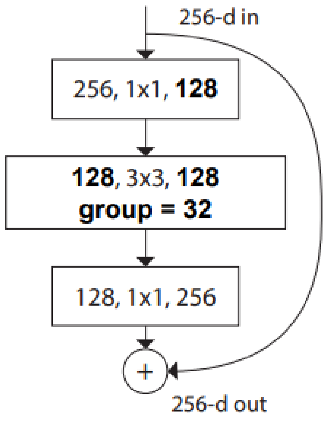
	setting	top-1 error (%)
ResNet-50	1 × 64d	23.9
ResNeXt-50	2 × 40d	23.0
ResNeXt-50	4 × 24d	22.6
ResNeXt-50	8 × 14d	22.3
ResNeXt-50	32 × 4d	**22.2**
ResNet-101	1 × 64d	22.0
ResNeXt-101	2 × 40d	21.7
ResNeXt-101	4 × 24d	21.4
ResNeXt-101	8 × 14d	21.3
ResNeXt-101	32 × 4d	**21.2**

Experiments

- Wider VS Cardinality I.

	setting	top-1 err (%)	top-5 err (%)
1× complexity references:			
ResNet-101	1 × 64d	22.0	6.0
ResNeXt-101	32 × 4d	21.2	5.6
2× complexity models follow:			
ResNet-**200** [15]	1 × 64d	21.7	5.8
ResNet-101, wider	1 × **100**d	21.3	5.7
ResNeXt-101	**2** × 64d	20.7	5.5
ResNeXt-101	**64** × 4d	**20.4**	**5.3**

2번 Equivalent



- Wider VS Cardinality II.

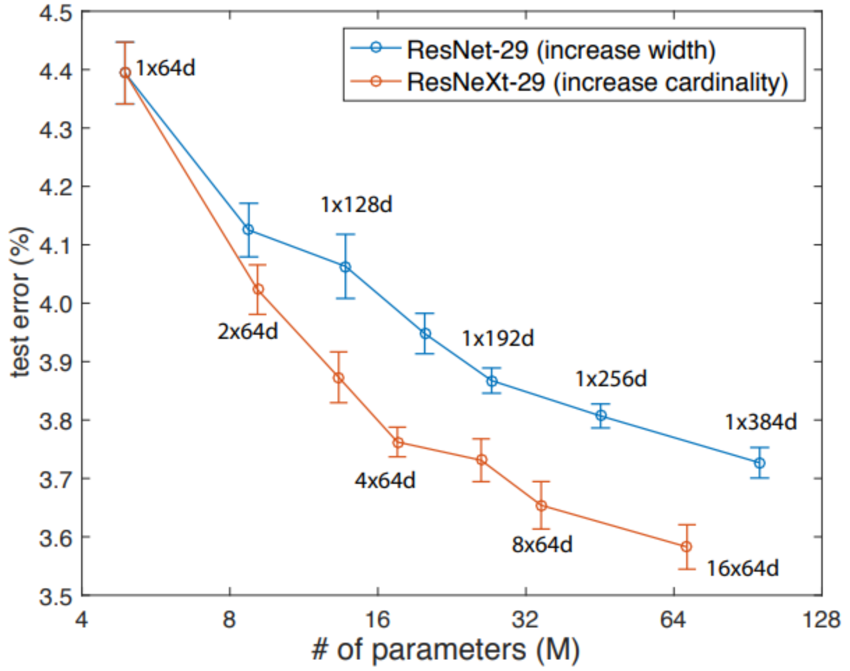


Figure 7. Test error *vs*. model size on CIFAR-10. The results are computed with 10 runs, shown with standard error bars. The labels show the settings of the templates.

실험 결과의 모든 내용이 성능 향상을 보임.

Q & A