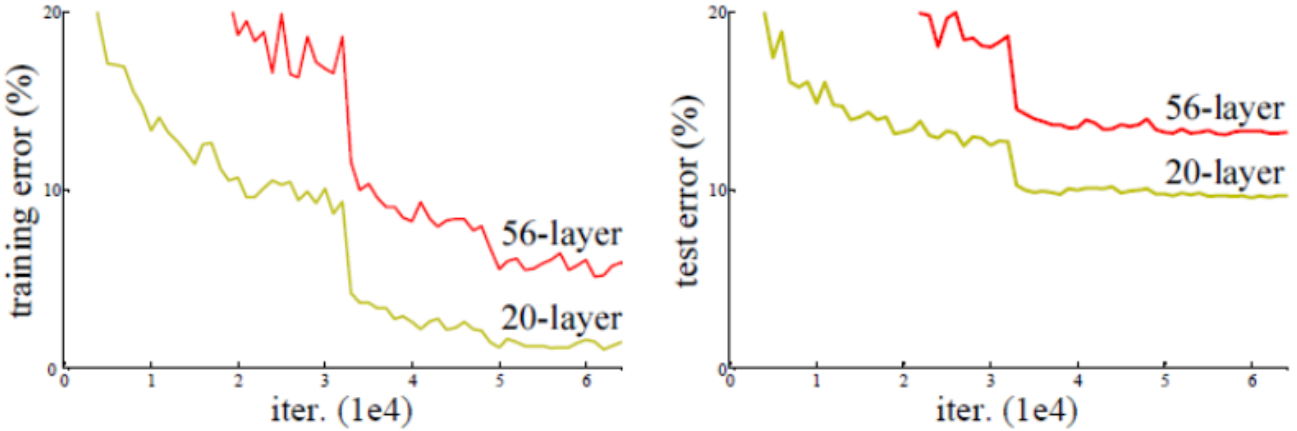
ResNet
&
ResNeXt

KU-big 13기 오원석

ResNet

Worse performance in deep layer



Causes?
1. gradient vanishing/exploding
2. overfitting

Deployment

Considering the phenomenon of **degradation** where the error of training data increases as the layer thickens, the conclusion is that learning itself is difficult when the layer thickens.

Hypotosis

Adding meaningless layers, such as **identity layers**, does a shallow model should still have the same or higher performance compared to a shallow model.

Structure

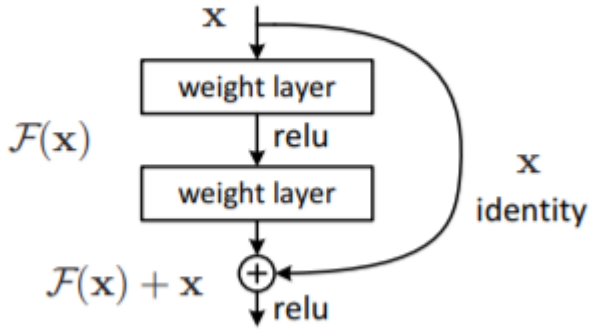
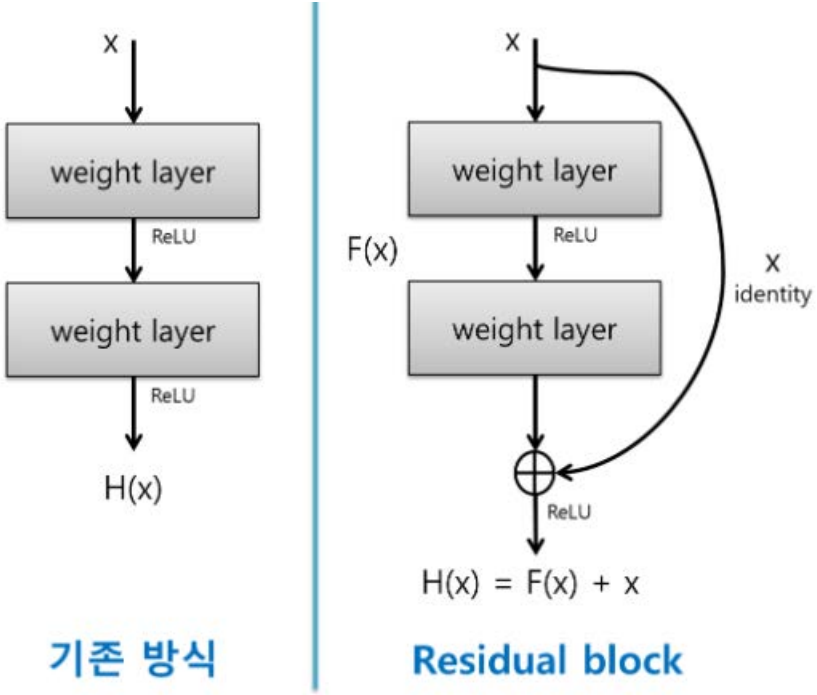


Figure 2. Residual learning: a building block.

Residuals = F(x) = H(x) – x -> H(x) = F(x) + x(shortcut): Get the previously learned x and learn about the remaining parts of F(x) to learn fast / high performance

Comparison


x
weight layer
ReLU
weight layer
ReLU
H(x)
기존 방식
x
weight layer
F(x)
ReLU
weight layer
x
identity
ReLU
H(x) = F(x) + x
Residual block

Performance

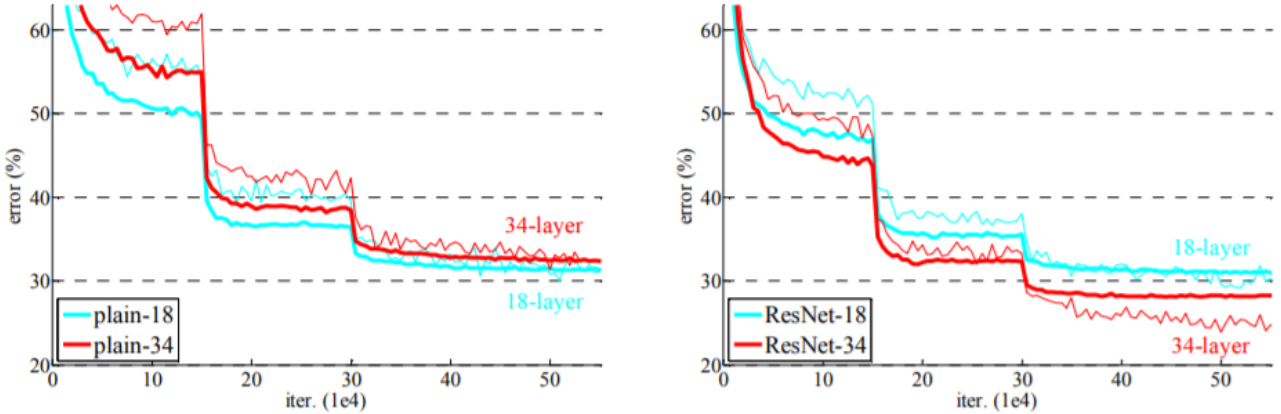
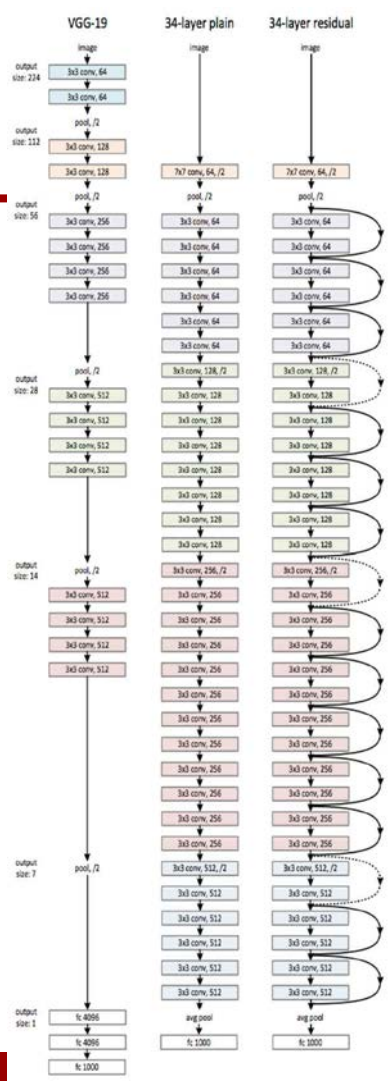


Figure 4. Training on **ImageNet**. Thin curves denote training error, and bold curves denote validation error of the center crops. Left: plain networks of 18 and 34 layers. Right: ResNets of 18 and 34 layers. In this plot, the residual networks have no extra parameter compared to their plain counterparts.

Resnet



ppt제목

KU-BIG

ResNeXt

Structure

ResNeXt also stacks the same layers, the way VGG and ResNet use them. Additionally, it uses a **split transform merge method** that splits one input in several directions, similar to that used in the input.
What differs from Inception-ResNet is that it has the same layer configuration for each path. This is called a **grouped convolution**

Structure

Differences from Resnet: cardinality

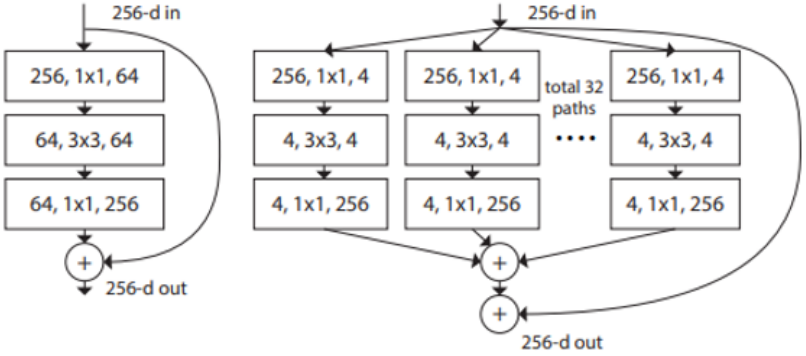


Figure 1. **Left**: A block of ResNet [14]. **Right**: A block of ResNeXt with cardinality = 32, with roughly the same complexity. A layer is shown as (# in channels, filter size, # out channels).

Configuration in paper

The configuration of ResNet-50 and ResNeXt-50 written in the paper

On ResNet, you can see that one convolution is made into a deep channel, while on ResNeXt, it is slightly deeper, but with 32 group convolution, you can see a significant reduction in the amount of computation.

stage	output	ResNet-50	**ResNeXt-50 (32×4d)**
conv1	112×112	7×7, 64, stride 2	7×7, 64, stride 2
conv2	56×56	3×3 max pool, stride 2 $\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	3×3 max pool, stride 2 $\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128, C=32 \\ 1\times1, 256 \end{bmatrix}\times3$
conv3	28×28	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256, C=32 \\ 1\times1, 512 \end{bmatrix}\times4$
conv4	14×14	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512, C=32 \\ 1\times1, 1024 \end{bmatrix}\times6$
conv5	7×7	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 1024 \\ 3\times3, 1024, C=32 \\ 1\times1, 2048 \end{bmatrix}\times3$
	1×1	global average pool 1000-d fc, softmax	global average pool 1000-d fc, softmax
# params.		**25.5**×10^6	**25.0**×10^6
FLOPs		**4.1**×10^9	**4.2**×10^9

Table 1. (**Left**) ResNet-50. (**Right**) ResNeXt-50 with a 32×4d template (using the reformulation in Fig. 3(c)). Inside the brackets are the shape of a residual block, and outside the brackets is the number of stacked blocks on a stage. "C=32" suggests grouped convolutions [24] with 32 groups. *The numbers of parameters and FLOPs are similar between these two models.*