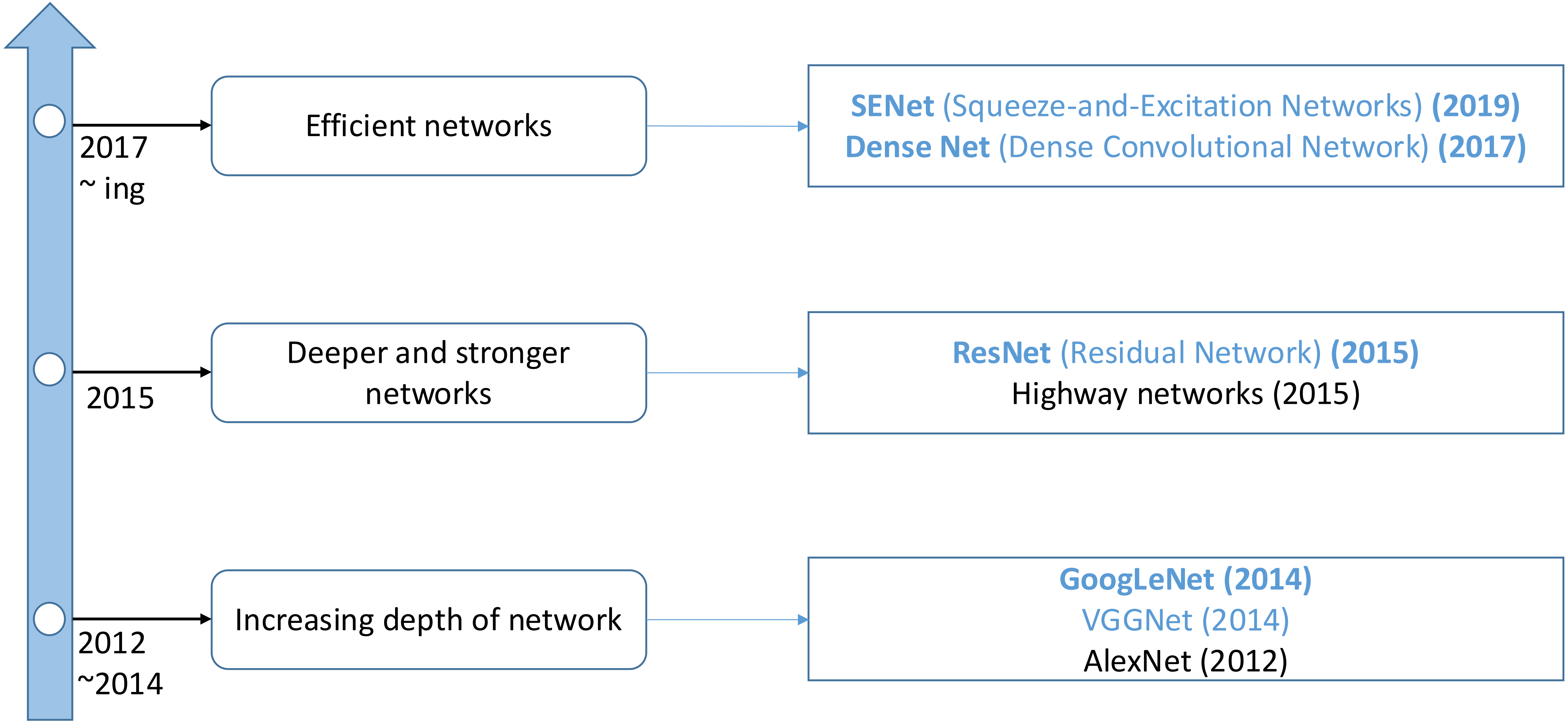
Deep Learning Network Model

2. CNN Architectures

2022. 02. 24 Suyeon Park

1. CNN Architectures



2. Increasing depth of network

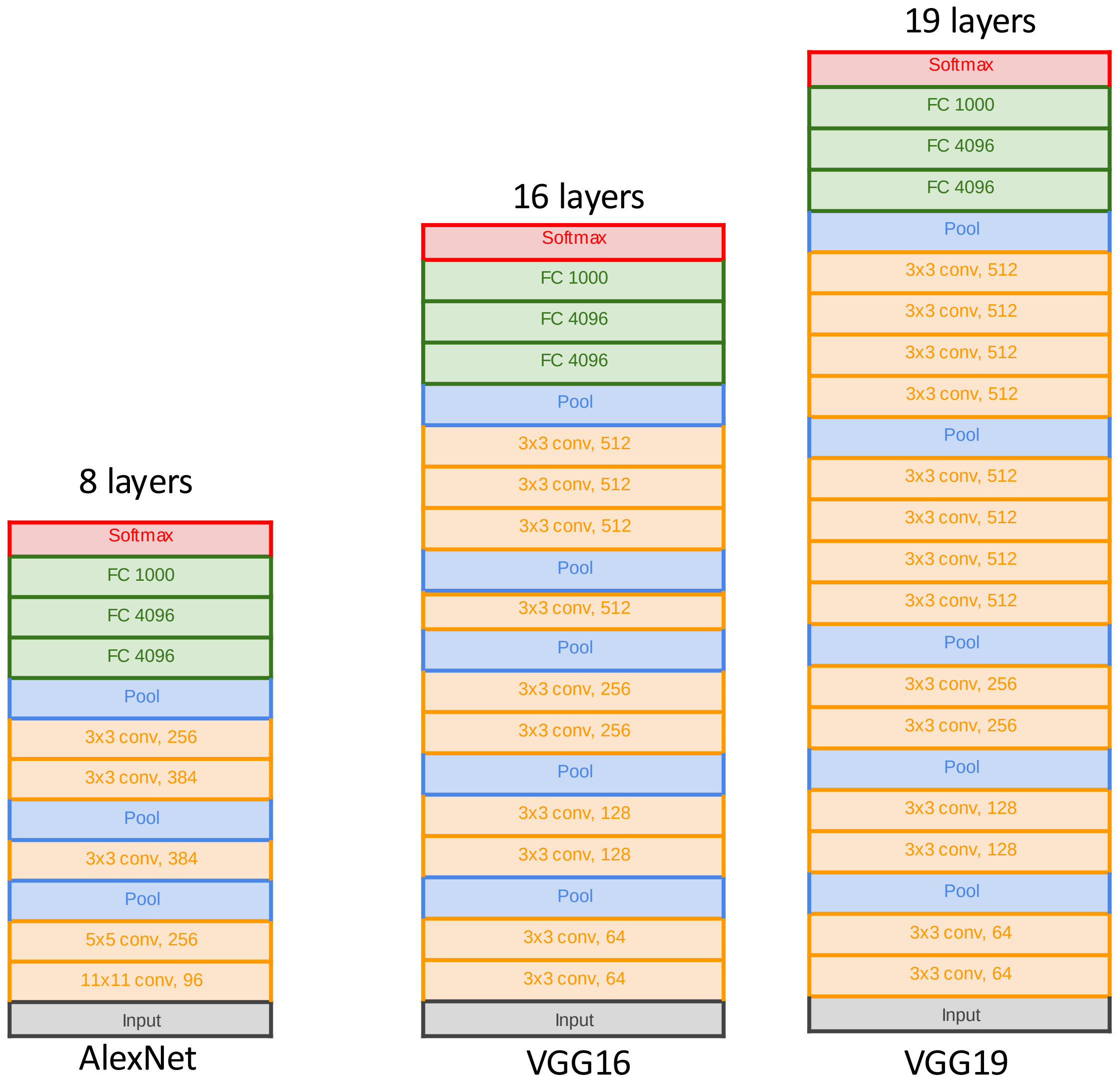
2.1. VGGNet (2014)

Small filters, Deeper networks

16-19 layers (VGG16, VGG19)

Only 3x3 CONV stride 1, pad 1
2x2 MAX POOL stride2

-> 7.3% top 5 error in ILSVRC'14



2. Increasing depth of network

2.1. VGGNet (2014)

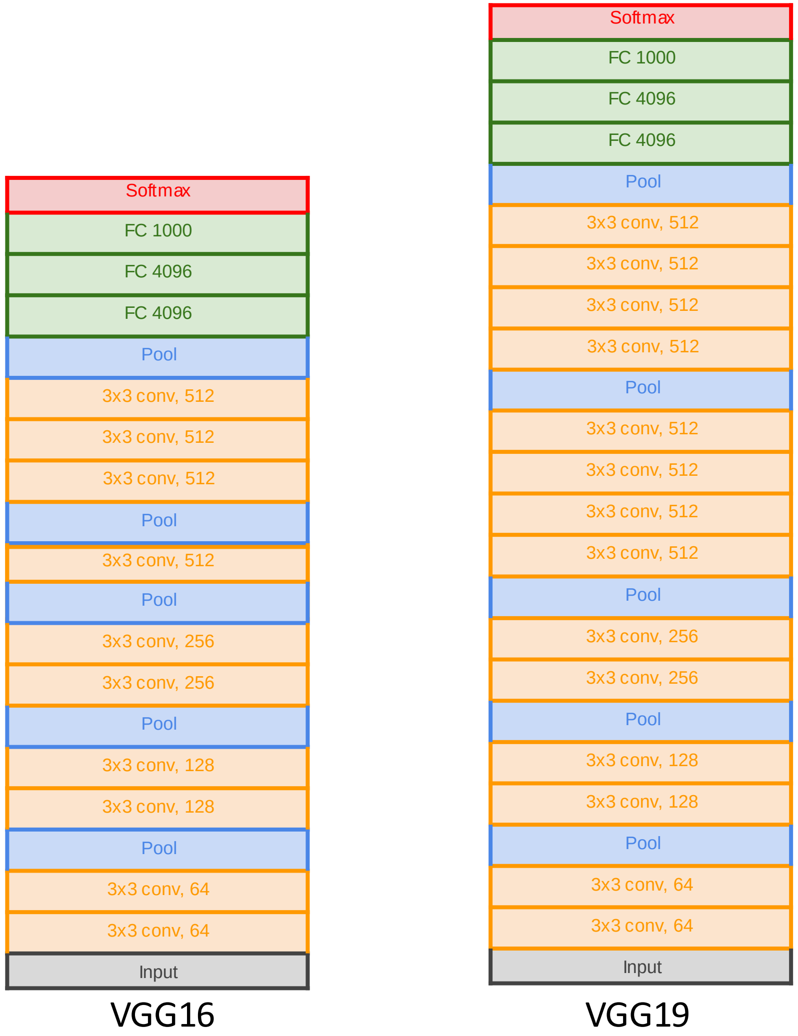
1. Use 3x3 Conv (stride 1)

Stack of three 3x3 Conv (stride 1) layers

II

one 7x7 conv layer

-> Same **effective receptive field**



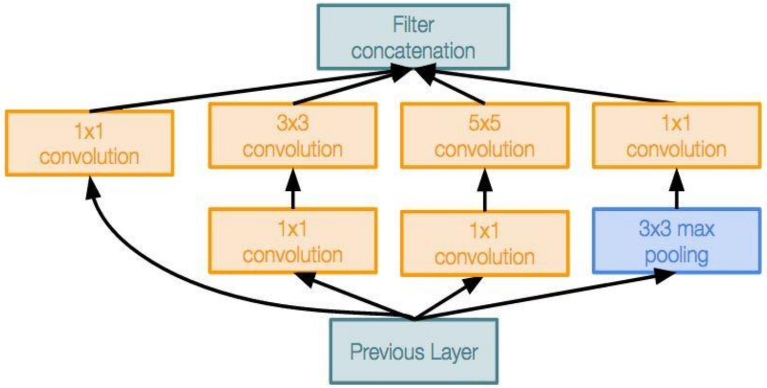
2. Increasing depth of network

2.2. GoogLeNet (2014)

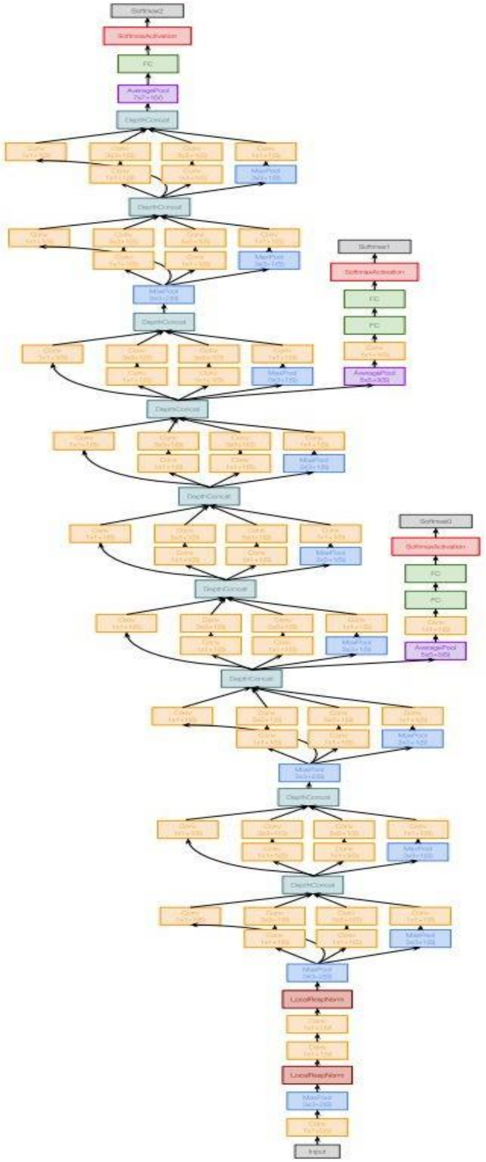
Deeper networks, with computational efficiency

Efficient ‘Inception’ module

22 layers
Only 5 million parameters
ILSVRC’14 classification winner (6.7% top 5 error)



Inception module

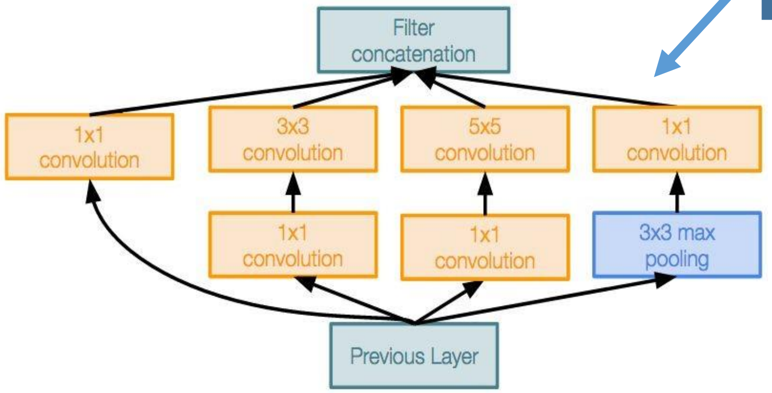


2. Increasing depth of network

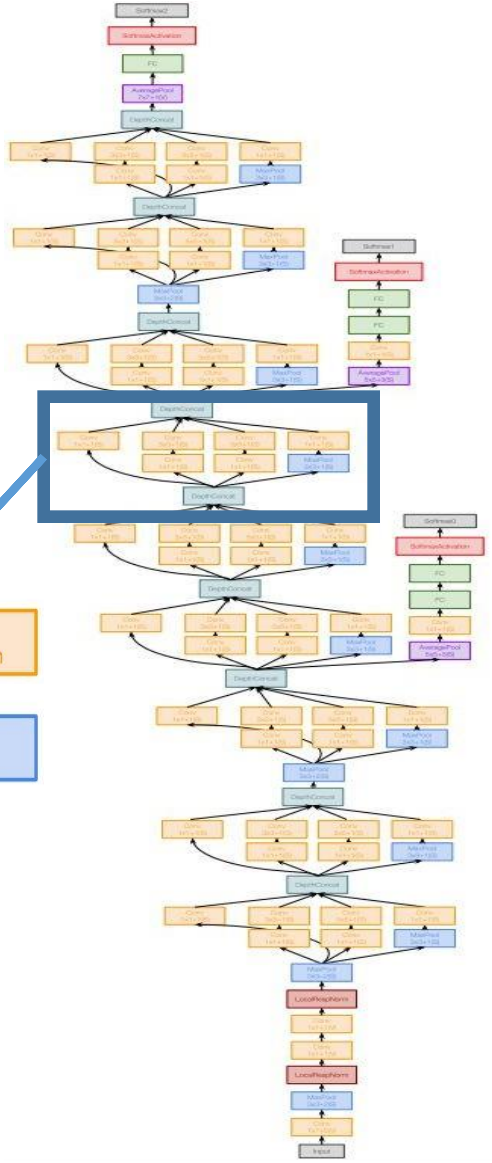
2.2. GoogLeNet (2014)

Neural networks perform well -> sparsity
In computer operations, the operation matrix -> Dense
to reduce useless resource loss.

GoogLeNet Architecture
Overall, reduces the connection -> sparsity
detailed matrix operations -> Dense



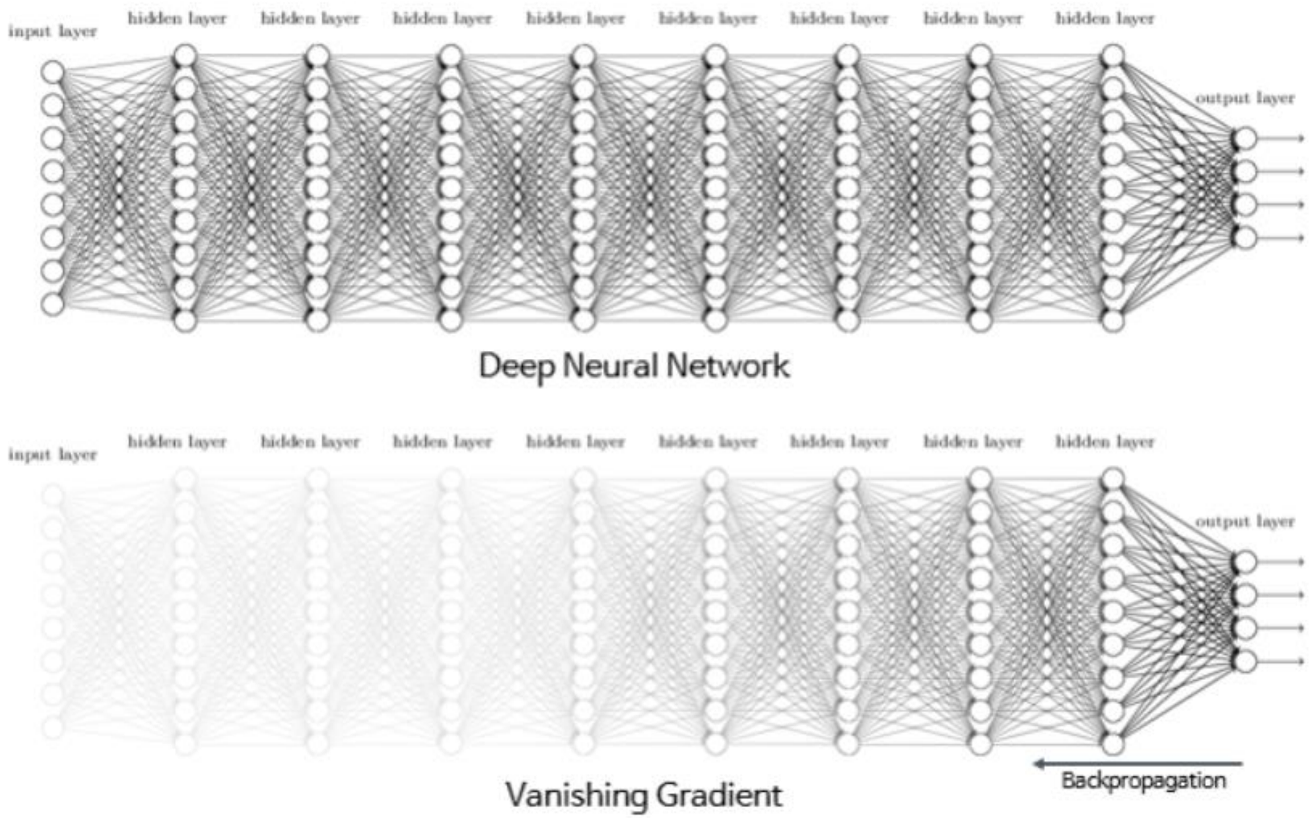
Inception module



3. Deeper and stronger networks

The Problem occurs increasing depth of network

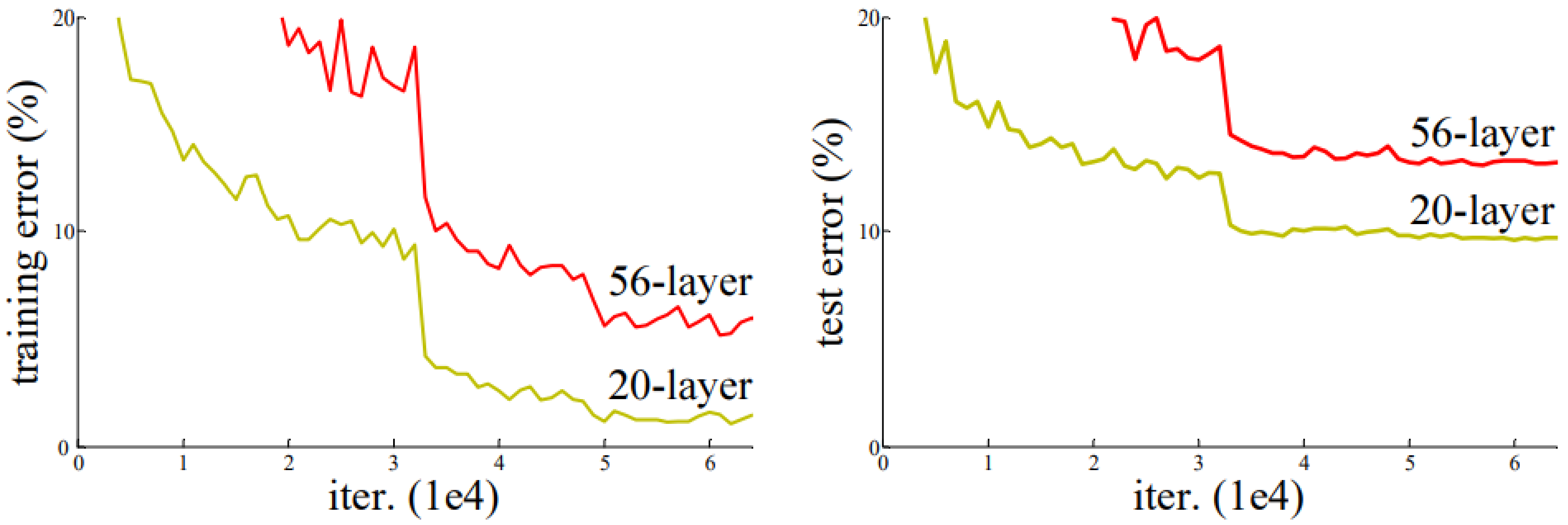
1. Problem of vanishing/exploding gradients



3. Deeper and stronger networks

The Problem occurs increasing depth of network

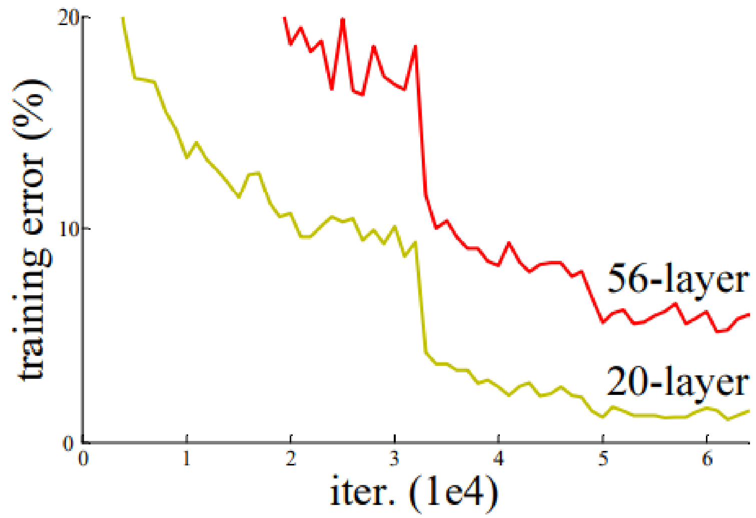
2. Adding more layers to a suitably deep model leads to **higher training error**

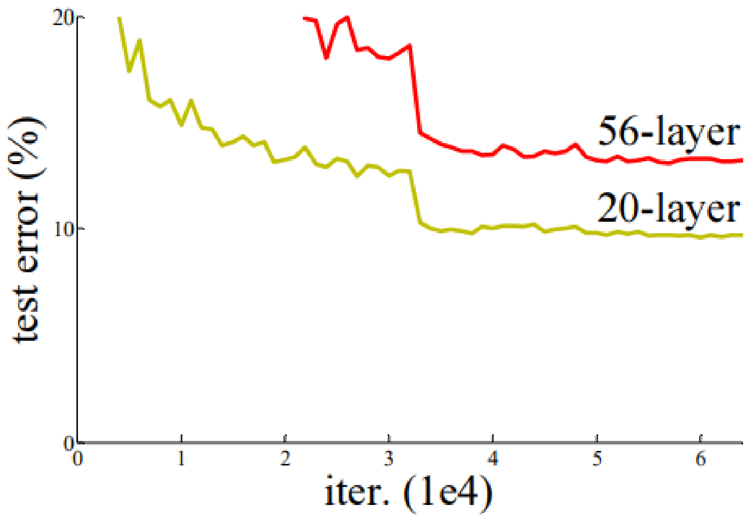


3. Deeper and stronger networks

The Problem occurs increasing depth of network

2. Adding more layers to a suitably deep model leads to **higher training error**





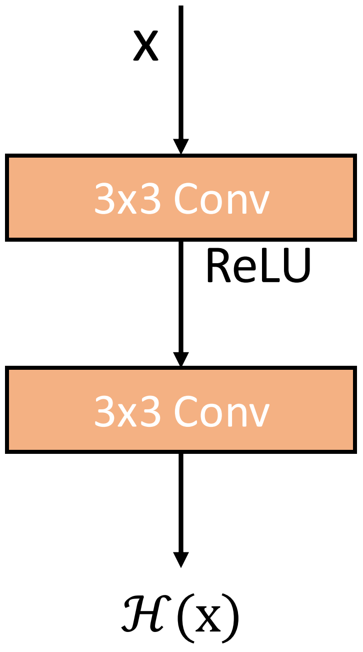
⇨ Degradation is not caused by overfitting

Hypothesis: the problem is an *optimization* problem, deeper models are harder to optimize

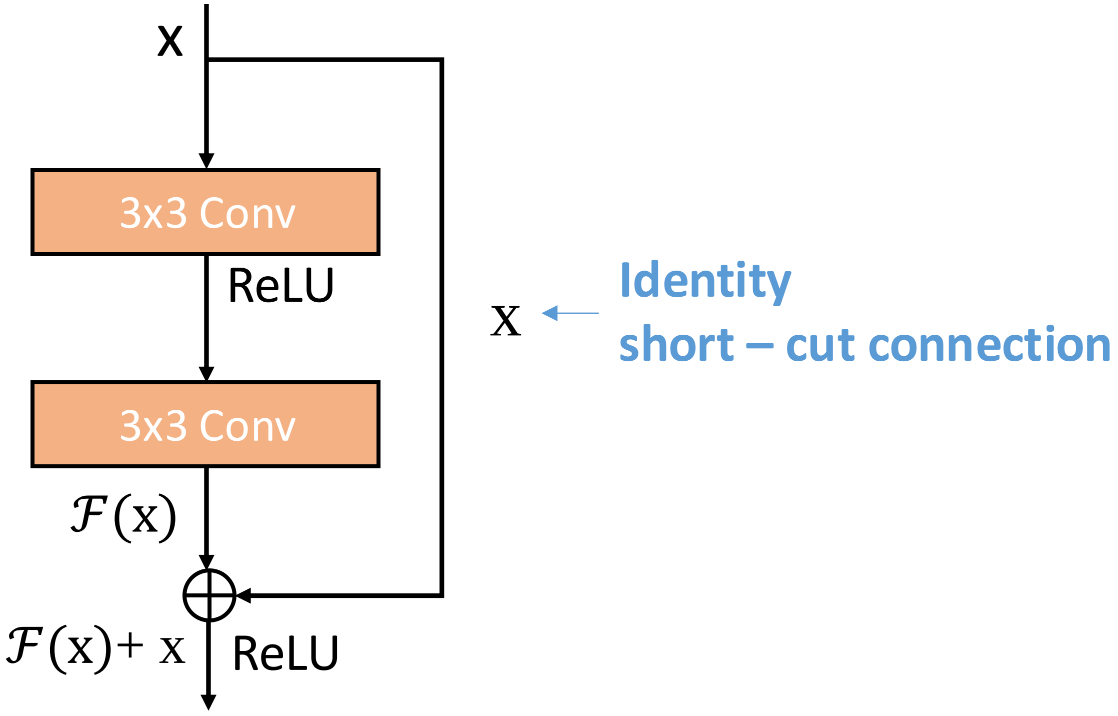
3. Deeper and stronger networks

3.1. ResNet (Residual Network)

Hypothesize : it is easier to optimize the residual mapping than to optimize the original



'Plane' layer



Residual block

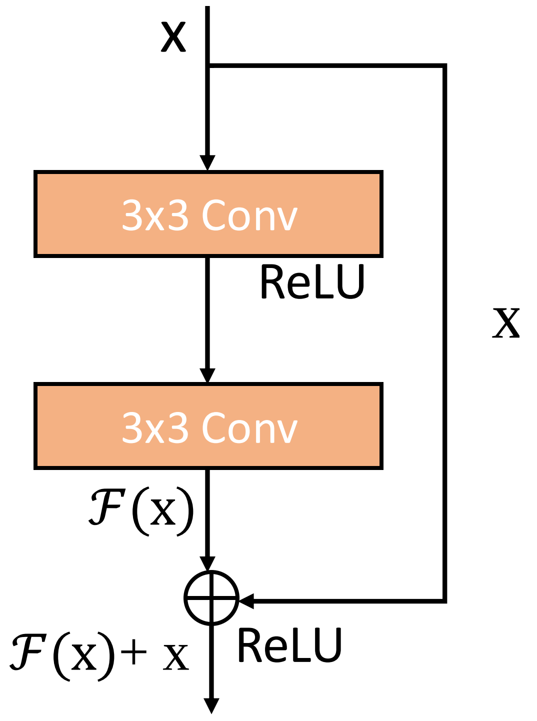
3.1. ResNet (Residual Network)

Full ResNet architecture:

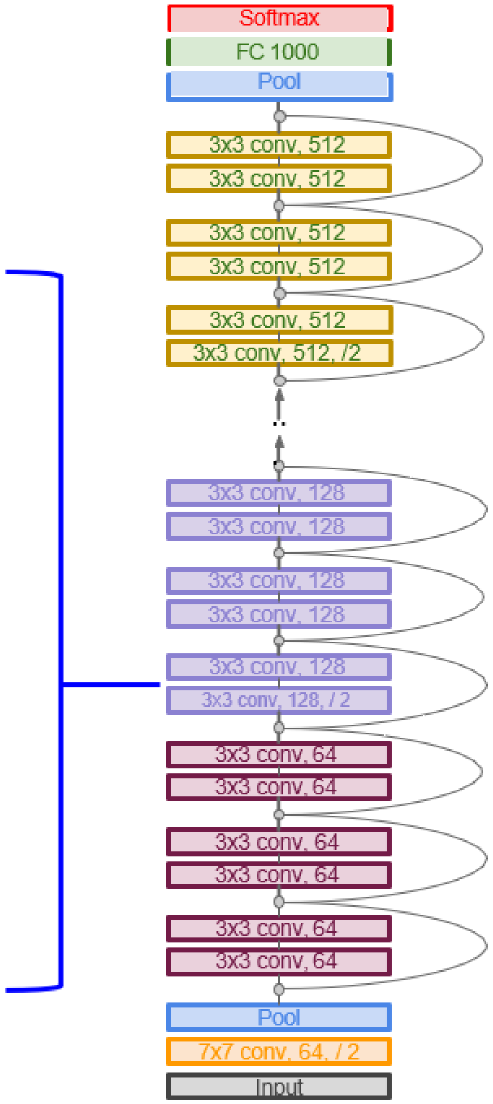
Stack residual blocks

Every residual block has two 3x3 conv layers

Total depths of 34, 50, 101, or 152 layers for ImageNet



Residual block

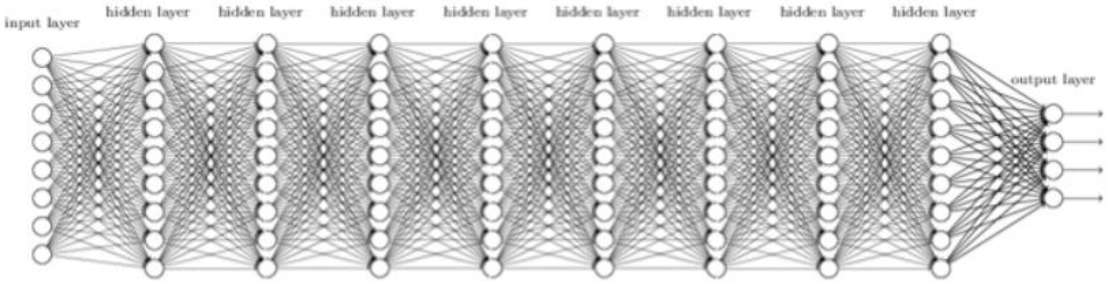


4. Efficient networks

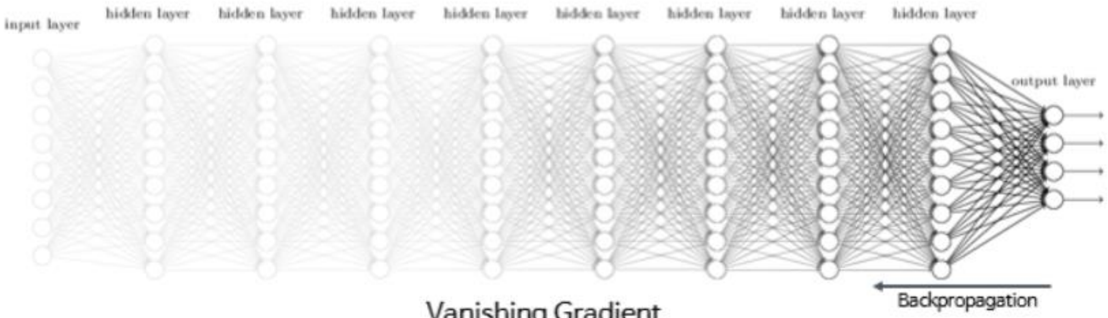
4.1. DenseNet (Dense Convolutional Network)

The Problem occurs increasing depth of network

Problem of vanishing/exploding gradients



Deep Neural Network



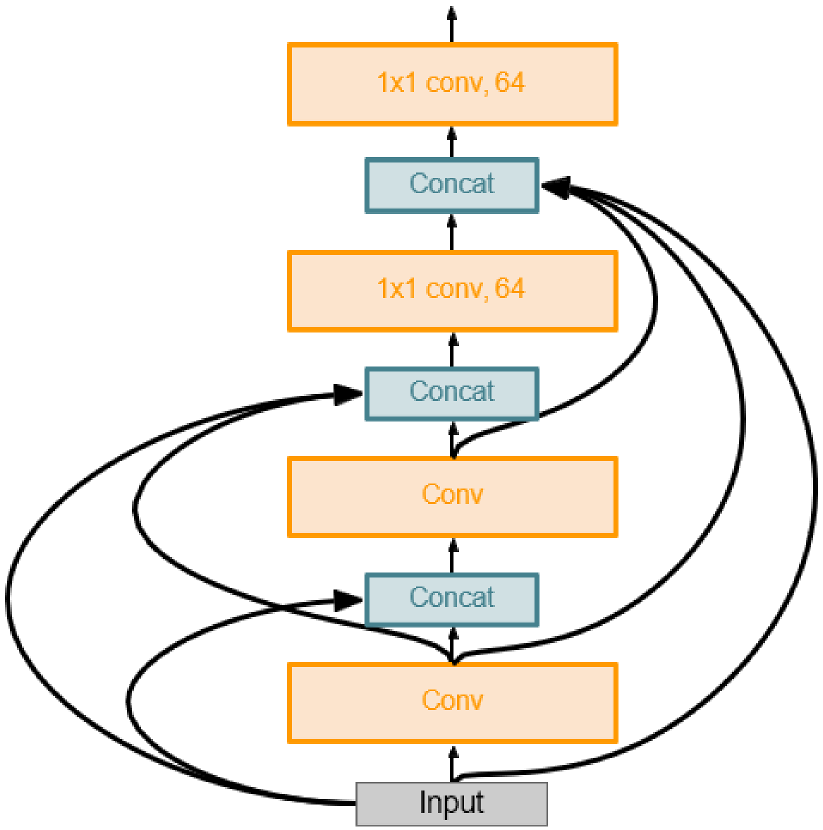
Vanishing Gradient

Solution : to create short path from early layers to later layers!

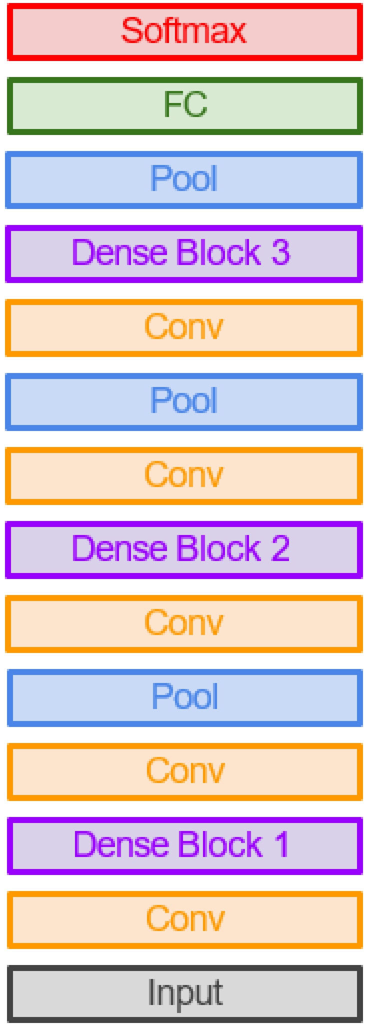
4.1. DenseNet

1. Connects each layers to each layers

- Alleviates vanishing gradient problem
- Strengthens feature propagation
- Encourages feature reuse
- Substantially reduce the number of parameters

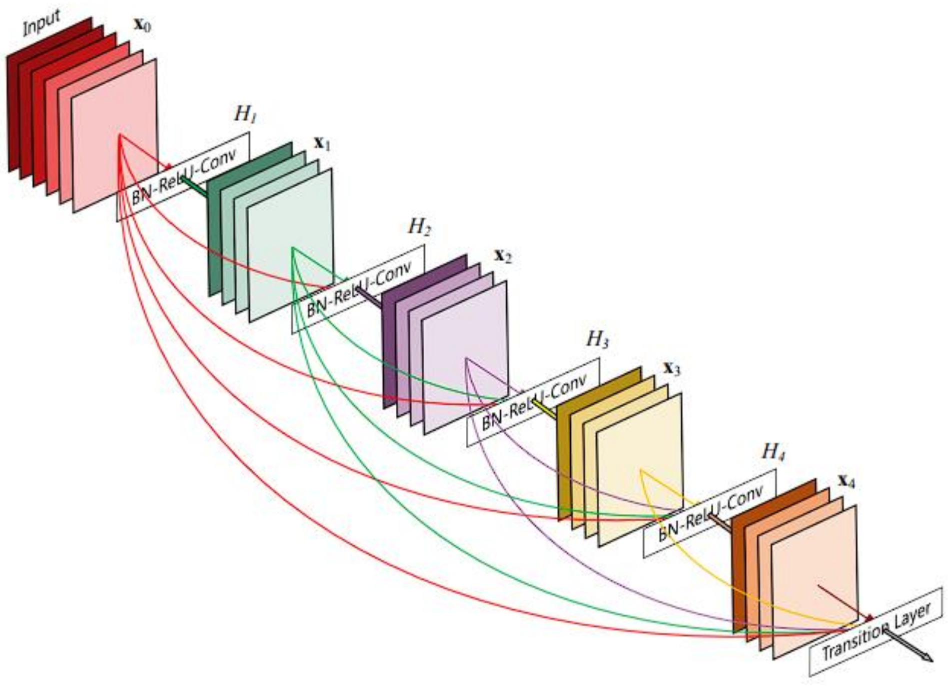


Dense Block



4.1. DenseNet

1. Dense connectivity(Connects each layers to each layers)



Dense Block

a single image x_0, output of the L th layer as x_L
non-linear transformation $H_L(\cdot)$

ResNets

$x_L = H_L(x_{L-1}) + x_{L-1}$ ⇨ combined by summation

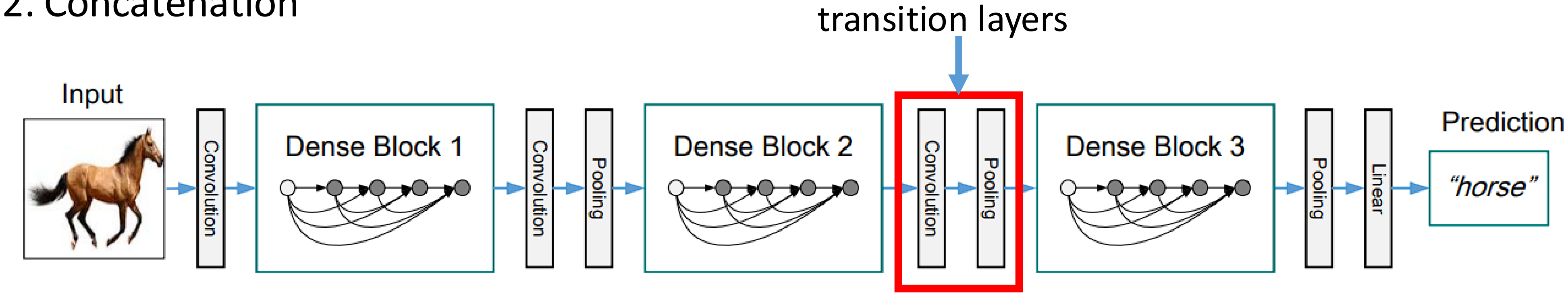
Dense connectivity

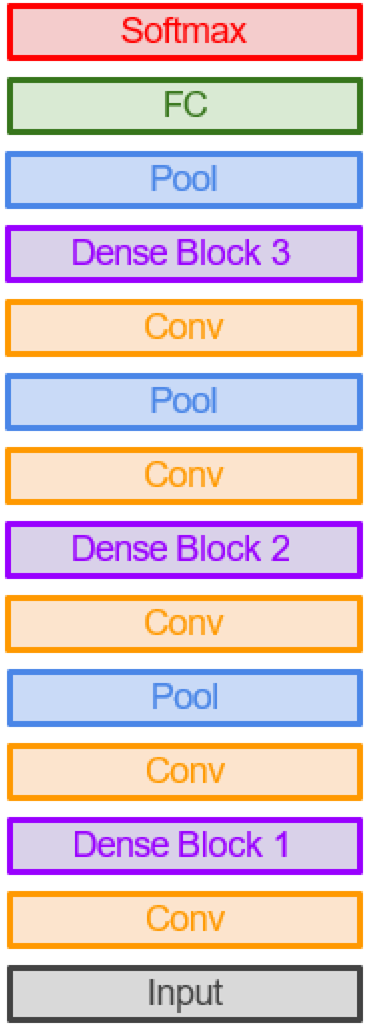
$x_L = H_L([x_0 , x_1 , . . . , x_{L-1}])$ ⇨ concatenation

$[x_0 , x_1 , . . . , x_{L-1}]$: concatenation of the feature-maps

4.1. DenseNet

2. Concatenation



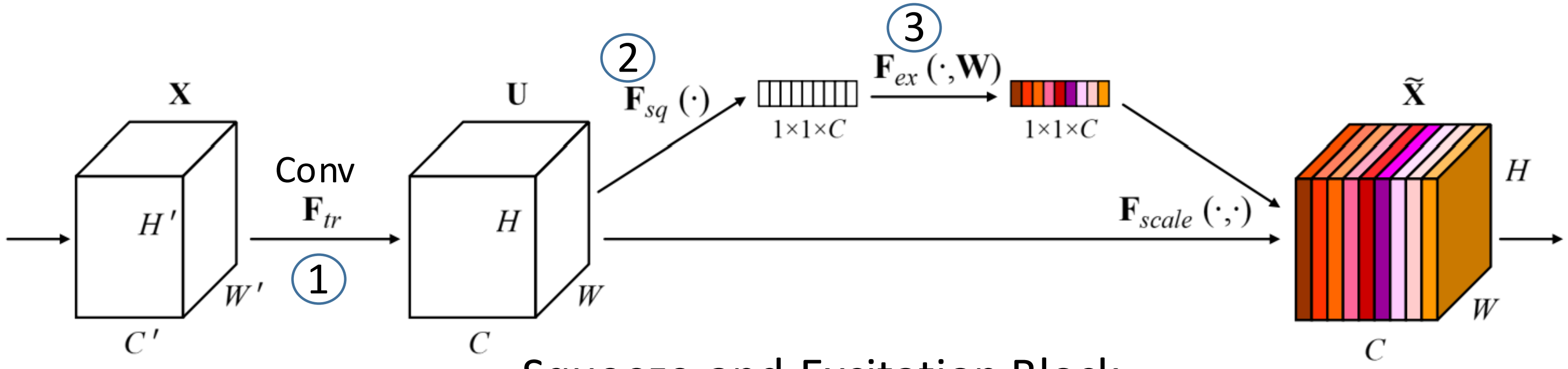


The concatenation : not viable when the size of feature-maps changes
Down-sampling layers that change the size of feature-maps
Essential part of convolutional networks

Facilitate down-sampling : multiple densely connected dense blocks
Concatenation : Pooling layers

4.2. SENet

1. Squeeze-and-Excitation block



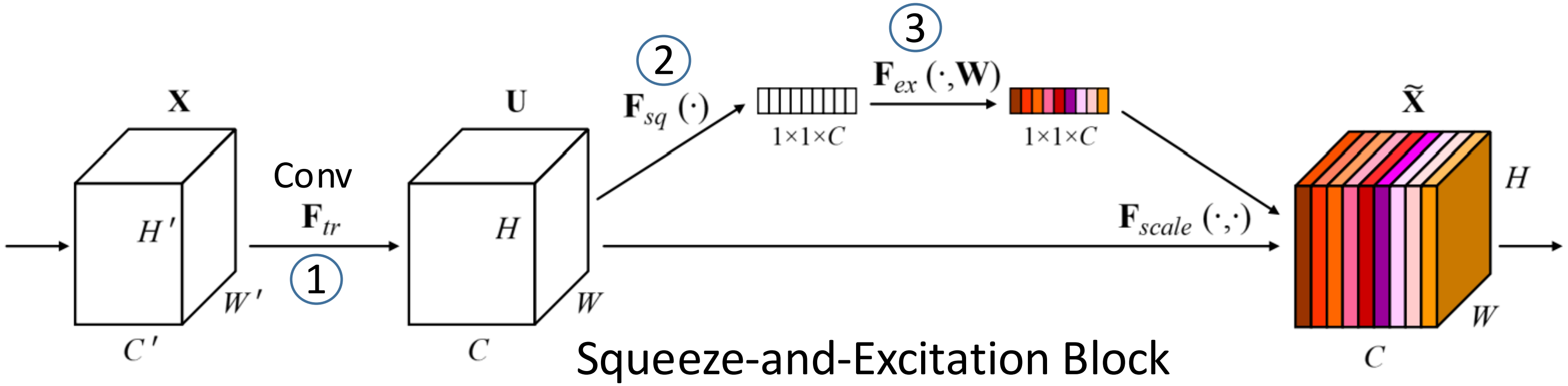
Squeeze-and-Excitation Block

1 U : output feature (each channel) : local receptive field
-> unable to exploit contextual information outside of region

Squeeze global spatial information

4.2. SENet

1. Squeeze-and-Excitation block

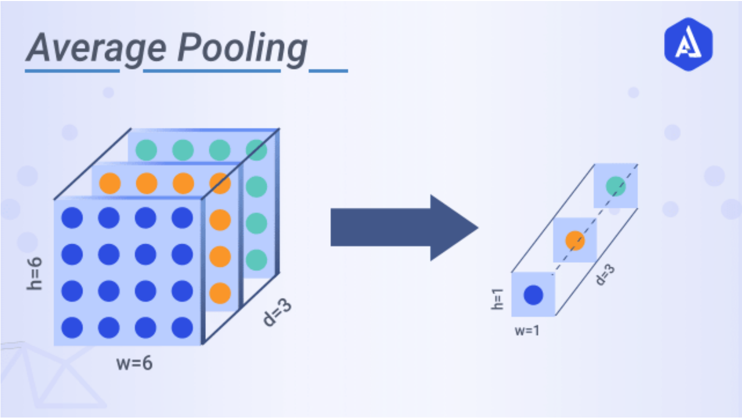


Squeeze-and-Excitation Block

2 Squeeze: Global Information Embedding

$$z_c = F_{sq}(u_c) = \frac{1}{H \times W} X \sum_{i=1}^{H} \sum_{j=1}^{W} u_c(i,j)$$

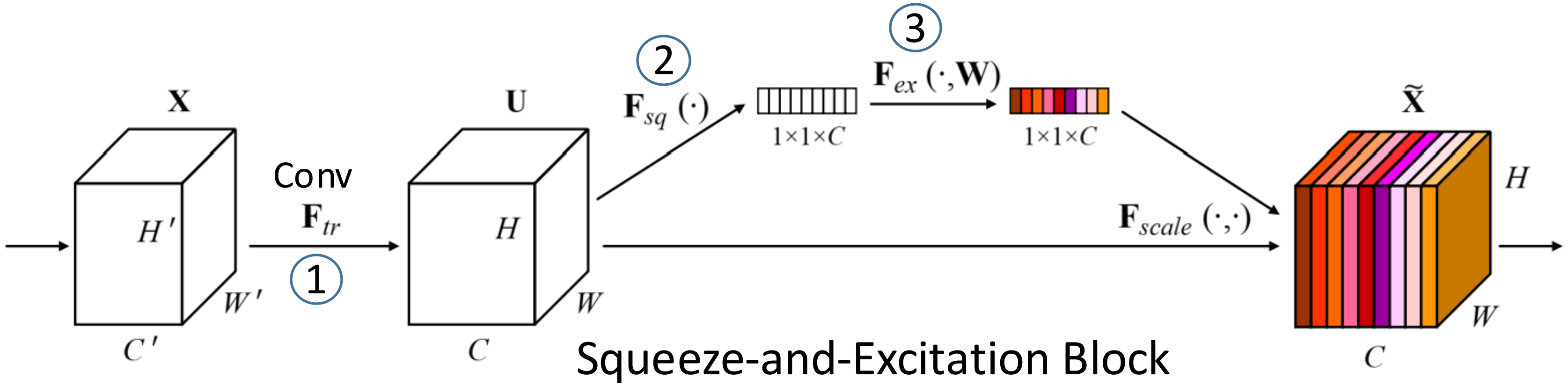
(Global average Pooling)



Squeeze: **extracting important information** in each channel

4.2. SENet

1. Squeeze-and-Excitation block



Squeeze-and-Excitation Block

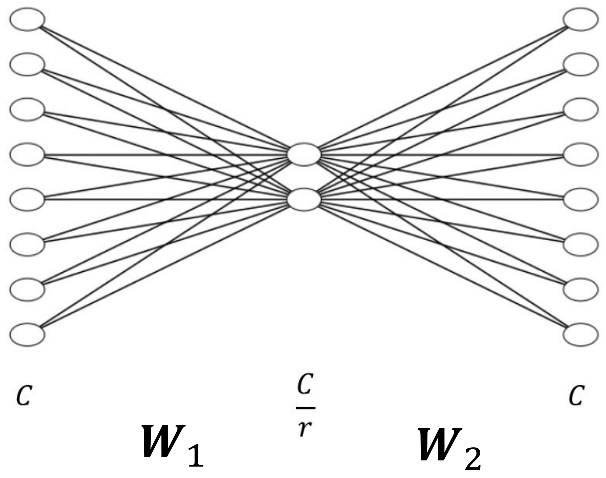
3 Excitation: Adaptive Recalibration

$$s = F_{ex}(\boldsymbol{z}, \boldsymbol{W}) = \sigma(g(\boldsymbol{z}, \boldsymbol{W})) = \sigma(\boldsymbol{W}_2 \delta(\boldsymbol{W}_1 \boldsymbol{z}))$$

δ : ReLU function

σ : sigmoid function

$\boldsymbol{W}_1$, $\boldsymbol{W}_2$: fully connected layer



r : reduction ratio

Thank you