Densely Connected Convolutional Networks

SKT Fellowship

LEE JINKYU

Department of Civil, Environmental and Architectural Engineering, Korea University

December 10, 2022

Introduction

About DenseNet

1. What is DenseNet

2. ResNet to DenseNet

3. Structure of DenseNet

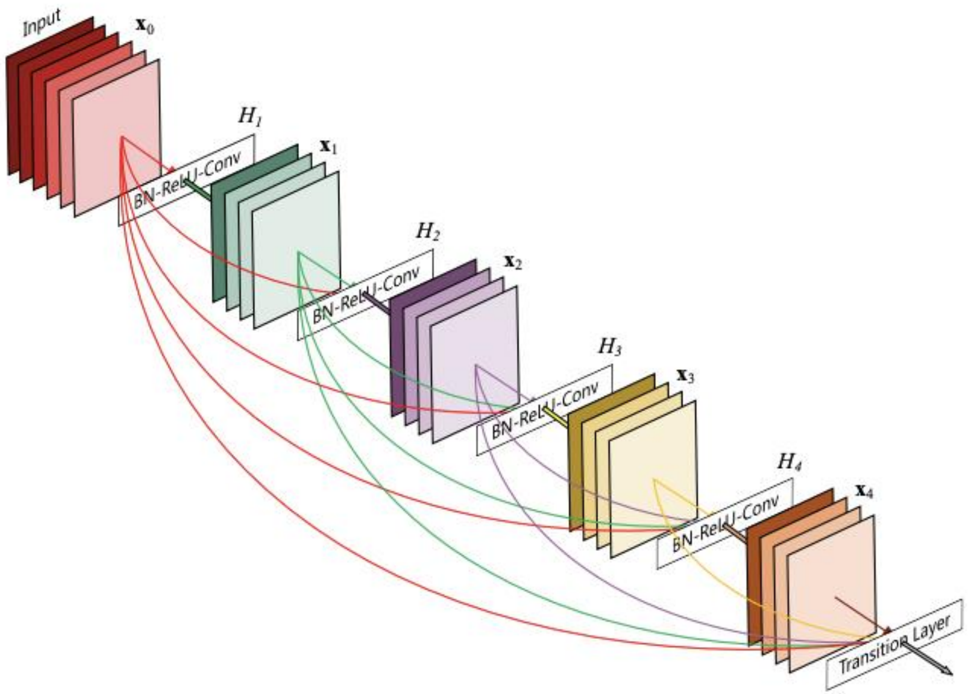
- Dense connectivity
- Pooling layers
- Bottleneck layers
- Compression

4. Advantages of DenseNet

- Strong Gradient Flow
- Parameters & Computational Efficiency
- Maintain Low Complexity Features

What is DenseNet

Densely Connected Network



Network가 Deep해짐에 따라 Vanishing gradient가 발생한다는 문제를 해결하기 위해 Skip-Connection 사용(Resnet)

Skip-Connection을 adjacent layer끼리 사용하는 것이 아닌 모든 layer에 연결하여 Dense Connectivity를 만듦

→ Dense-Connectivity로 인한 Gradient의 직접적 연결을 통해 Deep Network로 인해 발생하는 Optimize 문제를 해결

→ 적은 Parameters와 Narrow Network

ResNet to DenseNet

Why DenseNet is better

Resnet의 후속 연구를 통해 깊은 네트워크에서 모든 layer가 유의미하지 않음을 확인 (Stochastic Depth, dropout layers randomly

→ 깊은 레이어를 쌓는 것 보다 각 레이어를 병렬적으로 연결해 Compact Network를 생성함 (Feature reuse, Efficient)

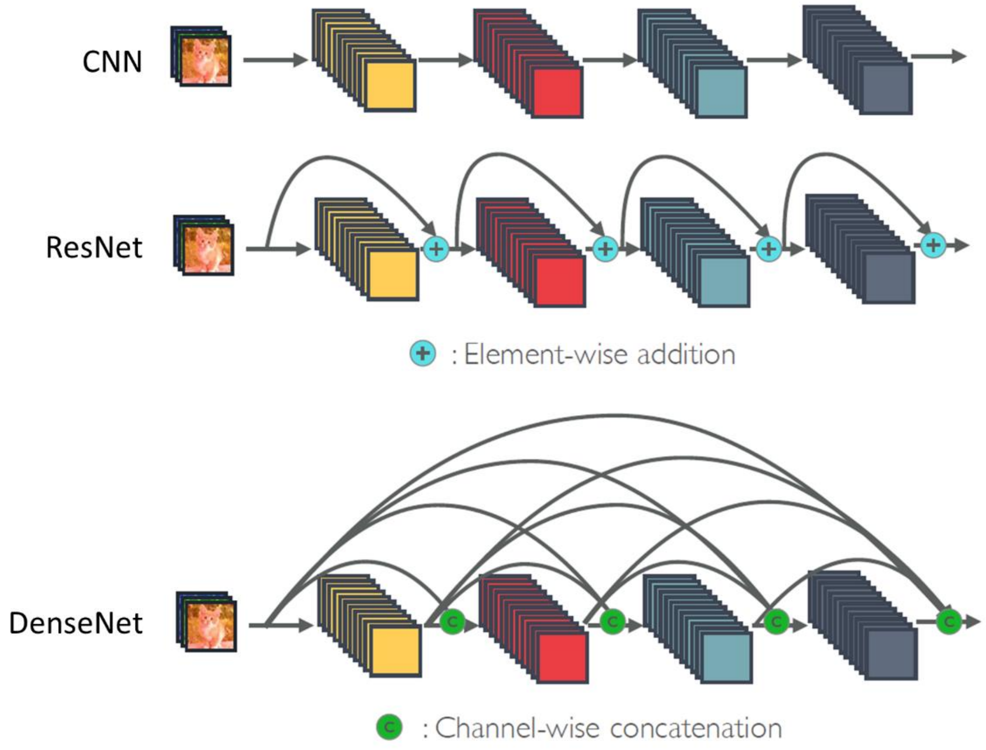
+ Summation Skip-Connection 방식이 input information과 skip-connected information을 혼합시켜 오히려 성능을 방해할 수 있다 (DenseNet은 concate을 통해 두 information을 구분)

Resnet

$$x_l = H_l(x_{l-1}) + x_{l-1}$$

Densenet

$$x_l = H_l([x_0, x_1, \ldots, x_{l-1}])$$

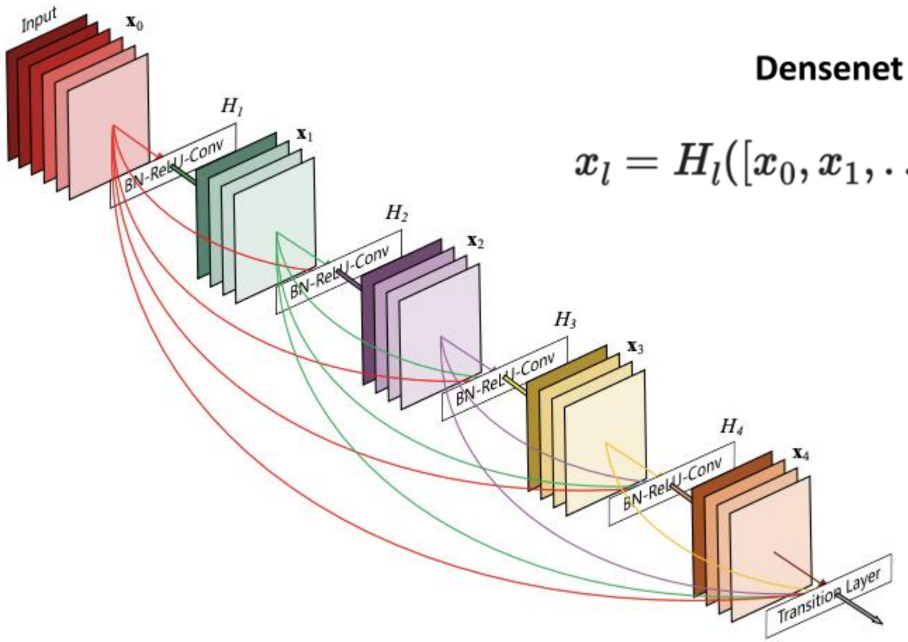


DenseNet – Dense Connectivity

DenseNet Structure

H_I(X_I)에서 나온 값을 모두 합쳐 input information으로 활용

Composite function H_1을 BN – RELU – 3X3 CONV로 구성하고 input과 output size를 유지한다 (Basic DenseNet)



Densenet

$$x_l = H_l([x_0, x_1, \ldots, x_{l-1}])$$

DenseNet – Pooling layers

DenseNet Structure

Information들의 concate을 위해선 information들의 size가 모두 동일해야한다 But, CNN의 기본 개념은 **차원, 크기 축소에 따른 특징 추출에** 있다 (Pooling)

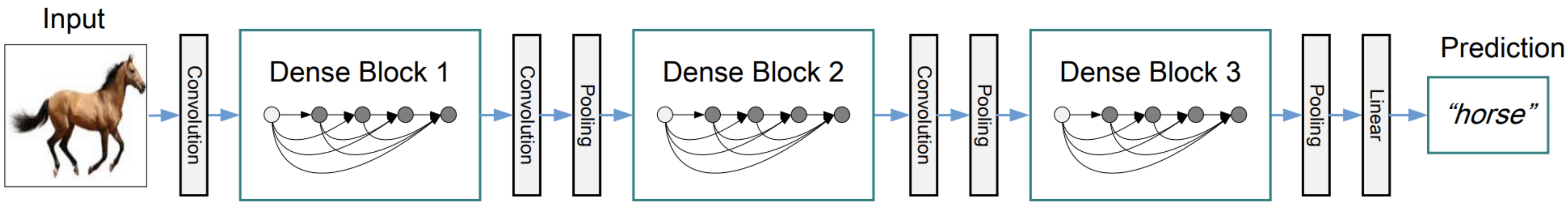


Figure 2: A deep DenseNet with three dense blocks. The layers between two adjacent blocks are referred to as transition layers and change feature-map sizes via convolution and pooling.

→ Network를 Dense Blocks로 나눈다

→ 각 Dense Block은 Dense Connectivity 상태이며 size가 동일하다

→ Dense Block 사이에 transition layers (CONV, POOL)를 통해 차원 축소를 한다

DenseNet – Pooling layers

DenseNet Structure

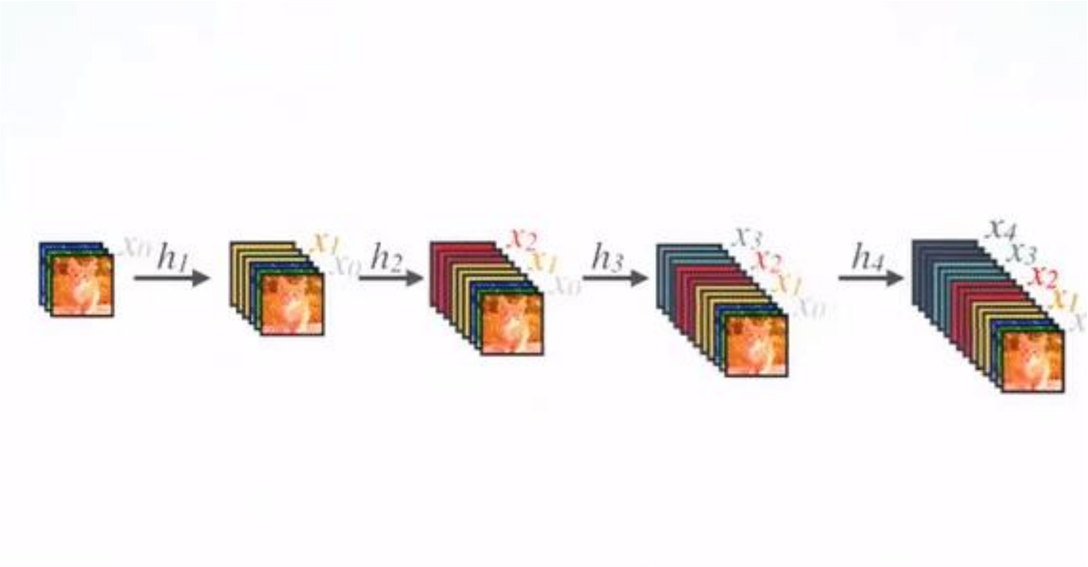
Layers	Output Size	DenseNet-121	DenseNet-169	DenseNet-201	DenseNet-264
Convolution	112×112	7×7 conv, stride 2			
Pooling	56×56	3×3 max pool, stride 2			
Dense Block (1)	56×56	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$
Transition Layer (1)	56×56	1×1 conv			
	28×28	2×2 average pool, stride 2			
Dense Block (2)	28×28	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$
Transition Layer (2)	28×28	1×1 conv			
	14×14	2×2 average pool, stride 2			
Dense Block (3)	14×14	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 24$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 48$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 64$
Transition Layer (3)	14×14	1×1 conv			
	7×7	2×2 average pool, stride 2			
Dense Block (4)	7×7	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 16$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 48$
Classification Layer	1×1	7×7 global average pool			
		1000D fully-connected, softmax			

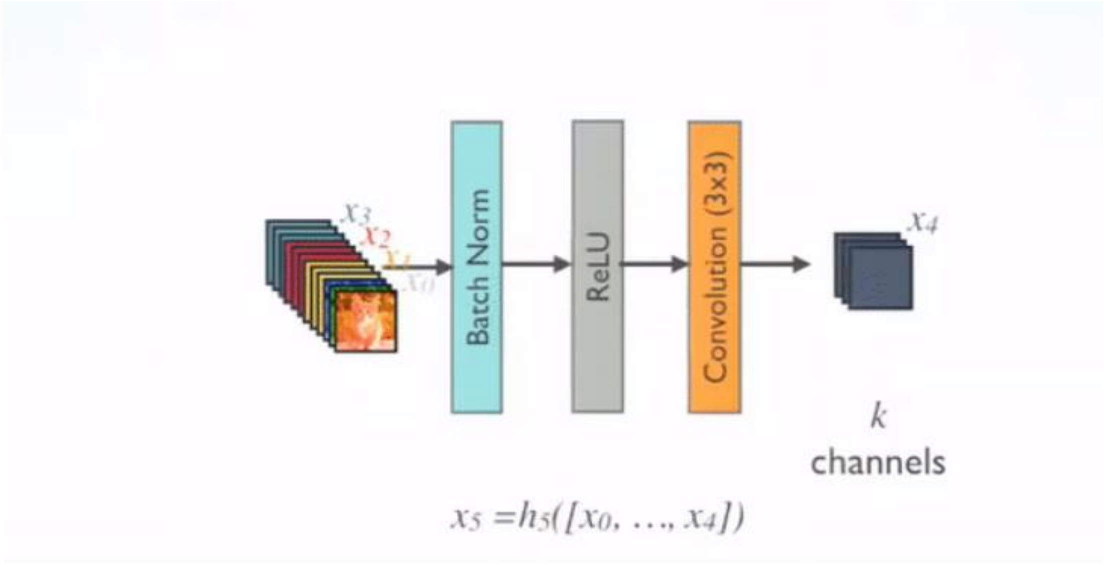
DenseNet – Growth Rate

DenseNet Structure

H_l을 통해 k feature-maps가 생성된다고 했을 때, l_th layer는 k_0 + k(l-1) feature-maps를 갖는다
(k : hyper-parameters, Network Depth를 결정)

→ DenseNet이 narrow 할 수 있는 이유

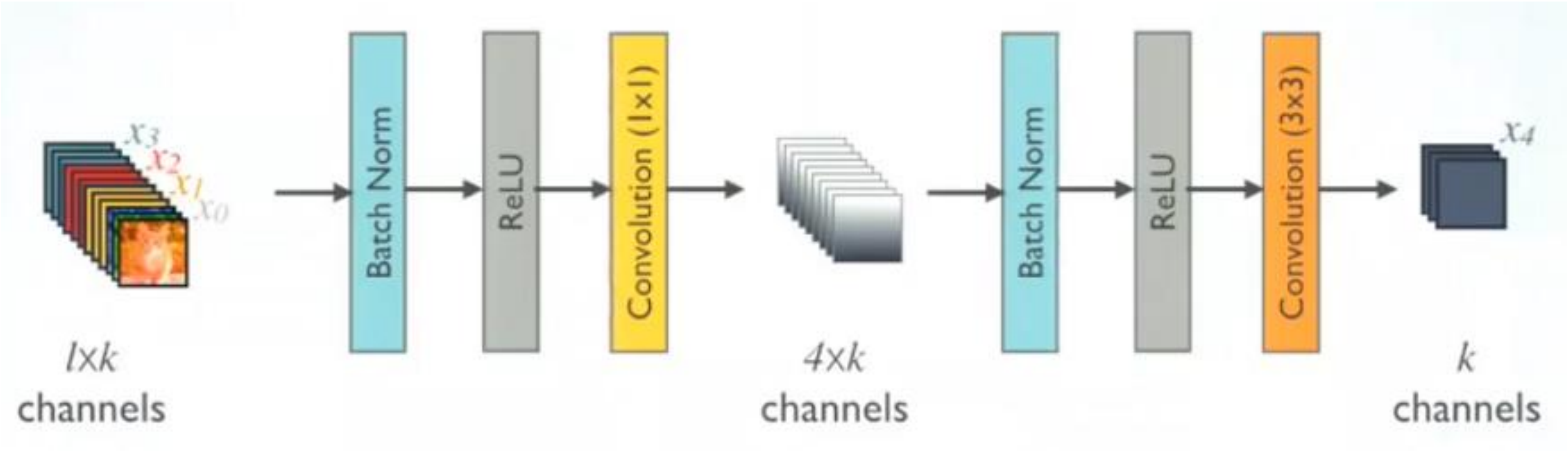




DenseNet – Bottleneck layers

DenseNet Structure

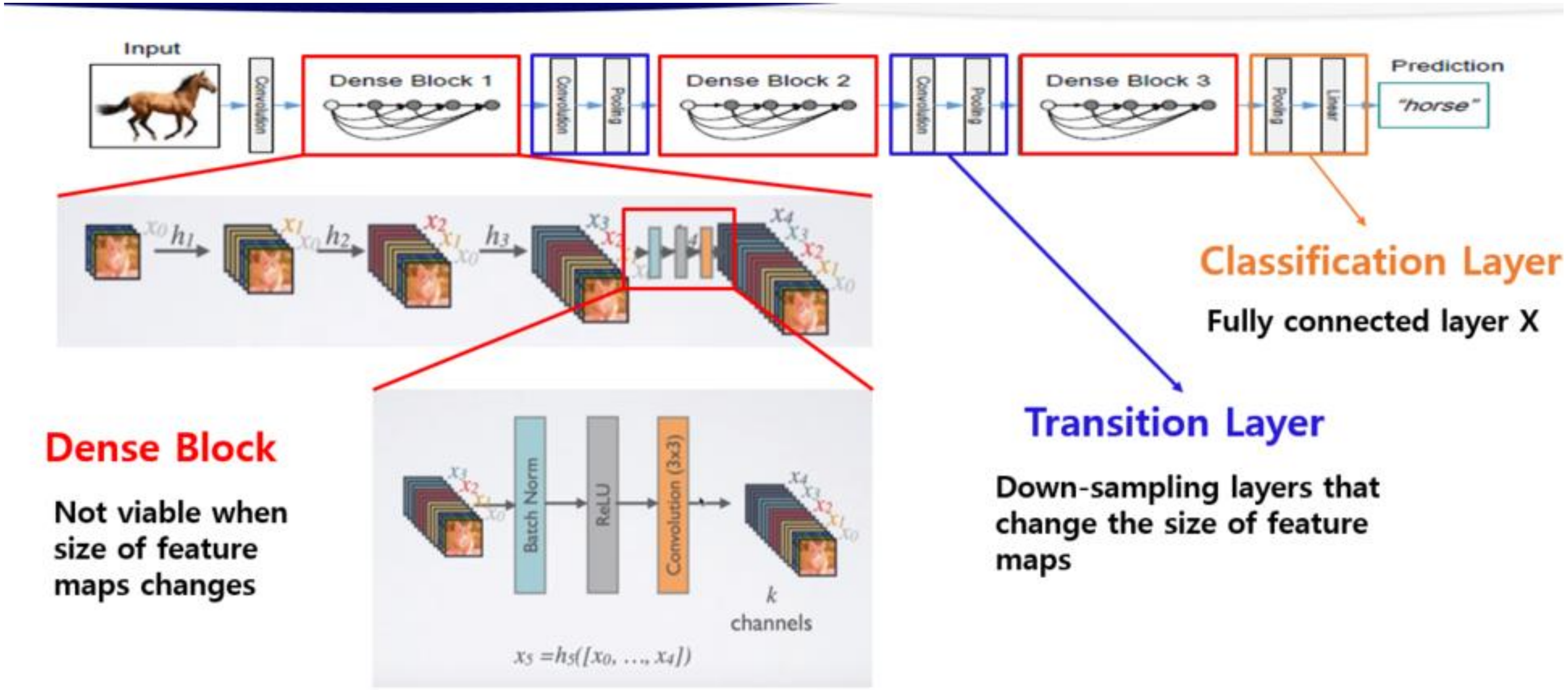
H_l : 더 efficient한 network를 위해 Conv(1x1)을 통해 각 layer의 input Channel을 줄이고 계산 (Densenet-B)



DenseNet – Compression

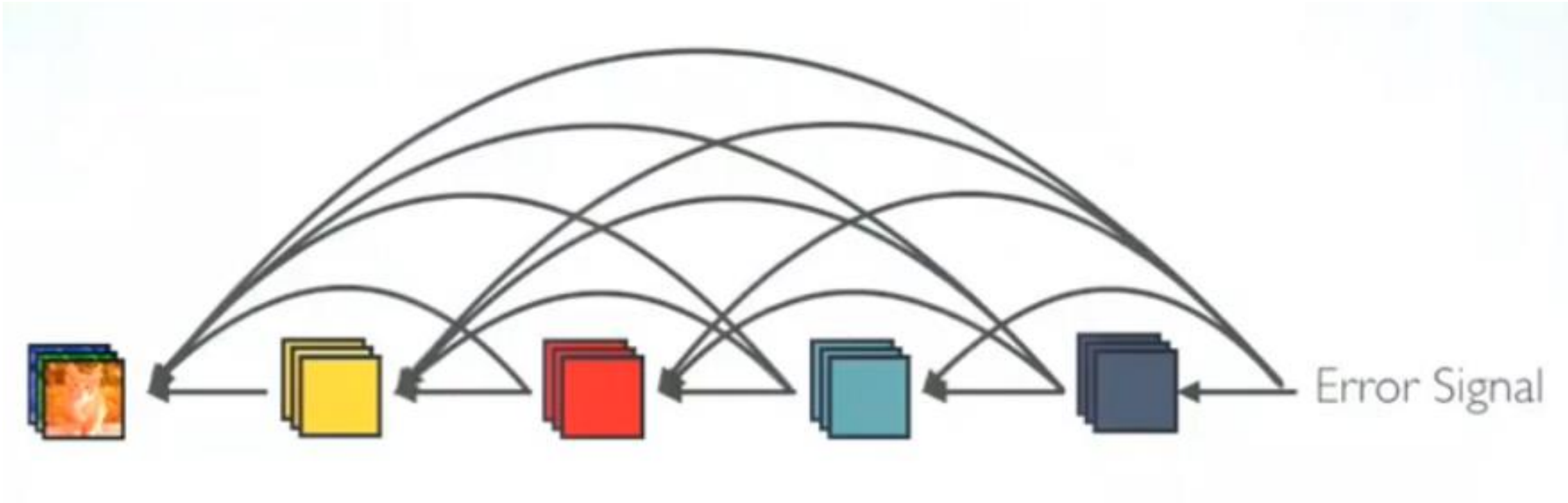
DenseNet Structure

Network의 Compactness를 위해 Dense Block에서 뿐만이 아니라 Transition layer에서도 thelta를 통해 Channels의 크기를 축소 (Densenet-BC)



Advatages of DenseNet – Strong Gradient Flow

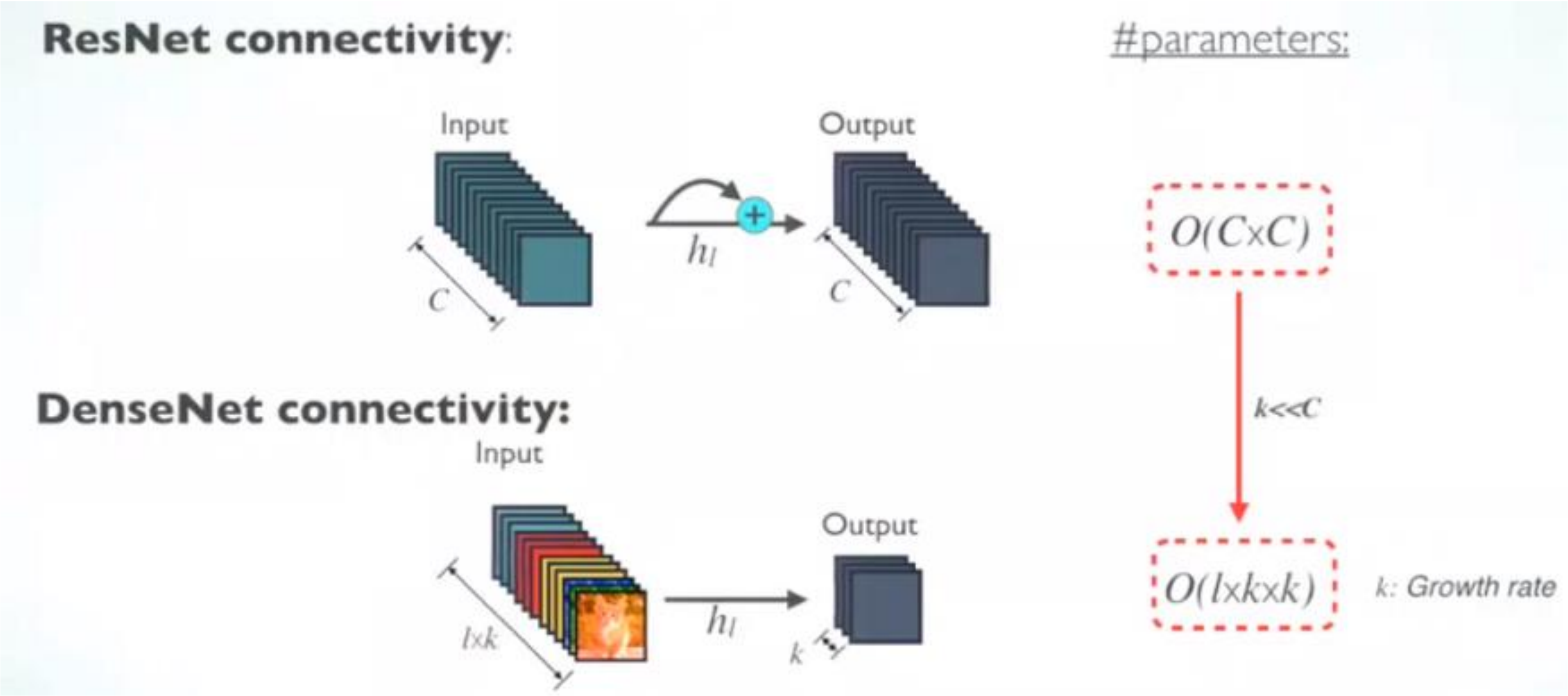
Avoid Vanishing Gradient



초기 layers와도 직접적인 연결을 통해 Vanishing Gradient 없이 back-propagation 진행

Advatages of DenseNet – Parameters & Computational Efficiency

Cost Efficient Modeling



Fixed Growth rate, Bottleneck을 통해 Narrow한 Network를 만들어 기존 CNN, ResNet에 비해 적은 Computational Cost

Advatages of DenseNet – Maintain Low Complexity Features

Use all size of complexity

Low & High level feature들을 병렬적으로 사용함으로서 적은 데이터로도 특징 추출이 잘 일어날 수 있게 함
(모든 Complexity의 Features 사용)