DENSENET

小组成员：陈晨，郭秉宁，刘泳麟

2021年8月20日

AGENDA

I. DenseNet 历史

II. DenseNet 运行过程

III. DenseNet 优劣分析

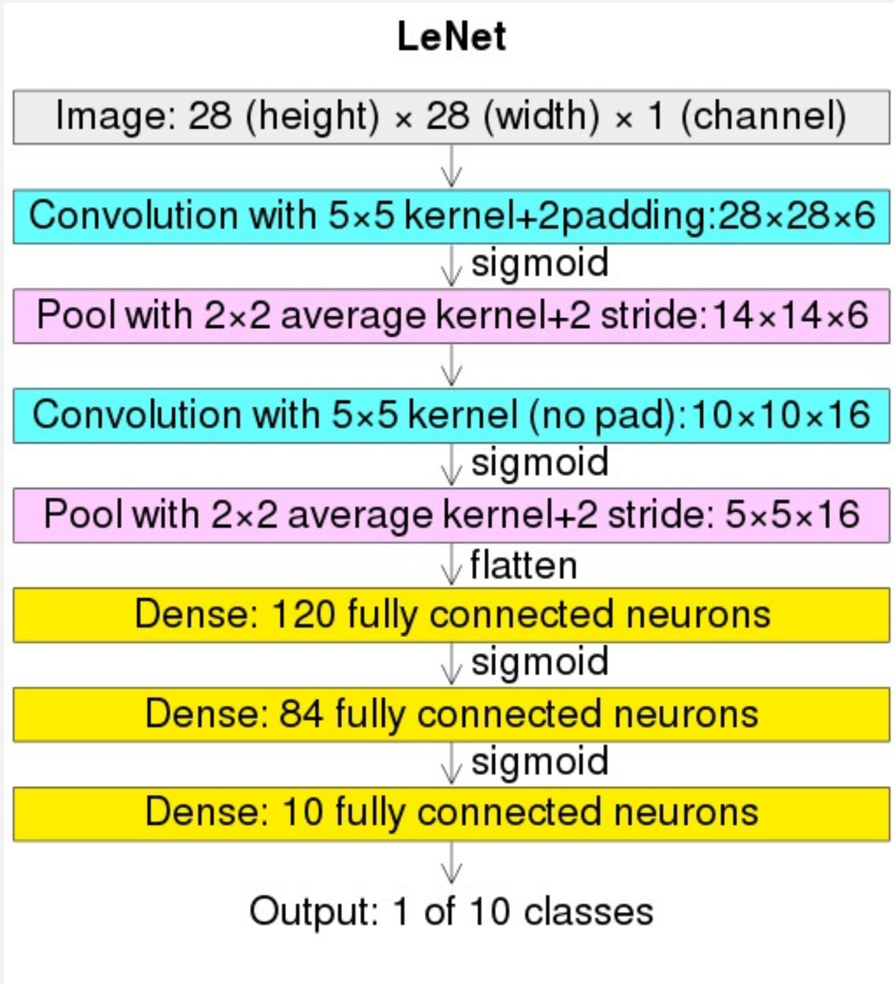
1. DENSENET历史

LENET

- 1989 Yann LeCun et al. proposed the original form of LeNet
- 1990 Their paper describes the application of backpropagation networks in handwritten digit recognition once again
- 1998 They reviewed various methods applied to handwritten character recognition and compared them with standard handwritten digit recognition benchmarks. The results show that convolutional neural networks outperform all other models.

LENET

- As a representative of the early convolutional neural network, LeNet possesses the basic units of convolutional neural network, such as convolutional layer, pooling layer and full connection layer, laying a foundation for the future development of convolutional neural network. As shown in the figure (input image data with 32*32 pixels) : LeNet-5 consists of seven layers. In addition to input, every other layer can train parameters. In the figure, Cx represents convolution layer, Sx represents sub-sampling layer, Fx represents complete connection layer, and x represents layer index.


LeNet
Image: 28 (height) × 28 (width) × 1 (channel)
Convolution with 5×5 kernel+2padding:28×28×6
sigmoid
Pool with 2×2 average kernel+2 stride:14×14×6
Convolution with 5×5 kernel (no pad):10×10×16
sigmoid
Pool with 2×2 average kernel+2 stride: 5×5×16
flatten
Dense: 120 fully connected neurons
sigmoid
Dense: 84 fully connected neurons
sigmoid
Dense: 10 fully connected neurons
Output: 1 of 10 classes

ALEXNET

- AlexNet is the name of a convolutional neural network (CNN) architecture, designed by Alex Krizhevsky in collaboration with Ilya Sutskever and Geoffrey Hinton, who was Krizhevsky's Ph.D. advisor.
- AlexNet competed in the ImageNet Large Scale Visual Recognition Challenge on September 30, 2012. The network achieved a top-5 error of 15.3%, more than 10.8 percentage points lower than that of the runner up.

ALEXNET

- AlexNet contained eight layers; the first five were convolutional layers, some of them followed by max-pooling layers, and the last three were fully connected layers. It used the non-saturating ReLU activation function, which showed improved training performance over tanh and sigmoid.

ZFNET

- In 2013, ZFNet was invented by Dr. Rob Fergus and his PhD student at that moment, Dr. Matthew D. Zeiler in NYU.

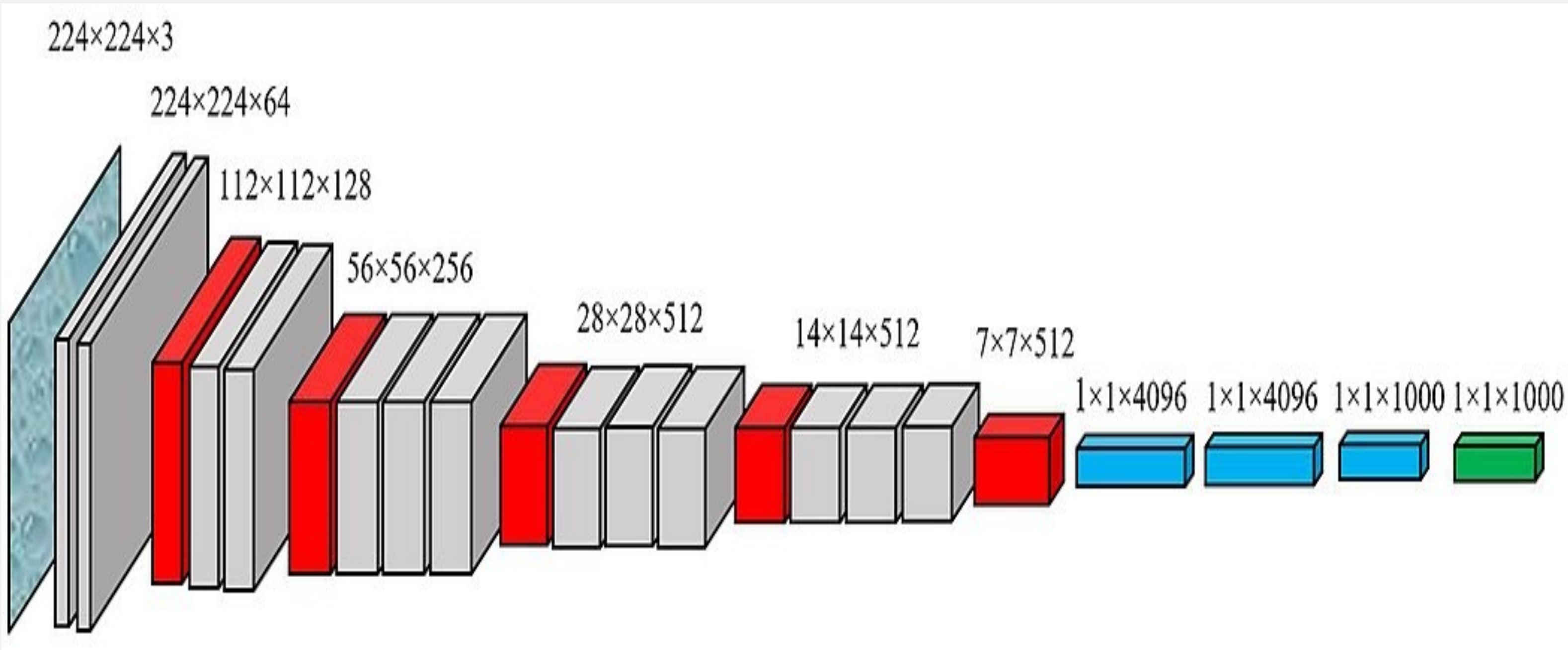
- ZFNet has significantly improved the image classification error rate compared with AlexNet, the winner in ILSVRC 2012.

VGG

- VGG Net is the name of a pre-trained convolutional neural network (CNN) invented by Simonyan and Zisserman from Visual Geometry Group (VGG) at University of Oxford in 2014 and it was able to be the 1st runner-up of the ILSVRC 2014 in the classification task.

VGG

- VGG was invented with the purpose of enhancing classification accuracy by increasing the depth of the CNNs. VGG 16 and VGG 19, having 16 and 19 weight layers, respectively, have been used for object recognition. VGG Net takes input of 224×224 RGB images and passes them through a stack of convolutional layers with the fixed filter size of 3×3 and the stride of 1. There are five max pooling filters embedded between convolutional layers in order to down-sample the input representation (image, hidden-layer output matrix, etc.). The stack of convolutional layers are followed by 3 fully connected layers, having 4096, 4096 and 1000 channels, respectively. The last layer is a soft-max layer.


224×224×3
224×224×64
112×112×128
56×56×256
28×28×512
14×14×512
7×7×512
1×1×4096
1×1×4096
1×1×1000
1×1×1000

GOOGLENET

- GoogLeNet is a 22-layer deep convolutional neural network that's a variant of the Inception Network, a Deep Convolutional Neural Network developed by researchers at Google.

- The GoogLeNet architecture presented in the ImageNet Large-Scale Visual Recognition Challenge 2014(ILSVRC14) solved computer vision tasks such as image classification and object detection.

GOOGLENET

- The GoogLeNet architecture consists of 22 layers (27 layers including pooling layers), and part of these layers are a total of 9 inception modules.

type	**patch size/ stride**	**output size**	**depth**	#1×1	#3×3 **reduce**	#3×3	#5×5 **reduce**	#5×5	**pool proj**	**params**	**ops**
convolution	7×7/2	112×112×64	1							2.7K	34M
max pool	3×3/2	56×56×64	0								
convolution	3×3/1	56×56×192	2		64	192				112K	360M
max pool	3×3/2	28×28×192	0								
inception (3a)		28×28×256	2	64	96	128	16	32	32	159K	128M
inception (3b)		28×28×480	2	128	128	192	32	96	64	380K	304M
max pool	3×3/2	14×14×480	0								
inception (4a)		14×14×512	2	192	96	208	16	48	64	364K	73M
inception (4b)		14×14×512	2	160	112	224	24	64	64	437K	88M
inception (4c)		14×14×512	2	128	128	256	24	64	64	463K	100M
inception (4d)		14×14×528	2	112	144	288	32	64	64	580K	119M
inception (4e)		14×14×832	2	256	160	320	32	128	128	840K	170M
max pool	3×3/2	7×7×832	0								
inception (5a)		7×7×832	2	256	160	320	32	128	128	1072K	54M
inception (5b)		7×7×1024	2	384	192	384	48	128	128	1388K	71M
avg pool	7×7/1	1×1×1024	0								
dropout (40%)		1×1×1024	0								
linear		1×1×1000	1							1000K	1M
softmax		1×1×1000	0								

RESNET

- 2015 He, Kaiming; Zhang, Xiangyu; Ren, Shaoqing; Sun, Jian ."Deep Residual Learning for Image Recognition"

RESNET

- The core idea of ResNet is introducing a so-called "identity shortcut connection" that skips one or more layers.
- The authors argue that stacking layers shouldn't degrade the network performance, because we could simply stack identity mappings (layer that doesn't do anything) upon the current network, and the resulting architecture would perform the same. This indicates that the deeper model should not produce a training error higher than its shallower counterparts. They hypothesize that letting the stacked layers fit a residual mapping is easier than letting them directly fit the desired underlaying mapping. And the residual block above explicitly allows it to do precisely that.

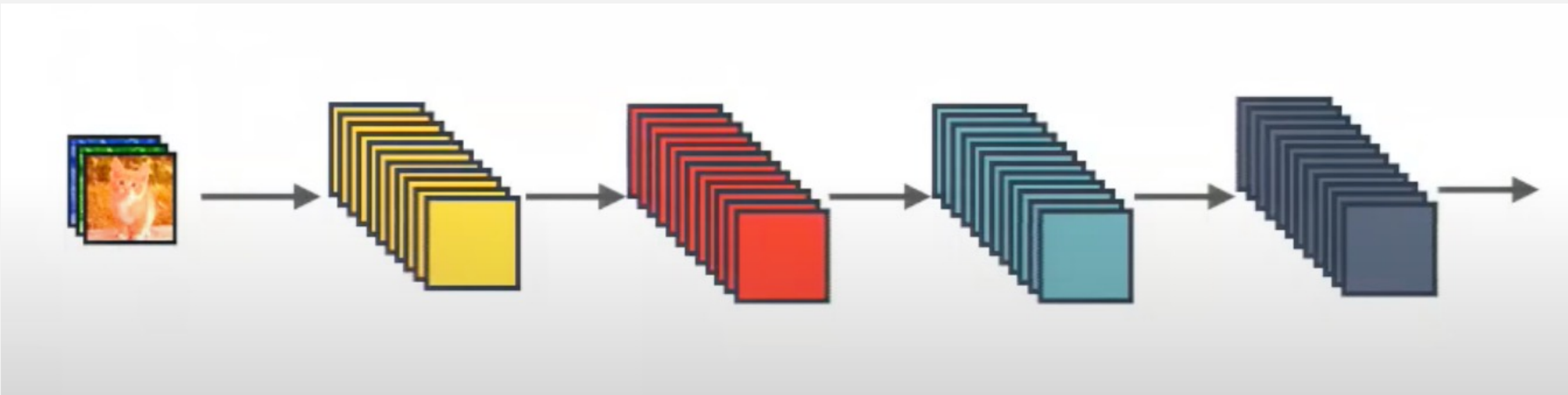
DENSENET

- 2018 G. Huang, Z. Liu and L. van der Maaten, "Densely Connected Convolutional Networks"

- DenseNet is one of the new discoveries in neural networks for visual object recognition. DenseNet is quite similar to ResNet with some fundamental differences. ResNet uses an additive method (+) that merges the previous layer (identity) with the future layer, whereas DenseNet concatenates (.) the output of the previous layer with the future layer.

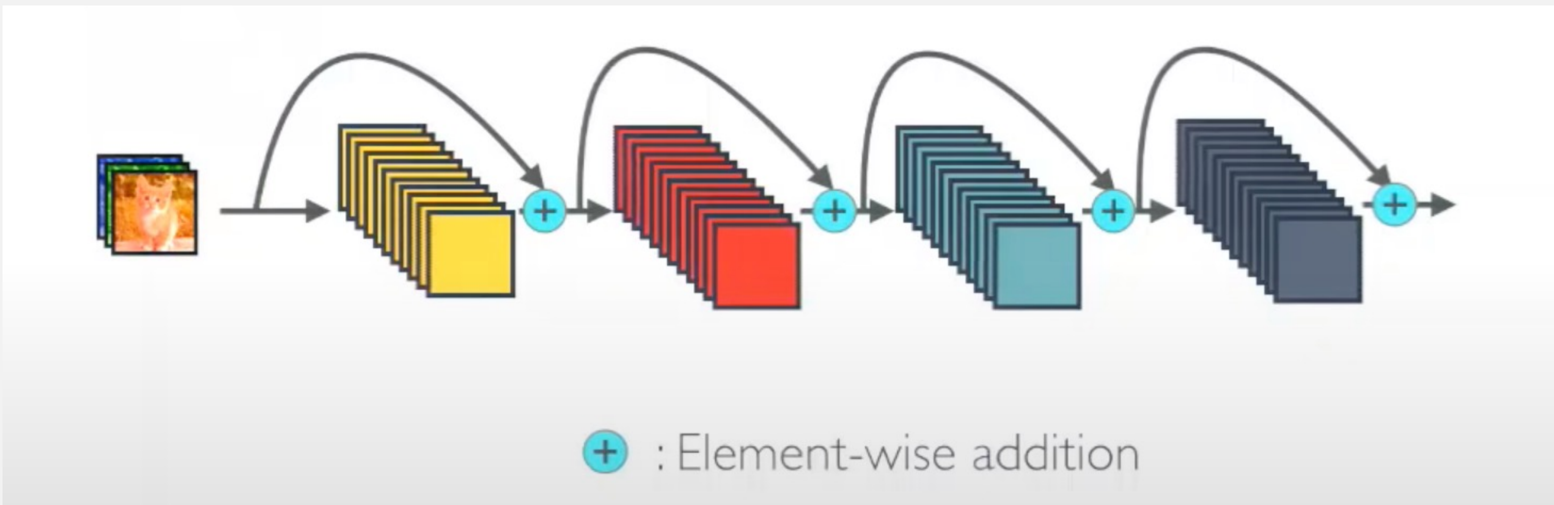
II. DENSENET运行过程

Standard Connectivity – Before ResNet



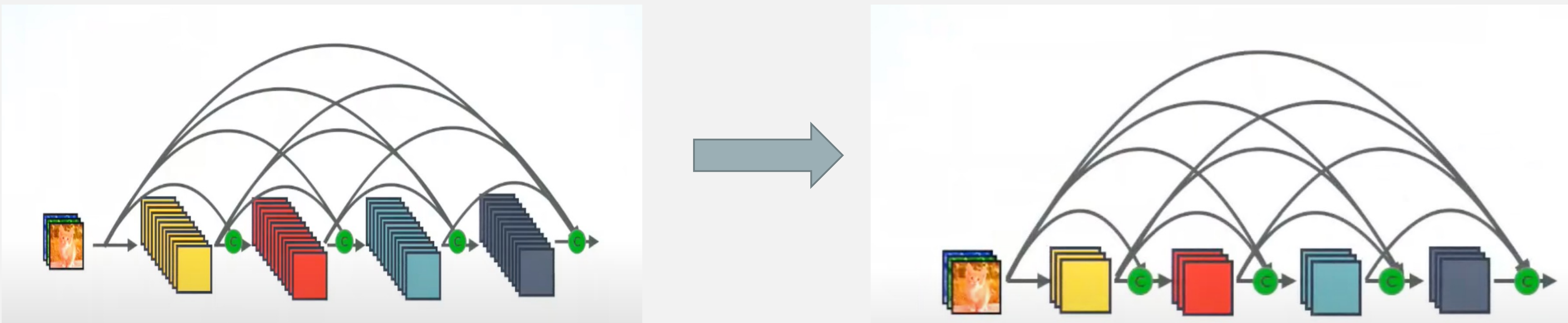
Each layer receives input from previous layer, and generate features for the layer after.

ResNet



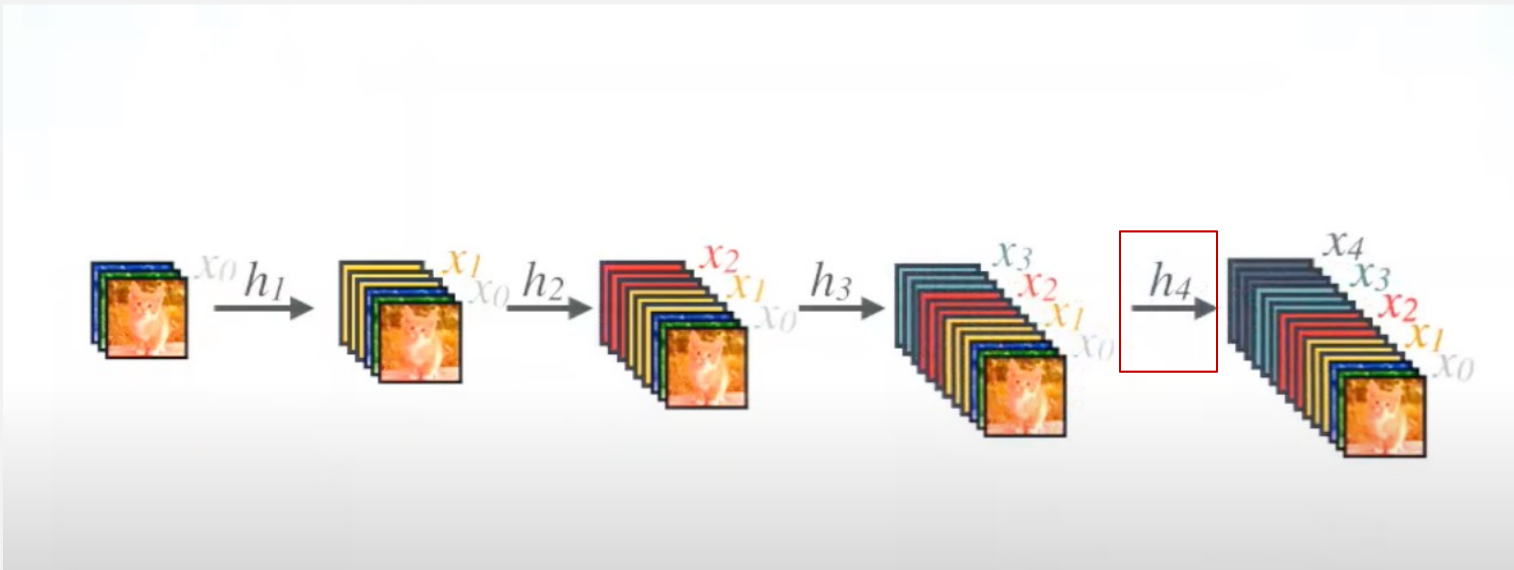
ResNet incorporates identity mapping and Element-wise addition, which promotes gradient propagation.

Dense Connectivity (DenseNet)

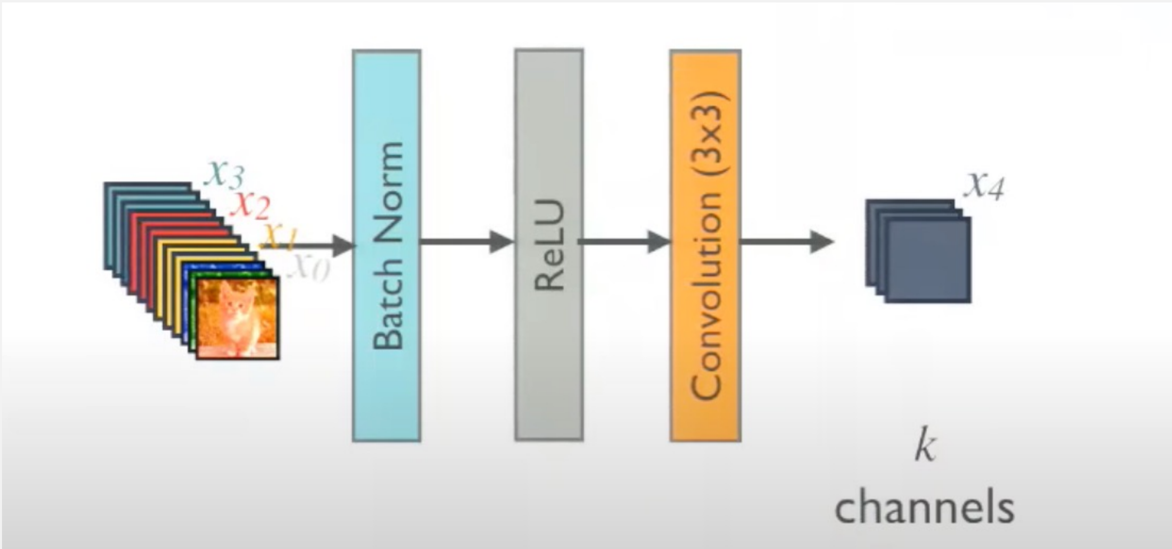


With DenseNet, each layer receives additional input from **all preceding layers** through **Channel-wise Concatenation.** This reduces **information bottleneck,** thus enabling thinner layers and higher computational efficiency.

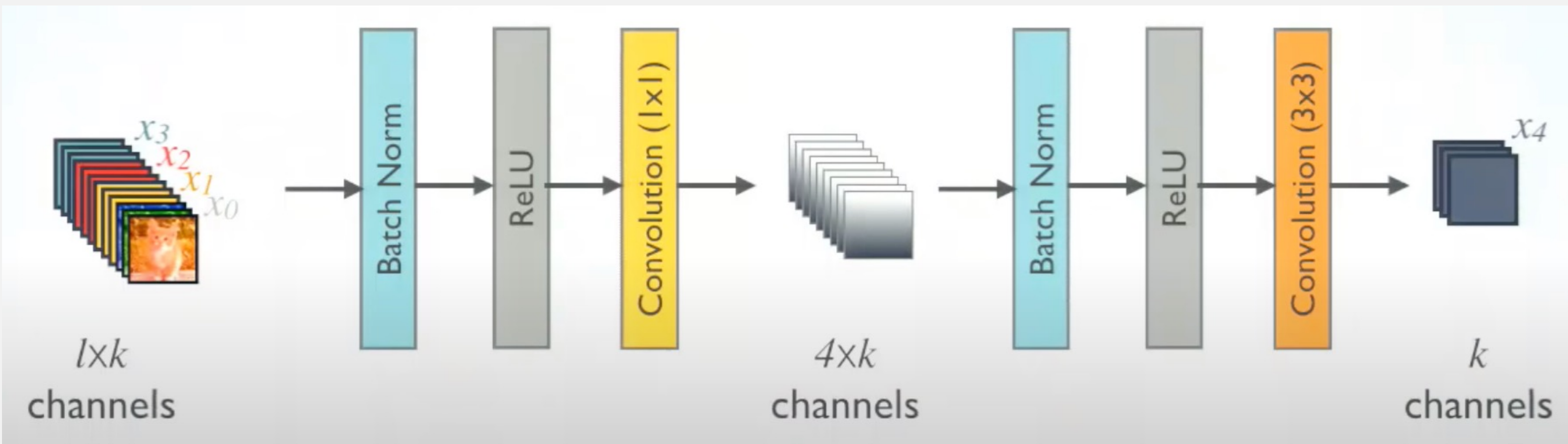
Feature Generation: Channel-wise Concatenation



Composite Layer in DenseNet

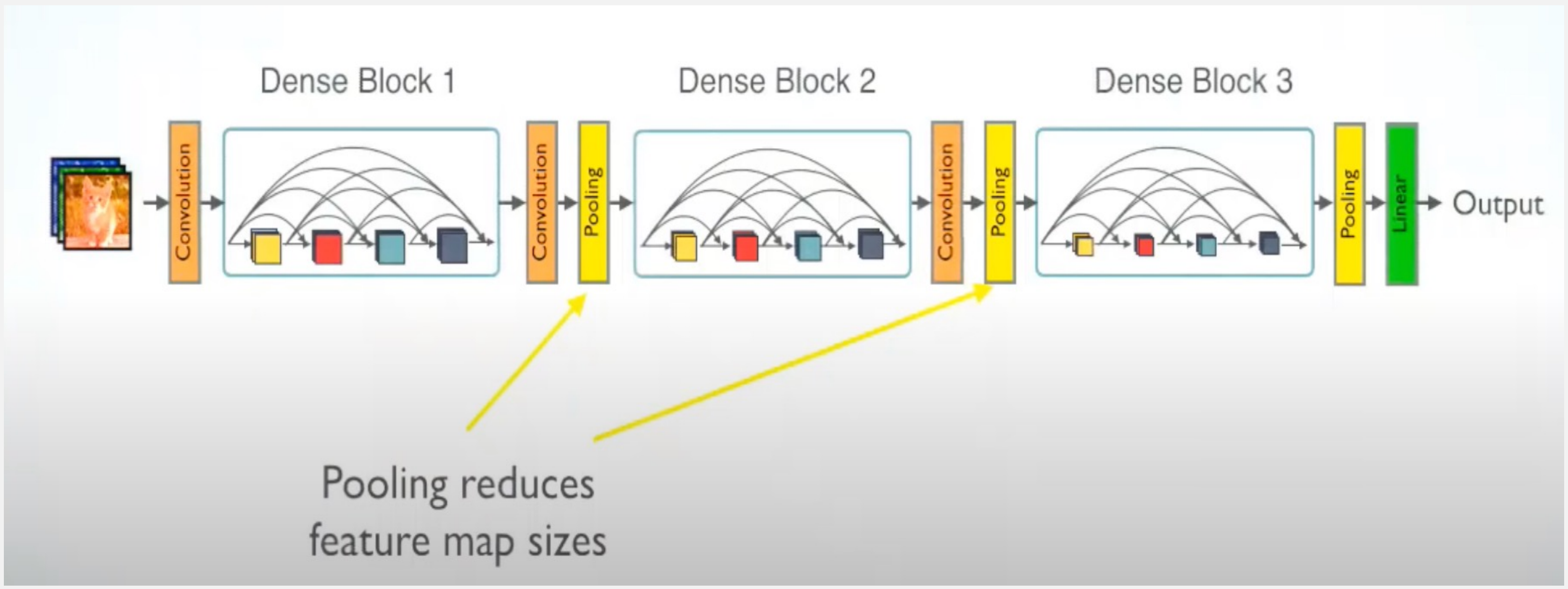


Improving Parameter and Computational efficiency



With all the concatenating, input for deeper layers may be too wide. This is addressed by adding **Bottleneck Layer** (1x1 convolution).

Tying it all together: DenseNet + CNN

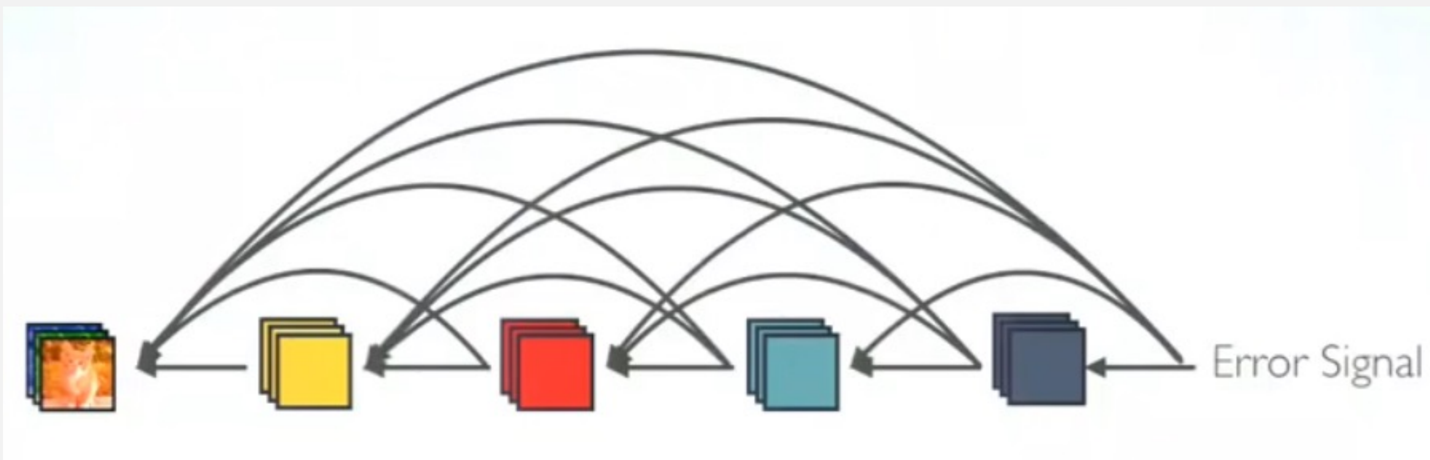


III. DENSENET优势与劣势

Advantages of DenseNet

a.Strong Gradient Flow

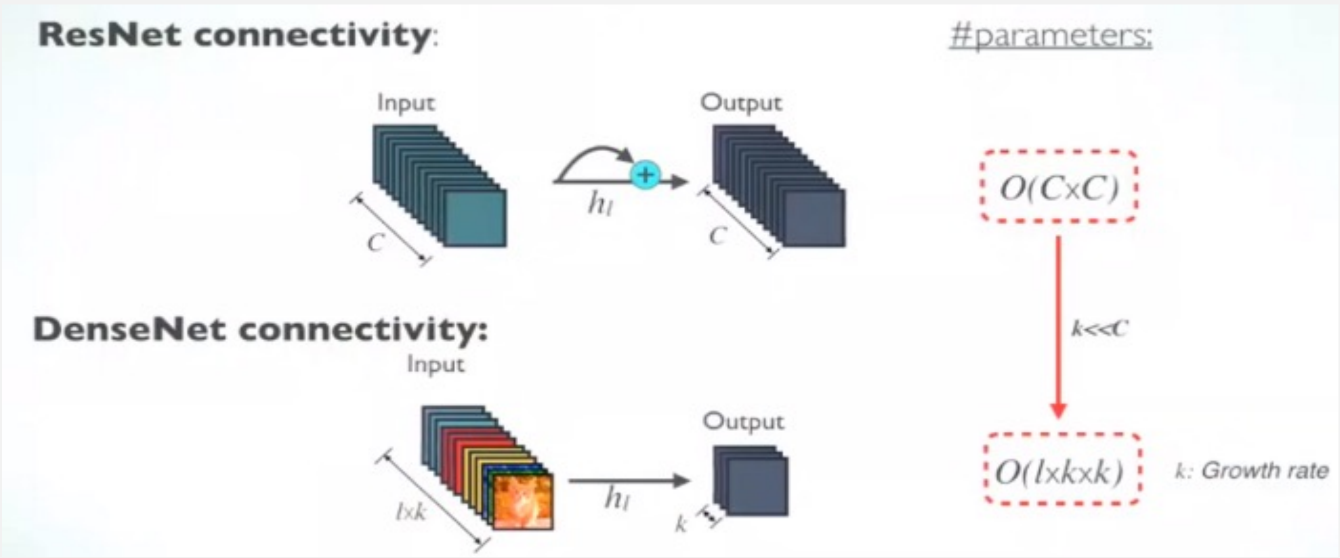
Each layer has direct access to the gradients from the loss function and the original input signal, leading to an implicit deep supervision. Thus it is possible to mitigate the vanishing-gradient.



b.Parameter & Computational Efficiency

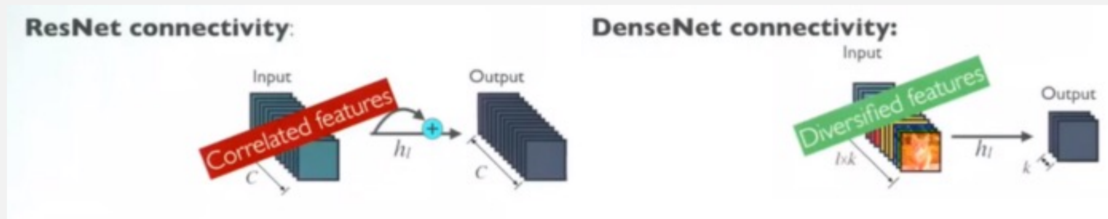
For each layer, number of parameters in ResNet is directly proportional to C×C while Number of parameters in DenseNet is directly proportional to l×k×k.

Since k<<C, DenseNet has much smaller size than ResNet.



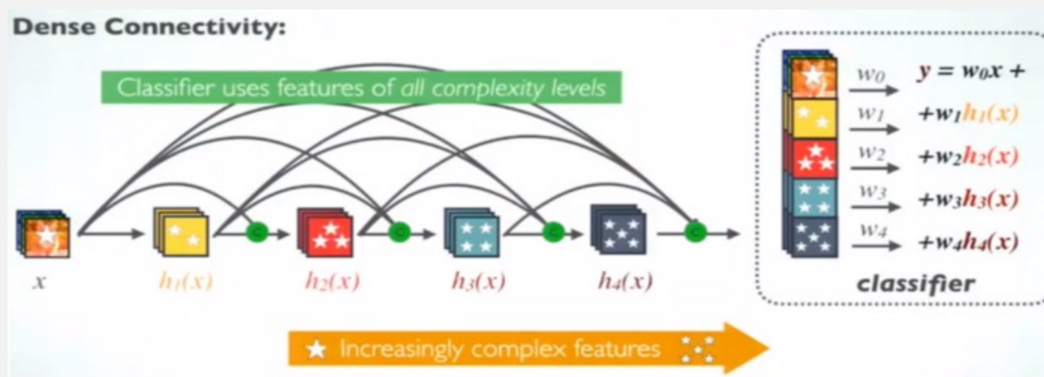
c. More Diversified Features

Since each layer in DenseNet receive all preceding layers as input, more diversified features and tends to have richer patterns.



d. Maintains Low Complexity Features

In DenseNet, classifier uses features of all complexity levels. It tends to give more smooth decision boundaries. It also explains why DenseNet performs well when training data is insufficient.



Disadvantages of DenseNet

a.Memory consumption

Compared with ResNet, DenseNet uses a lot more memory, as the tensors from different are concatenated together.