

بینایی کامپیوتر

فصل دوم

معماری‌های کانولوشنی (مطالعه موردی)

محمد جواد فدائی‌اسلام

بهمن ۱۴۰۳

- Classic network architectures (included for historical purposes)
 - LeNet-5
 - AlexNet
 - VGG 16
- Modern network architectures
 - Inception (GoogLeNet)
 - ResNet
 - EfficientNet
 - MobileNet



Large Scale Visual Recognition Challenge 2010 (ILSVRC2010)

The images were collected from the web and labeled by **human labelers** using Amazon's Mechanical Turk crowd-sourcing tool. Starting in 2010, as part of the Pascal Visual Object Challenge, an annual competition called the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) has been held. ILSVRC uses a subset of ImageNet with roughly **1000** images in each of **1000 categories**. In all, there are roughly **1.2 million** training images, **50,000 validation** images, and **150,000 testing images**.

IMAGENET Large Scale Visual Recognition Challenge 2017 (ILSVRC2017)

[Introduction](#) [News](#) [History](#) [Timetable](#) [Challenges](#) [FAQ](#) [Citation](#) [Contact](#)

Introduction

This challenge evaluates algorithms for object localization/detection from images/videos at scale. Most successful and innovative teams will be invited to present at **CVPR 2017 workshop**.

- I. [Object localization](#) for 1000 categories.
- II. [Object detection](#) for 200 fully labeled categories.
- III. [Object detection from video](#) for 30 fully labeled categories.

News

- Jul 26, 2017: We are passing the baton to [Kaggle](#). From now on, all three challenges(LOC-CLS, DET, VID) will be hosted on Kaggle!

History

[2016](#), [2015](#), [2014](#), [2013](#), [2012](#), [2011](#), [2010](#)

ALEXNET

- The paper “ImageNet Classification with Deep Convolutional Neural Networks”(2012) by **Alex Krizhevsky** et al. built a new landscape in Computer Vision by destroying old ideas in one masterful stroke.
- The paper used a CNN to get a Top-5 error rate (rate of not finding the true label of a given image among its top 5 predictions) of 15.3%. The next best result trailed far behind (26.2%).
- This architecture is popularly called AlexNet.

ALEXNET RESULTS



mite

container ship

motor scooter

leopard

	mite
	black widow
	cockroach
	tick
	starfish

	container ship
	lifeboat
	amphibian
	fireboat
	drilling platform

	motor scooter
	go-kart
	moped
	bumper car
	golfcart

	leopard
	jaguar
	cheetah
	snow leopard
	Egyptian cat



grille

mushroom

cherry

Madagascar cat

	convertible
	grille
	pickup
	beach wagon
	fire engine

	agaric
	mushroom
	jelly fungus
	gill fungus
	dead-man's-fingers

	dalmatian
	grape
	elderberry
	ffordshire bullterrier
	currant

	squirrel monkey
	spider monkey
	titi
	indri
	howler monkey

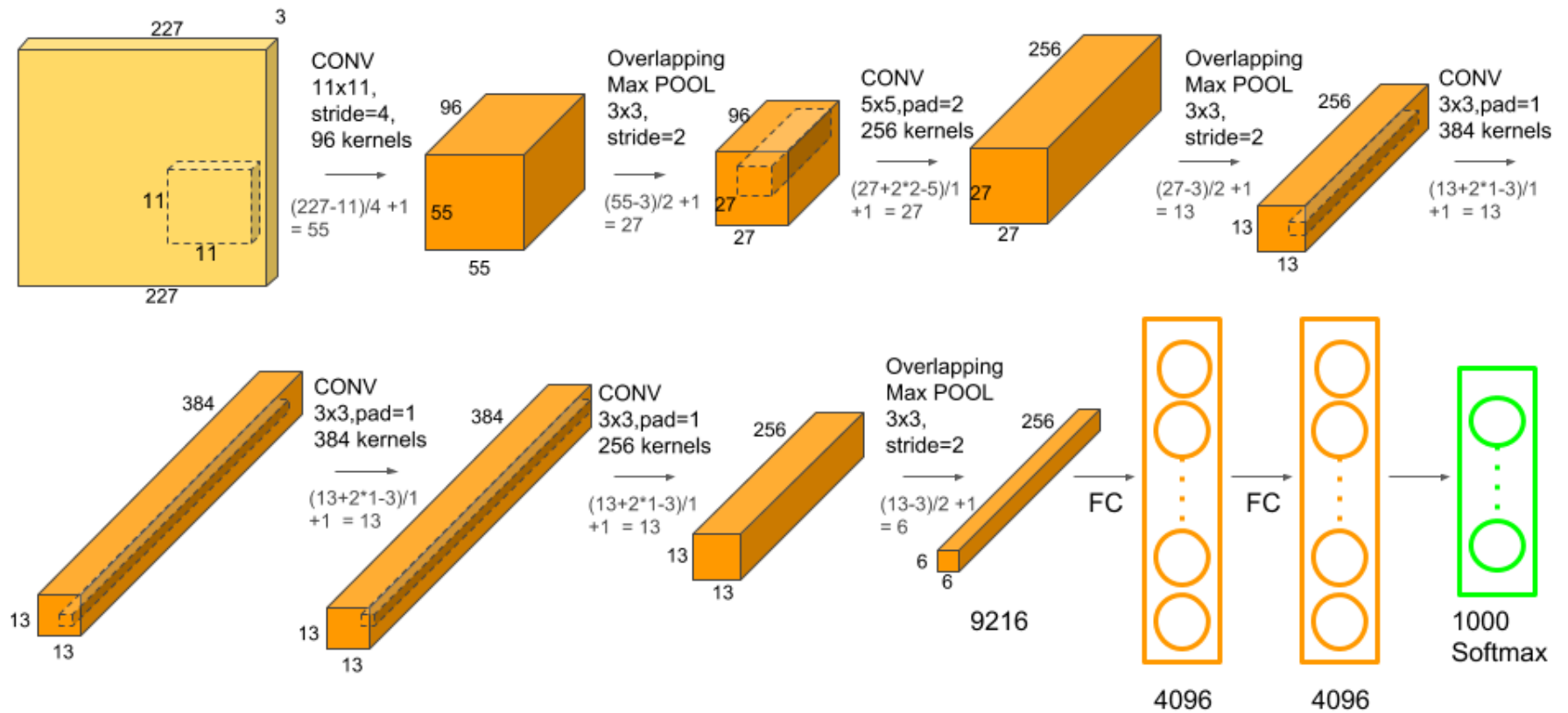
ALEXNET-PREPROCESSING

- ImageNet consists of variable-resolution images, while our system requires a constant input dimensionality.
- Therefore, we down-sampled the images to a fixed resolution of 256×256 .
- Given a rectangular image, we first rescaled the image such that the shorter side was of length 256, and then cropped out the central 256×256 patch from the resulting image.
- We did not pre-process the images in any other way, except for subtracting the mean activity over the training set from each pixel.
- So the model was trained on the (centered) raw RGB values of the pixels.

DATA AUGMENTATION

- They extracted **random 227×227 patches** (and their horizontal reflections) from the 256×256 images and training the network on these extracted patches.
- This increases the size of our training set by a factor of **2048**.
- Without this scheme, the network suffers from **substantial overfitting**.
- At test time, the network makes a prediction by extracting five 227×227 patches (**the four corner patches and the center patch**) as well as **their horizontal reflections** (hence ten patches in all), and averaging the predictions made by the network's softmax layer on the ten patches.

ALEXNET



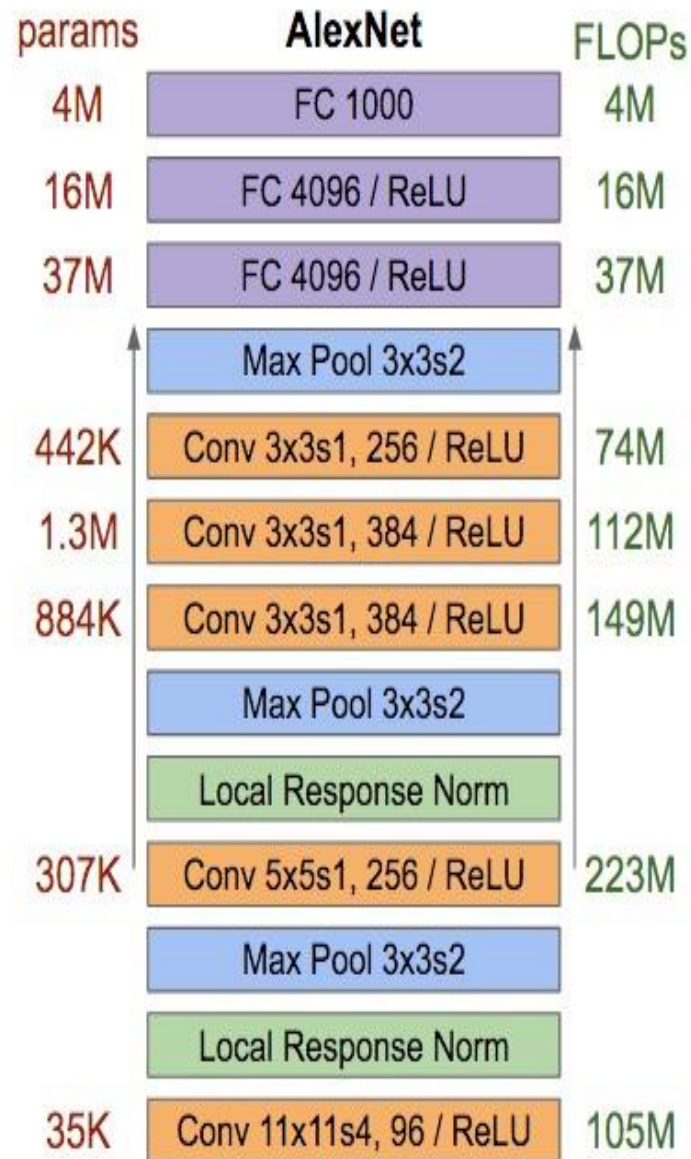
REAL-WORLD EXAMPLE



Each of the 96 filters shown here is of size $[11 \times 11 \times 3]$, and each one is shared by the 55×55 neurons in one depth slice.

ALEX-NET

The Network had a very similar architecture to LeNet, but was **deeper**, **bigger**, and **featured Convolutional Layers** stacked on top of each other. (previously it was common to only have a single CONV layer always immediately followed by a POOL layer)



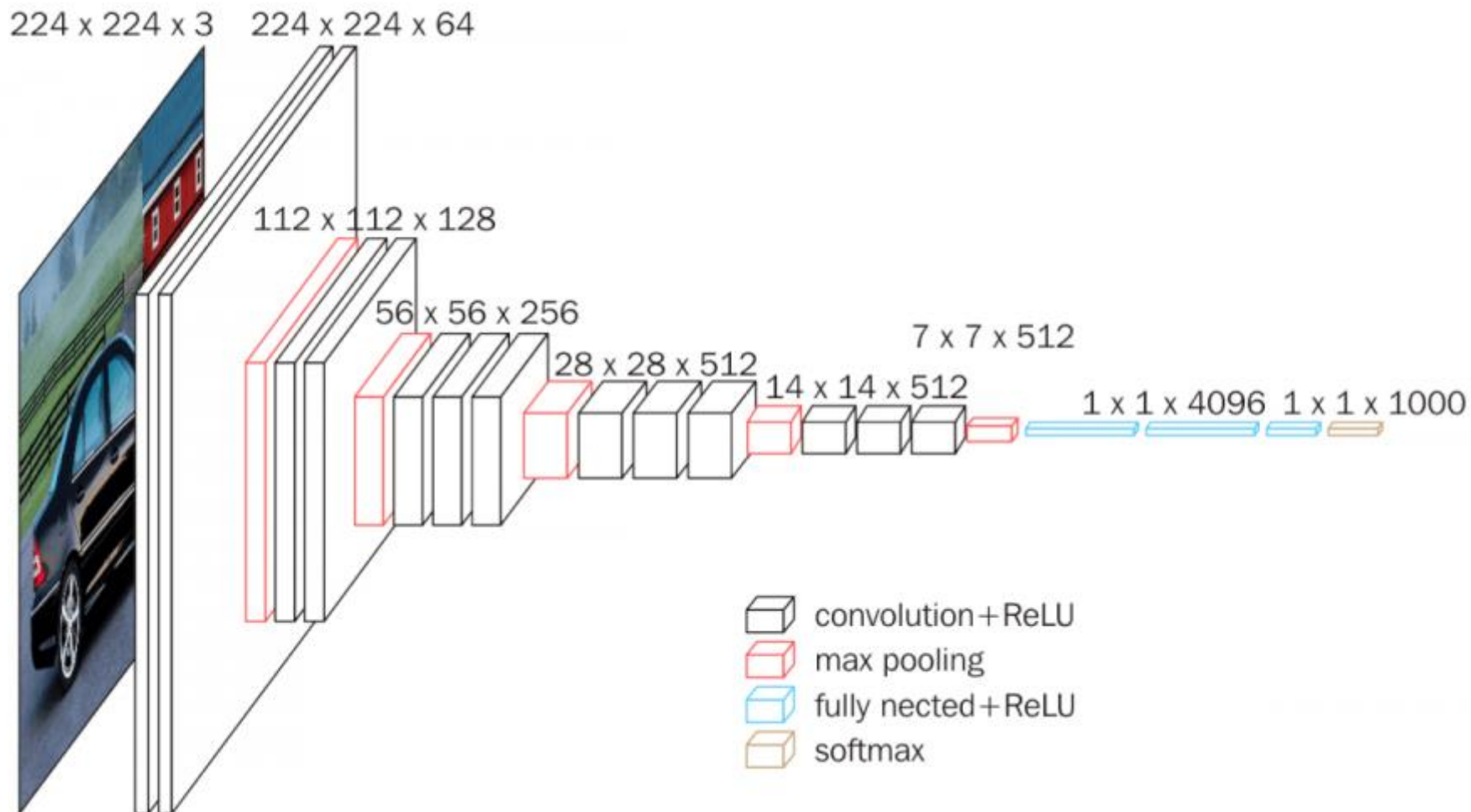
DROPOUT

- With about **60M parameters** to train, the authors proposed another way to reduce overfitting too.
- In dropout, a neuron is dropped from the network with a probability of 0.5.
- When a neuron is dropped, it does not contribute to either forward or backward propagation.
- So every input goes through a different network architecture. As a result, the learnt weight parameters are more robust and do not get over-fitted easily.
- During testing, there is no dropout and the whole network is used, but output is scaled by a factor of 0.5 to account for the missed neurons while training.
- **Dropout increases the number of iterations needed to converge by a factor of 2, but without dropout, AlexNet would overfit substantially.**

VGG16

- It was one of the famous model submitted to ILSVRC-2014.
- It makes the improvement over AlexNet by replacing large kernel-sized filters (11 and 5 in the first and second convolutional layer, respectively) with multiple 3×3 kernel-sized filters one after another.

VGG16



VGG

- There are two major drawbacks with VGGNet:
 - It is *painfully slow* to train.
 - The network architecture weights are quite large.
- VGG16 has a total of **138 million** parameters.



Semnan University

معماری‌های کانلوشنی مدرن (مطالعه موردی)

محمد جواد فدائی‌اسلام

16

TO CREATE BETTER DEEP LEARNING MODEL

- You can just make a bigger model, either in the number of layers, or the number of neurons in each layer.
- But this can often create complications:
 - **Bigger the model, more vulnerable to overfitting.**
 - This is particularly noticeable when the training data is small
 - **Increasing the number of parameters means you need to increase your existing computational resources.**

GOOGLENET

- Dataset: ILSVRC2014 (ImageNet2014)
- 12 times fewer parameters than AlexNet, while being significantly more accurate.
 - AlexNet: 60M parameters
 - GoogLeNet: 4M parameters

Team	Year	Place	Error (top-5)	Uses external data
SuperVision	2012	1st	16.4%	no
VGG	2014	2nd	7.32%	no
GoogLeNet	2014	1st	6.67%	no

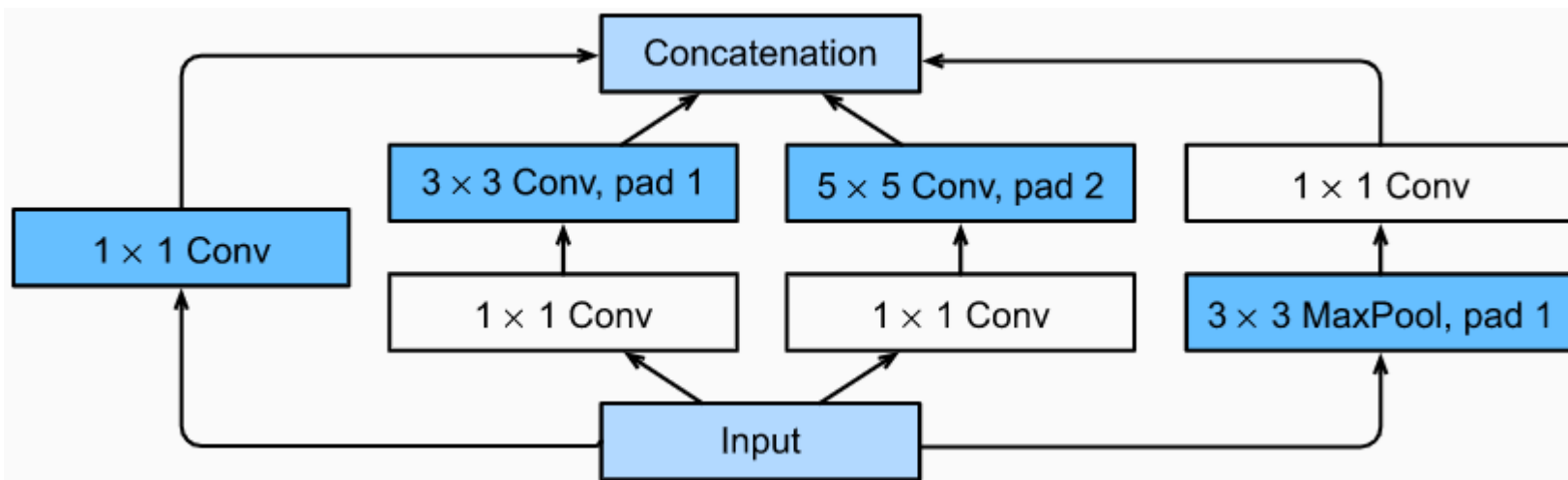
GOOGLENET (KERNEL SIZE)

One focus of GoogLeNet was to address the question of **which kernel size** is best.

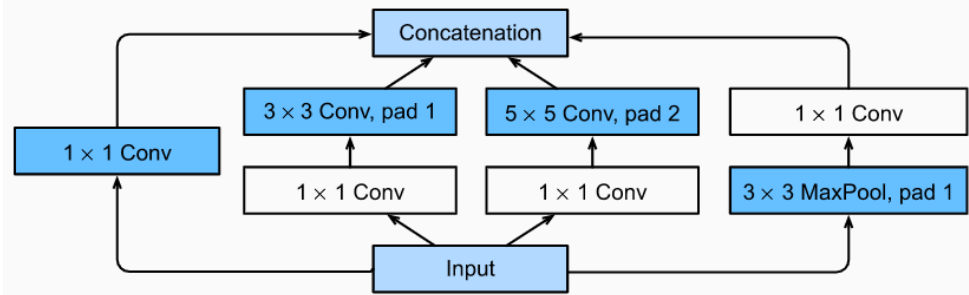
Previous popular networks employed choices as small as 1×1 and as large as 11×11 .

One insight in GoogLeNet was that sometimes it can be advantageous to employ **a combination** of variously-sized kernels.

Inception Block, The basic convolutional block in GoogLeNet



INCEPTION BLOCK

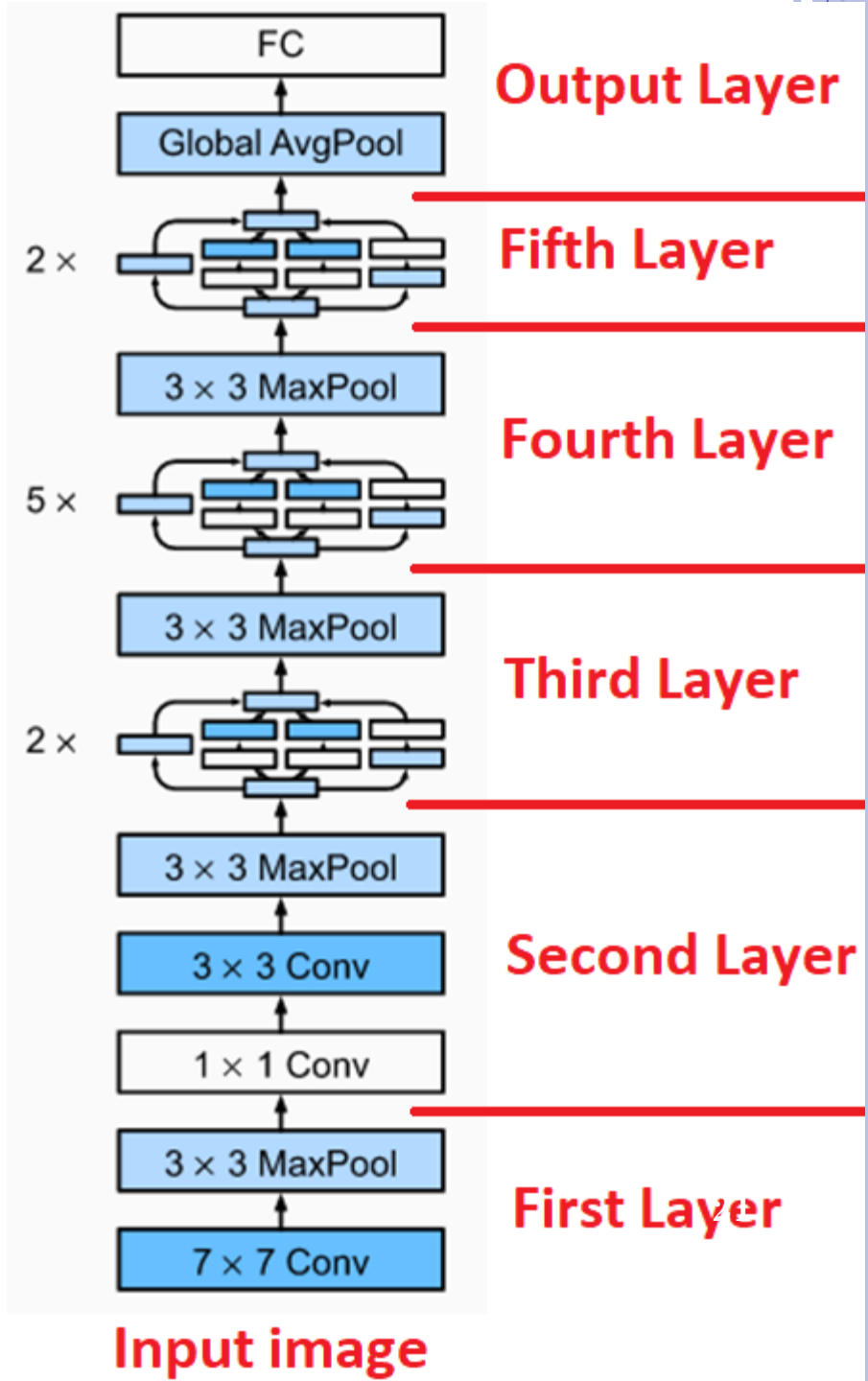


- It consists of four parallel paths.
- The first three paths use 1×1 , 3×3 , and 5×5 conv. layers to extract information from different spatial sizes.
- The middle two paths perform a 1×1 convolution on the input to **reduce the number of channels**.
- The fourth path uses a 3×3 maximum pooling layer, followed by a 1×1 convolutional layer to change the number of channels.
- The four paths all use appropriate padding to give the **input and output the same height** and width.
- Finally, the outputs along each path are concatenated

GOOGLNET

THE NUMBER OF LAYER

- The network is **22 layers** deep when counting only layers with parameters (or 27 layers if we also count pooling).
- The overall number of layers (independent building blocks) used for the construction of the network is about 100



RESNET

- Deep CNNs have led to a series of breakthroughs for image classification.
- Deep networks naturally integrate low/mid/high level features in an end-to-end multilayer fashion,
- The “**levels**” of features can be **enriched** by the number of stacked layers (**depth**).

RESNET

- Driven by the significance of depth, a question arises:

Is learning better networks as easy as stacking more layers?

- An obstacle to answering this question was the notorious problem of **vanishing/exploding** gradients, which prevent convergence from the beginning.
- This problem, however, has been largely addressed by **normalized initialization and intermediate normalization layers**, which enable networks with tens of layers to start converging for stochastic gradient descent (SGD) with backpropagation.

DEGRADATION PROBLEM

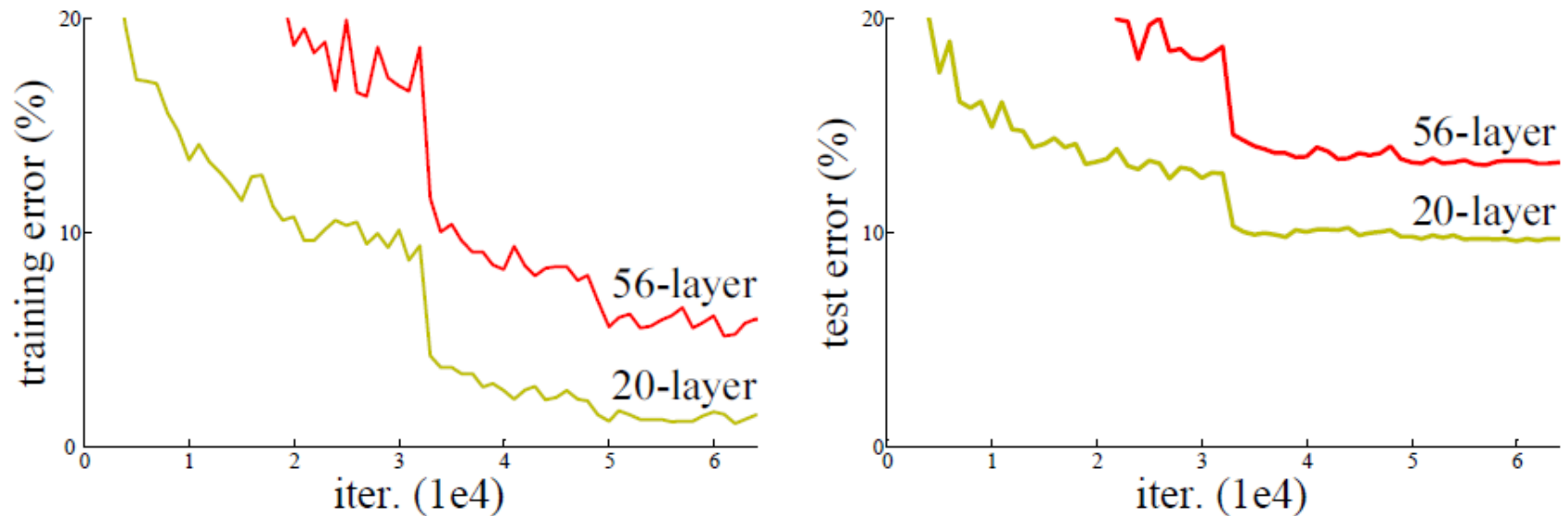
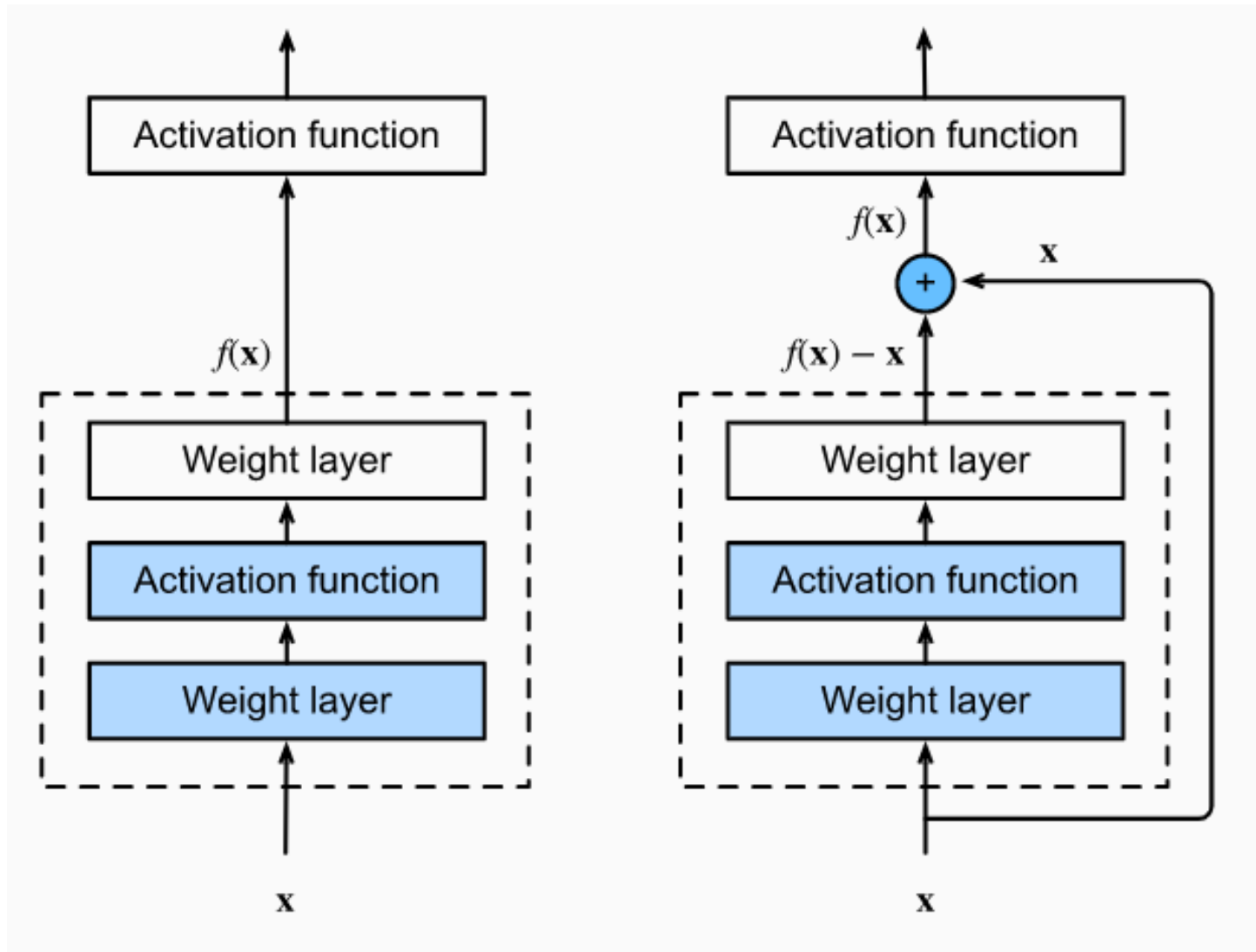


Figure 1. Training error (left) and test error (right) on CIFAR-10 with 20-layer and 56-layer “plain” networks. The deeper network has higher training error, and thus test error. Similar phenomena on ImageNet is presented

RESNET

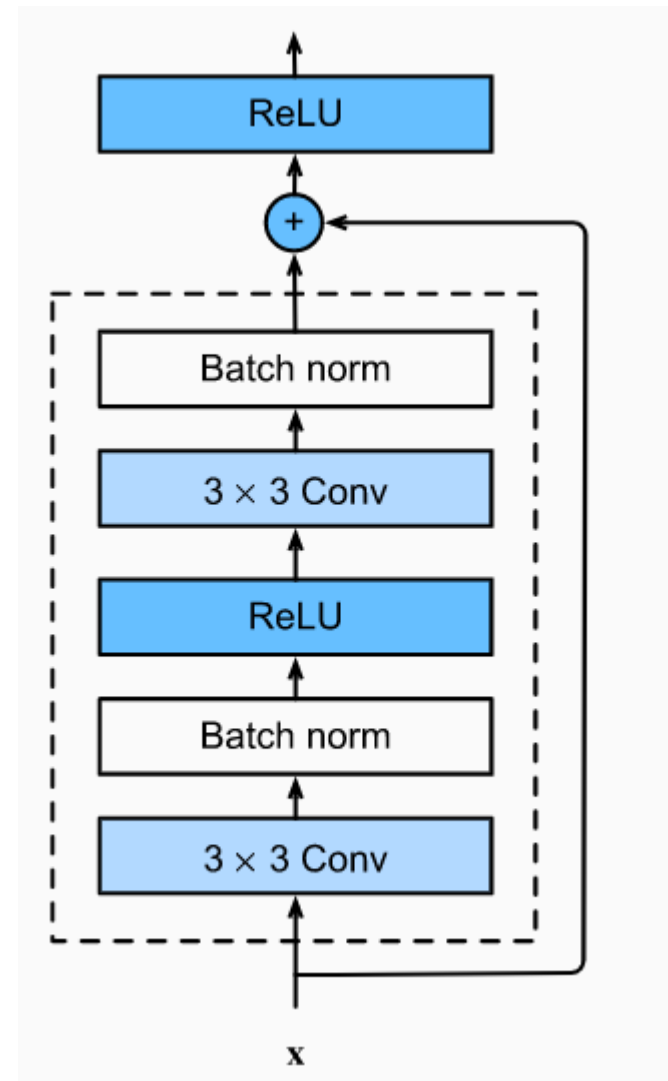
- A widely observed phenomenon in deep learning is the **degradation problem**: increasing the depth of a network leads to a decrease in performance on both test and training data
- Unexpectedly, such degradation is not caused by **overfitting**.

A REGULAR BLOCK (LEFT) AND A RESIDUAL BLOCK (RIGHT)

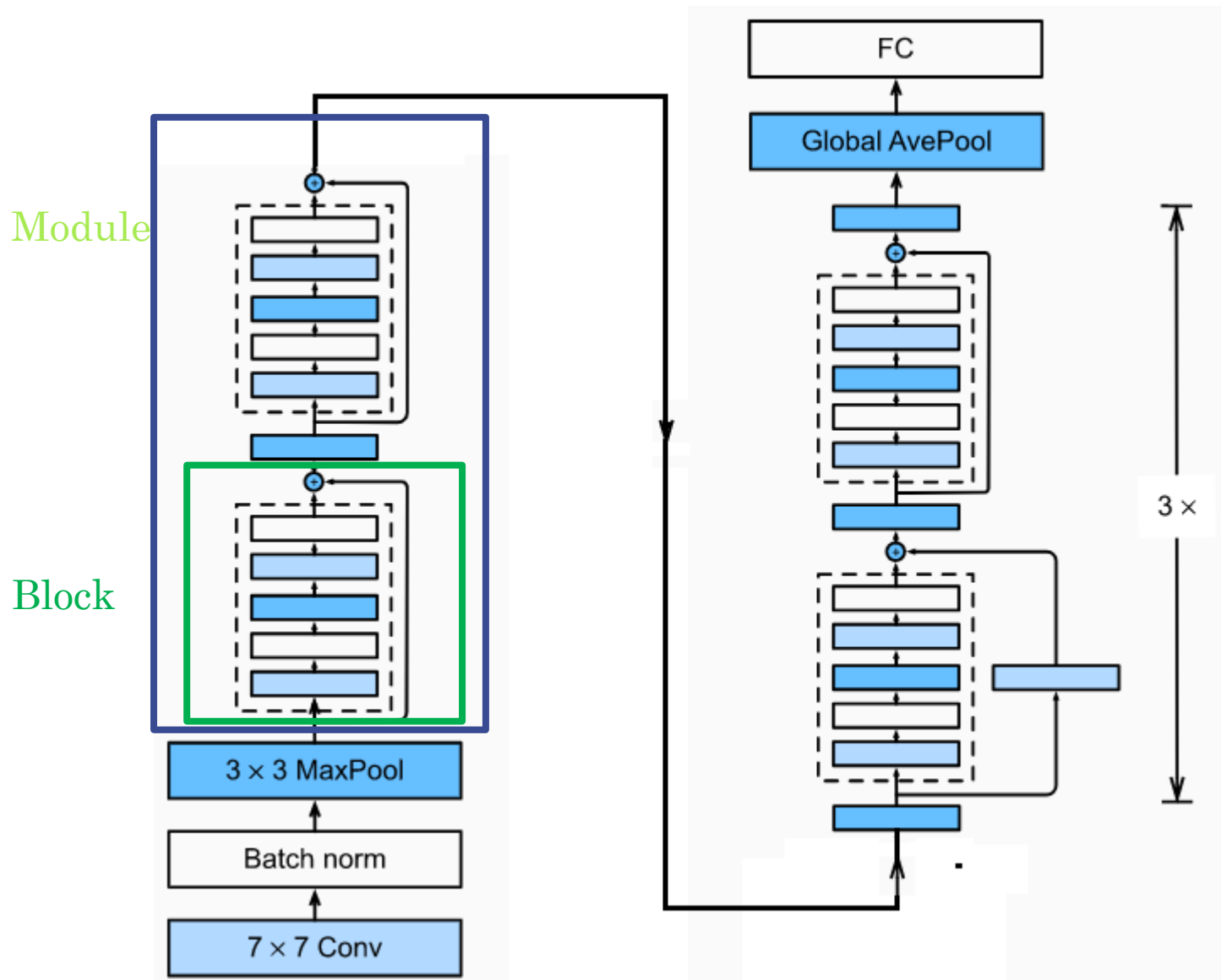


RESNET BLOCK

- ResNet follows VGG's full 3×3 convolutional layer design.
- The residual block has two 3×3 convolutional layers.
- This design requires that the output of the two convolutional layers has to be of the **same shape** as the input, so that they can be added together.

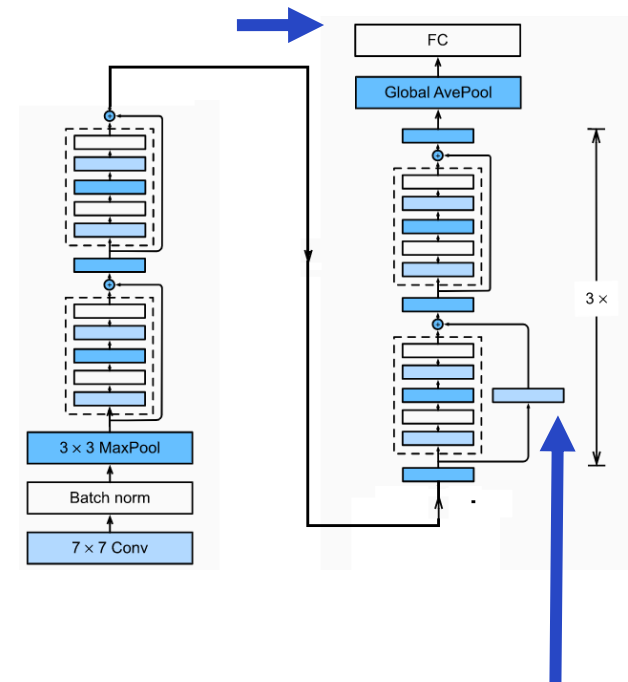


THE RESNET-18 ARCHITECTURE



RESNET MODEL

- ResNet uses **four modules**.
- **Two residual blocks** are used for each module.
- The number of channels in the **first** module is the same as the number of input channels.
- A max pooling layer with a stride of 2 has already been used between two blocks in each module.
- In the first residual block for each of the **subsequent modules**, the number of channels is **doubled** compared with that of the previous module.
- Finally, just like GoogLeNet, we add a **global average pooling** layer, followed by the fully-connected layer output.



RESNET MODELS - LAYERS

- There are 4 convolutional layers in each module (excluding the 1×1 convolutional layer). Together with the first 7×7 convolutional layer and the final fully-connected layer, there are 18 layers in total. Therefore, this model is commonly known as **ResNet-18**.
- By configuring different numbers of channels and residual blocks in the module, we can create different ResNet models, such as the deeper 152-layer ResNet-152.
- Although the main architecture of ResNet is similar to that of GoogLeNet, **ResNet's structure is simpler and easier to modify.**

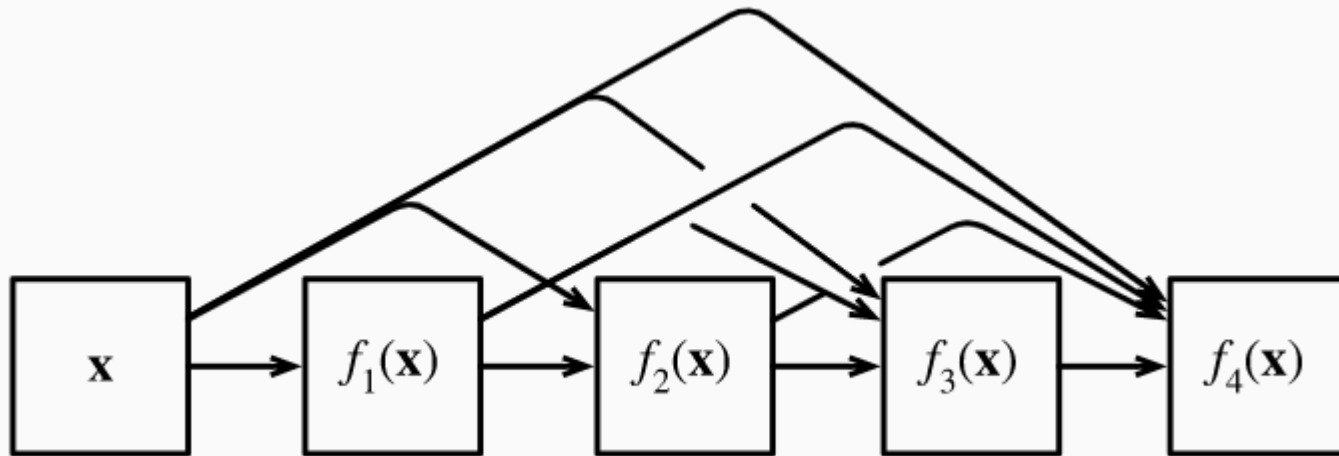
BATCH NORMALIZATION

- Batch normalization is one of the reasons why deep learning has made such outstanding progress in recent years.
- It enables the use of higher learning rates, greatly accelerating the learning process.
- "covariate shift" refers to the change in the distribution of the input values to a learning algorithm.
- For instance, if the train and test sets come from entirely different sources (e.g. training images come from the web while test images are pictures taken on the iPhone), the distributions would differ.
- The reason covariance shift can be a problem is that the behavior of machine learning algorithms can change when the input distribution changes.

BATCH NORMALIZATION

- The basic idea behind batch normalization is to limit covariate shift by normalizing the activations of each layer (transforming the inputs to be mean 0 and unit variance).
- This allows each layer to learn on a more stable distribution of inputs, and would thus accelerate the training of the network.

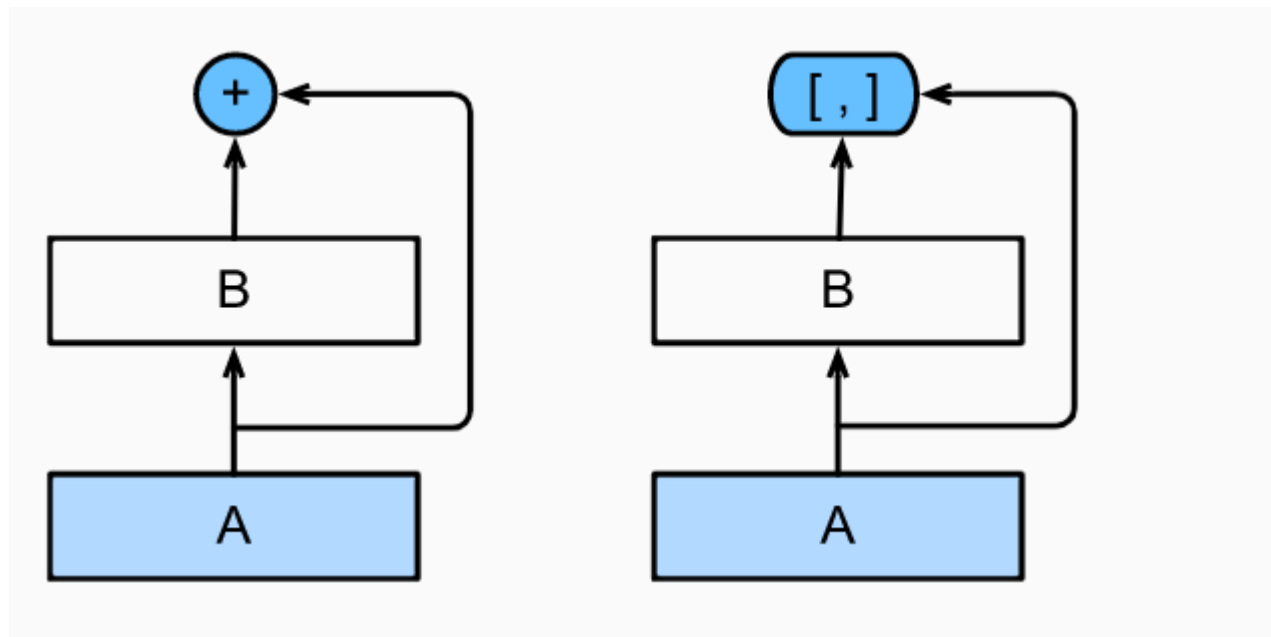
DENSENET2017



- The name DenseNet arises from the fact that the dependency graph between variables becomes quite dense.
- The last layer of such a chain is densely connected to all previous layers.

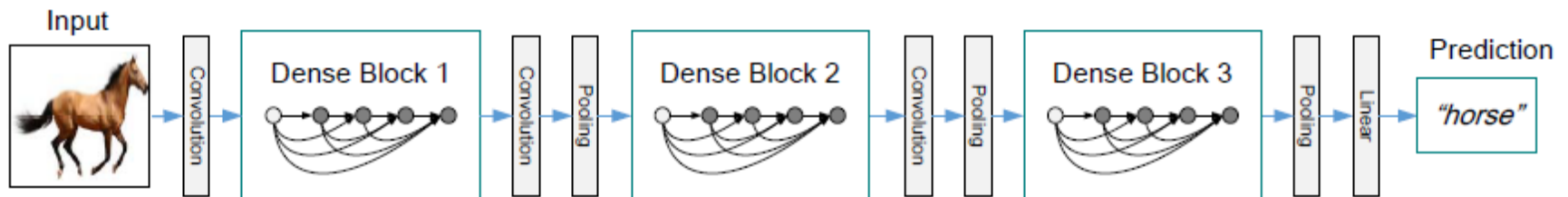
THE MAIN DIFFERENCE BETWEEN RESNET AND DENSENET

- The main difference between ResNet (left) and DenseNet (right) in cross-layer connections: use of addition and use of concatenation.



DENSENET ARCHITECTURE

- The main components that compose a DenseNet are
 - *dense blocks* and
 - *transition layers*.
- The former define how the inputs and outputs are concatenated, while the latter (the layers between two adjacent dense blocks) control the number of channels so that it is not too large.



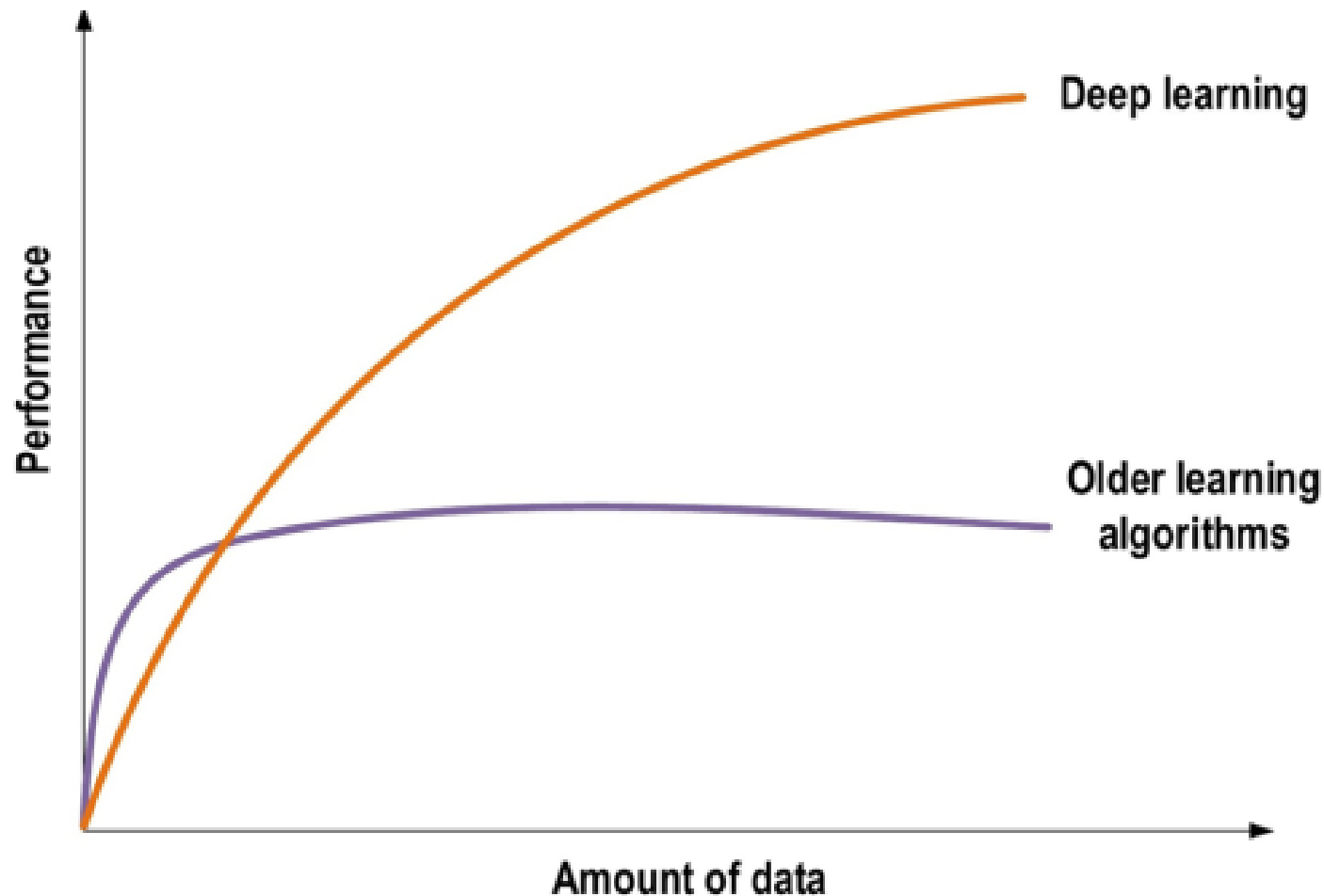
A DenseNet with three dense blocks

NUMBER OF PARAMETERS

- LeNet > 60K
- AlexNet > 60.97M
- VGGNet16 > 138.36M
- GoogLeNet > 4M
- ResNet-18 > 11M
- DenseNet-121 > 8M



THE PERFORMANCE OF DL REGARDING THE AMOUNT OF DATA

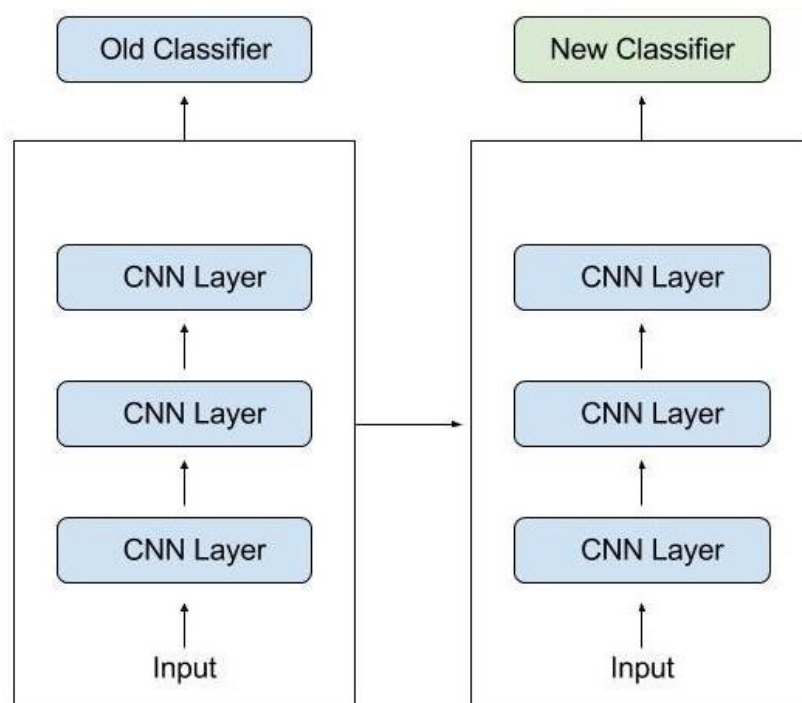


TRANSFER LEARNING

- The general idea is to use the knowledge a model has learned from a task with **a lot of available labeled training data** in a new task that doesn't have **much data**. We transfer the weights that a network has learned at "task A" to a new "task B."
- Instead of starting the learning process from scratch, we start with patterns learned from solving a related task.
- The main advantages of it are saving **training time**, **better performance** of neural networks (in most cases), and **not needing a lot of data**.

TRANSFER LEARNING IN COMPUTER VISION

- In computer vision, for example, neural networks usually try to detect edges in the earlier layers, shapes in the middle layer and some task-specific features in the later layers. In transfer learning, the early and middle layers are used and we only retrain the latter layers.



REFERENCE

- GoogLeNet: C Szegedy et al. “Going Deeper with Convolutions”, *CVPR2015*.
- https://d2l.ai/chapter_convolutional-modern/googlenet.html
- ResNet: K. He; X. Zhang; S. Ren; J. Sun “Deep Residual Learning for Image Recognition”, *CVPR 2016*.
- S. Ioffe, C. Szegedy, “Batch normalization: accelerating deep network training by reducing internal covariate shift”, *arXiv preprint arXiv:1502.03167*.
- DenseNet: G Huang et. al, “Densely Connected Convolutional Networks”, *CVPR 2017*.
- AlexNet: Alex Krizhevsky, I. Sutskever, G. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks”, *Neural Information Processing Systems*, 2012.
- VGG16: K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition”, *ICLR 2015*.