Convolutional Neural Networks II

(Many figures adapted from Stanford CS230, Stanford CS231n, Illinois CS 498 and Berkeley Stat 157)

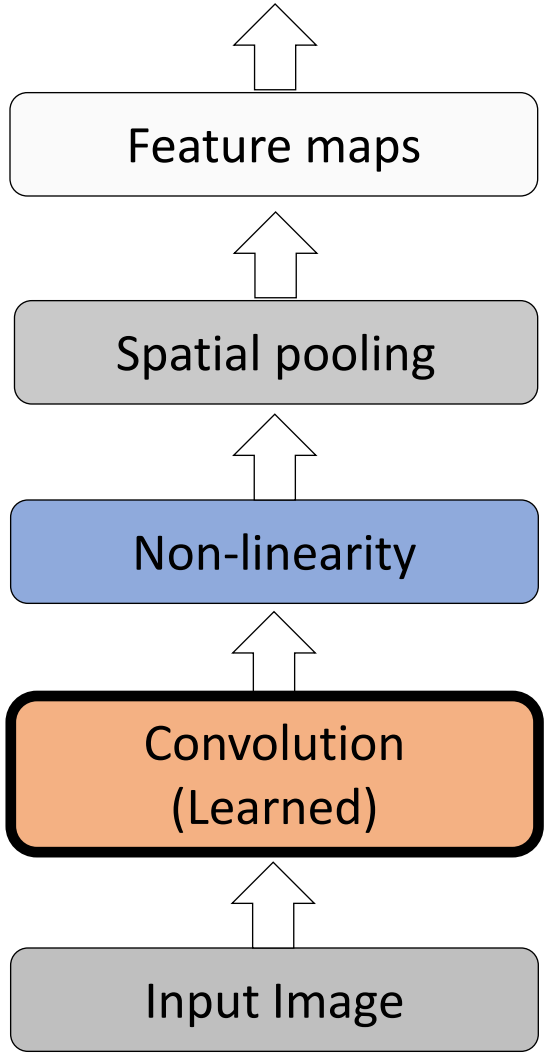
Outline

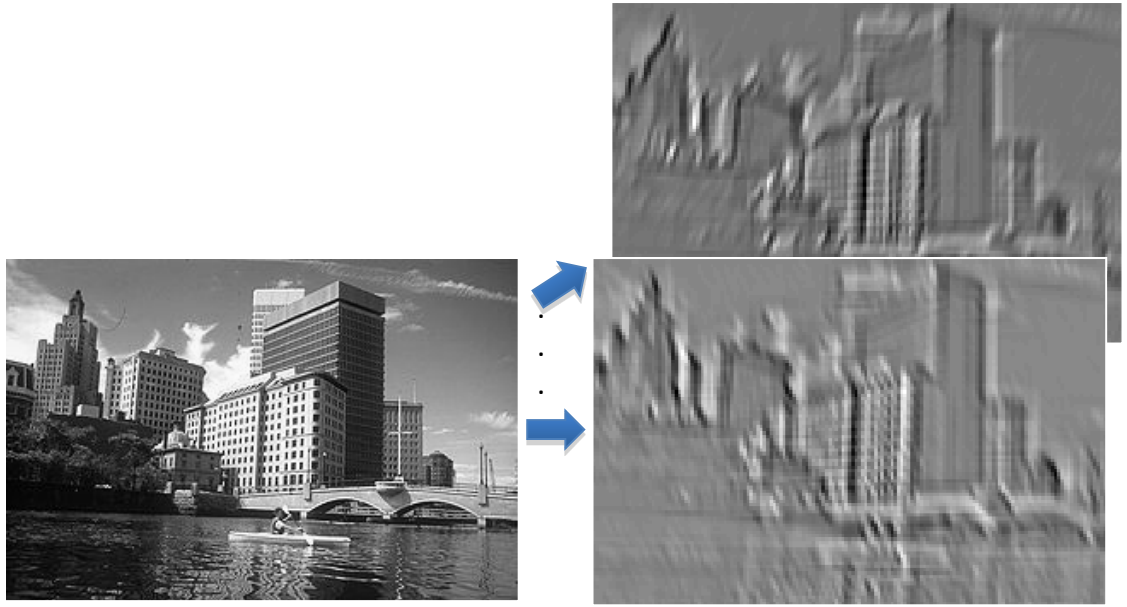
CNN Pipeline

CNN Architectures

- LeNet-5
- AlexNet
- VGG Net
- Network in Network and 1×1 Convolutions
- GoogLeNet and Inception

CNN Pipeline (1/4)



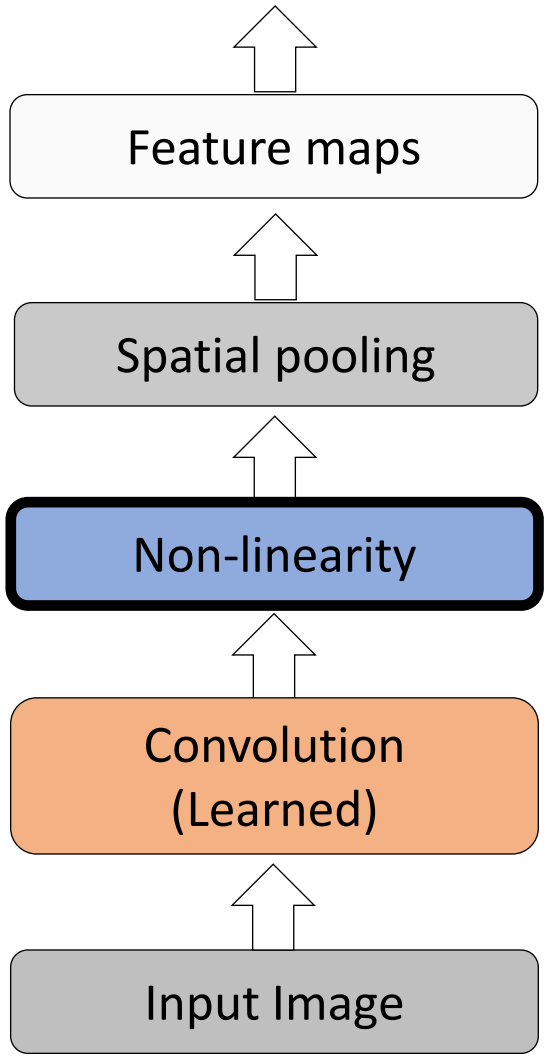


Input

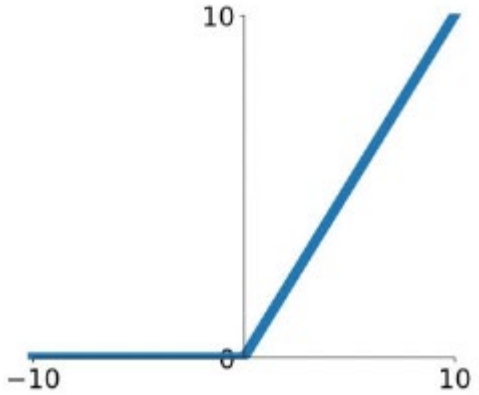
Feature Map

Source: R. Fergus, Y. LeCun

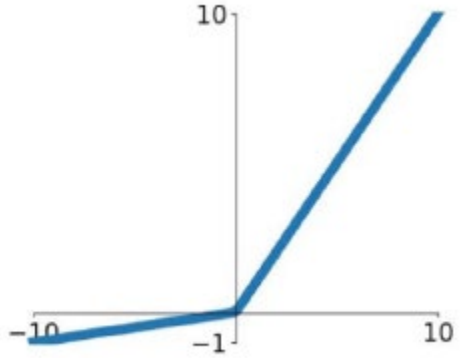
CNN Pipeline (2/4)



Source: R. Fergus, Y. LeCun



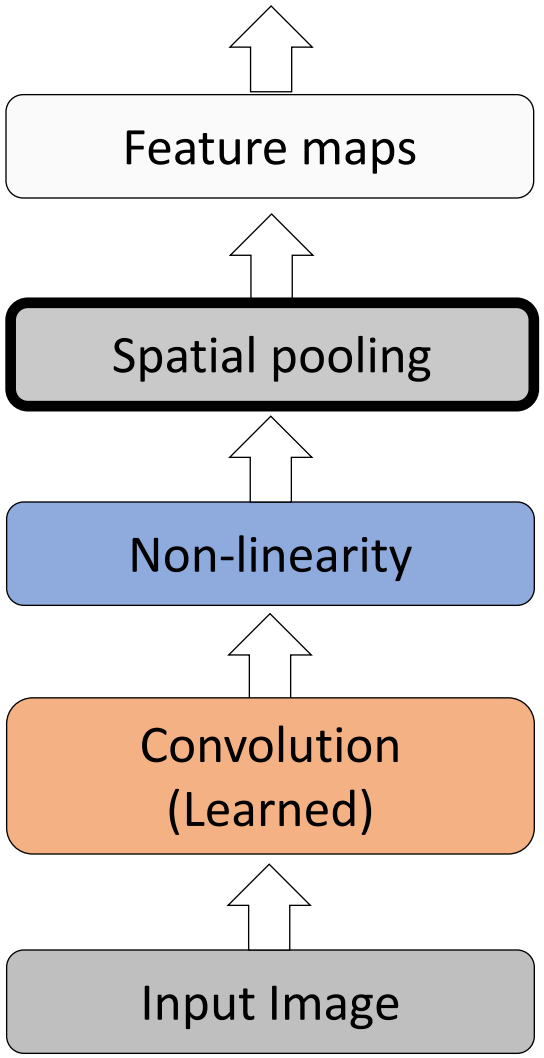
ReLu

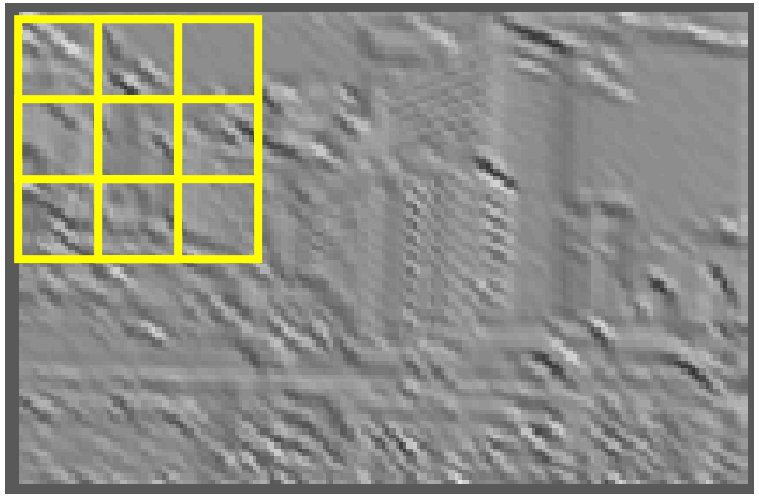


Leaky
ReLu

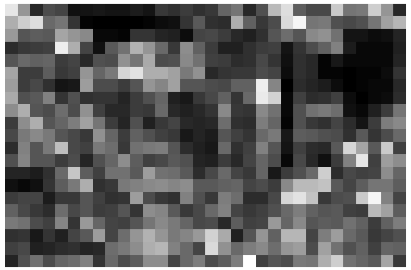
Source: Stanford 231n

CNN Pipeline (3/4)



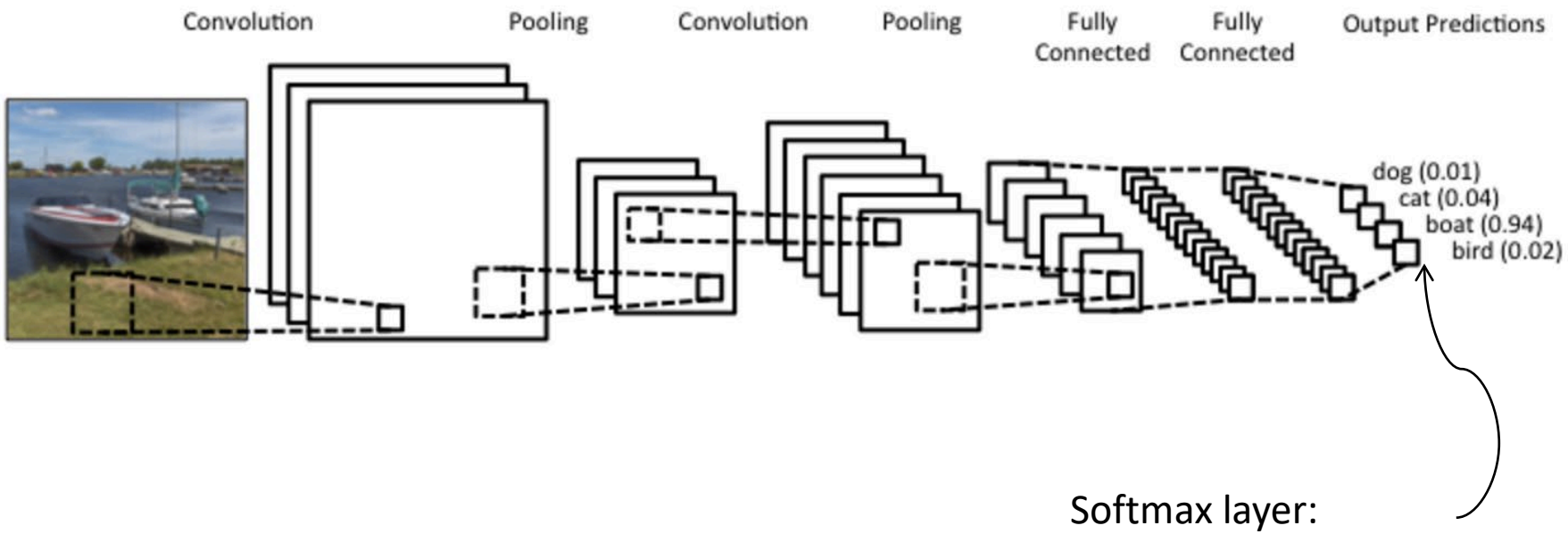


Max
(or Avg)

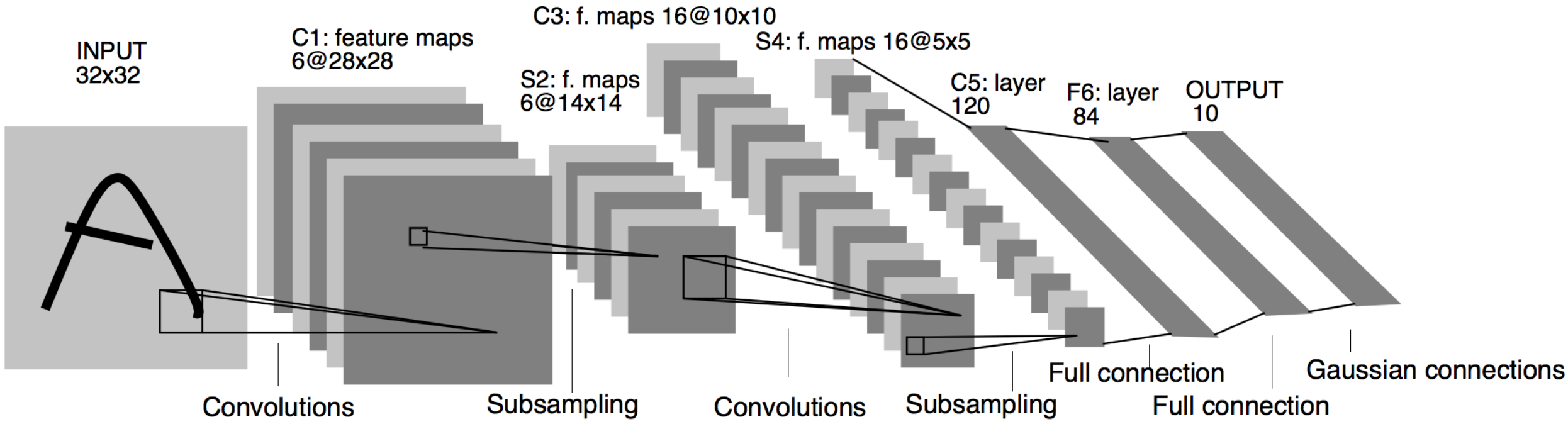


Source: R. Fergus, Y. LeCun

CNN Pipeline (4/4)



LeNet-5 (1/4)



LeCun et al., 1998. Gradient-based learning applied to document recognition

LeNet-5 (2/4)

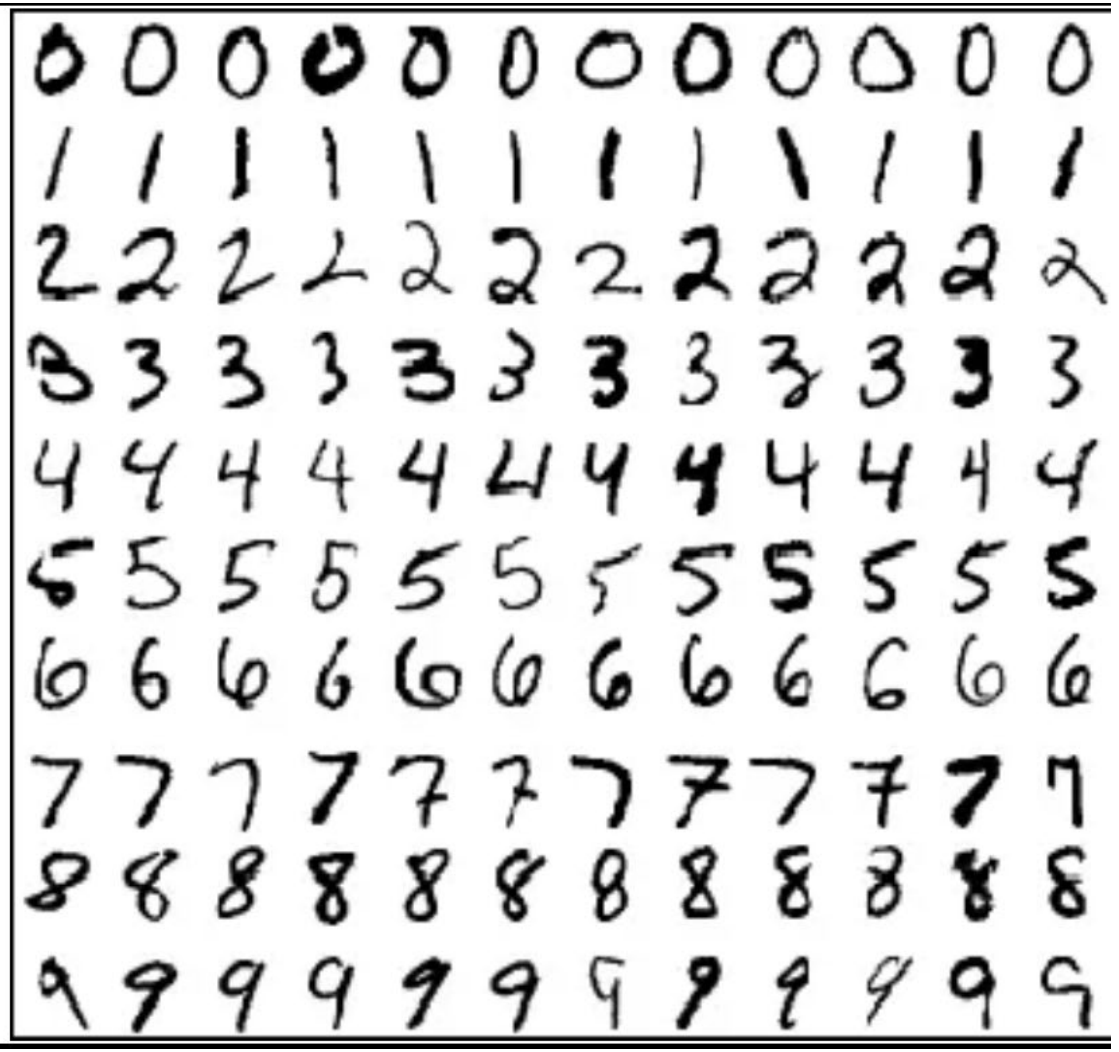
- The goal of LeNet-5 was to recognize handwritten digits.
  - –Input image: 32×32×1
  - –Grayscale(灰階) image
    - ➢ Thus the number of channels is 1
  - –10 classes

LeNet-5 (3/4)

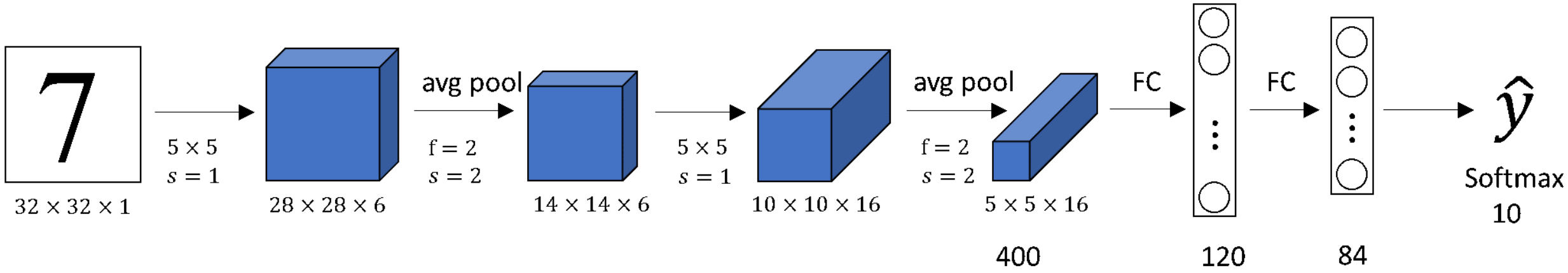
The MNIST database is a large database of handwritten digits that is commonly used for training various image processing systems.

MNIST

- 60,000 training data
- 10,000 test data
- 28 x 28 images
- 10 classes



LeNet-5 (4/4)



60k parameters

$n_{h,}\, n_w \downarrow$ $n_{c,} \uparrow$

conv pool conv pool FC FC output

The architecture of the LeNet−5 neural network leads to a decrease of height and width of volume and to increase the number of channels

Summary: LeNet-5 (1/2)

- LeNet-5 is the first (第一個) convolutional neural network.
- LeNet-5 consists of one or more convolutional layers followed by a pooling layer and then some fully-connected layers and ends up with an output layer, which is a softmax layer.

⇒ This may be the key feature of deep learning for images since this paper.

- The number of channels of LeNet-5 increases.
  - It goes from 1 to 6 to 16.
- It has a small number of parameters – 60,000.
  - Today we use neural networks that have from 10 million to 100 million parameters.

Summary: LeNet-5 (2/2)

Some other notes on LeNet-5:

- At the time, there was no GPU to help with training, and even CPUs were slow.
- Back then, people used sigmoid and tanh nonlinearities.
- The authors used average(平均) pooling rather than max pooling.

AlexNet (1/4)

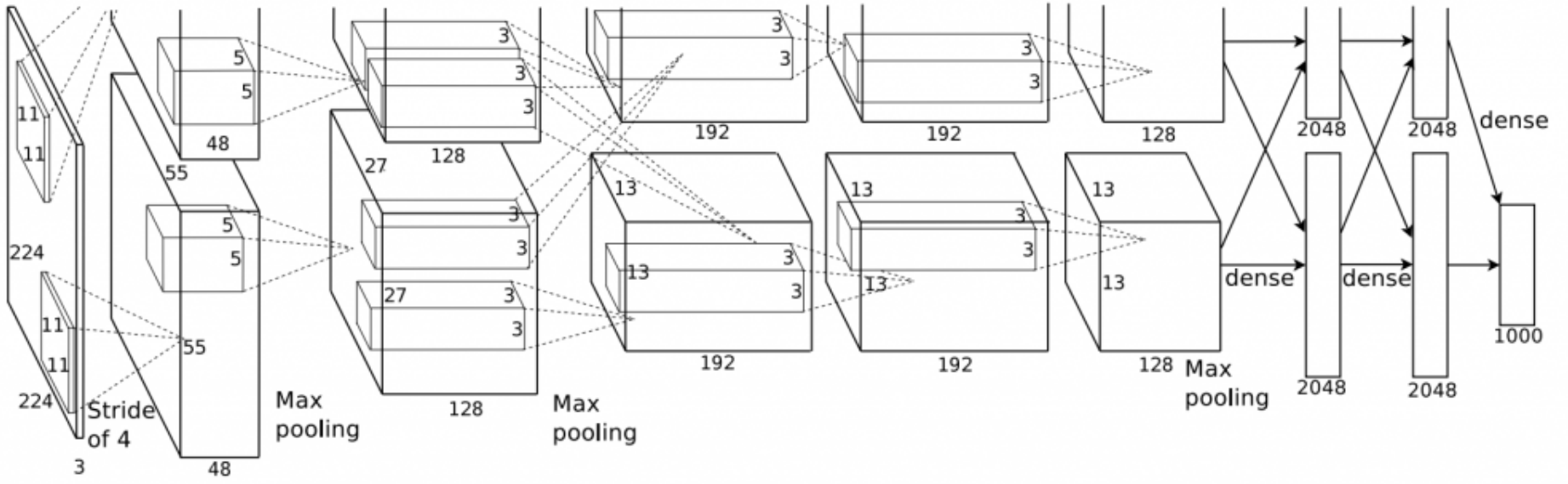
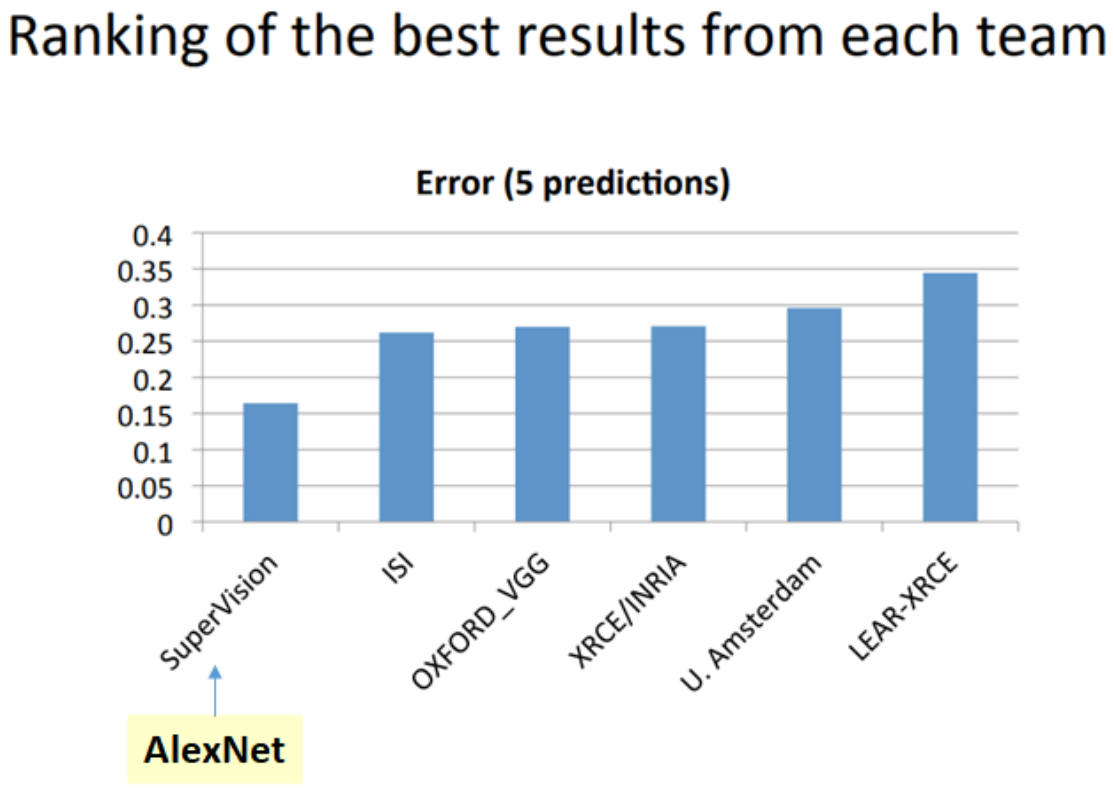


Figure 2: An illustration of the architecture of our CNN, explicitly showing the delineation of responsibilities between the two GPUs. One GPU runs the layer-parts at the top of the figure while the other runs the layer-parts at the bottom. The GPUs communicate only at certain layers. The network's input is 150,528-dimensional, and the number of neurons in the network's remaining layers is given by 253,440–186,624–64,896–64,896–43,264–4096–4096–1000.

Krizhevsky et al., 2012. ImageNet classification with deep convolutional neural networks

AlexNet (2/4)

- AlexNet achieved a top-5 error of 15.3% in the ImageNet 2012 Challenge.
  - –It was 10.8% lower than the second place.



What is the ImageNet Challenge?

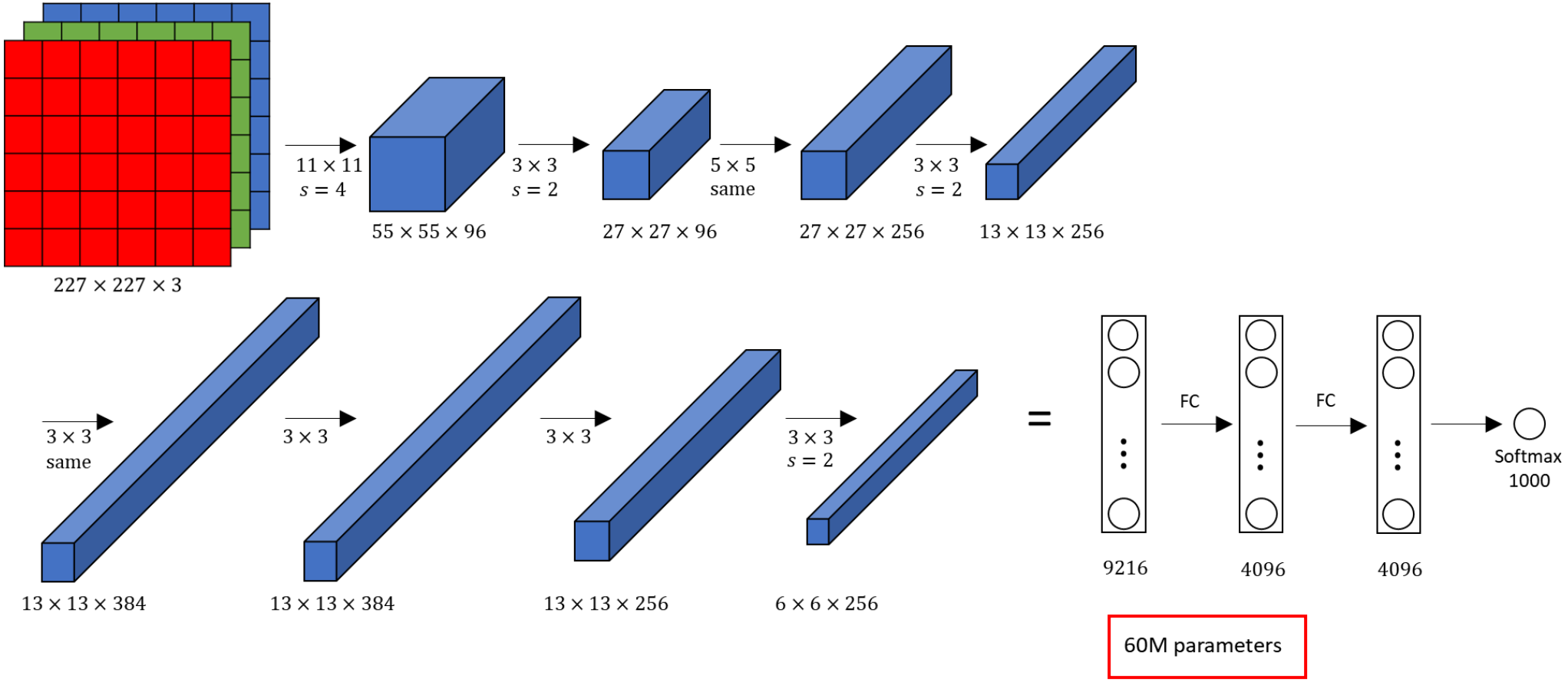
- ~14 million labeled images, 20k classes
- Images gathered from Internet
- Human labels via Amazon MTurk
- ImageNet Large-Scale Visual Recognition Challenge (ILSVRC): 1.2 million training images, 1000 classes

www.image-net.org/challenges/LSVRC/

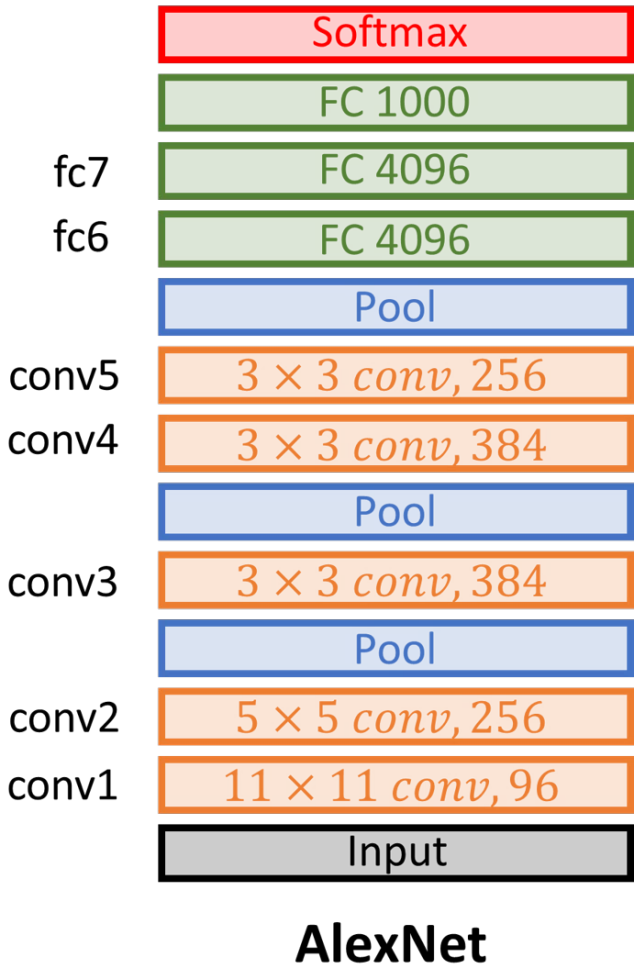
AlexNet (3/4)

- An input to this neural network is 227×227×3.
  - A color image
    - 3 channels
  - 1000 classes
- AlexNet has 8 layers — 5 convolutional and 3 fully-connected.
  - Three max-pooling layers.
- AlexNet was the first to use Rectified Linear Units (ReLUs) as activation functions.(是第一個使用ReLUs的)

AlexNet (4/4)


227 × 227 × 3
11 × 11
s = 4
55 × 55 × 96
3 × 3
s = 2
27 × 27 × 96
5 × 5
same
27 × 27 × 256
3 × 3
s = 2
13 × 13 × 256
3 × 3
same
13 × 13 × 384
3 × 3
13 × 13 × 384
3 × 3
13 × 13 × 256
3 × 3
s = 2
6 × 6 × 256
=
9216
FC
4096
FC
4096
Softmax
1000
60M parameters

AlexNet VS. LeNet-5 (1/2)



AlexNet

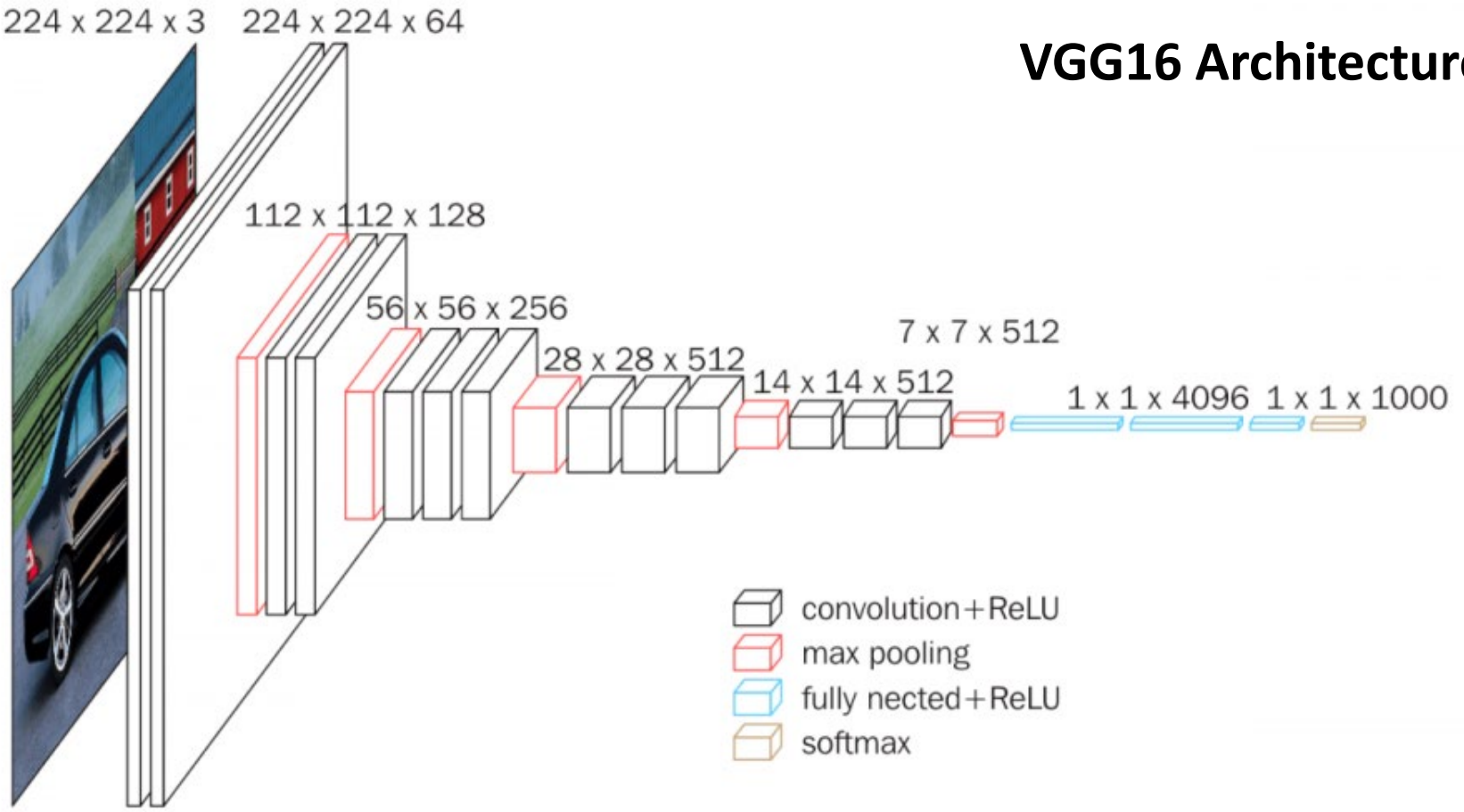
- AlexNet is similar to LeNet-5 but much larger.
- LeNet-5 has about 60,000 parameters.
- AlexNet has about 60 million parameters.
  - Computationally expensive

⇒ Use GPUs during training

LeNet-5 VS. AlexNet (2/2)

- Some other tricks in AlexNet
  - –Change activation function from sigmoid to ReLu
    - ➢Avoid vanishing gradient 避免梯度消失
  - –Add a dropout layer after two hidden dense layers
    - ➢Better robustness/regularization
  - –Data augmentation
    - ➢Better robustness/regularization

VGG 16 (1/4)



VGG16 Architecture

Simonyan & Zisserman 2015. Very deep convolutional networks for large-scale image recognition

VGG 16 (2/4)

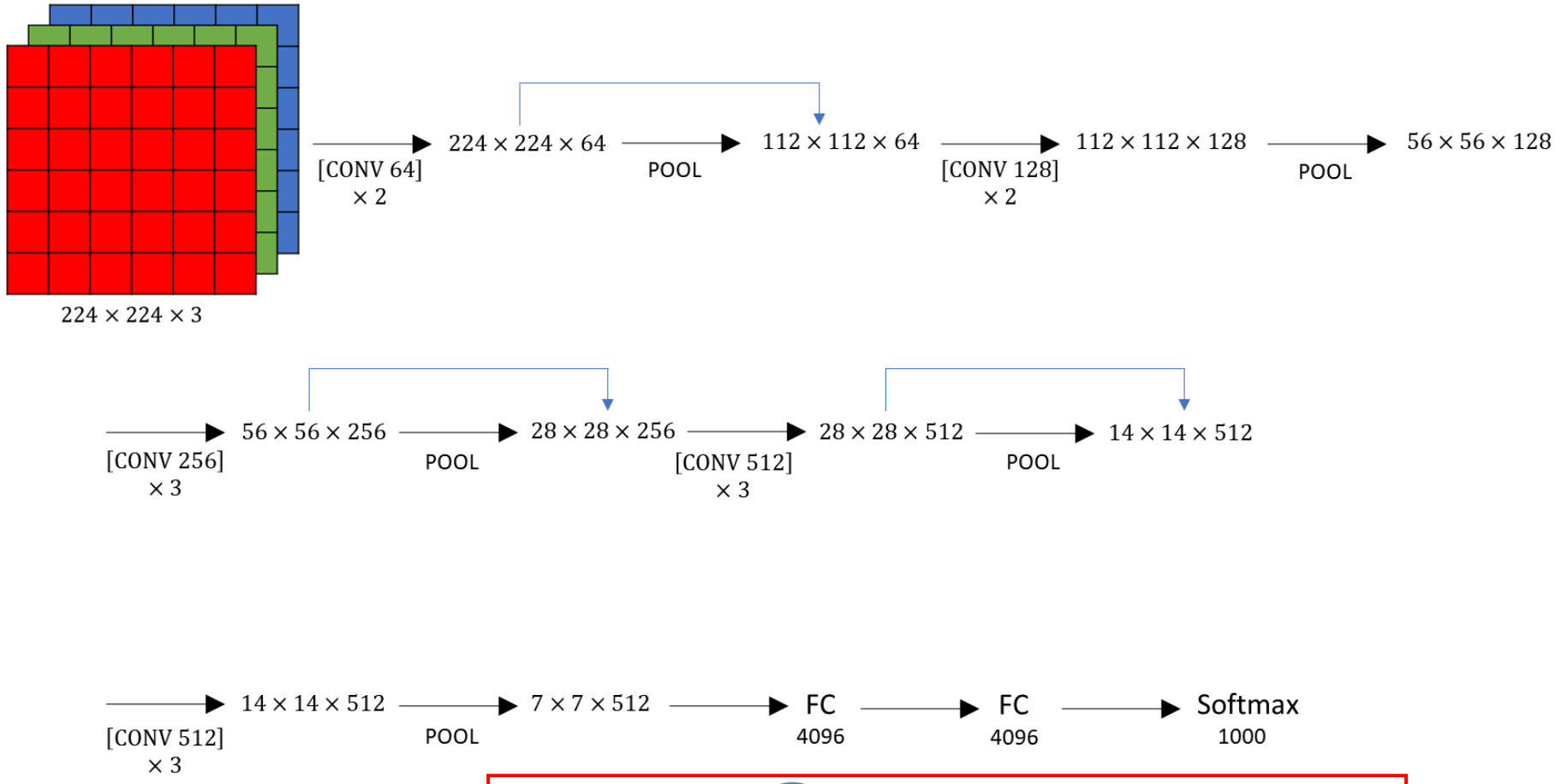
- VGG Net is invented by VGG (Visual Geometry Group) from the University of Oxford.
- VGG Net is the first one to achieve less than 10% error rate for ImageNet competition.

VGG 16 (3/4)

- An input to this neural network is 224×224×3.
  - A color image
    - 3 channels
  - 1000 classes
- The VGG 16 network has 16 layers — 13 convolutional and 3 fully-connected.
  - The 16 in VGG16 refers to it has 16 layers that have weights.

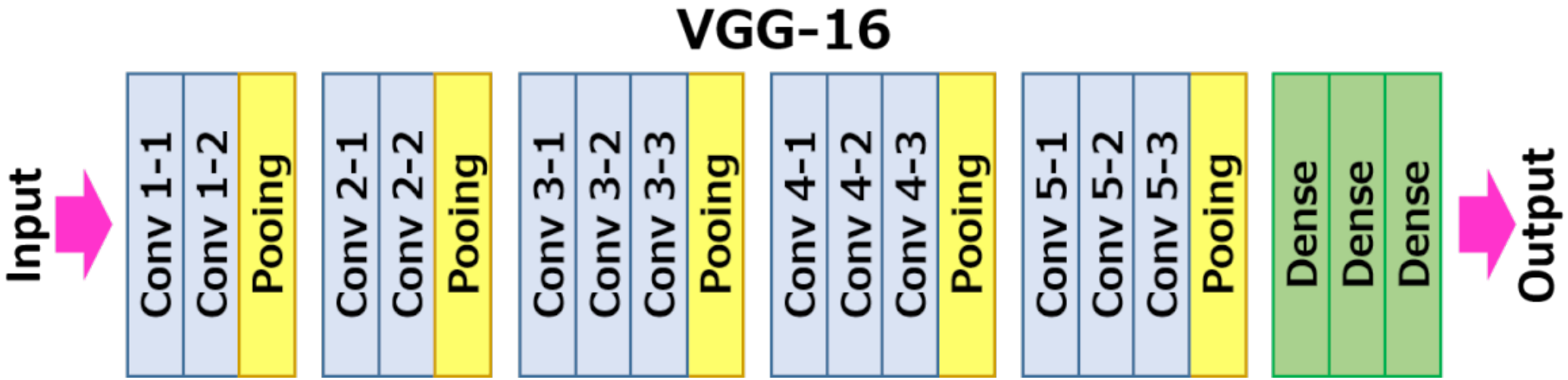
VGG 16 (4/4)

CONV = 3 x 3 filter, s=1, same MAX-POOL = 2 x 2, s=2



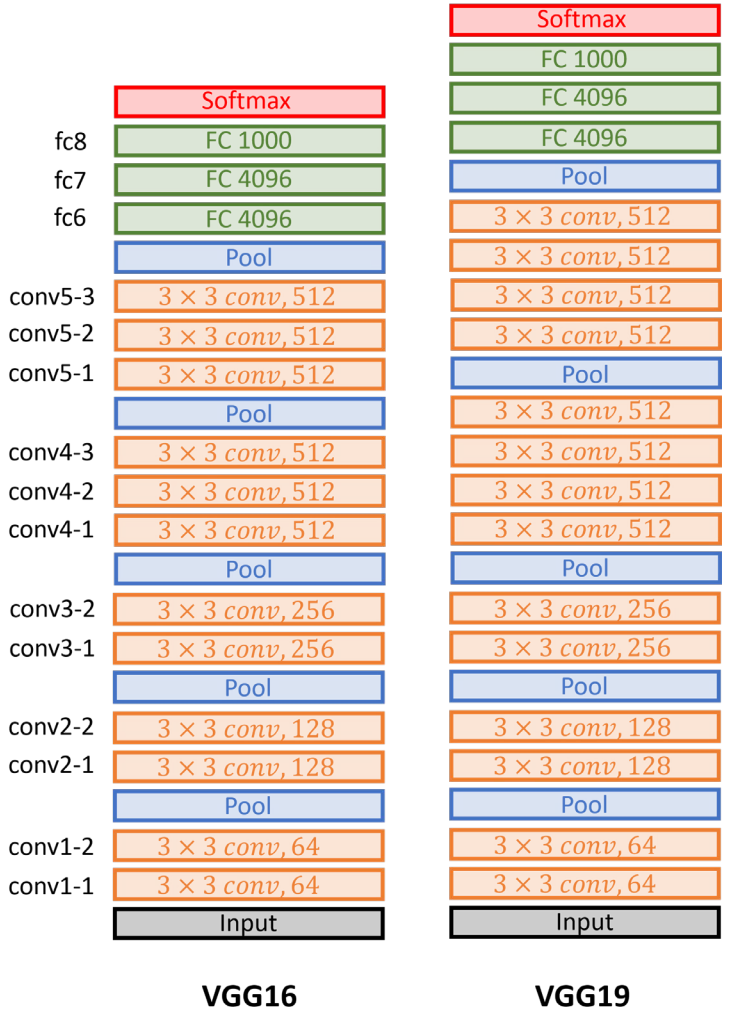
Summary: VGG 16

- Compared to the AlexNet network, VGG 16 network has smaller filters but deeper layers.
- All convolutional layers are divided into 5 groups and each group is followed by a max-pooling layer.



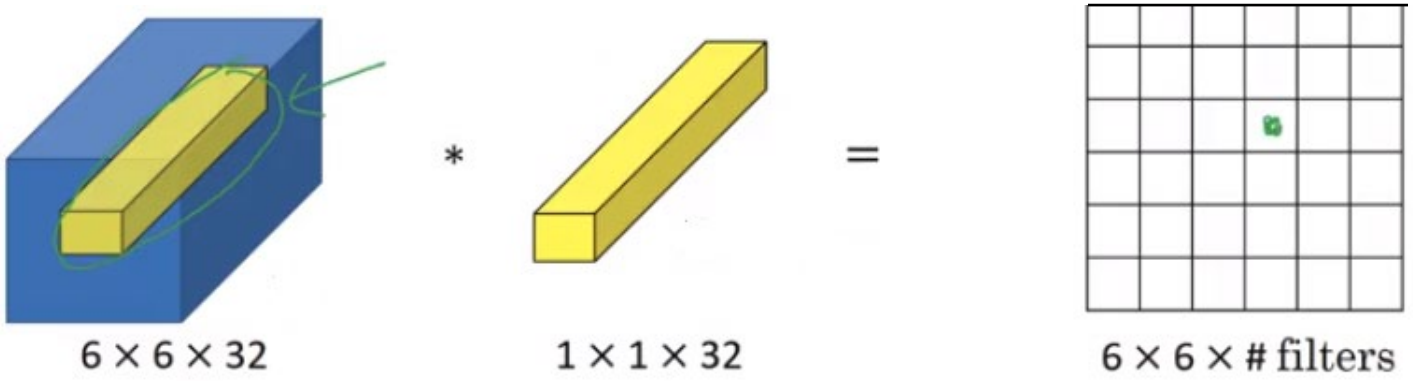
VGG 16 and VGG 19

The VGG Net has two versions, VGG16 and VGG19, corresponding to the number of layers in the network.



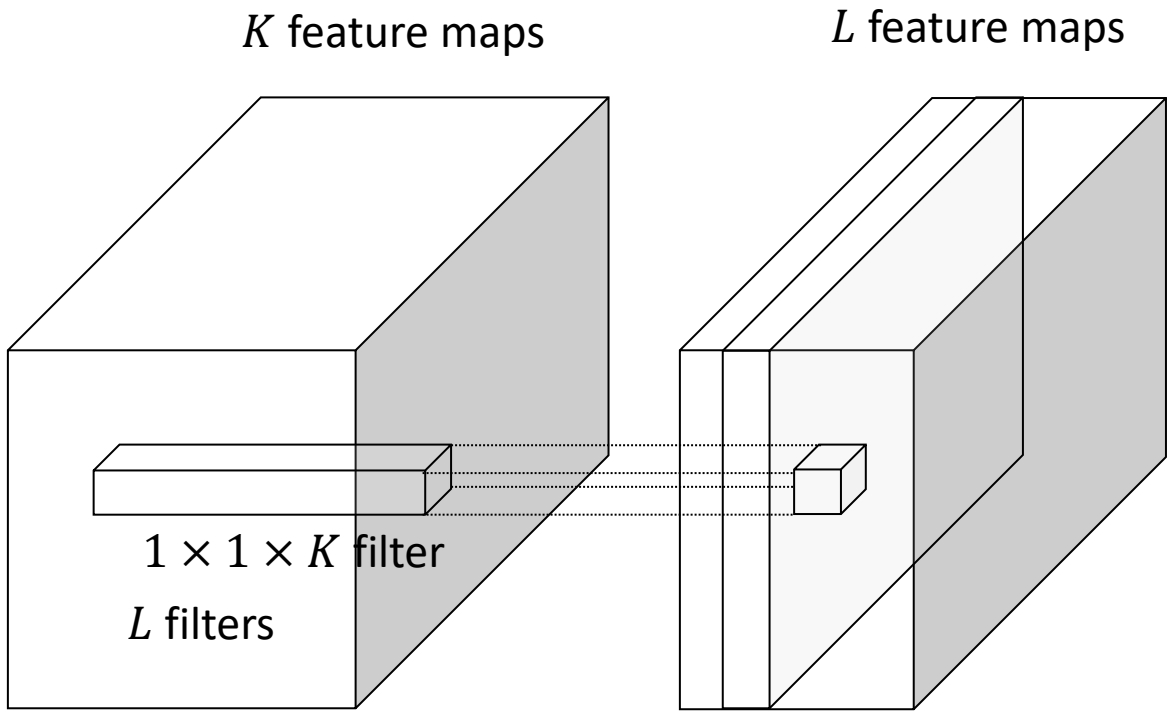
Network in Network and 1×1 Convolutions (1/5)

- Network-in-network (NiN) uses 1x1 convolutions to provide more combinational power to the features of a convolutional layers.



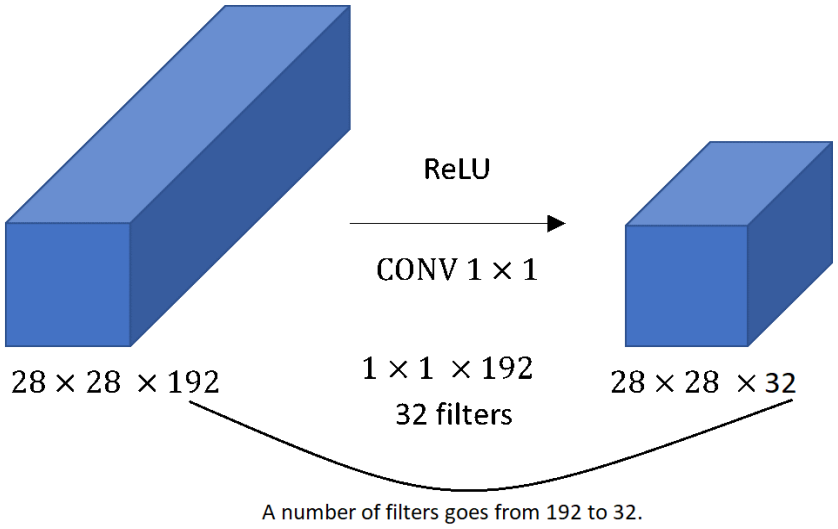
Lin et al., 2013. Network in network

Network in Network and 1×1 Convolutions (2/5)



1×1 conv layer

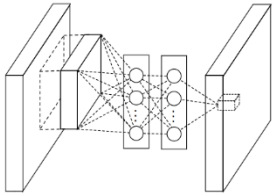
Example: K = 192, L = 32



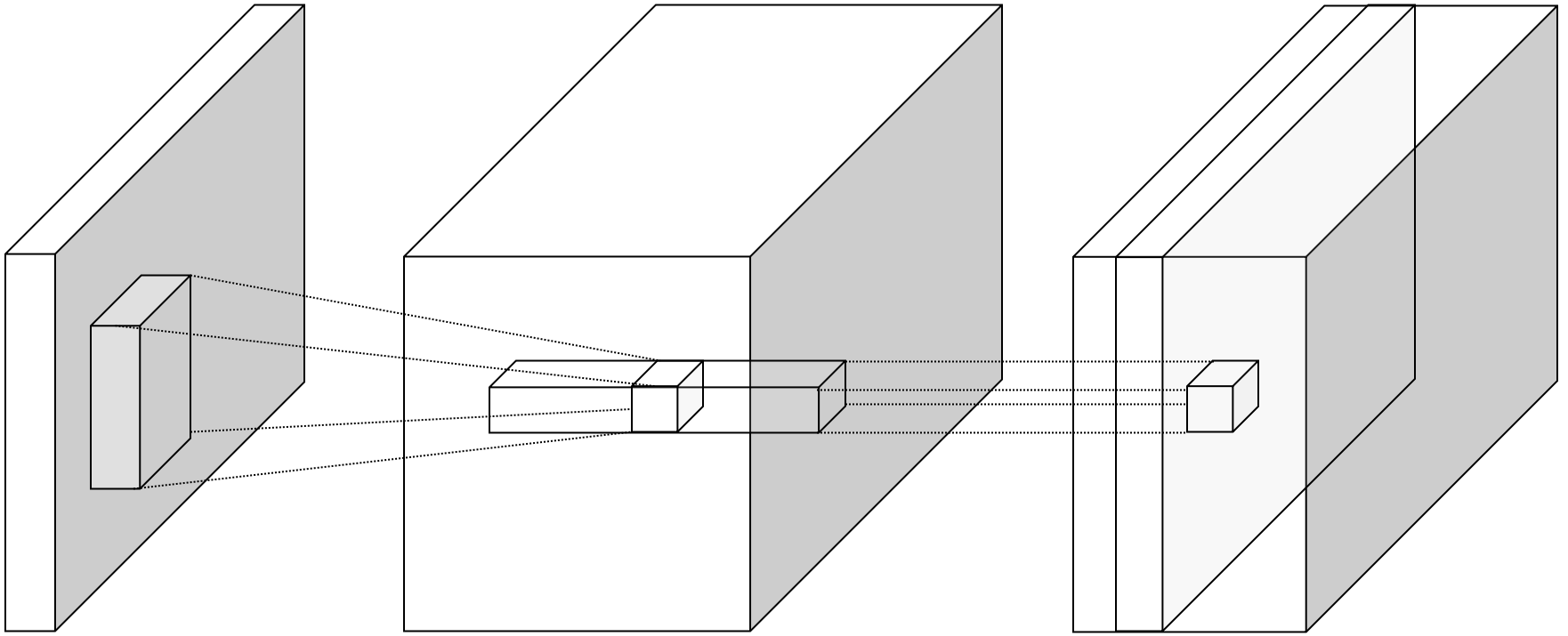
Network in Network and 1×1 Convolutions (3/5)

- Here 1×1 convolutions are used to spatially combine features across features maps after convolution.
  - So they effectively use very few parameters, shared across all pixels of these features.
- One way to think about the 1×1 convolution is that it basically has a fully connected neuron network, and the 1×1 convolution is also called Network in Network.

Network in Network and 1×1 Convolutions (4/5)



"network in network"



M. Lin, Q. Chen, and S. Yan, Network in network, ICLR 2014

Network in Network and 1×1 Convolutions (5/5)

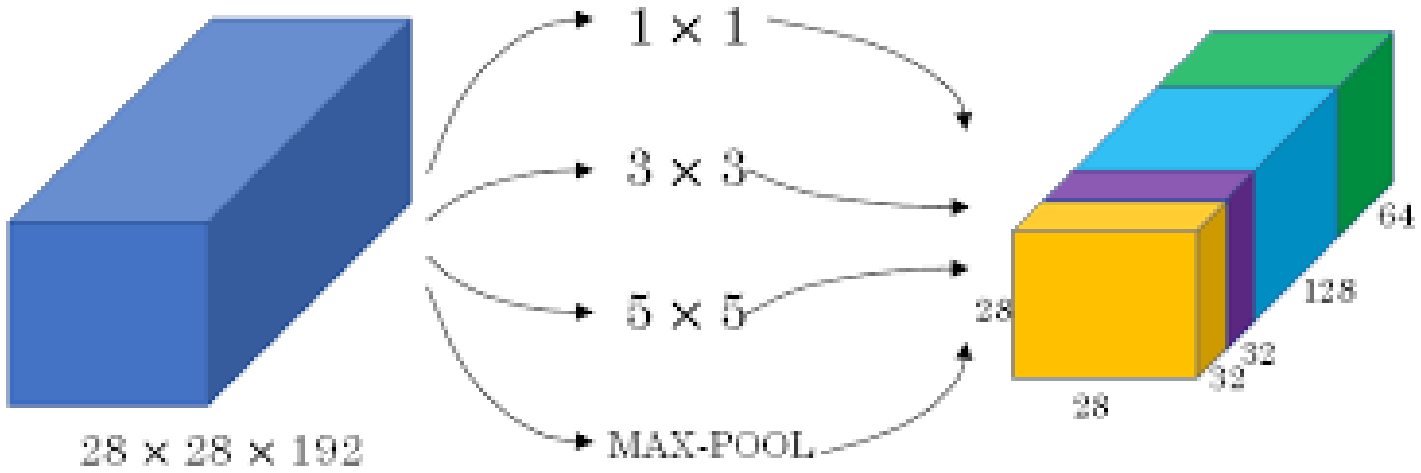
- Though the Network in Network architecture is not used widely, this idea of a 1×1 convolution has influenced many other neural network architectures, including the inception network.

GoogLeNet and Inception (1/7)

- GoogLeNet devised a sparse connection between nodes, called inception modules.
- It also utilized filters of different sizes to capture details at varied scales(5×5, 3×3, 1×1).

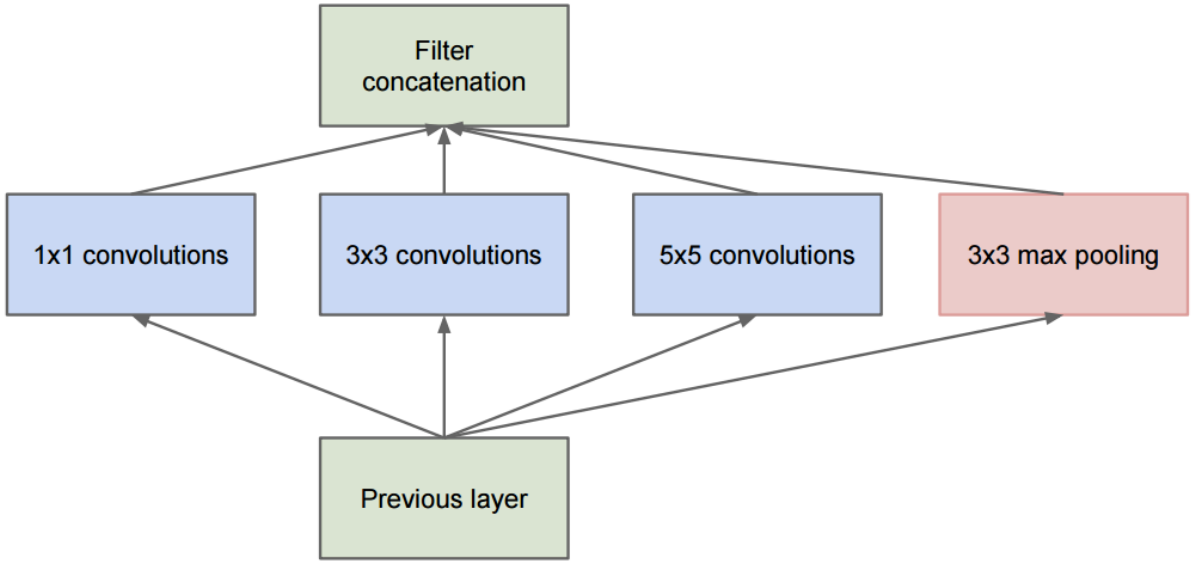
GoogLeNet and Inception (2/7)

Motivation for inception network



GoogLeNet and Inception (3/7)

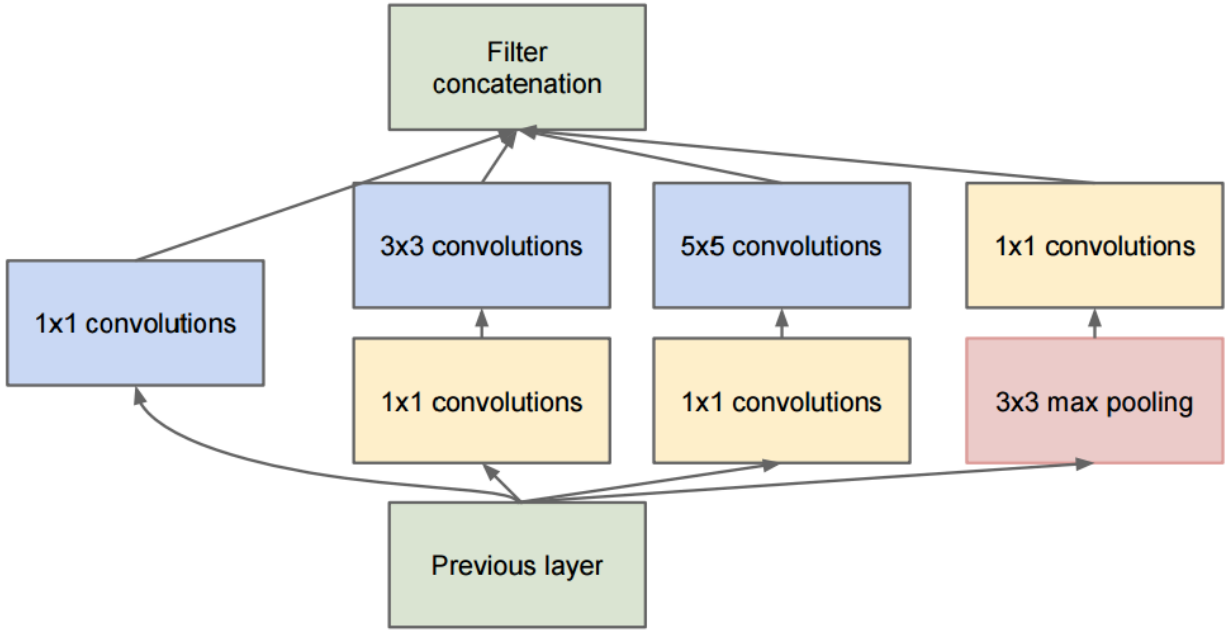
- The Inception Module
  - Parallel paths with different receptive field sizes and operations are meant to capture sparse patterns of correlations in the stack of feature maps



C. Szegedy et al., Going deeper with convolutions, CVPR 2015

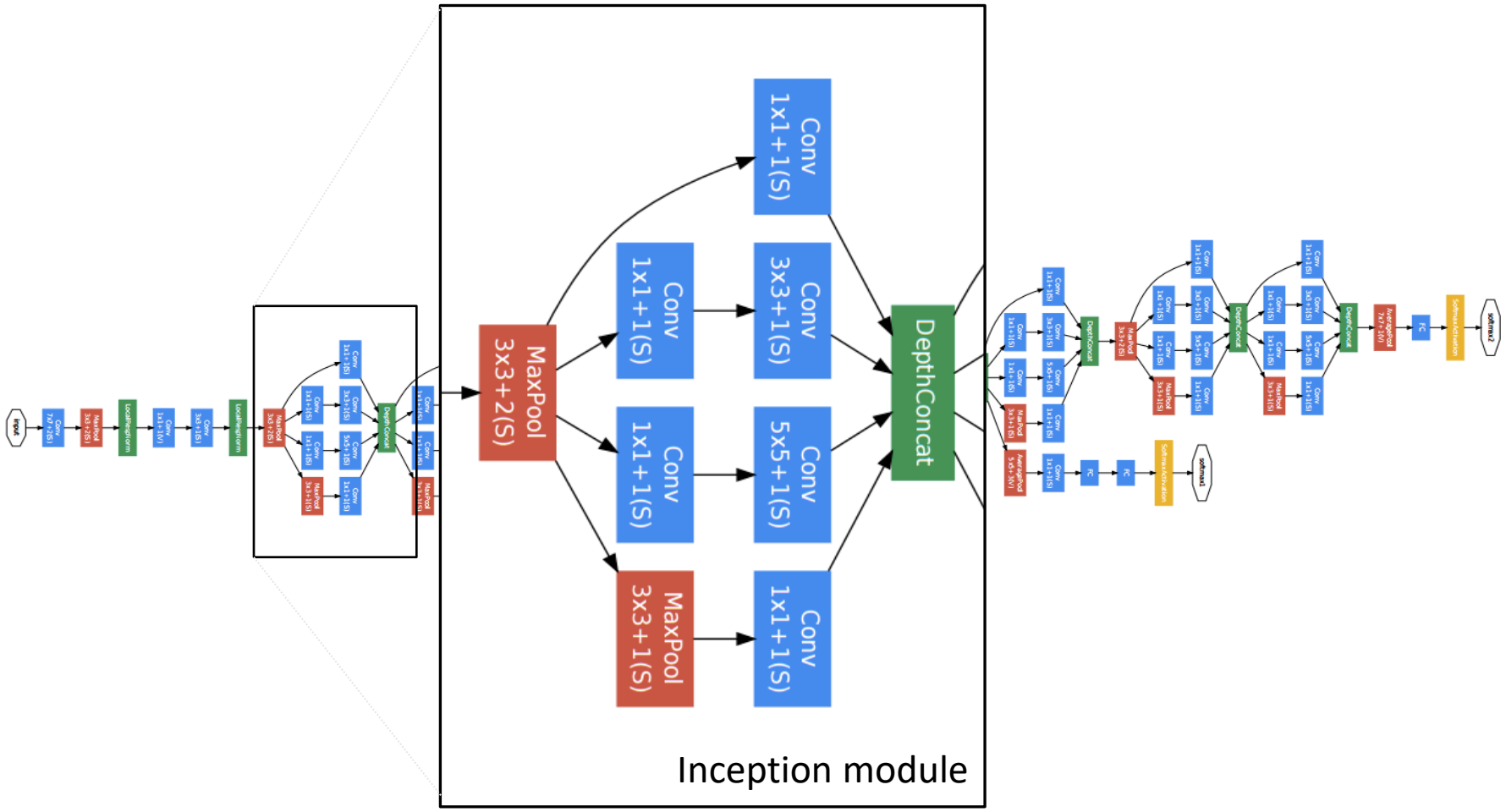
GoogLeNet and Inception (4/7)

- The Inception Module
  - Use 1x1 convolutions for dimensionality reduction before expensive convolutions



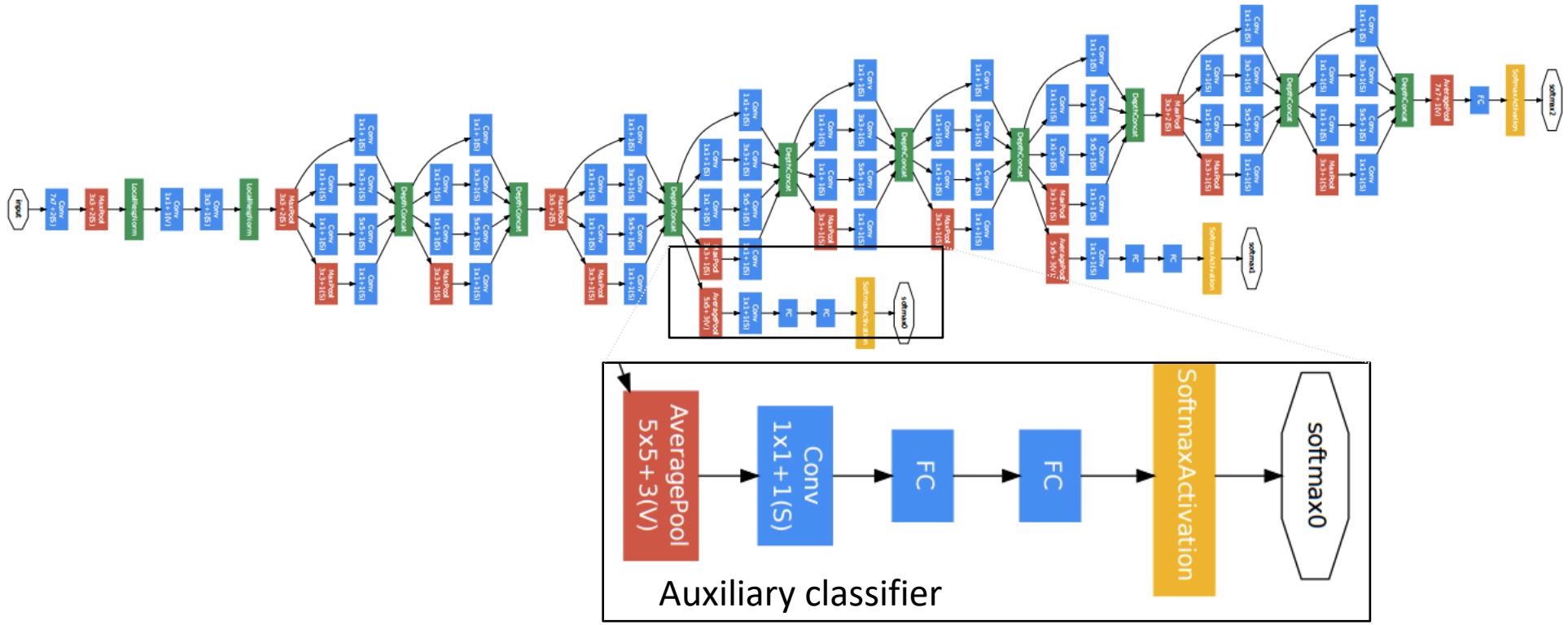
C. Szegedy et al., Going deeper with convolutions, CVPR 2015

GoogLeNet and Inception (5/7)



C. Szegedy et al., Going deeper with convolutions, CVPR 2015

GoogLeNet and Inception (6/7)



C. Szegedy et al., Going deeper with convolutions, CVPR 2015

GoogLeNet and Inception (7/7)

Summarizing The Insights

1. Exploit fully the fact that, in Images, correlation tend to be local
   - Concatenate 1X1, 3X3, 5x5 convolutions along with pooling
2. Decrease dimensions wherever computation requirements increase via a 1X1 Dimension Reduction Layer
3. Stack Inception Modules Upon Each Other
4. Counter-Balance Back-Propagation Downsides in Deep Network
   - Uses intermediate losses in the final loss
5. End with Global Average Pooling Layer Instead of Fully Connected Layer

References and Resources

- Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, Gradient-based learning applied to document recognition, Proc. IEEE 86(11): 2278–2324, 1998.
- A. Krizhevsky, I. Sutskever, and G. Hinton, ImageNet Classification with Deep Convolutional Neural Networks, NIPS 2012
- M. Zeiler and R. Fergus, Visualizing and Understanding Convolutional Networks, ECCV 2014
- K. Simonyan and A. Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition, ICLR 2015
- M. Lin, Q. Chen, and S. Yan, Network in network, ICLR 2014
- C. Szegedy et al., Going deeper with convolutions, CVPR 2015
- Stanford CS230: Deep Learning
- Stanford CS231n: Convolutional Neural Networks for Visual Recognition
- Illinois CS 498: Introduction to Deep Learning
- Berkeley Stat 157: Introduction to Deep Learning
- https://towardsdatascience.com/neural-network-architectures-156e5bad51ba