Advanced Python for Neuroscientists
Lecture 7: Convolutional Neural network

Summer 2022

Princeton Neuroscience Institute
Instructors: Yisi Zhang & Bichan Wu

July 19, 2022

Recap

Lecture 5

- Feedforward neural network
- Gradient descent

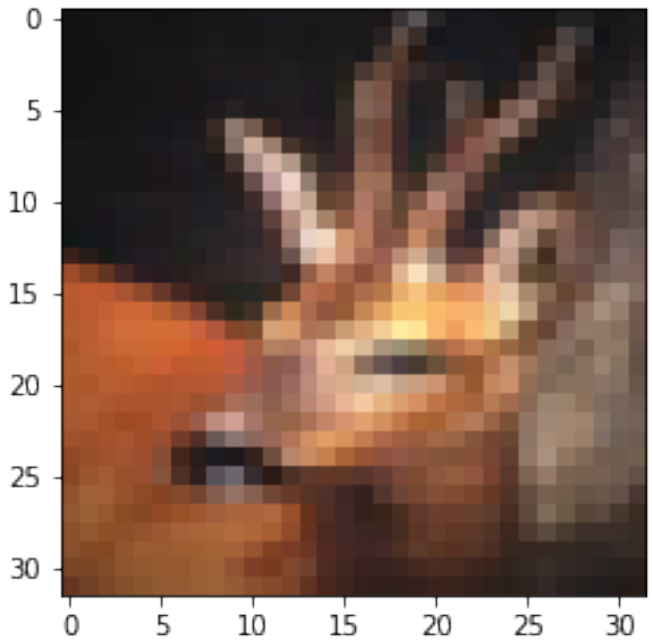
Lecture 6

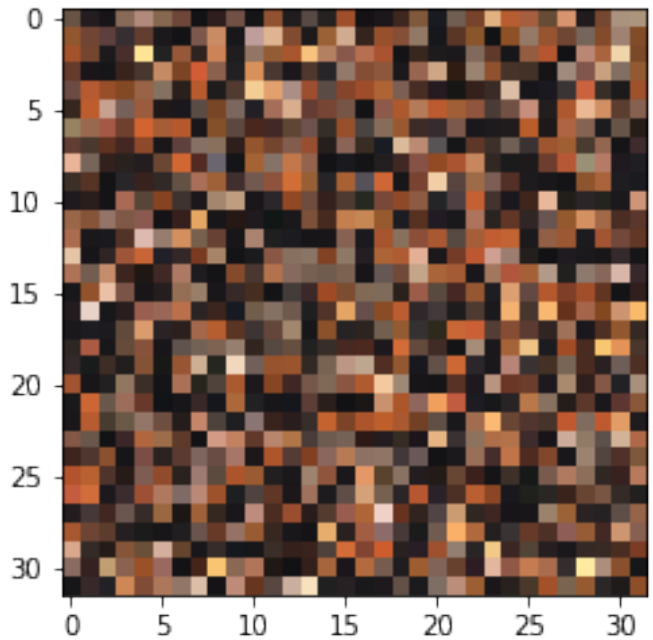
- Backpropagation
- Stochastic Gradient Descent
- Application

Outline

- Motivation & concept
- Overview
- Architectures

7.1 Motivation & concept



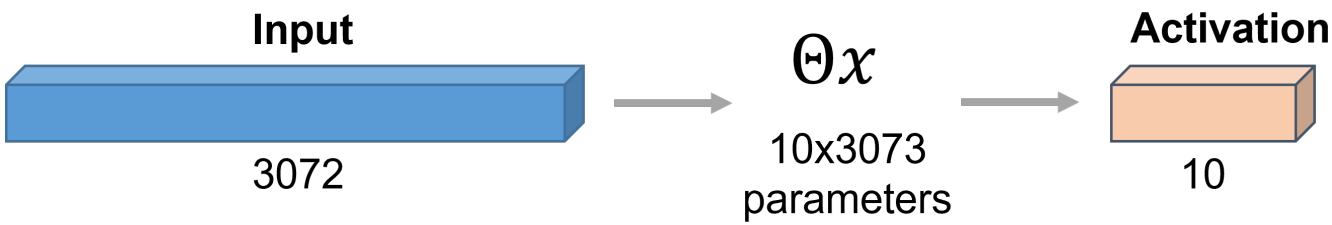


We may want to preserve the spatial structure.

7.1 Motivation & concept

Fully connected layer

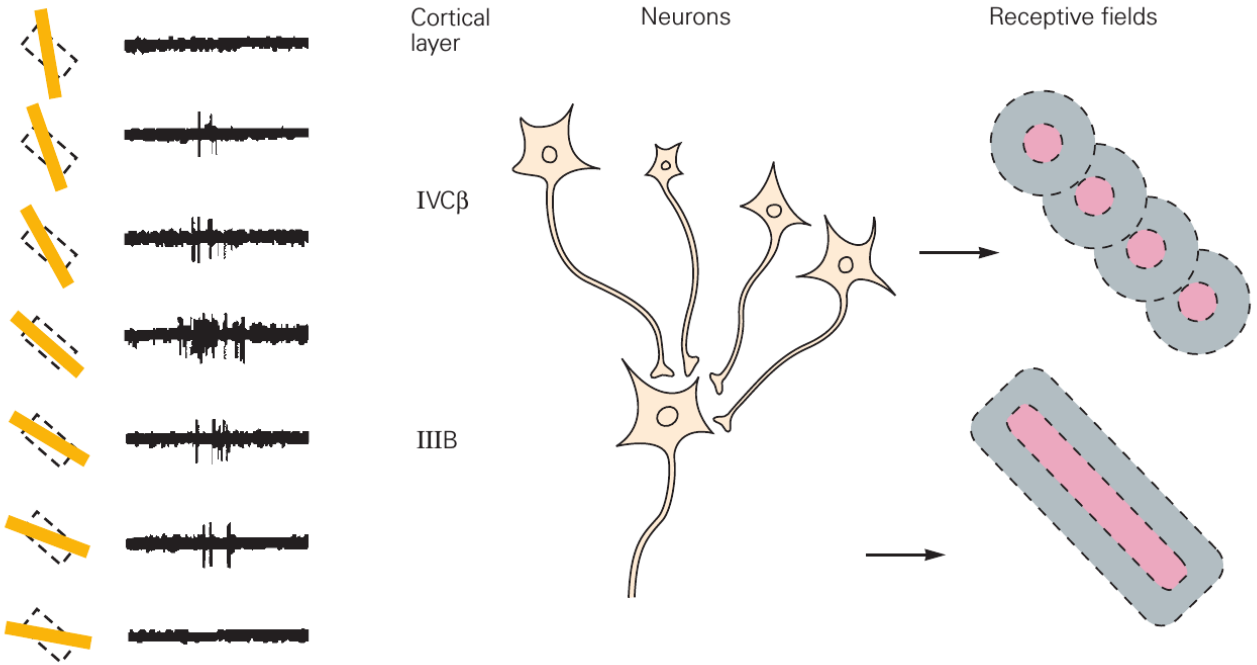
A CIFAR10 image size 32x32x3 $\rightarrow$ 3072 x 1



This does not maintain the spatial information.

7.1 Motivation & concept

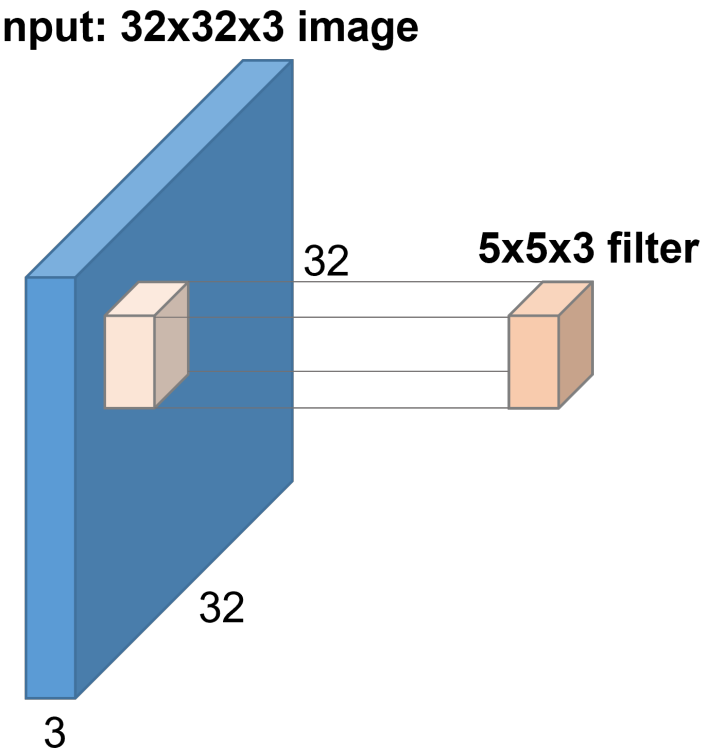
Receptive field



Cite Kandel et al., *Principles of neural science*. Ed6

7.1 Motivation & concept

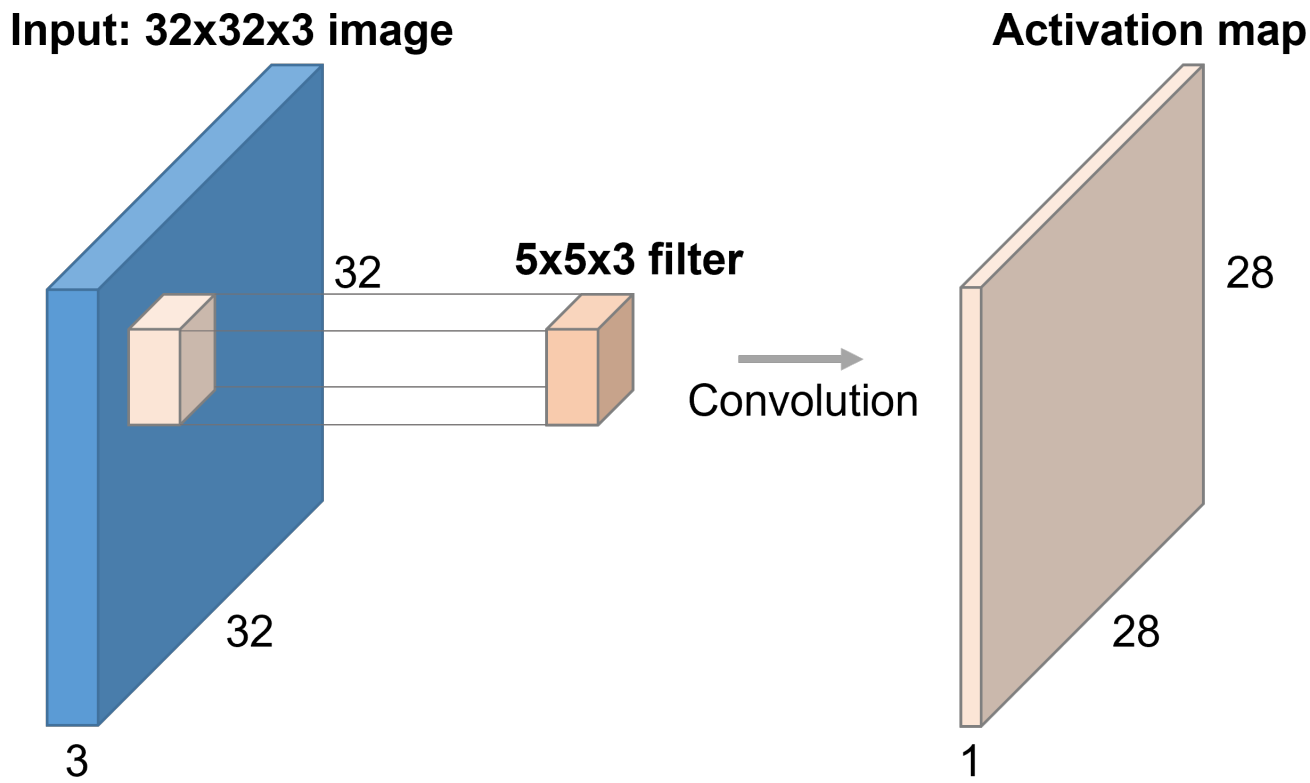
Convolutional neural network (ConvNet)



Slide (convolve) the filter over all spatial locations. At each location perform a dot product.

7.1 Motivation & concept

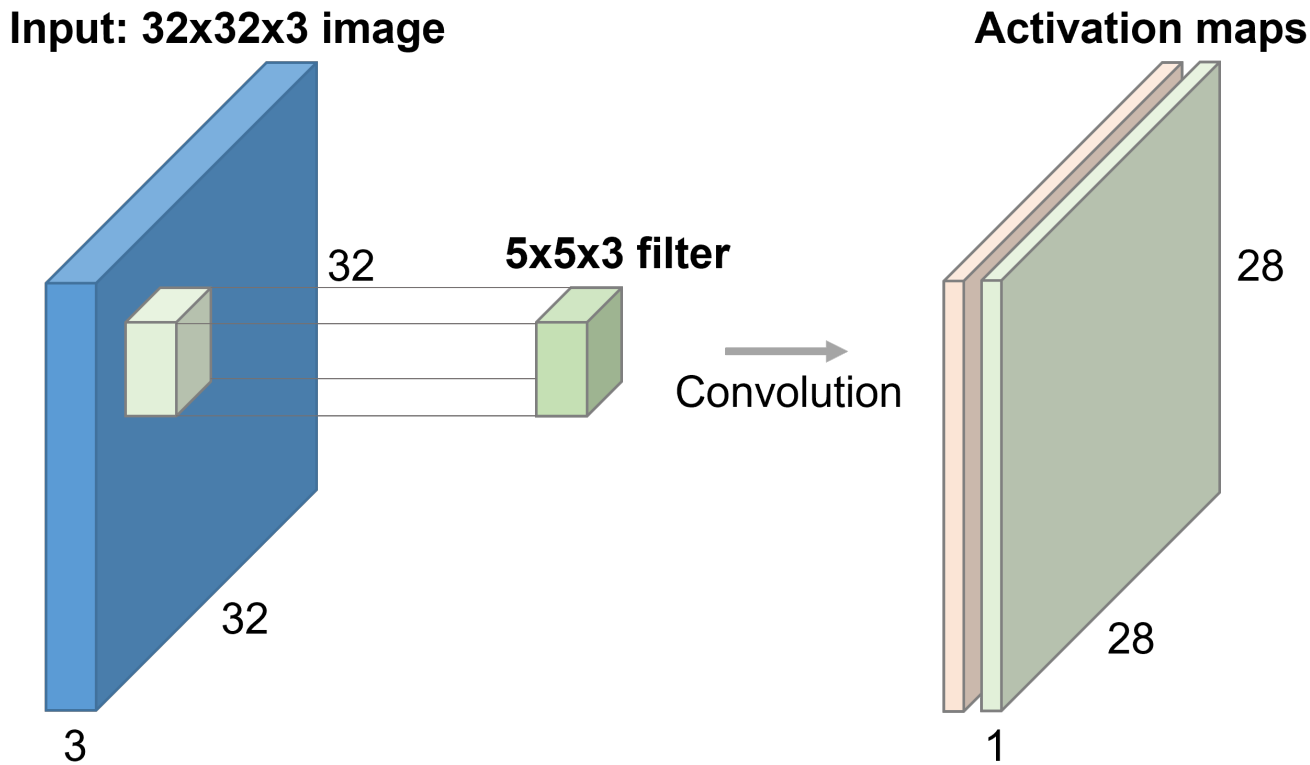
Convolutional neural network (ConvNet)



Each filter produces a 2D activation map.

7.1 Motivation & concept

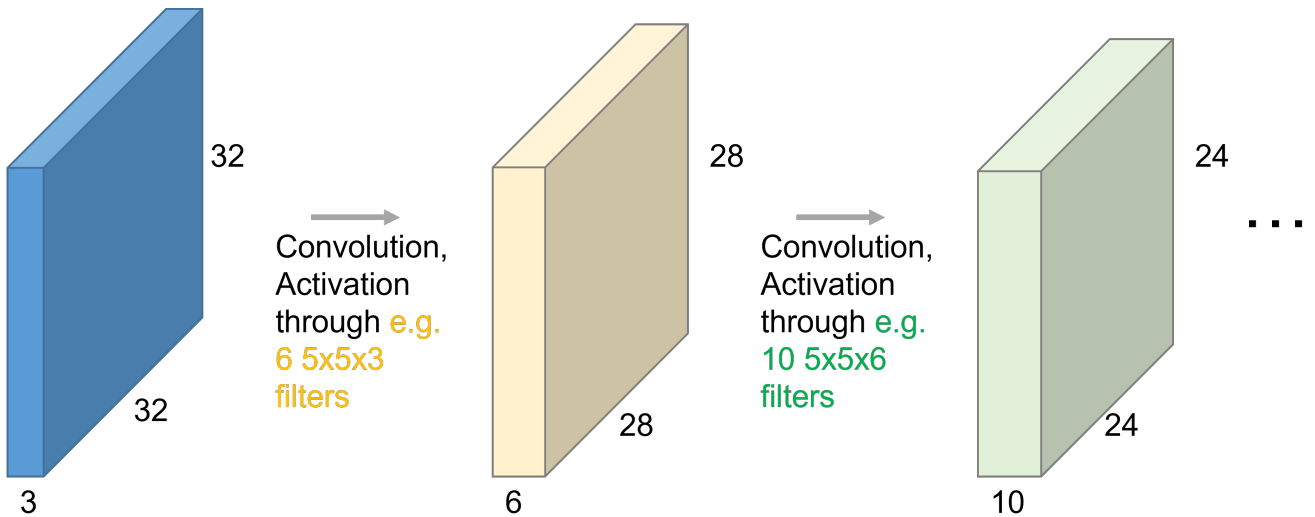
Convolutional neural network (ConvNet)



Each filter produces a 2D activation map.

7.1 Motivation & concept

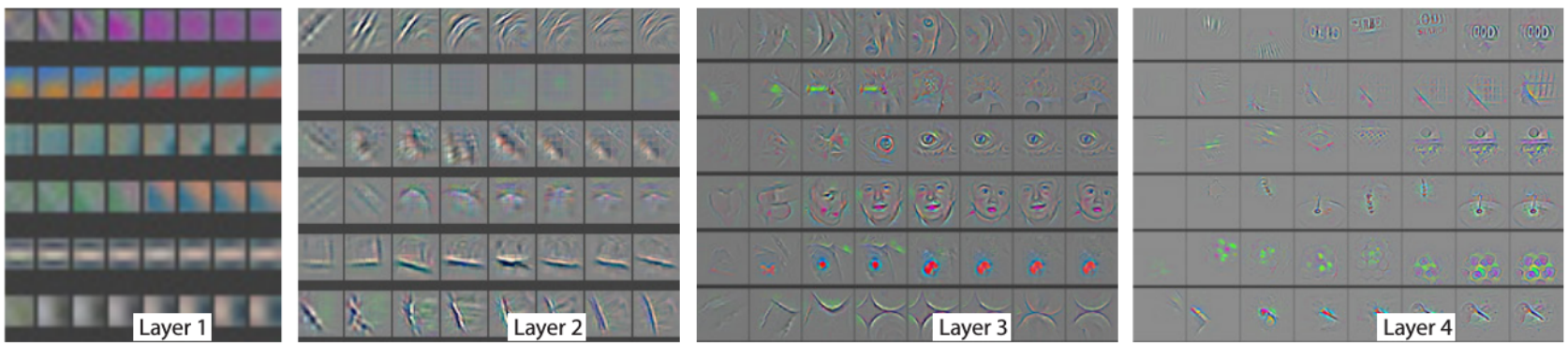
Convolutional neural network (ConvNet)



The same logic applied to each layer.

7.1 Motivation & concept

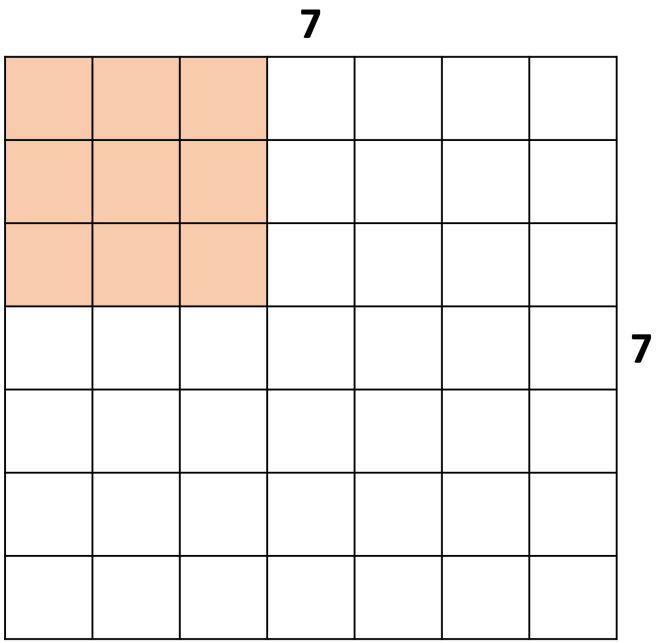
Low-level → Mid-level → High-level features → linearly separable classifier



[Zeiler & Fergus 2013]

7.2 Overview

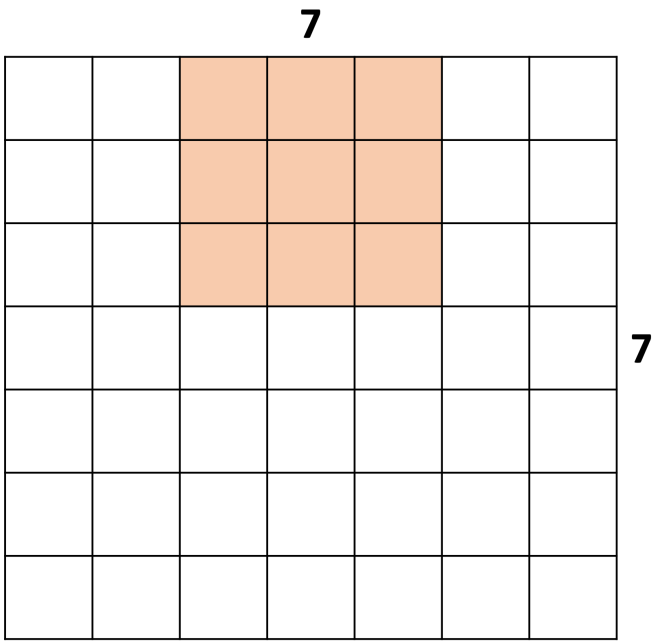
How does the convolution work?



Apply a 3x3 filter to a 7x7 grid with stride 2

7.2 Overview

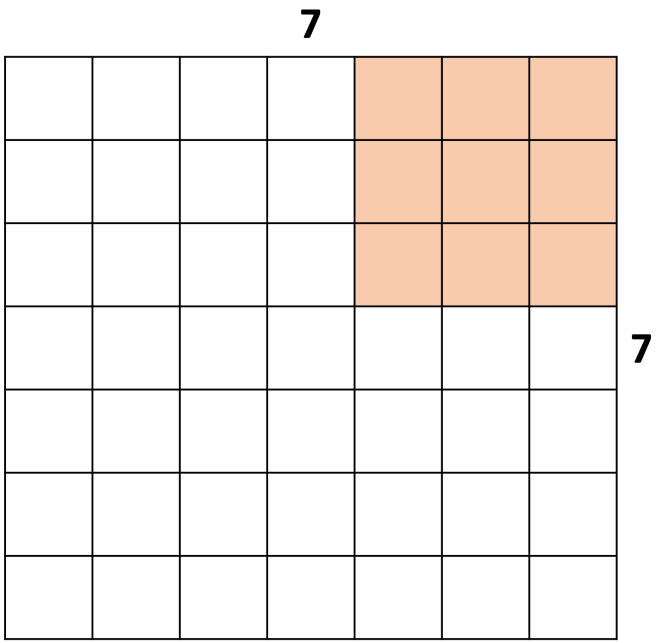
How does the convolution work?



Apply a 3x3 filter to a 7x7 grid with stride 2

7.2 Overview

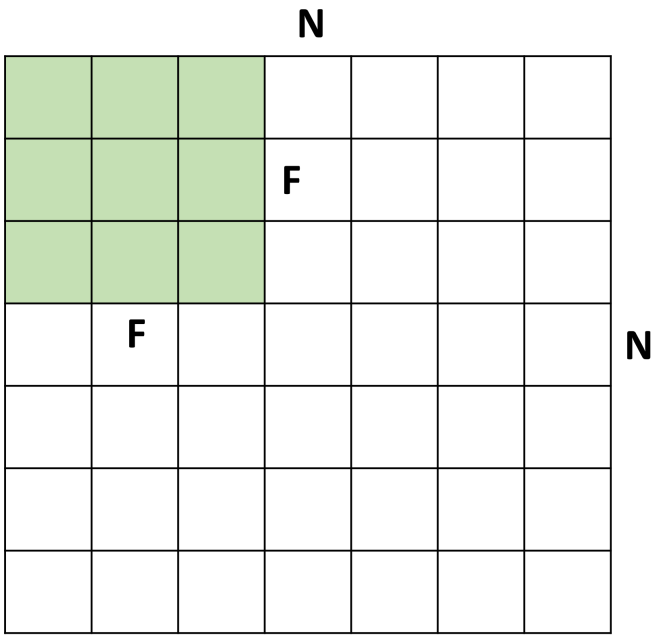
How does the convolution work?



Apply a 3x3 filter to a 7x7 grid with stride 2

7.2 Overview

How does the convolution work?

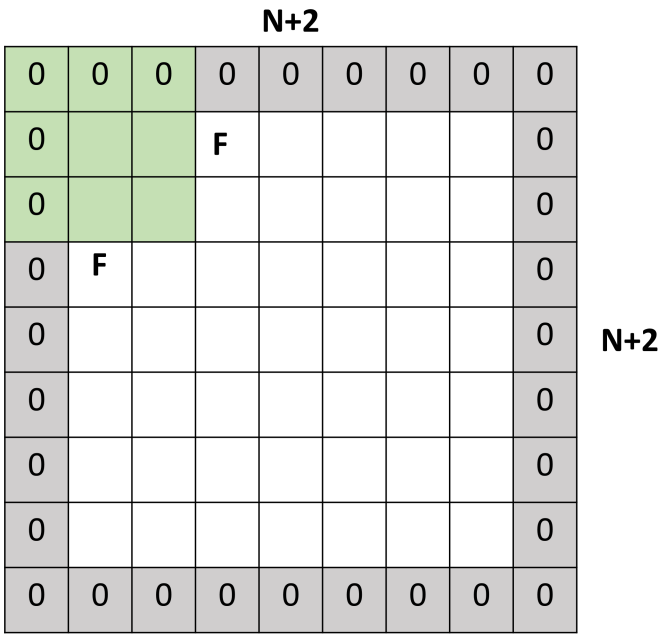


Output size
= (N-F)/stride + 1

e.g. (7-3)/2 + 1 = 3

7.2 Overview

Common to zero-pad the border



If N = 7, F = 3, stride = 3
what is the output size?

7.2 Overview

$$I[x, y] * F[x, y] = \sum_m \sum_n I[m, n] F[x - m, y - n]$$

Input

0	0	0	0	0	0	0	0	0
0	1	2	3	4	12	9	8	0
0	5	2	3	4	12	9	8	0
0	5	2	1	4	10	9	8	0
0	7	2	1	4	12	7	8	0
0	7	2	1	4	14	9	8	0
0	5	2	3	4	12	7	8	0
0	5	2	1	1	12	9	8	0
0	0	0	0	0	0	0	0	0

*

Filter 3x3

1	2	1
2	4	2
1	2	1

Stride=3

=

Output

20	69	75
60	84	96
36	53	73

7.2 Overview

How many parameters in this layer (or what are being learned)?

Image: 32x32x3, Filter: 10 of 5x5x3, stride 1, pad 2

7.2 Overview

How many parameters in this layer (or what are being learned)?

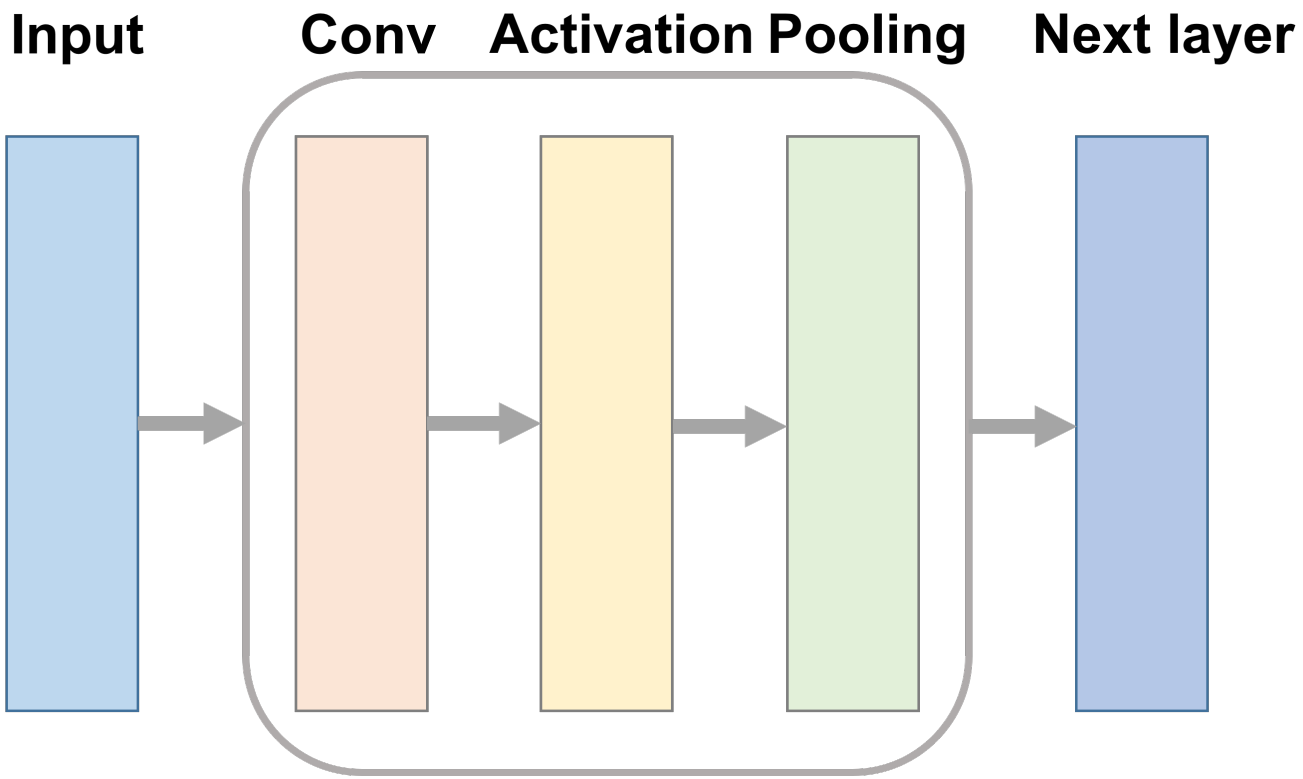
Image: 32x32x3, Filter: 10 of 5x5x3, stride 1, pad 2 each filter has

5x5x3 + 1 = 76 parameters (+1 for bias)
total # parameters = 76 x 10 = **760**

Consider convolution as Θx

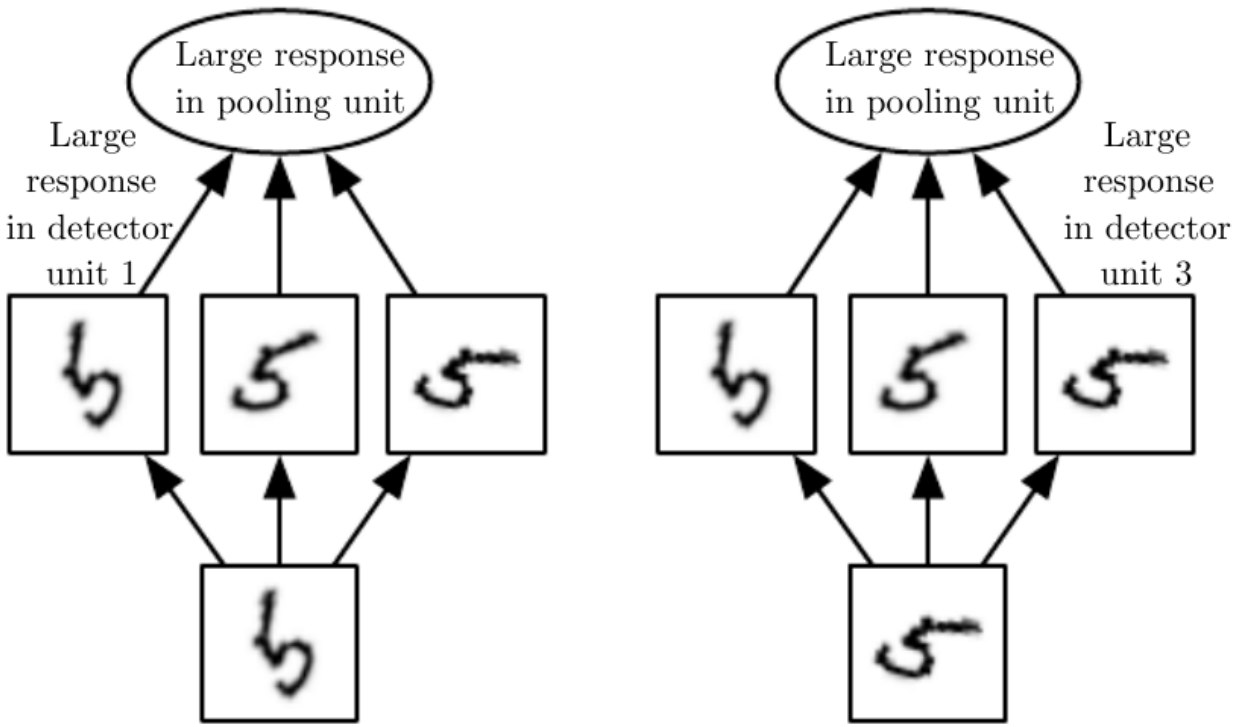
7.2 Overview

A complex layer consists of Convolution, Activation (e.g. ReLU), Pooling (e.g. Max pooling)



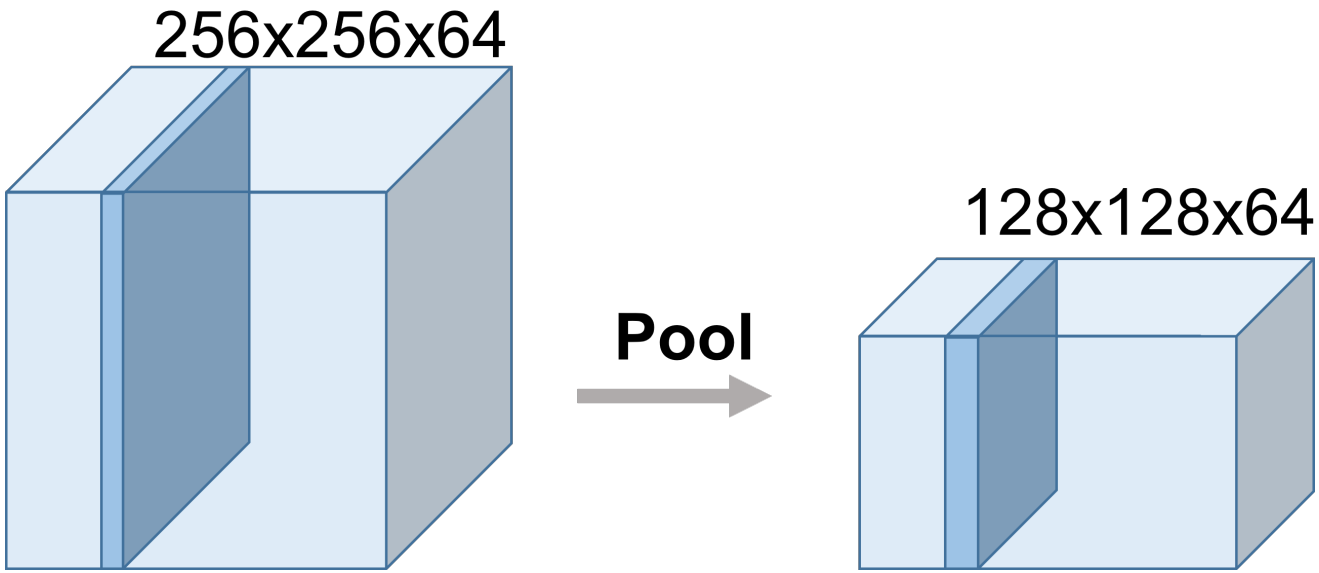
7.2 Overview

Intuition of pooling: Invariance to local translation



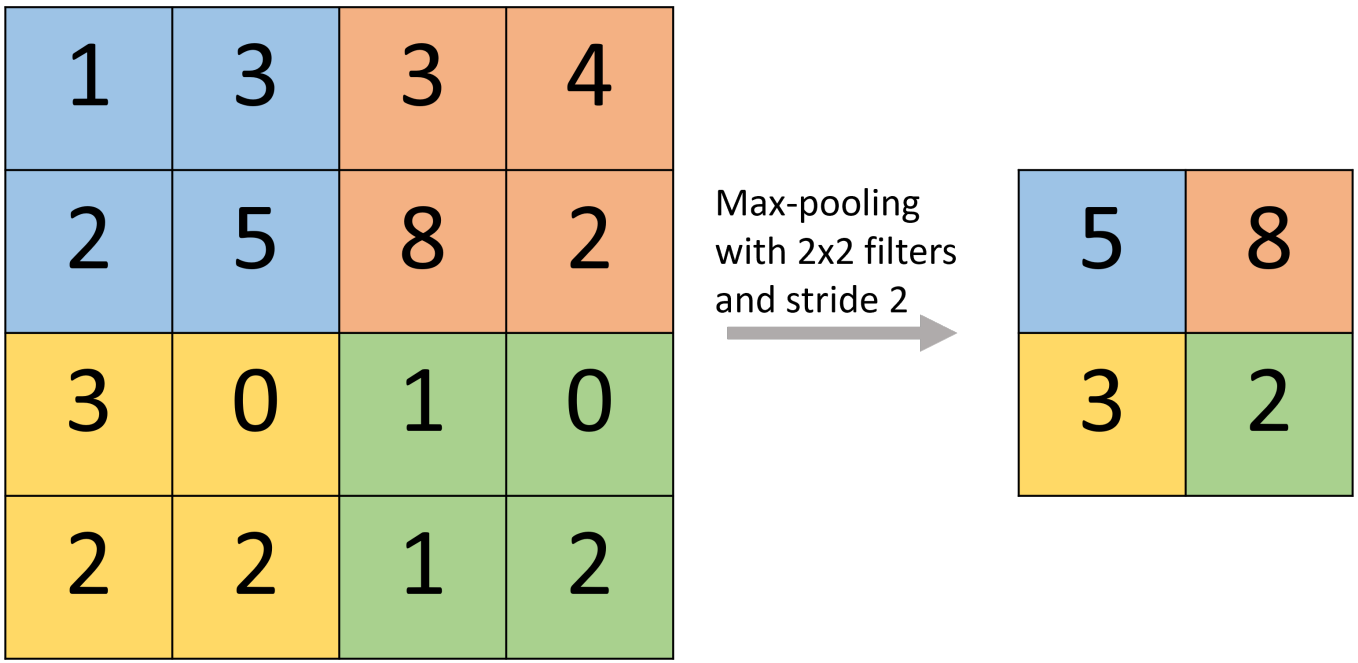
7.2 Overview

Pooling over individual activation map



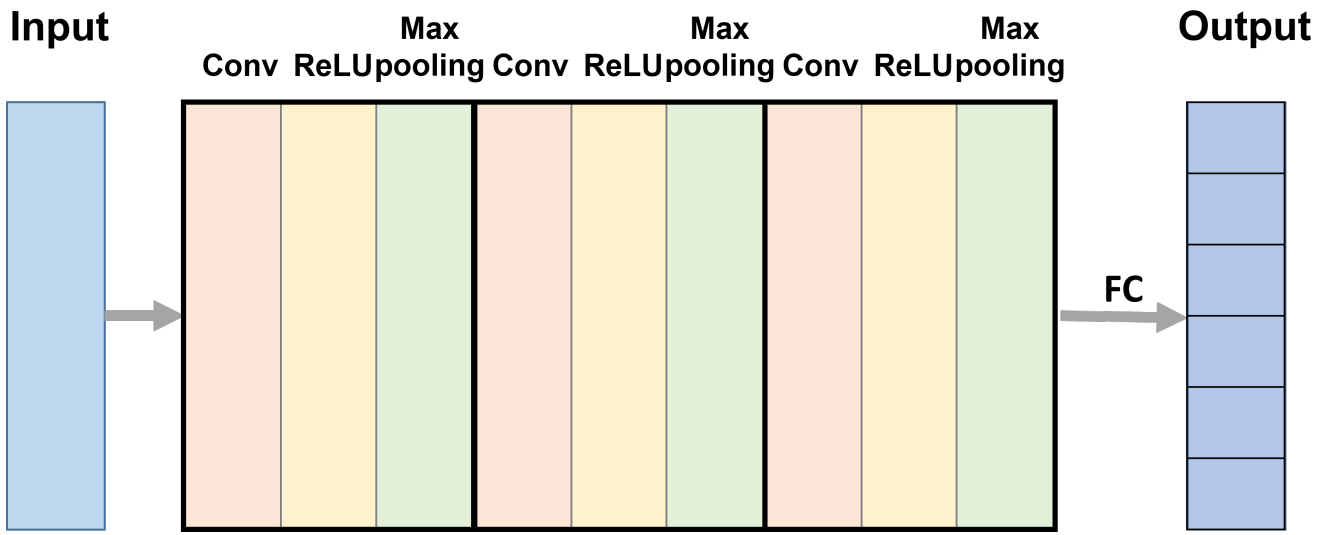
7.2 Overview

Max pooling



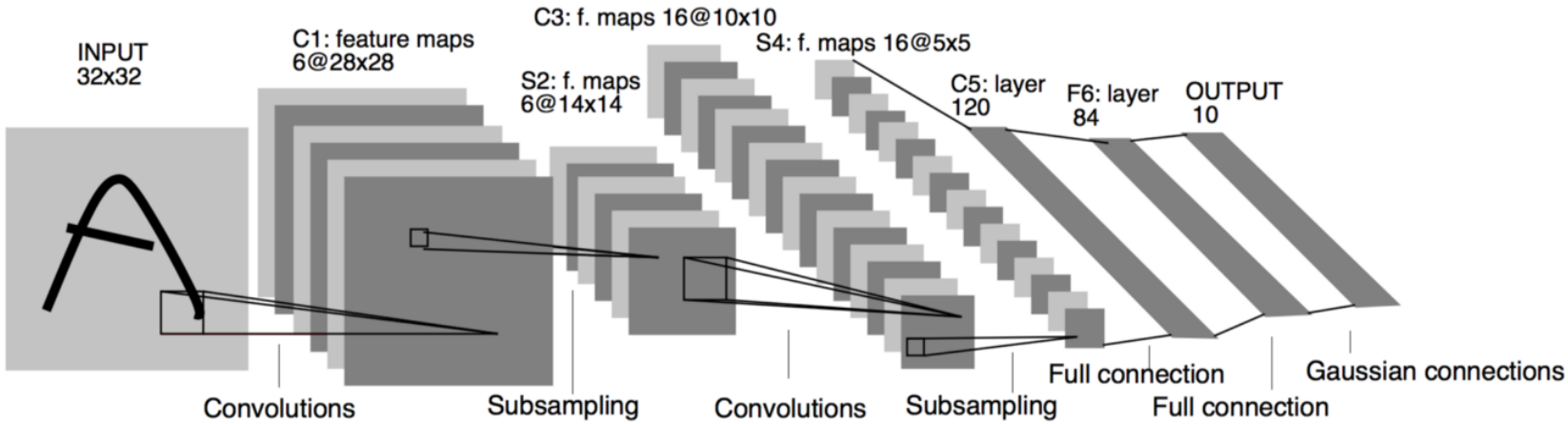
7.2 Overview

A complete ConvNet for classification



7.3 Architectures

LeNet-5

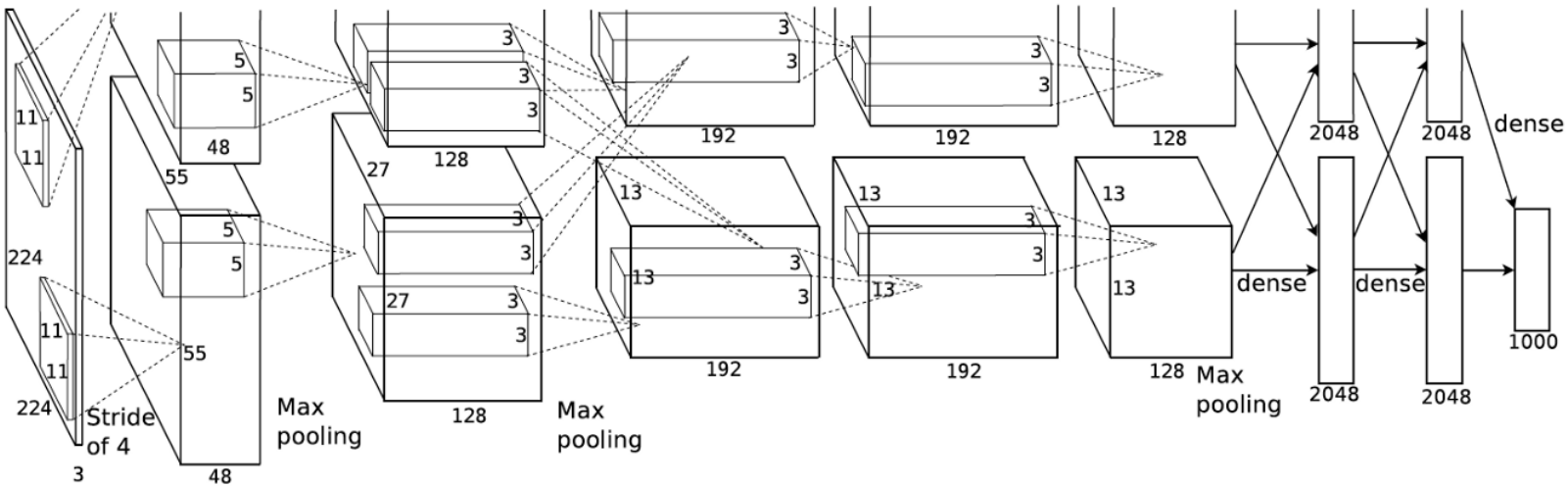


[LeCun et al., 1998]

Conv filters 5x5, stride 1, 2x2 pooling at stride 2

7.3 Architectures

AlexNet

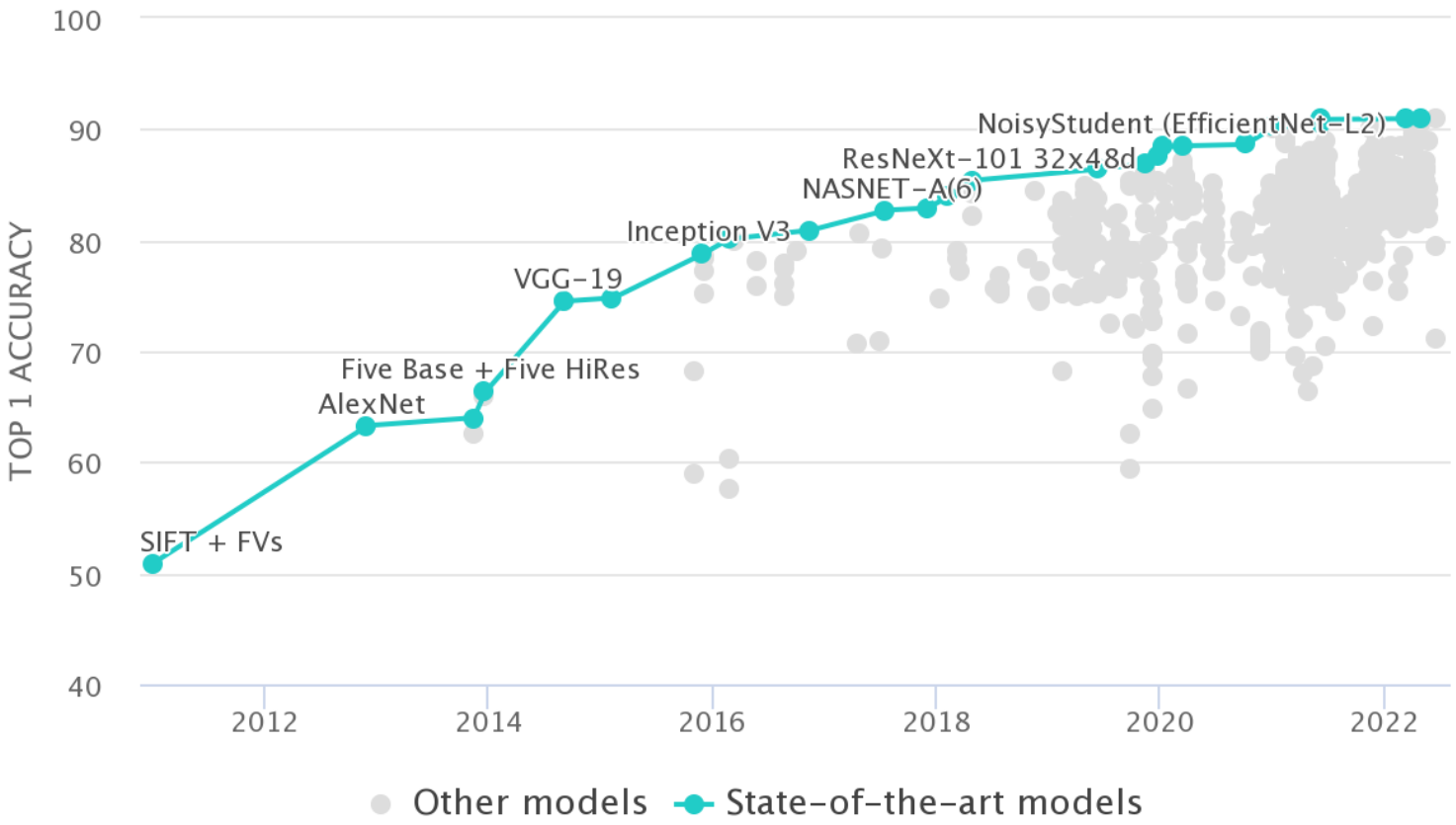


[Krizhevsky et al., 2012]

CONV1→MAX POOL1→ CONV2→MAX POOL2→
CONV3→CONV4→CONV5→ MAX POOL3→FC6→FC7→FC8

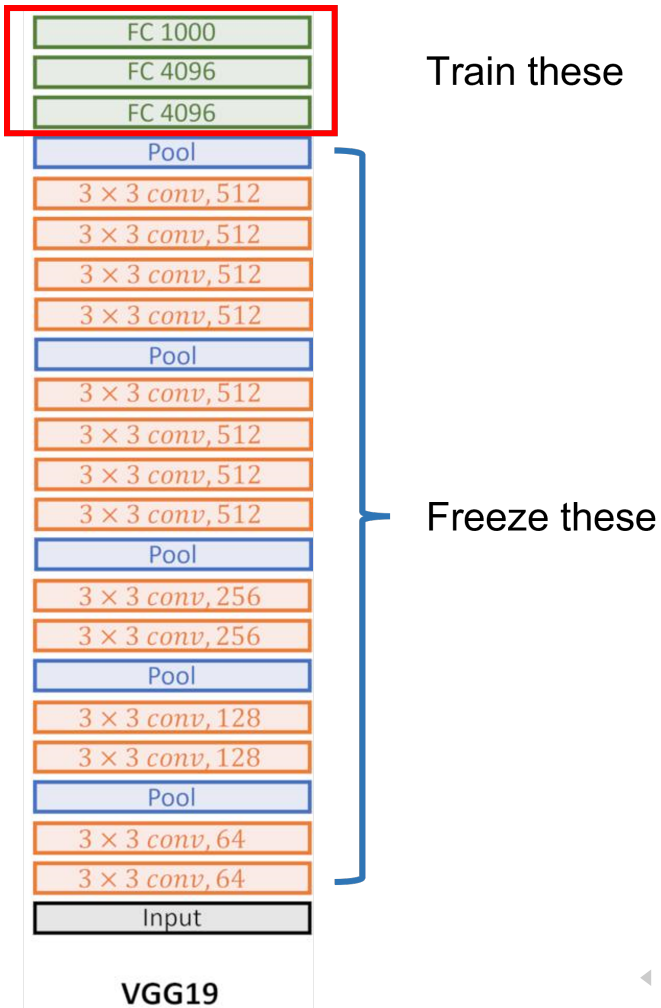
7.3 Architectures

ImageNet Large Scale Visual Recognition Challenge (ILSVRC) winners



7.3 Architectures

Transfer Learning



Homework

- Make sure you understand all the exercises above
- Run through the codes here that should replicate all the figures
  https://github.com/yisiszhang/AdvancedPython/blob/main/colab/Lecture7.ipynb
- Make sure to understand the inputs to Conv and FC layers
- Try to improve the performance of the ConvNet model