Computer Vision: Fall 2022 — Lecture 10

Dr. Karthik Mohan

Univ. of Washington, Seattle

November 2, 2022

Check-In

1. Identify your team mate through the spreadsheet

Check-In

1. Identify your team mate through the spreadsheet
2. First Check Point/Deadline for Mini-Project due November 6

References

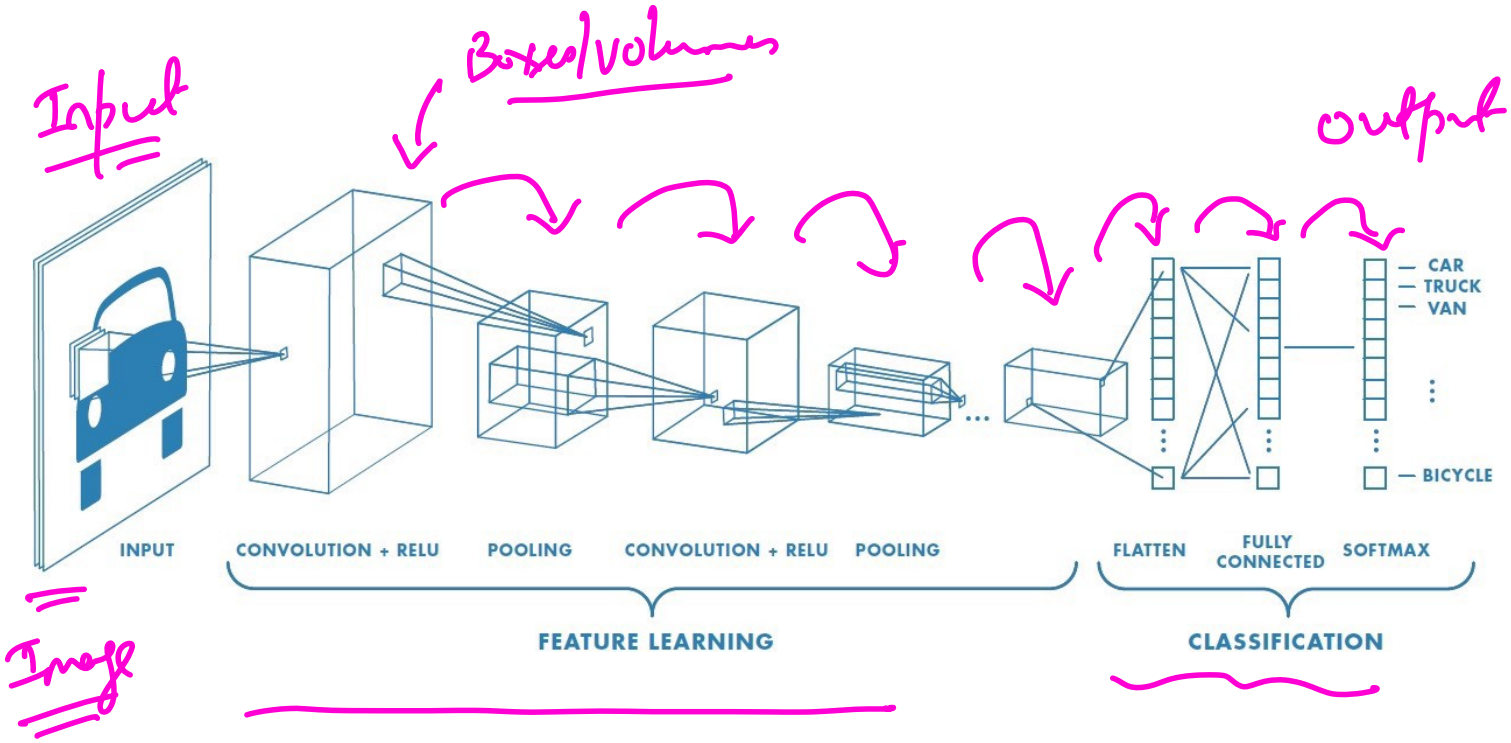
1. Good Book for Machine Learning Concepts
2. Deep Learning Reference
3. Convolutional Neural Networks for Visual Recognition
4. Convolutional Neural Net Tutorial

Today

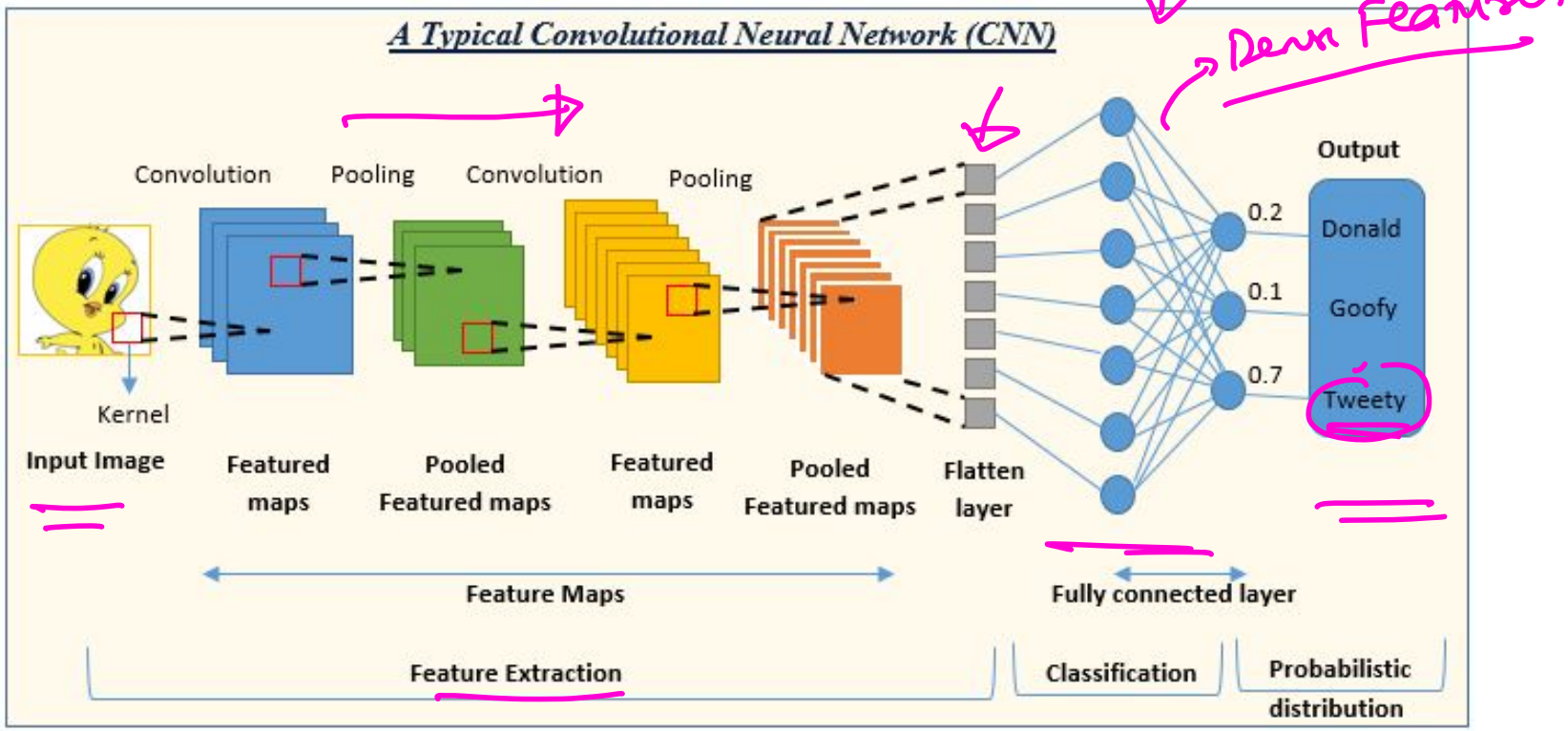
1. Convolutional Neural Networks - Introduction

(CNN)

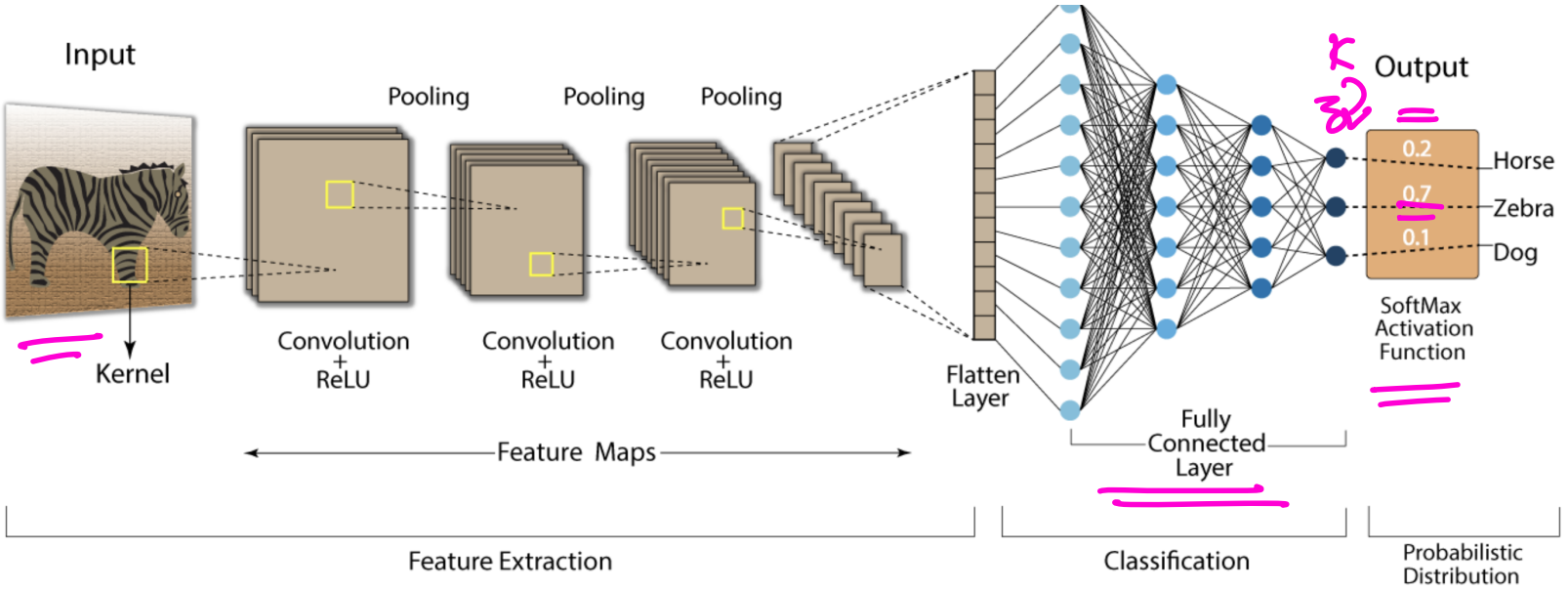
Convolutional Neural Networks - Introduction



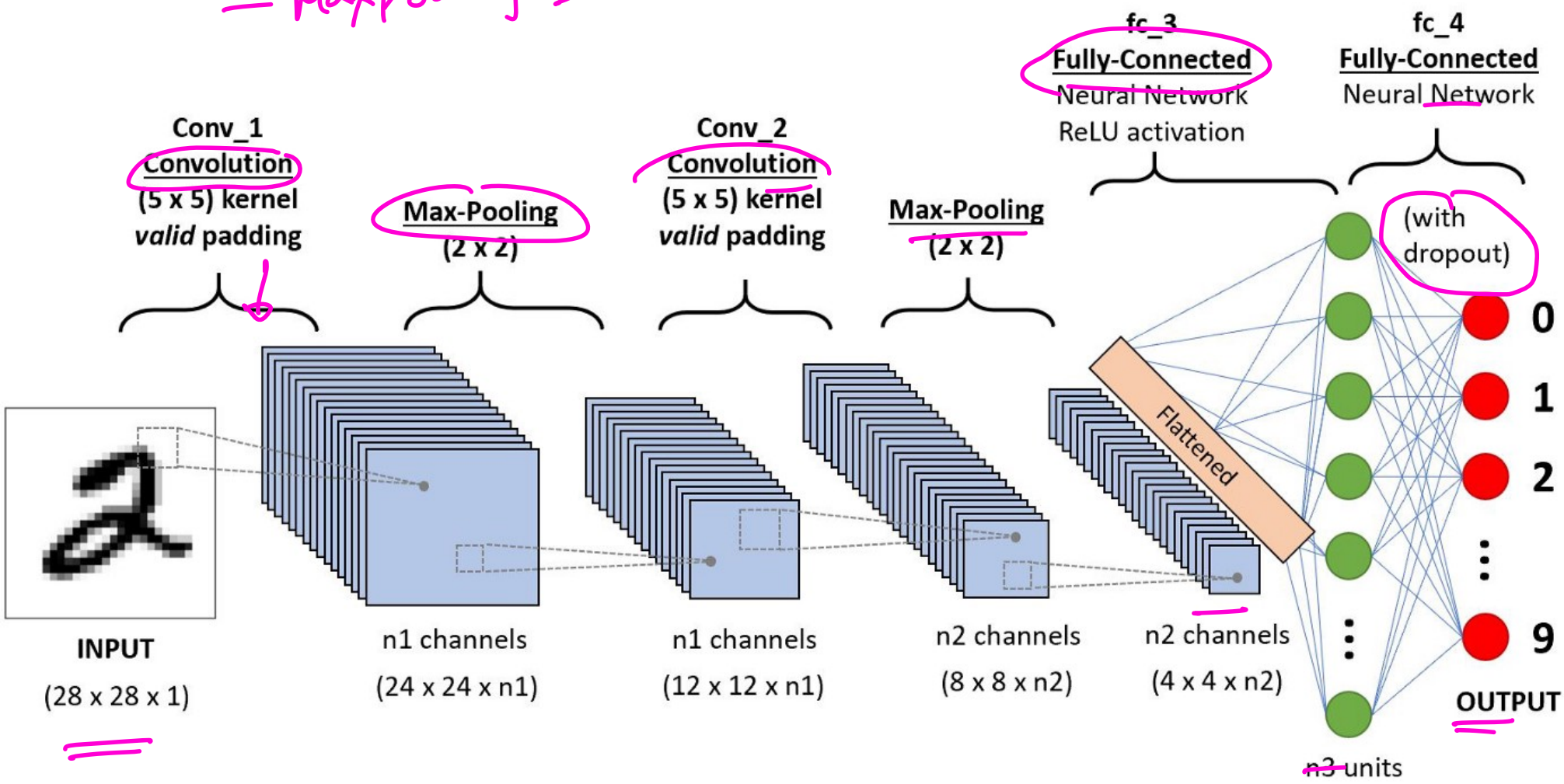
Convolutional Neural Networks - Functionality Breakdown



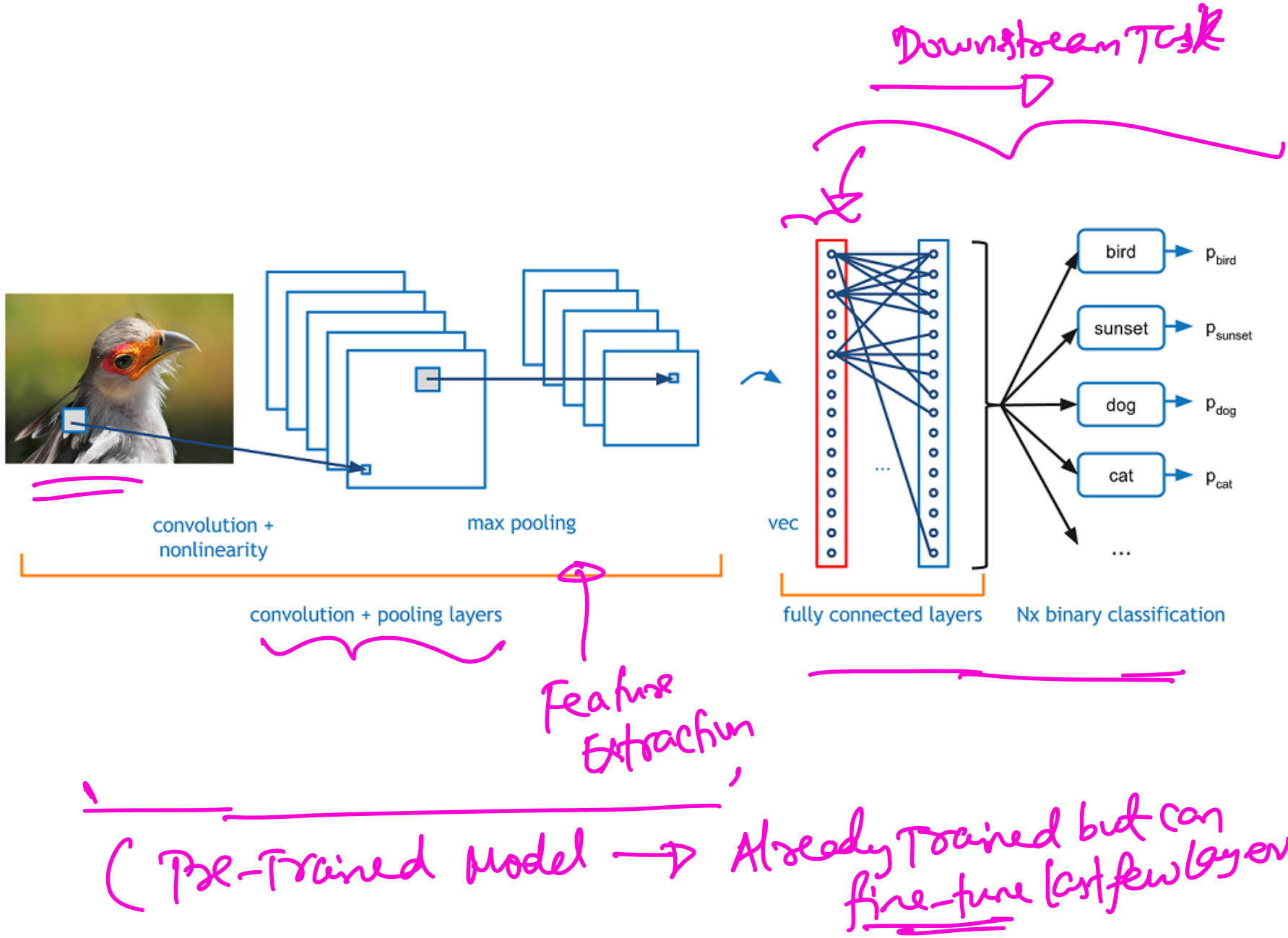
Convolutional Neural Networks - Functionality Breakdown



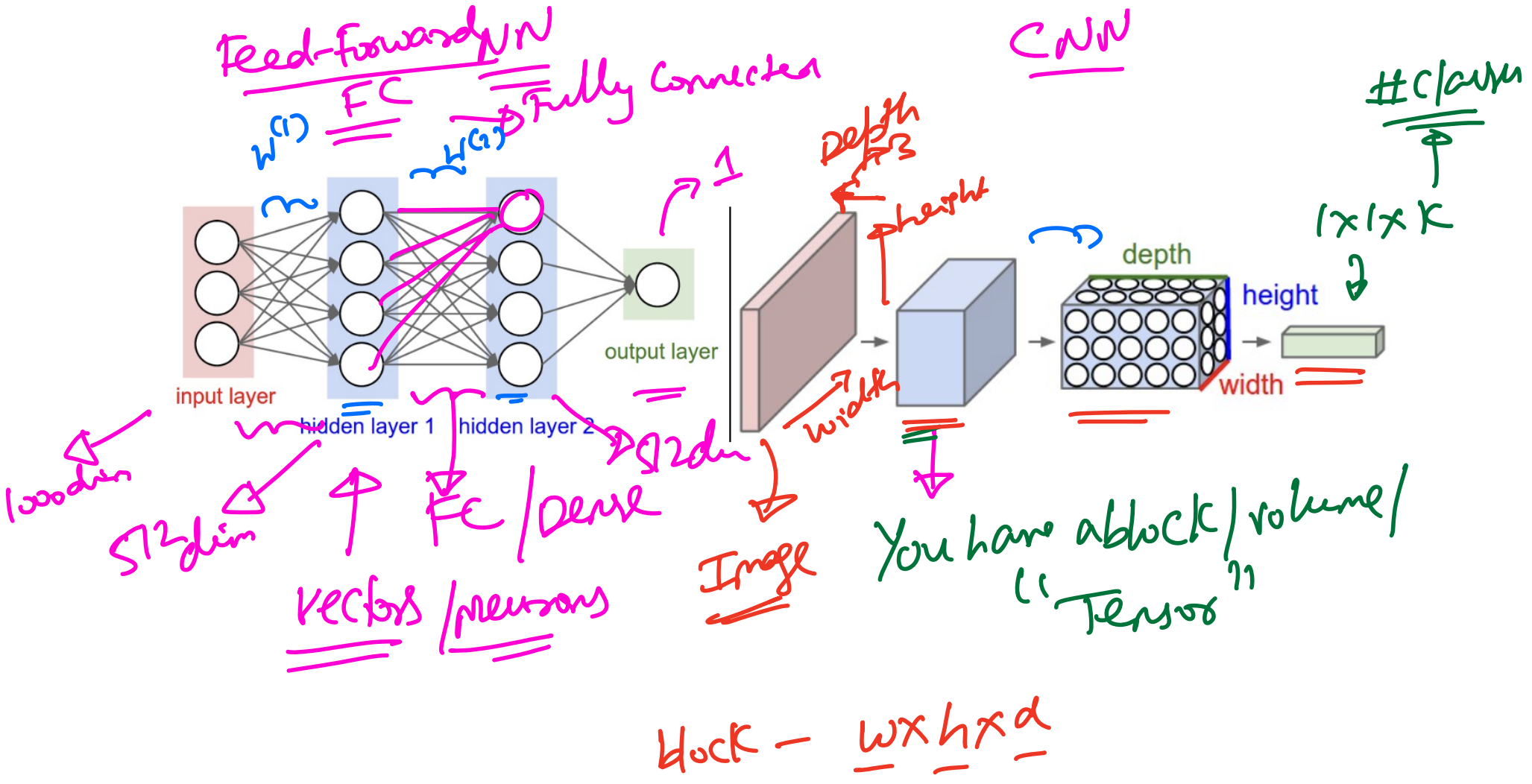
Convolutional Neural Networks - Layers Breakdown



Convolutional Neural Networks - Layers Breakdown



NN vs CNN


Feed-forward NN
FC
Fully Connected
CNN
#C/conv
1x1xK
depth
height
width
input layer
hidden layer 1
hidden layer 2
output layer
1000dim
512dim
512dim
FC/Dense
vectors/neurons
Image
You have a block/volume/
"Tensor"
block - wxhxd

NN vs CNN

1. CNN is a special type of NN

NN vs CNN

1. CNN is a special type of NN
2. Specialized to Images

NN vs CNN

1. CNN is a special type of NN
2. Specialized to Images
3. FC layers yield too many parameters/weights for NN arch

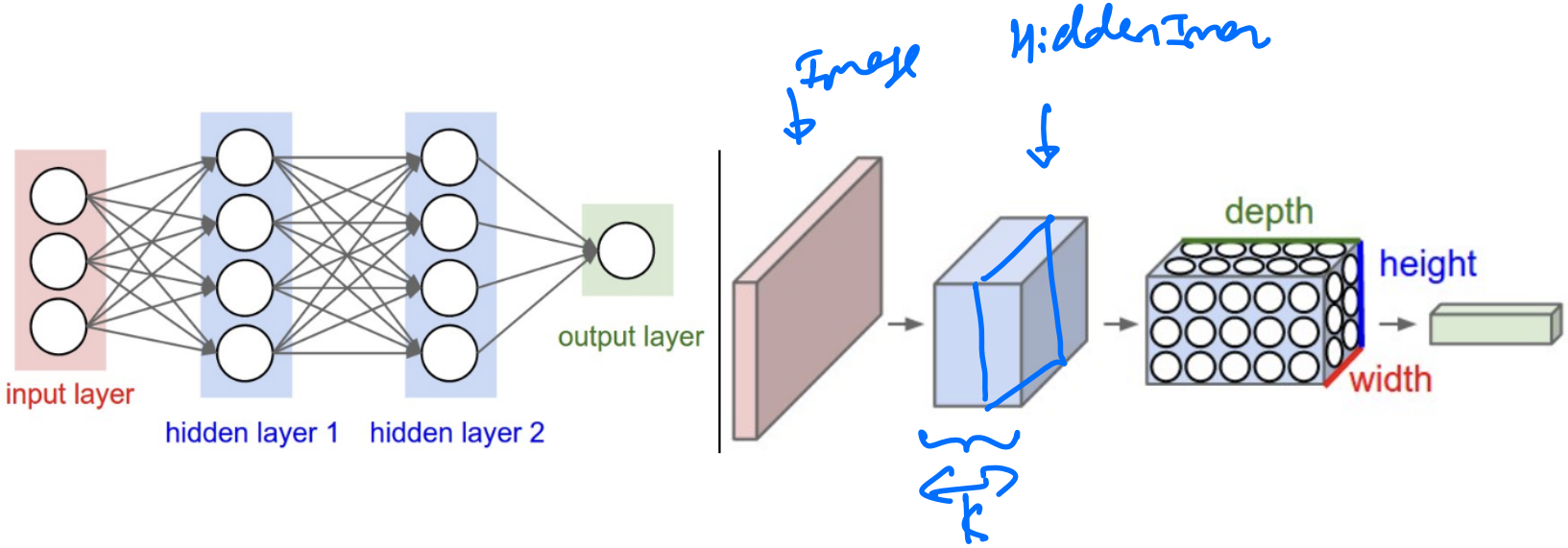
NN vs CNN

1. CNN is a special type of NN
2. Specialized to Images
3. FC layers yield too many parameters/weights for NN arch
4. More intuitive feature engineering (in terms of convolutions) done by CNN as compared to a regular NN

NN vs CNN

1. CNN is a special type of NN
2. Specialized to Images
3. FC layers yield too many parameters/weights for NN arch
4. More intuitive feature engineering (in terms of convolutions) done by CNN as compared to a regular NN
5. Works on a block with height, width and depth as compared to a NN, where the layers are encoded as vectors.

NN vs CNN


input layer
hidden layer 1
hidden layer 2
output layer
depth
height
width

Types of Layers/Transforms in CNN

Layers at the (End of CNN)

FC Layer

This is the same as in a feed-forward NN arch. Every neuron in the next layer is connected to every neuron in previous layer - Hence FC or *fully connected*

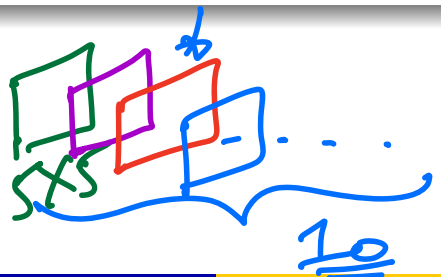
Types of Layers/Transforms in CNN

FC Layer

This is the same as in a feed-forward NN arch. Every neuron in the next layer is connected to every neuron in previous layer - Hence FC or *fully connected*

Conv Layer

This is the most important and frequently used layer in a CNN arch - Here one or more Convolution Kernels (learned as parameters in training) are each convolved with the input to produce an output block with the same depth as the number of convolution kernels.



Types of Layers/Transforms in CNN

Pooling Layer

Usually used to reduce the total number of parameters in the CNN network - Pooling can reduce the number of neurons from one layer to next with simple operations.

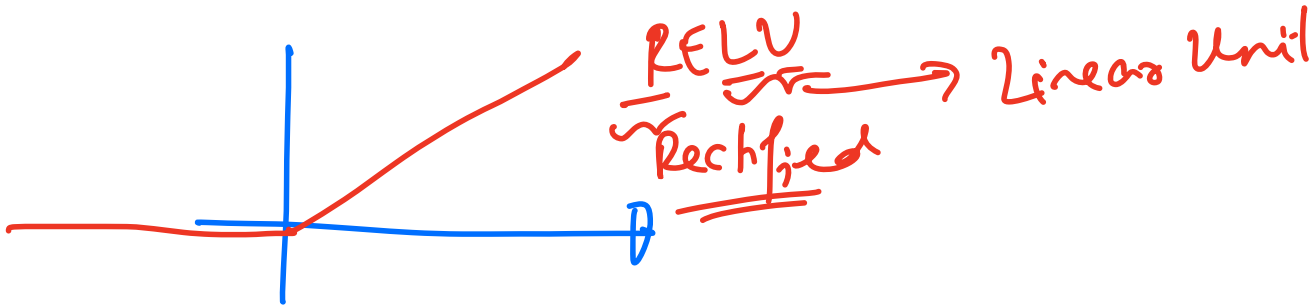
Types of Layers/Transforms in CNN

Pooling Layer

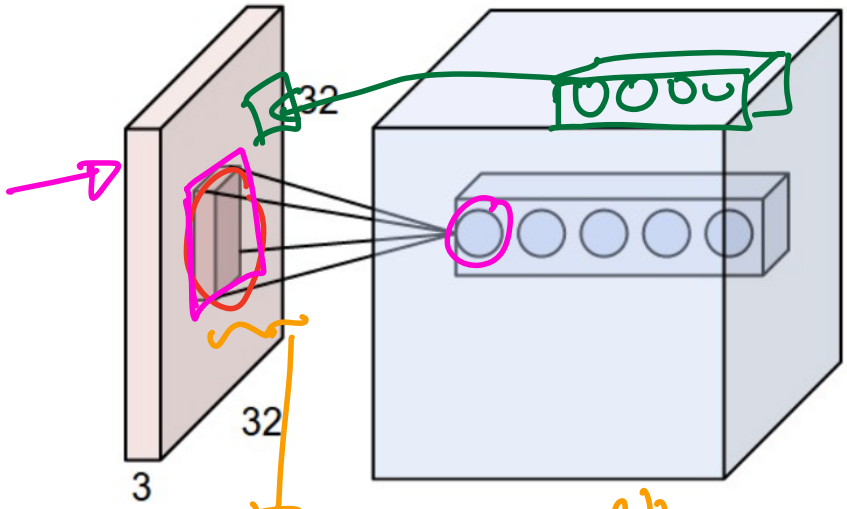
Usually used to reduce the total number of parameters in the CNN network - Pooling can reduce the number of neurons from one layer to next with simple operations.

RELU Activation

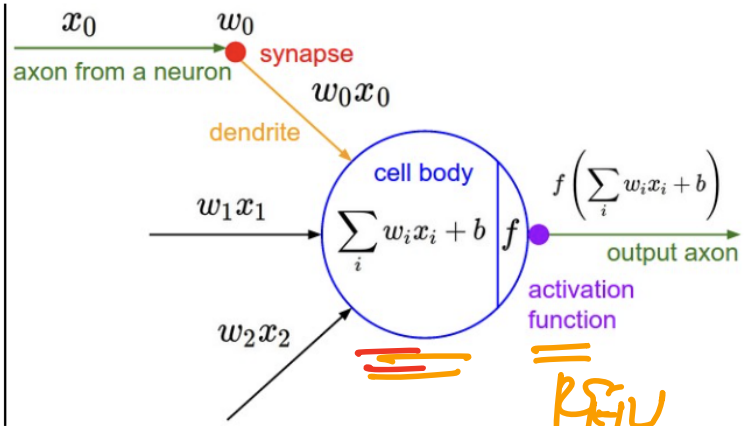
Just like in NN arch - RELU is used in CNN as well as a non-linear transformation of neurons.

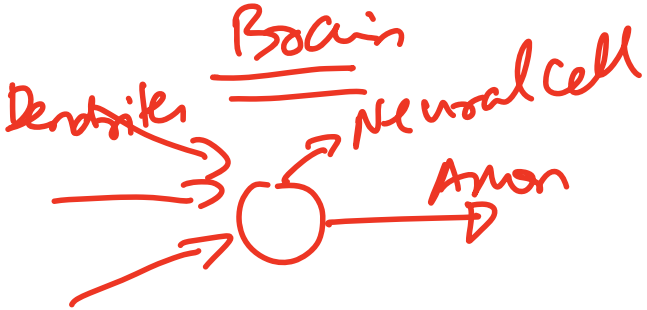


Conv Layer in CNN



"Locally connected"

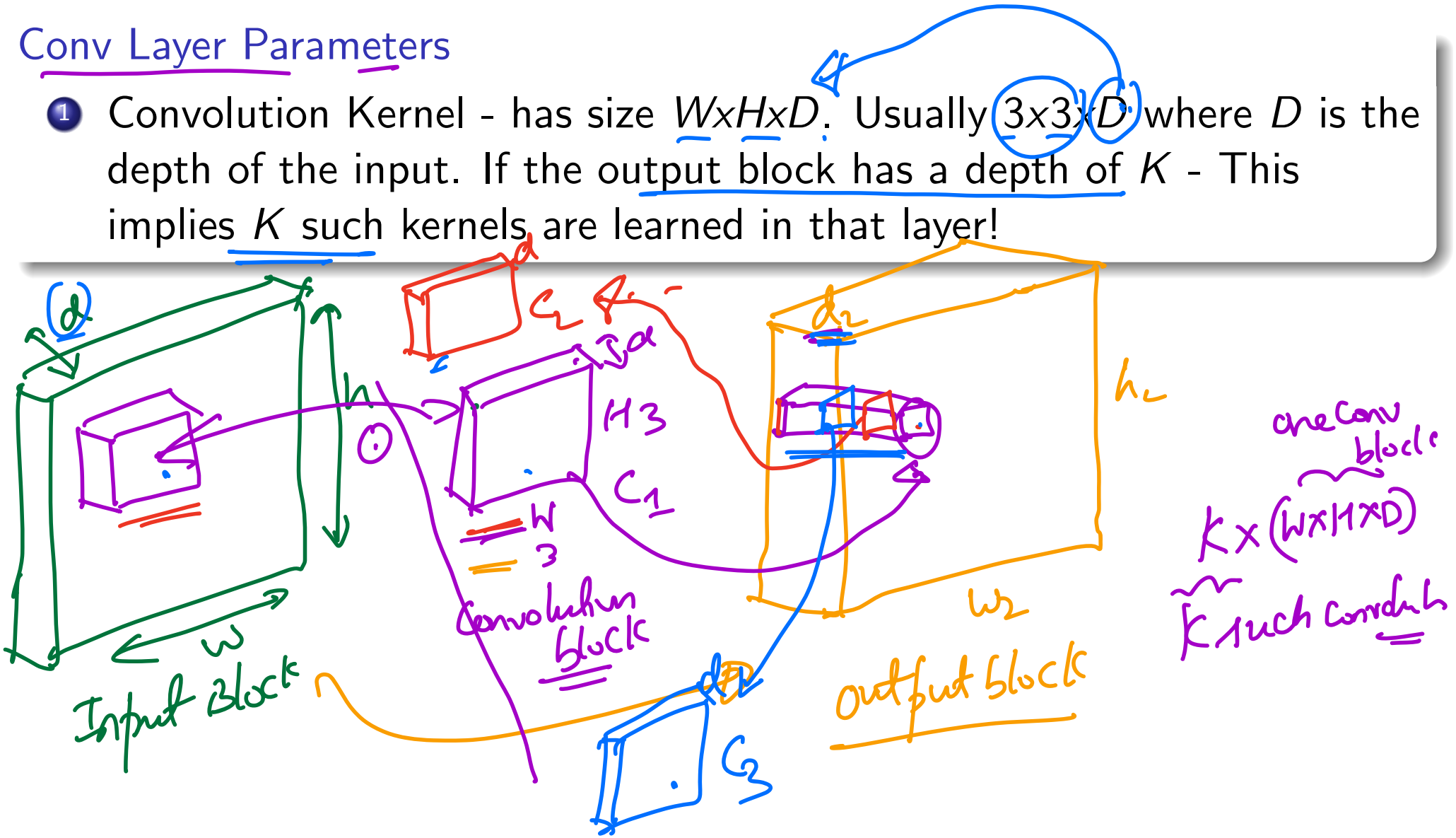




Conv Layer

Conv Layer Parameters

1. Convolution Kernel - has size $WxHxD$. Usually $3x3xD$ where D is the depth of the input. If the output block has a depth of K - This implies K such kernels are learned in that layer!



Conv Layer

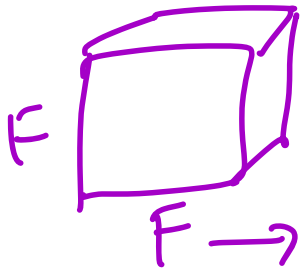
Conv Layer Parameters

1. Convolution Kernel - has size $WxHxD$. Usually $3x3xD$ where D is the depth of the input. If the output block has a depth of K - This implies K such kernels are learned in that layer!

Conv Layer Hyper-Parameters

1. K or depth of the output block or the number of convolution kernels/filters
2. Stride Length, S: How much to shift the convolution kernel by when passing through the input
3. Zero-Padding, P: How much to pad the input before convolution (this impacts the output size!)

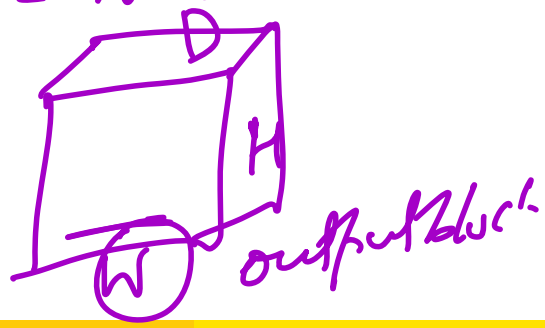
Conv Layer



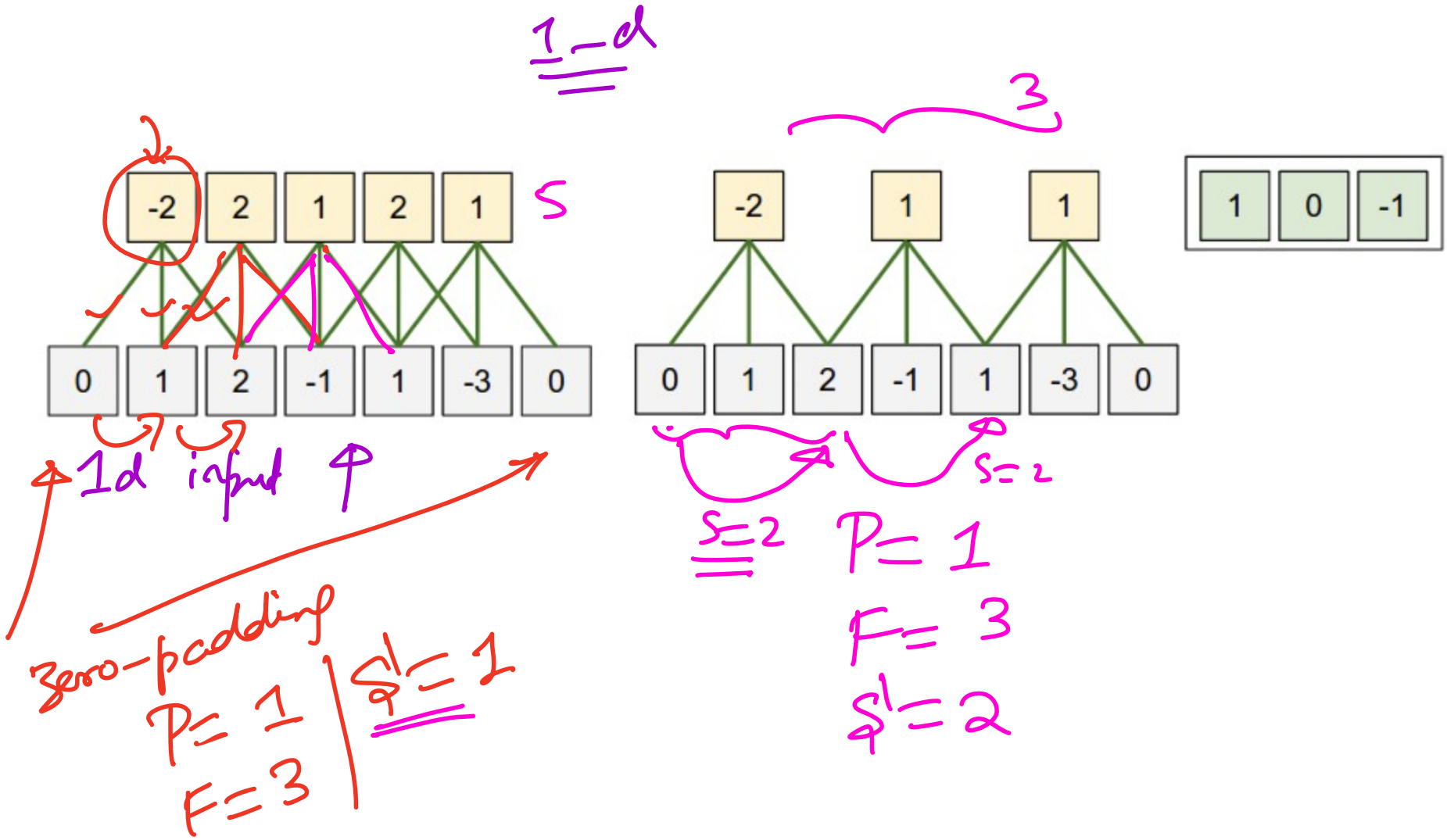
Receptive Field size of a conv kernel

1. Let F be the receptive field size of the convolution Kernel
2. Let S be the stride length
3. Let P be the zero-padding
4. Width of the output block is now $(W - F + 2P)/S + 1$!

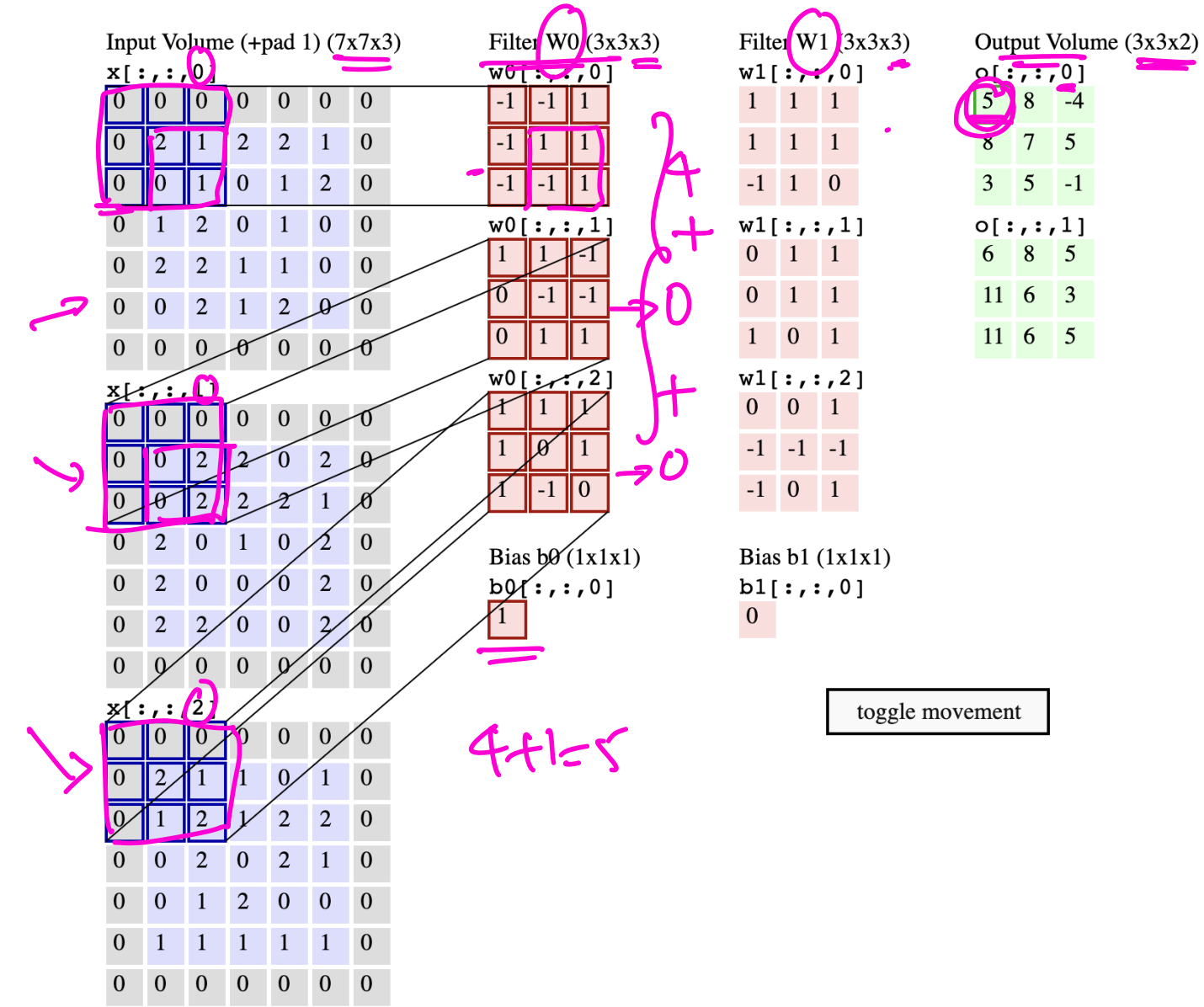
$S=1$, $P=0$, F $\Rightarrow$ output width $= W - F + 1$



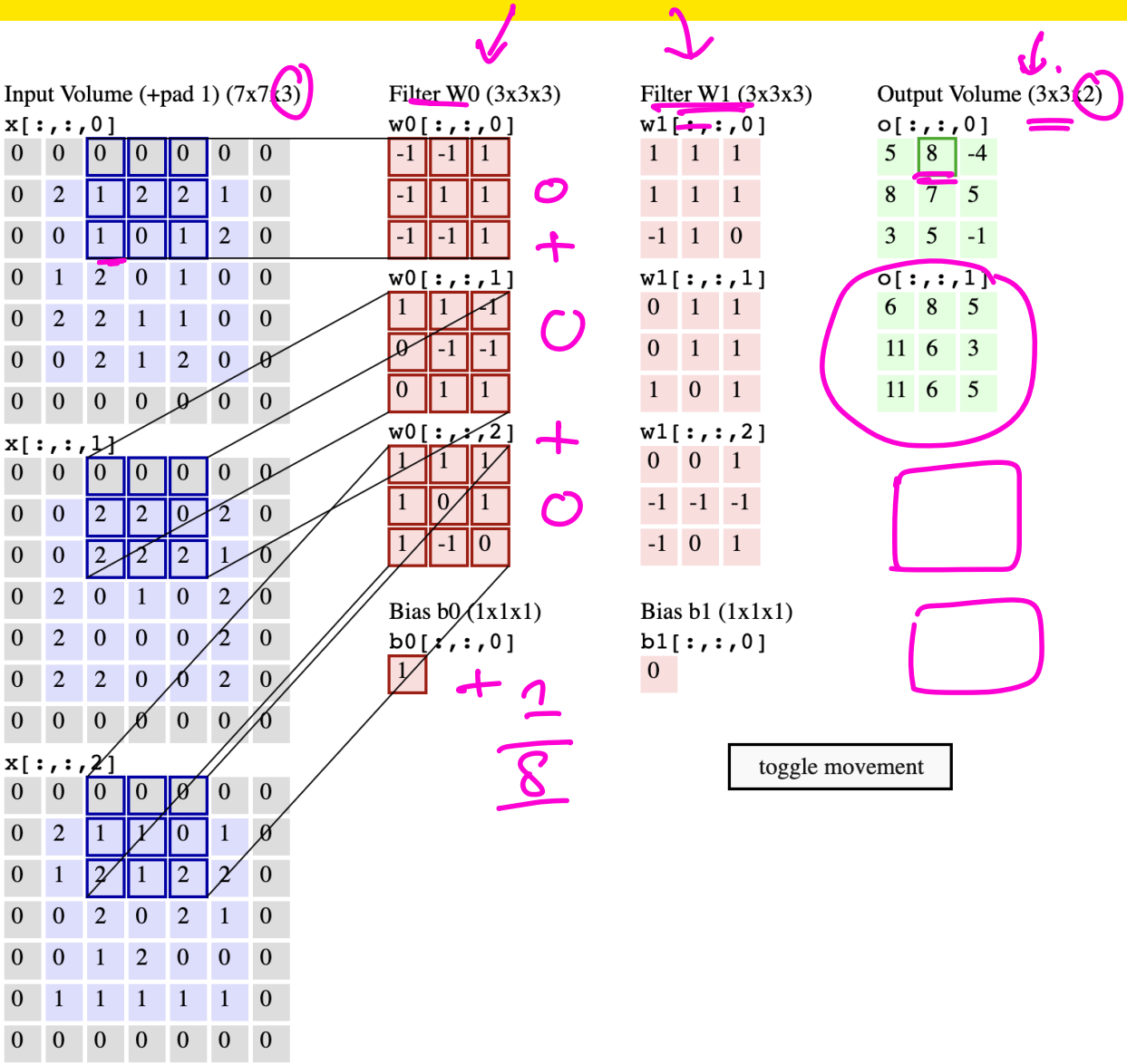
Convolution with Strides



Conv Layer Computations



Conv Layer Computations



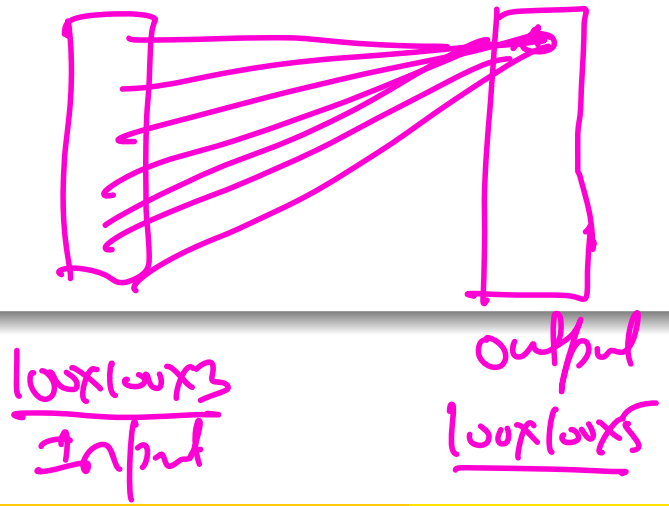
Conv Layer Computation Animation

ICE #1

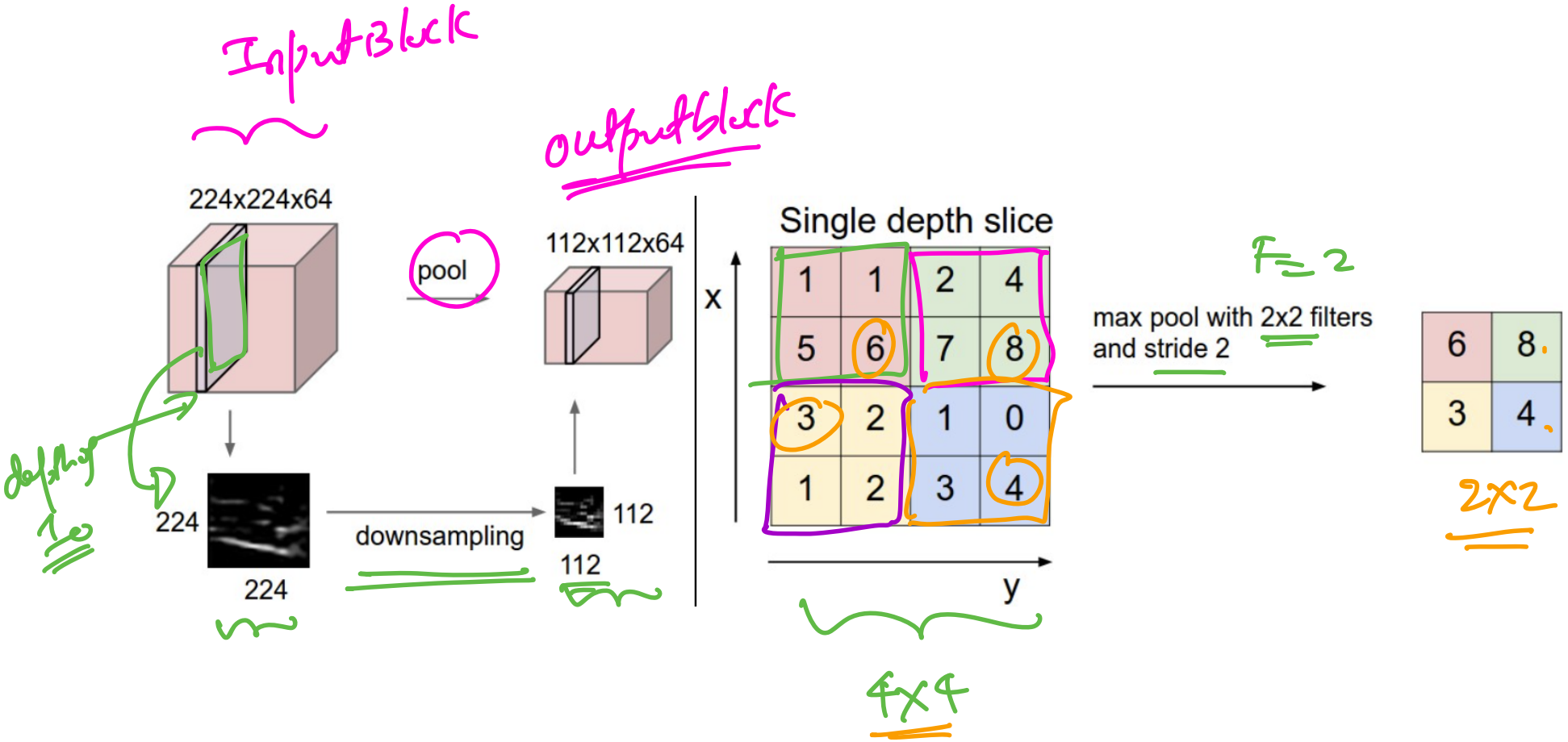
FC vs CNN

Consider an input block (let's say an input image) of size $100x100x3$ (width, height, depth). Depth obviously corresponds to R, G, B. Let's say the first layer was a Conv layer with 5 kernels of size $3x3x3$ with no zero-padding and a stride length of 1. Note that the output block size is $98x98x5$. The number of parameters in this conv layer and the number of parameters if there was a FC layer instead are closest to:

1. 135 and 1.5 million
2. 1.5 million and 135
3. 135 and 1.5 billion
4. 125 and 100 million



Pooling Layer - Max Pooling Example



Pooling Layer

1. Reduces size of layers in CNN and hence reduces number of parameters

Pooling Layer

1. Reduces size of layers in CNN and hence reduces number of parameters
2. Usually $F = 2, S = 2$, i.e non-overlapping pooling with $2x2$ size - Downsample each dimension by 2!

Pooling Layer

1. Reduces size of layers in CNN and hence reduces number of parameters
2. Usually $F = 2, S = 2$, i.e non-overlapping pooling with $2x2$ size - Downsample each dimension by 2!
3. In pooling - Depth doesn't change from input to output layer. So pool across each depth slice. Contrast this with conv layer - where depth of output depends on the number of convolution kernels K, used!

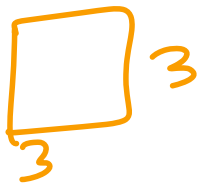
Pooling Layer

1. Reduces size of layers in CNN and hence reduces number of parameters
2. Usually $F = 2, S = 2$, i.e non-overlapping pooling with $2x2$ size - Downsample each dimension by 2!
3. In pooling - Depth doesn't change from input to output layer. So pool across each depth slice. Contrast this with conv layer - where depth of output depends on the number of convolution kernels K, used!
4. Pooling can be max or average - Max pooling works best!

ICE #2

Consider you are max pooling with $F = 3$ and stride length of 3. By what percentage have the input block neurons been reduced to in the output block after max pooling?

1. 75 %
2. 80 %
3. 85 %
4. 90 %



$\frac{1}{9}$ I

$\left(1-\frac{1}{9}\right) \times 100 \approx 90\%$

Real World Example of Conv Net used

Image Net Competition Winner 2012

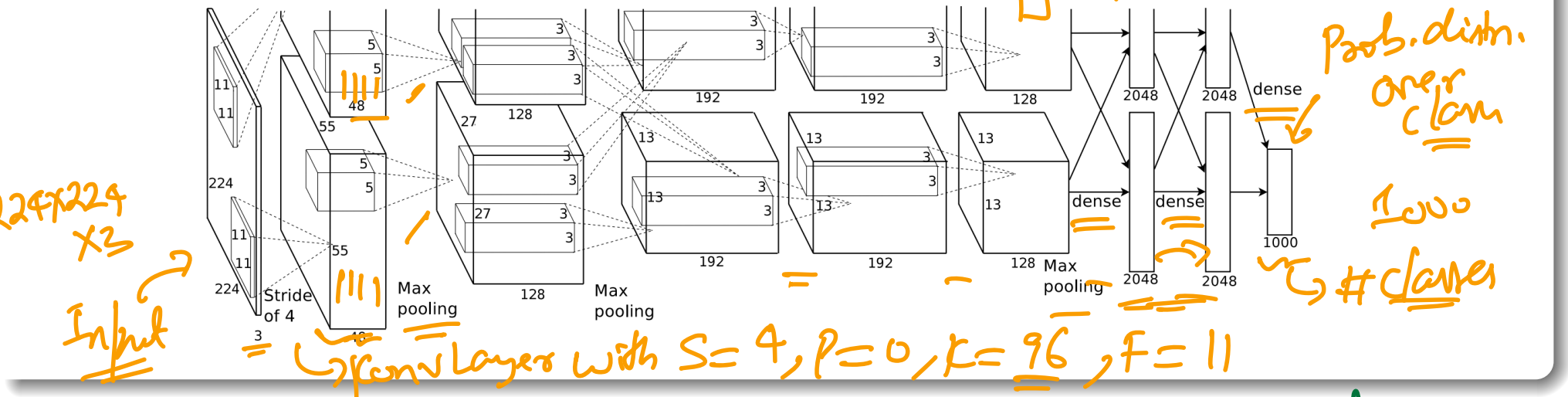
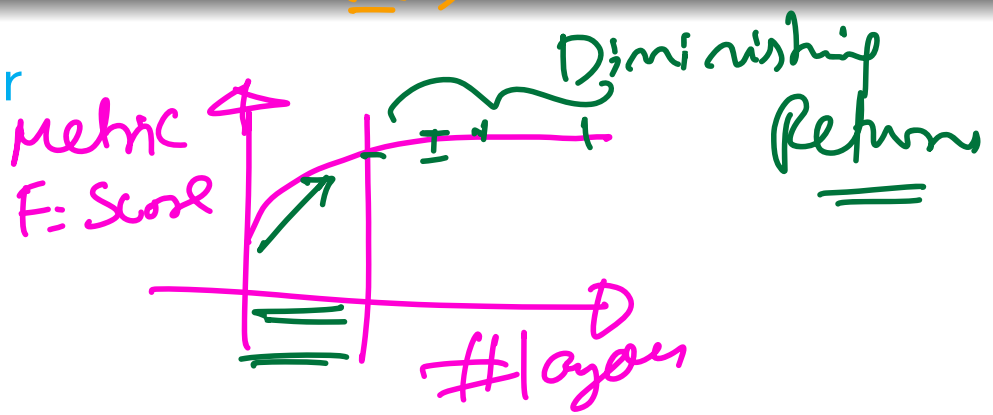


Image Net 2012 Competition Winner



96 depth activations learned in First Conv Layer

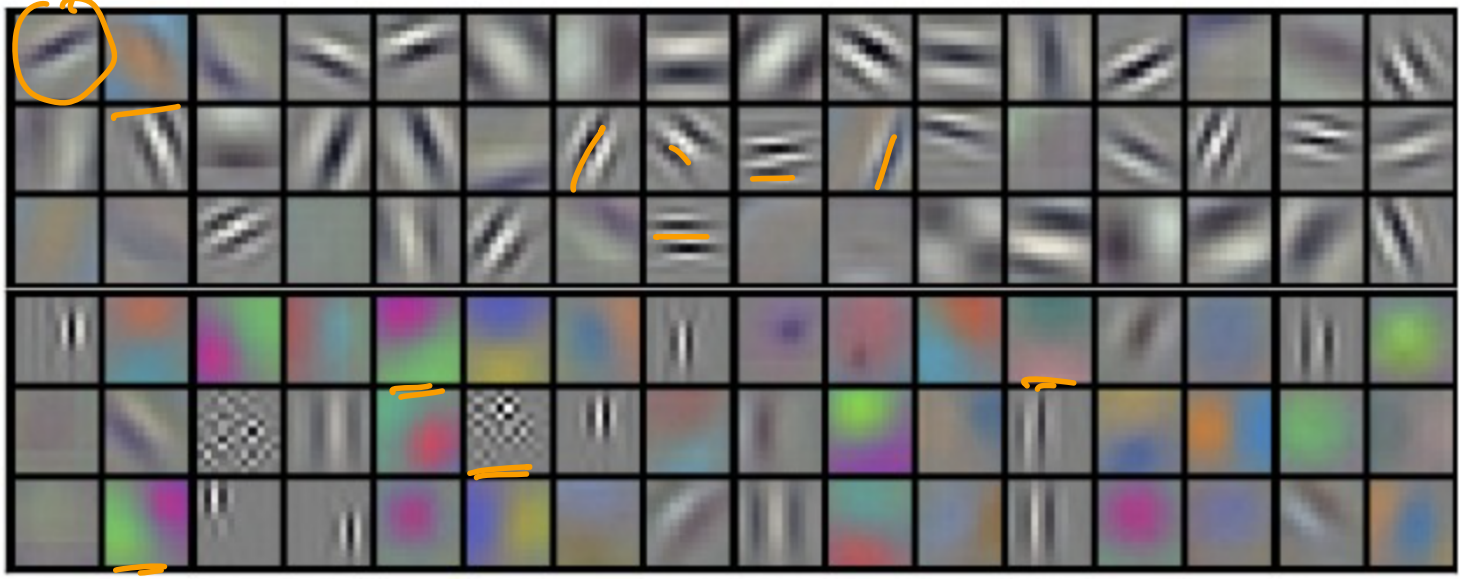


Image Net 2012 competition prize paper

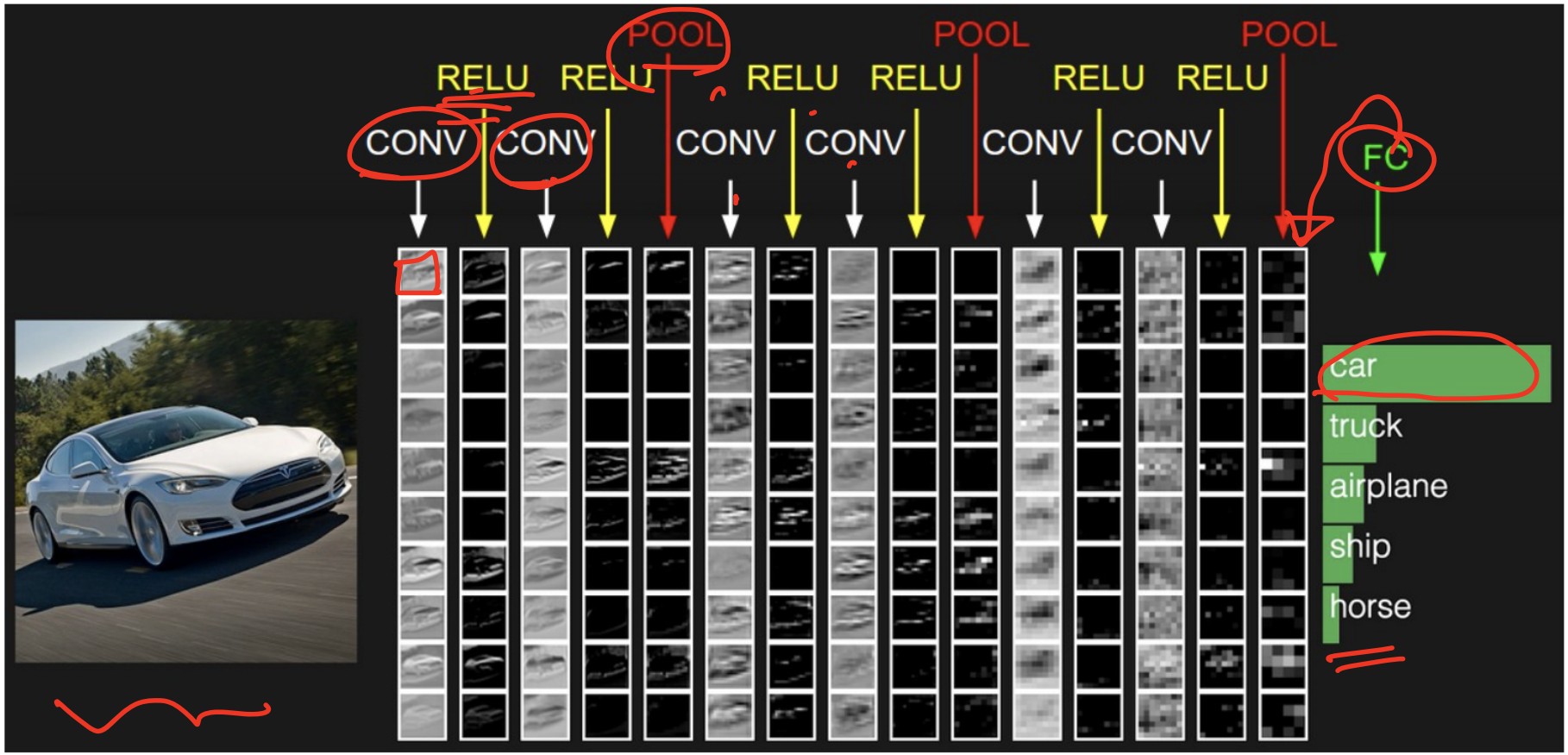
ICE #3

Max Pool Dimensions

Consider you are max pooling with $F = 2$ and stride length of 2. Let the input block be $128x128x15$. What would be the dimensions of the output block after max pooling?

1. 64x64x3
2. 128x64x15
3. 64x64x45
4. 64x64x15

CNN Layers example

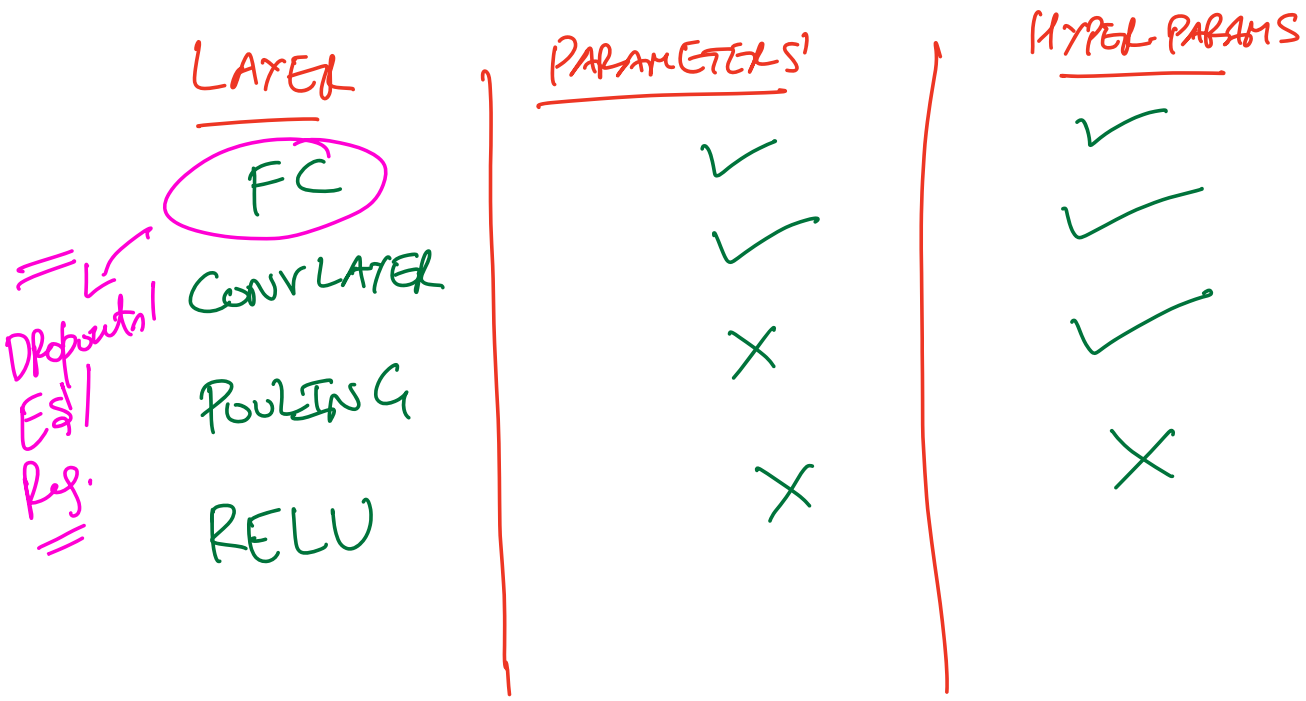


ICE #4

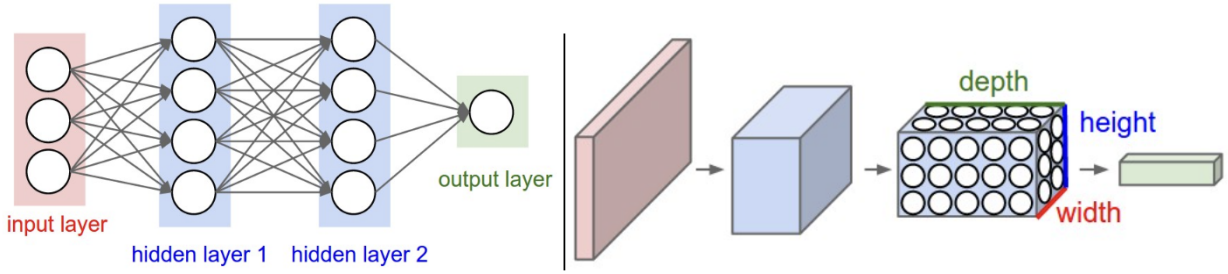
RELU

Which of the following layers have parameters associated with it and which have neither parameters nor hyper parameters?

1. (FC, Conv Layer) and (Max Pool)
2. (FC) and (RELU)
3. (FC, Conv Layer) and (RELU) ✓
4. (FC, Conv Layer) and (RELU, Max Pool)


LAYER
PARAMETERS'
HYPER PARAMS
FC
CONV LAYER
POOLING
RELU
Dropout
Eg'l
Reg.

CNN vs NN



1. Fundamental unit in CNN is a block (with width, W, height H, and depth D). Fundamental unit in NN is a vector of neurons.
2. NN only has a feedforward connection (mostly) from one vector of neurons to another. CNN has 3 different types of connections - FC, Conv, and Pooling.
3. NN has full connectivity. CNN has local connectivity (e.g. conv Layer and Pooling) shared parameters
4. Feedforward NN parameter space would be prohibitively large for Images. Conv Nets have shared parameter space and keep the parameter space manageable.

Next Lecture

1. Popular Conv Nets that have worked in practice
2. Intuition behind some of the archs