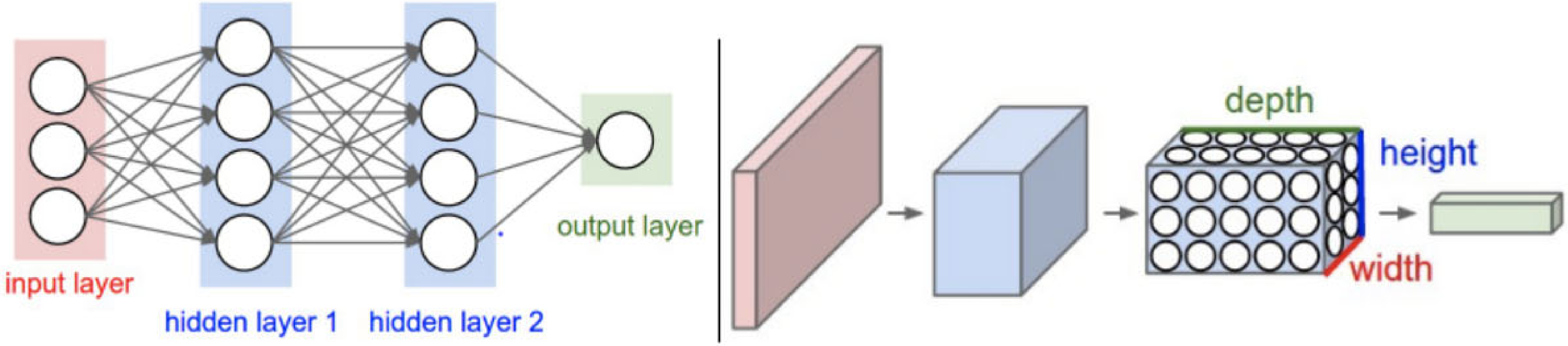
Machine Learning
Multilayer Networks

Jian Liu

Part 1: MLP

Part 2: CNN

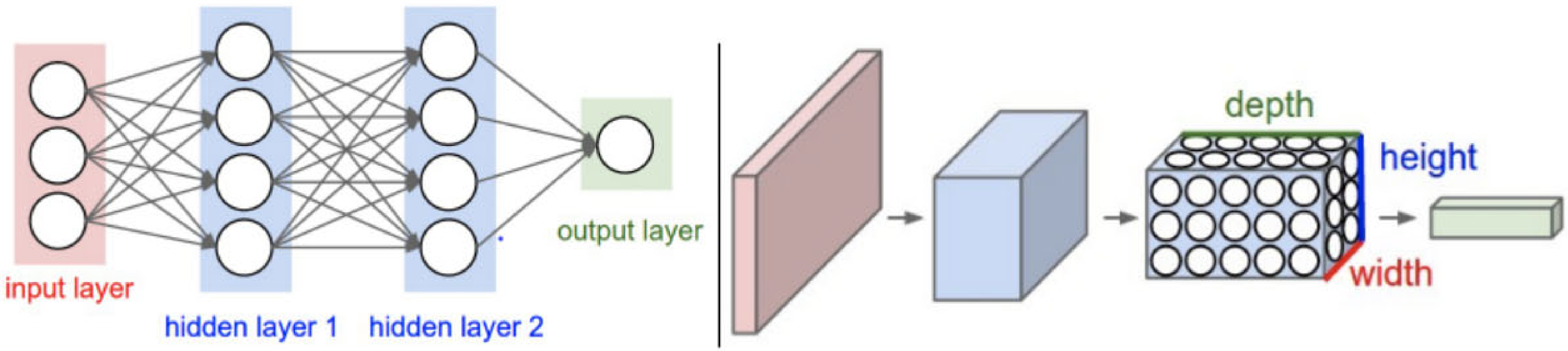


MLP towardsdatascience CNN

Machine Learning
Multilayer Networks

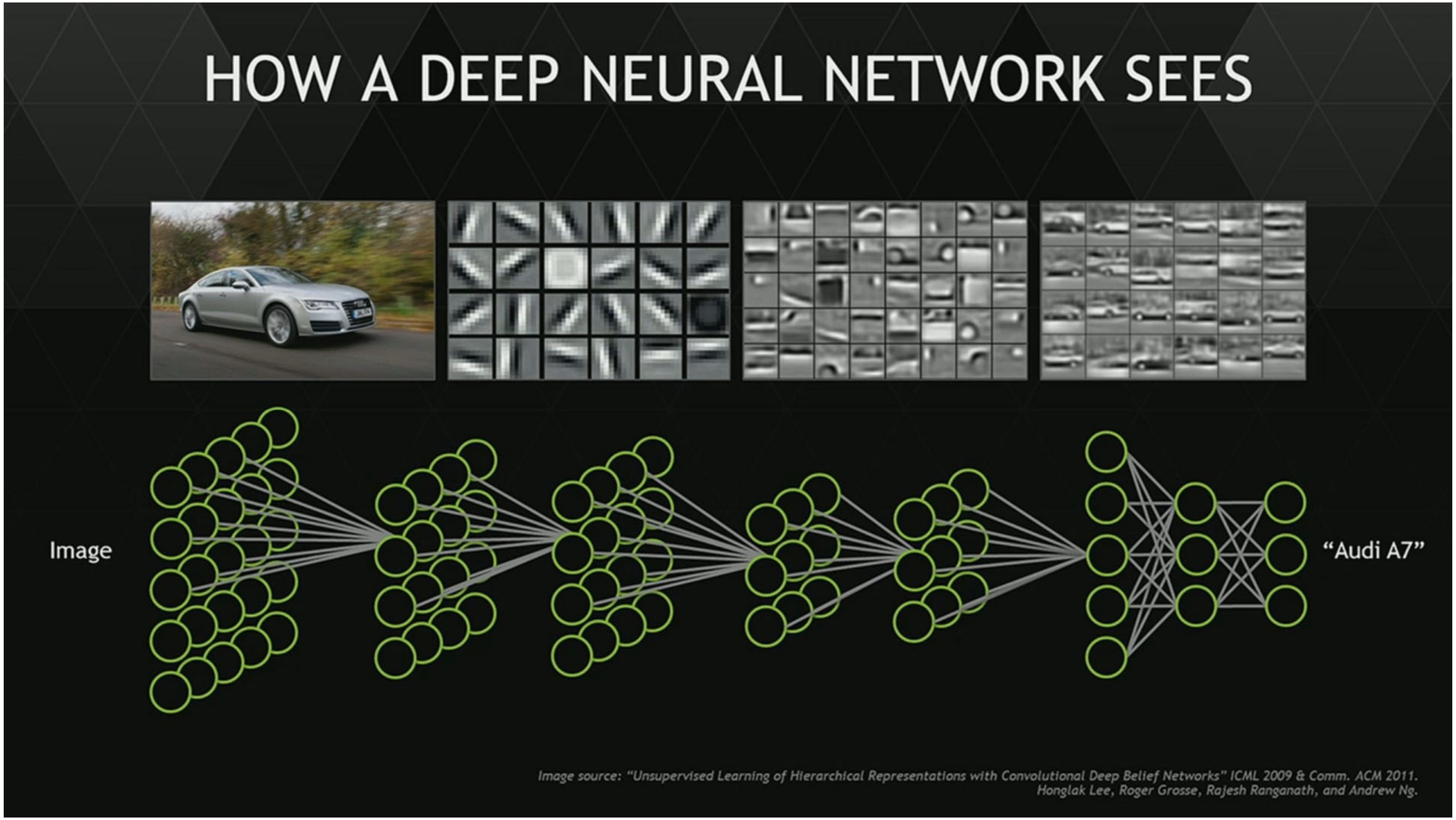
Jian Liu

Part 2: CNN

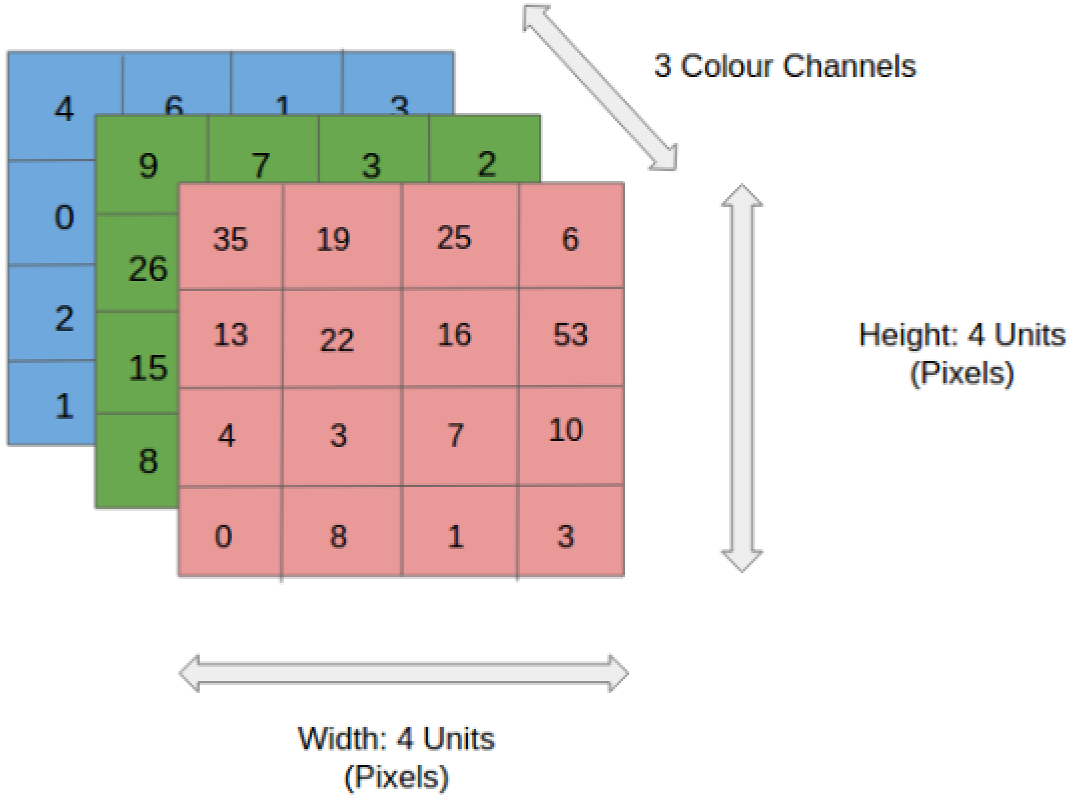


MLP towardsdatascience CNN

Why Convnets?

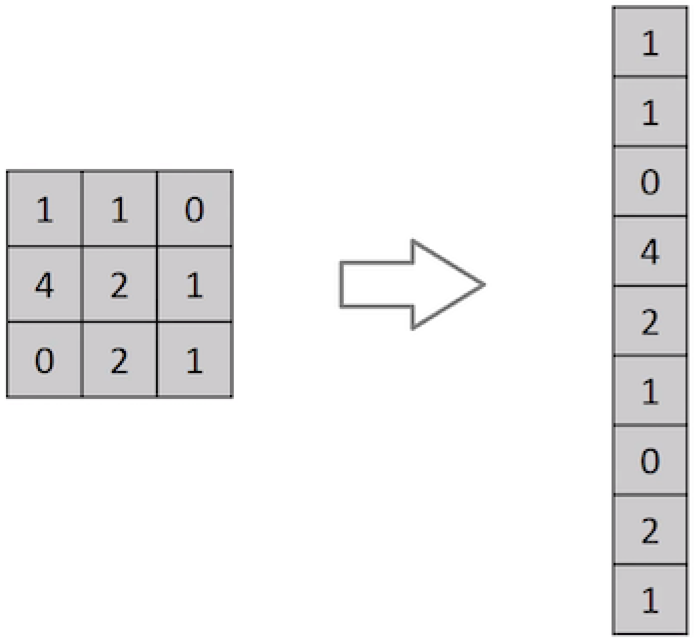


Why Convnets?



4x4x3 RGB Image

Why Convnets?

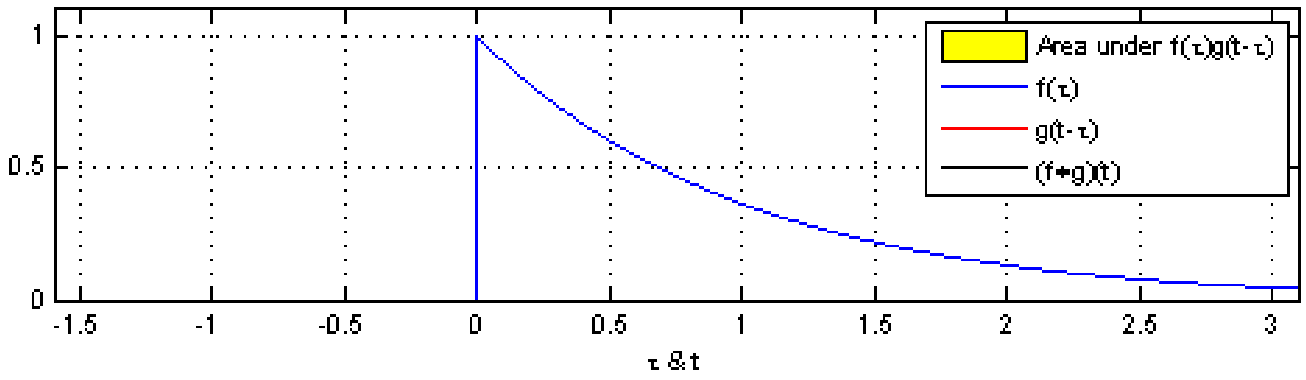


Flattening of a 3x3 image matrix into a 9x1 vector

A ConvNet is able to successfully capture the **Spatial and Temporal dependencies** in an image through the application of relevant **filters**.

What is the convolution?

$$f(x)*g(x)=\int_{-\infty}^{\infty} f(\tau)g(x-\tau)d\tau$$



[From Wikipedia]

Filter application

0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	1	1	1	1	1	0
0	1	0	0	0	1	0
0	1	1	1	1	1	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

1				

1	1	1
0	0	1
0	0	1

Filter application

0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	1	1	1	1	1	0
0	1	0	0	0	1	0
0	1	1	1	1	1	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

1	1			

1	1	1
0	0	1
0	0	1

Filter application

$$\boldsymbol{x} = \begin{array}{|c|c|c|c|c|c|c|} \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ \hline 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ \hline 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline \end{array}$$

$$\begin{array}{|c|c|c|c|c|} \hline 1 & 1 & 1 & 1 & 0 \\ \hline 1 & 1 & 1 & 2 & 0 \\ \hline 3 & 4 & 4 & 5 & 2 \\ \hline 2 & 2 & 1 & 2 & 1 \\ \hline 2 & 3 & 3 & 3 & 2 \\ \hline \end{array}$$

$$= \boldsymbol{w}^T x + w_0$$

$$\boldsymbol{w} = \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 0 & 0 & 1 \\ \hline 0 & 0 & 1 \\ \hline \end{array} \quad + w_0$$

The filter can be implemented with a neuron!

However, the input is not “static” because the filter is slid across the image

Padding

$x=$

0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	1	1	1	1	1	0	0
0	0	1	0	0	0	1	0	0
0	0	1	1	1	1	1	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

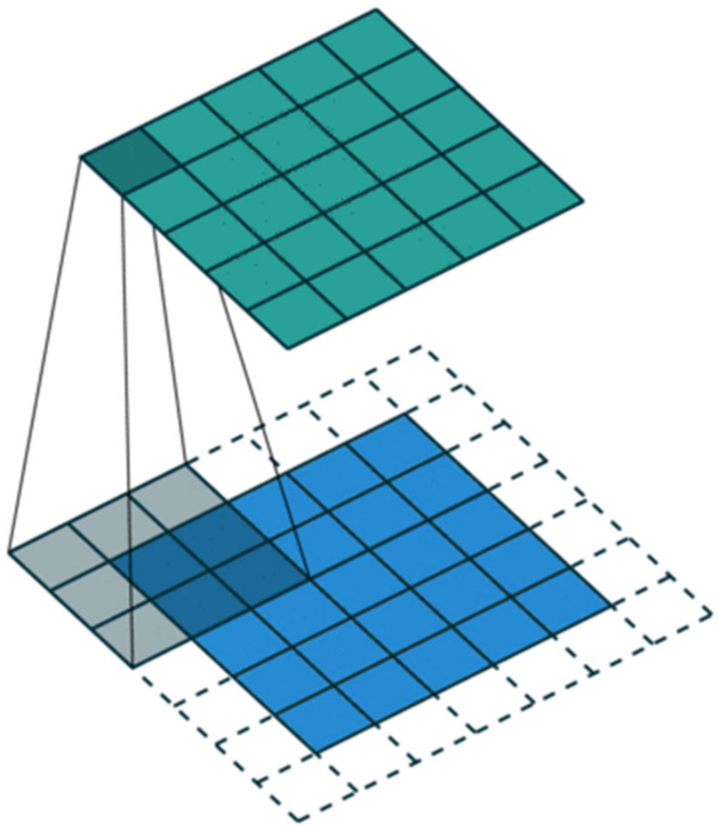
0	0	0	0	0	0	0
1	1	1	1	1	0	
...	1	1	1	2	0	
	3	4	4	5	2	
	2	2	1	2	1	
	2	3	3	3	2	

The application of the filter would reduce the size of the image. This can be prevented by padding the image, typically with zeros.

$w=$

1	1	1
0	0	1
0	0	1

Padding



SAME padding: 5x5x1 image is padded with 0s
to create a 6x6x1 image

Same Padding.
augment the 5x5x1 image into a 6x6x1 image and then apply the 3x3x1 kernel over it, we find that the convolved matrix turns out to be of dimensions 5x5x1.

Stride

$$x = \begin{array}{|c|c|c|c|c|c|c|} \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ \hline 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ \hline 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline \end{array}$$

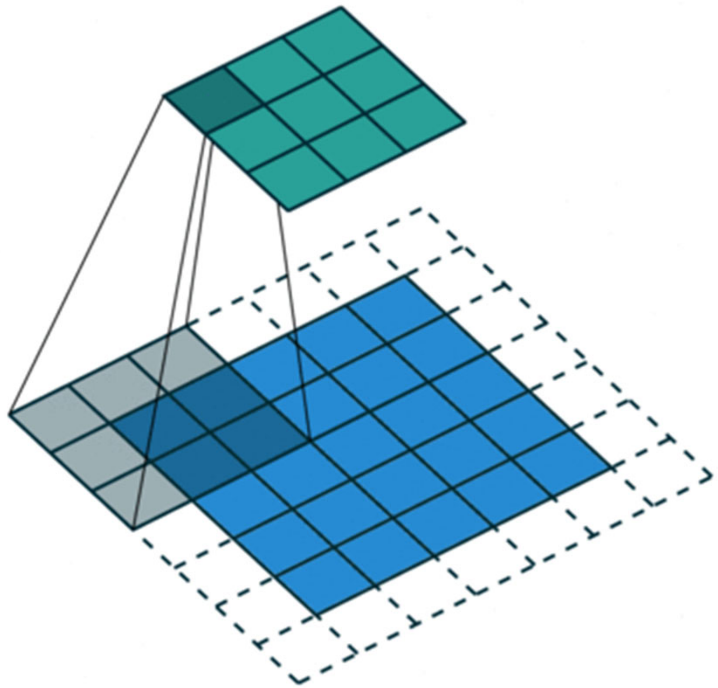
$$\begin{array}{|c|c|c|} \hline 1 & 1 & 0 \\ \hline 3 & 4 & 2 \\ \hline 2 & 3 & 2 \\ \hline \end{array}$$

The stride can be more than 1, which downsamples the image.

Clearly not all strides are possible. For instance in this image 2 is ok, but 3 would not work.

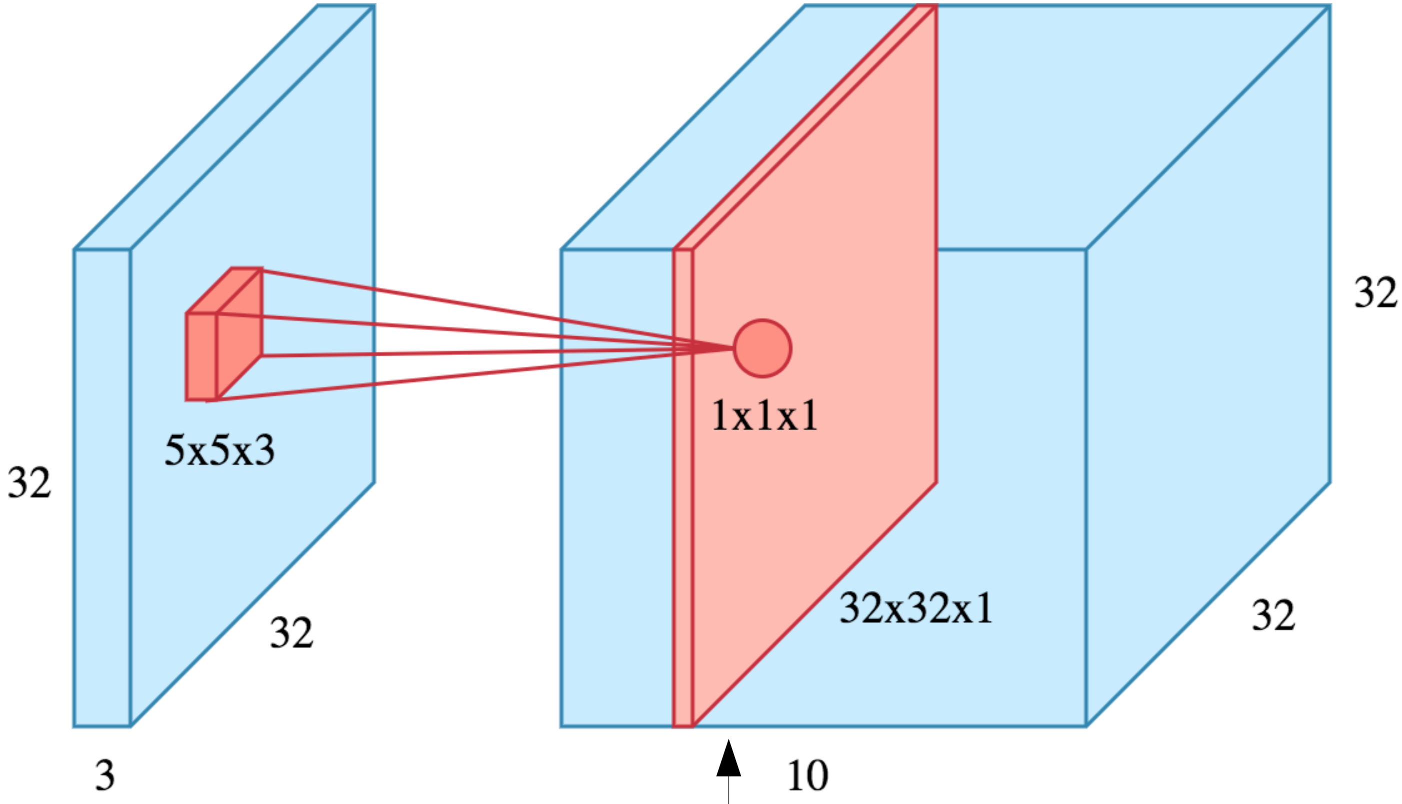
$$w = \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 0 & 0 & 1 \\ \hline 0 & 0 & 1 \\ \hline \end{array}$$

Stride



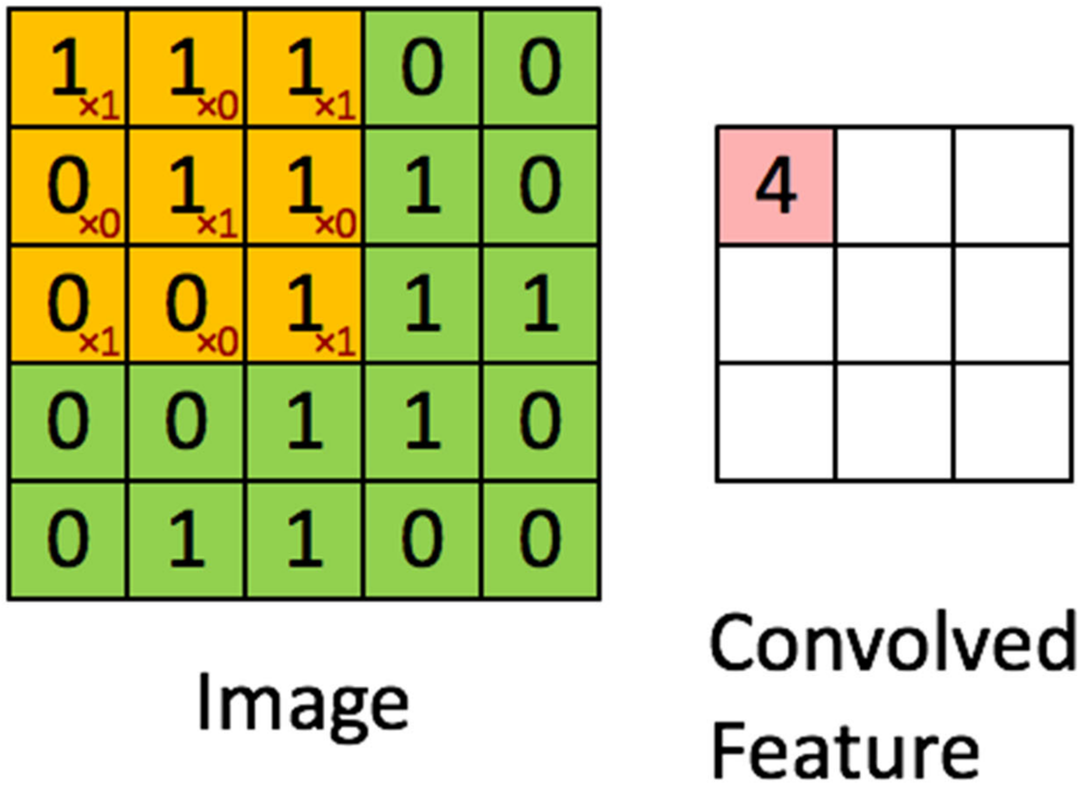
Convolution Operation with Stride Length = 2

Images and filters



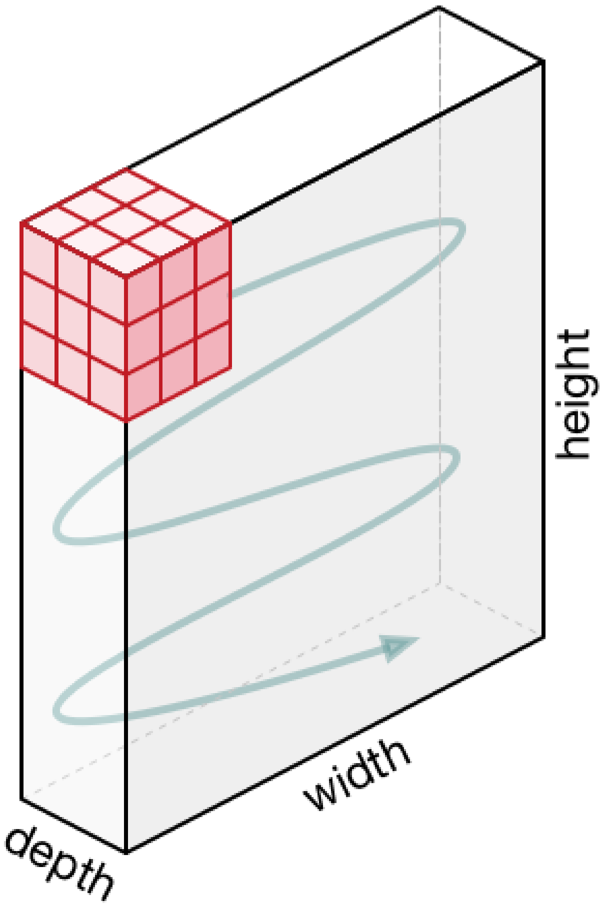
Activation layers (one per filter)

Filter/Kernel



Convoluting a 5x5x1 image with a 3x3x1 kernel to get a 3x3x1 convolved feature

Filter/Kernel

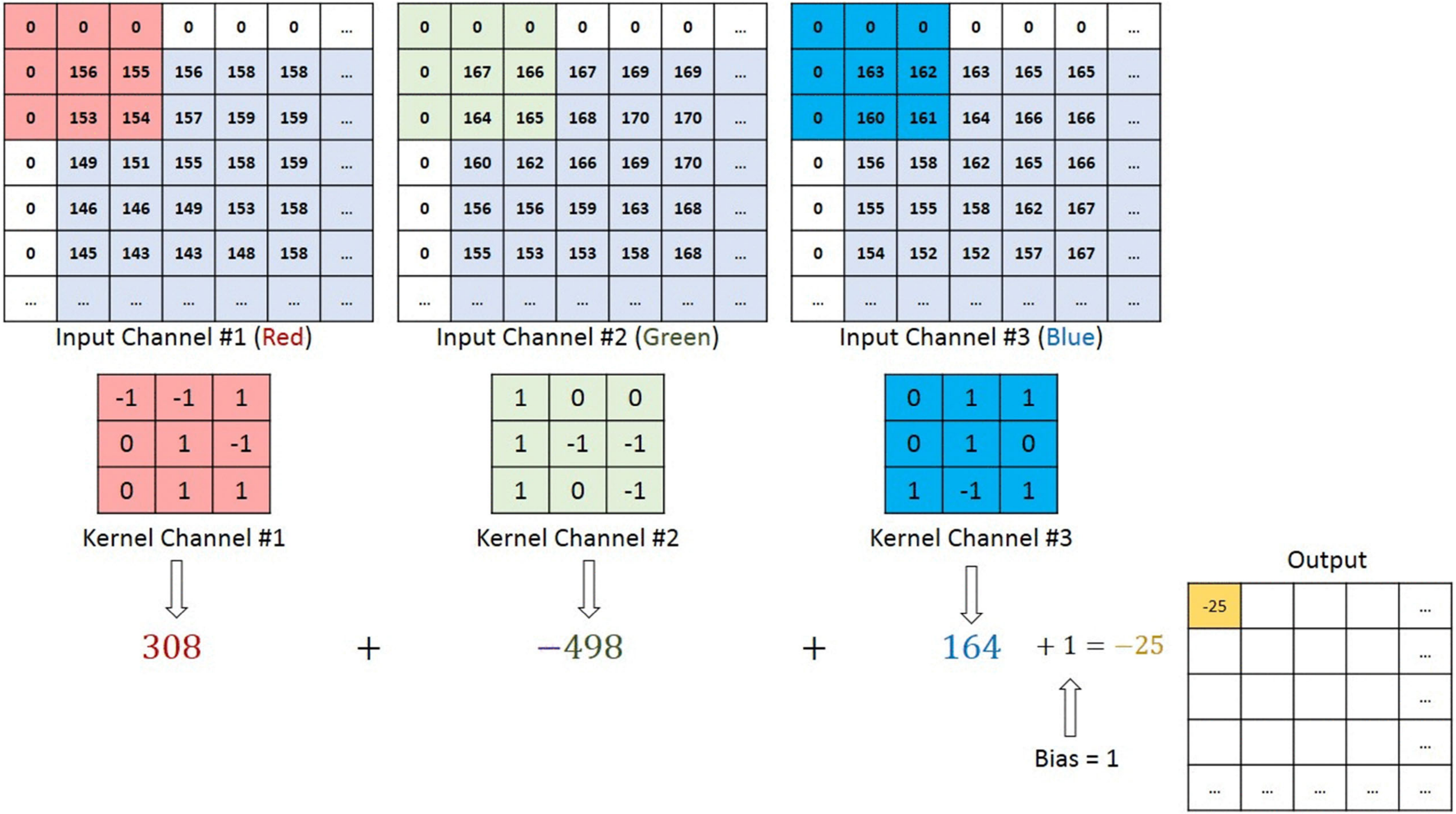


Movement of the Kernel

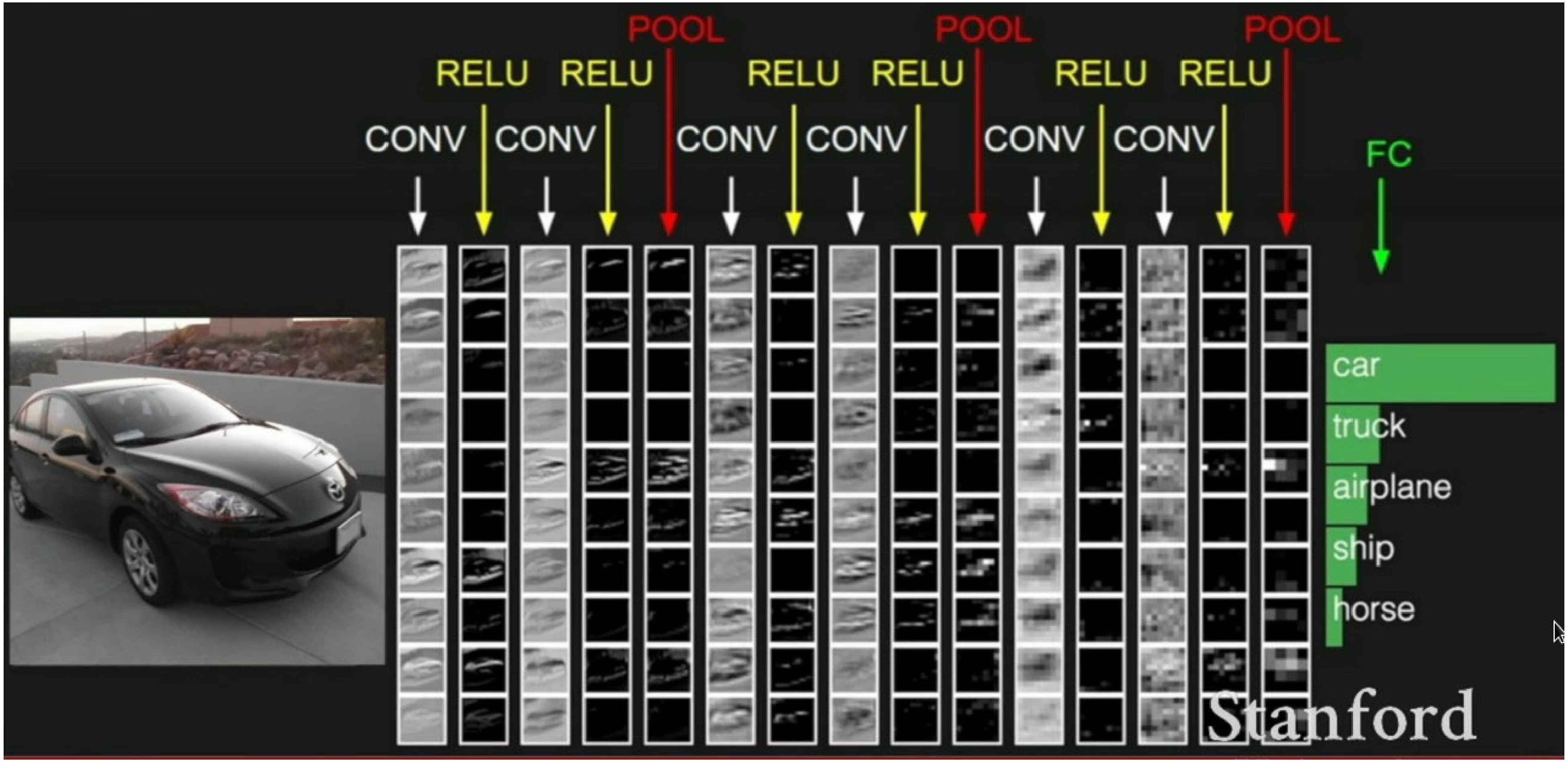
The Kernel shifts 9 times because of **Stride Length = 1 (Non-Strided)**

Every time performing a **matrix multiplication operation between K and the portion P of the image** over which the kernel is hovering

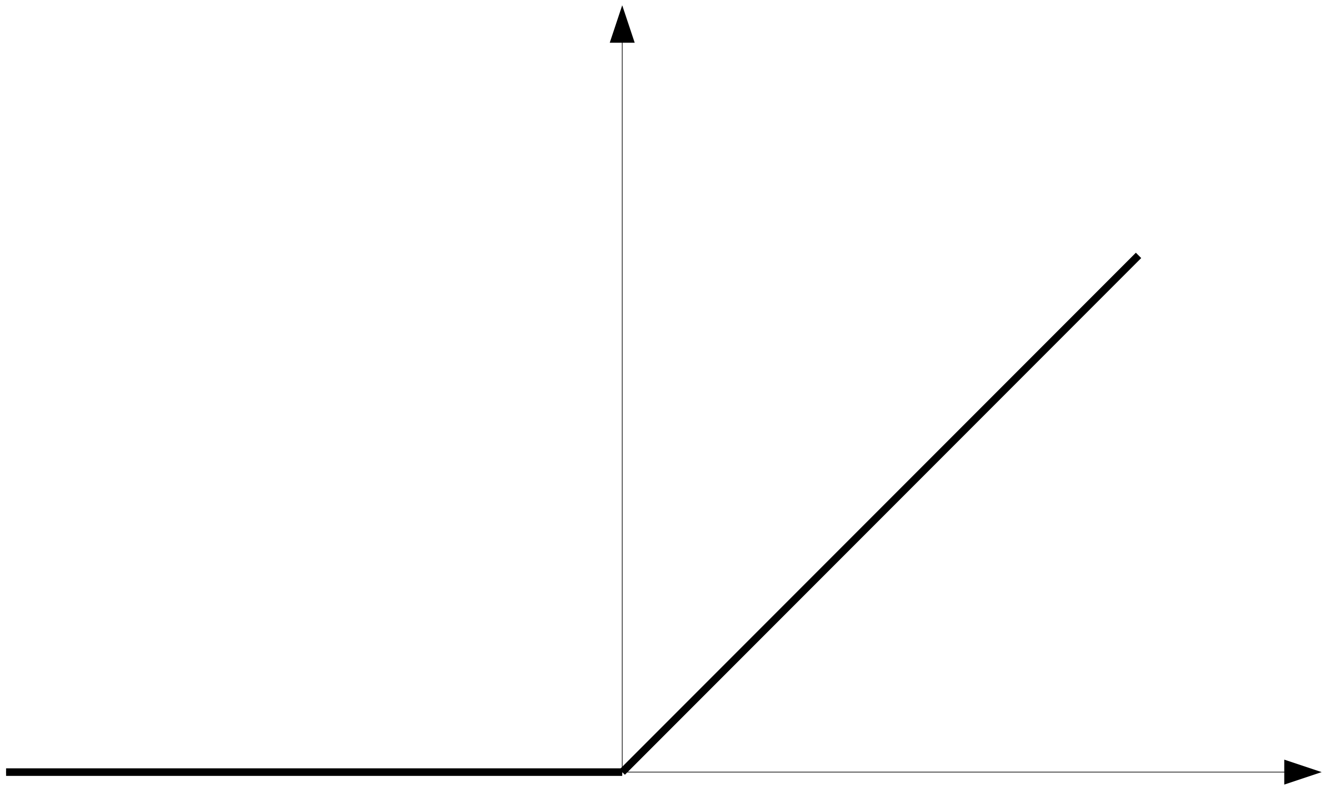
Convolution operation
on a MxNx3 image matrix with a 3x3x3 Kernel



Architecture

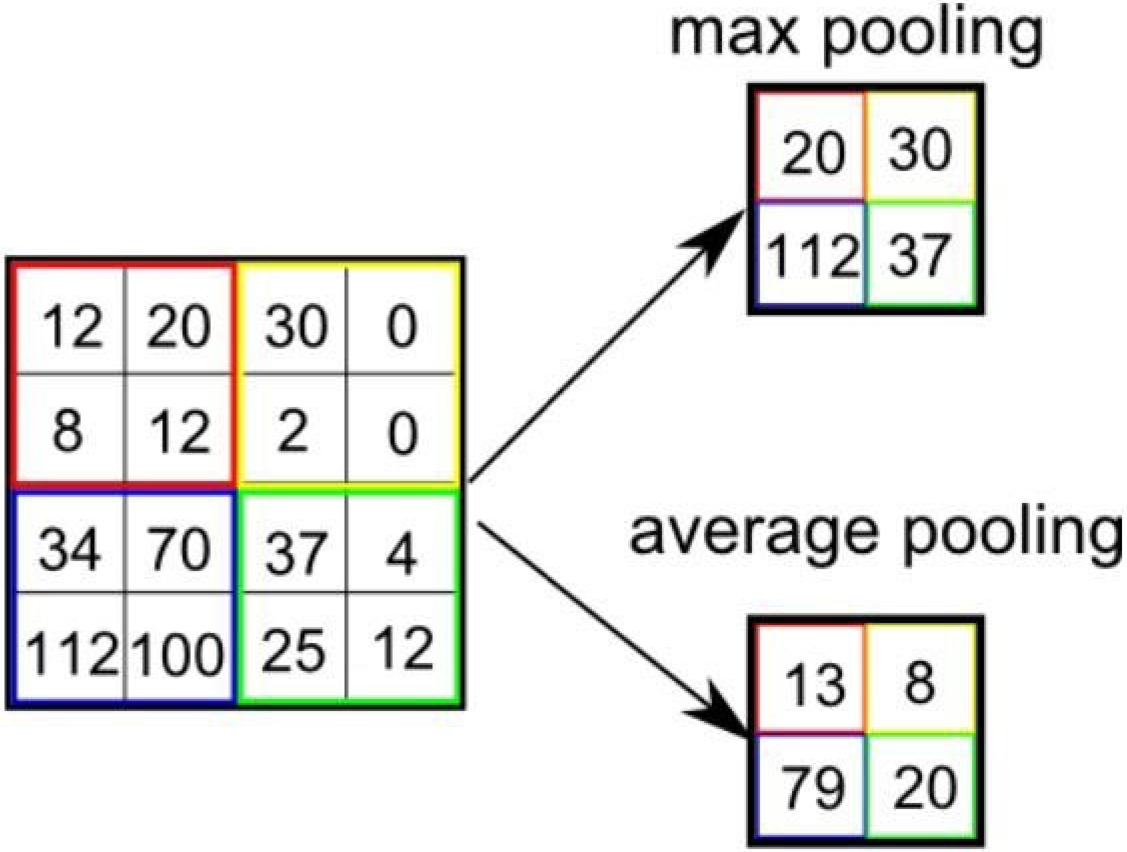


Rectified Linear Units (ReLUs)



$$o(x) = \begin{cases} 0 & \text{if} \quad x < 0 \\ x & \text{if} \quad x \geq 0 \end{cases}$$

Pooling



max pooling
average pooling
12
20
30
0
8
12
2
0
34
70
37
4
112
100
25
12
20
30
112
37
13
8
79
20

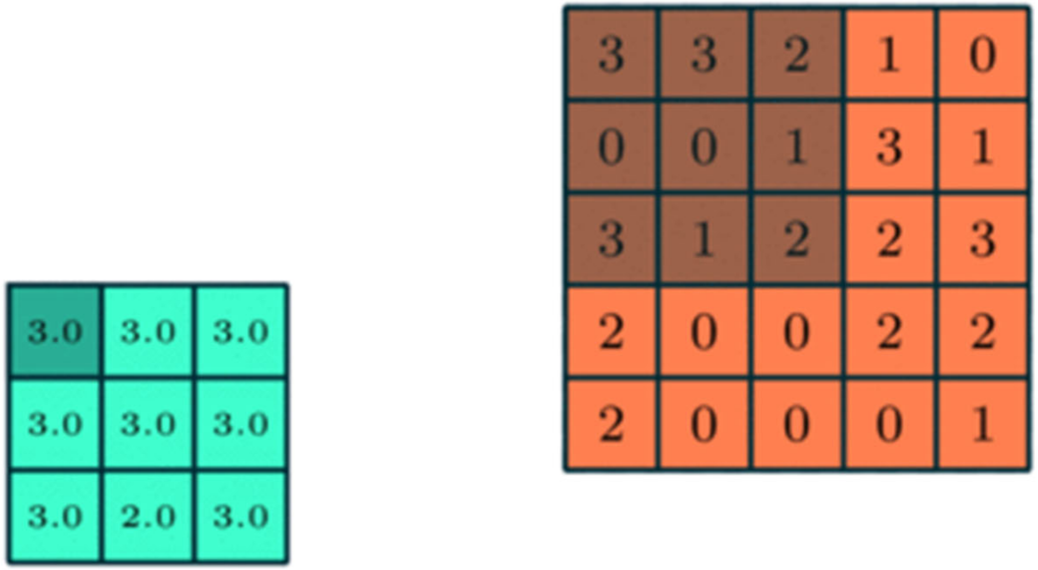
(Max) Pooling

1	1	1	1
1	1	1	2
3	4	4	5
2	2	1	2

1	2
4	5

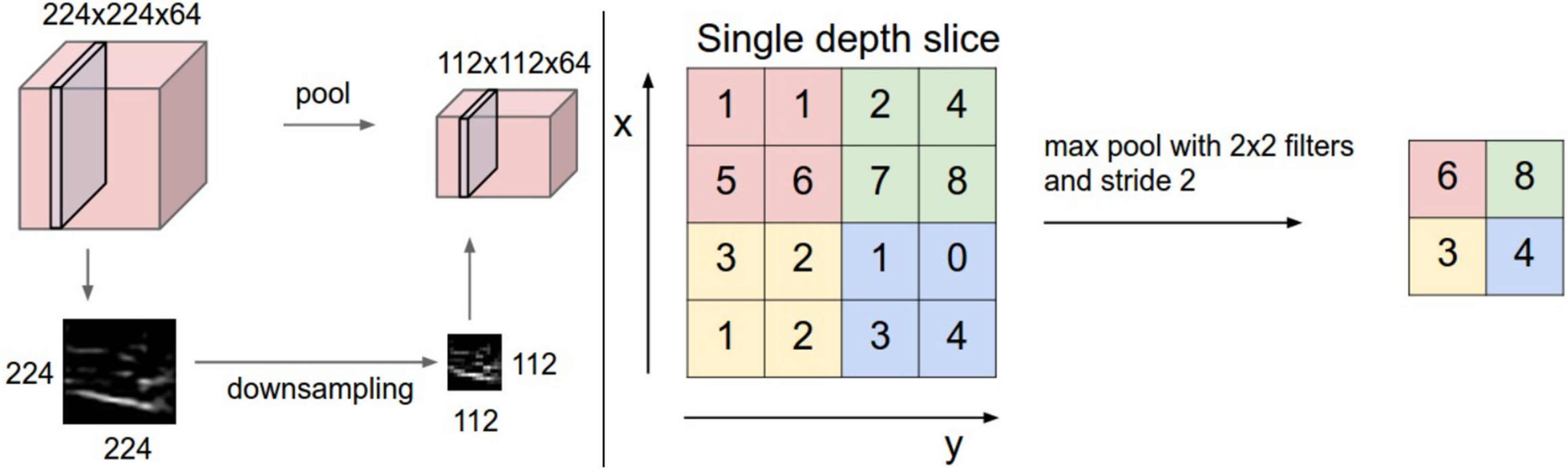
Max pooling is the most common way to downsample the image, in order to focus on higher-level patterns.

(Max) Pooling



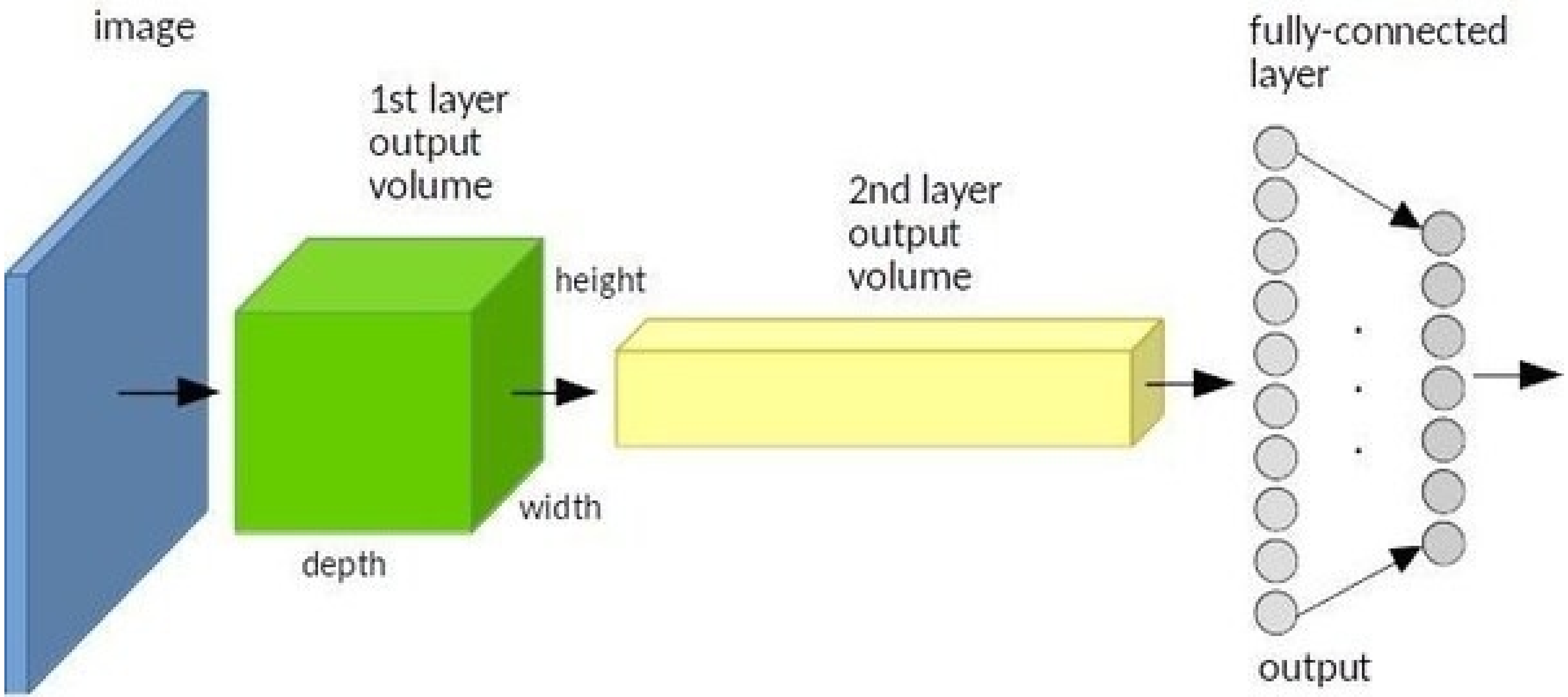
3x3 pooling over 5x5 convolved feature

Pooling

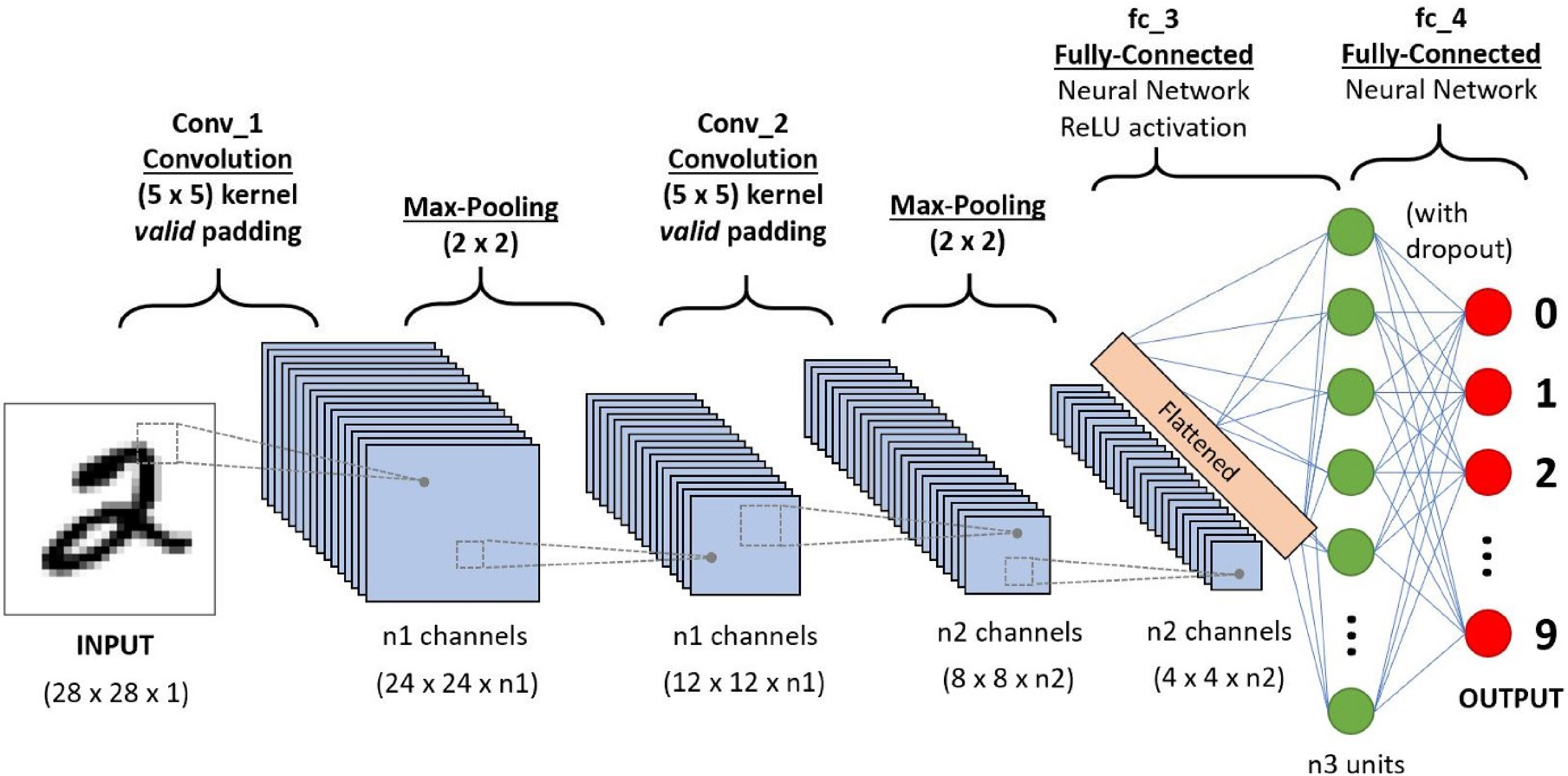


Tsspmrk$pe}iv$hs{rweqtpiwxlizspyqi$wtexmepp}0$
mrhitirhirxp}mriegl$hitxl$wpmgisjxli$mrtyx$zspyqi2

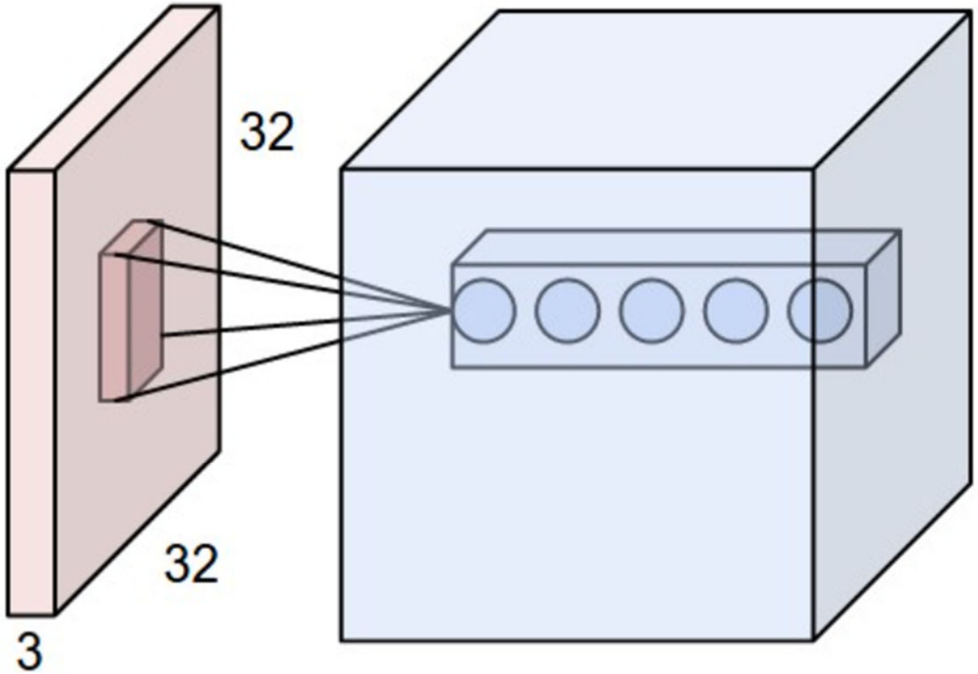
From convolutional to MLP

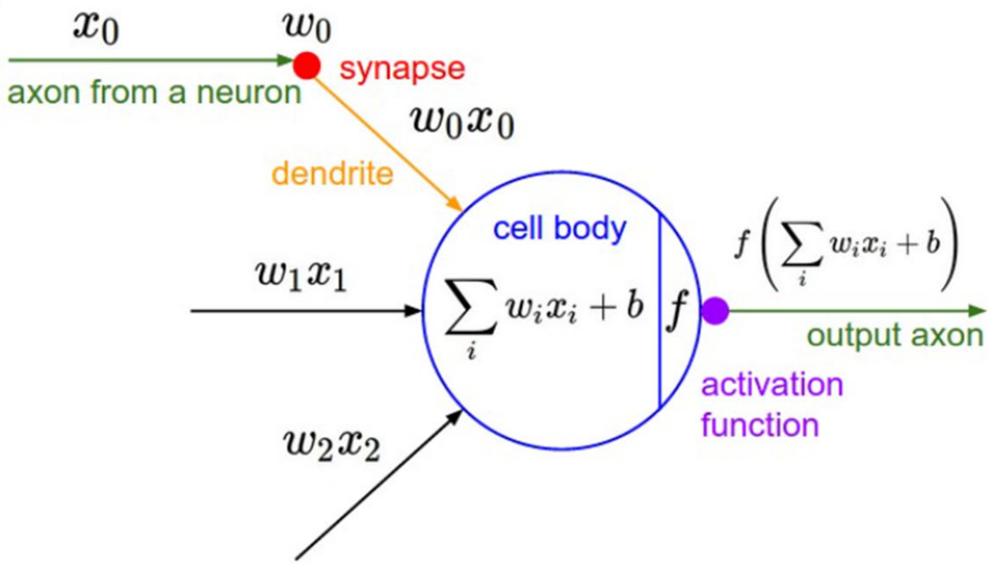


Classification — Fully Connected Layer (FC Layer)



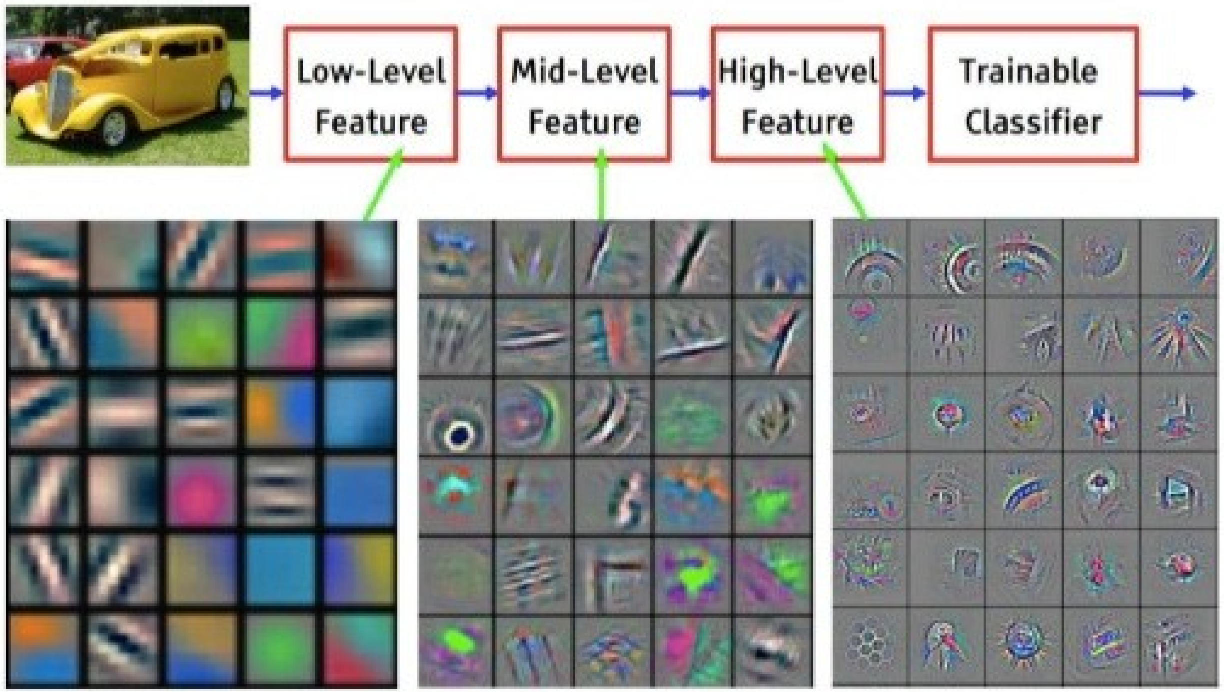
Convnets


32
32
3


x_0
w_0
synapse
axon from a neuron
$w_0 x_0$
dendrite
cell body
$f\left(\sum_i w_i x_i + b\right)$
$w_1 x_1$
$\sum_i w_i x_i + b$
f
output axon
activation
function
$w_2 x_2$

Features

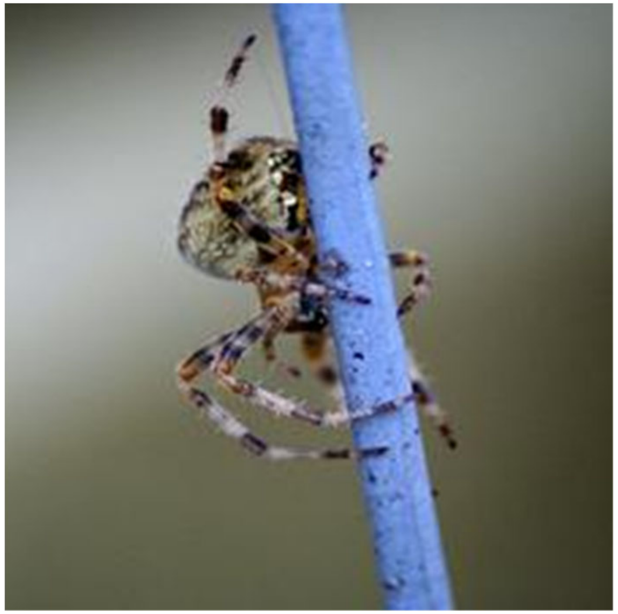
Convolutional Neural Network



Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]

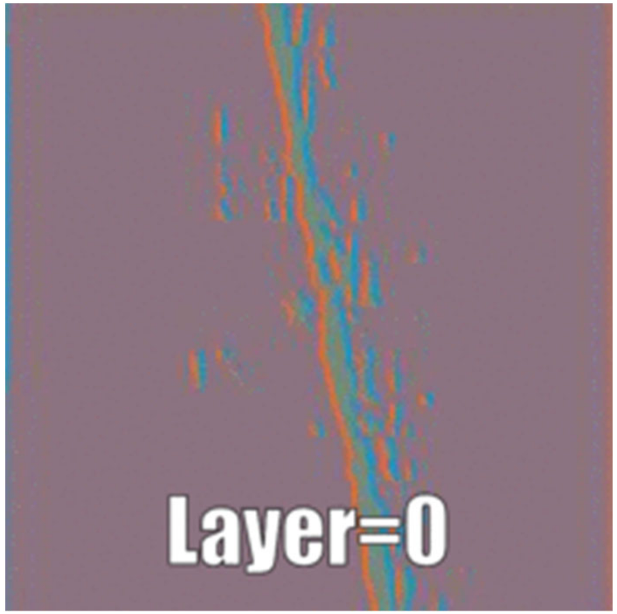
CNN Visualizations

Input Image



Layer Vis. (Filter=0)

Filter Vis. (Layer=29)



https://github.com/utkuozbulak/pytorch-cnn-visualizations

Architectures

LeNet LeCun et al. in 1998

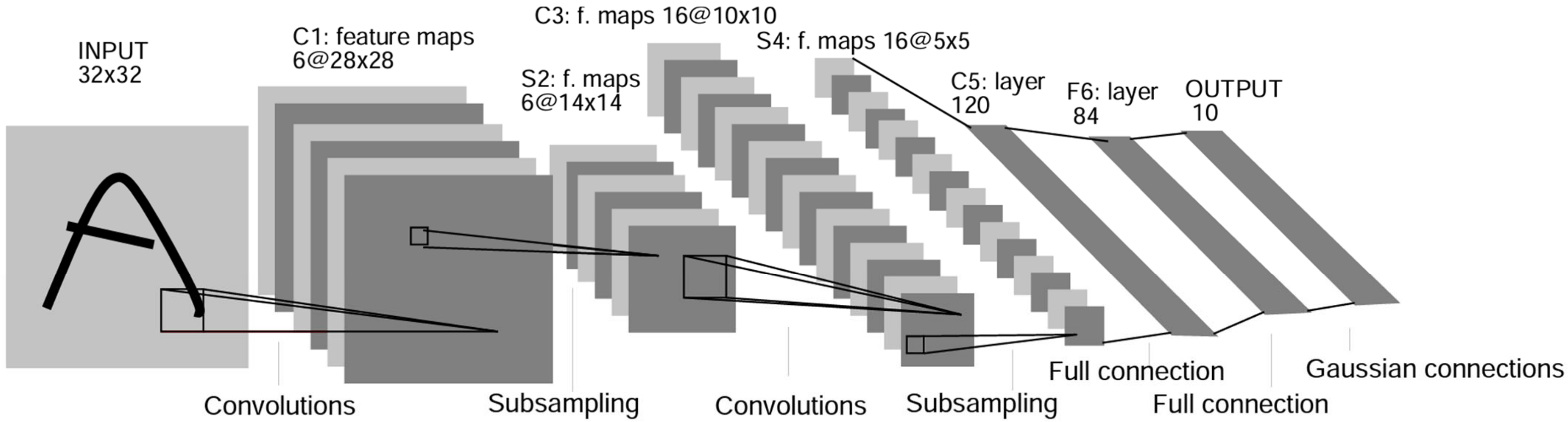
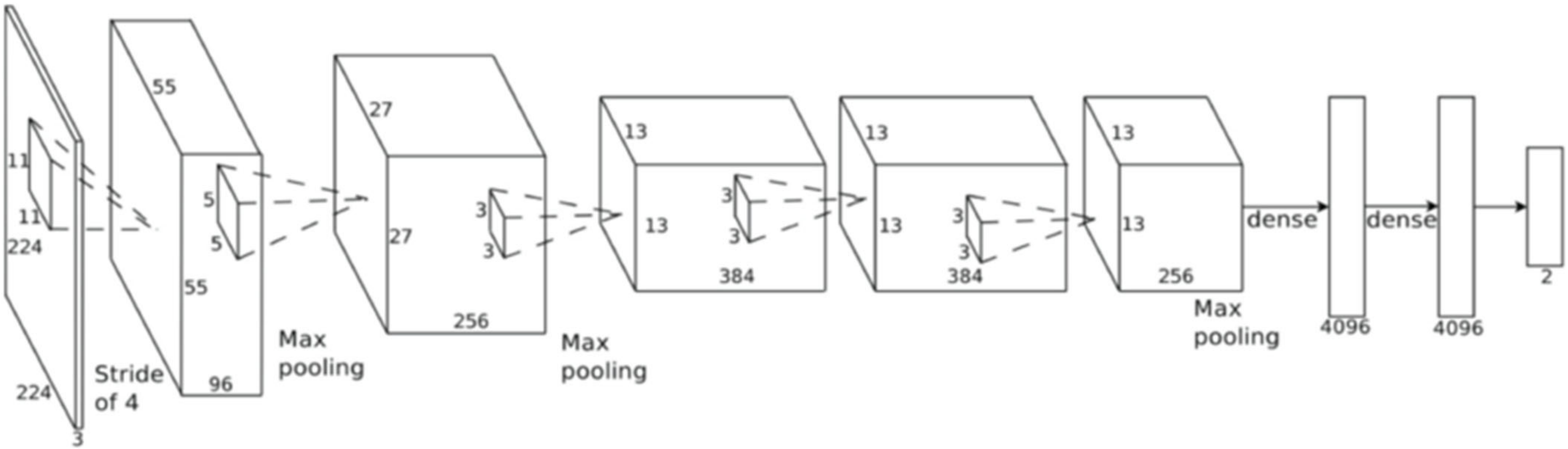


Fig. 2. Architecture of LeNet-5, a Convolutional Neural Network, here for digits recognition. Each plane is a feature map, i.e. a set of units whose weights are constrained to be identical.

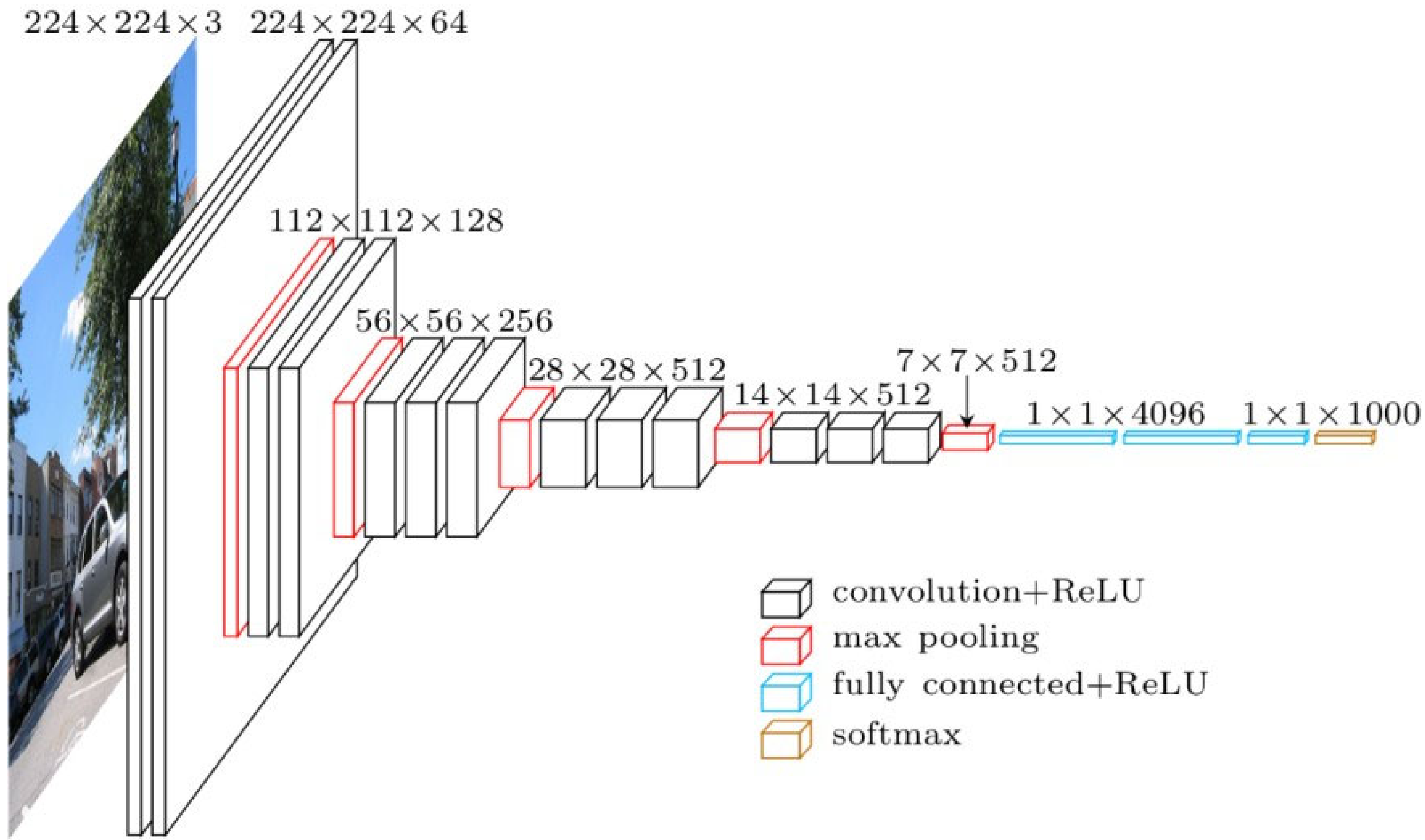
Architectures

AlexNet Krizhevsky et al. in 2012



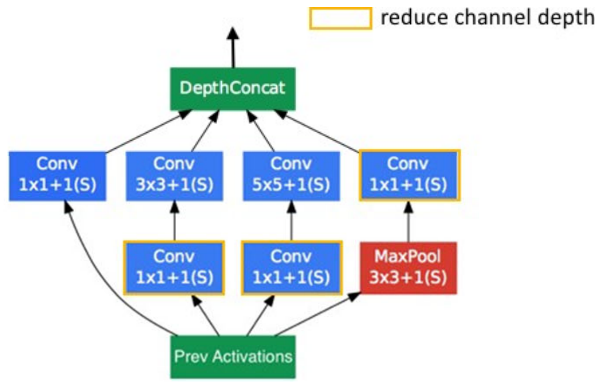
Architectures

VGG16 Simonyan and Zisserman 2014

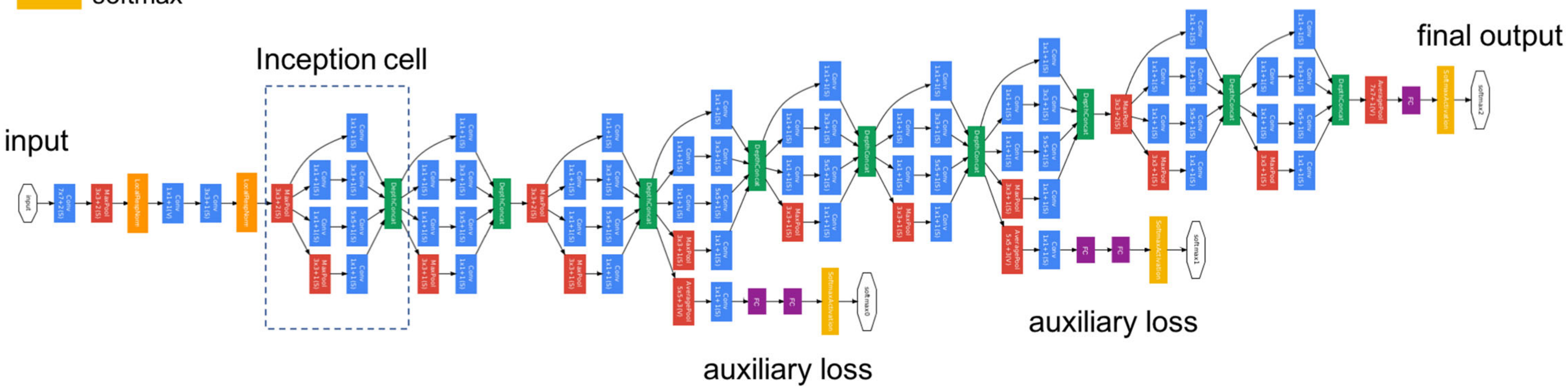


Architectures

Inception (Google) Szegedy, et al 2015



- convolution
- max pooling
- channel concatenation
- channel-wise normalization
- fully-connected layer
- softmax



Architectures

ResNet He, et al 2015

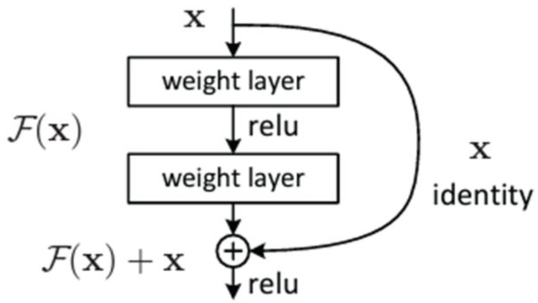
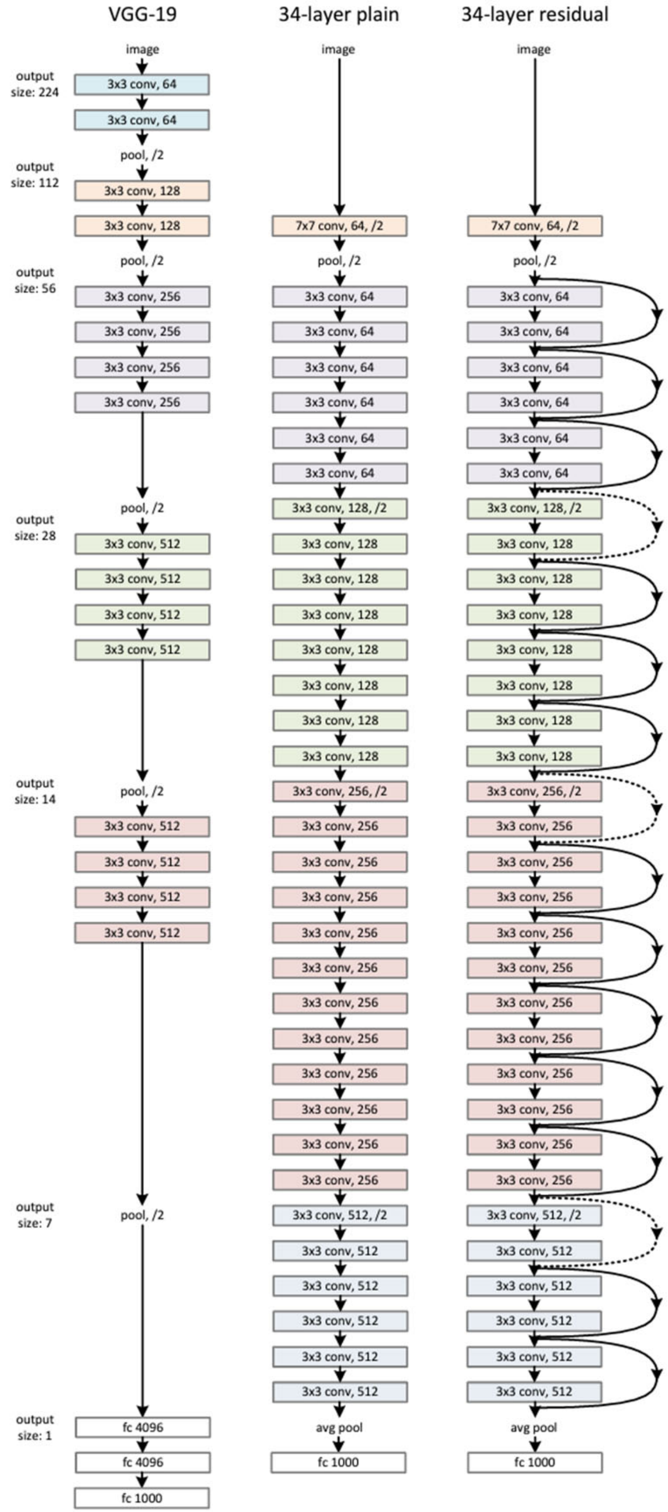


Figure 2. Residual learning: a building block.



Deep learning