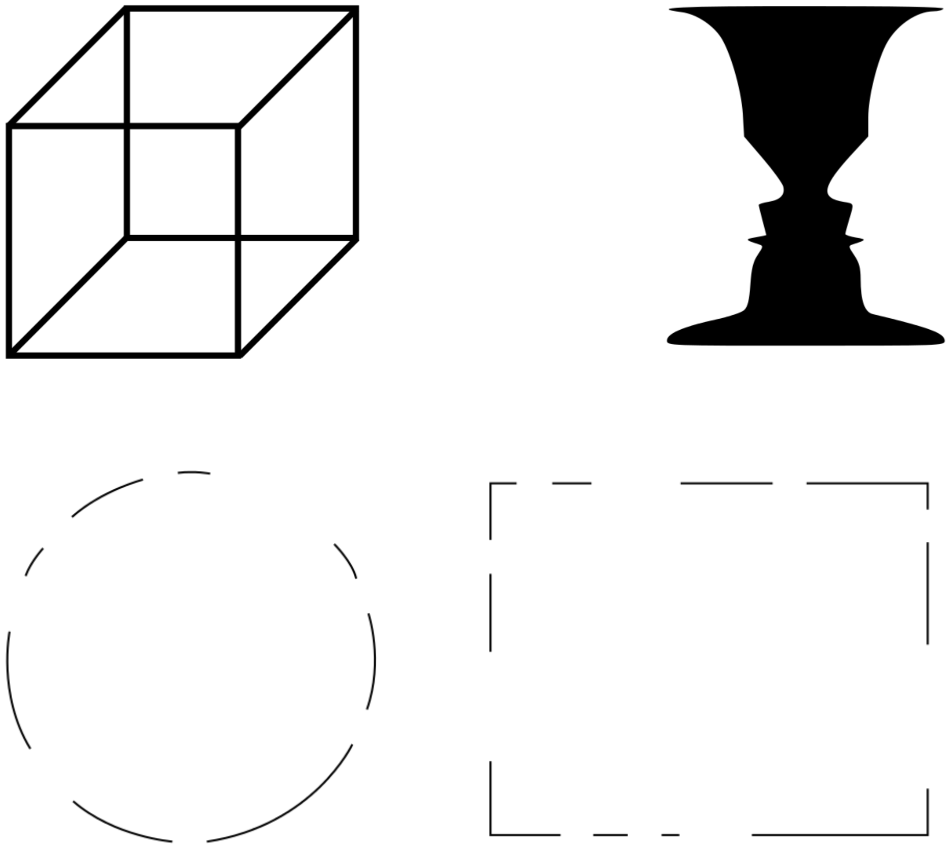
합성곱 신경망
Convolutional Neural Networks

오영민,
Ph.D.

Problem of Perception 지각의 문제

지각(perception)은 감각 정보에 대한 능동적인 해석, 혹은 번역이다.

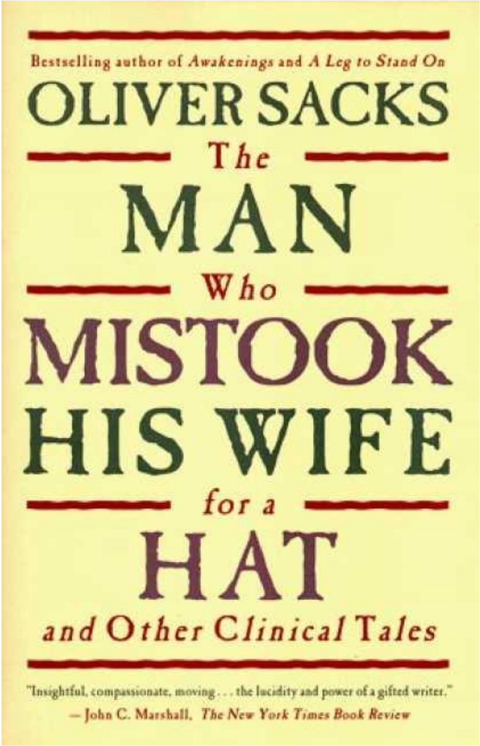


https://en.wikipedia.org/wiki/Perception

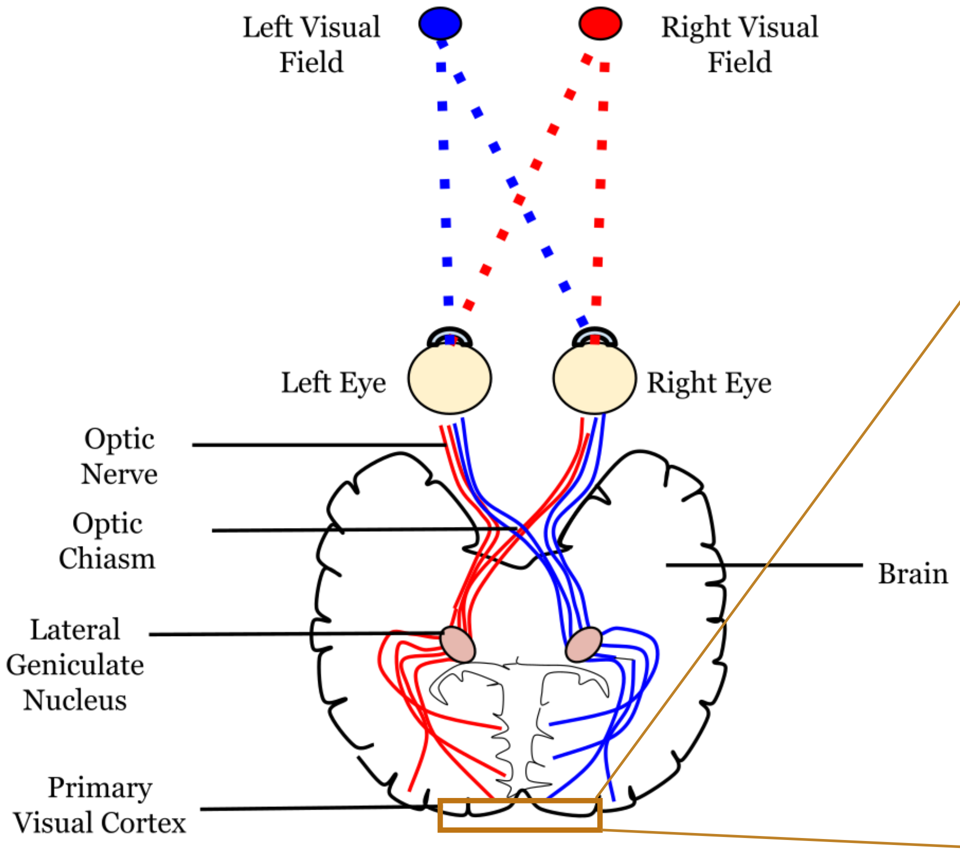
Visual Agnosia 시각실인증

시각 기능과 인지 능력이 온전함에도 불구하고, 특정 사물을 인식하지 못하는 증상.
시각적 인식(지각)이 sub-conscious level에서 이루어지는 고도의 기능임을 알 수 있다.
예) 안면실인증

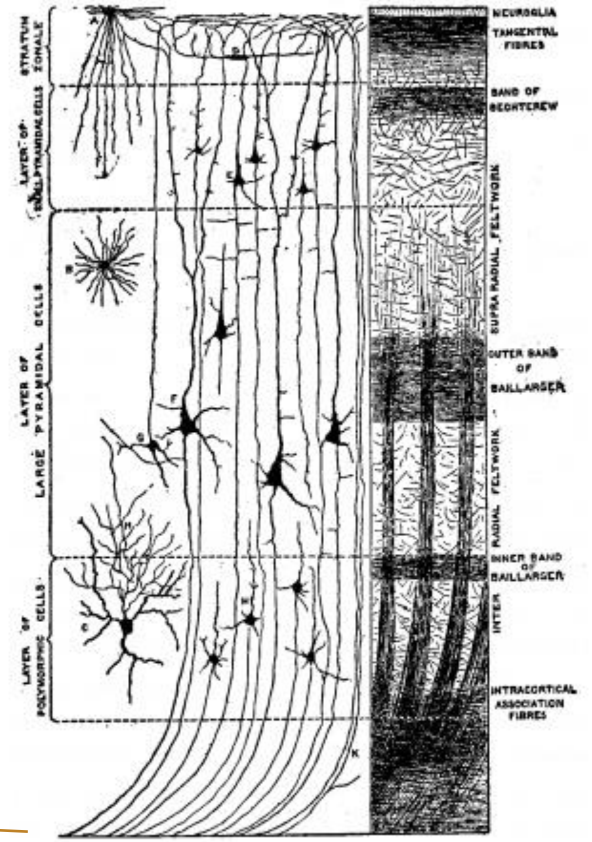
https://mozanunal.com/2019/11/img2sh/



Visual Recognition of Brain 뇌의 시각 정보 인식



https://upload.wikimedia.org/wikipedia/commons/4/4d/Neural_pathway_diagram.svg



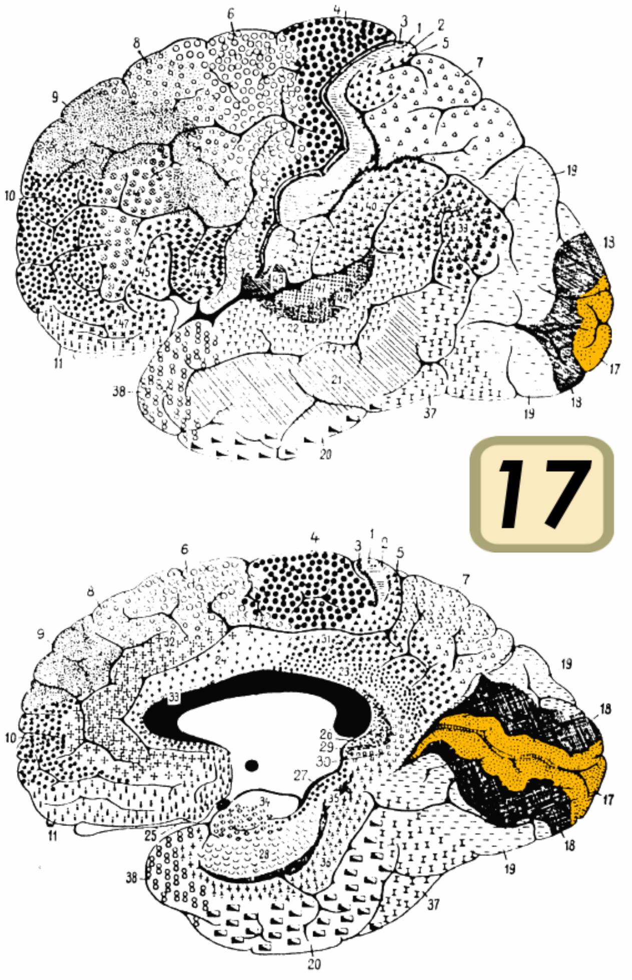
https://en.wikipedia.org/wiki/Cerebral_cortex

Layers of Cerebral Cortex 대뇌피질의 층 구조

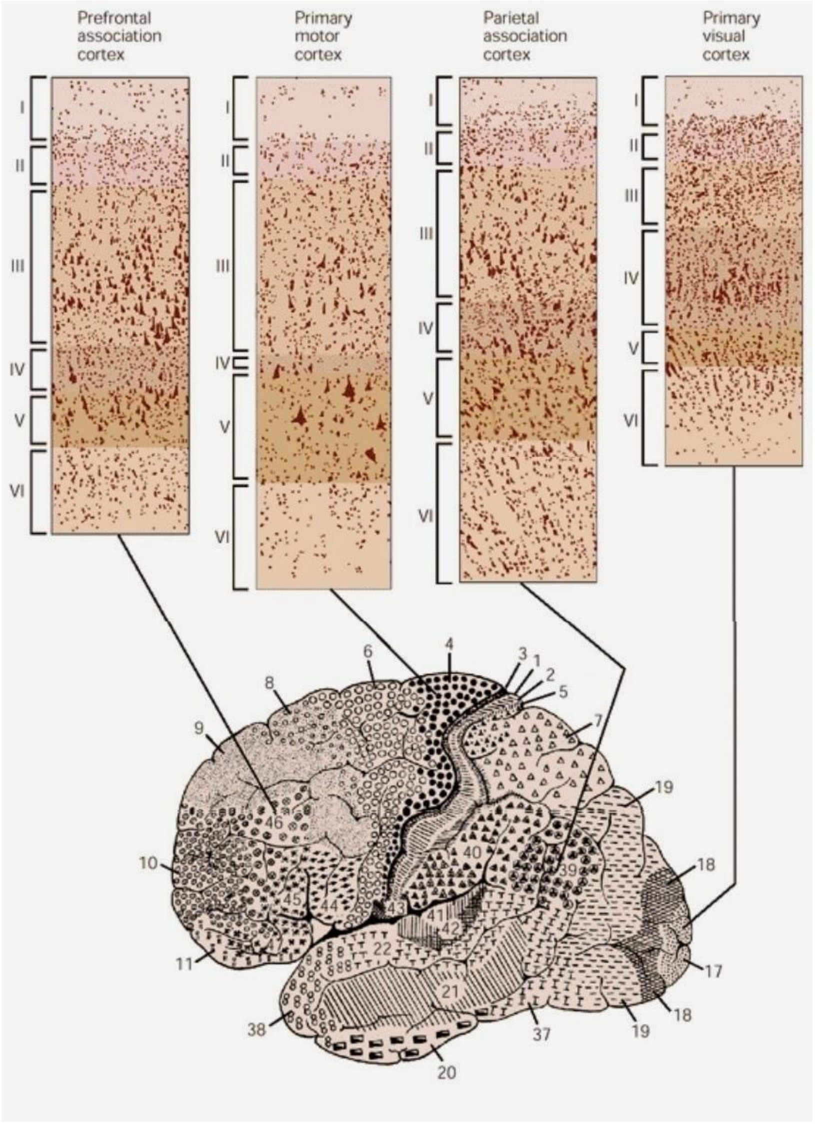
인지, 추론, 운동, 학습 등을 담당하는 대뇌 피
질은
구분되는 여러 개의 신경세포 층들로 이루어져
있다.

대뇌피질의 각 영역과, 서로 다른 층은 각기 다
른
뇌 기능을 담당하고 있고, 이들은 서로 연결되
어 있다.

시각 정보는 Primary Visual Cortex (V1)에서
처리한다.



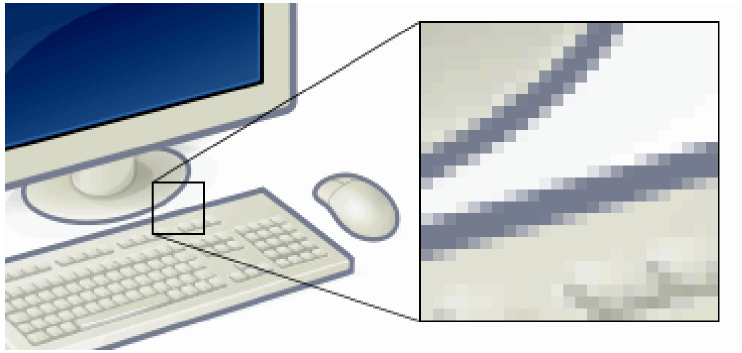
https://commons.wikimedia.org/wiki/File:Brodmann_Cytoarchitectonics_17.png



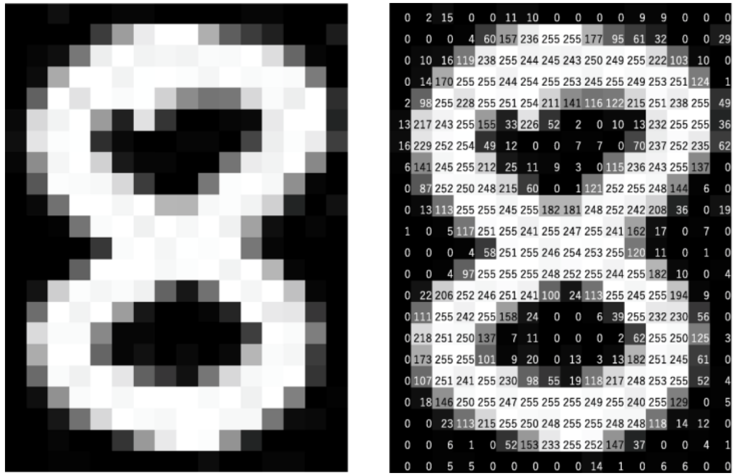
http://theanalogicalinstinct.blogspot.kr/

Computer Vision 컴퓨터 비전

픽셀(pixel)로 이루어진 이미지(동영상)로부터 물체의 종류, 위치, 정보, 관계 등 추상적인 정보를 추출하는 작업



https://en.wikipedia.org/wiki/Pixel

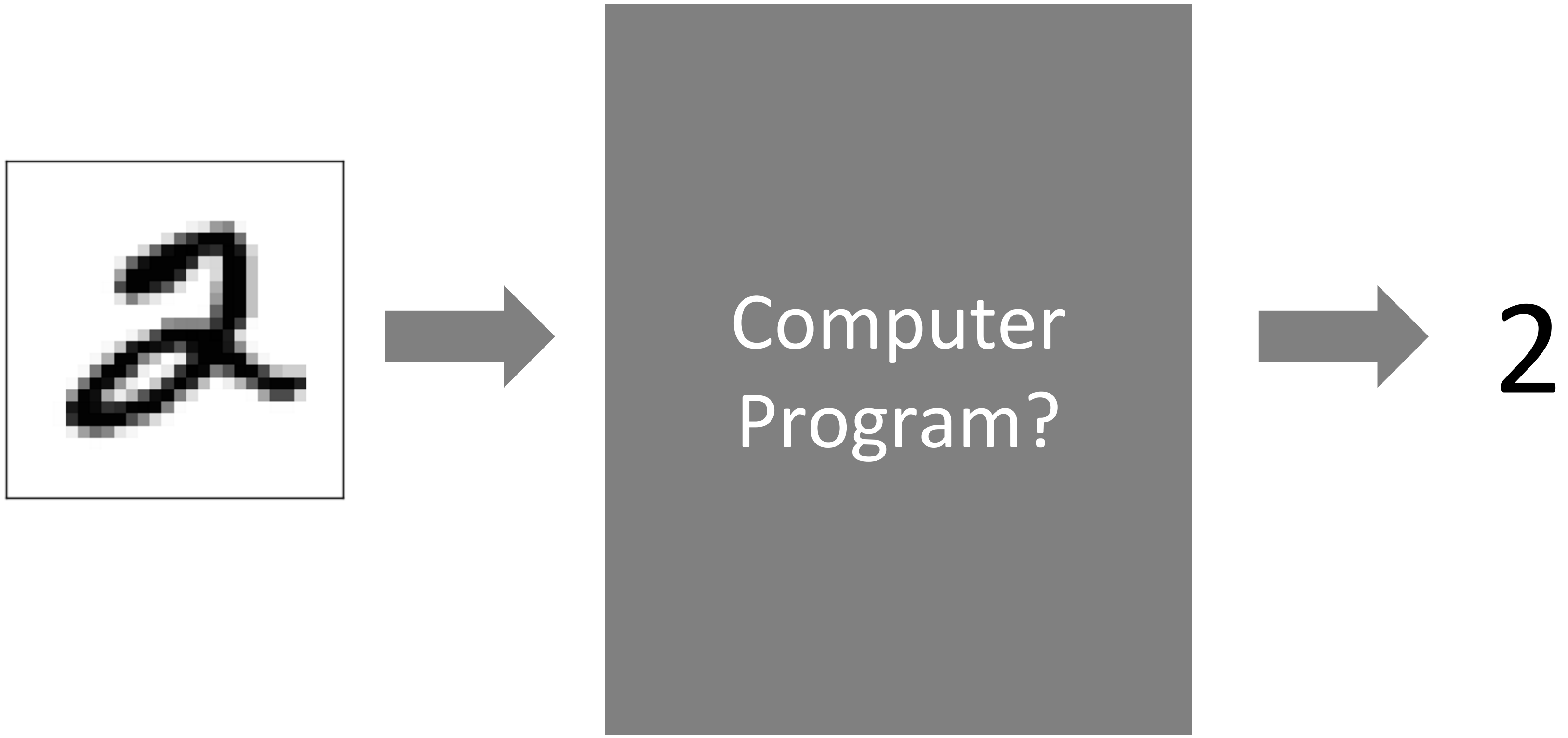


0	2	15	0	0	11	10	0	0	0	0	9	9	0	0	0
0	0	0	4	60	157	236	255	255	177	95	61	32	0	0	29
0	10	16	119	238	255	244	245	243	250	249	255	222	103	10	0
0	14	170	255	255	244	254	255	253	245	255	249	253	251	124	1
2	98	255	228	255	251	254	211	141	116	122	215	251	238	255	49
13	217	243	255	155	33	226	52	2	0	10	13	232	255	255	36
16	229	252	254	49	12	0	0	7	7	0	70	237	252	235	62
6	141	245	255	212	25	11	9	3	0	115	236	243	255	137	0
0	87	252	250	248	215	60	0	1	121	252	255	248	144	6	0
0	13	113	255	255	245	255	182	181	248	252	242	208	36	0	19
1	0	5	117	251	255	241	255	247	255	241	162	17	0	7	0
0	0	0	4	58	251	255	246	254	253	255	120	11	0	1	0
0	0	4	97	255	255	255	248	252	255	244	255	182	10	0	4
0	22	206	252	246	251	241	100	24	113	255	245	255	194	9	0
0	111	255	242	255	158	24	0	0	6	39	255	232	230	56	0
0	218	251	250	137	7	11	0	0	0	2	62	255	250	125	3
0	173	255	255	101	9	20	0	13	3	13	182	251	245	61	0
0	107	251	241	255	230	98	55	19	118	217	248	253	255	52	4
0	18	146	250	255	247	255	255	255	249	255	240	255	129	0	5
0	0	23	113	215	255	250	248	255	255	248	248	118	14	12	0
0	0	6	1	0	52	153	233	255	252	147	37	0	0	4	1
0	0	5	5	0	0	0	0	0	14	1	0	6	6	0	0

https://mozanunal.com/2019/11/img2sh/

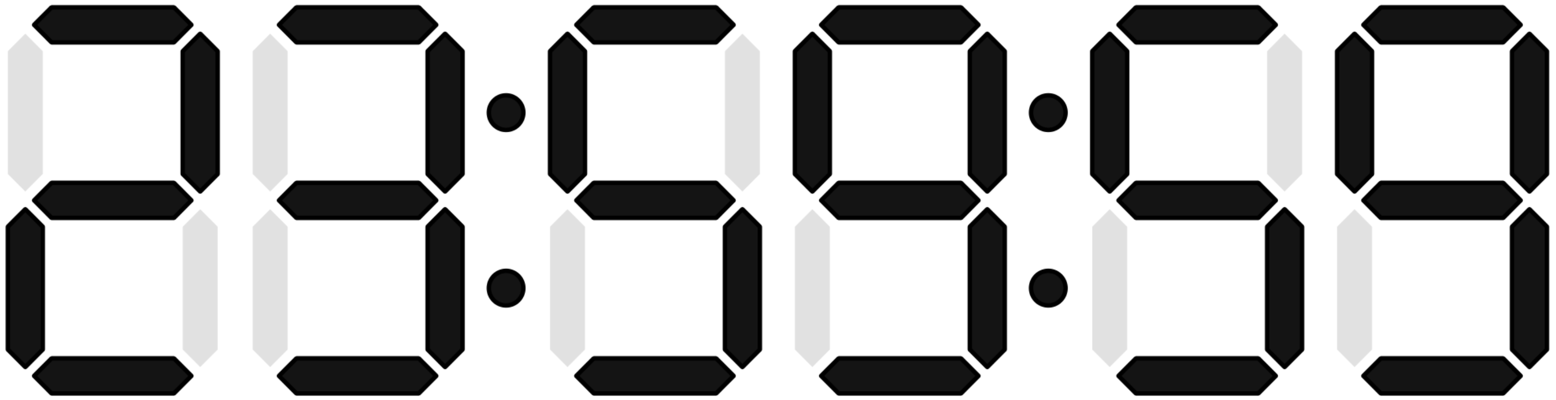
Visual Recognition. 시각 인지(인식)

이미지(숫자들의 배열)이 나타내는 대상의 종류를 파악하는 지능
From an array of numbers to a category



Visual Recognition. 시각 인지(인식)

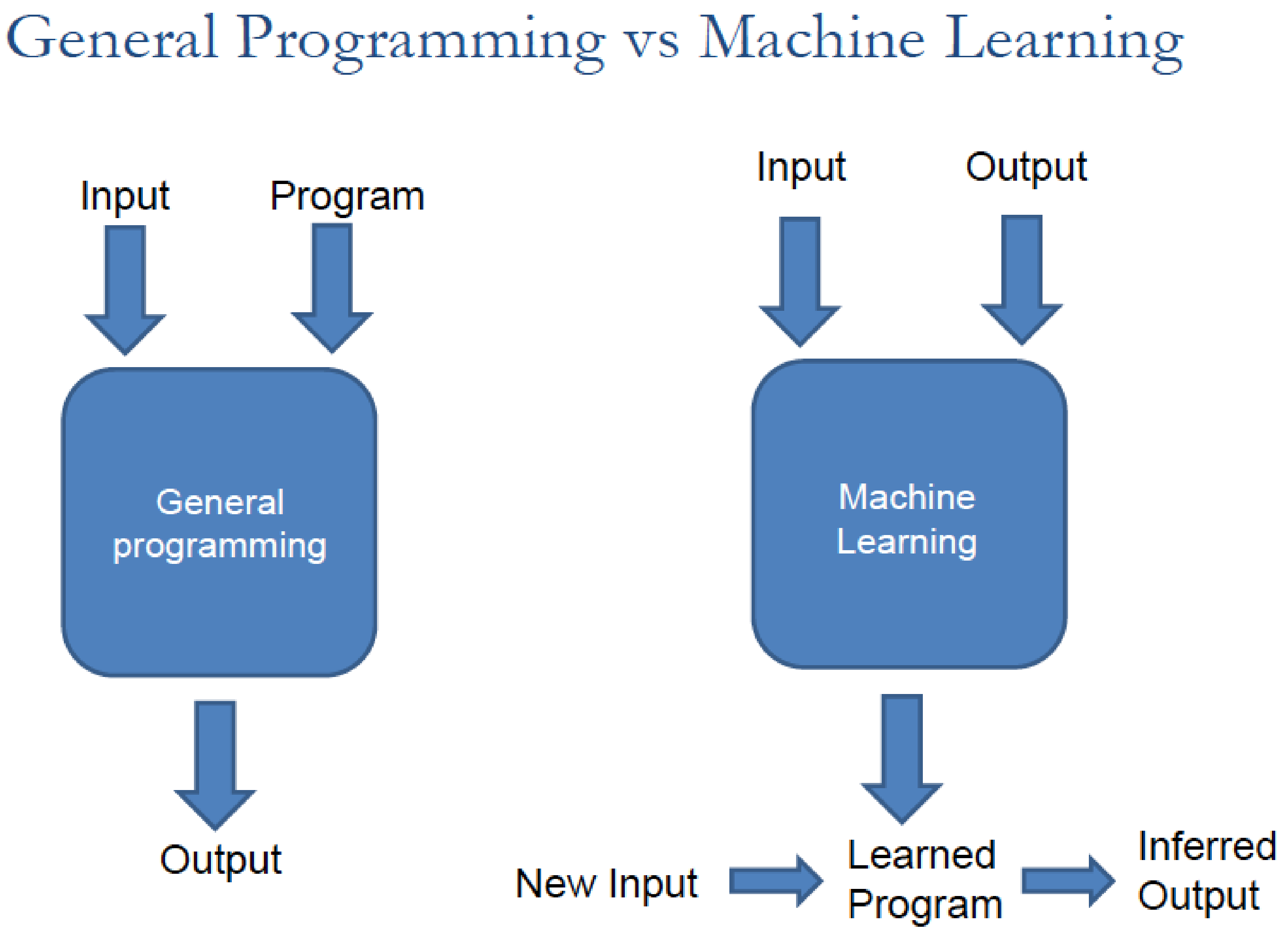
이미지(숫자들의 배열)이 나타내는 대상의 종류를 파악하는 지능
From an array of numbers to a category



https://www.pngkey.com/maxpic/u2q8t4t4u2e6w7e6/

MNIST: Dataset for Handwritten Digit Recognition

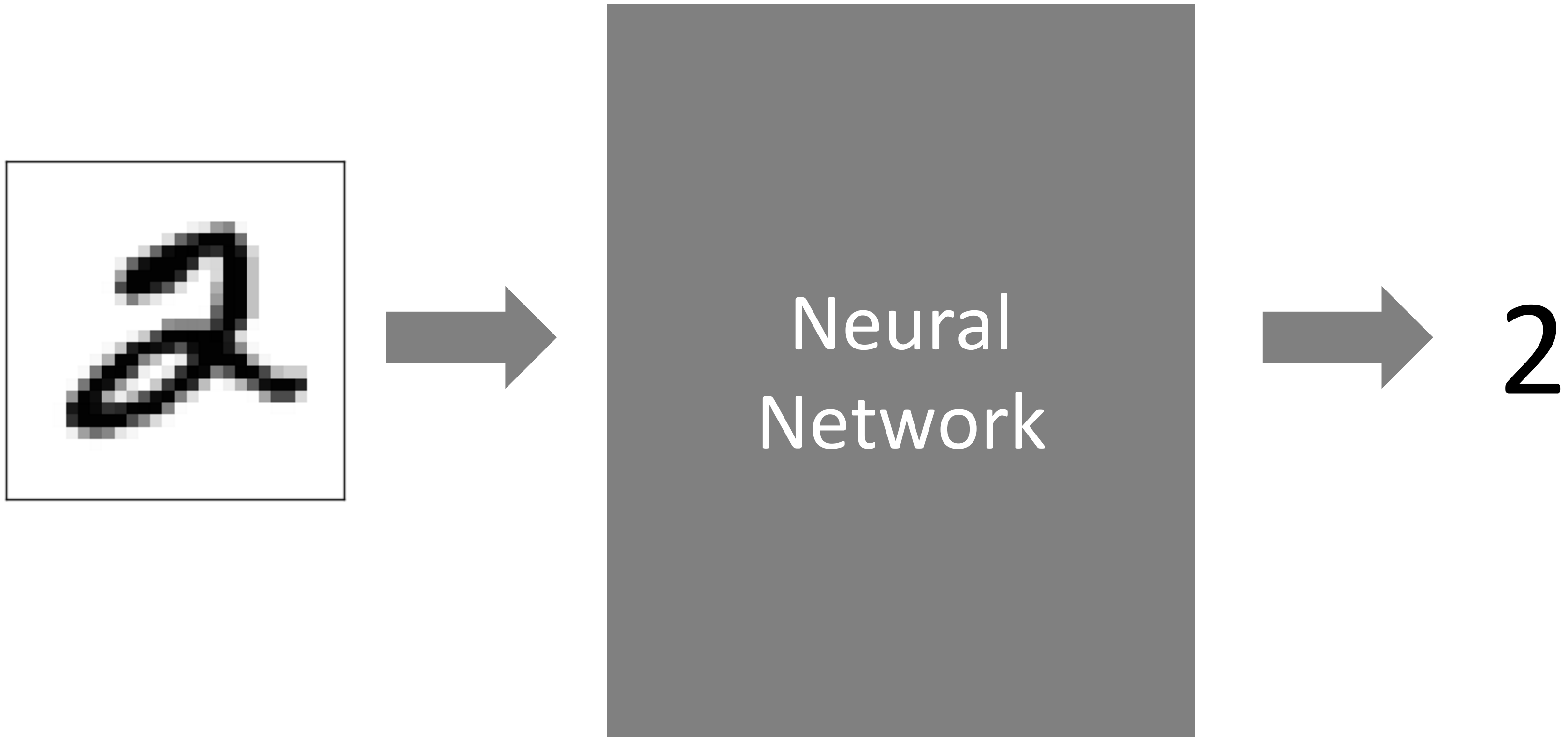
Machine Learning 기계학습



Ibishal, Eindhoven Developers Meetup, 2015

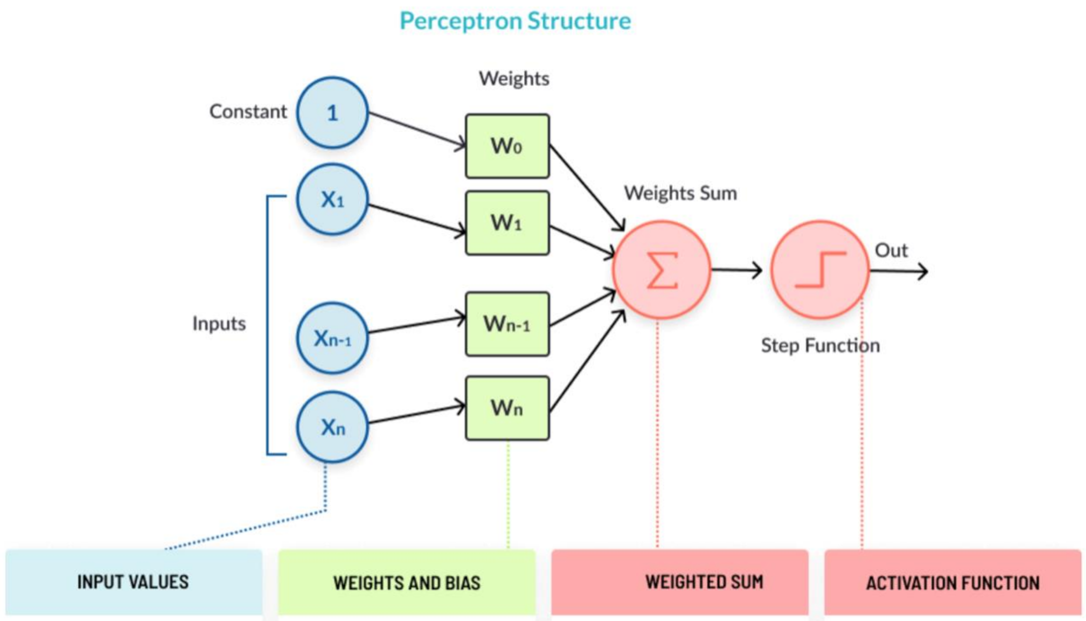
Visual Recognition. 시각 인지(인식)

이미지(숫자들의 배열)이 나타내는 대상의 종류를 파악하는 지능
From an array of numbers to a category

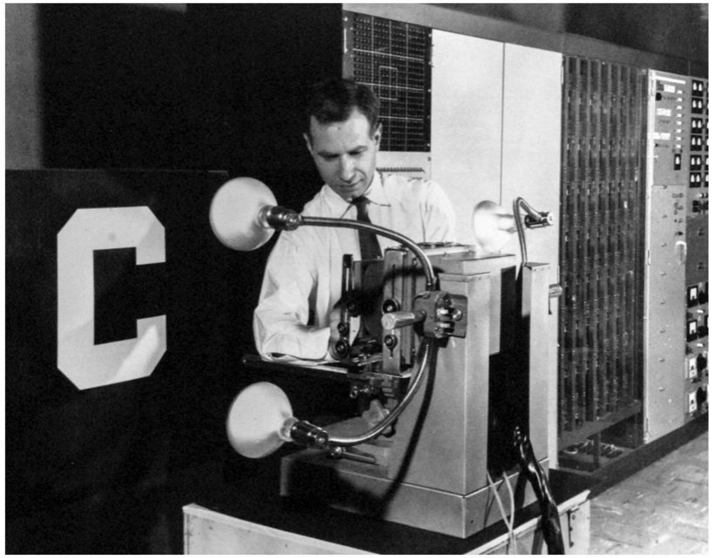


Mark I Perceptron

1st Generation: Binary Neurons

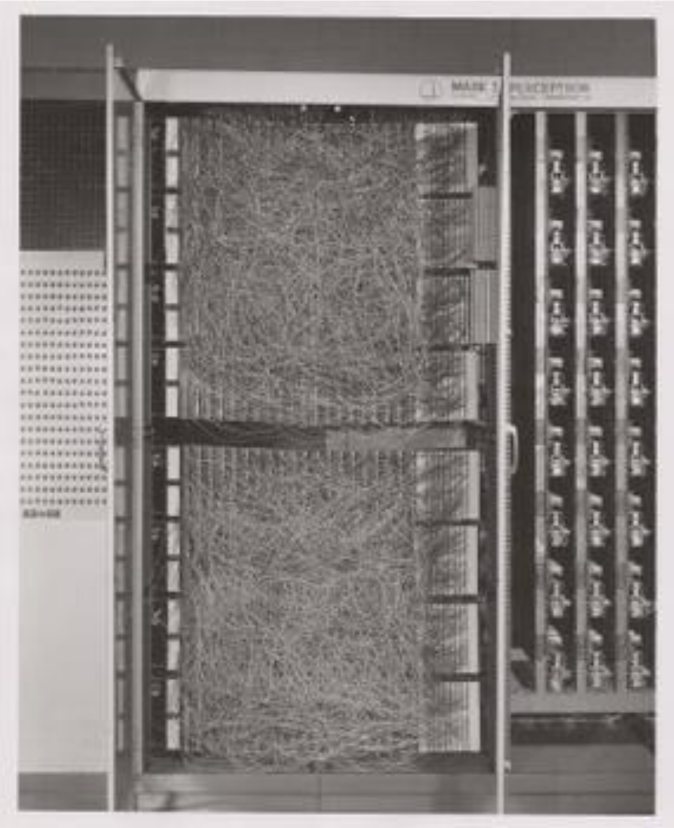


https://missinglink.ai/guides/neural-network-concepts/

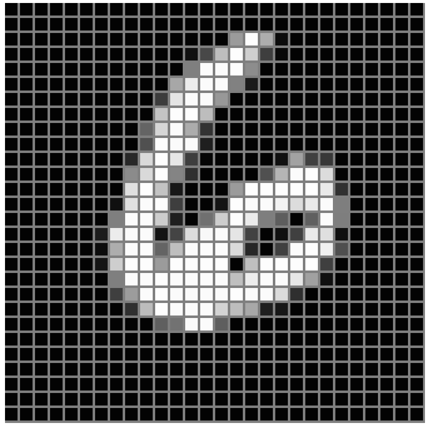


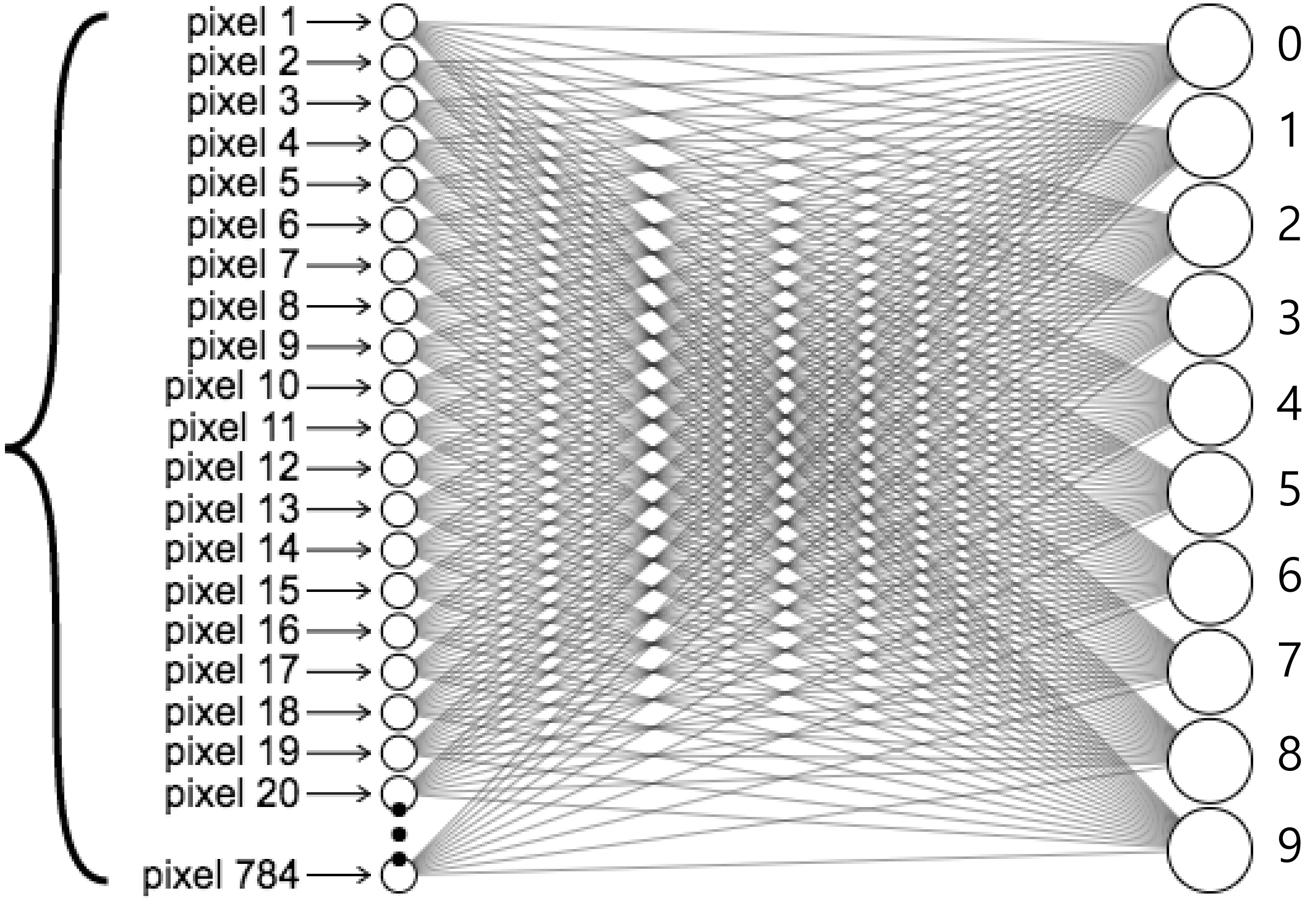
https://news.cornell.edu/stories/2019/09/professors-perceptron-paved-way-ai-60-years-too-soon

Mark I Perceptron

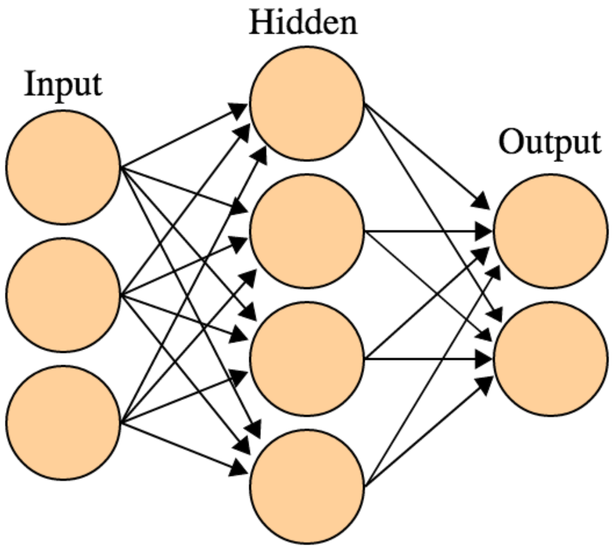


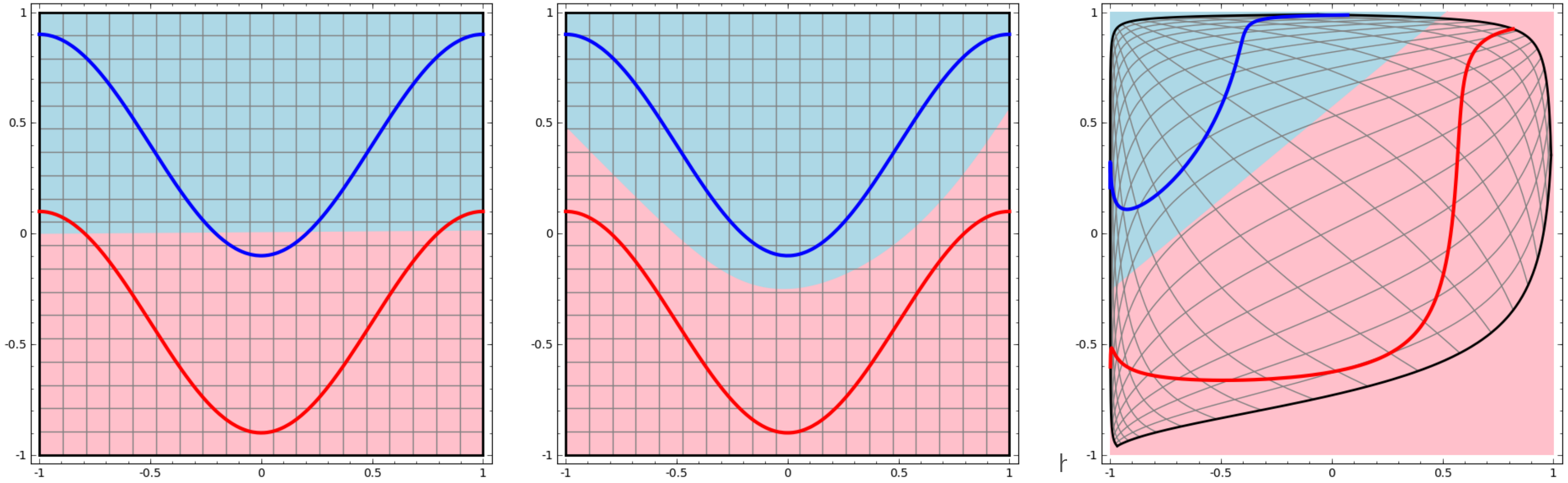
https://digital.library.cornell.edu/catalog/ss:550351





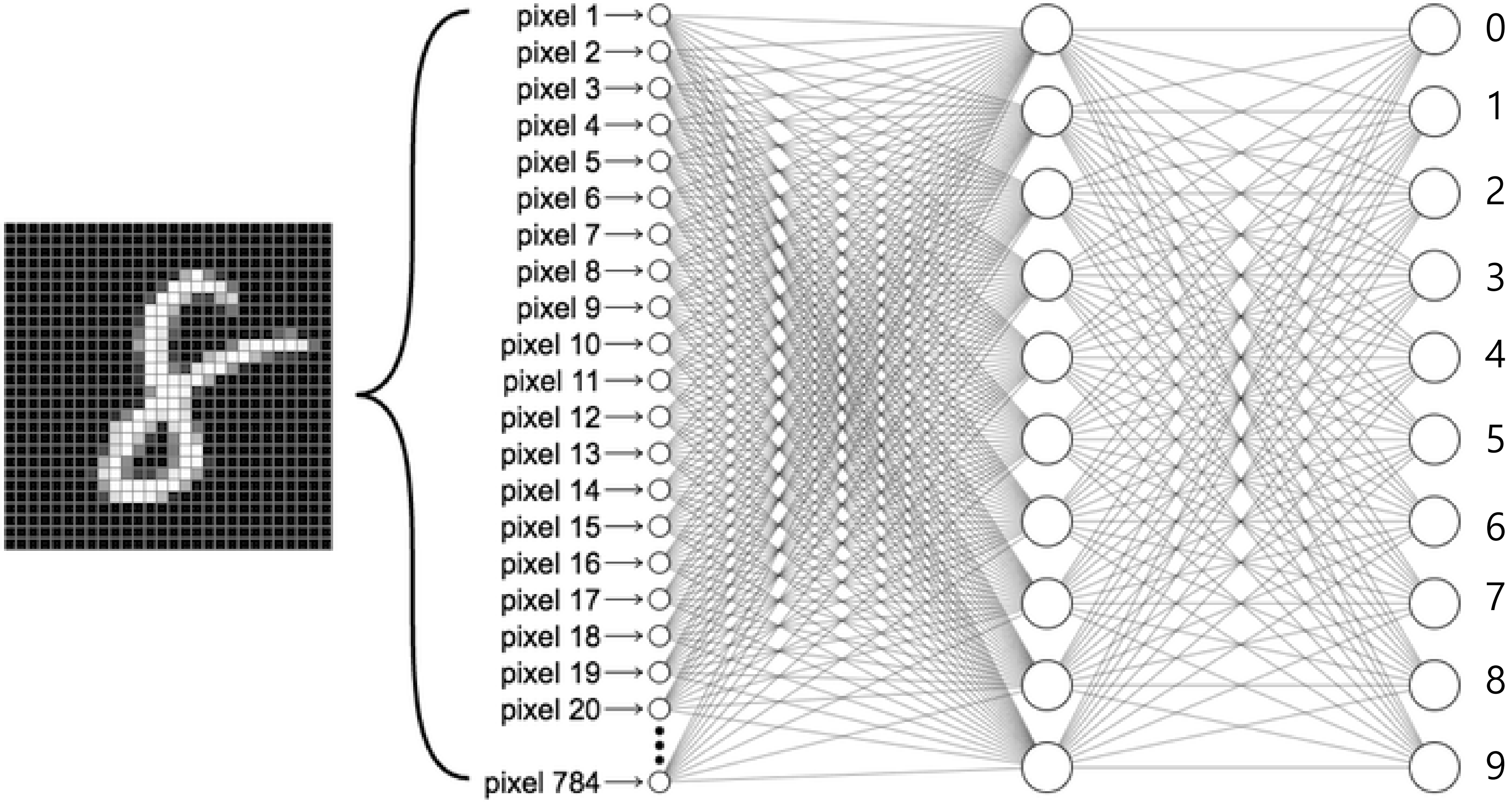
Multilayer Perceptron





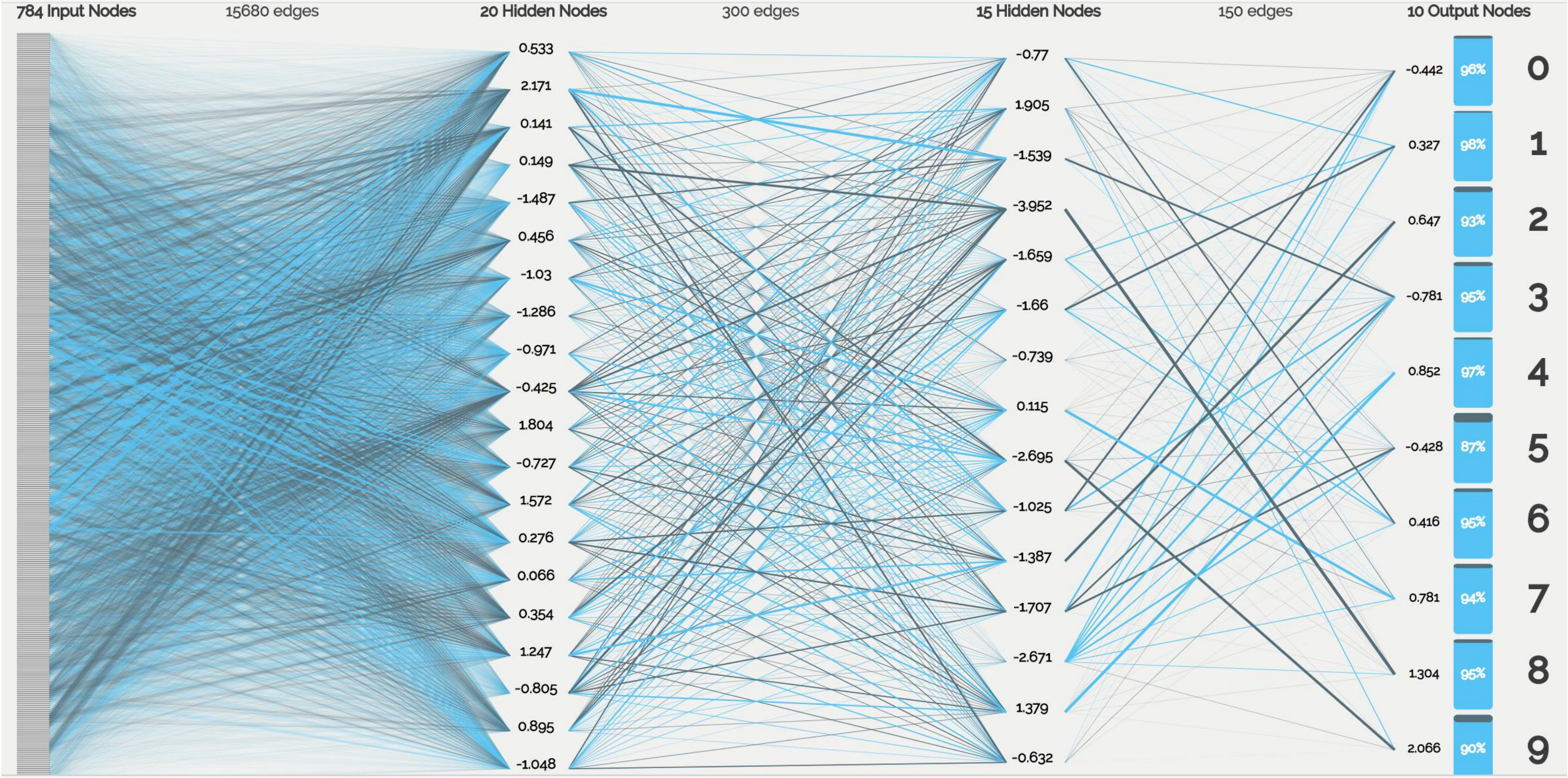
colah.github.io

Multilayer Perceptron or Fully-connected Neural Network



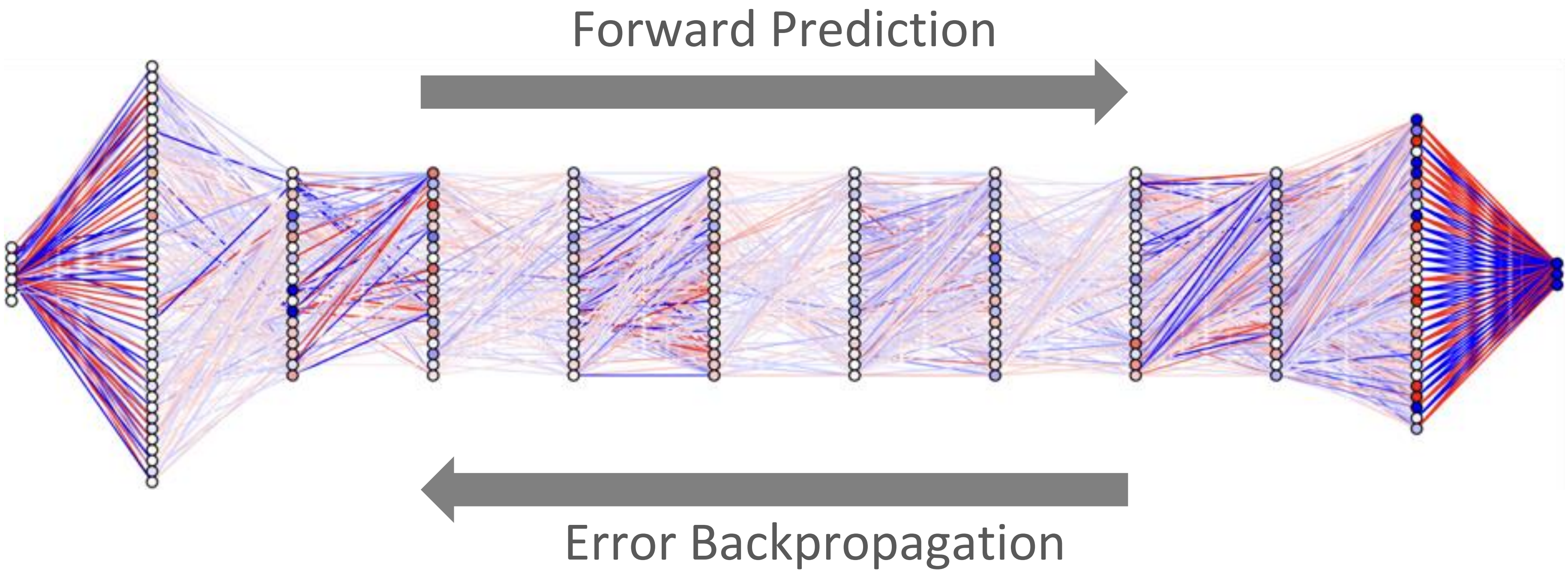
https://ml4a.github.io/ml4a/looking_inside_neural_nets/

Multilayer Perceptron or Fully-connected Neural Network



http://nn-mnist.sennabaum.com/

Deep Neural Networks with Backpropagation



https://www.mghpcc.org/neural-networks-earthquakes/

Moravec’s Paradox

Moravec Paradox & The nature of intelligence

“The main lesson of thirty-five years of AI research is that. **the hard problems are easy and the easy problems are hard**”

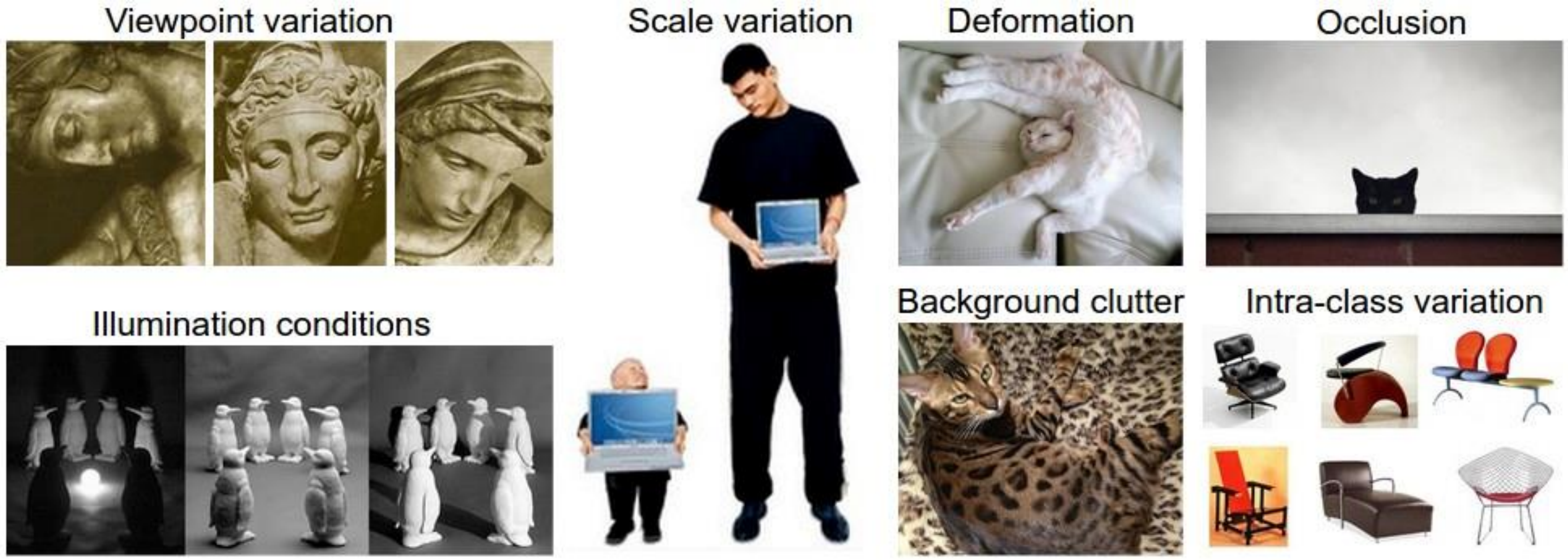
Linguist and cognitive scientist *Steven Pinker*

Kasparov vs. machine (2003)
Fake victory of AI (brut force computation)

“It is easy to make computers exhibit adult level performance on intelligence tests or playing checkers, and **difficult or impossible to give them the skills of a one-year-old** when it comes to perception and mobility” - 1988

Moravec

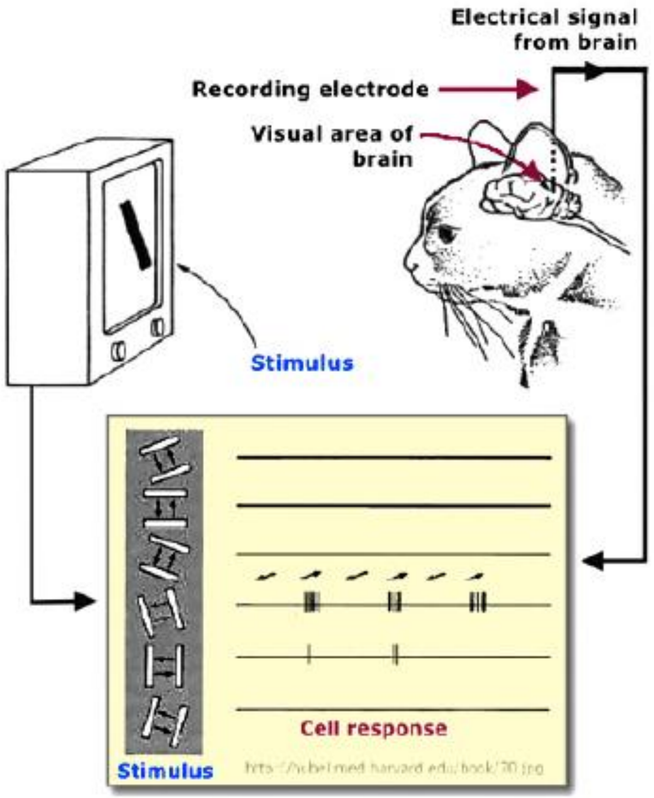
Challenges of Visual Recognition

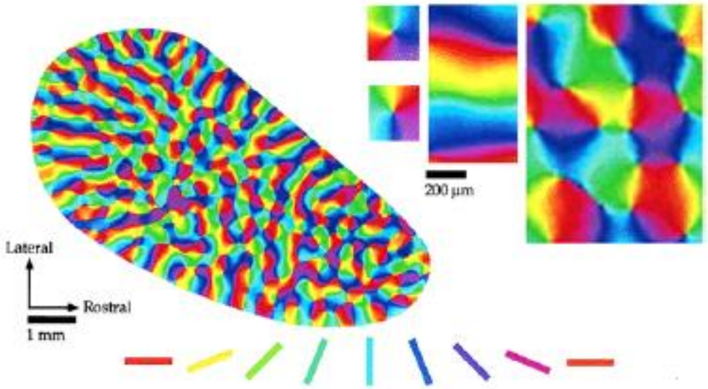


Hint from Neuroscience: The Hubel and Wiesel Experiment

https://www.youtube.com/watch?v=IOHayh06LJ4

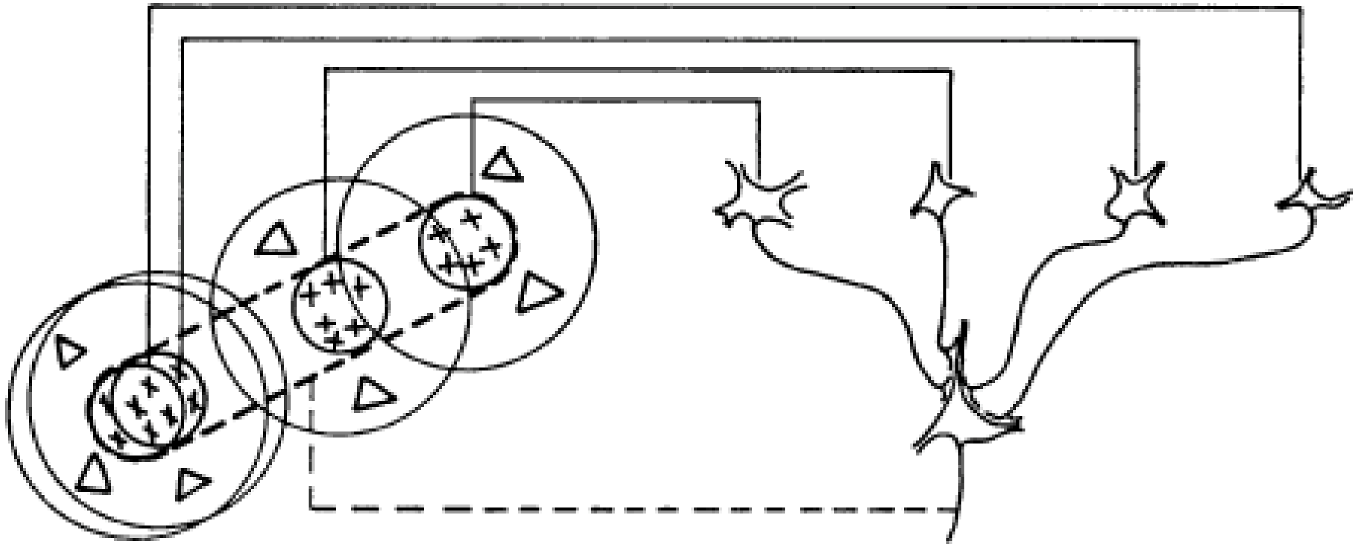
Neurons responding to a certain visual pattern





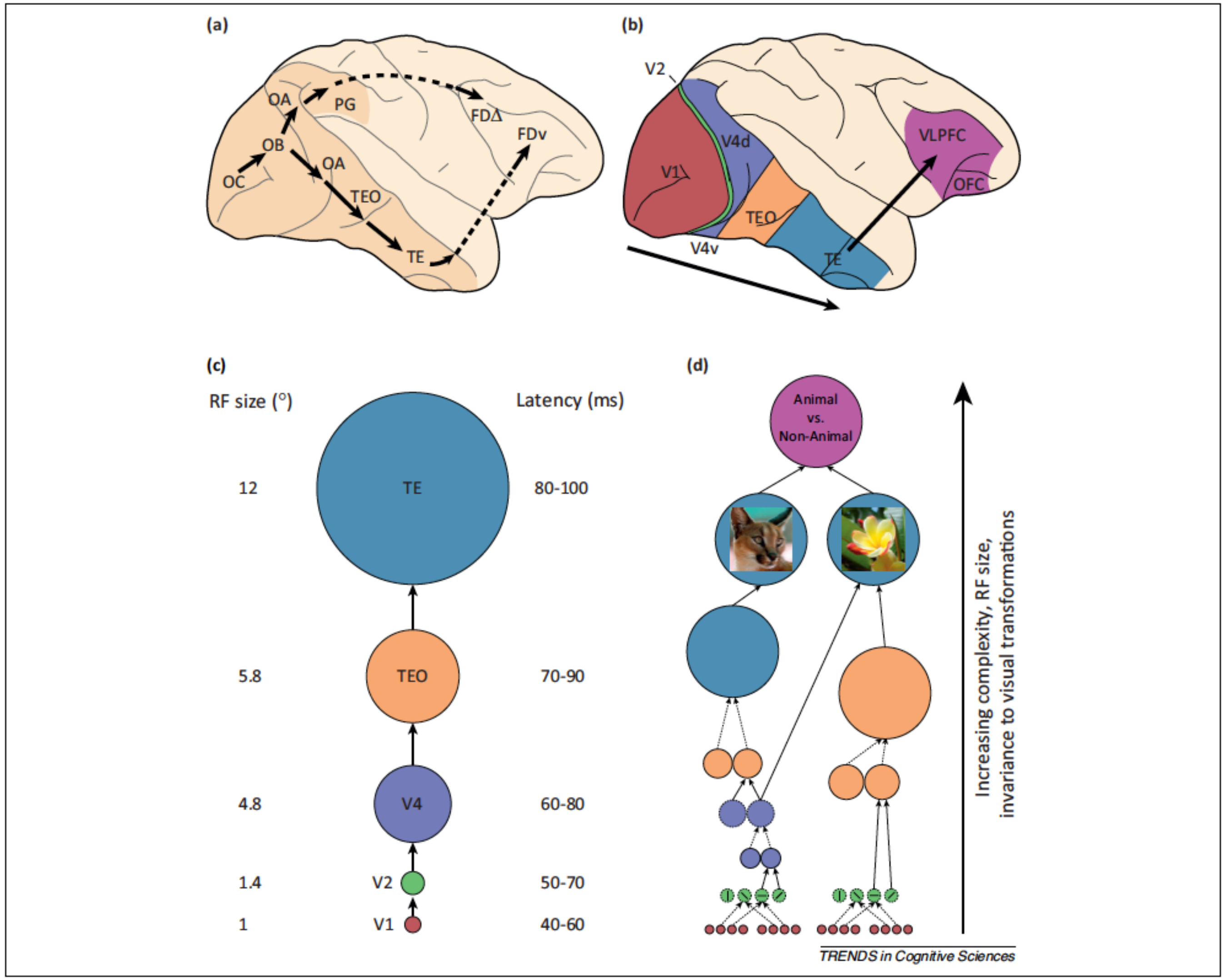
Oxford Machine Learning Course

Receptive Fields



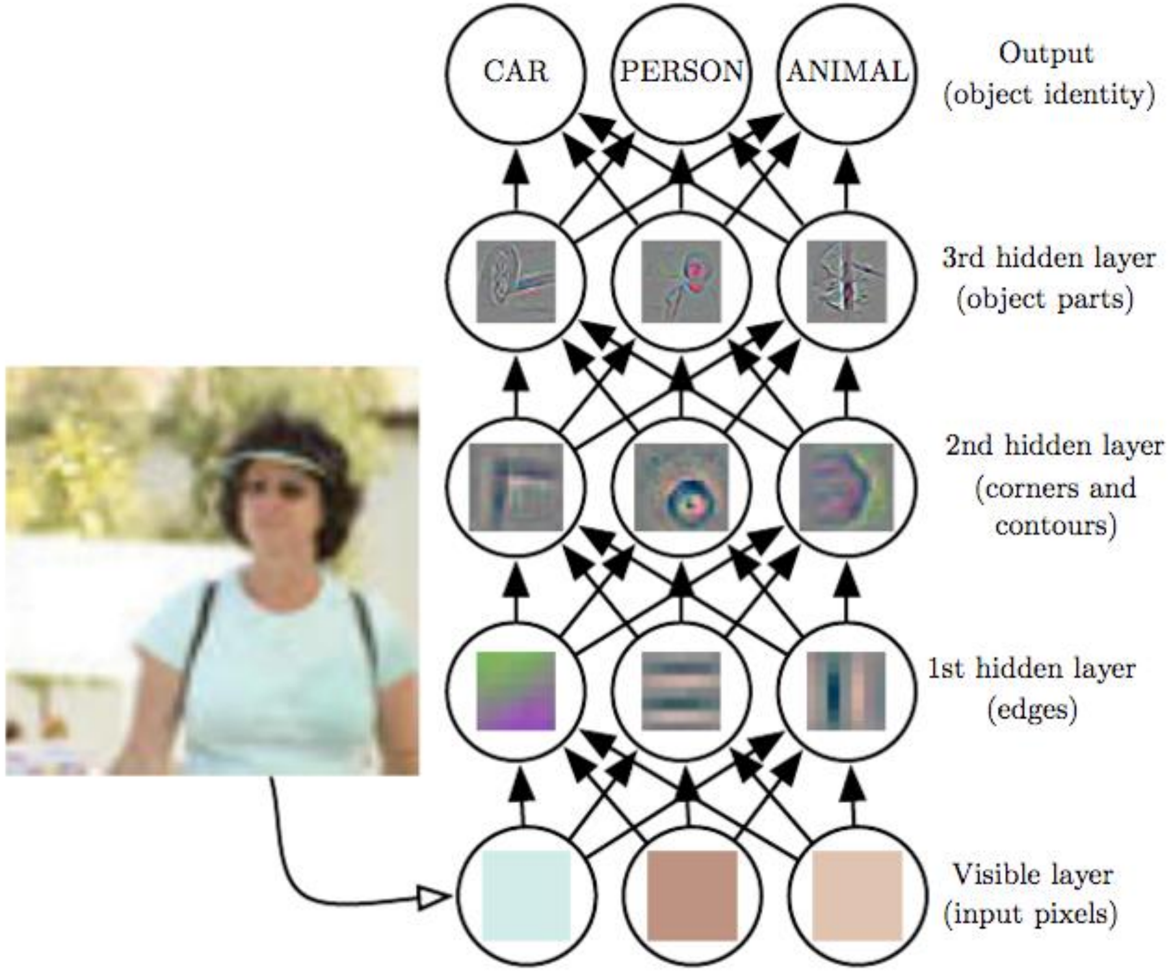
Hubel and Wiesel, *RECEPTIVE FIELDS, BINOCULAR INTERACTION AND FUNCTIONAL ARCHITECTURE IN THE CAT'S VISUAL CORTEX*, Journal of Physiology (1962)

Hierarchical Organization of Visual Cortex



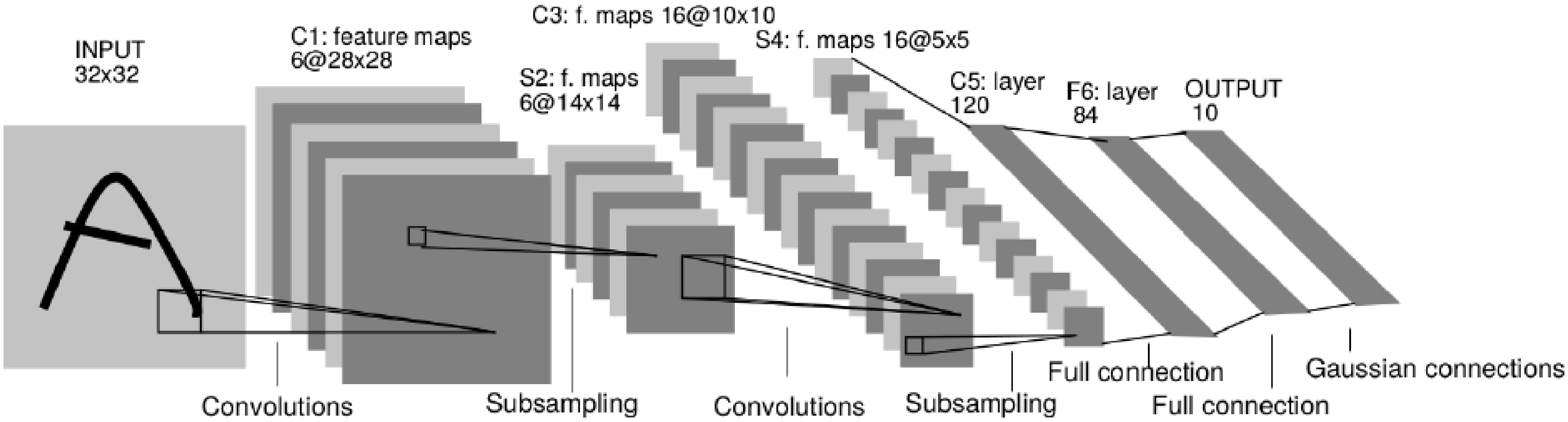
Kravitz et al., "The ventral visual pathway: an expanded neural framework for the processing of object quality", *Trends in Cognitive Sciences*, 2013

Representation Learning for Visual Recognition



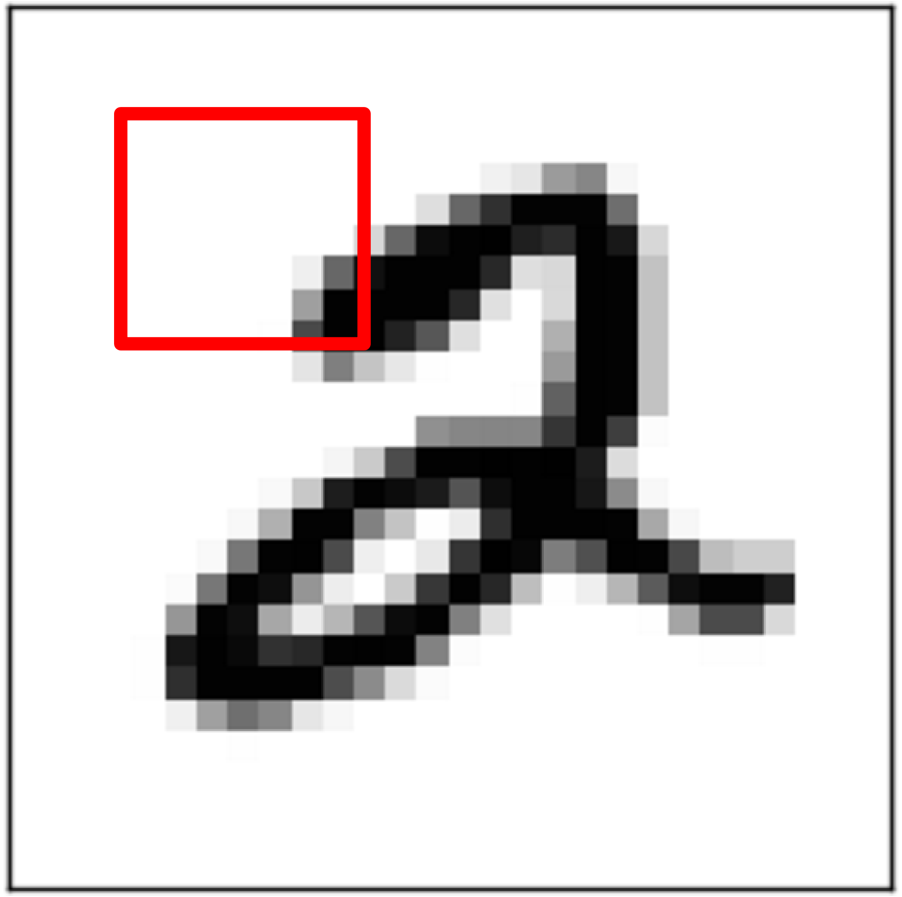
Goodfellow et al., 2015

Convolutional Neural Network: LeNet-5

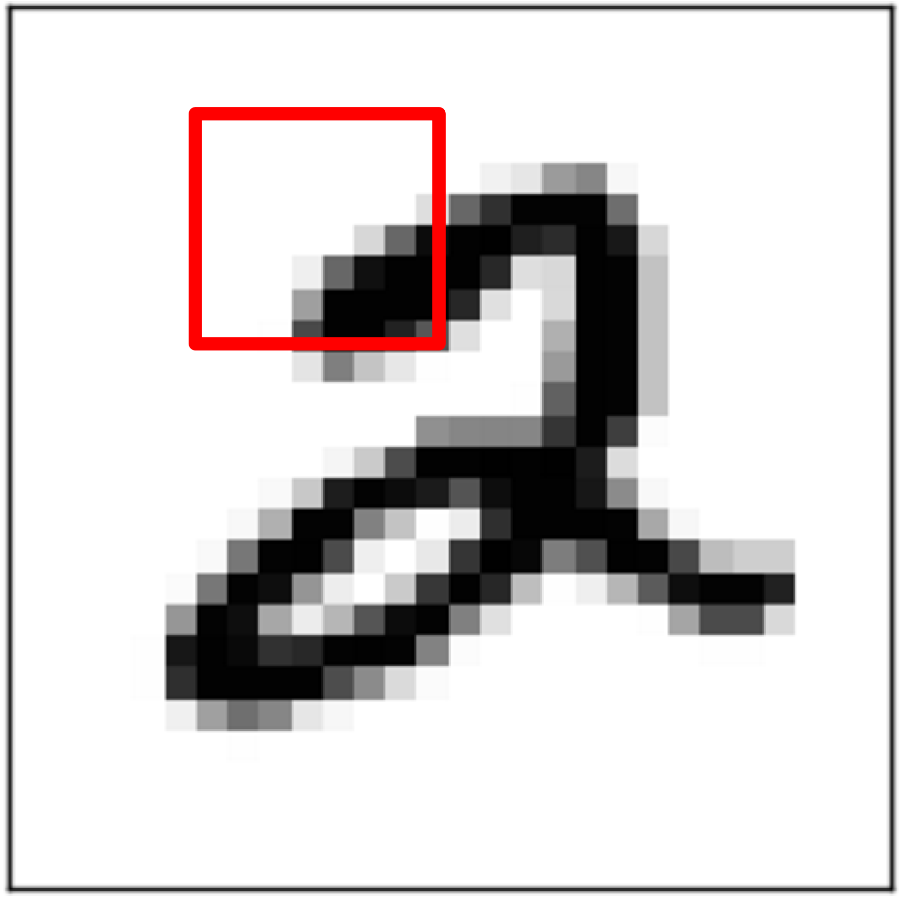


LeCun et al., Gradient-based learning applied to document recognition. Proceedings of the IEEE 86, 11 (1998)

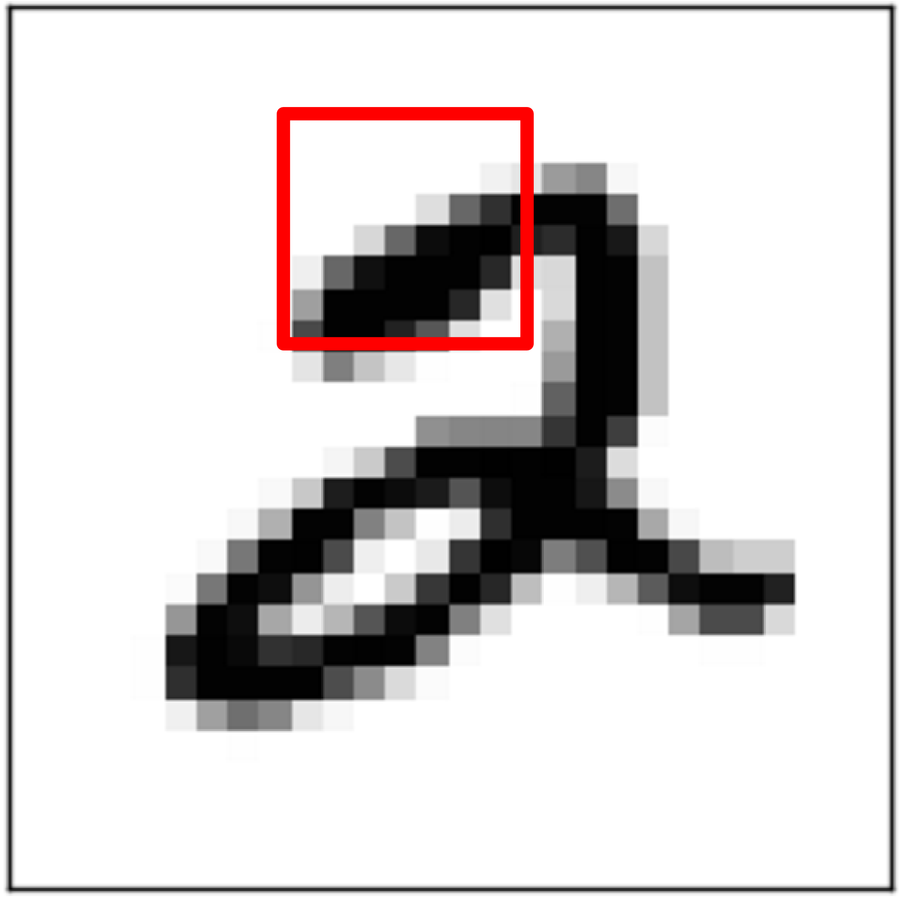
Convolution Operation (2D)



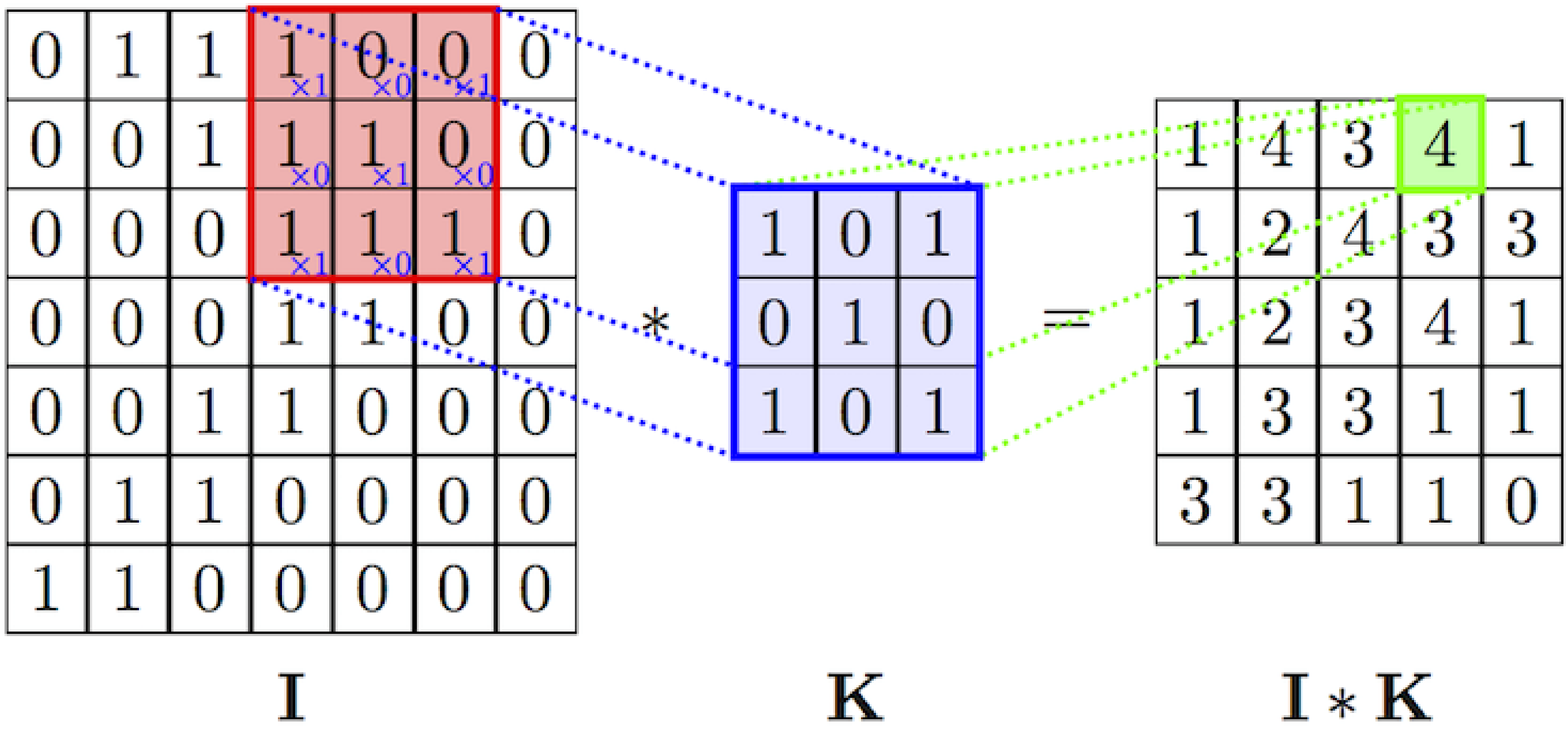
Convolution Operation (2D)



Convolution Operation (2D)

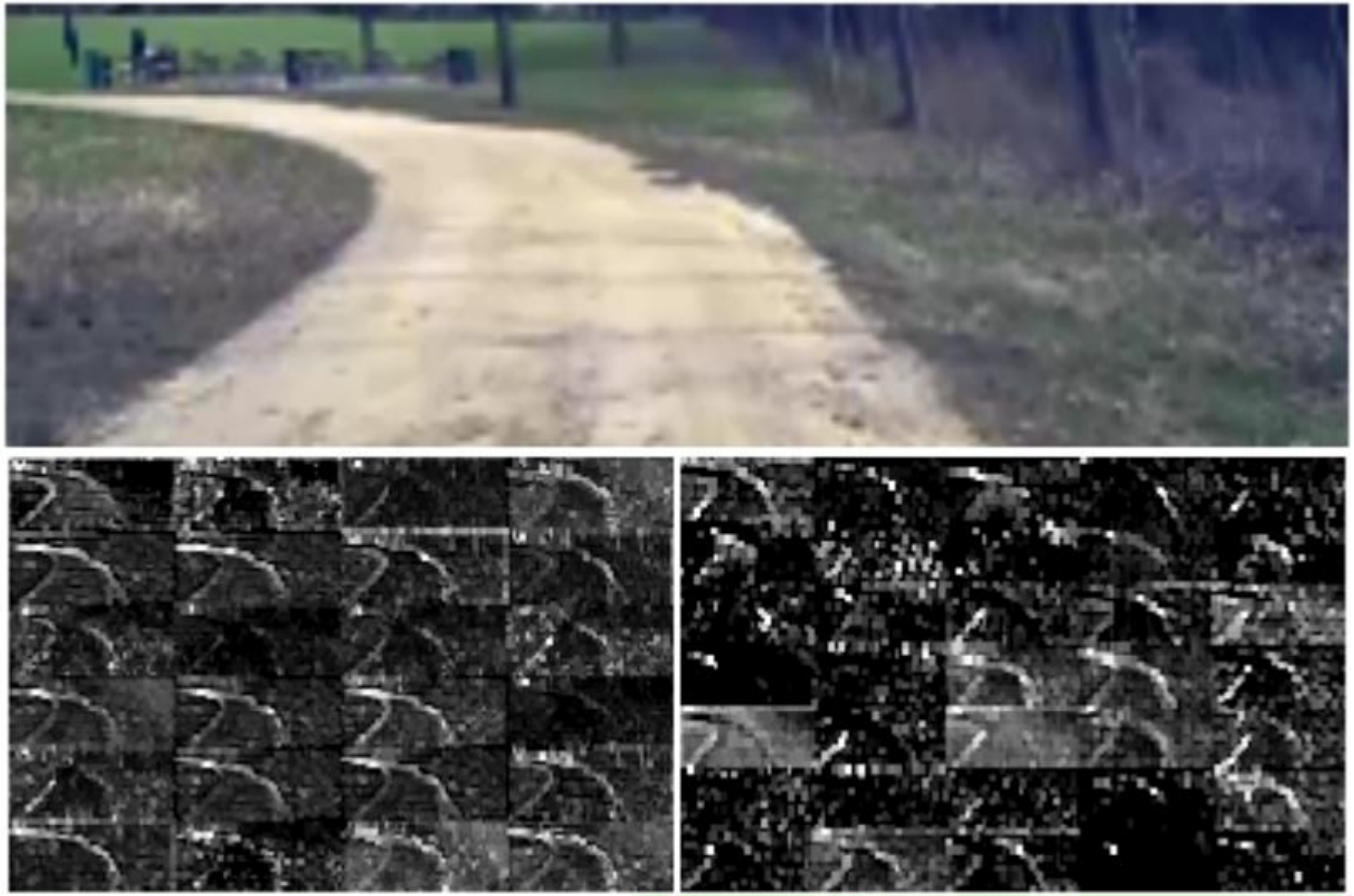


Convolution Operation (2D)



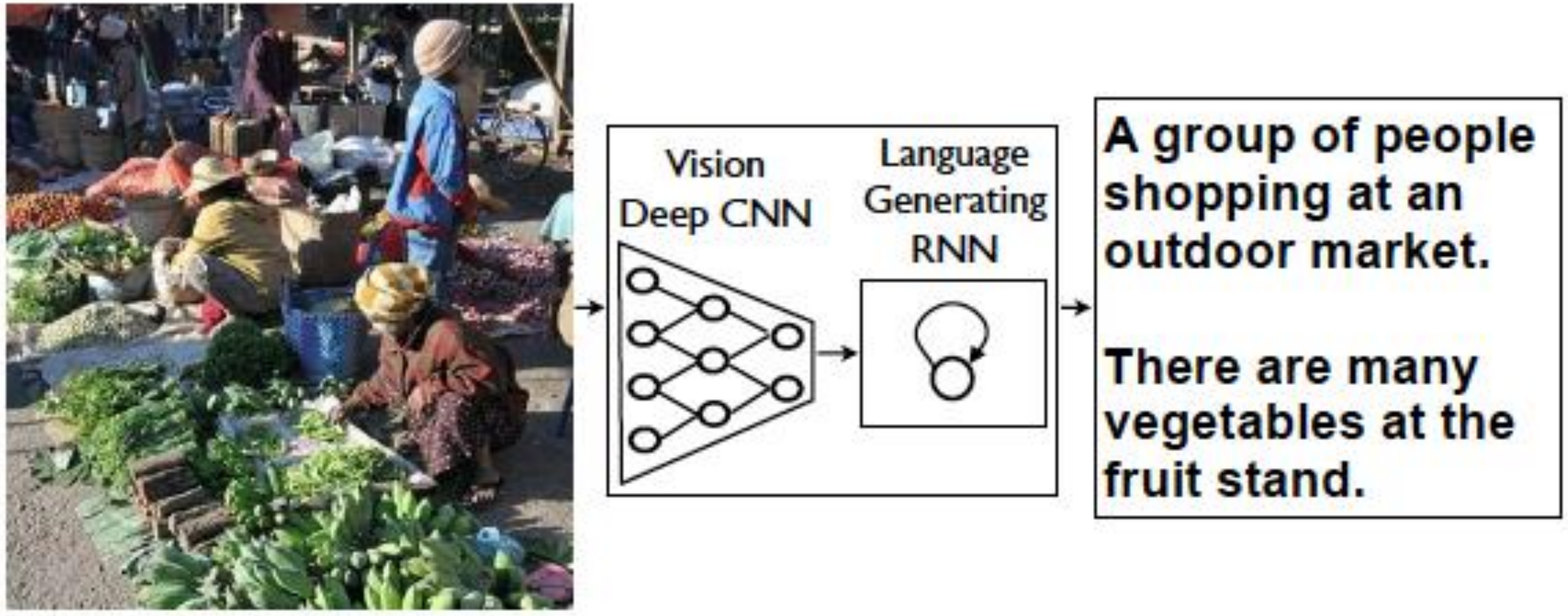
https://cambridgespark.com/

Self-driving Cars



Bojarski et al., “End-to-End Learning for Self-Driving Cars”, *arXiv preprint*, 2016

Image Captioning



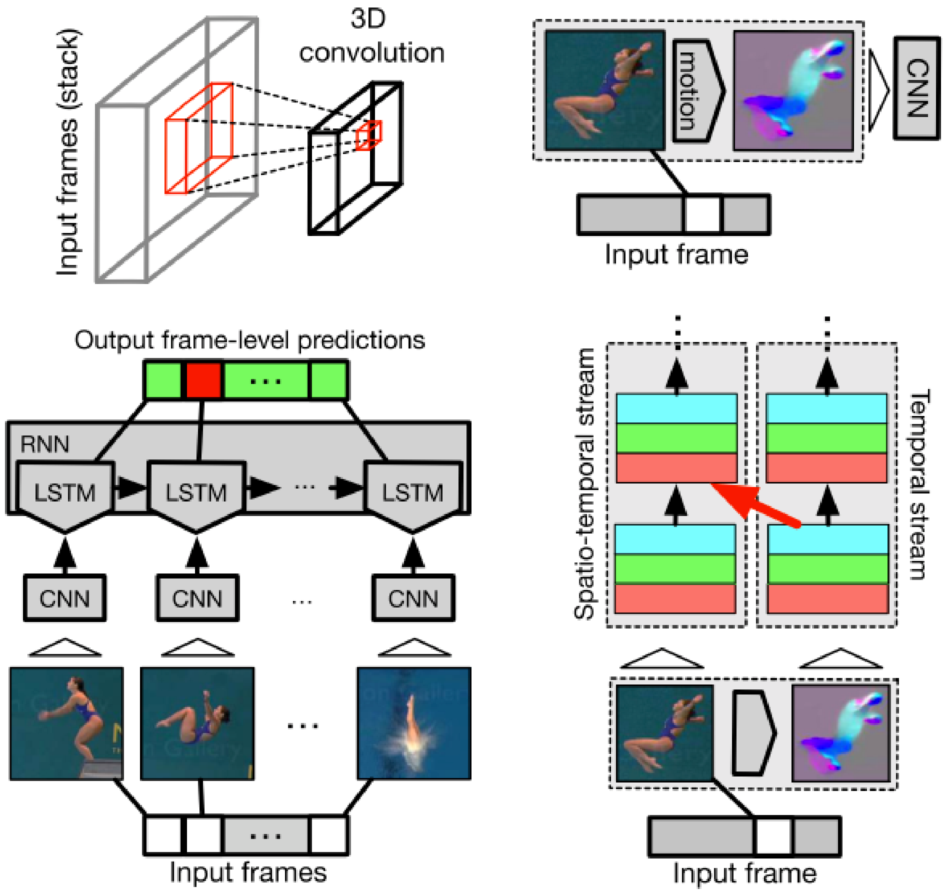
Vinyals et al., *Show and Tell: A Neural Image Caption Generator*, arXiv (2015)

Video Understanding

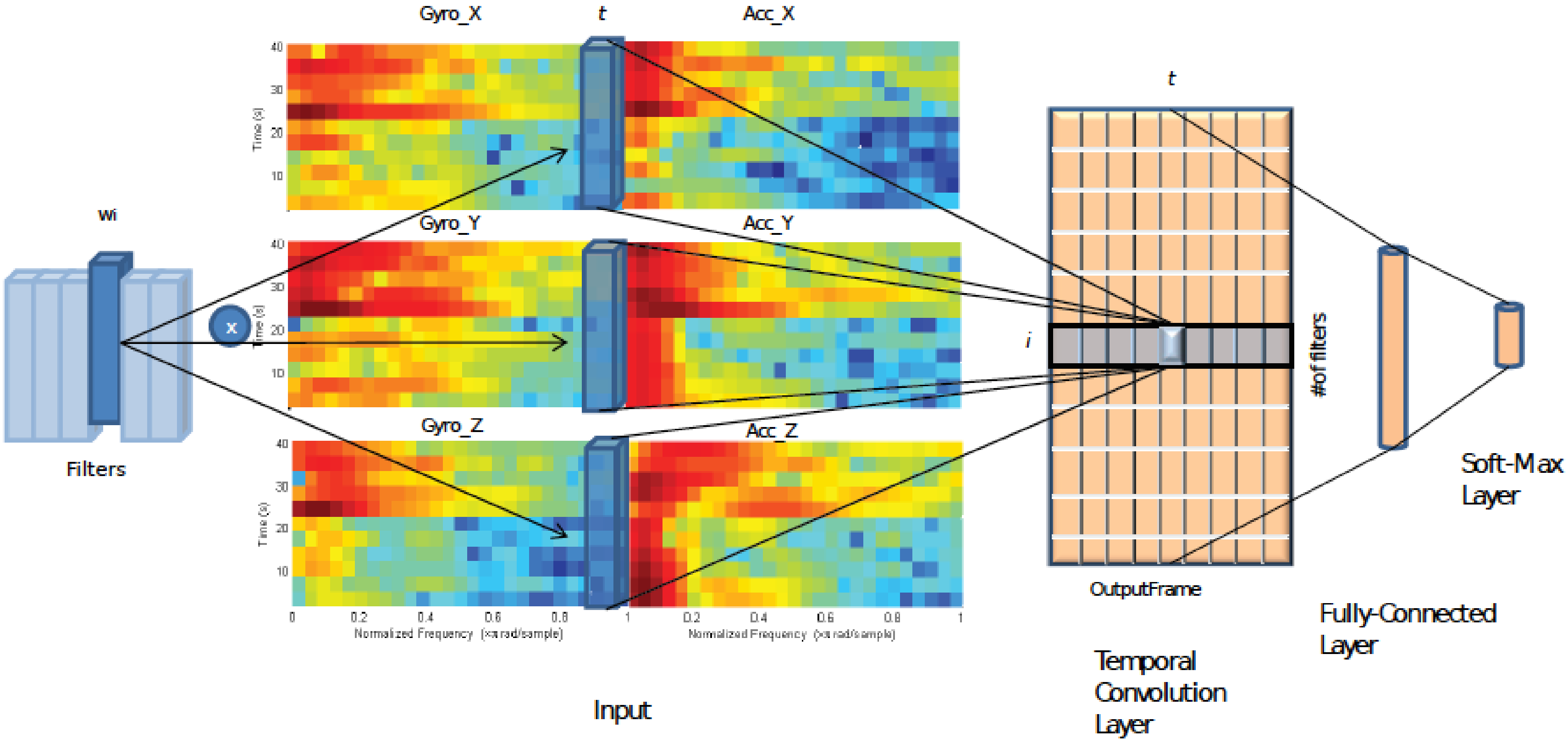
2017 IEEE 12th International Conference on Automatic Face & Gesture Recognition

A survey on deep learning based approaches for action and gesture recognition in image sequences

Maryam Asadi-Aghbolaghi[1,2,3], Albert Clapés[2,3] , Marco Bellantonio[4], Hugo Jair Escalante[5], Víctor Ponce-López[2,3,6], Xavier Baró[6], Isabelle Guyon[7], Shohreh Kasaei[1], Sergio Escalera[2,3]



Temporal Convolution

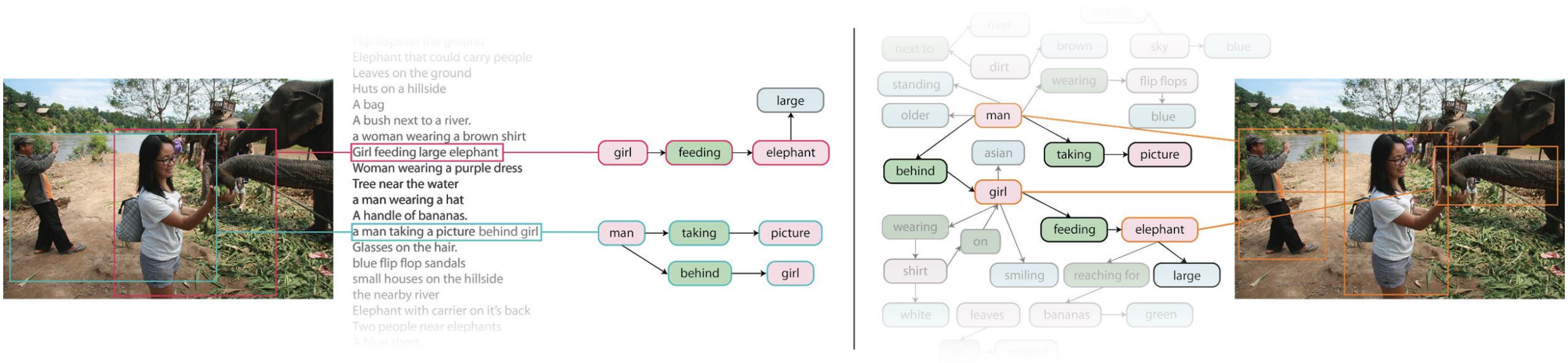


Ravi et al., 2016

Visual Question Answering (VQA)

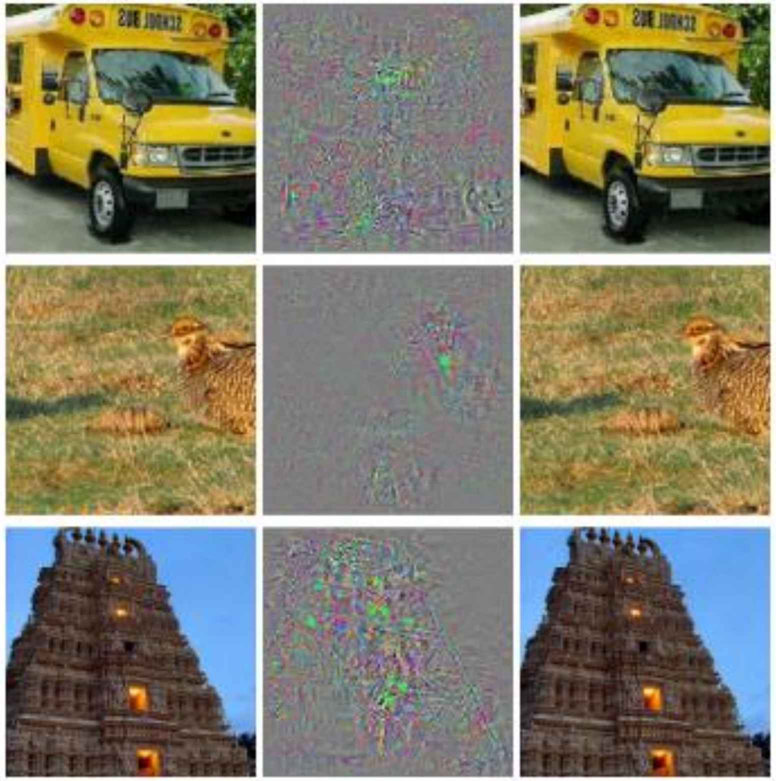
Goyal et al. (2017), Making the V in VQA Matter: Elevating the Role of Image Understanding in Visual Question Answering

Scene Graph



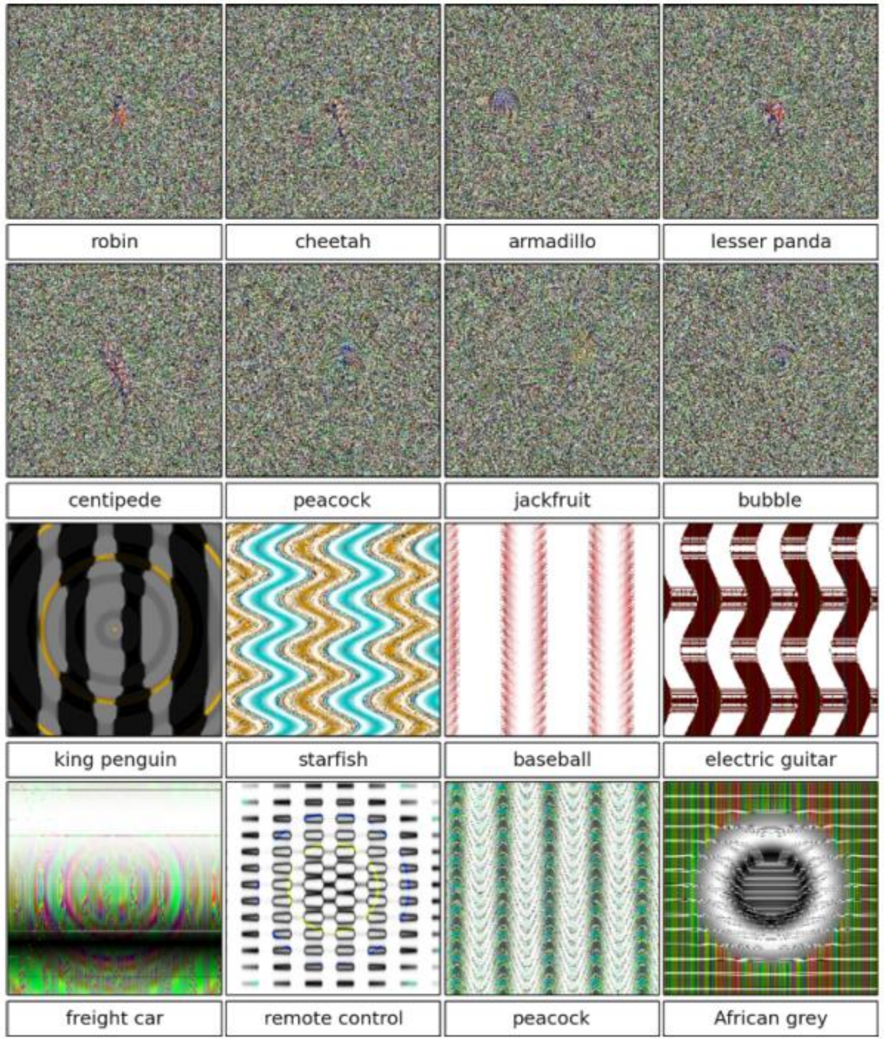
Krishna et al. (2016), Visual Genome:Connecting Language and Vision Using Crowdsourced Dense Image Annotations

Fooling Deep Neural Networks



Szegedy et al., “Intriguing Properties of Neural Networks”, *arXiv preprint*, 2014

Fooling Deep Neural Networks



Nguyen et al., "Deep Neural Networks are Easily Fooled: High Confidence Predictions for Unrecognizable Images", *CVPR*, 2015

Convolution Operation (1D)

Discrete 1-dim convolution with a kernel of size 3

Dot product of the kernel and the corresponding part of the original series sliding with stride of 1

Original Series	1	0	2	3	5	1	2	3
Kernel	-1	2	1					
Convolution Series	1							

$$(1, 0, 2) \cdot (-1, 2, 1) = 1*(-1) + 0*2 + 2*1 = 1$$

Convolution Operation (1D)

Discrete 1-dim convolution with a kernel of size 3

Dot product of the kernel and the corresponding part of the original series sliding with stride of 1

Original Series	1	0	2	3	5	1	2	3
Kernel		-1	2	1				
Convolution Series	1	7						

$(0, 2, 3) \cdot (-1, 2, 1) = 0*(-1) + 2*2 + 3*1 = 7$

Convolution Operation (1D)

Discrete 1-dim convolution with a kernel of size 3

Dot product of the kernel and the corresponding part of the original series sliding with stride of 1

Original Series	1	0	2	3	5	1	2	3
Kernel			-1	2	1			
Convolution Series	1	7	9					

$(2, 3, 5) \cdot (-1, 2, 1) = 2*(-1) + 3*2 + 5*1 = 9$

Convolution Operation (1D)

Discrete 1-dim convolution with a kernel of size 3

Dot product of the kernel and the corresponding part of the original series sliding with stride of 1

Original Series					1	0	2	3	5	1	2	3
Kernel								-1	2	1		
Convolution Series					1	7	9	8				

(3, 5, 1) · (-1, 2, 1) = 3*(-1) + 5*2 + 1*1 = 8

Convolution Operation (1D)

Discrete 1-dim convolution with a kernel of size 3

Dot product of the kernel and the corresponding part of the original series sliding with stride of 1

Original Series	1	0	2	3	5	1	2	3
Kernel					-1	2	1	
Convolution Series	1	7	9	8	-1			

$(5, 1, 2) \cdot (-1, 2, 1) = 5*(-1) + 1*2 + 2*1 = -1$

Convolution Operation (1D)

Discrete 1-dim convolution with a kernel of size 3

Dot product of the kernel and the corresponding part of the original series sliding with stride of 1

Original Series					1	0	2	3	5	1	2	3
Kernel										-1	2	1
Convolution Series					1	7	9	8	-1	6		

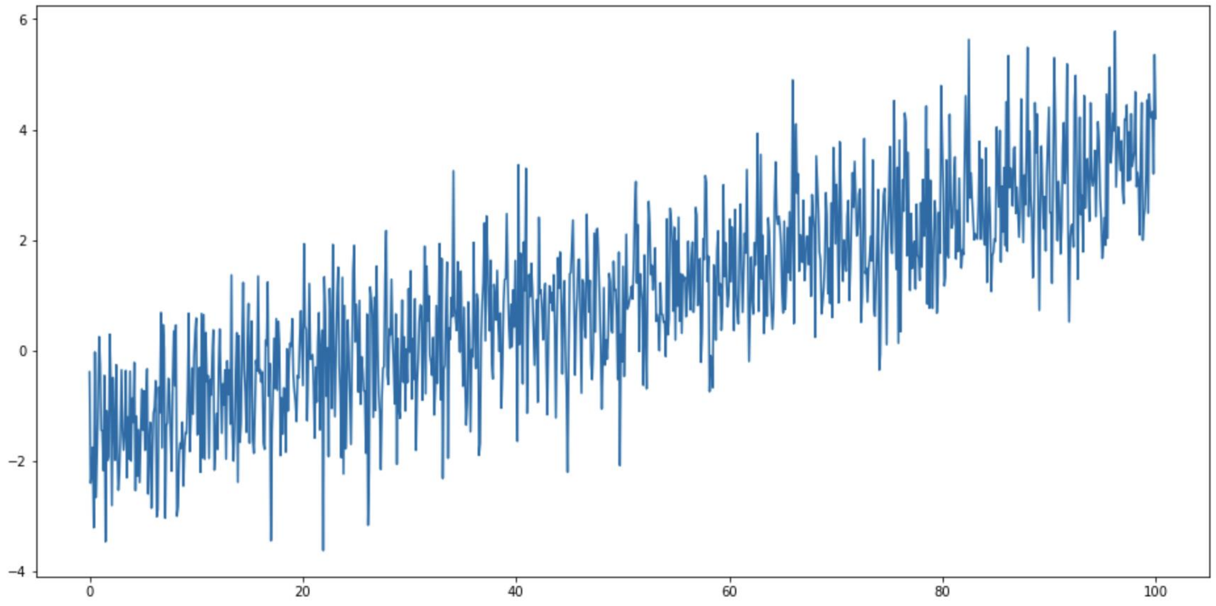
$(1, 2, 3) \cdot (-1, 2, 1) = 1*(-1) + 2*2 + 3*1 = 6$

Convolution Operation (1D)

In [1]:
```
import numpy as np
import matplotlib.pyplot as plt
```

In [2]:
```
N_series = 1000
x = np.linspace(0, 100, N_series)
y = 0.05 * x - 1.5 + np.random.randn(N_series)
```

In [3]:
```
plt.figure(figsize=(16, 8))
plt.plot(x, y)
plt.show()
```



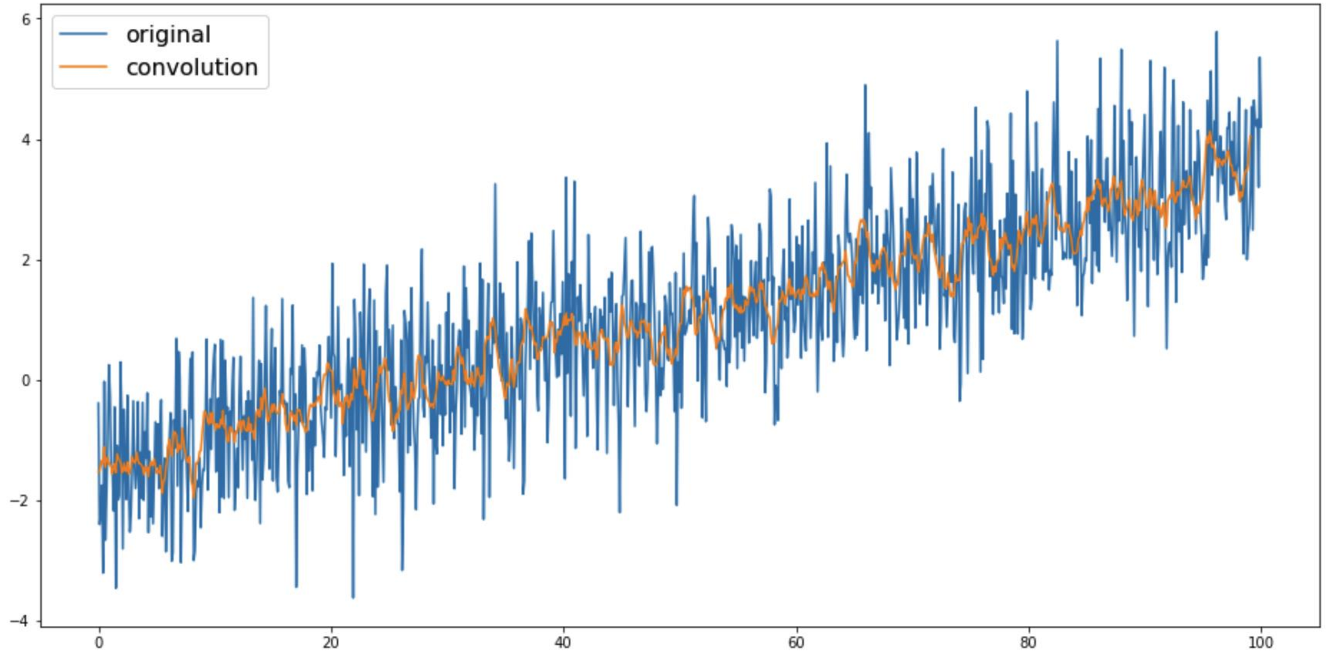
Convolution Operation (1D)

```python
In [4]: kernel = 0.1 * np.ones(10)

In [5]: def conv_1D(x, kernel):
            output = np.zeros((len(x) - len(kernel) + 1))
            for i in range(len(output)):
                output[i] = np.dot(kernel, x[i:i+len(kernel)])
            return output

In [6]: z = conv_1D(y, kernel)

In [7]: plt.figure(figsize=(16, 8))
        plt.plot(x, y, label='original')
        plt.plot(x[:len(z)], z, label='convolution')
        plt.legend(fontsize=16)
        plt.show()
```



Convolution Operation (1D)

Discrete 1-dim convolution with a kernel of size 3

Dot product of the kernel and the corresponding part of the original series sliding with stride of 1

Zero padding

Original Series		0	1	0	2	3	5	1	2	3	0
Kernel		-1	2	1							
Convolution Series			2								

$(0, 1, 0) \cdot (-1, 2, 1) = 0*(-1) + 1*2 + 0*1 = 2$

Convolution Operation (1D)

Discrete 1-dim convolution with a kernel of size 3

Dot product of the kernel and the corresponding part of the original series sliding with stride of 1

Zero padding

Original Series	0	1	0	2	3	5	1	2	3	0
Kernel		-1	2	1						
Convolution Series		2	1							

$(1, 0, 2) \cdot (-1, 2, 1) = 1*(-1) + 0*2 + 2*1 = 1$

Convolution Operation (1D)

Discrete 1-dim convolution with a kernel of size 3

Dot product of the kernel and the corresponding part of the original series sliding with stride of 1

Zero padding

Original Series	0	1	0	2	3	5	1	2	3	0
Kernel			-1	2	1					
Convolution Series		2	1	7						

$(0, 2, 3) \cdot (-1, 2, 1) = 0*(-1) + 2*2 + 3*1 = 7$

Convolution Operation (1D)

Discrete 1-dim convolution with a kernel of size 3

Dot product of the kernel and the corresponding part of the original series sliding with stride of 1

Zero padding

Original Series	0	1	0	2	3	5	1	2	3	0
Kernel				-1	2	1				
Convolution Series		2	1	7	9					

$$(2, 3, 5) \cdot (-1, 2, 1) = 2*(-1) + 3*2 + 5*1 = 9$$

Convolution Operation (1D)

Discrete 1-dim convolution with a kernel of size 3

Dot product of the kernel and the corresponding part of the original series sliding with stride of 1

Zero padding

Original Series	0	1	0	2	3	5	1	2	3	0
Kernel					-1	2	1			
Convolution Series		2	1	7	9	8				

$(3, 5, 1) \cdot (-1, 2, 1) = 3*(-1) + 5*2 + 1*1 = 8$

Convolution Operation (1D)

Discrete 1-dim convolution with a kernel of size 3

Dot product of the kernel and the corresponding part of the original series sliding with stride of 1

Zero padding

Original Series	0	1	0	2	3	5	1	2	3	0
Kernel						-1	2	1		
Convolution Series		2	1	7	9	8	-1			

(5, 1, 2) · (-1, 2, 1) = 5*(-1) + 1*2 + 2*1 = -1

Convolution Operation (1D)

Discrete 1-dim convolution with a kernel of size 3

Dot product of the kernel and the corresponding part of the original series sliding with stride of 1

Zero padding

Original Series	0	1	0	2	3	5	1	2	3	0
Kernel							-1	2	1	
Convolution Series		2	1	7	9	8	-1	6		

$(1, 2, 3) \cdot (-1, 2, 1) = 1*(-1) + 2*2 + 3*1 = 6$

Convolution Operation (1D)

Discrete 1-dim convolution with a kernel of size 3

Dot product of the kernel and the corresponding part of the original series sliding with stride of 1

Zero padding

Original Series	0	1	0	2	3	5	1	2	3	0
Kernel								-1	2	1
Convolution Series		2	1	7	9	8	-1	6	4	

$(2, 3, 0) \cdot (-1, 2, 1) = 2*(-1) + 3*2 + 0*1 = 4$

Convolution Operation (1D)

Discrete 1-dim convolution with a kernel of size 3

Dot product of the kernel and the corresponding part of the original series sliding with stride of 2

Zero padding

Original Series	0	1	0	2	3	5	1	2	0
Kernel	-1	2	1						
Convolution Series		2							

(0, 1, 0) · (-1, 2, 1) = 0*(-1) + 1*2 + 0*1 = 2

Convolution Operation (1D)

Discrete 1-dim convolution with a kernel of size 3

Dot product of the kernel and the corresponding part of the original series sliding with stride of 2

Zero padding

Original Series	0	1	0	2	3	5	1	2	0
Kernel			-1	2	1				
Convolution Series		2	7						

$(0, 2, 3) \cdot (-1, 2, 1) = 0*(-1) + 2*2 + 3*1 = 7$

Convolution Operation (1D)

Discrete 1-dim convolution with a kernel of size 3

Dot product of the kernel and the corresponding part of the original series sliding with stride of 2

Zero padding

Original Series	0	1	0	2	3	5	1	2	0
Kernel					-1	2	1		
Convolution Series		2	7	8					

(3, 5, 1) · (-1, 2, 1) = 3*(-1) + 5*2 + 1*1 = 8

Convolution Operation (1D)

Discrete 1-dim convolution with a kernel of size 3

Dot product of the kernel and the corresponding part of the original series sliding with stride of 2

Zero padding

Original Series	0	1	0	2	3	5	1	2	0
Kernel							-1	2	1
Convolution Series		2	7	8	3				

(1, 2, 0) · (-1, 2, 1) = 1*(-1) + 2*2 + 0*1 = 3

Convolution (1D) Summing Up

Input of size W

Kernel of size K

Stride of S

Padding of P

Output size = ?

Convolution (1D) Summing Up

Input of size W

Padding of P

Kernel of size K

Stride of S

Output size = (W + 2*P - K) / S + 1

=> Must be an integer for valid convolution!

Convolution Operation (2D)

Discrete 2-dim convolution with a kernel of size 3x3

Dot product of the kernel and the corresponding part of the input matrix sliding with stride of 1

Input Matrix

1	0	2	1	-1	1
0	1	-1	1	0	2
1	0	0	0	-1	1
0	-1	-1	0	2	0
1	1	0	-2	1	0
0	0	1	-1	0	1

Kernel

-1	0	1			
0	2	1			
1	0	-1			

Output Matrix

3			

$(1, 0, 2, 0, 1, -1, 1, 0, 0) \cdot (-1, 0, 1, 0, 2, 1, 1, 0, -1) = 3$

Convolution Operation (2D)

Discrete 2-dim convolution with a kernel of size 3x3

Dot product of the kernel and the corresponding part of the input matrix sliding with stride of 1

Input Matrix

1	0	2	1	-1	1
0	1	-1	1	0	2
1	0	0	0	-1	1
0	-1	-1	0	2	0
1	1	0	-2	1	0
0	0	1	-1	0	1

Kernel

	-1	0	1		
	0	2	1		
	1	0	-1		

Output Matrix

3	0		

$(0, 2, 1, 1, -1, 1, 0, 0, 0) \cdot (-1, 0, 1, 0, 2, 1, 1, 0, -1) = 0$

Convolution Operation (2D)

Discrete 2-dim convolution with a kernel of size 3x3

Dot product of the kernel and the corresponding part of the input matrix sliding with stride of 1

Input Matrix

1	0	2	1	-1	1
0	1	-1	1	0	2
1	0	0	0	-1	1
0	-1	-1	0	2	0
1	1	0	-2	1	0
0	0	1	-1	0	1

Kernel

		-1	0	1	
		0	2	1	
		1	0	-1	

Output Matrix

3	0	0	

(2, 1, -1, -1, 1, 0, 0, 0, -1) · (-1, 0, 1, 0, 2, 1, 1, 0, -1) = 0

Convolution Operation (2D)

Discrete 2-dim convolution with a kernel of size 3x3

Dot product of the kernel and the corresponding part of the input matrix sliding with stride of 1

Input Matrix

1	0	2	1	-1	1
0	1	-1	1	0	2
1	0	0	0	-1	1
0	-1	-1	0	2	0
1	1	0	-2	1	0
0	0	1	-1	0	1

Kernel

			-1	0	1
			0	2	1
			1	0	-1

Output Matrix

3	0	0	2

$(1, -1, 1, 1, 0, 2, 0, -1, 1) \cdot (-1, 0, 1, 0, 2, 1, 1, 0, -1) = 2$

Convolution Operation (2D)

Discrete 2-dim convolution with a kernel of size 3x3

Dot product of the kernel and the corresponding part of the input matrix sliding with stride of 1

Input Matrix

1	0	2	1	-1	1
0	1	-1	1	0	2
1	0	0	0	-1	1
0	-1	-1	0	2	0
1	1	0	-2	1	0
0	0	1	-1	0	1

Kernel

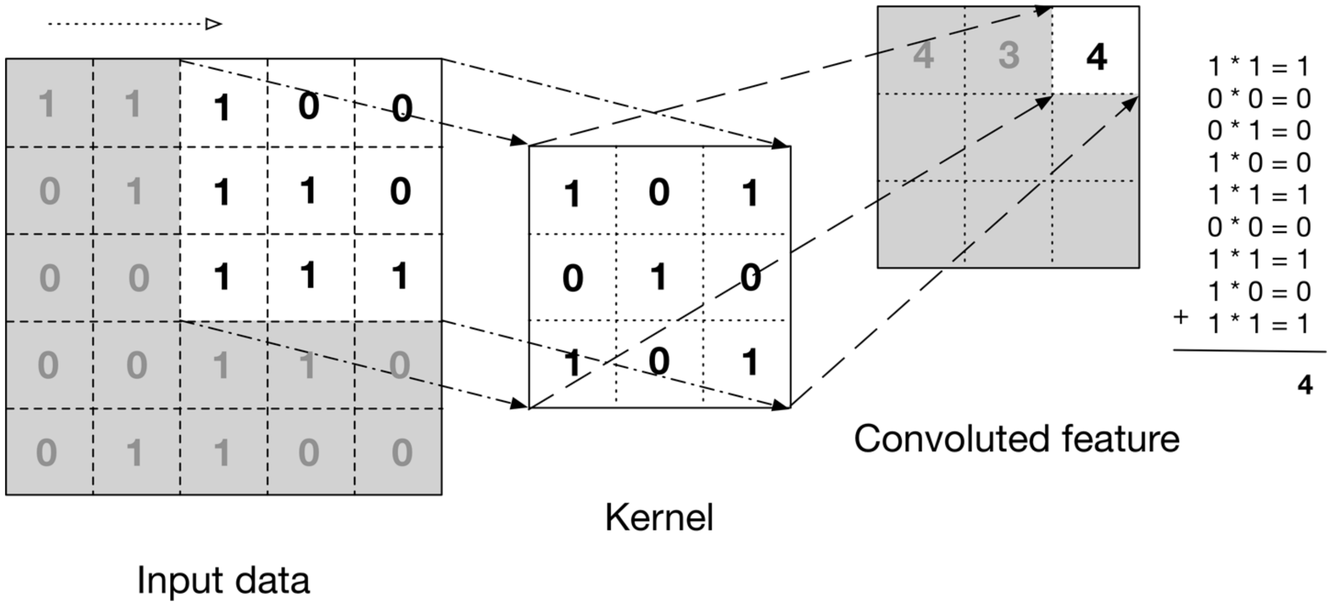
-1	0	1			
0	2	1			
1	0	-1			

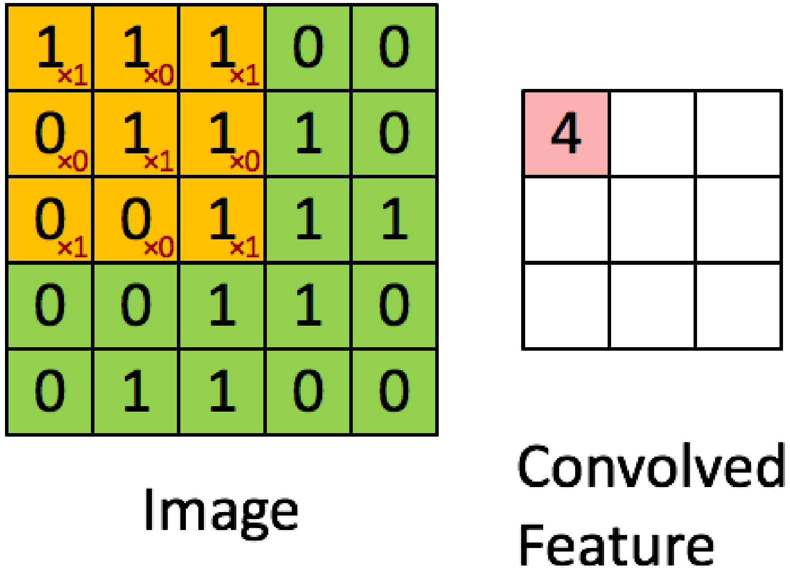
Output Matrix

3	0	0	2
0			

$(0, 1, -1, 1, 0, 0, 0, -1, -1) \cdot (-1, 0, 1, 0, 2, 1, 1, 0, -1) = 0$

Convolution Operation (2D)

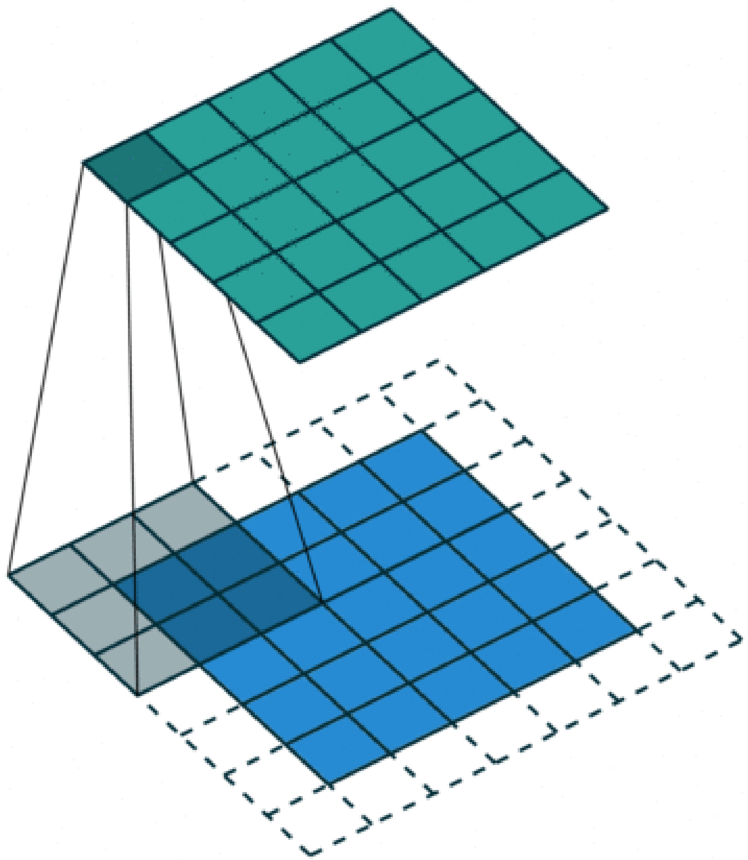




https://www.analyticsvidhya.com/blog/2021/05/convolutional-neural-networks-cnn/

https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a5

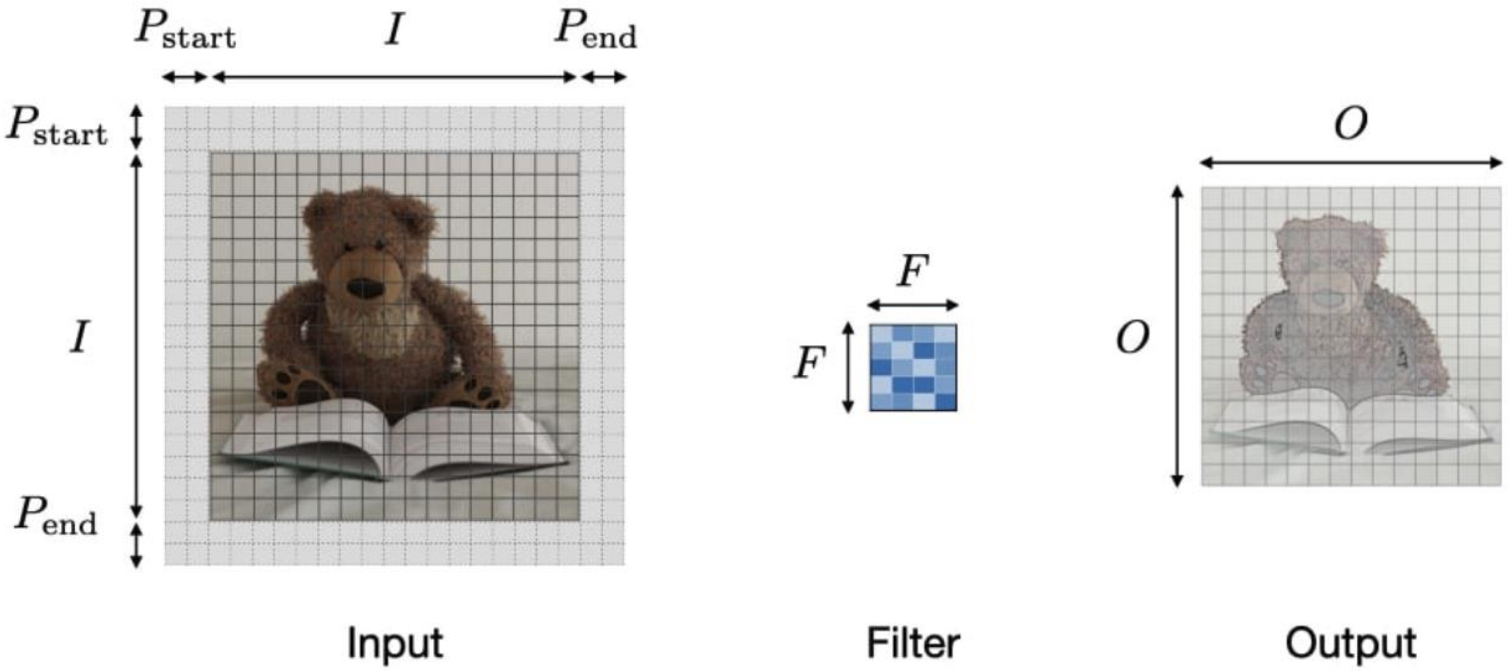
Padding in 2D Convolution



https://github.com/vdumoulin/conv_arithmetic

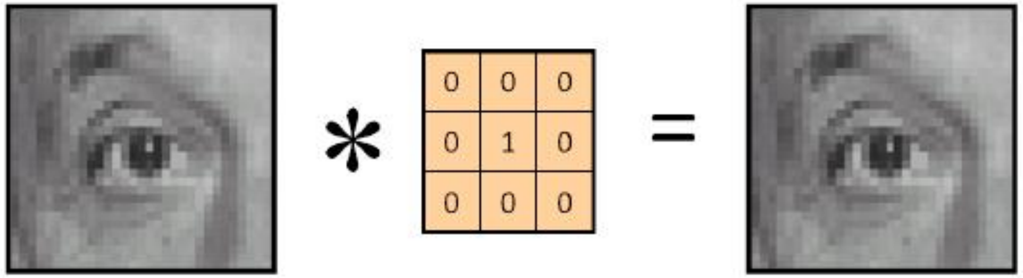
Input and Output Sizes

$$O = \frac{I - F + P_{\text{start}} + P_{\text{end}}}{S} + 1$$



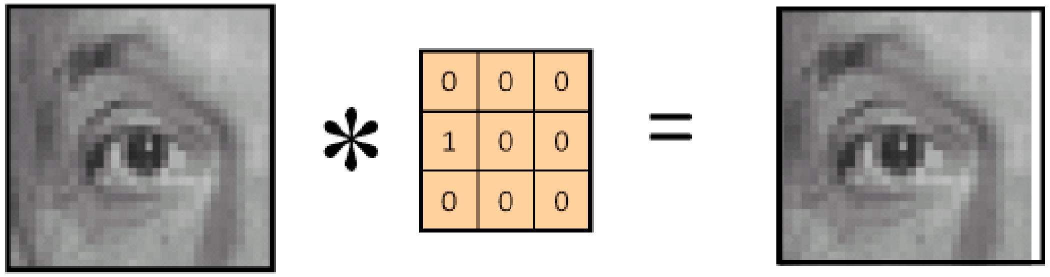
https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-convolutional-neural-networks

Effects of Convolution Kernels (Filters)



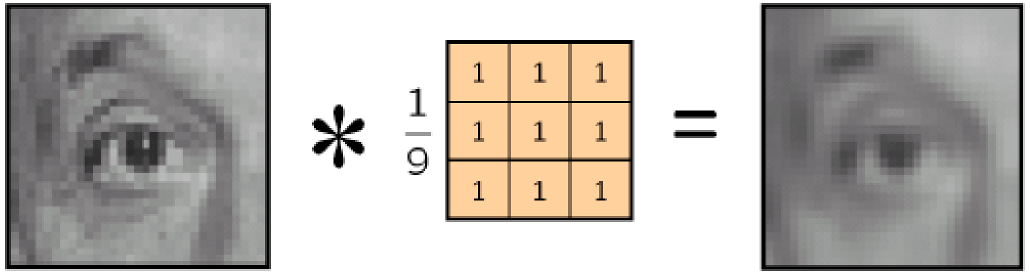
Original

Identical image



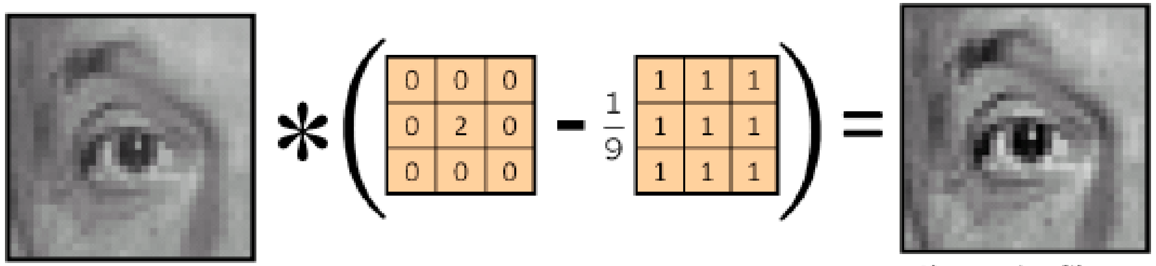
Original

Shifted left
By 1 pixel



Original

Blur (with a mean filter)



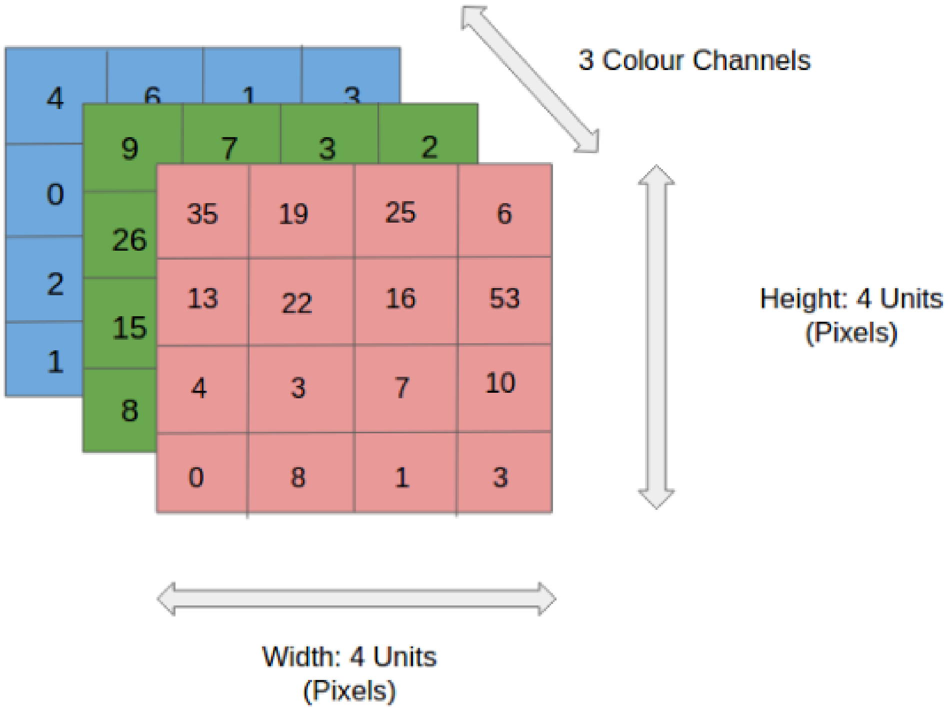
Original

Sharpening filter
(accentuates edges)

https://www.cs.cornell.edu/courses/cs1114/2013sp/sections/S06_convolution.pdf

Color Image has Three Channels! (RGB)

Color input image = Tensor of rank 3
Tensor of rank N = N-dim array of numbers (simplified definition)

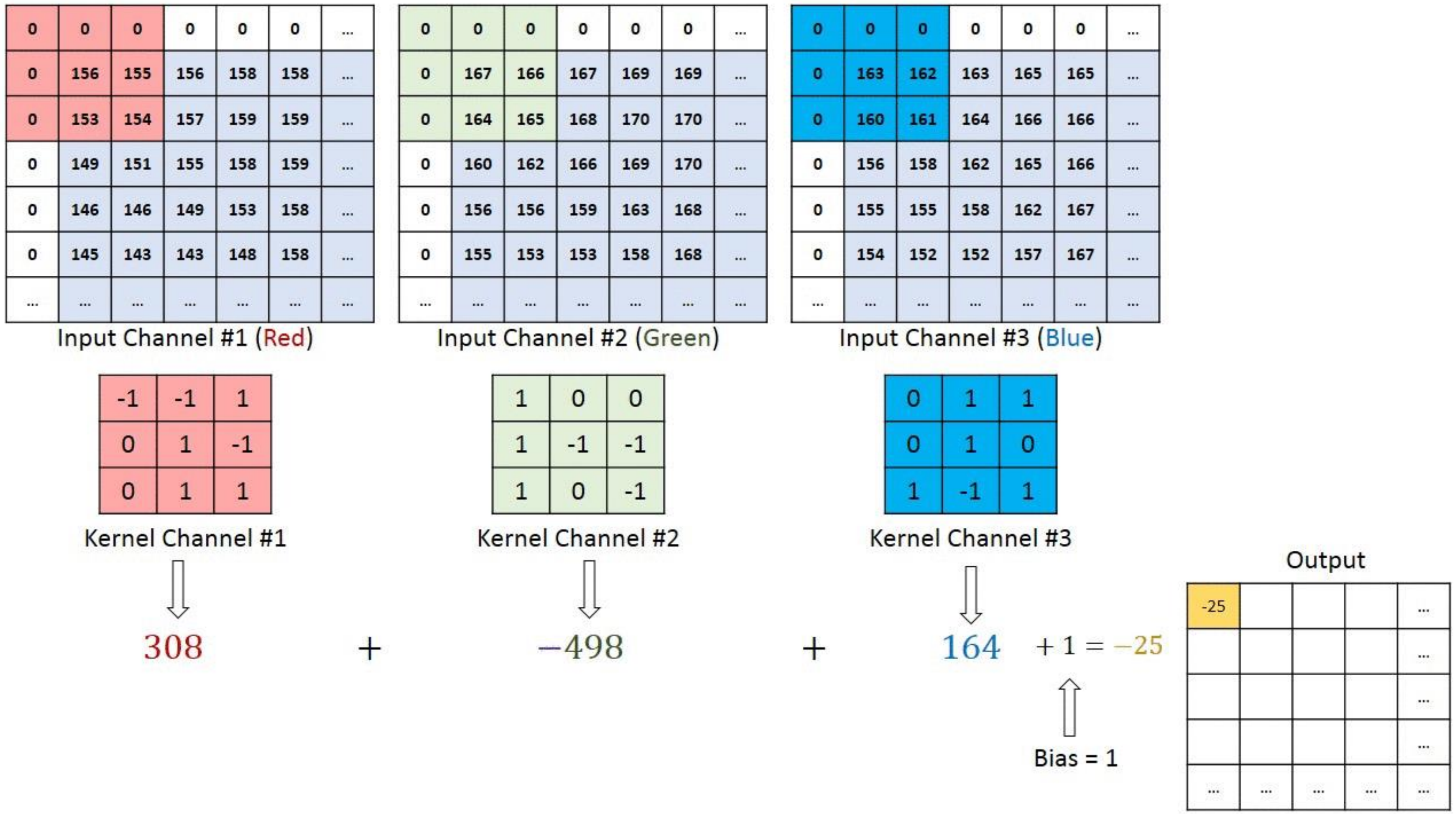


[w, h, c] = [4, 4, 3]

https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53

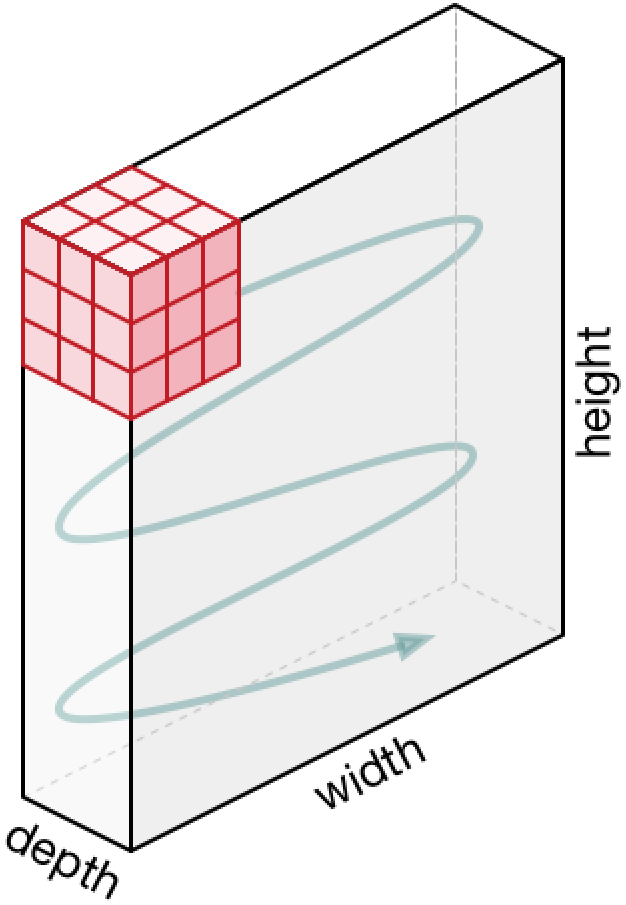
Kernel has parameters for all input channels (depth)

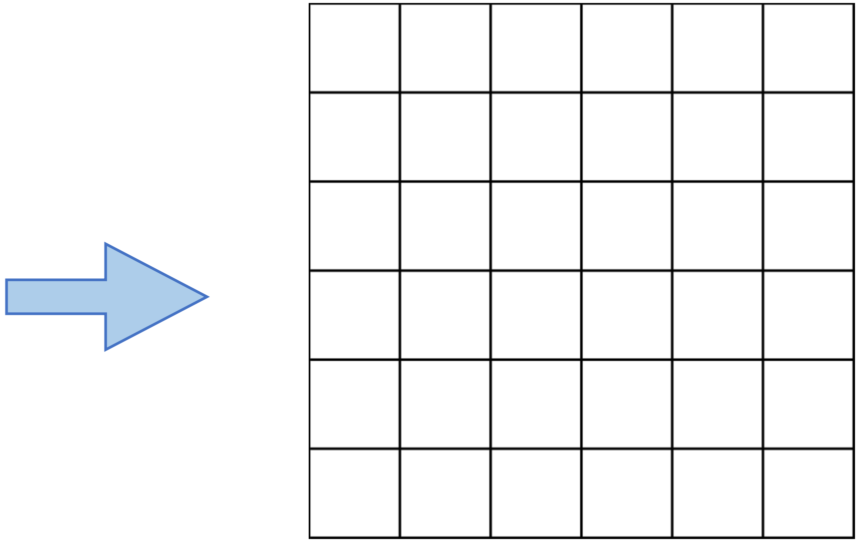
Kernel size: [w, h, c] = [3, 3, 3]
Note: This is still 2D convolution! (Why?)



https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53

2D Convolution may have a depth dim., but moves in 2D

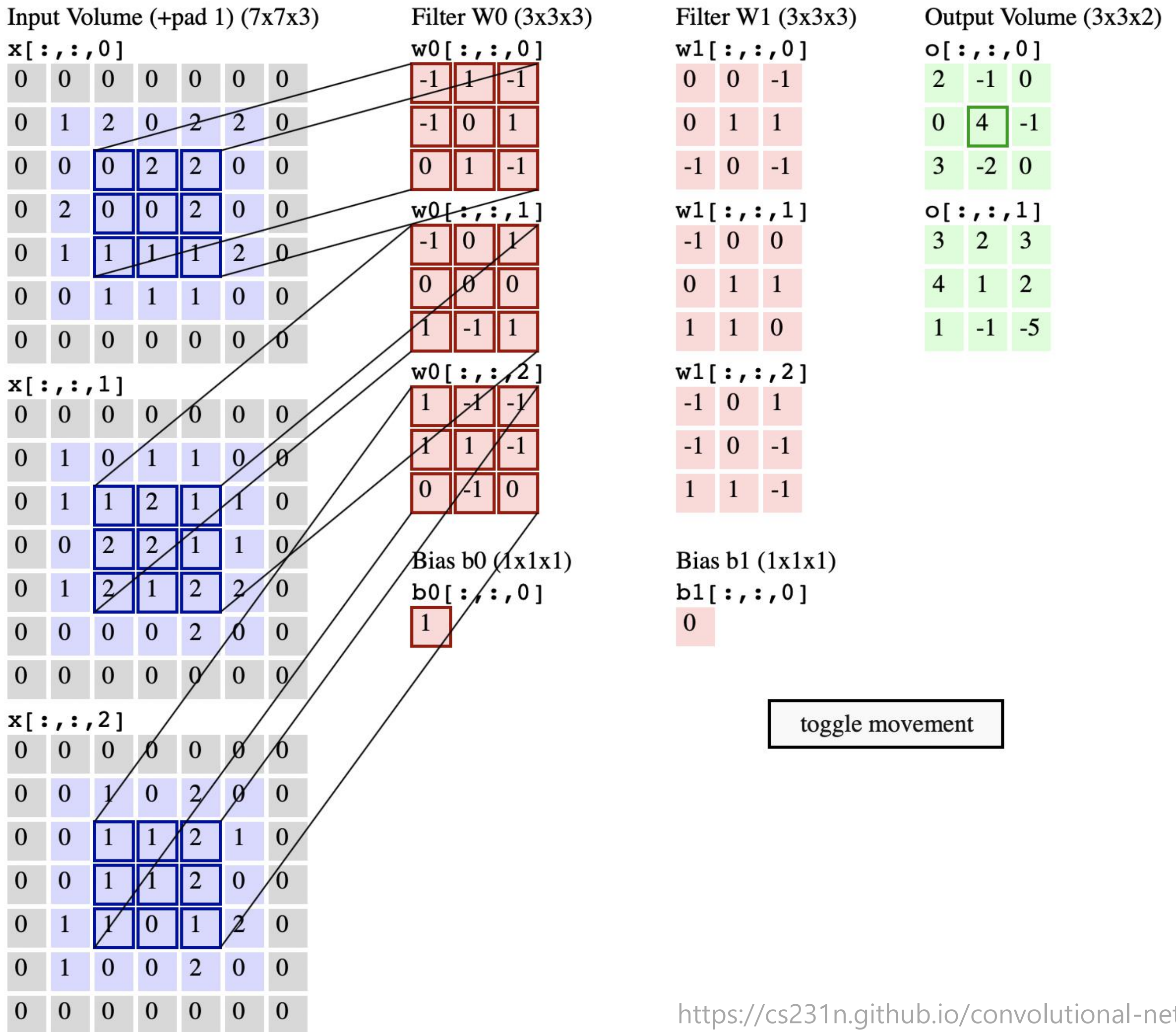




Output size: [w, h, 1]

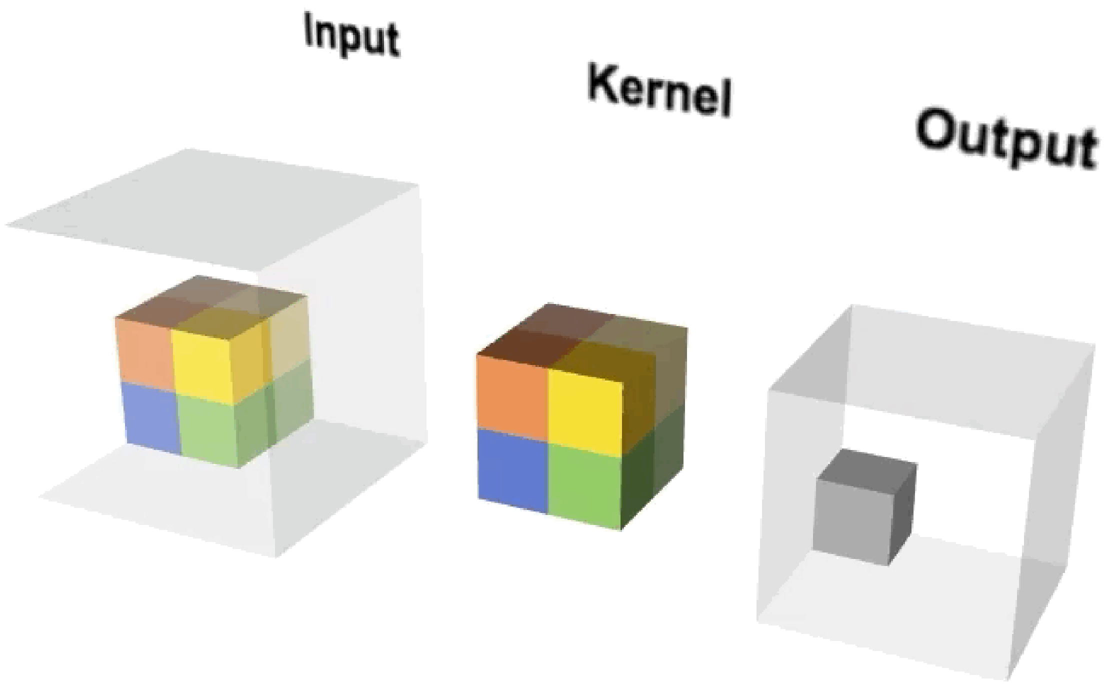
.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53

Multiple kernels (filters) for the same input image



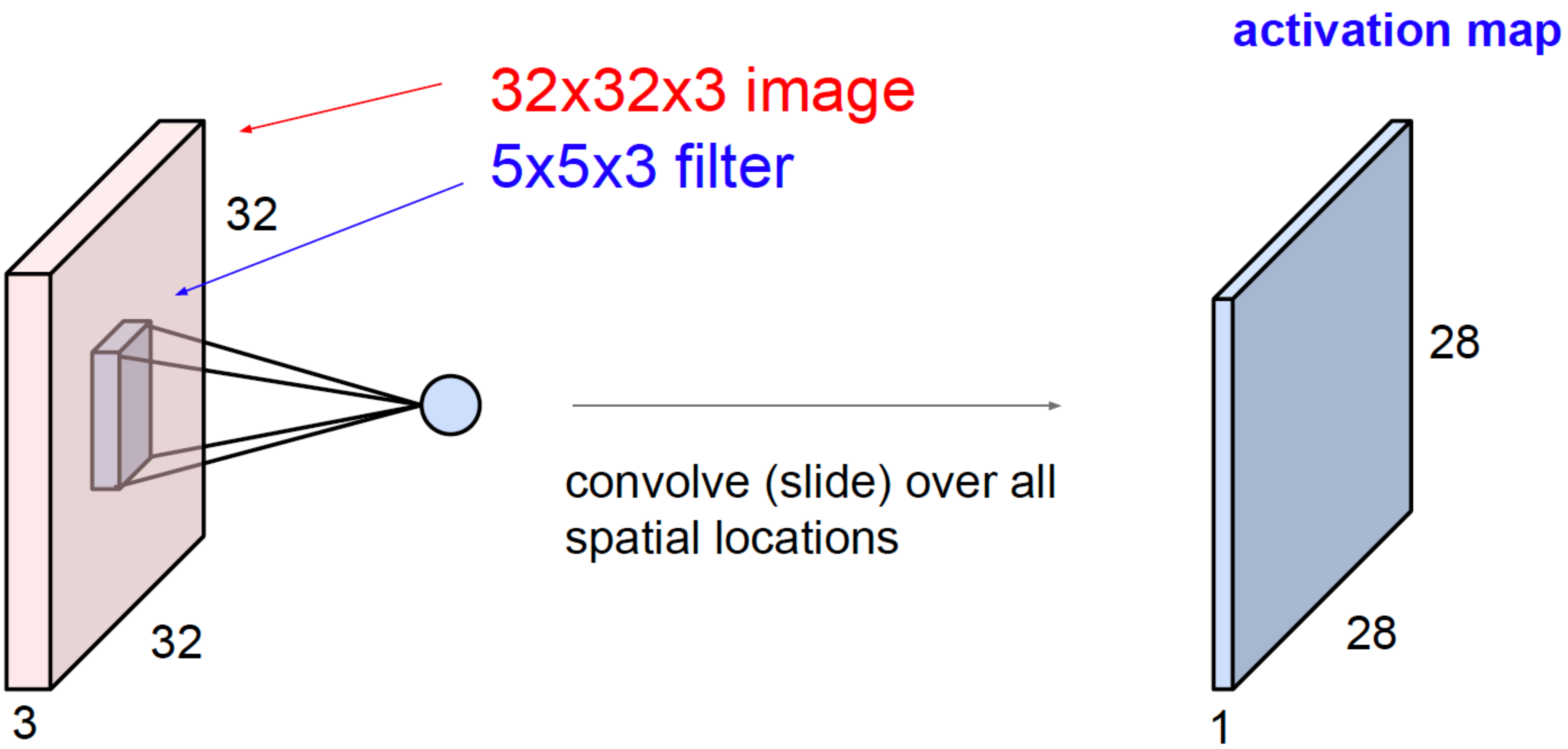
https://cs231n.github.io/convolutional-networks/

3D Convolution



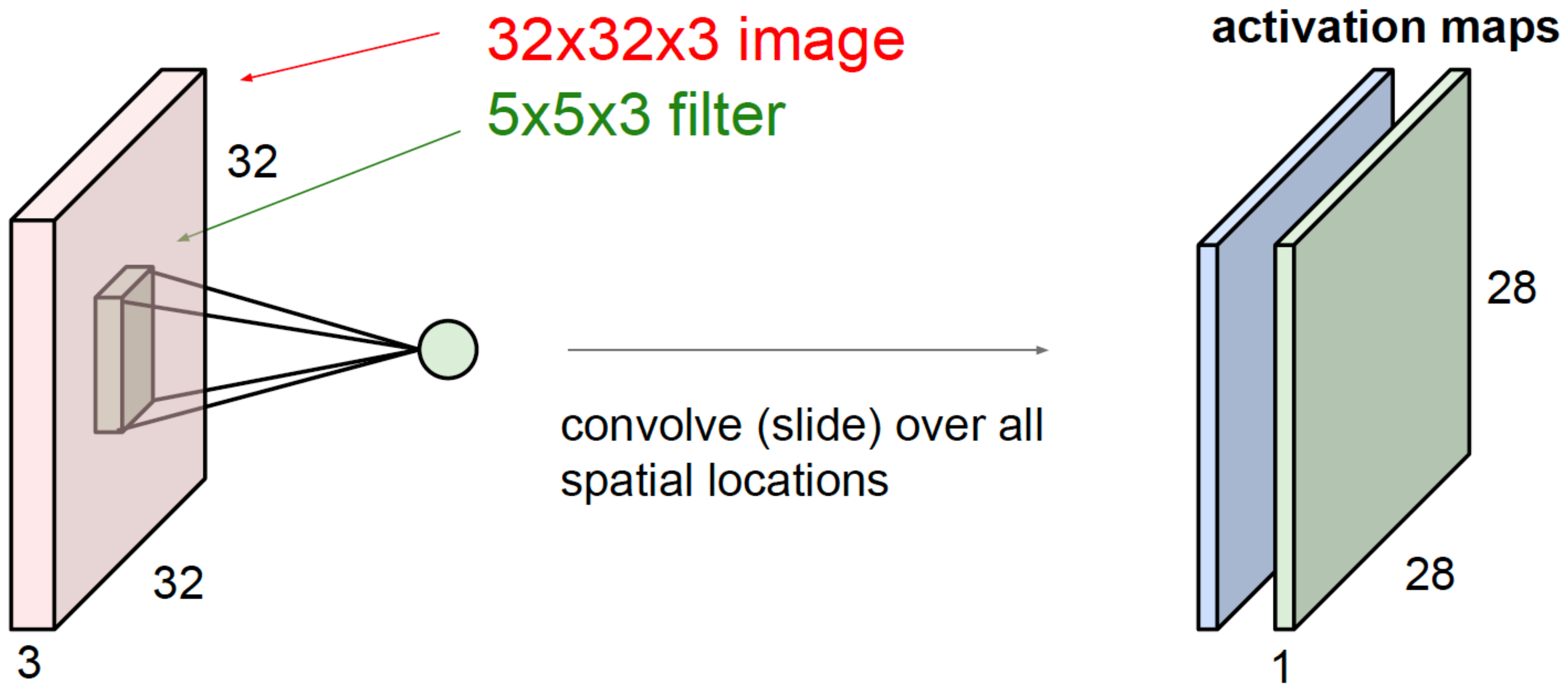
https://thomelane.github.io/convolutions/3DConv.html

Convolutional Layer

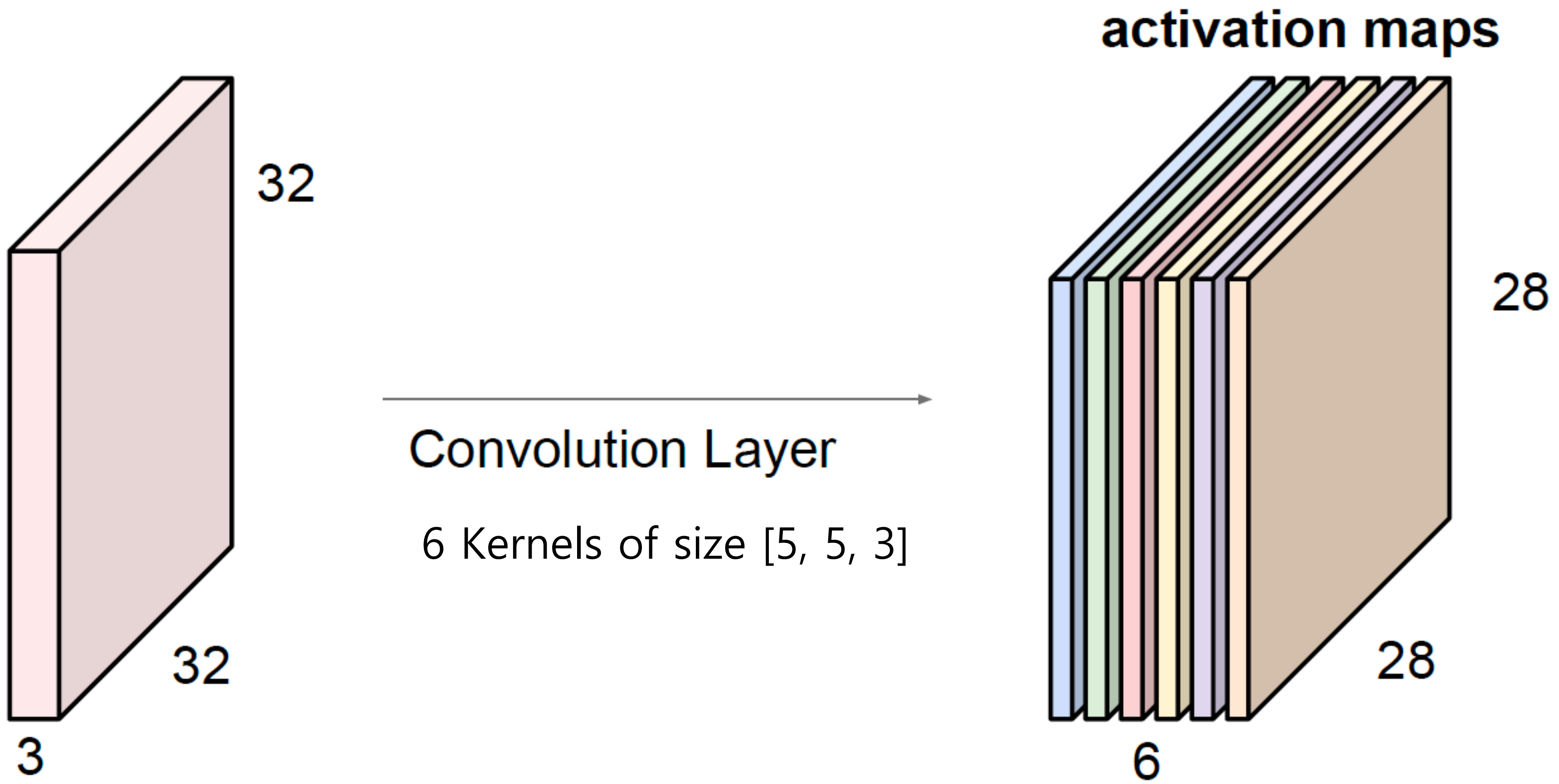


CS231n: Lecture 5, Stanford University

Convolutional Layer



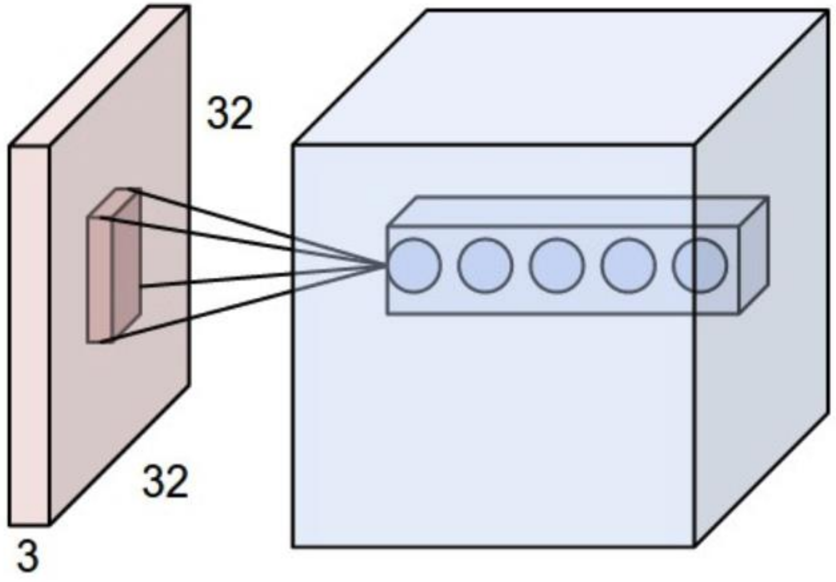
Convolutional Layer

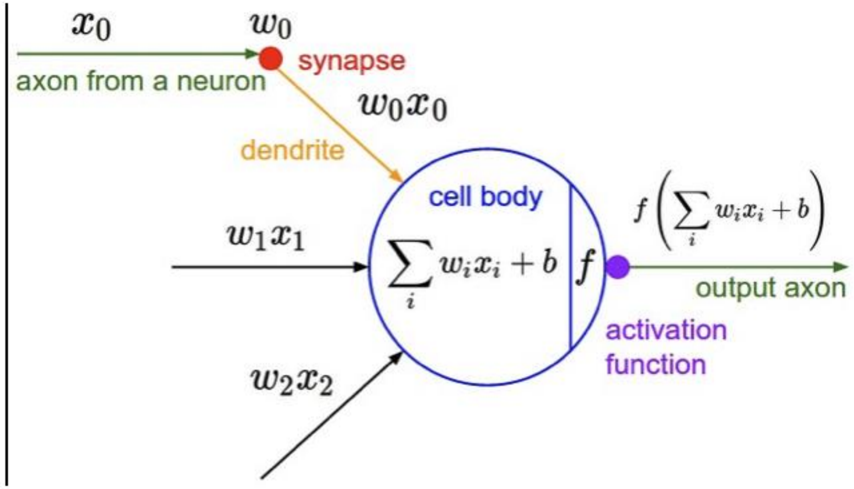


New “image” of size [28, 28, 6]

Local Connectivity & Parameter Sharing

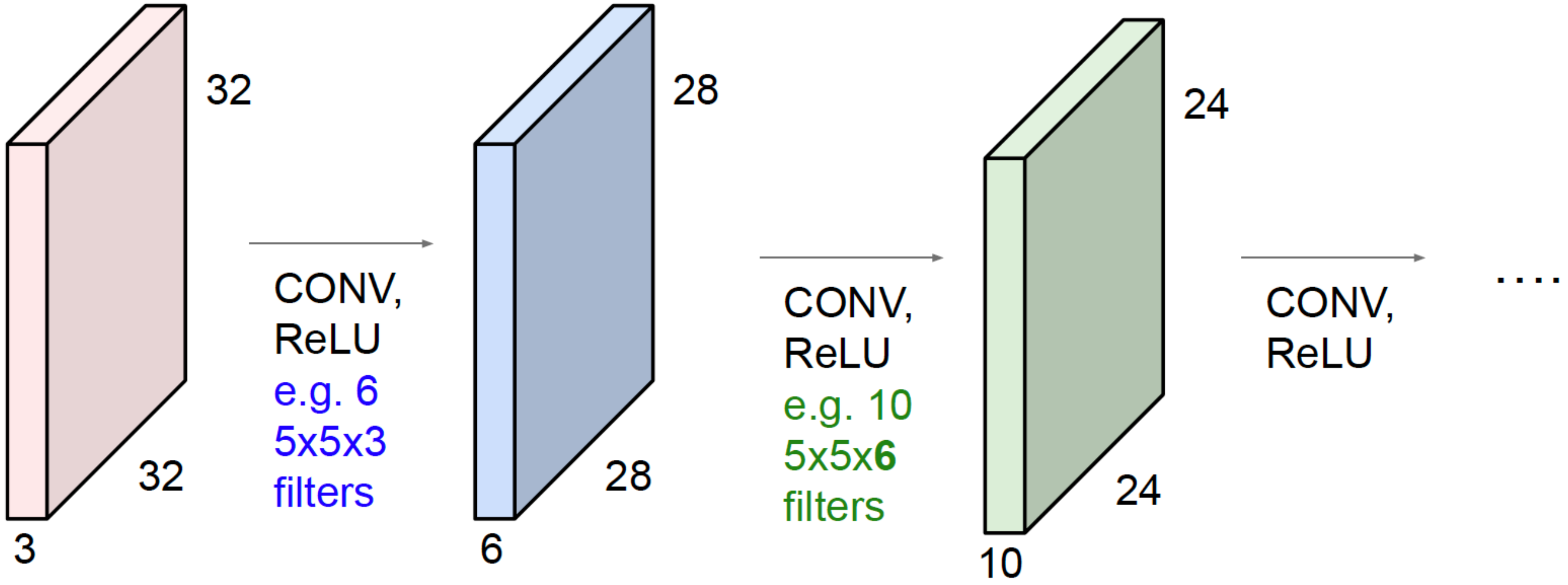
Convolution -> Local connectivity and parameter sharing of neurons





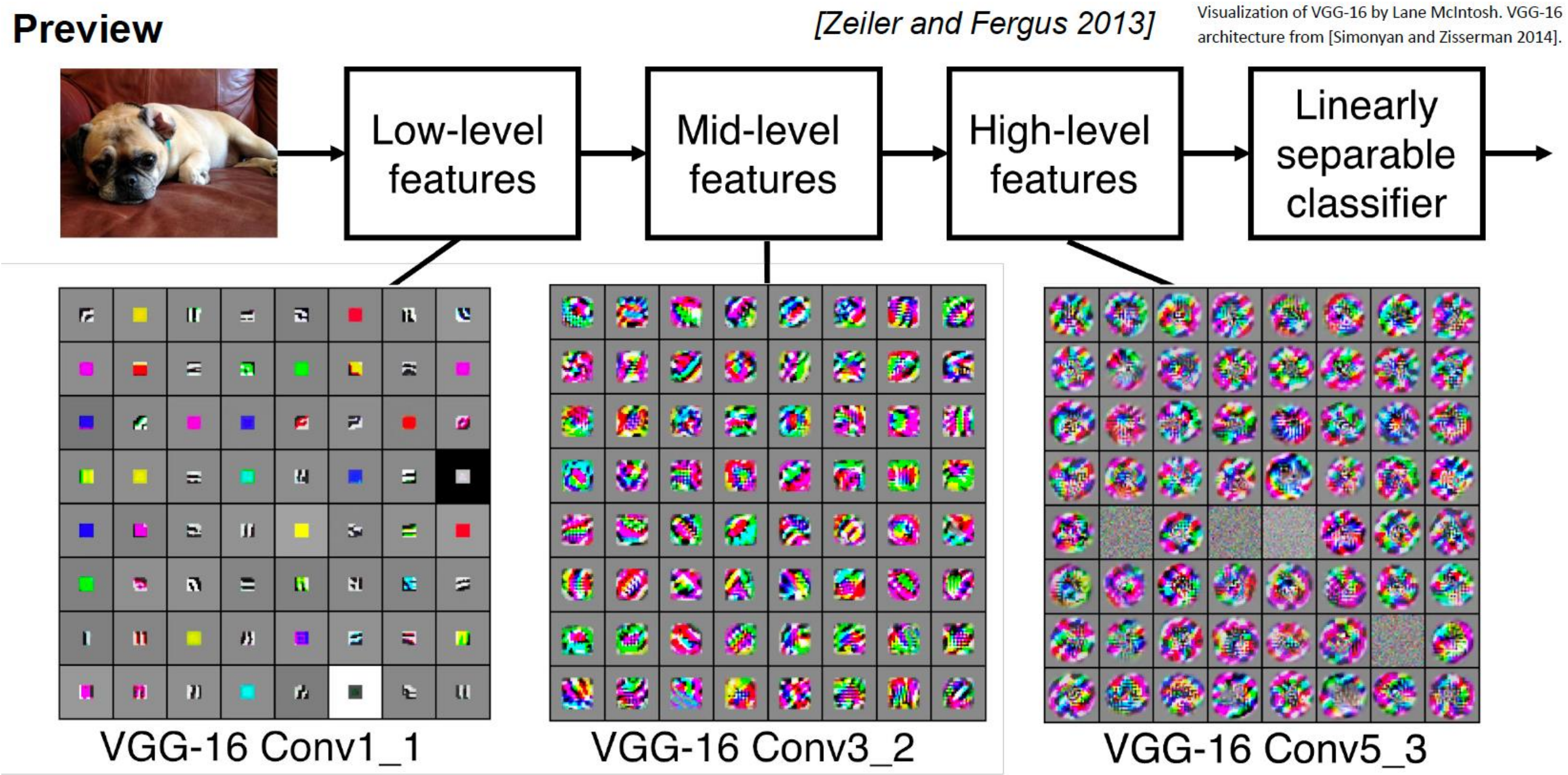
https://cs231n.github.io/convolutional-networks/

Convolutional Layers



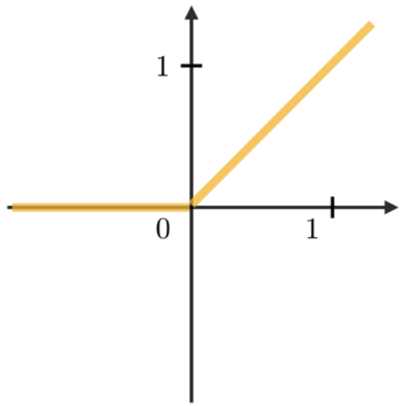
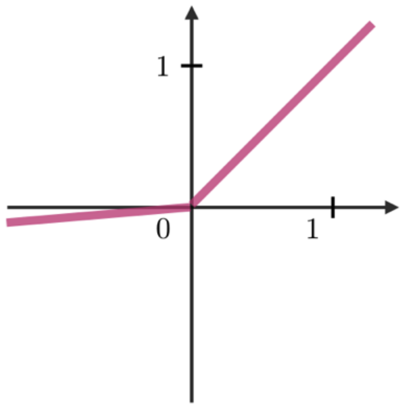
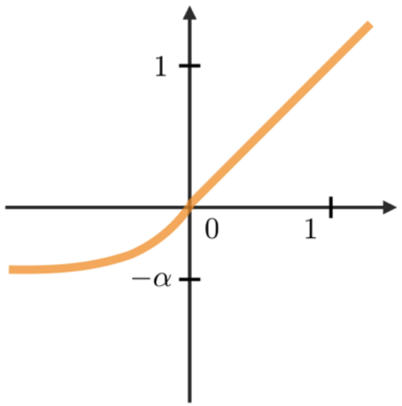
CS231n: Lecture 5, Stanford University

Hierarchical Feature Learning



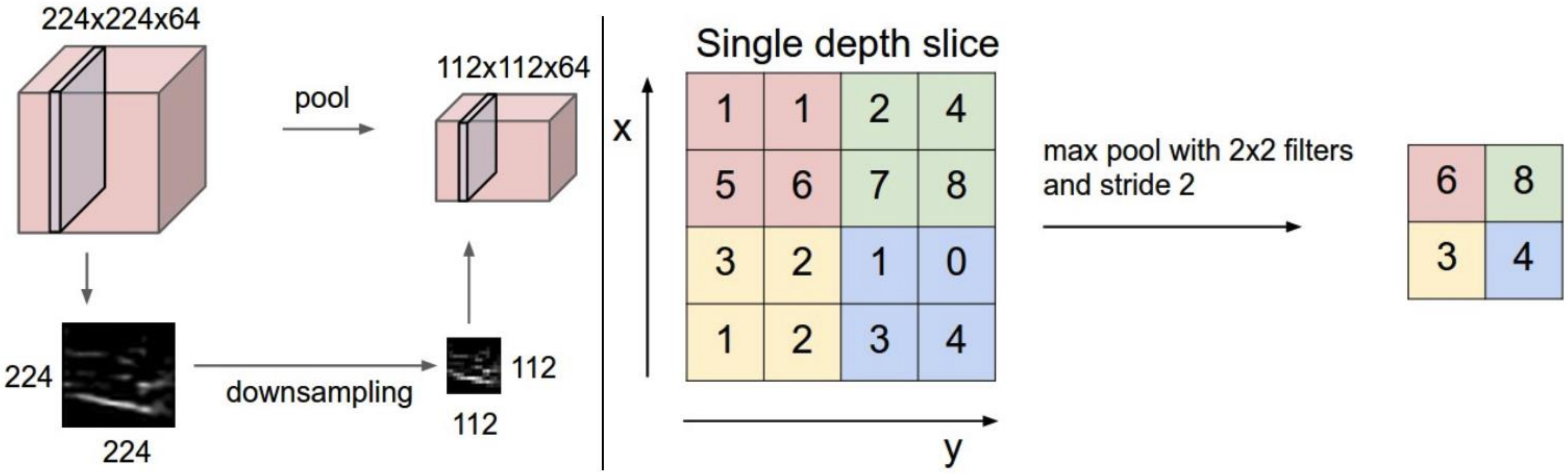
CS231n: Lecture 5, Stanford University

Activation Functions for Convolutional Layers

ReLU	Leaky ReLU	ELU
$g(z) = \max(0, z)$	$g(z) = \max(\epsilon z, z)$ with $\epsilon \ll 1$	$g(z) = \max(\alpha(e^z - 1), z)$ with $\alpha \ll 1$
		
• Non-linearity complexities biologically interpretable	• Addresses dying ReLU issue for negative values	• Differentiable everywhere

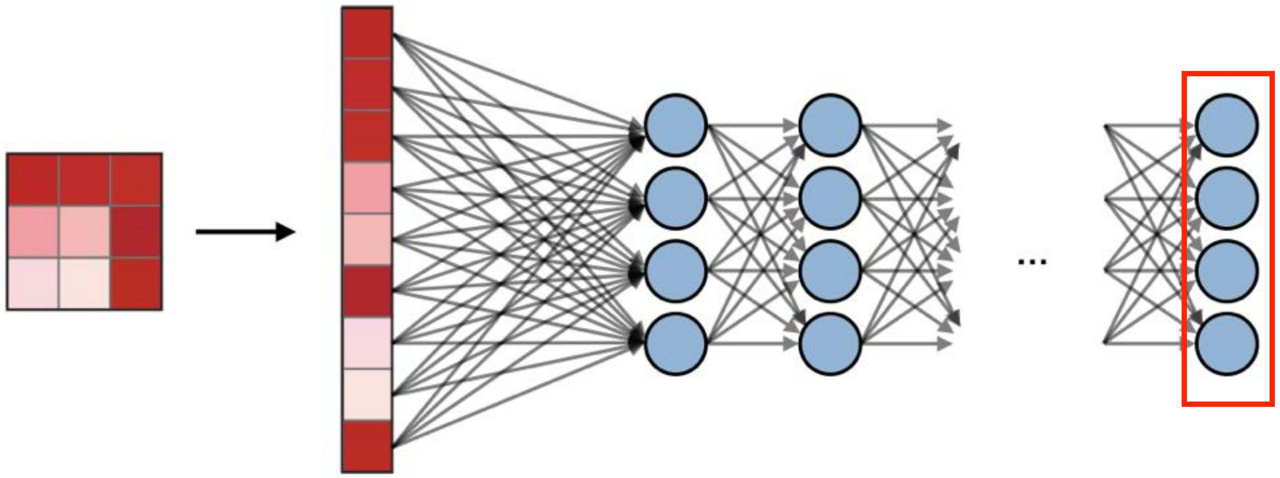
https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-convolutional-neural-networks

Pooling Layer (Max or Average)



https://cs231n.github.io/convolutional-networks/

Fully-connected Layer

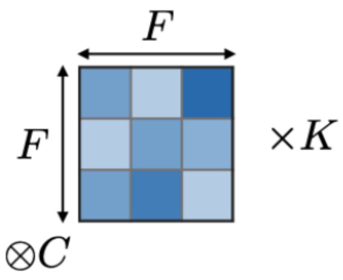
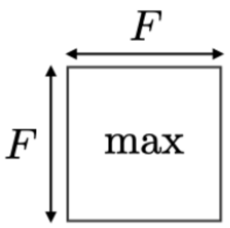
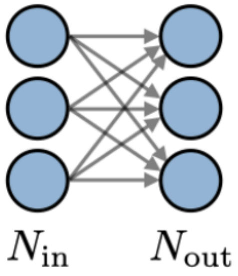


# of neurons =
# of categories

softmax activation

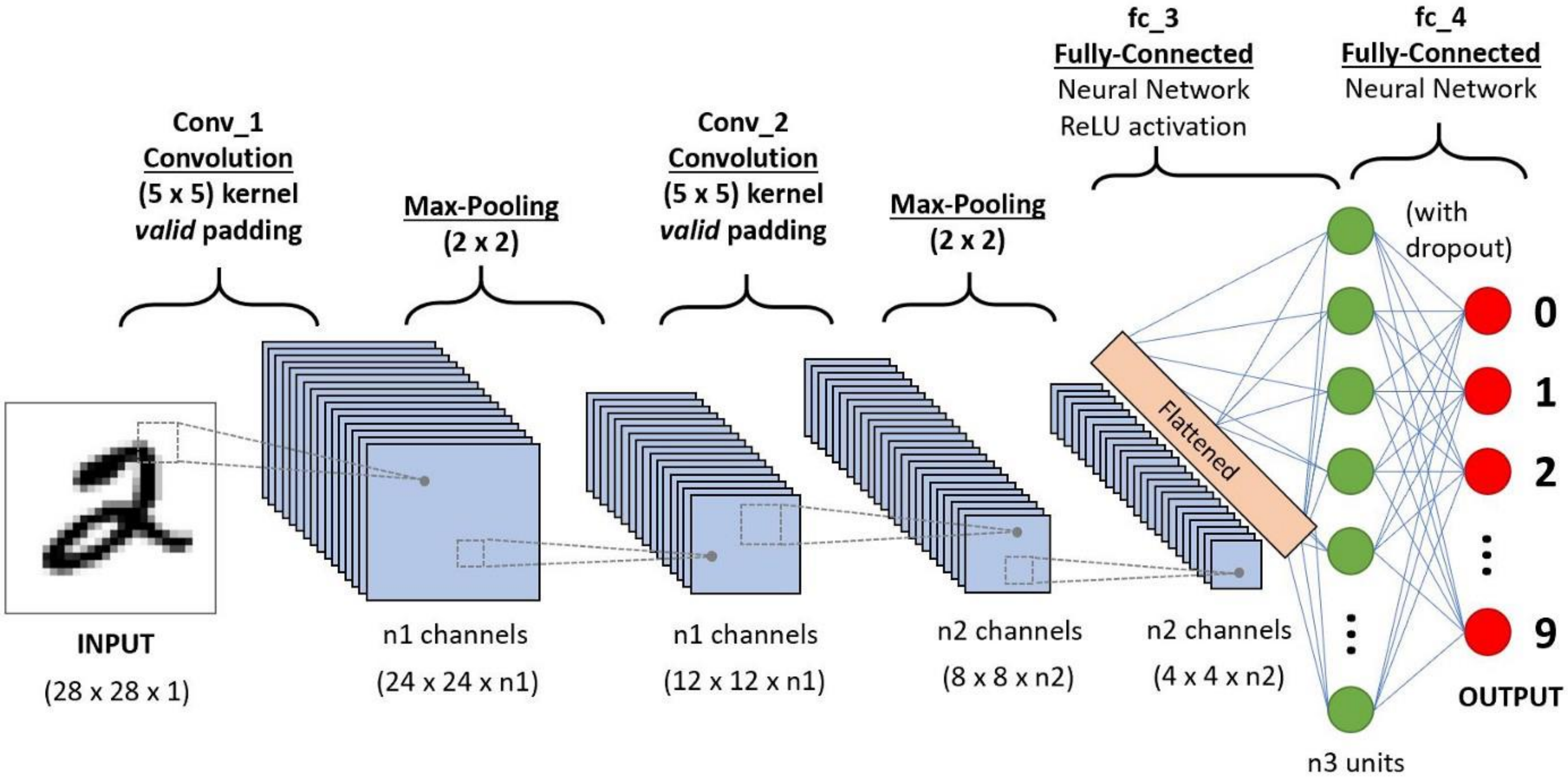
https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-convolutional-neural-networks

Summary for Layer Sizes

	CONV	POOL	FC
Illustration			
Input size	$I \times I \times C$	$I \times I \times C$	N_{in}
Output size	$O \times O \times K$	$O \times O \times C$	N_{out}
Number of parameters	$(F \times F \times C + 1) \cdot K$	0	$(N_{\text{in}} + 1) \times N_{\text{out}}$
Remarks	• One bias parameter per filter • In most cases, $S < F$ • A common choice for K is $2C$	• Pooling operation done channel-wise • In most cases, $S = F$	• Input is flattened • One bias parameter per neuron • The number of FC neurons is free of structural constraints

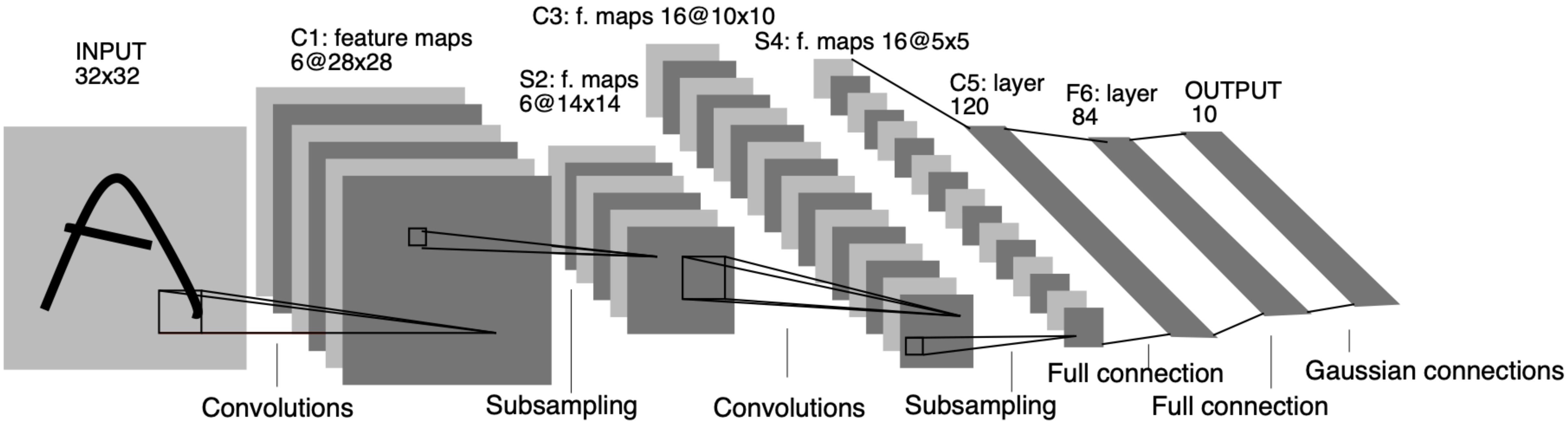
https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-convolutional-neural-networks

Example of a Simple Convolutional Neural Network



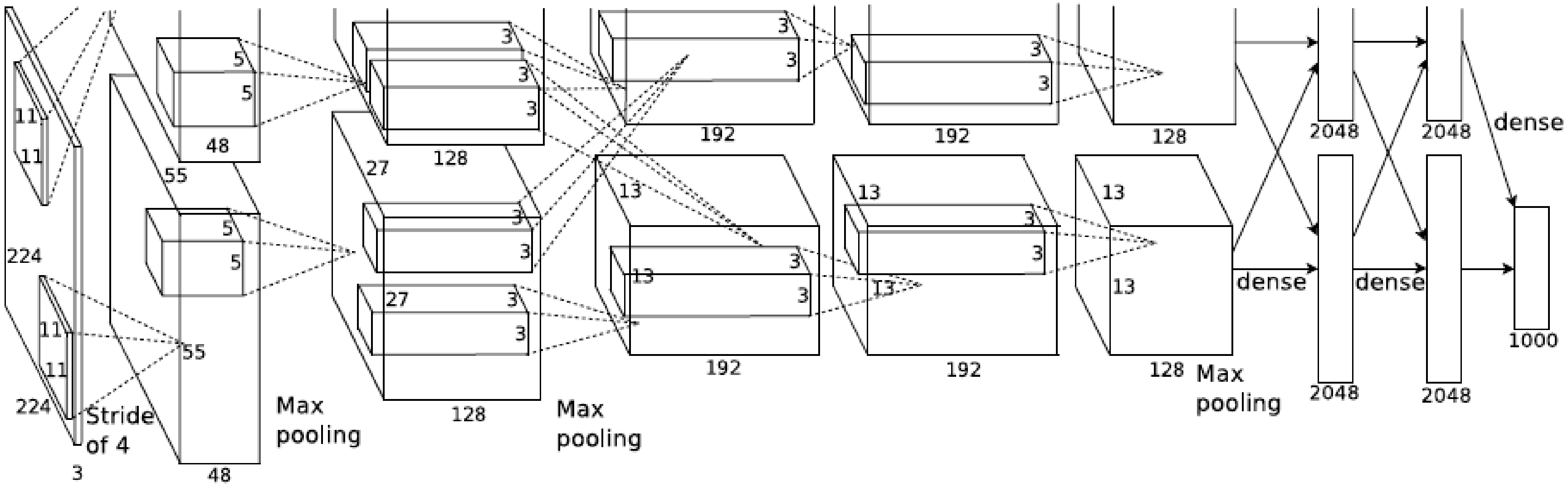
https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53

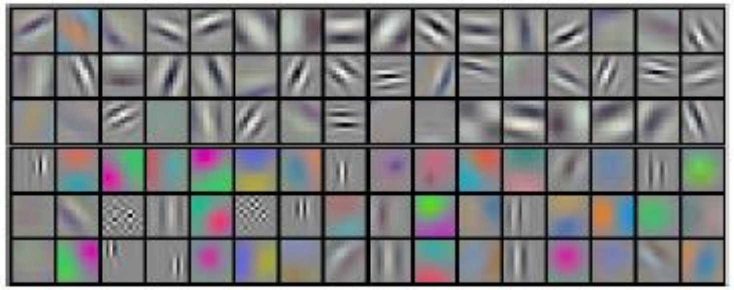
Popular CNN Models: LeNet



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

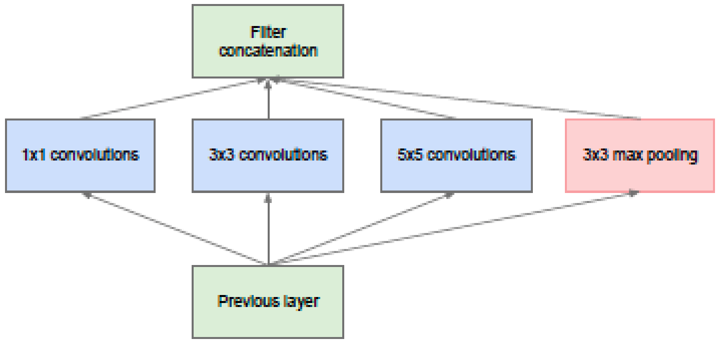
Popular CNN Models: AlexNet



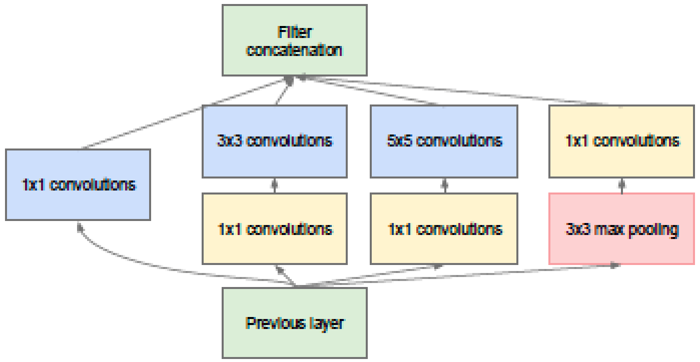


Krizhevsky et al., ImageNet Classification with Deep Convolutional Neural Networks, 2012

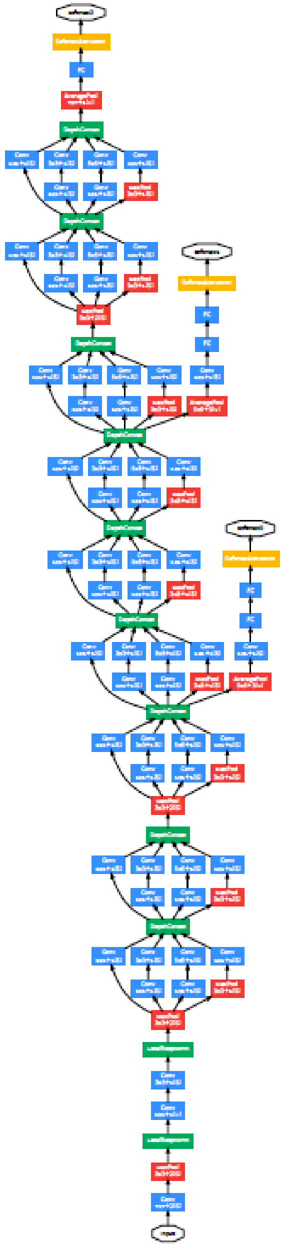
Popular CNN Models: GoogLeNet (Inception)



(a) Inception module, naïve version

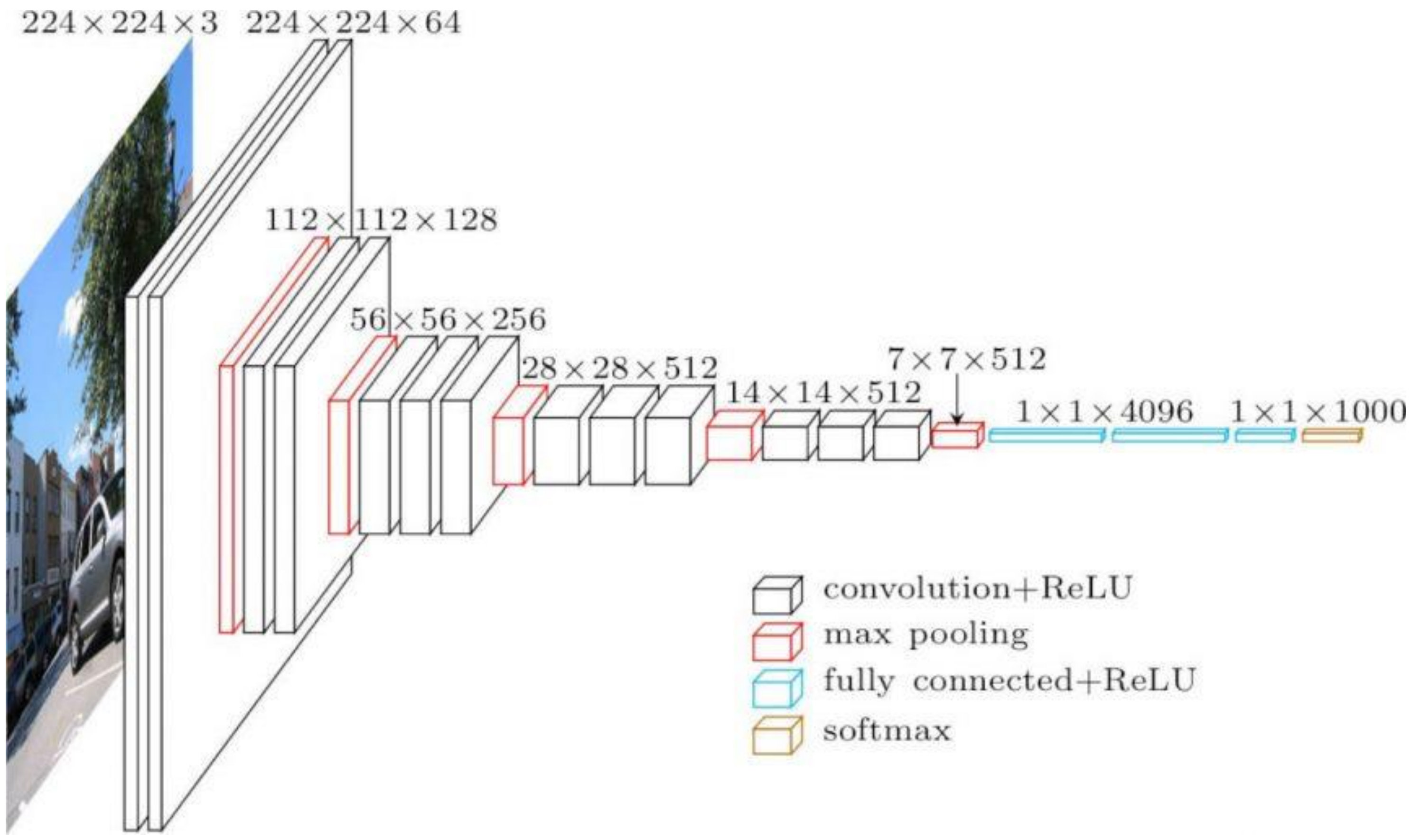


(b) Inception module with dimension reductions



Szegedy et al., Going deeper with convolutions, 2014

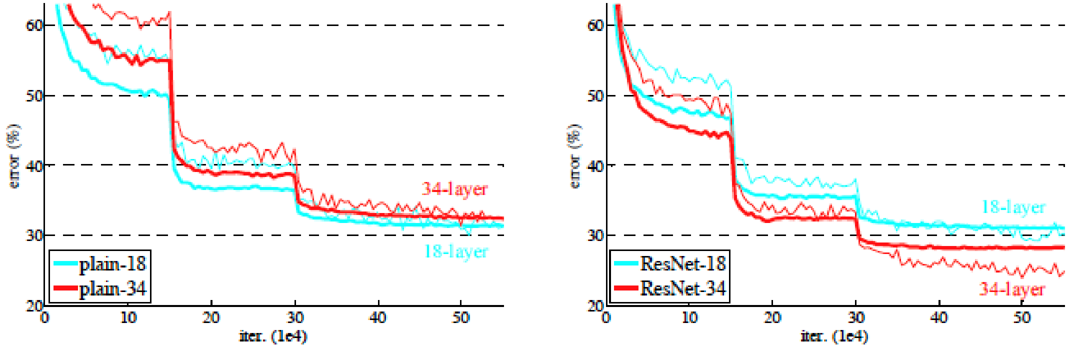
Popular CNN Models: VGGNet



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

Image from https://neurohive.io/en/popular-networks/vgg16/

Popular CNN Models: ResNet



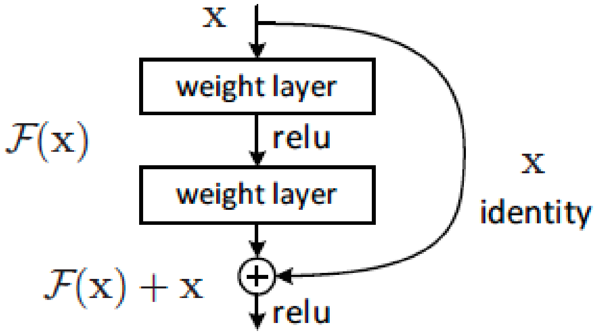
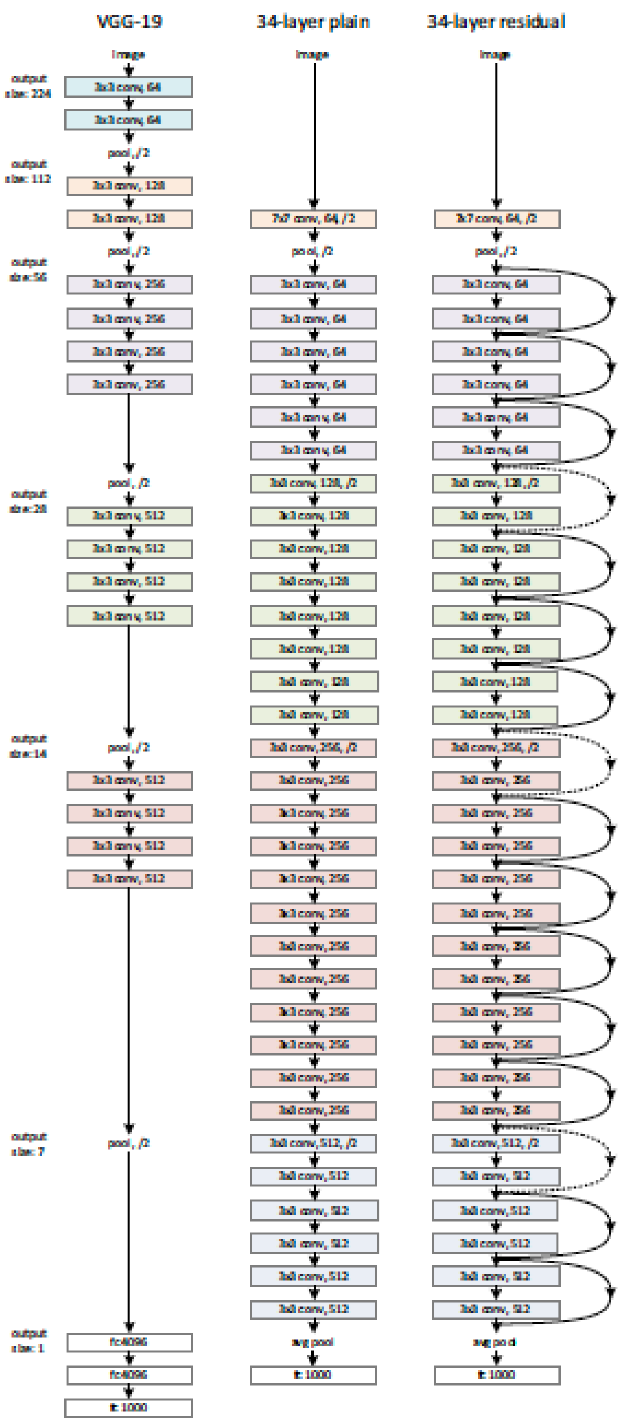


Figure 2. Residual learning: a building block.



He et al.,Deep Residual Learning for Image Recognition, 2015