Computer Vision with Convolutional Neural Networks

Roman Naumov

MADS student at the University of Canterbury

Christchurch Data Science Meetup

New Zealand

09.04.2024

Contents

1. Introduction into Computer Vision (CV)
2. Examples of CV tasks
3. History of Convolutional Neural Networks (CNN)
4. Examples of CNN tasks
5. The technical side of CNN
6. Real-world applications of CV

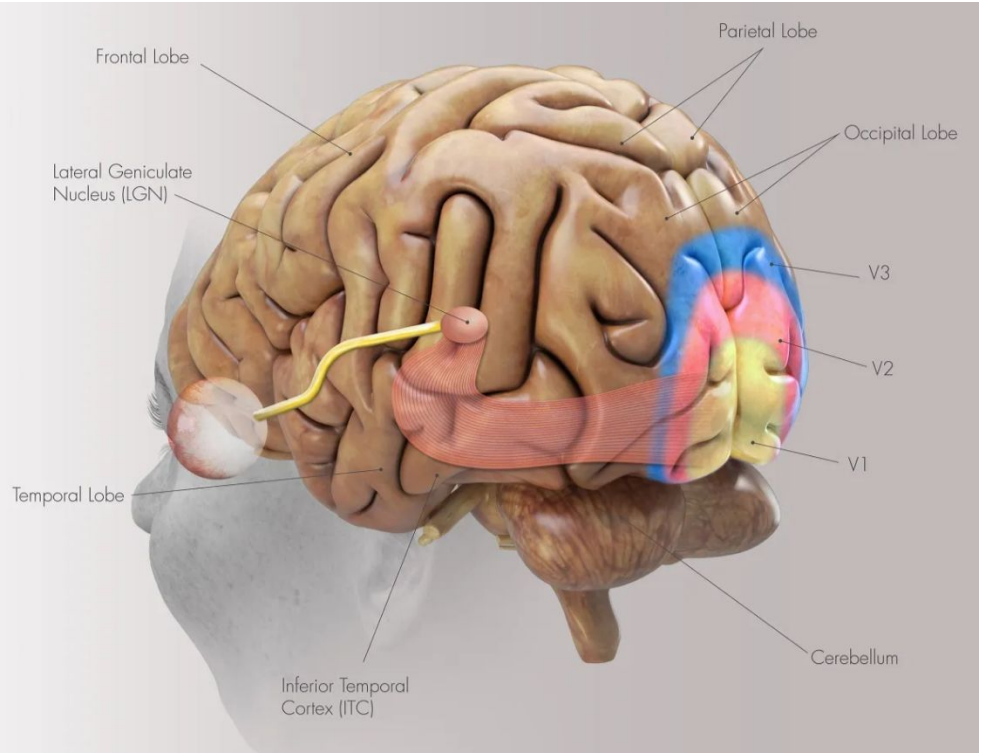
Computer Vision (CV)

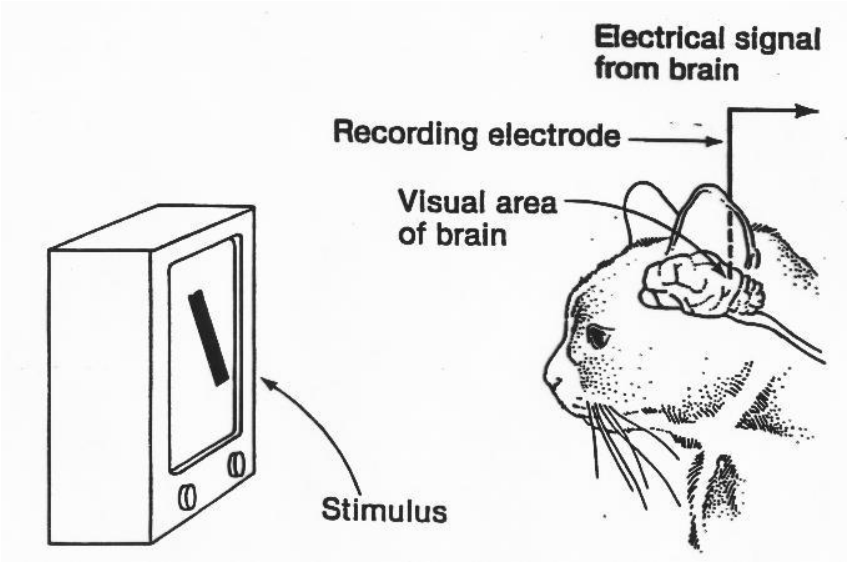
- Understand digital images or videos.
- Automate tasks that the human visual system can perform.

Examples of CV tasks:

1. Acquiring
   a. Sensors
   b. Perception
2. Processing
   a. Compression/Decompression
   b. Denoising
   c. Transformations
3. Analysis
   a. Object recognition
   b. Object tracking
   c. Image segmentation
   d. Pose estimation

Human visual system

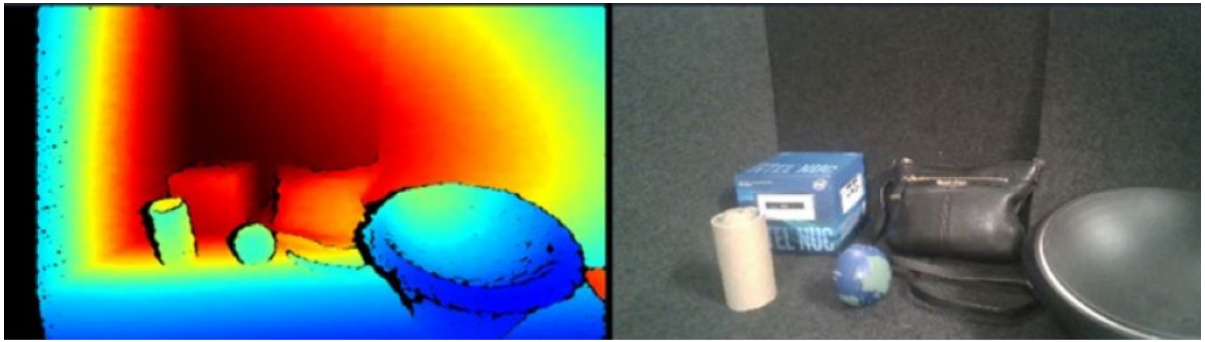




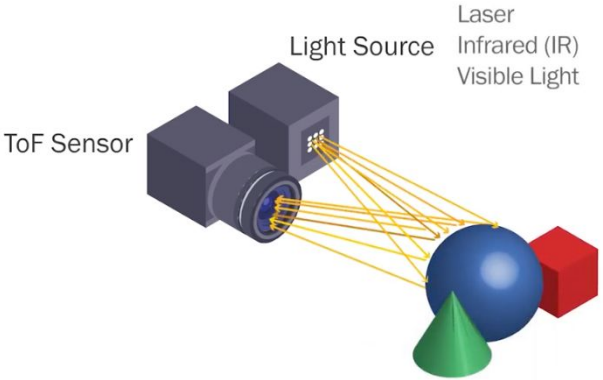
Hubel, D. H., & Wiesel, T. N. (1959). Receptive fields of single neurons in the cat's striate cortex. The Journal of Physiology, 148(3), 574–591. https://doi.org/10.1113/jphysiol.1959.sp006308

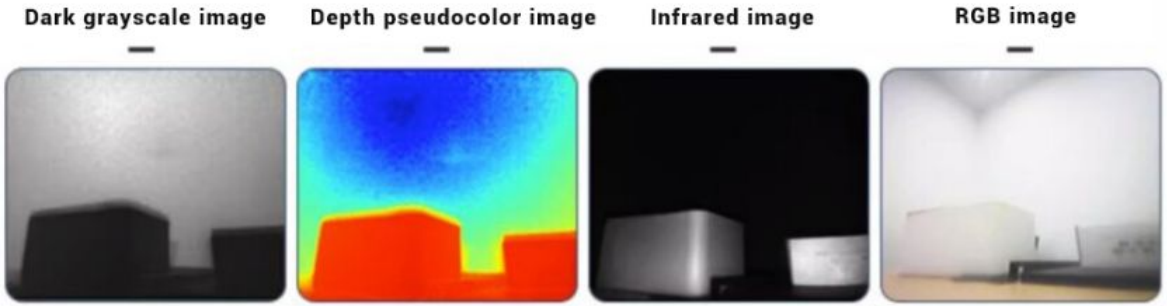
Image acquiring

Intel RealSense Depth 435



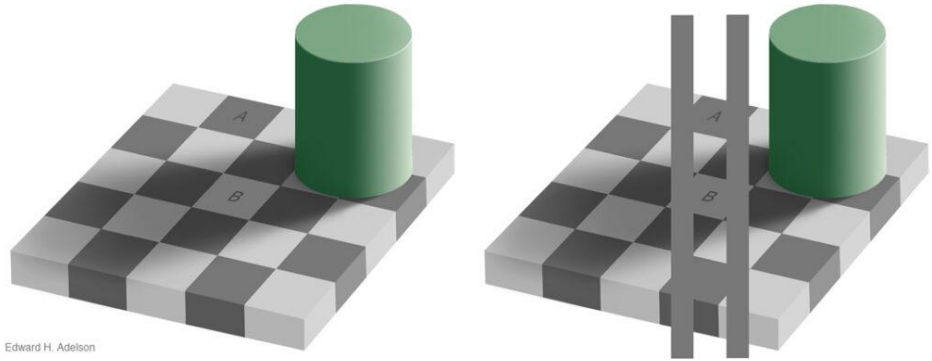
Time of Flight



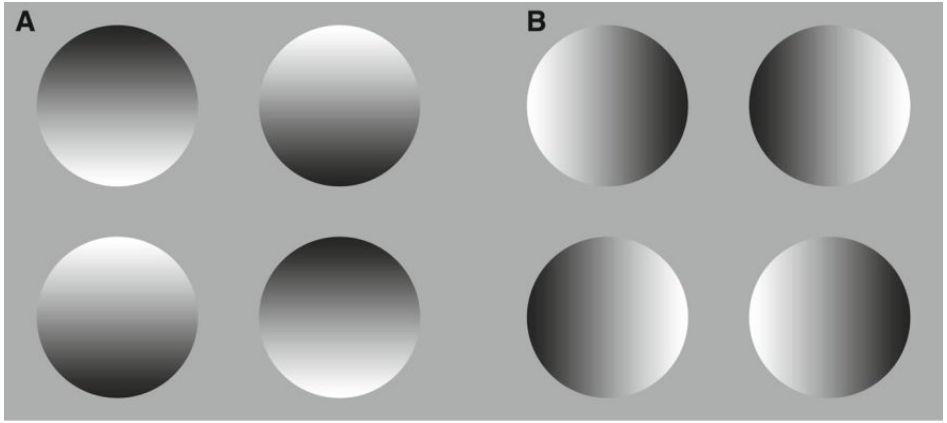


Time-of-Flight (ToF) technology of the RGB-D camera

Image perception



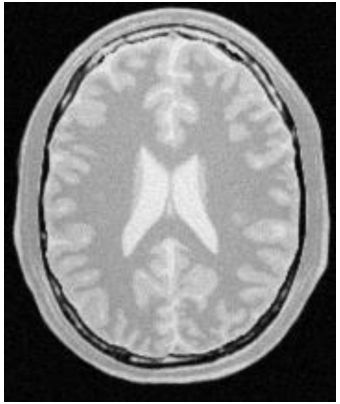
The squares marked A and B are the same shade of gray.

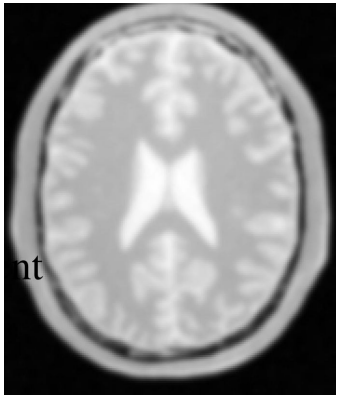


Circular contours filled with A horizontally and B vertically aligned linear contrast gradients.

Image processing

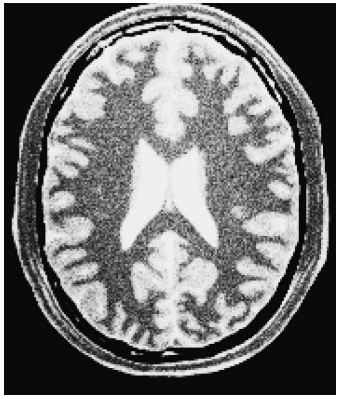
Image compression

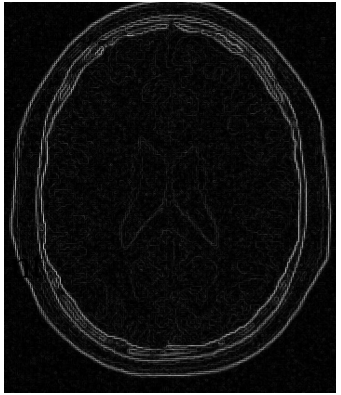




Noise reduction

Contrast enhancement





Edge extraction

Segmentation, Thresholding, Morphology, Image restoration, etc.

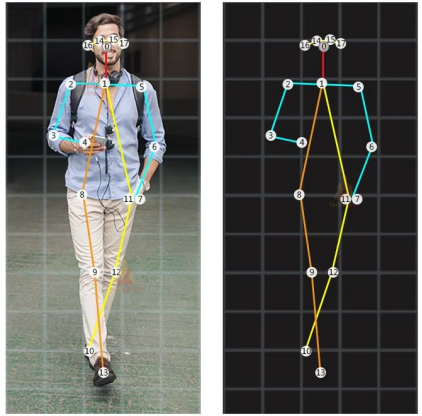
Image analysis

Image classification

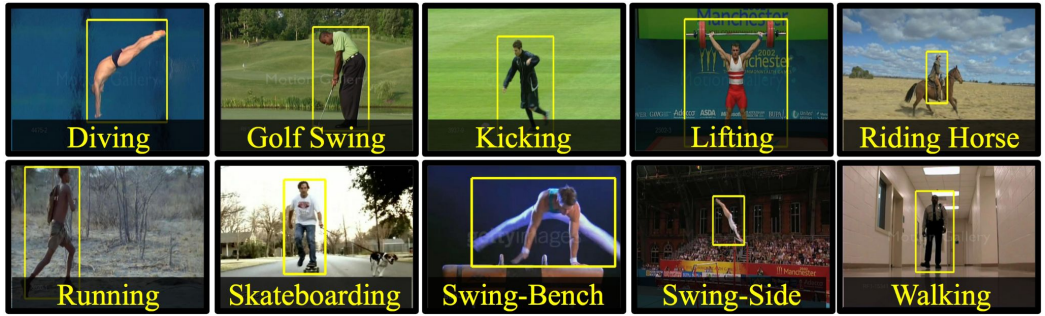
CAT

Object tracking

Pose estimation



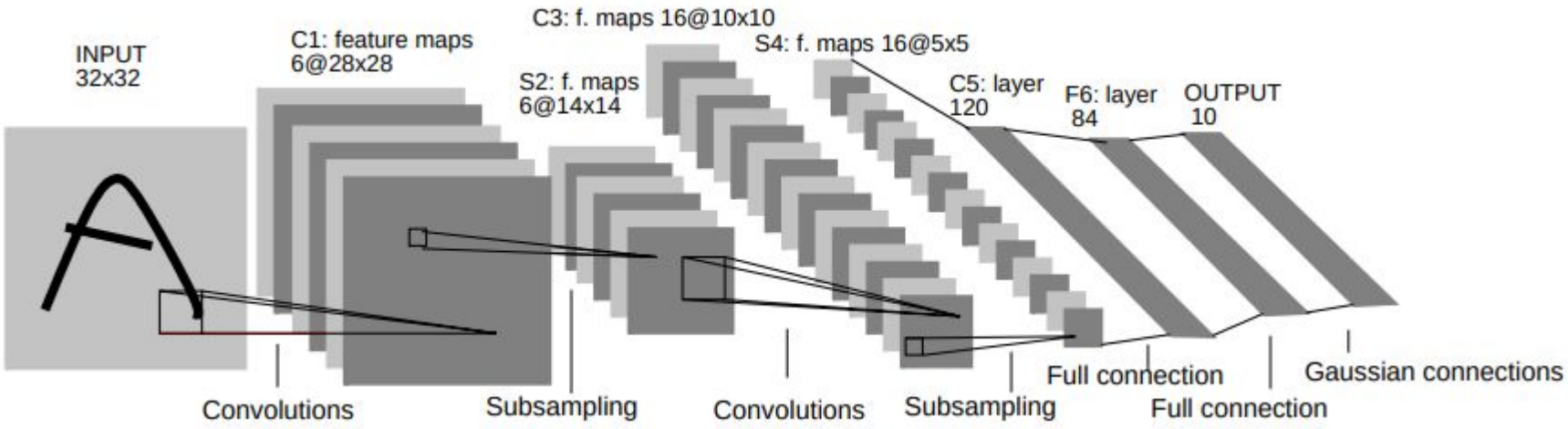
Action recognition



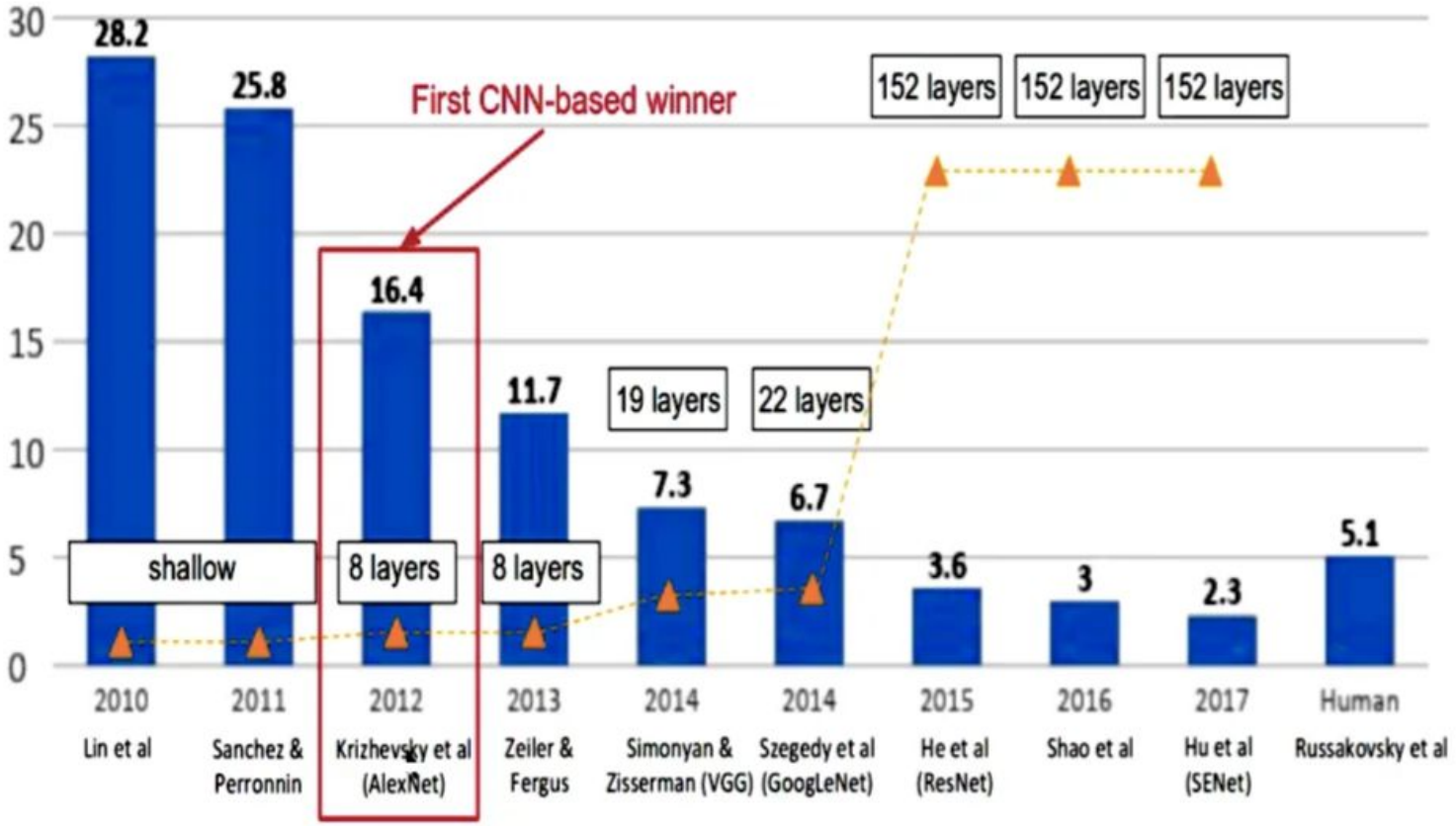
Convolutional Neural Networks (CNN)

The term “convolutional neural networks” appeared in connection with the development of the LeNet neural network by a team led by Yann LeCun (LeCun, Y., Bottou, L., Bengio, Y., & Ha, P. (1998). Gradient-Based Learning Applied to Document Recognition.)

The network architecture was developed between 1989 and 1998 for the task of handwritten digit recognition.

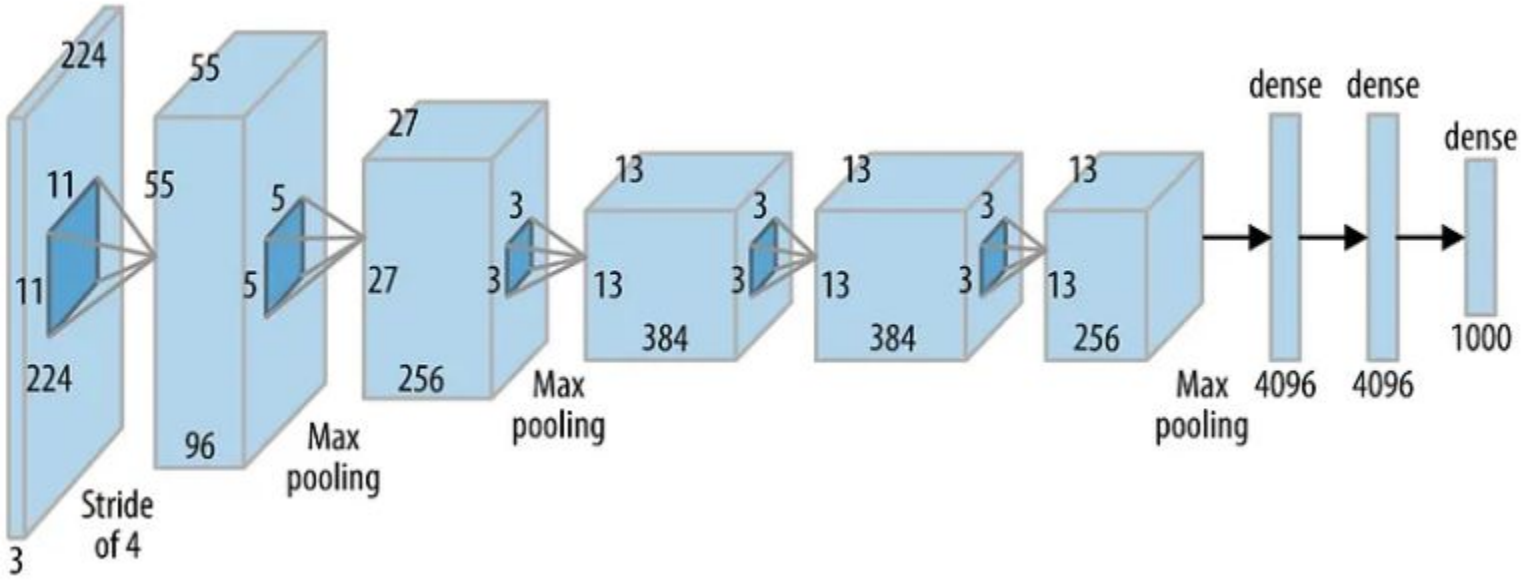


History of CNNs



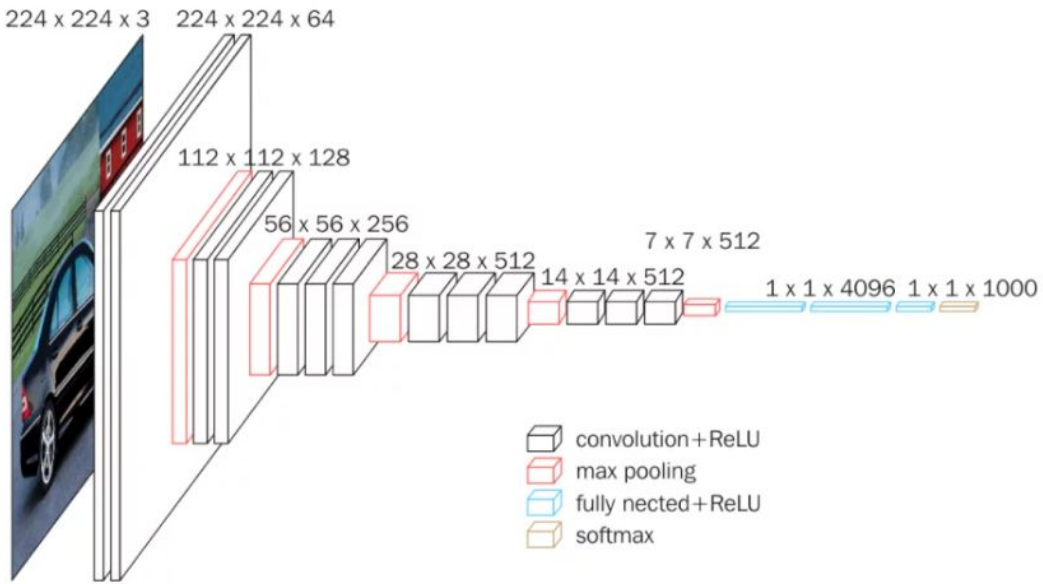
AlexNet, 2012

- AlexNet was born out of the need to improve the results of the **ImageNet** challenge.
- Accuracy of 84.7% - winner of ILSVRC in 2012.
- 5 Convolutional (CONV) layers and 3 Fully Connected (FC) layers. Activation function - Rectified Linear Unit (ReLU).
- The paper - [Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. Advances in Neural Information Processing Systems, 25.]



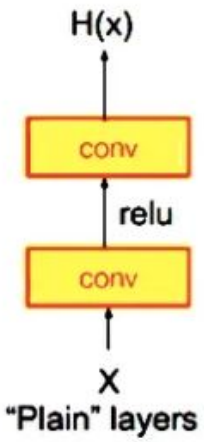
VGG-16, 2014

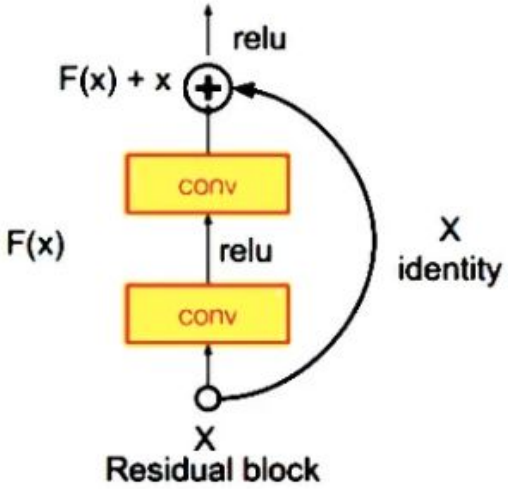
- Accuracy of 92.7%.
- Fixed-size smaller filters.
- 13 Convolutional layers and 3 Fully Connected (FC) layers.
- The paper - [Simonyan, K., & Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition.]

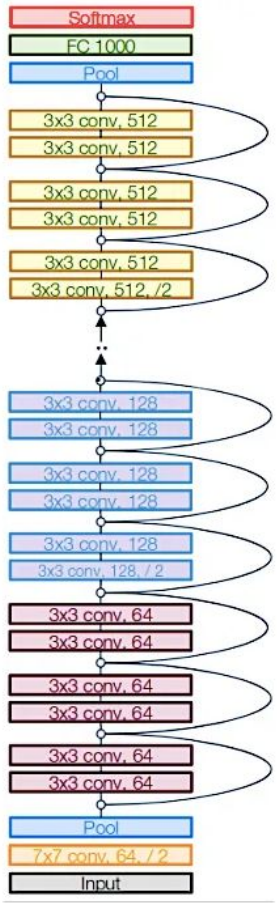


ResNet-152, 2015

- Accuracy of 96.4% - winner of ILSVRC in 2015.
- ResNet architecture makes use of shortcut connections to solve the vanishing gradient problem.
- 152 layers.
- The paper - [He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep Residual Learning for Image Recognition.]







Examples of CNN tasks

Is this a dog?

Image Classification

What is there in image and where?

Object Detection

Which pixels belong to which object?

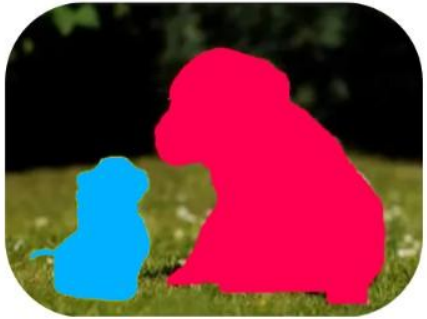


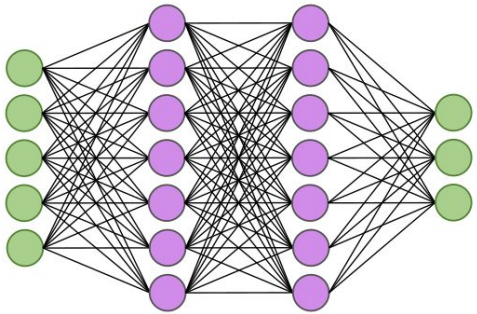
Image Segmentation

Why not use fully connected neural networks?

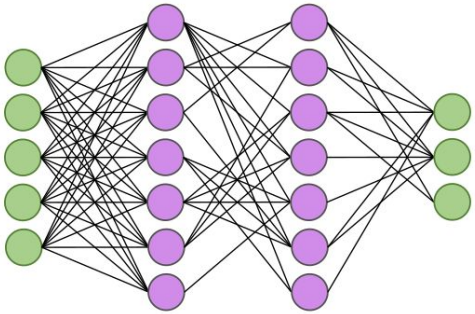
Disadvantages of fully connected NN when working with images:

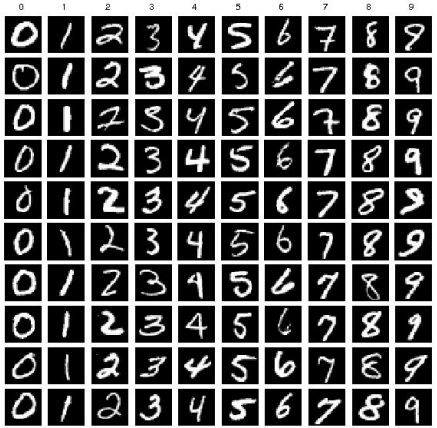
- Translational Invariance
- Spatial dependencies are broken
- Too many connections

Fully Connected



Partially Connected?

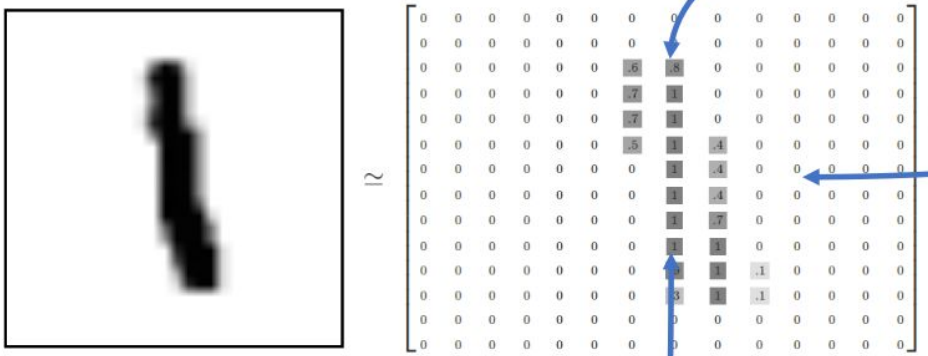




Well-trained classifier

Suppose we've got a well-trained MNIST classifier...

this pixel gets weight 0.6



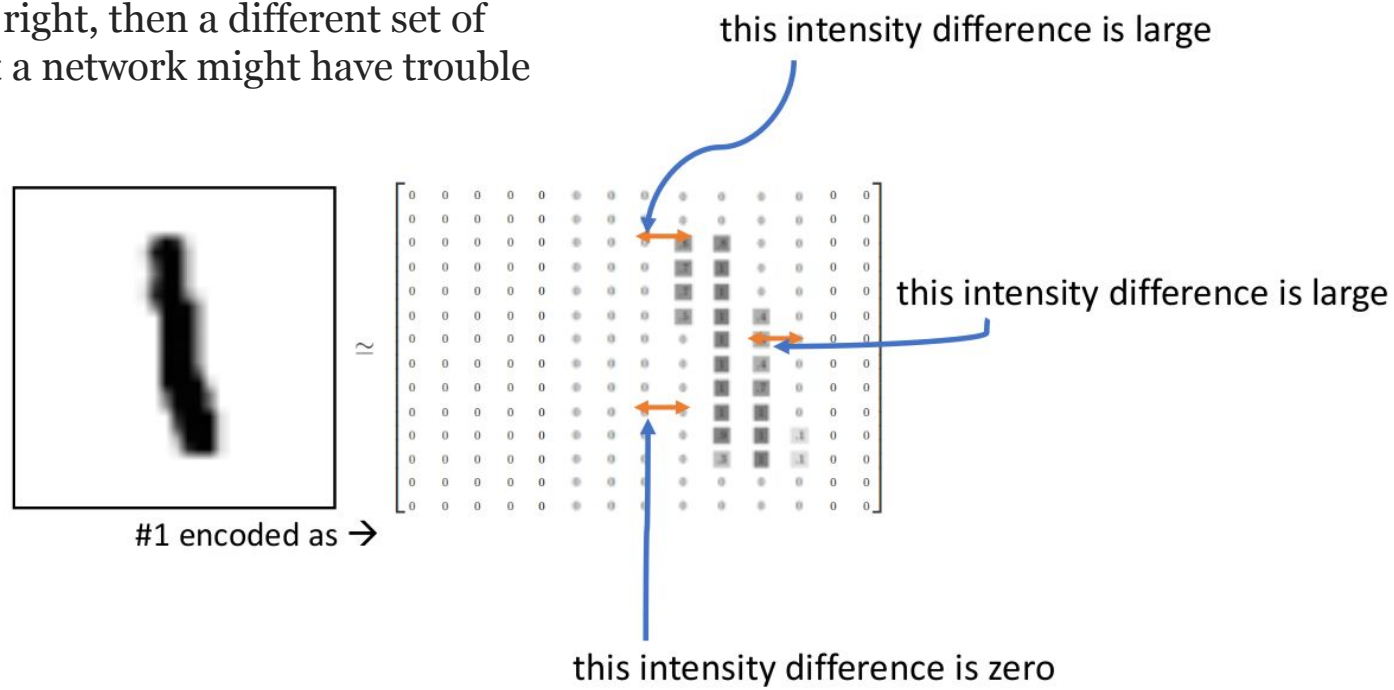
this pixel gets weight 0.1

#1 encoded as →

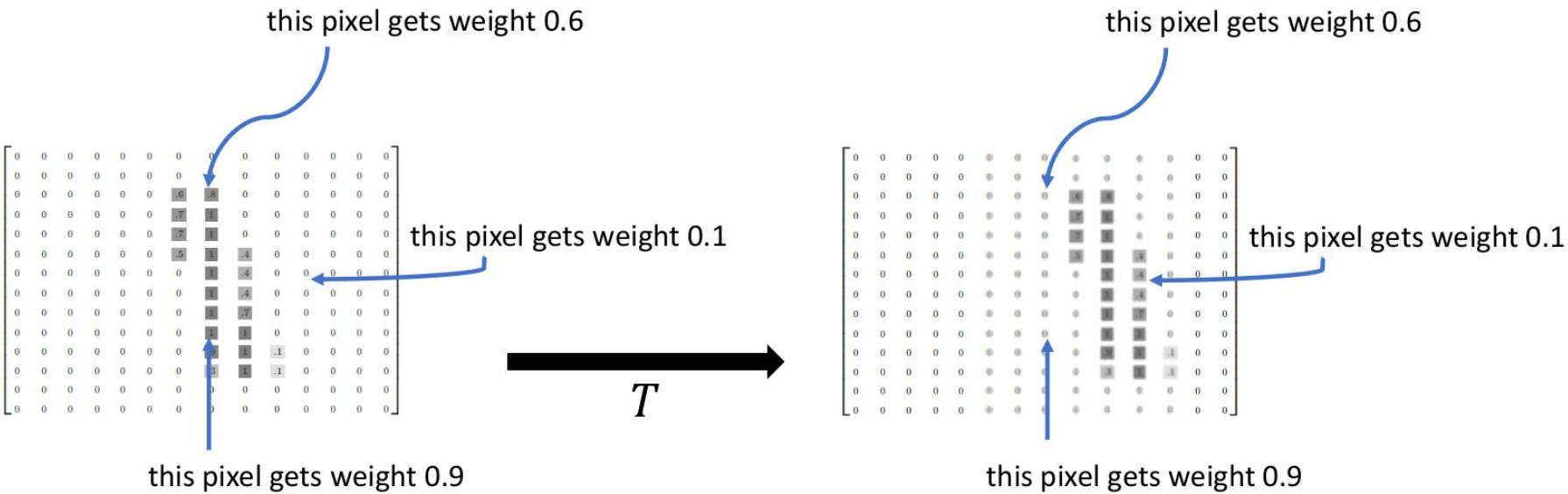
this pixel gets weight 0.9

This is not a problem for the human visual system.

If we shift the digit to the right, then a different set of weights becomes relevant a network might have trouble classifying this as a 1.



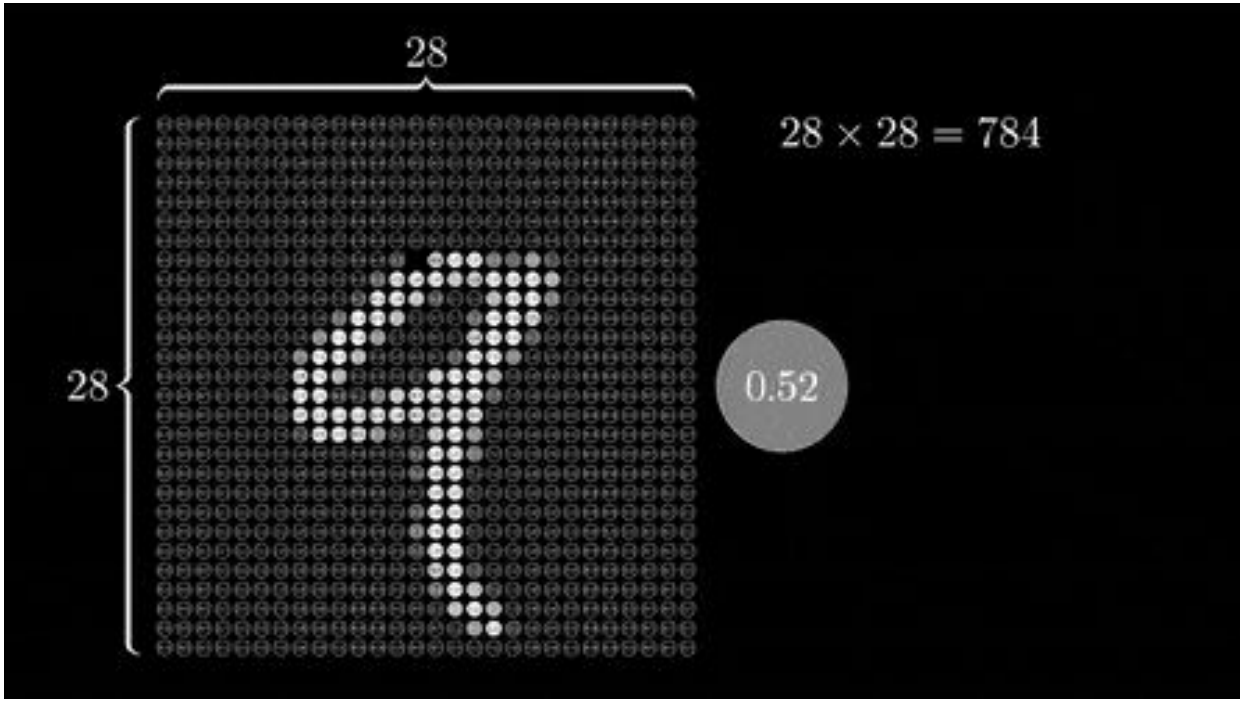
Translational Invariance



Sum of these three: $0.6 \cdot 0.8 + 0.1 \cdot 0 + 0.9 \cdot 1 = 1.38$

Sum of these three: $0.6 \cdot 0 + 0.1 \cdot 0.4 + 0.9 \cdot 0 = 0.4$

Spatial dependencies are broken



Pixels aren't independent, nearby pixels are correlated and areas in the image have local connectivity.

When we convert a matrix to a vector, we break pixel dependencies.

Too many connections

There are two problems:

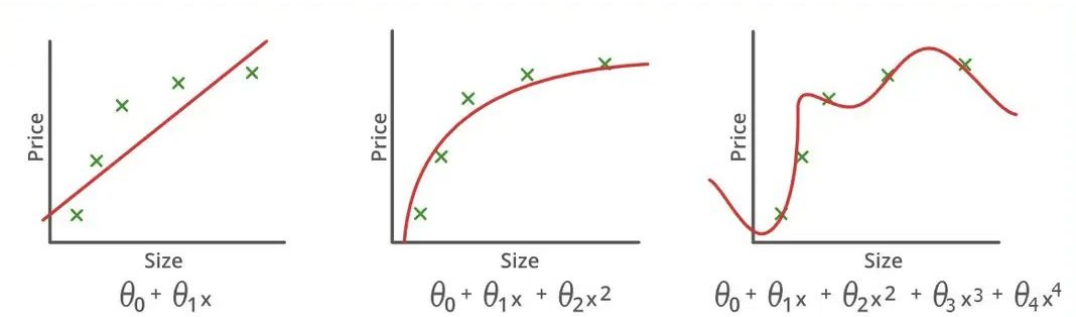
1. Computing resources.
2. The fully-connected (FC) neural network can memorize the image.

28x28x3 = 2352 inputs

1024x1024x3 = 3,145,728 inputs

The FC neural network can memorize the image

Underfitting Goodfitting Overfitting



x1	x2	x3	...	xN	y
2108	0	0.445	...	0.019	1
1823	1	0.474	...	0.022	0
1608	1	0.477	...	0.023	1
1370	1	0.448	...	0.024	1
1170	1	0.435	...	0.025	1
1021	1	0.44	...	0.027	0
896	1	0.433	...	0.028	0
780	1	0.439	...	0.031	1
665	1	0.458	...	0.035	1
568	1	0.448	...	0.037	0

Feature Engineering

stepAIC CNN

What features are there in the image?

What is the difference between a four and an eight?

The number 4 has horizontal and vertical lines, while the number 8 has curved lines.

Can we build networks that are:

1. **Translationally invariant**
2. **Don't overfit**
3. **Can extract features from images?**

Main building blocks for convolution

Convolution is an operation that takes two inputs:

An image

A filter (also called a kernel)

1	1	1
0	0	0
-1	-1	-1

How does convolution work?

image

2	0	1	3
7	1	1	0
0	2	5	0
0	5	1	4

$\otimes$

filter/kernel

1	1	1
0	0	0
−1	−1	−1

(We use this symbol for convolution)
(The verb form is “convolve”)

How does convolution work?

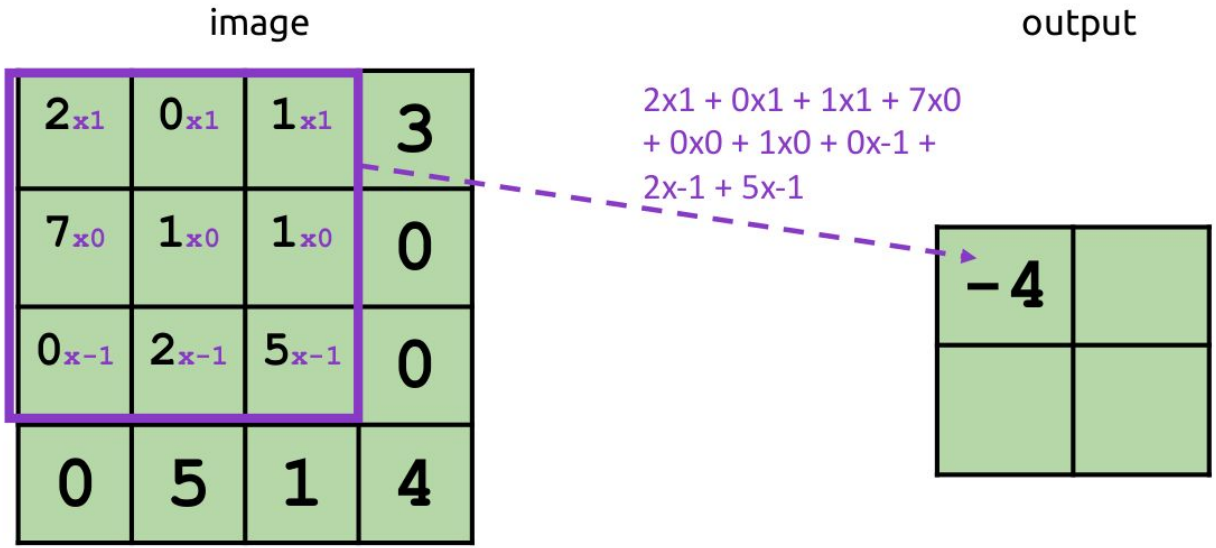
Overlay the filter on the image

image

1	1	1	3
0	0	0	0
-1	-1	-1	0
0	5	1	4

How does convolution work?

Sum up multiplied values to produce output value



How does convolution work?

Move the filter over by one pixel

image

1	1	1	3
0	0	0	0
-1	-1	-1	0
0	5	1	4

output

-4	

How does convolution work?

Move the filter over by one pixel

image

2	1	1	1
7	0	0	0
0	-1	-1	-1
0	5	1	4

output

-4	

How does convolution work?

Repeat (multiply, sum up)

image

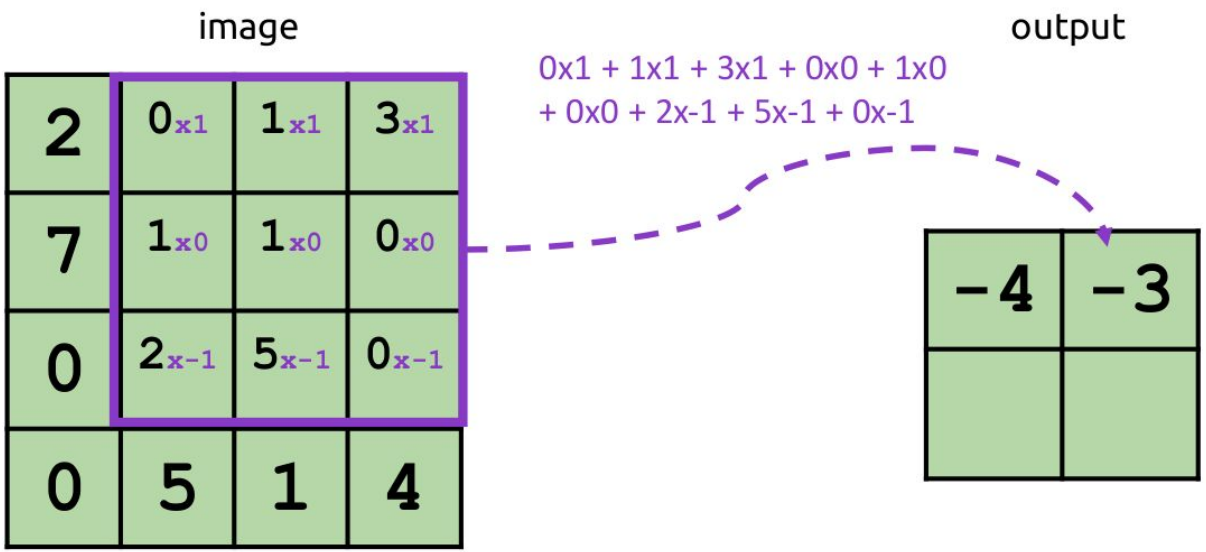
2	0_{x1}	1_{x1}	3_{x1}
7	1_{x0}	1_{x0}	0_{x0}
0	2_{x-1}	5_{x-1}	0_{x-1}
0	5	1	4

output

-4	

How does convolution work?

Repeat (multiply, sum up)



How does convolution work?

Repeat...

image

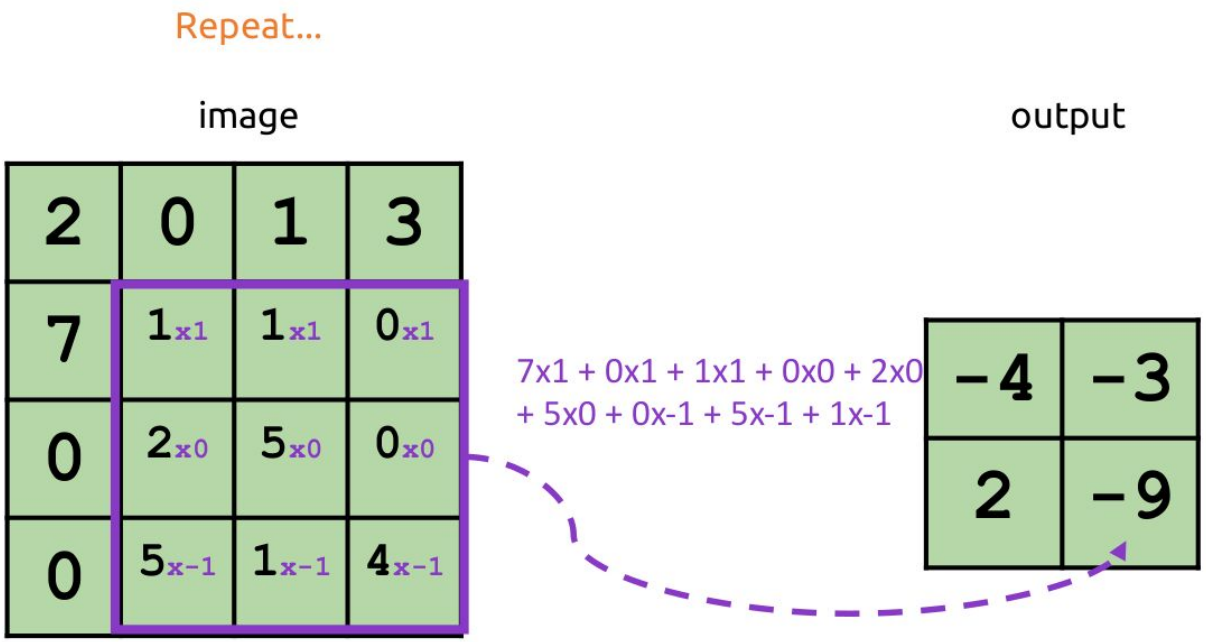
2	0	1	3
7_{x1}	1_{x1}	1_{x1}	0
0_{x0}	2_{x0}	5_{x0}	0
0_{x-1}	5_{x-1}	1_{x-1}	4

7x1 + 0x1 + 1x1 + 0x0 + 2x0
+ 5x0 + 0x-1 + 5x-1 + 1x-1

output

−4	−3
2	

How does convolution work?



How does convolution work?

In summary:

image

2	0	1	3
7	1	1	0
0	2	5	0
0	5	1	4

$\otimes$

filter/kernel

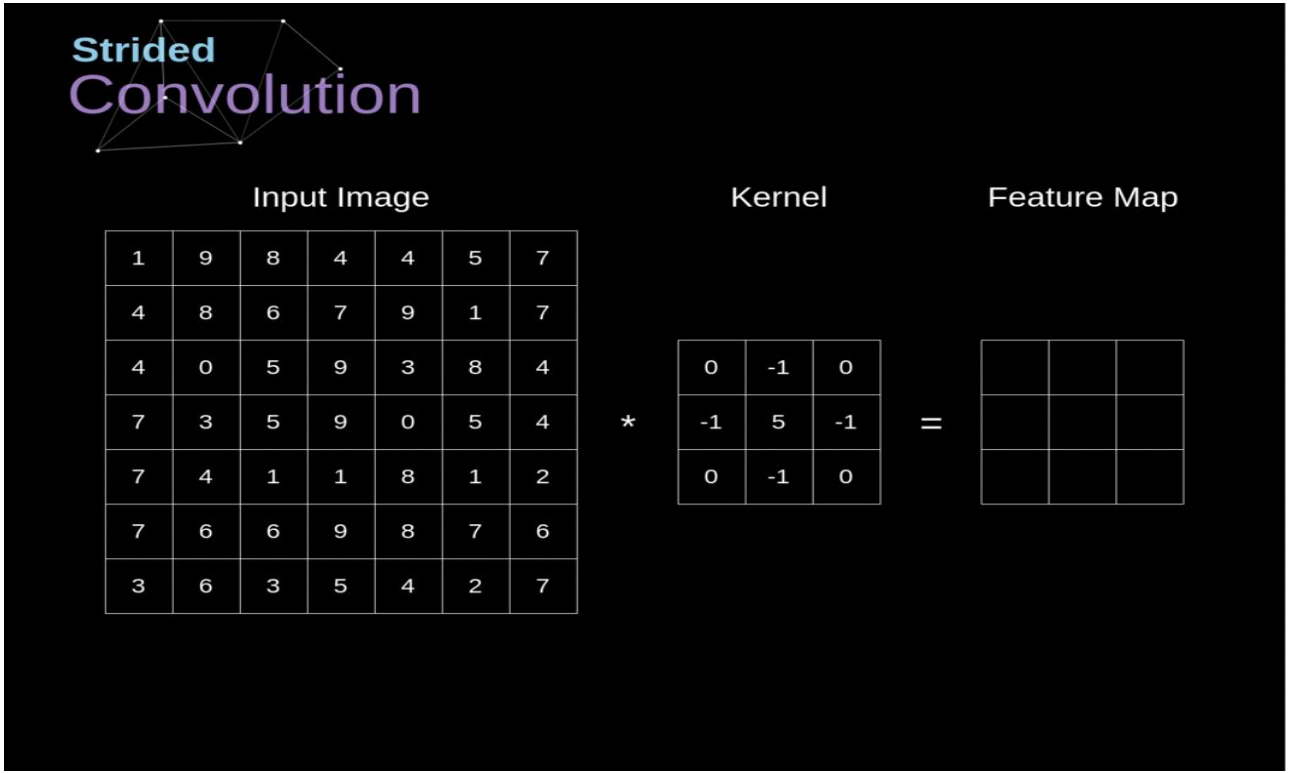
1	1	1
0	0	0
-1	-1	-1

$=$

output

-4	-3
2	-9

How does convolution work?

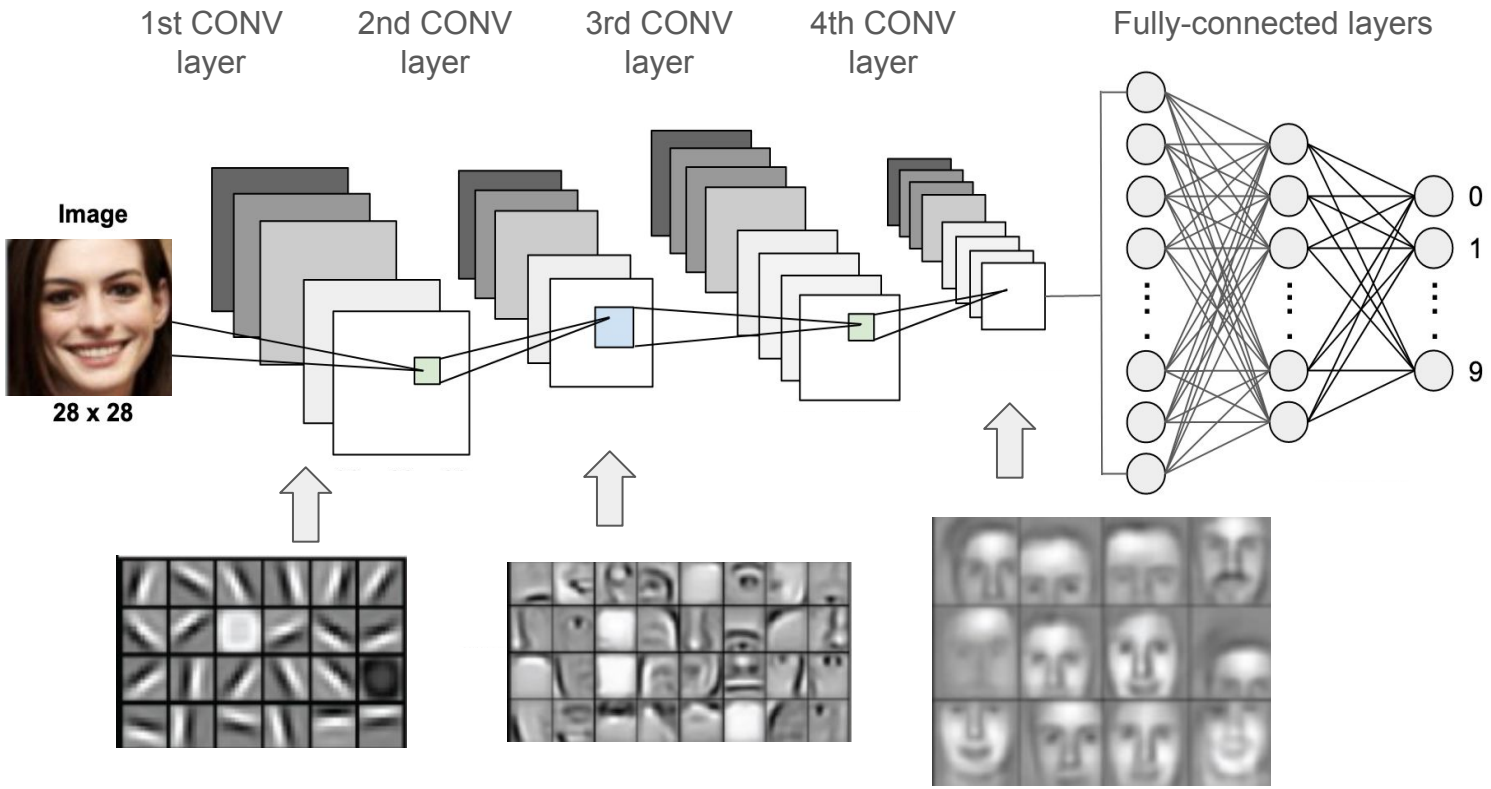


Different kernel can cause a wide range of effects

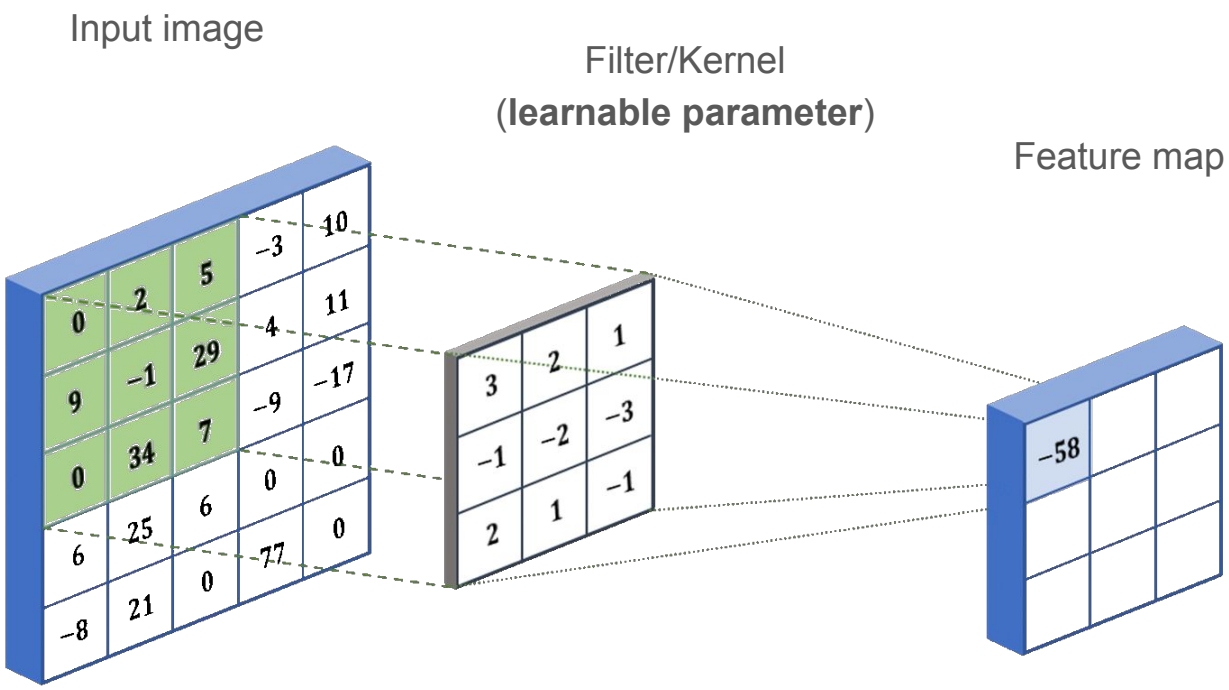
$\otimes$

Edge detection	$\begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix}$	
	$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$	
	$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$	
Sharpen	$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$	
Box blur (normalized)	$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$	

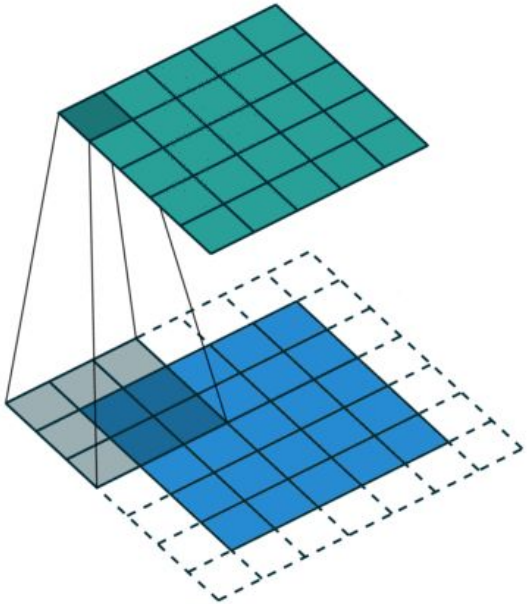
CNN feature maps



How to learn CNN?



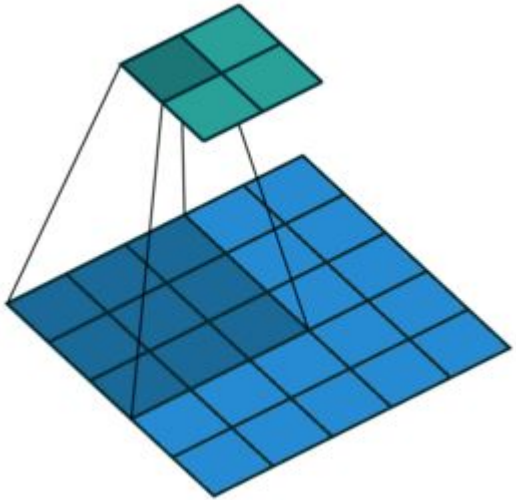
Input, Filter, Stride, Padding



Input = 5x5

Output = 5x5

Stride = 1,
Padding = 1



Input = 5x5

Output = 2x2

Stride = 2,
Padding = 0

Pooling

Input image

Kernel

-1	0	1
-1	0	1
-1	0	1

Feature Map

12	20	30	0
8	12	2	0
34	70	37	4
112	100	25	12

Pooling

12	20	30	0
8	12	2	0
34	70	37	4
112	100	25	12

max pooling

20	30
112	37

average pooling

13	8
79	20

Image -> conv -> act -> pool -> conv ...

Max Pooling

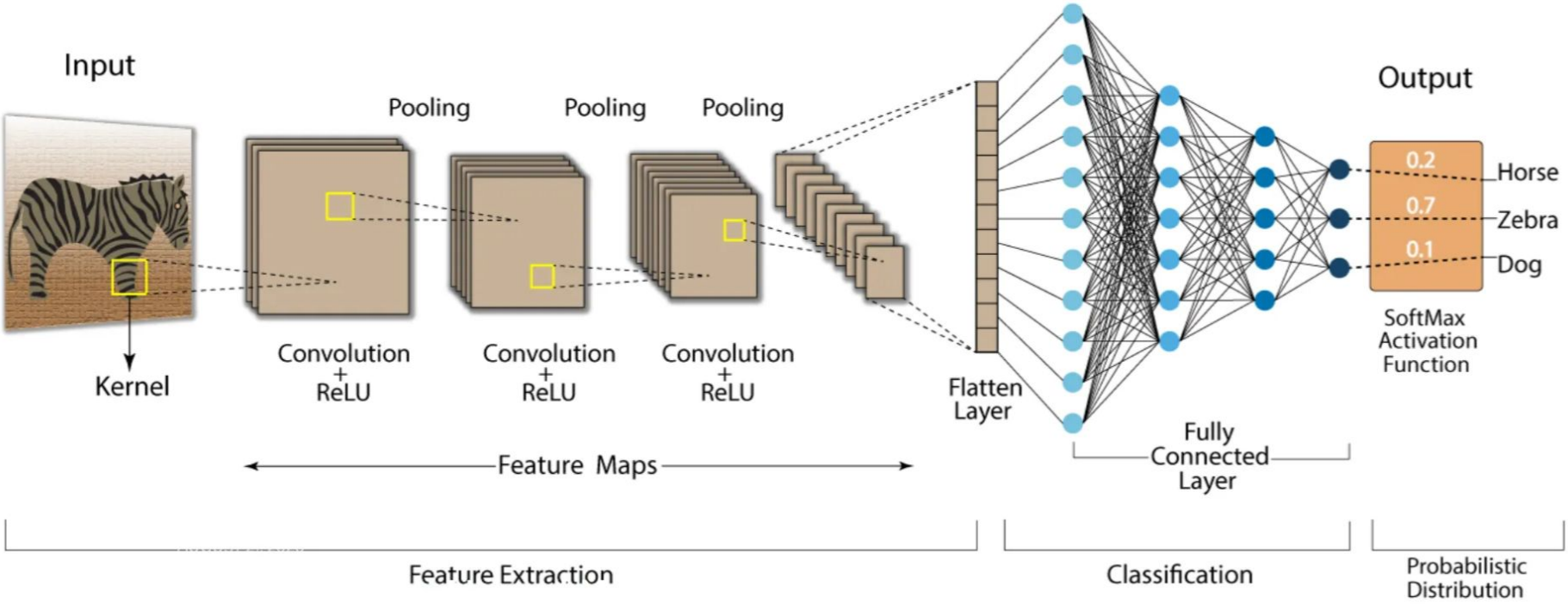
- Keeps track of regions with highest activations, indicating object presence.
- Controllable way to decrease resolution.
- Intended to give invariance to small translations.

Original Image

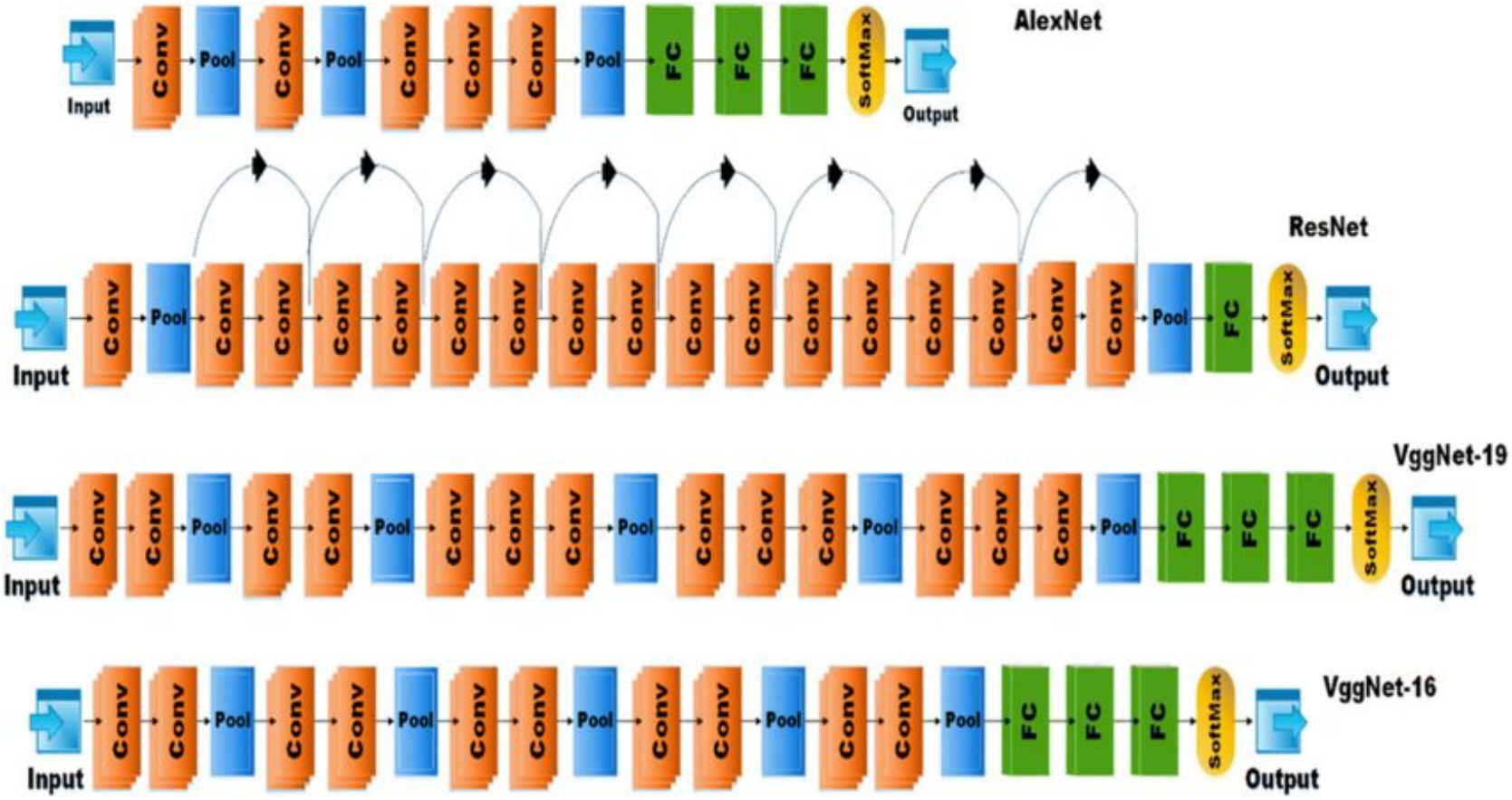
Convolution Output

After Pooling

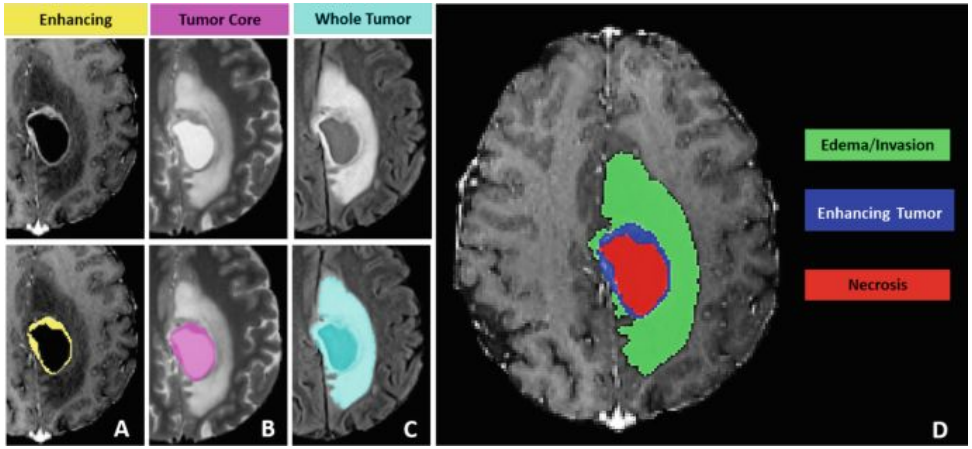
Typical CNN Architecture



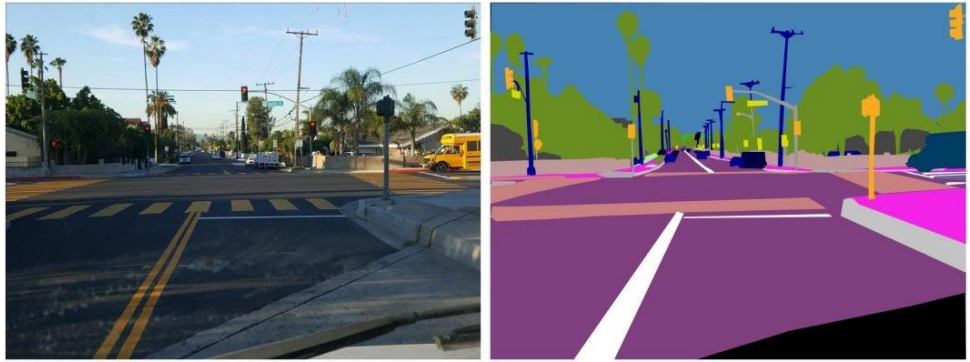
AlexNet, VGG, ResNet



Real-world applications of CV (segmentation)



Medical image segmentation for tumor detection.



Segmentation of street infrastructure and facilities. Used in the development of self-driving cars.

Real-world applications of CV (self-driving cars)

- detection
- classification
- segmentation
- search by images
- car position assessment

Real-world applications of CV (OCR)

Use one of your own files or choose from a sample below.

Drag and drop a file here
or
Browse for a file
or
Take a photo

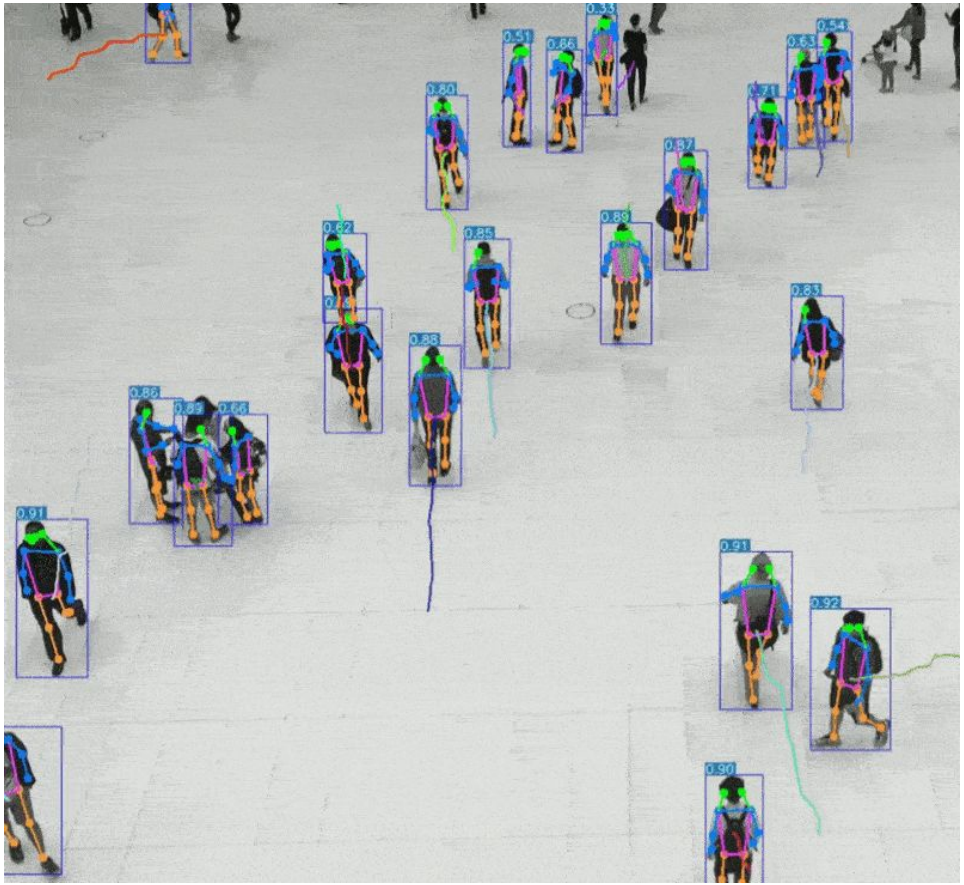
Sample form #3

Detected attributes JSON

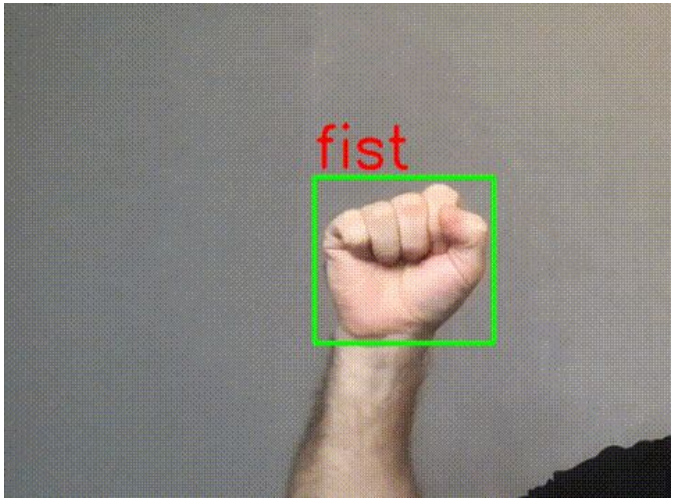
Nutrition Facts Amount Per Serving
Serving size: 1 bar (40g)
Serving Per Package: 4
Total Fat 13g
Saturated Fat 1.5g
Amount Per Serving
Trans Fat 0g
alories 190
Cholesterol 0mg
ories from Fat 110
Sodium 20mg
nt Daily Values are based on
Vitamin A 50%
calorie diet.

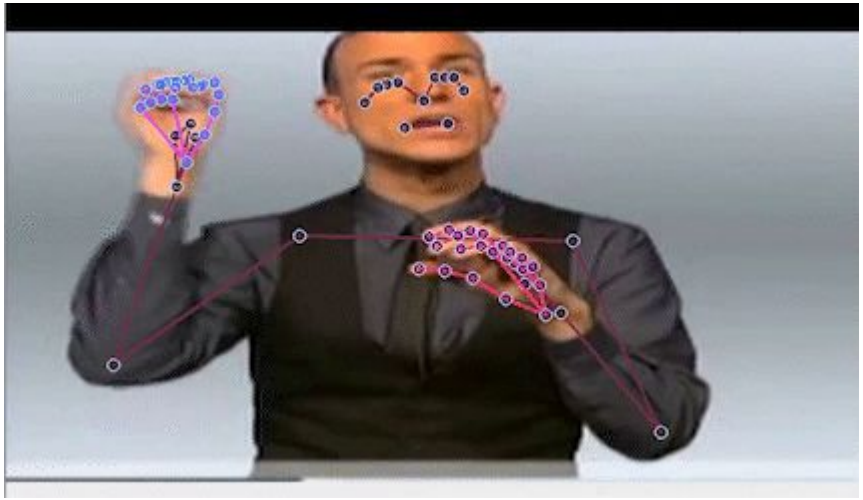
Real-world applications of CV (vehicle counting)

Real-world applications of CV (pose estimation)



Real-world applications of CV (hand gesture recognition)

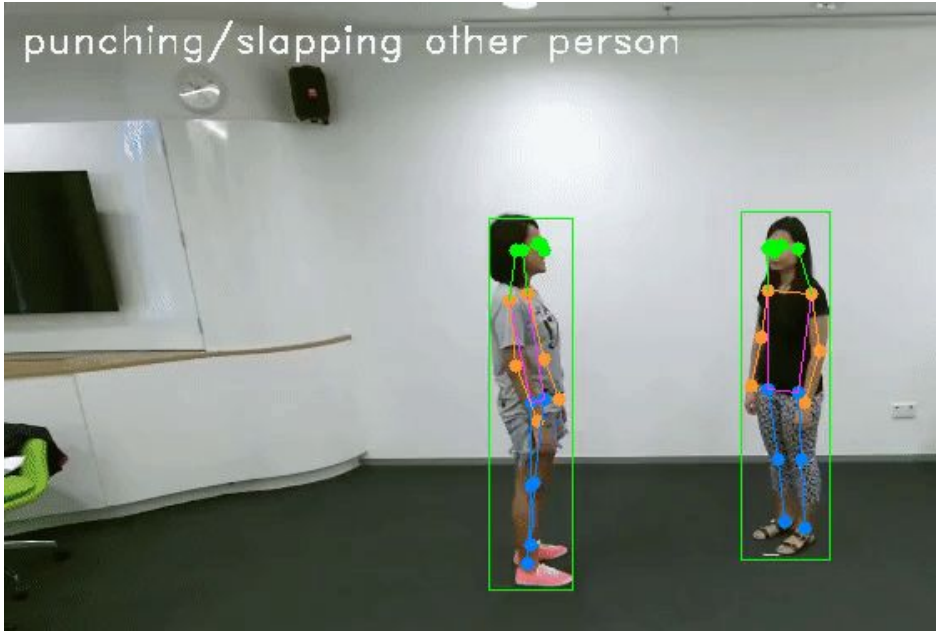




Real-world applications of CV (CNN + others)

Atari games - CNN + RL

Action human recognition - CNN + LSTM



Acknowledgments

- **Adam Green**, Host of Christchurch Data Science Meetup, Christchurch
  https://www.linkedin.com/in/adgefficiency/
- **Tatiana Gaintseva**, Lecture at the Deep Learning School, Moscow
  https://www.linkedin.com/in/tgaintseva/
- **James Atlas**, Senior Lecturer at the University of Canterbury, Christchurch
  https://www.linkedin.com/in/james-atlas-a13b9745/
- **Daniel Ritchie**, Assistant Professor at Brown University, USA
  https://dritchie.github.io/
- **Richard Green**, Professor at the University of Canterbury, Christchurch
  https://www.linkedin.com/in/richard-green-aa7a36223/