

An Introduction to Convolutional Neural Networks(CNN)

Simon Schupp | 24.07.23

Table-Of-Content

1. Introduction

2. Convolutional Neural Networks(CNN)

3. Self-trained CNN

4. Summary

Introduction
ooooo

Convolutional Neural Networks(CNN)
ooooooooo

Self-trained CNN
oooo

Summary
oo

Introduction

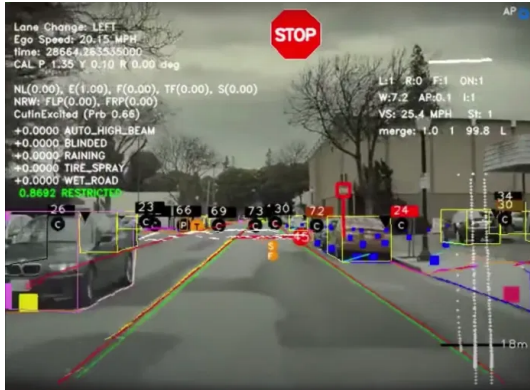
Introduction
●○○○○○

Convolutional Neural Networks(CNN)
○○○○○○○○○

Self-trained CNN
○○○○

Summary
○○

Visual Data and Algorithms



Source: bdechtalks, "Tesla AI chief explains why self-driving cars don't need Lidar"

Tesla Autopilot object Detection



Source: Google Blog "PaLM-E"

Google PaLM-E robot

Introduction
 ●●○○○

Convolutional Neural Networks(CNN)
 ○○○○○○○○

Self-trained CNN
 ○○○○

Summary
 ○○

How Computer see

What I see



Source: GitHub, hosamelsafty/Cats-VS-Dogs

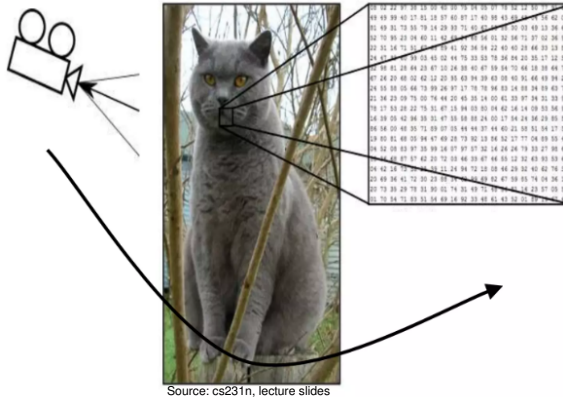
What a computer sees

```

08 02 22 97 38 15 00 40 00 75 04 05 07 78 52 12 50 77 91 08
49 49 99 40 17 81 18 57 60 87 17 40 98 43 69 48 04 56 62 00
81 49 31 73 55 79 14 29 93 71 40 67 53 88 30 03 49 13 36 65
52 70 95 23 04 60 11 42 69 24 68 56 01 32 56 71 37 02 36 91
22 31 16 71 51 67 63 89 41 92 36 54 22 40 40 28 66 33 13 80
24 47 32 60 99 03 45 02 44 75 33 53 78 36 84 20 35 17 12 50
32 98 81 28 64 23 67 10 26 38 40 67 59 54 70 66 18 38 64 70
67 26 20 68 02 62 12 20 95 63 94 39 63 08 40 91 66 49 94 21
24 55 58 05 66 73 99 26 97 17 78 78 96 83 14 88 34 89 63 72
21 36 23 09 75 00 76 44 20 45 35 14 00 61 33 97 34 31 33 95
78 17 53 28 22 75 31 67 15 94 03 80 04 62 16 14 09 53 56 92
16 39 05 42 96 35 31 47 55 58 88 24 00 17 54 24 36 29 85 57
86 56 00 48 35 71 89 07 05 44 44 37 44 60 21 58 51 54 17 58
19 80 81 68 05 94 47 69 28 73 92 13 86 32 17 77 04 89 55 40
04 52 08 83 97 35 99 16 07 97 57 32 16 26 26 79 33 27 98 66
88 36 68 87 57 62 20 72 03 46 33 67 46 55 12 32 63 93 53 69
04 42 16 73 38 25 39 11 24 94 72 18 08 46 29 32 40 62 76 36
20 69 36 41 72 30 23 88 34 62 99 69 82 67 59 85 74 04 36 16
20 73 35 29 78 31 90 01 74 31 49 71 48 86 81 16 23 57 05 54
01 70 54 71 83 51 54 69 16 92 33 48 61 43 52 01 89 19 67 48

```

Challenges: Viewpoint



When we change the viewpoint,
all the pixel change!

More challenges

- Viewpoint
- Illumination
- Occlusions
- Background Clutter
- etc.

Image Classifier

```
function CLASSIFYIMAGE(Image : Image)  
    // What to do here? Magic?  
    return label  
end function
```

It's not obvious how to hard code this.

Image Classifier

```
function CLASSIFYIMAGE(Image : Image)  
    // What to do here? Magic?  
    return label  
end function
```

It's not obvious how to hard code this.

Idea: Learn how to classify the Images → Use Convolutional Neural Networks(CNN)

Convolutional Neural Networks(CNN)

Introduction
○○○○○

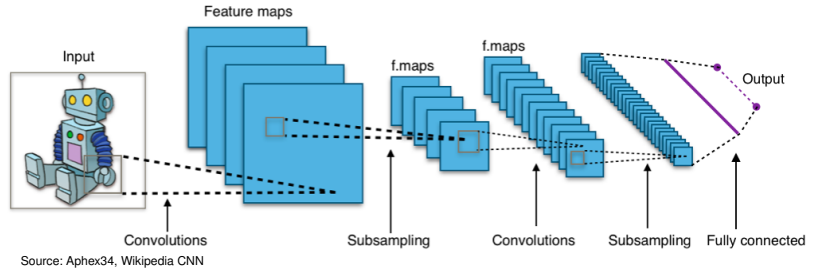
Convolutional Neural Networks(CNN)
●○○○○○○○

Self-trained CNN
○○○○

Summary
○○

CNN Overview

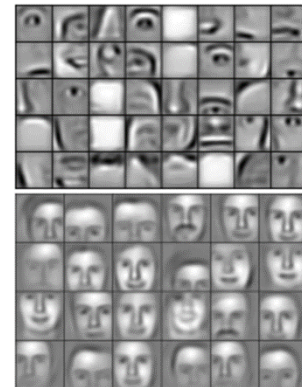
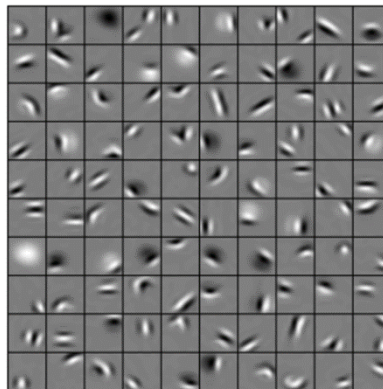
- Used mostly to classify images
- A CNN is a specialized NN
- It can pick up patterns and make sense
- Consists of the following Components:
 - Convolutional Layer
 - Pooling Layer



A Typical CNN

Patterns

- Simple patterns: edges, shapes, textures
- More complex patterns: ears, eyes, noses
- Even more complex patterns: faces, humans

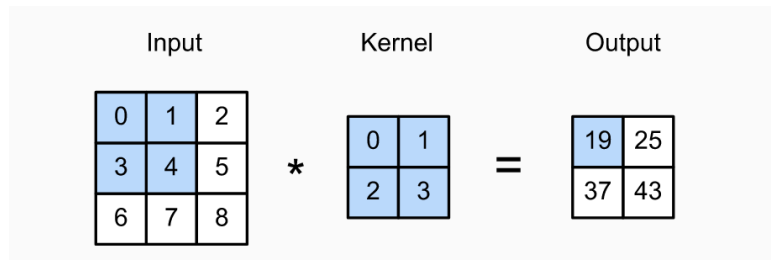


Source: Brandon Rohrer, course 193

Filters/Kernel

Filter/Kernel: A matrix

- initialized with random values
- values adjusted over time, to detect pattern



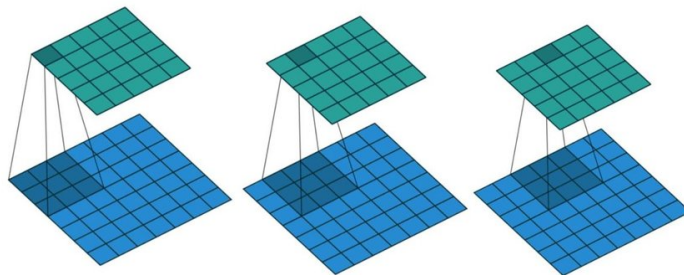
Source: d2l.ai, Chapter 7.2

Convolution Operation

Convolution

Two Important Hyperparameters:

- Stride
- Filter-Size



Source: Jelo Salomon, Lung Cancer Detection using Deep Learning

A Convolution with Stride 1

Convolution Animation

Convolution Animation

Introduction
○○○○○

Convolutional Neural Networks(CNN)
○○○○●○○

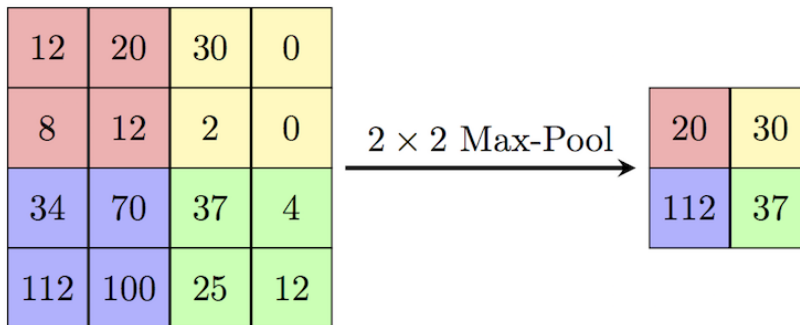
Self-trained CNN
○○○

Summary
○○

Pooling Layer

- A way to reduce the dimensionality of our Network
 - Requires less computation
- Many different ways to do this
 - Max-Pooling
 - Average-Pooling
 - 1x1 Convolutions

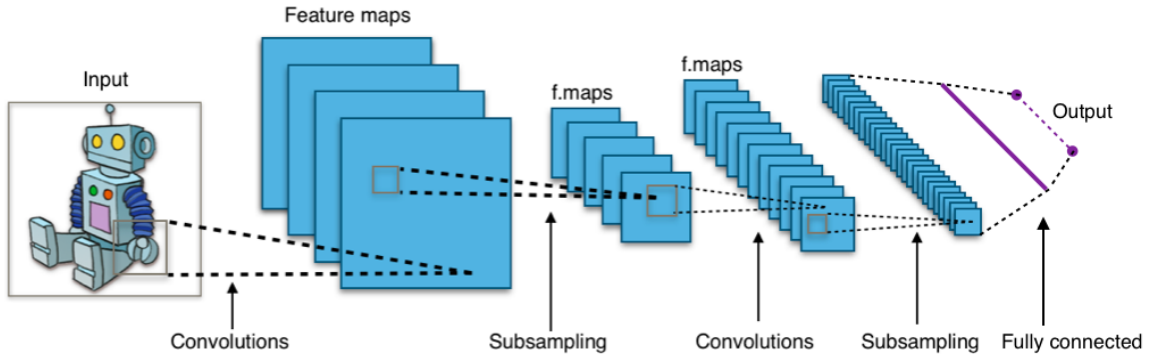
Pooling Layer



Source: cs231n lecture slides

A Max Pooling Layer

CNN: Bringing it Together



A Typical CNN

Self-trained CNN

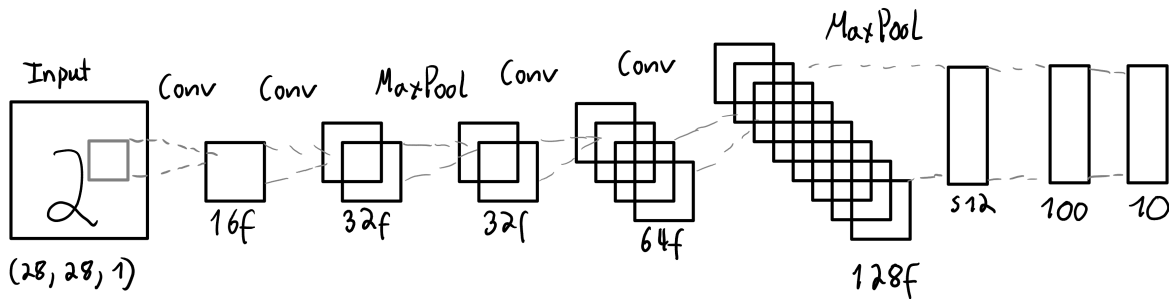
Introduction
○○○○○

Convolutional Neural Networks(CNN)
○○○○○○○○○

Self-trained CNN
●○○○

Summary
○○

Architecture



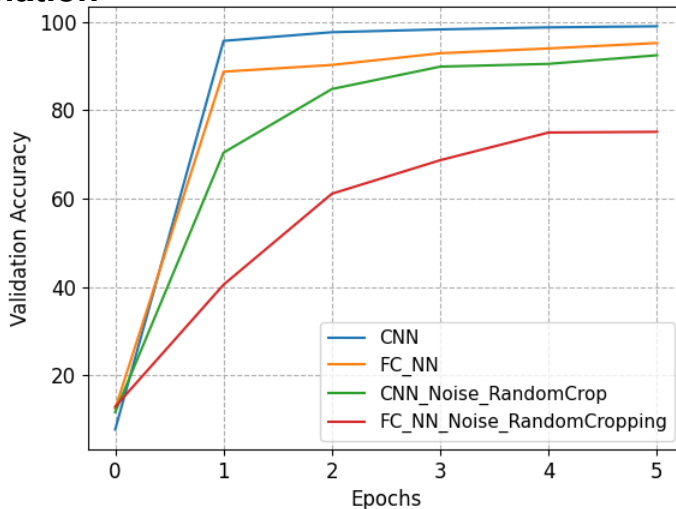
Trainings-Details

- Dataset: MNIST(60,000 Images)
- Trainings time: 5 Epochs
- Batch size: 1024
- Optimizer: Adam
- Learning-rate: 0.001
- Trained on Google Colab with GPU



Example Data from MNIST dataset

Results & Ablation



Summary

Introduction
○○○○○

Convolutional Neural Networks(CNN)
○○○○○○○○○

Self-trained CNN
○○○○

Summary
●○

Summary

- Teaching a Computer the semantic meaning of Images is hard

Summary

- Teaching a Computer the semantic meaning of Images is hard
- But it's possible with the help of CNN

Summary

- Teaching a Computer the semantic meaning of Images is hard
- But it's possible with the help of CNN
- Nowadays its rather easy to get good performance on simple datasets

Literature

- [1] Alex Krizhevsky, Ilya Sutskever und Geoffrey E Hinton. „ImageNet Classification with Deep Convolutional Neural Networks“. In: *Advances in Neural Information Processing Systems*. Hrsg. von F. Pereira u. a. Bd. 25. Curran Associates, Inc., 2012. URL: https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.
- [2] Y. LeCun u. a. „Backpropagation Applied to Handwritten Zip Code Recognition“. In: *Neural Computation* 1.4 (1989), S. 541–551. DOI: 10.1162/neco.1989.1.4.541.
- [3] Y. Lecun u. a. „Gradient-based learning applied to document recognition“. In: *Proceedings of the IEEE* 86.11 (1998), S. 2278–2324. DOI: 10.1109/5.726791.