Deep Learning

Lecture 0: Introduction

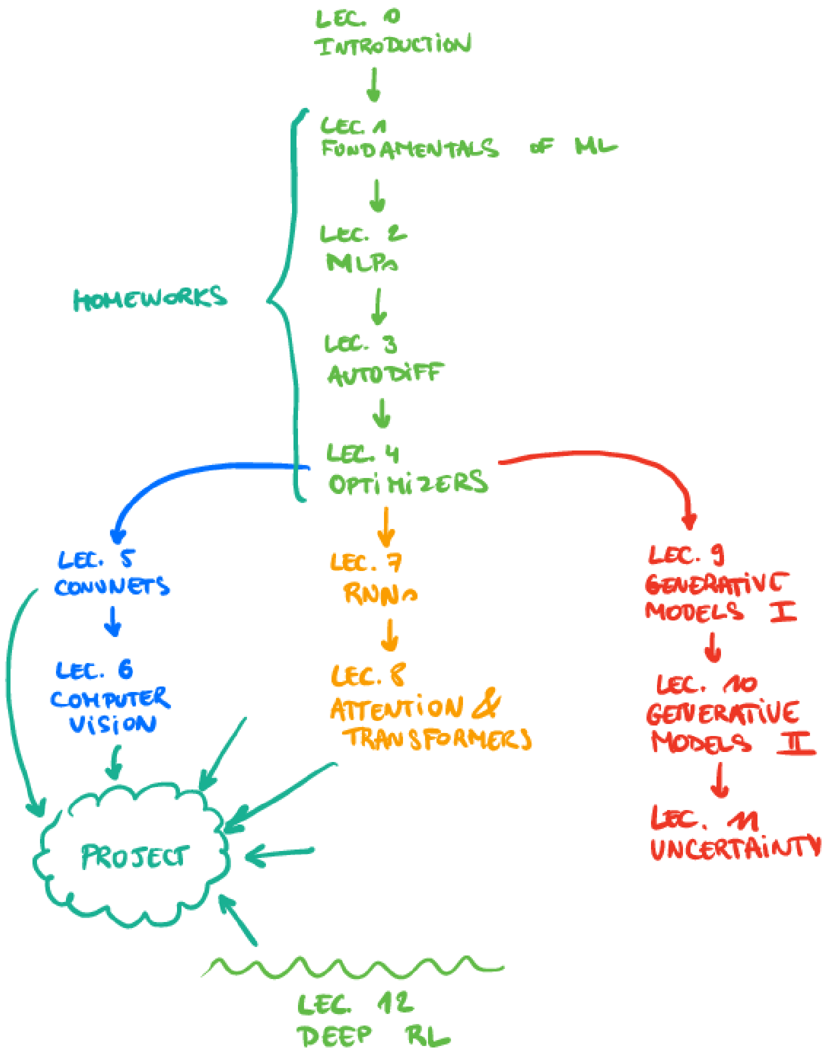
Prof. Gilles Louppe
g.louppe@uliege.be

Today

- Course outline
- Introduction to deep learning
- Fundamentals of machine learning

Outline

- Lecture 0: Introduction
- Lecture 1: Fundamentals of machine learning
- Lecture 2: Multi-layer perceptron
- Lecture 3: Automatic differentiation
- Lecture 4: Training neural networks
- Lecture 5: Convolutional neural networks
- Lecture 6: Computer vision
- Lecture 7: Recurrent neural networks
- Lecture 8: Attention and transformer networks
- Lecture 9: Generative models (Part 1)
- Lecture 10: Generative models (Part 2)
- Lecture 11: Uncertainty
- Lecture 12: Deep reinforcement learning


LEC. 0
INTRODUCTION
LEC. 1
FUNDAMENTALS OF ML
LEC. 2
MLPs
LEC. 3
AUTODIFF
LEC. 4
OPTIMIZERS
HOMEWORKS
LEC. 5
CONVNETS
LEC. 6
COMPUTER VISION
LEC. 7
RNNs
LEC. 8
ATTENTION &
TRANSFORMERS
LEC. 9
GENERATIVE
MODELS I
LEC. 10
GENERATIVE
MODELS II
LEC. 11
UNCERTAINTY
PROJECT
LEC. 12
DEEP RL

My mission

By the end of this course, you will have acquired a solid and detailed understanding of the field of deep learning.

You will have learned how to design deep neural networks for a wide range of advanced probabilistic inference tasks and how to train them.

These models seen in the course apply to a wide variety of artificial intelligence problems, with plenty of applications in engineering and science.

Why learning?

What do you see?

Sheepdog or mop?

—
Credits: Karen Zack, 2016.

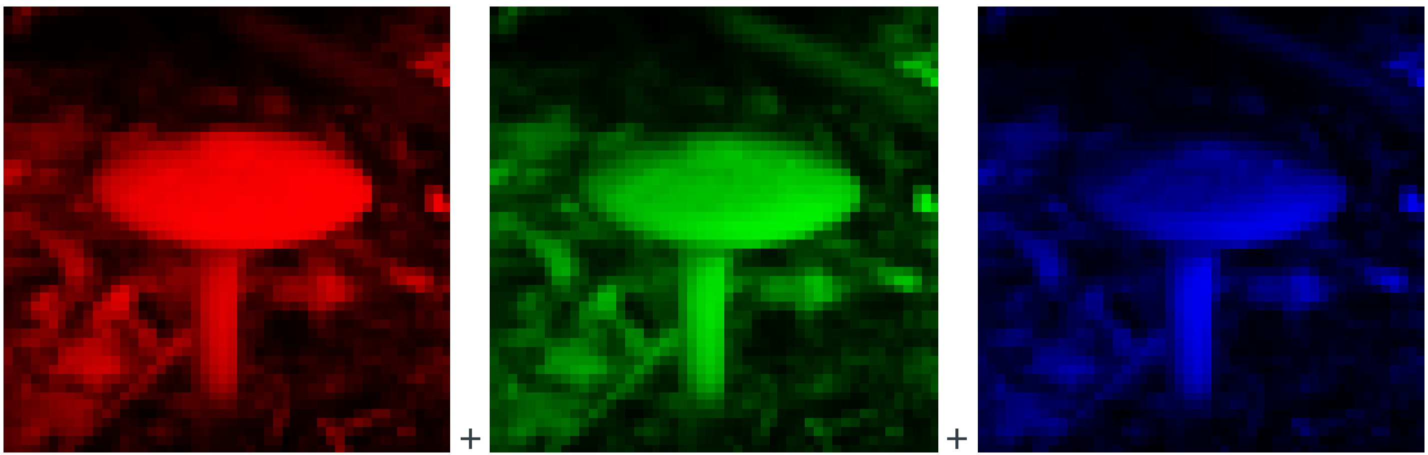
Chihuahua or muffin?

—
Credits: Karen Zack. 2016.

The (human) brain is so good at interpreting visual information that the gap between raw data and its semantic interpretation is difficult to assess intuitively:

This is a mushroom.

This is a mushroom.

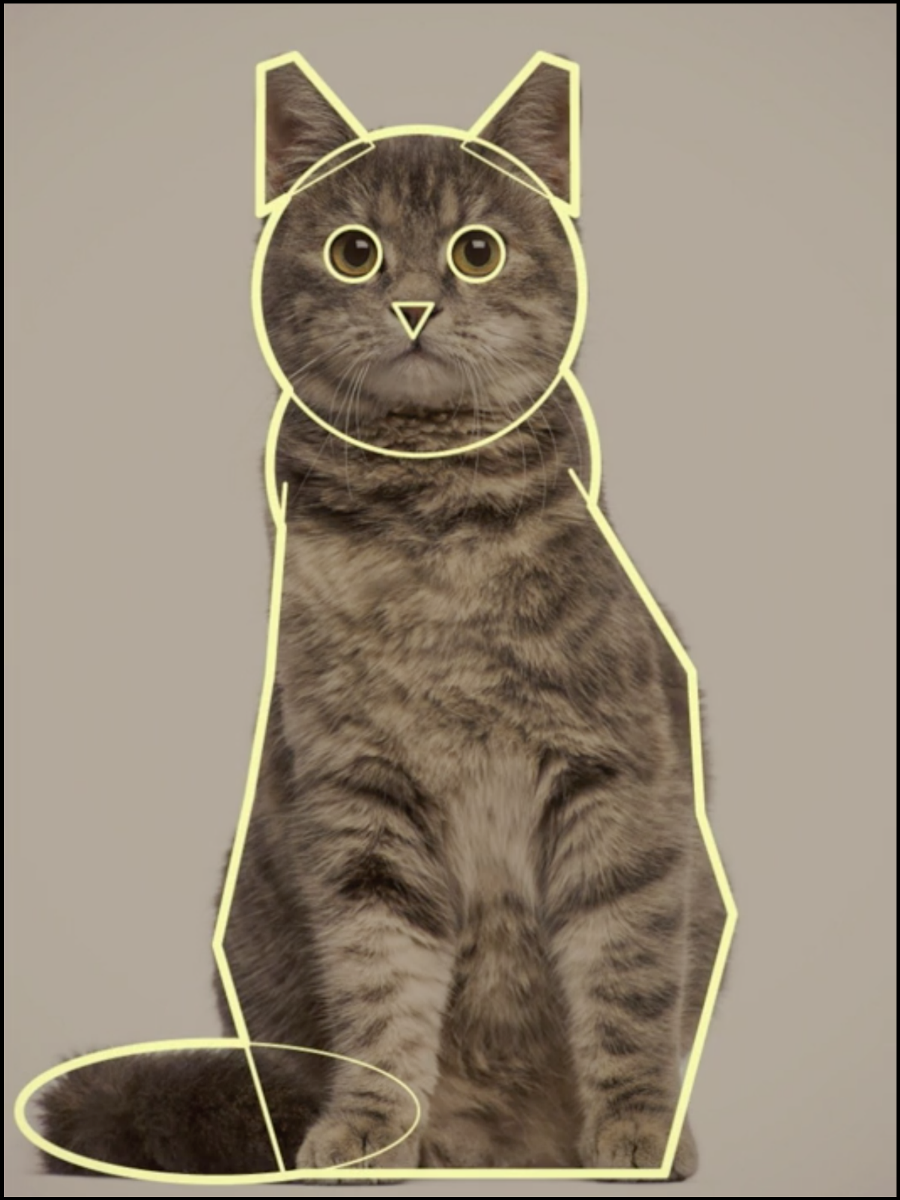


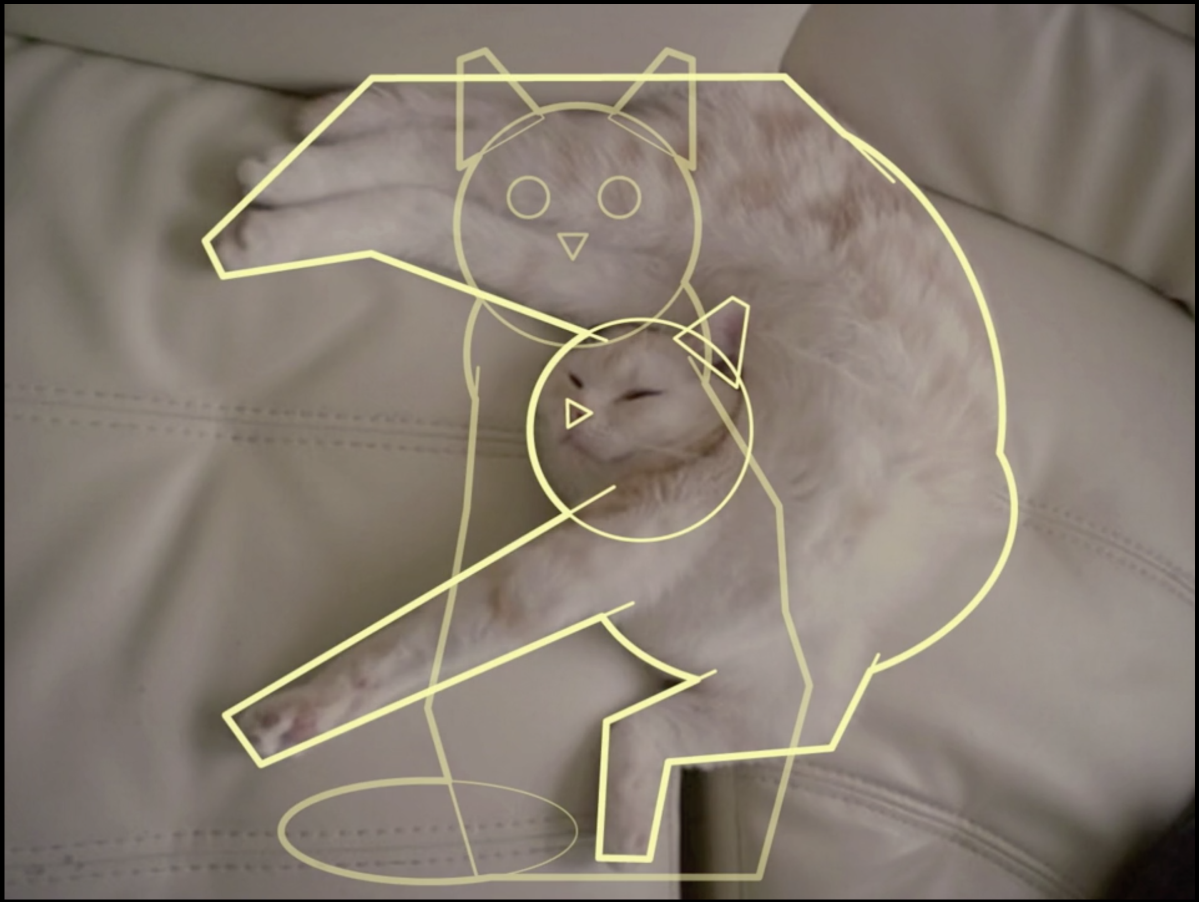
This is a mushroom.

```
array([[[0.03921569, 0.03529412, 0.02352941, 1.        ],
        [0.2509804 , 0.1882353 , 0.20392157, 1.        ],
        [0.4117647 , 0.34117648, 0.37254903, 1.        ],
        ...,
        [0.20392157, 0.23529412, 0.17254902, 1.        ],
        [0.16470589, 0.18039216, 0.12156863, 1.        ],
        [0.18039216, 0.18039216, 0.14117648, 1.        ]],

       [[0.1254902 , 0.11372549, 0.09411765, 1.        ],
        [0.2901961 , 0.2509804 , 0.24705882, 1.        ],
        [0.21176471, 0.2       , 0.20392157, 1.        ],
        ...,
        [0.1764706 , 0.24705882, 0.12156863, 1.        ],
        [0.10980392, 0.15686275, 0.07843138, 1.        ],
        [0.16470589, 0.20784314, 0.11764706, 1.        ]],

       [[0.14117648, 0.12941177, 0.10980392, 1.        ],
        [0.21176471, 0.1882353 , 0.16862746, 1.        ],
        [0.14117648, 0.13725491, 0.12941177, 1.        ],
        ...,
        [0.10980392, 0.15686275, 0.08627451, 1.        ],
        [0.0627451 , 0.08235294, 0.05098039, 1.        ],
        [0.14117648, 0.2       , 0.09803922, 1.        ]],

       ...,
```

This is a mushroom.

Writing a computer program that sees?





Extracting semantic information requires models of high complexity, which cannot be designed by hand.

However, one can write a program that learns the task of extracting semantic information.

The common approach used in practice consists of:

- defining a parametric model with high capacity,
- optimizing its parameters, by "making it work" on the training data.

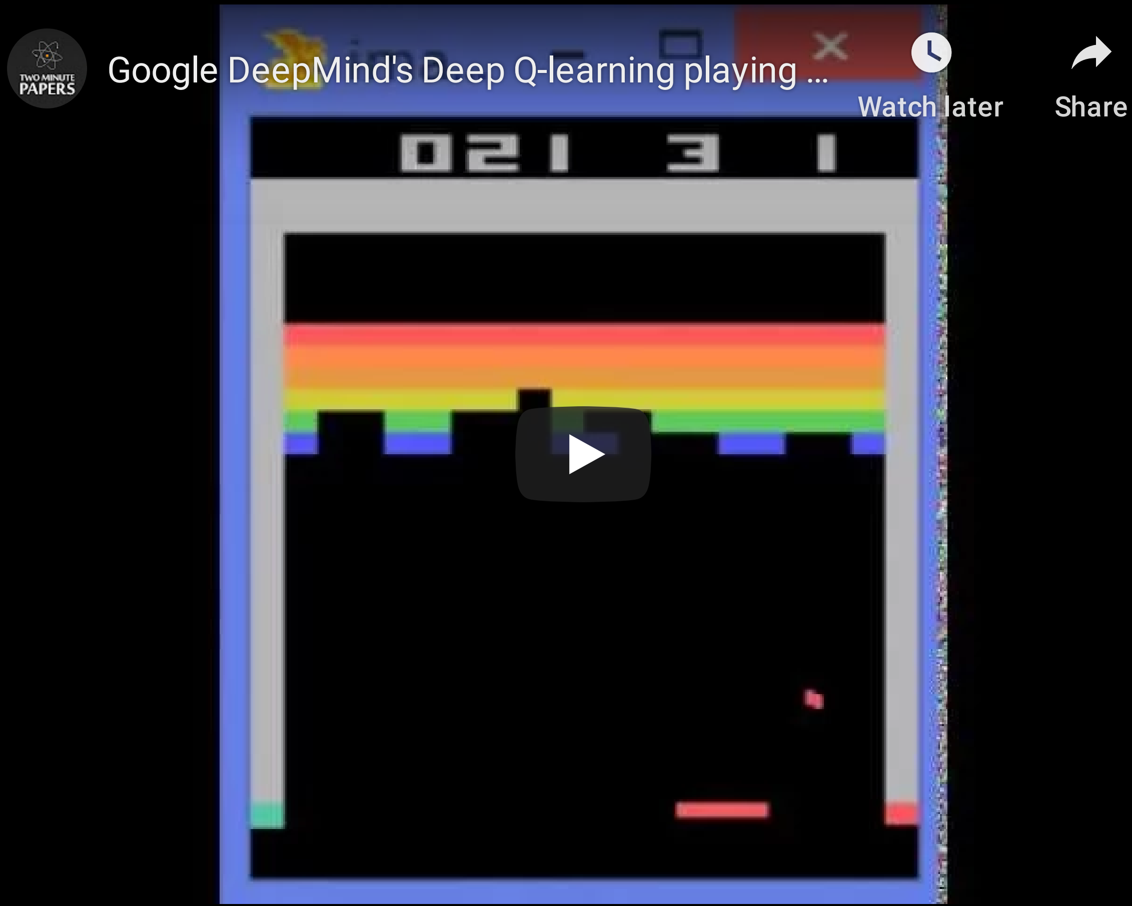

Konbini
techno
Il y a eu
des hauts et des bas.
0:00 / 4:41

Applications and successes

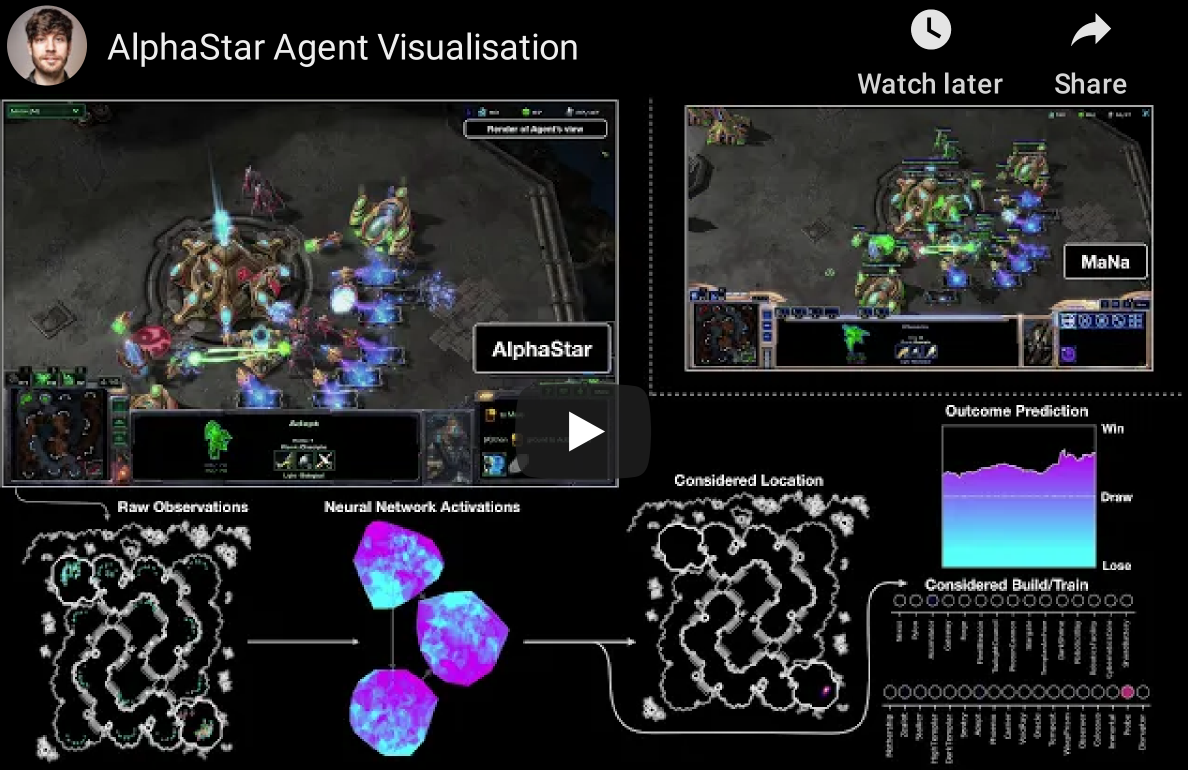
FB Detectron2: A PyTorch-based modular object detection...

Copy link

Object detection, pose estimation, segmentation (2019)



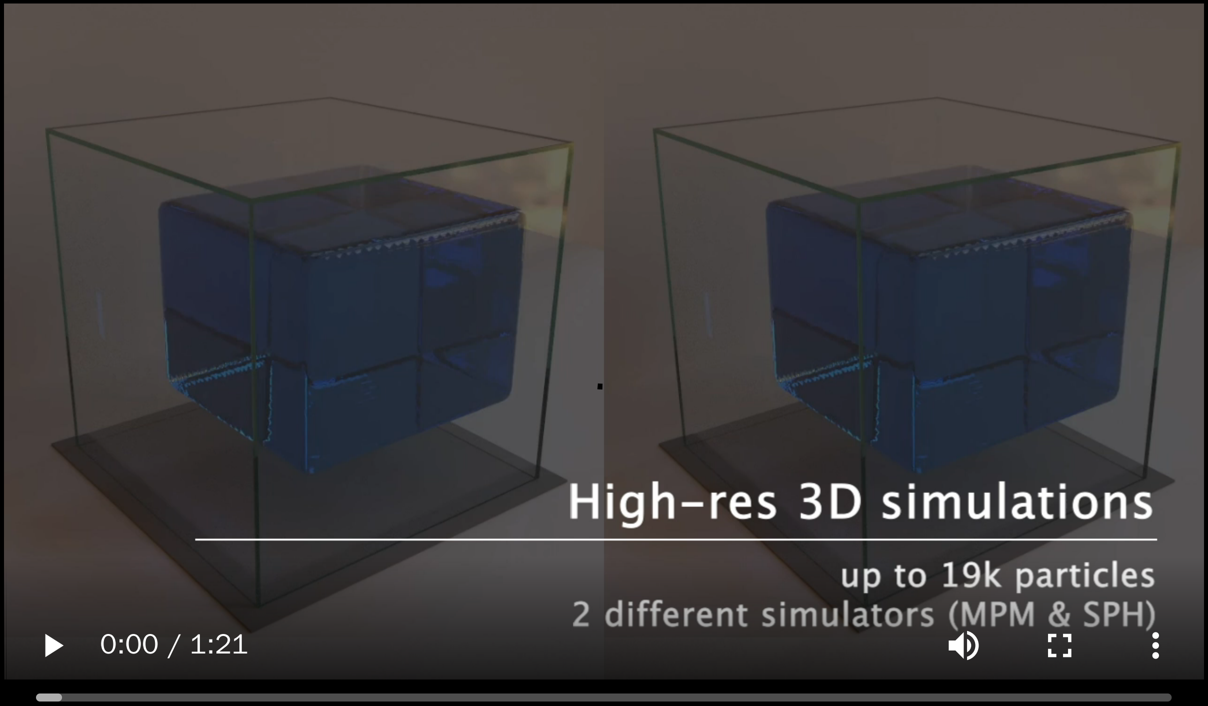
Reinforcement learning (Mnih et al, 2014)



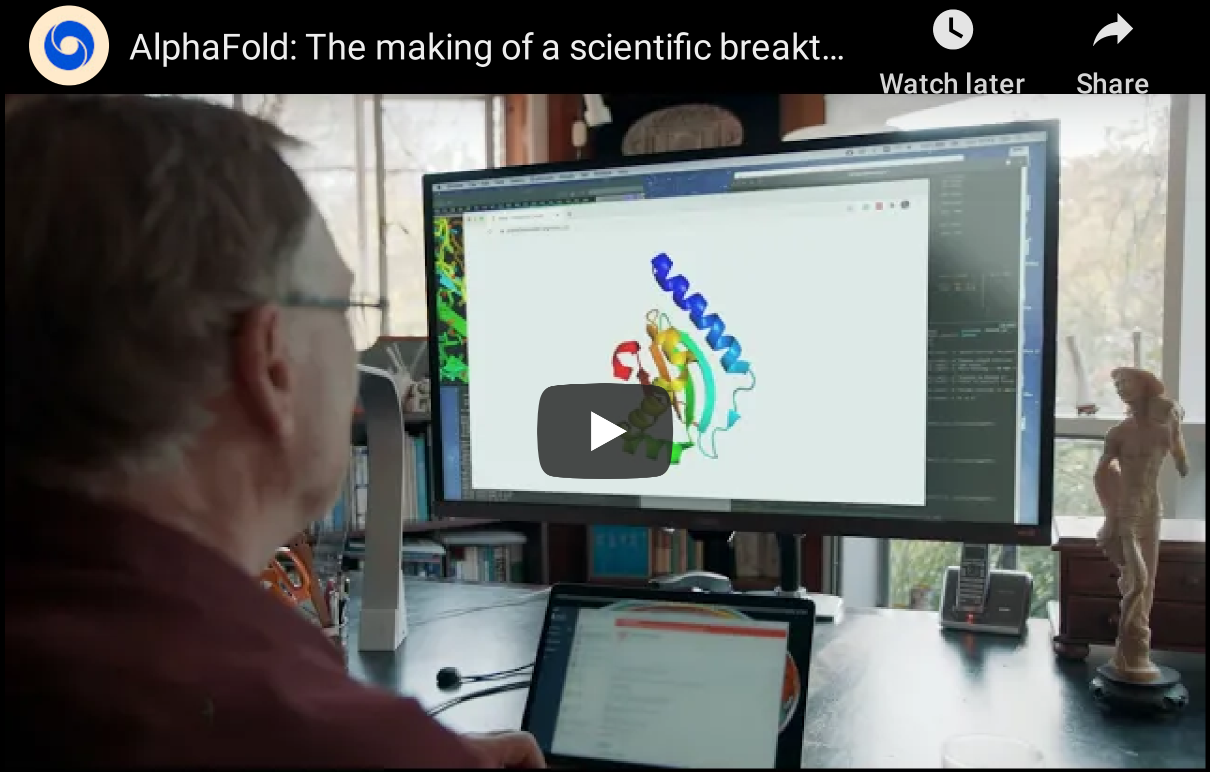
Strategy games (Deepmind, 2016-2018)

Autonomous cars (NVIDIA, 2016)

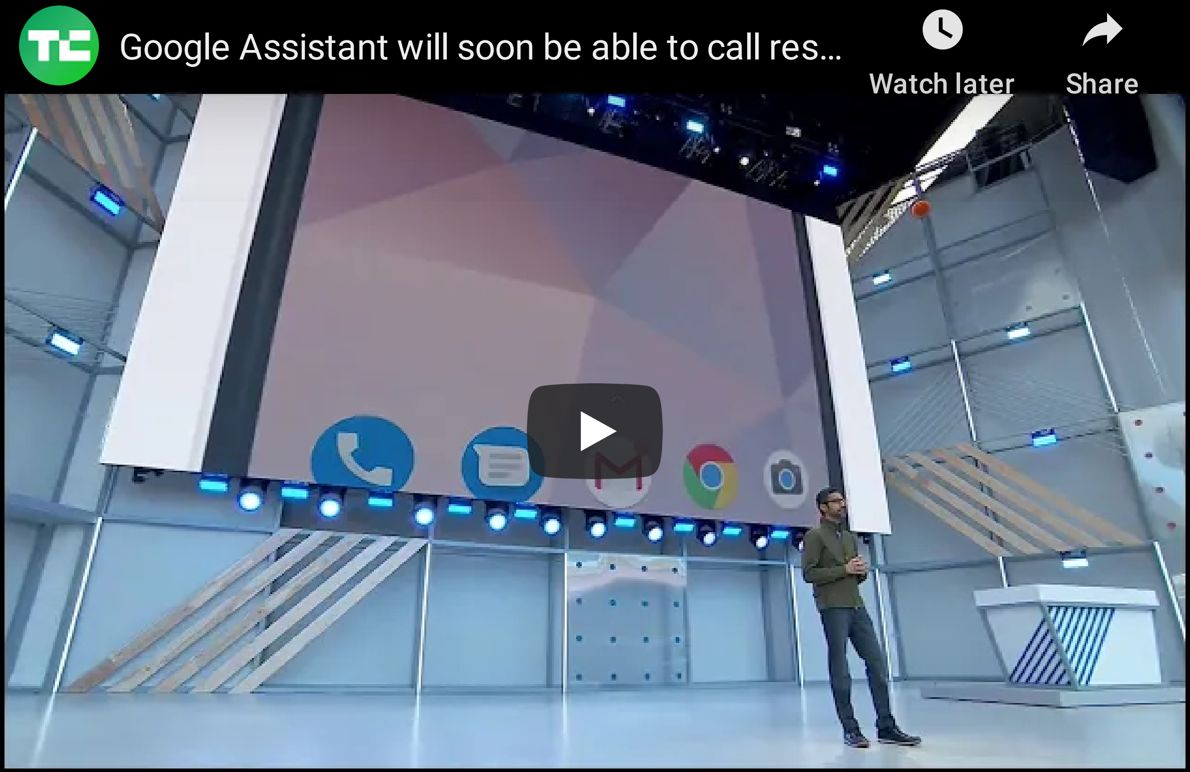
Autopilot (Tesla, 2019)



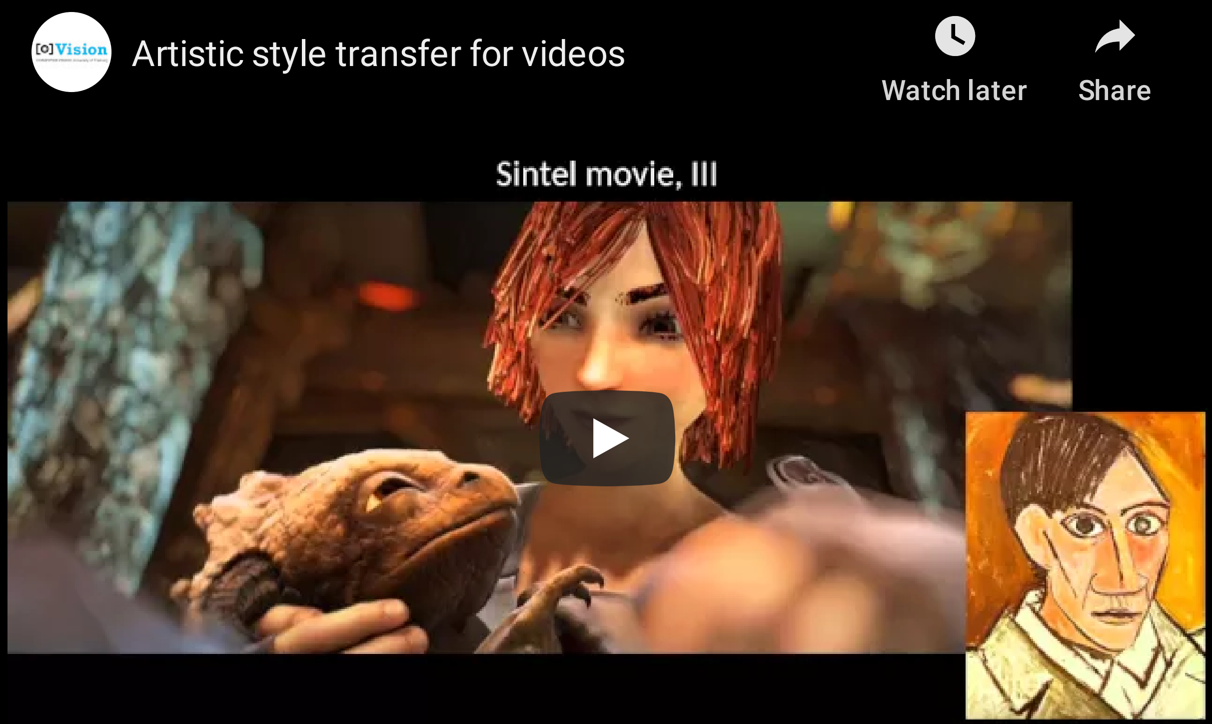
Physics simulation (Sanchez-Gonzalez et al, 2020)



AI for Science (Deepmind, AlphaFold, 2020)



Speech synthesis and question answering (Google, 2018)



Artistic style transfer (Ruder et al, 2016)

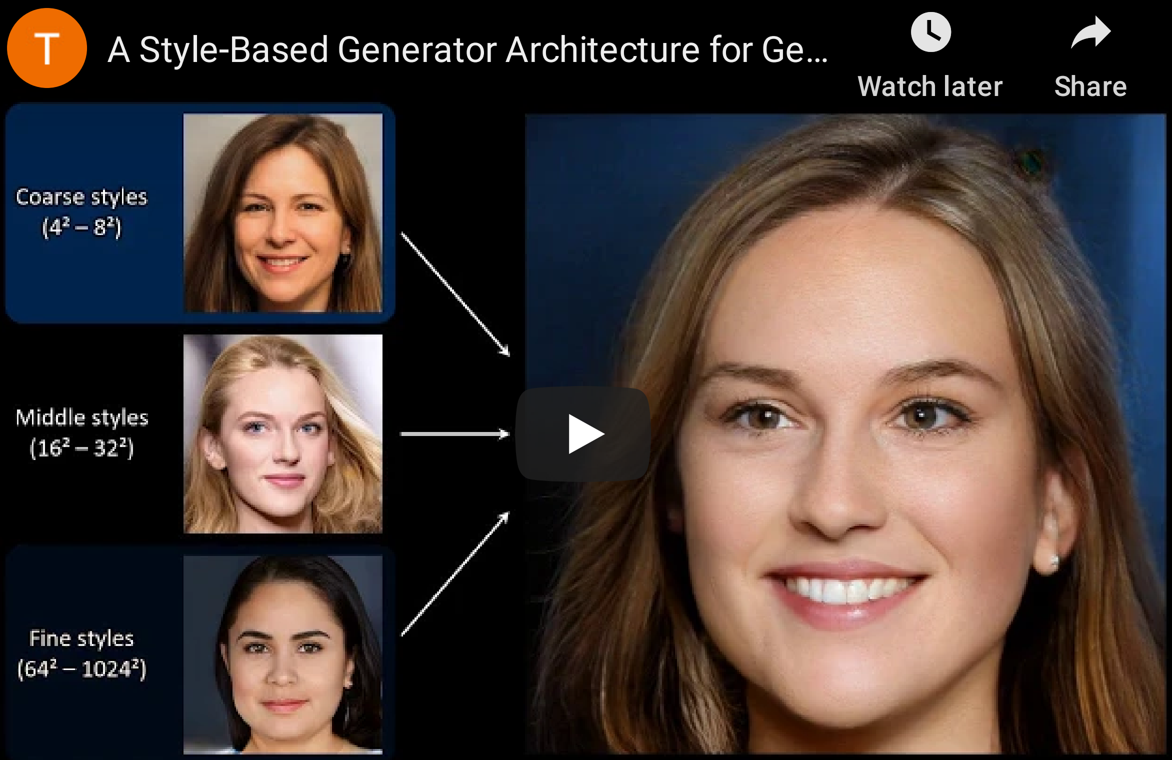


Image generation (Karras et al, 2018)

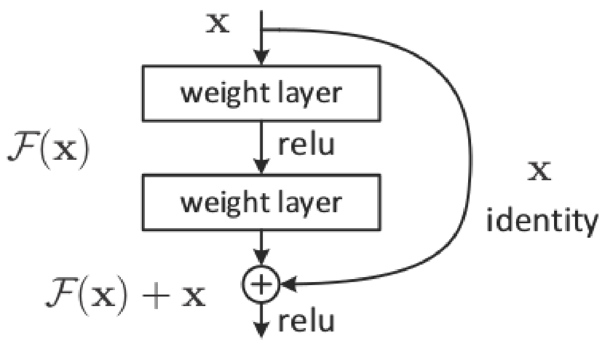
Music composition (NVIDIA, 2017)

Dali Lives (2019)

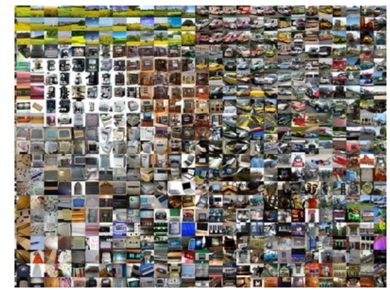
*"ACM named **Yoshua Bengio**, **Geoffrey Hinton**, and **Yann LeCun** recipients of the **2018 ACM A.M. Turing Award** for conceptual and engineering breakthroughs that have made deep neural networks a critical component of computing."*

Why does it work now?

Algorithms (old and new)



More data



Software

Faster compute engines

Building on the shoulders of giants

Five decades of research in machine learning provided

- a taxonomy of ML concepts (classification, generative models, clustering, kernels, linear embeddings, etc.),
- a sound statistical formalization (Bayesian estimation, PAC),
- a clear picture of fundamental issues (bias/variance dilemma, VC dimension, generalization bounds, etc.),
- a good understanding of optimization issues,
- efficient large-scale algorithms.

Deep learning

From a practical perspective, deep learning

- lessens the need for a deep mathematical grasp,
- makes the design of large learning architectures a system/software development task,
- allows to leverage modern hardware (clusters of GPUs),
- does not plateau when using more data,
- makes large trained networks a commodity.

For the last forty years we have programmed computers; for the next forty years we will train them.

Chris Bishop, 2020.

The end.

References

- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. nature, 521(7553), 436-444.