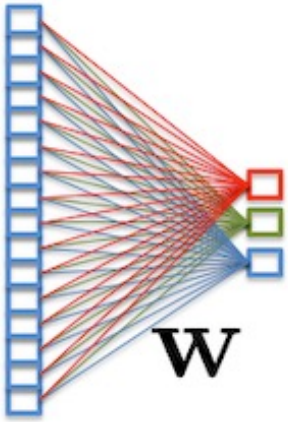
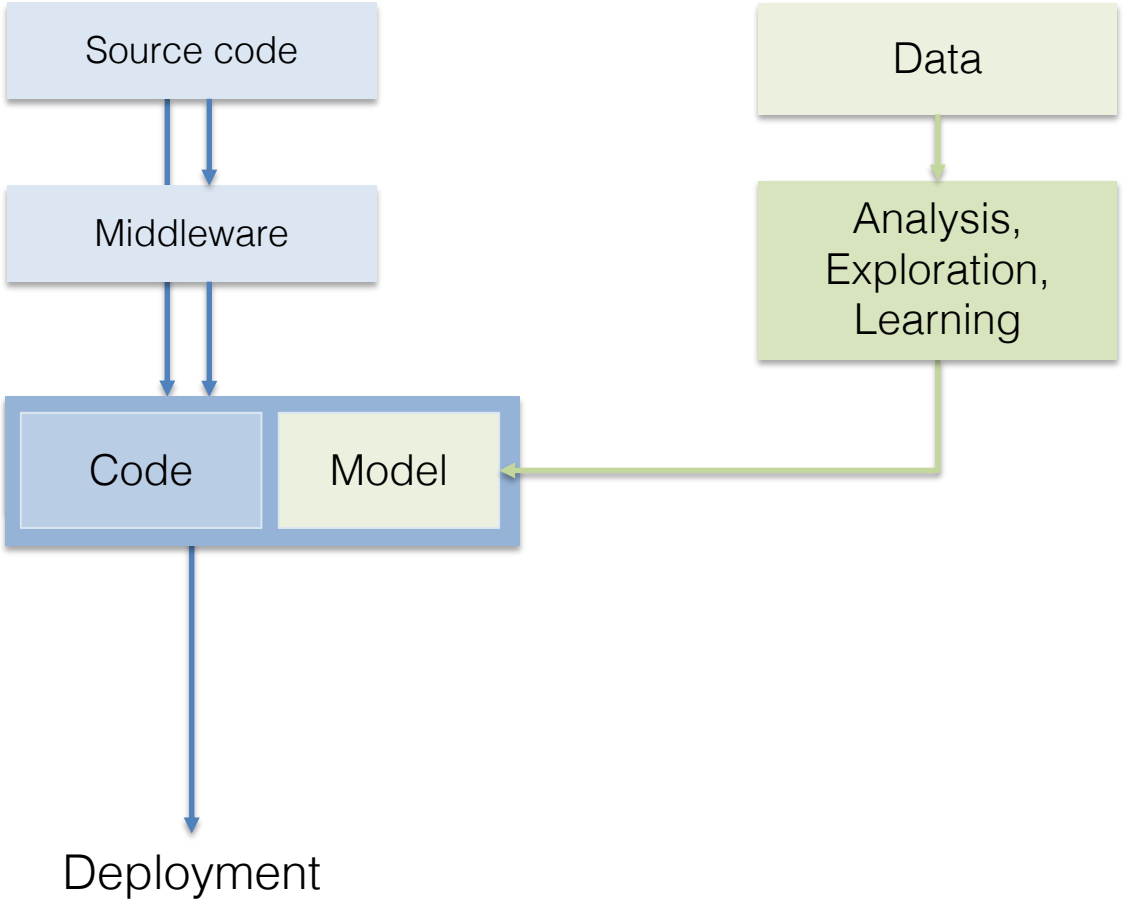
5IF - Deep Learning and Differentiable Programming



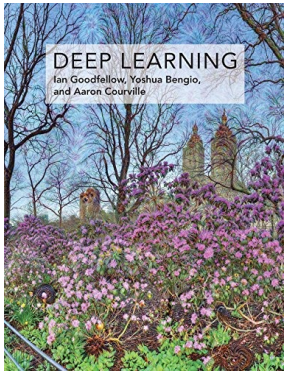
INSA LYON Christian Wolf

Software development ... and data



Approximation
Simplification
Handling Uncertainty
Modelling

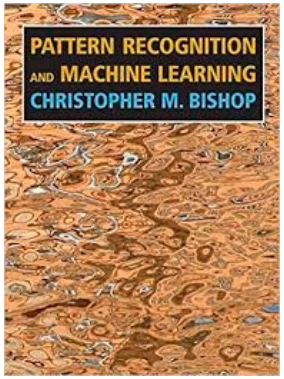
To go deeper



Ian Goodfellow, Yoshua Bengio, Aaron Courville, « Deep Learning », MIT Press

PyTorch online tutorials

https://pytorch.org/



Christopher Bishop, « Pattern Recognition and Machine Learning », 2006 (*Pre-deeplearning area, but a very pedagogical book on machine learning)*

Learn Python!

For example : https://learnxinyminutes.com/docs/python/

Learn X in Y minutes

Share this page

Select theme: light dark

Where X=python

Get the code: learnpython.py

Python was created by Guido Van Rossum in the early 90s. It is now one of the most popular languages in existence. I fell in love with Python for its syntactic clarity. It's basically executable pseudocode.

Feedback would be highly appreciated! You can reach me at @louiedinh or louiedinh [at] [google's email service]

Note: This article applies to Python 2.7 specifically, but should be applicable to Python 2.x. Python 2.7 is reaching end of life and will stop being maintained in 2020, it is though recommended to start learning Python with Python 3. For Python 3.x, take a look at the Python 3 tutorial.

It is also possible to write Python code which is compatible with Python 2.7 and 3.x at the same time, using Python `__future__` imports. `__future__` imports allow you to write Python 3 code that will run on Python 2, so check out the Python 3 tutorial.

```
# Single line comments start with a number symbol.

""" Multiline strings can be written
    using three "s, and are often used
    as comments
"""
```

1 Introduction

1 Introduction: machine learning, a couple of applications [36]
2 A short history of deep learning [8]
3 The basics of machine learning: fitting and generalization [16]

2 Neural networks and PyTorch

1 Frameworks and Tensors *{+PyTorch}* [24]
2 Simple models (linear regression, logistic regression) *{+PyTorch}* [32]
3 Multi layer models + universal approximation theorem *{+PyTorch}* [32]
4 Train/Validation/Test split; Tensorboard *{+PyTorch}* [14]
5 Gradient Backpropagation and Autograd *{+PyTorch}* [17]
6 Stochastic Gradient Descent and Variants (Adam, RMSProp) [15]
7 Shift invariance and Convolutions *{+PyTorch}* [38]

3 Scaling up: vision, transfer, visualization

1 Computer Vision [39]
2 Visualization [20]
3 Transfer learning [19]
4 Semi-supervised, Self-supervised learning [8]
5 GPUs – Software *{+CUDA, +PyTorch}* [14]

4 Structure: sequences, graphs, attention

1 Recurrent Neural Networks and variants [35]
2 Attention mechanisms in computer vision [41]
3 Graphs and relational reasoning [37]
4 Self attention and transformers [19]

5 Advanced applications

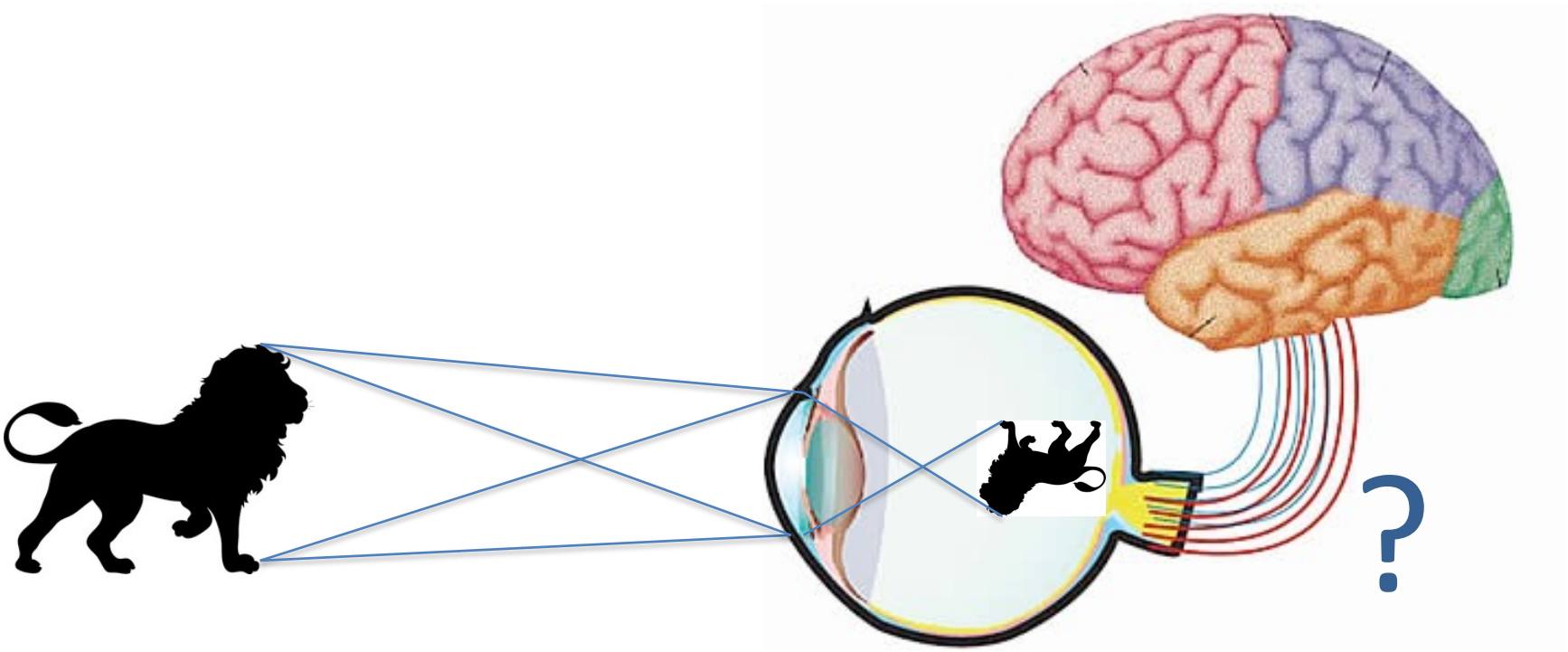
1 Reinforcement Learning [25]
2 Should we model or learn? [27]
3 Learning Robot Navigation [25]
4 Machine learning as an experimental science [9]
5 Conclusions [4]

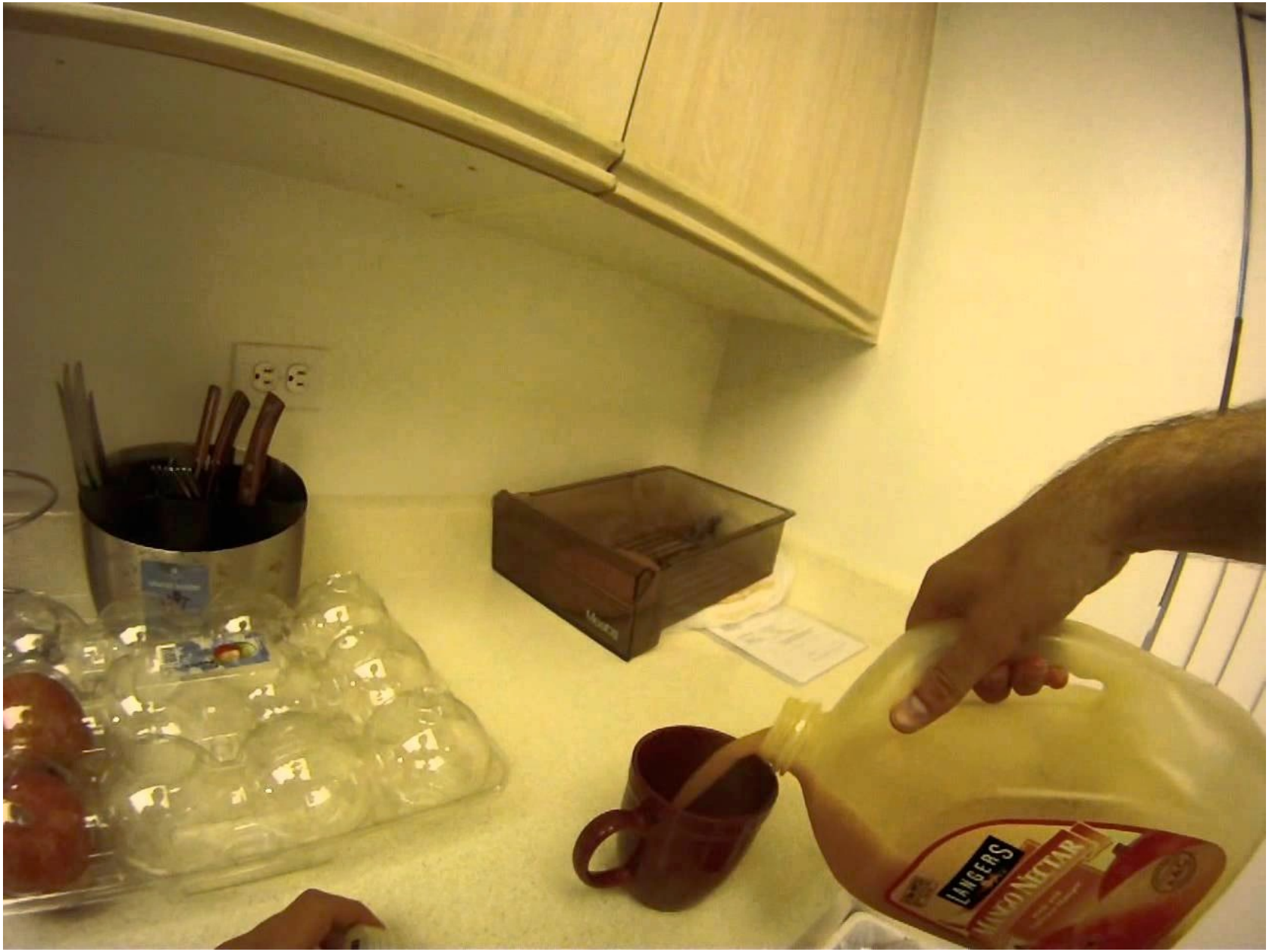
6 Evaluation

Multiple Choice Test (MOODLE)

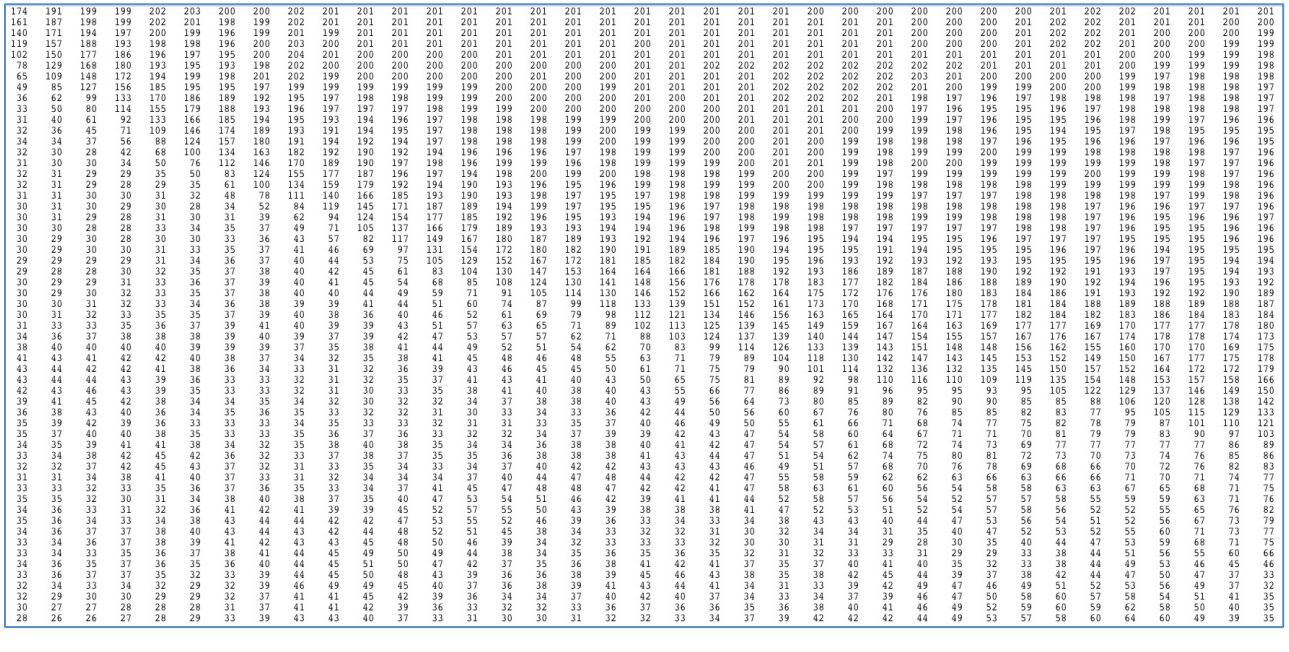
1.1 Introduction

The human visual system




LANGERS

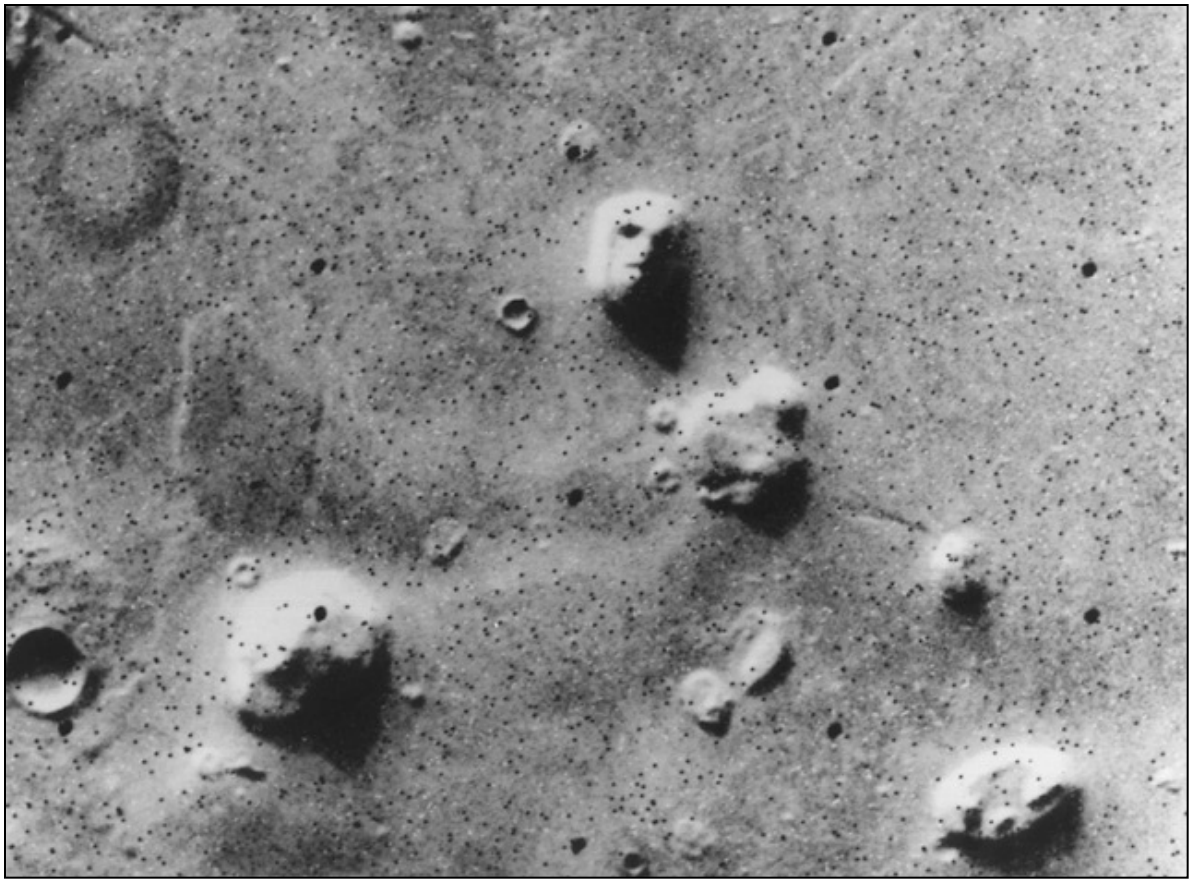
Interpretation



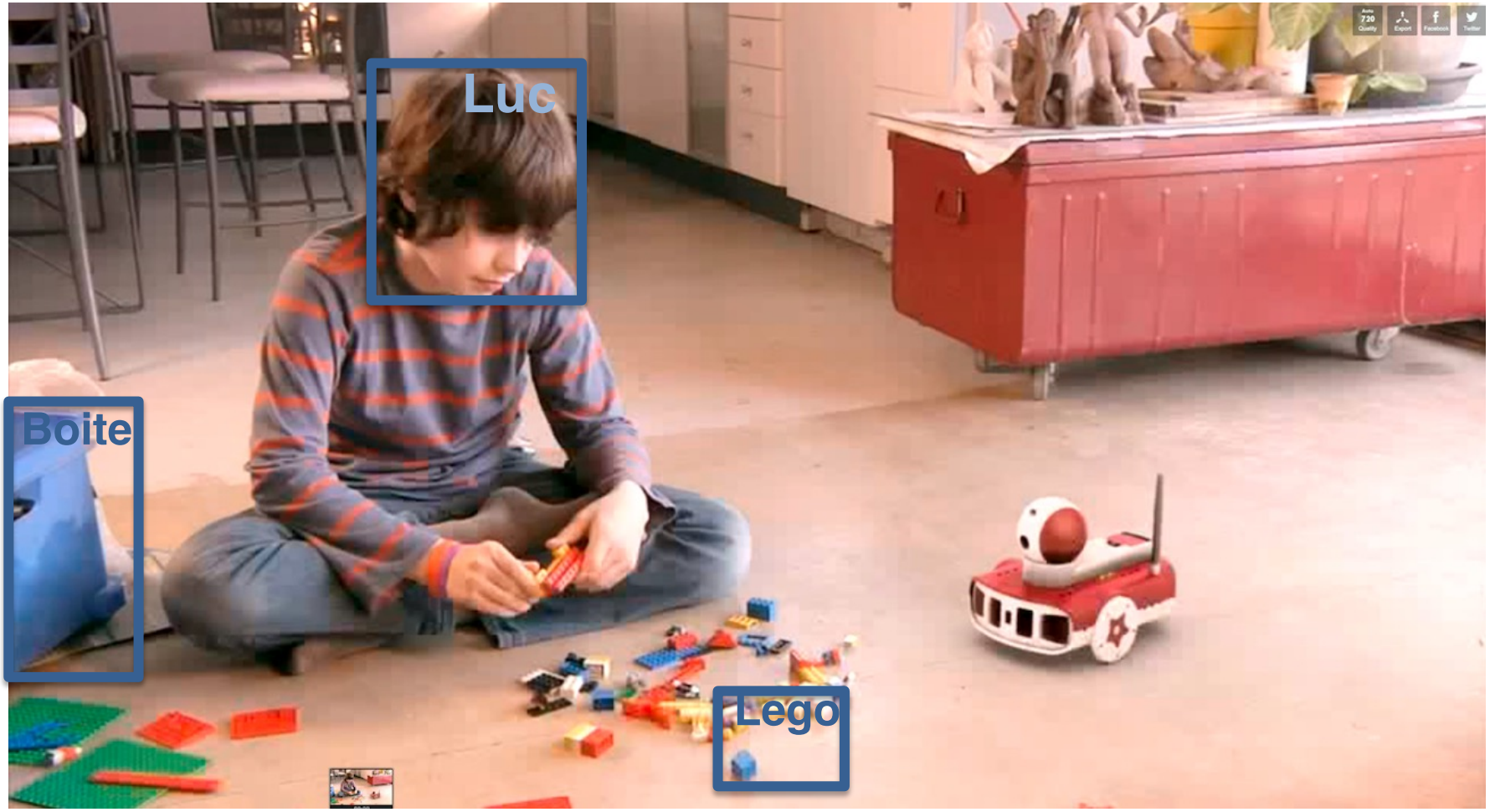
=?

Friend or foe?
Smile or run?

The brain specializes on faces



Gesture recognition: « compagnon » robot



Awabot

Navigation : where am I?
« Visual Landmarks »

Robot navigation

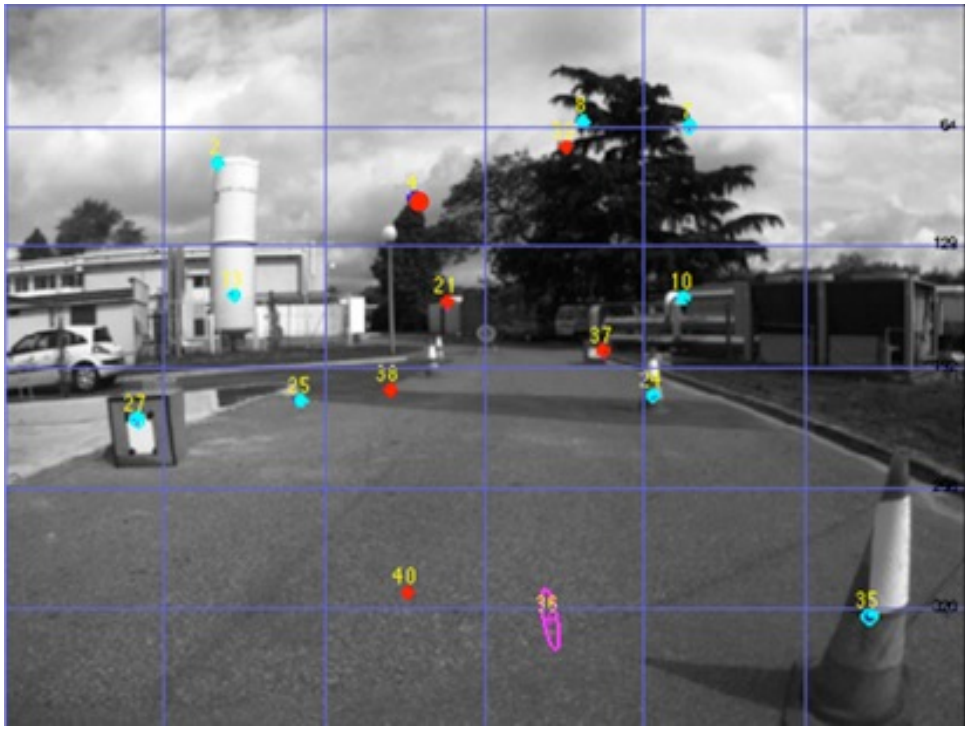


Figure : LAAS-CNRS, Toulouse

Visual SLAM using landmarks

Support for tools (motor control)

Visual servoeing

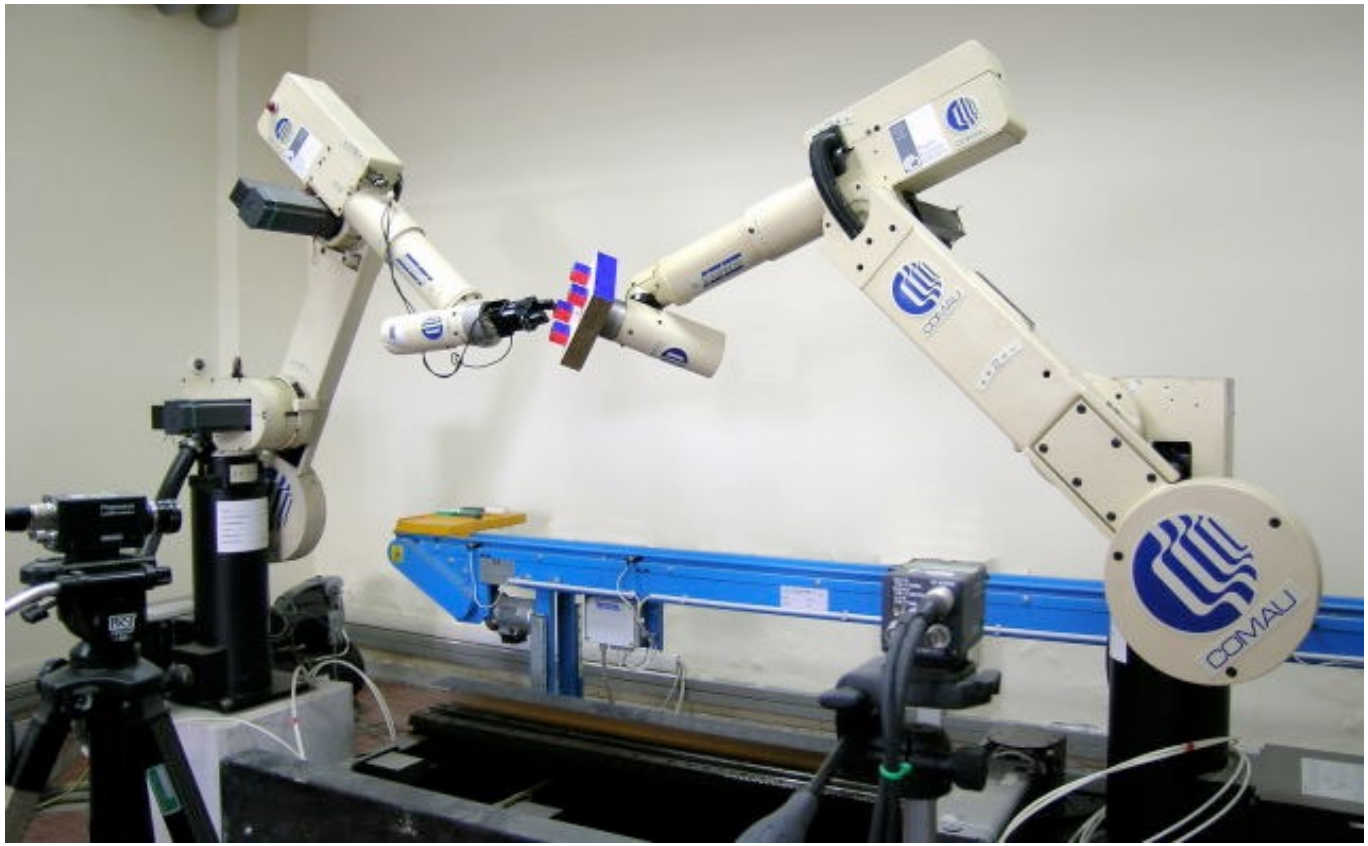
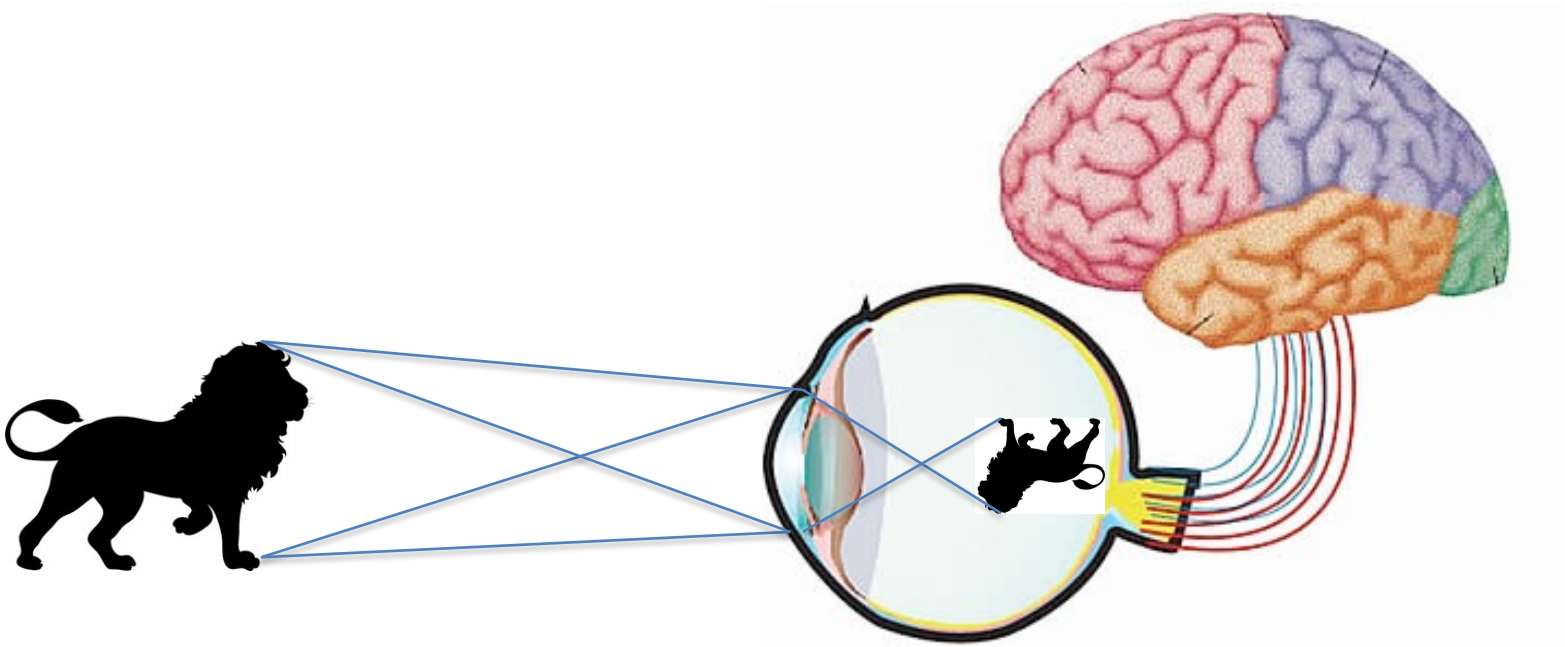
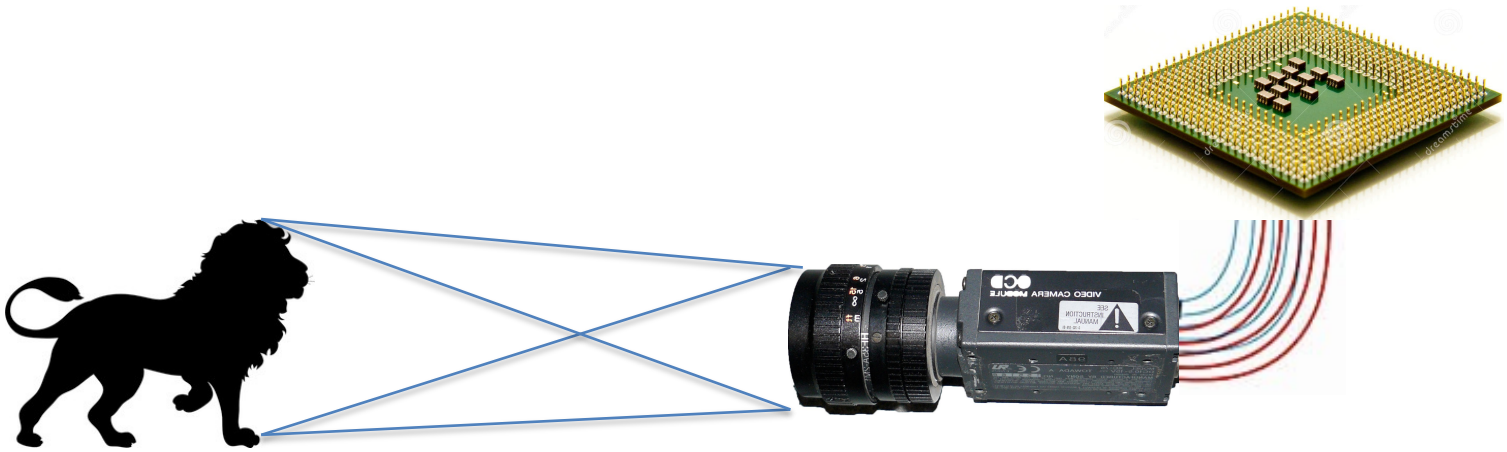
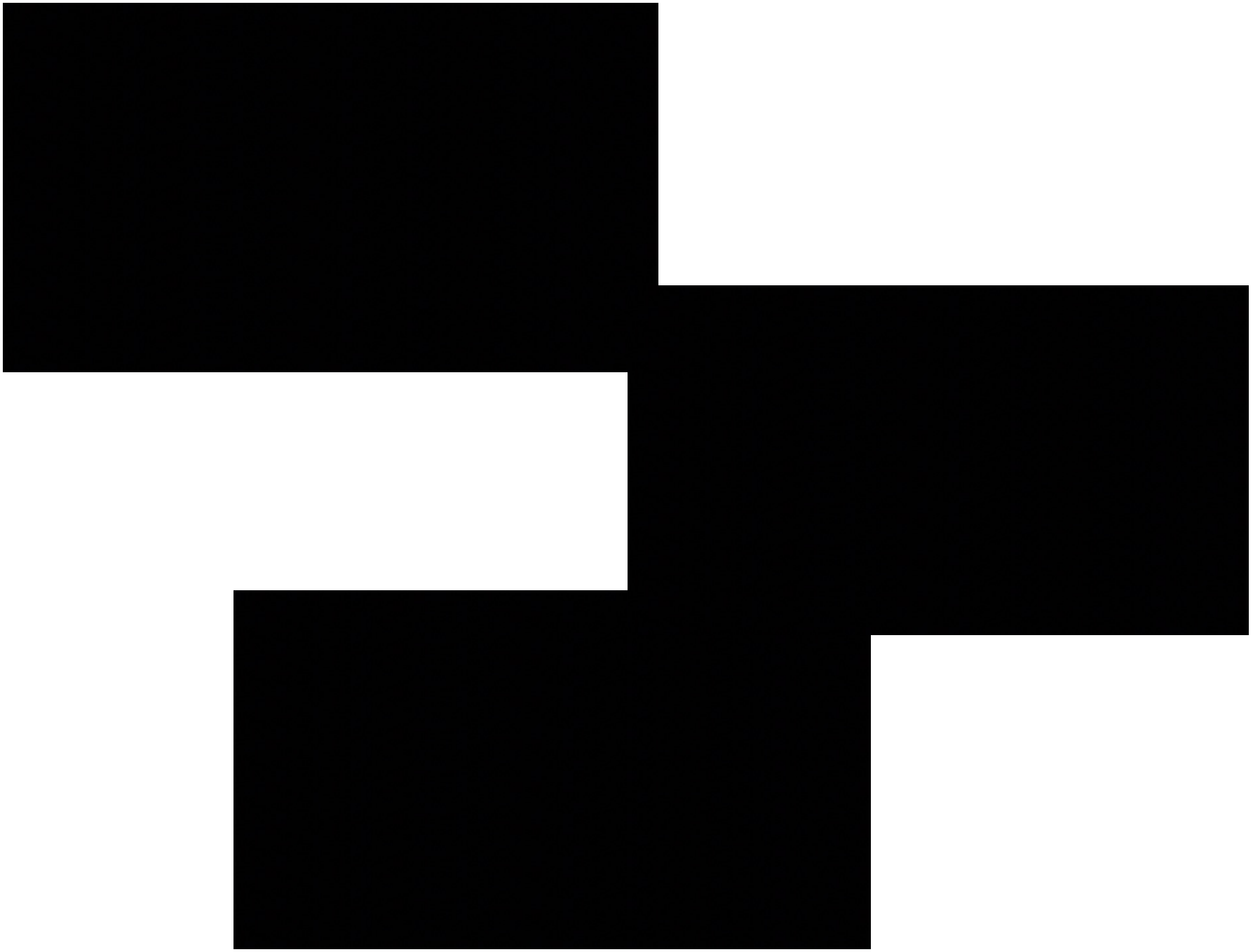


Figure : PRISMA Lab, Université de Naples




VIDEO CAMERA MODULE



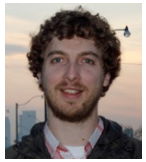
Some applications of Deep Learning

Semantic Segmentation

[Fourure, Emonet, Fromont, Muselet, Tremeau, Wolf, BMVC 2017]

Gesture recognition

Work of Natalia Neverova
LIRIS (now at Facebook AI Research)

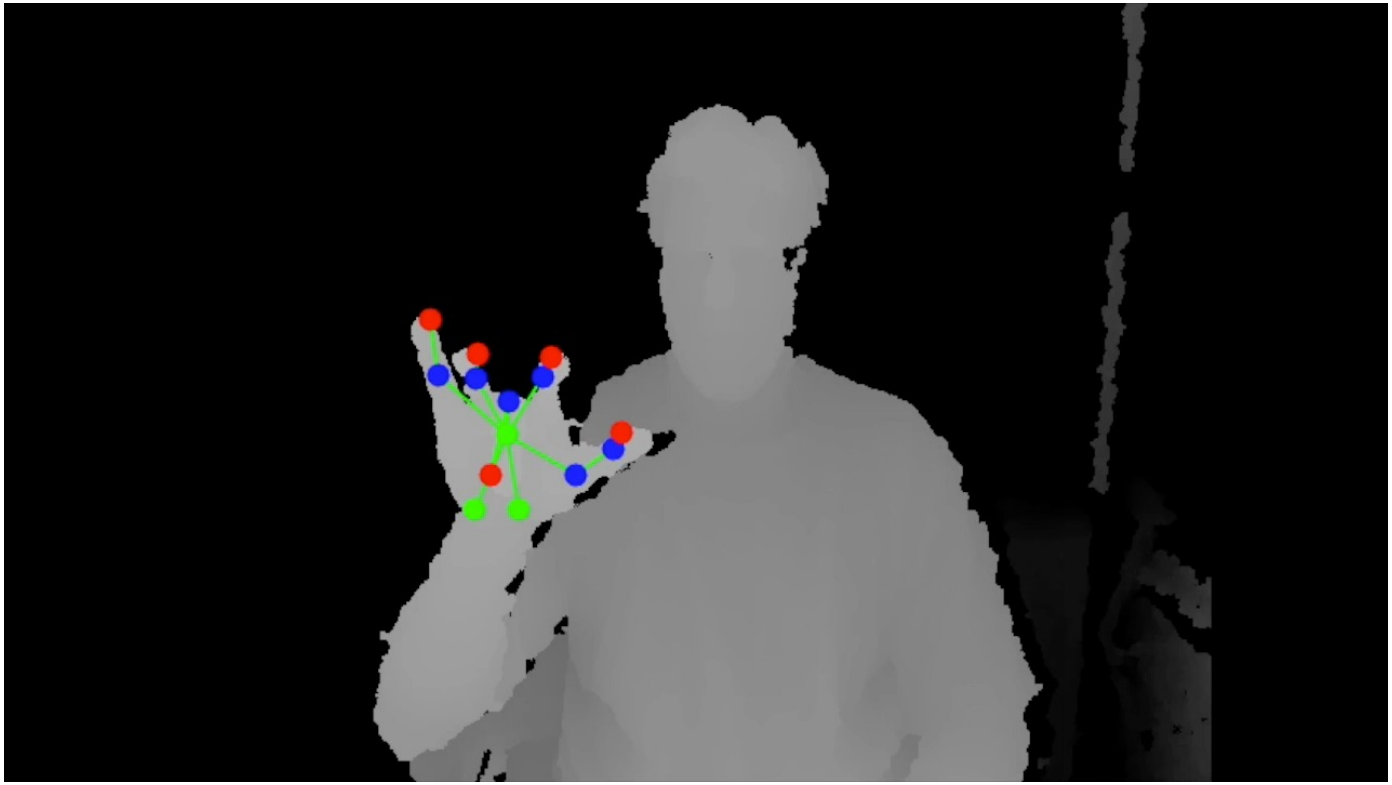


With Graham W. Taylor,
University of Guelph, Canada

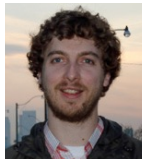
AwAbot
Beyond Robotics

[Neverova, Wolf, Taylor, Nebout, IEEE PAMI 2016]

Articulated pose estimation

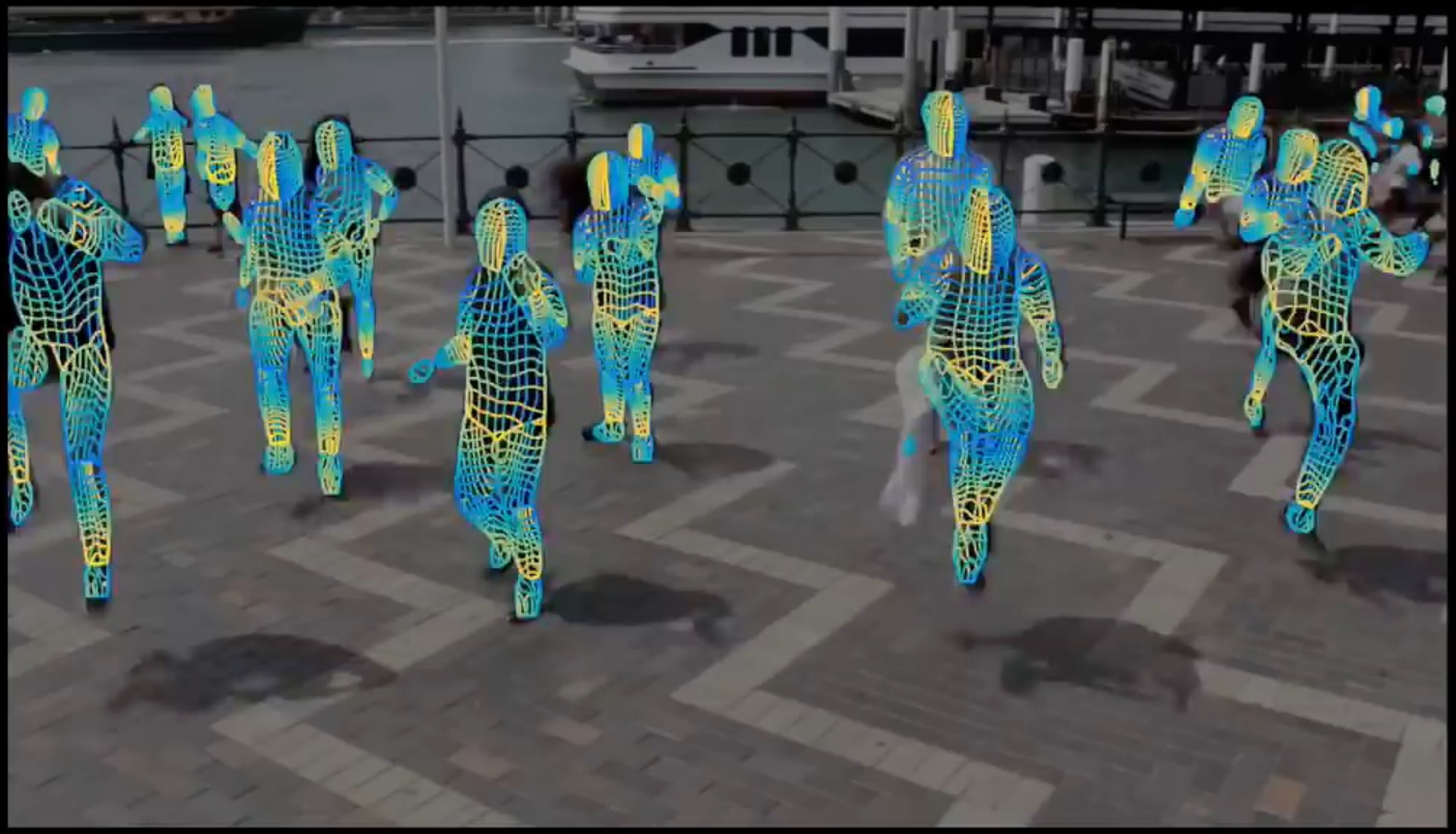


Work of Natalia Neverova
LIRIS (now at Facebook AI Research)



With Graham W. Taylor,
University of Guelph, Canada

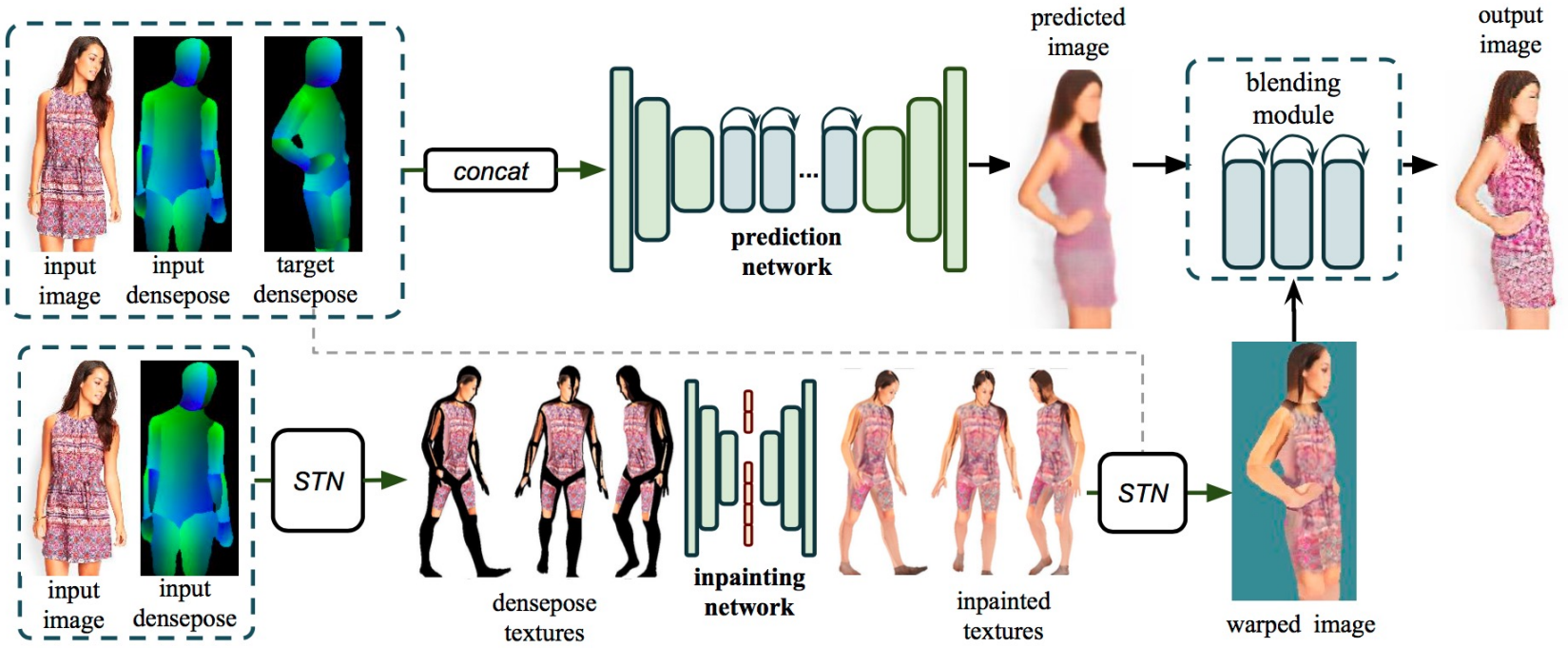
AwAbot
Beyond Robotics



Video Source: https://www.youtube.com/watch?v=2DiQUX11YaY

[Güler, Neverova, Kokkinos, CVPR 2018]

Learning to generate images



[Neverova, Güler, Kokkinos, ECCV 2018]

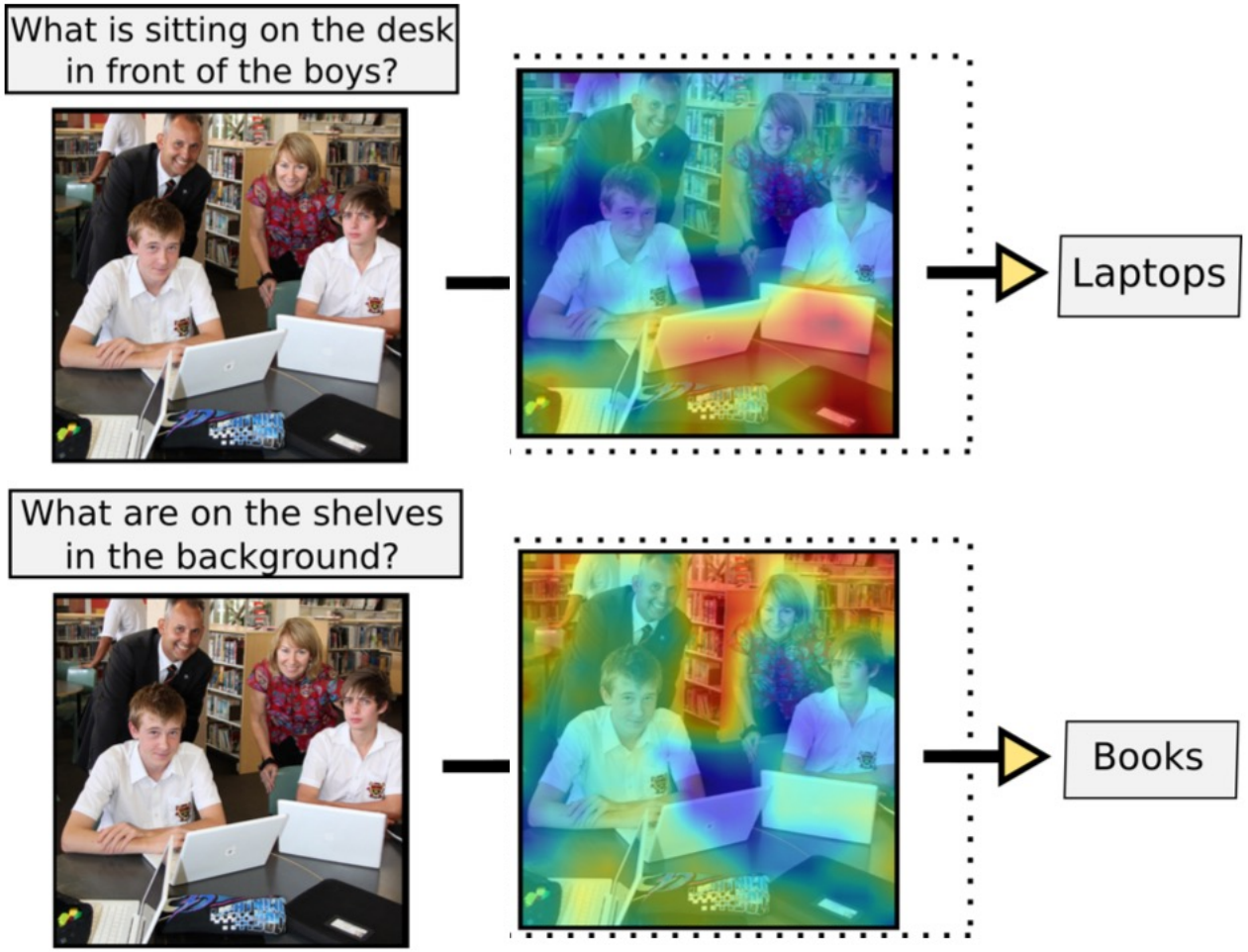
Learning to explain images

[Karpathy et al, 2015]

Visual Question Answering

“What is the moustache made of?”

Attend … to answer a question



[Ben-Younes, R. Cadene, N. Thome, M. Cord, ICCV 2017]

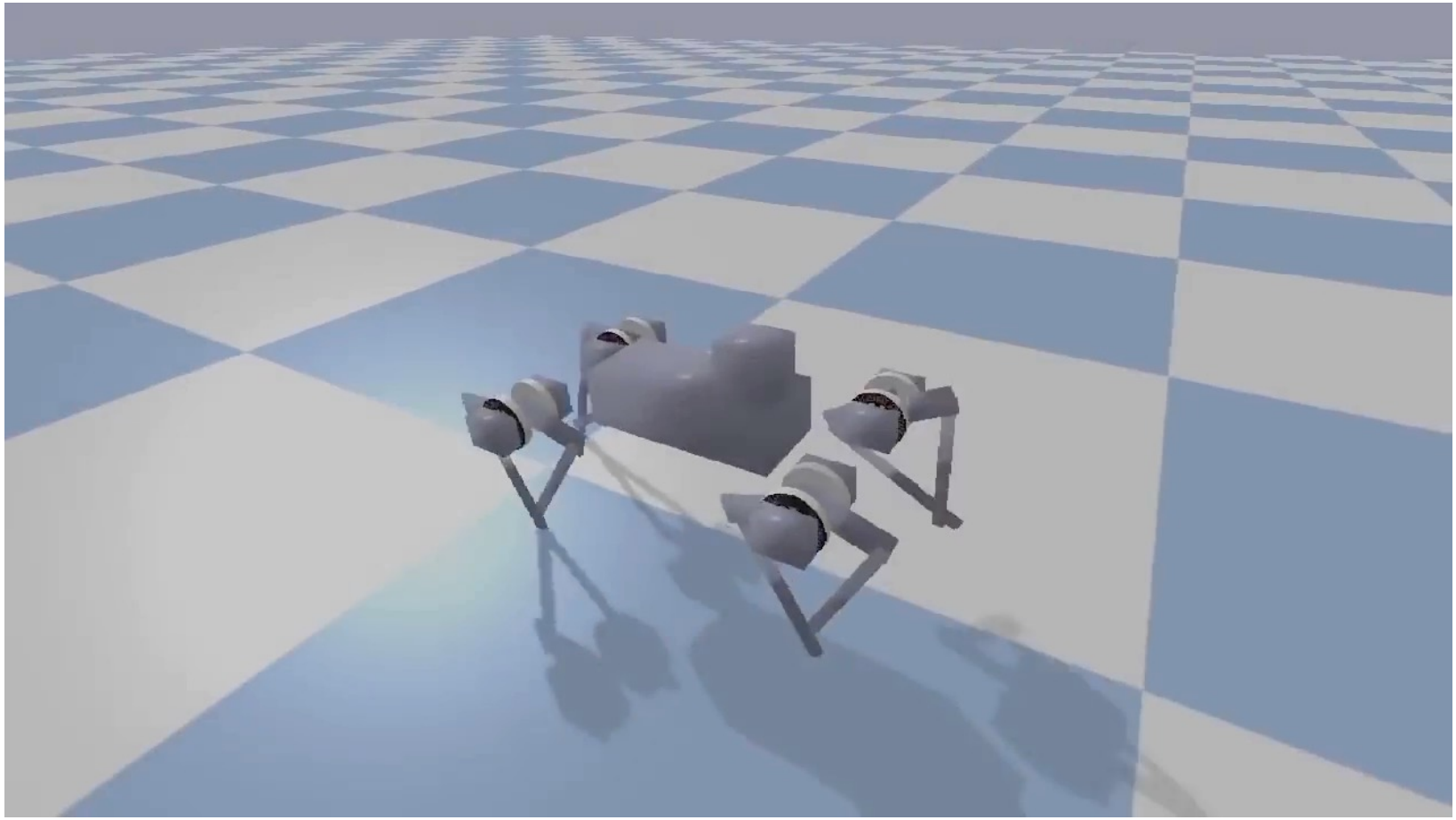
De-Oldify

https://github.com/jantic/DeOldify

Traduire l'art en photographies

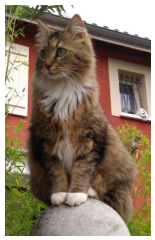
(by Denis Shiryaev, https://mymodernmet.com/denis-shiryaev-neural-network-art/)

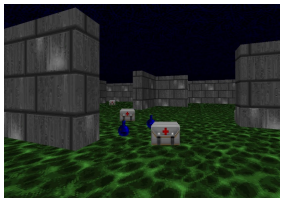
Learning to control

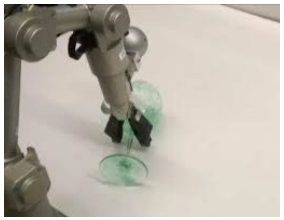


[Tan, Zhang, Coumans, Iscen, Bai, Hafner, Bohez, Vanhouke, RSS 2018]

Taking decisions







{dog, **cat**, avocado, chair, …}

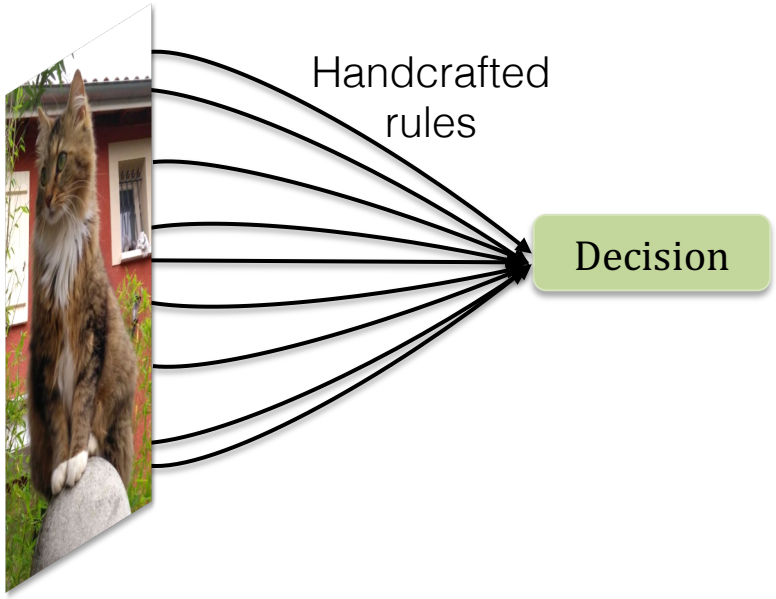
{0, 1, … 26, 27, **28**…, 98, 99, …}

{Left, right, forward, backward, …}

Motor control

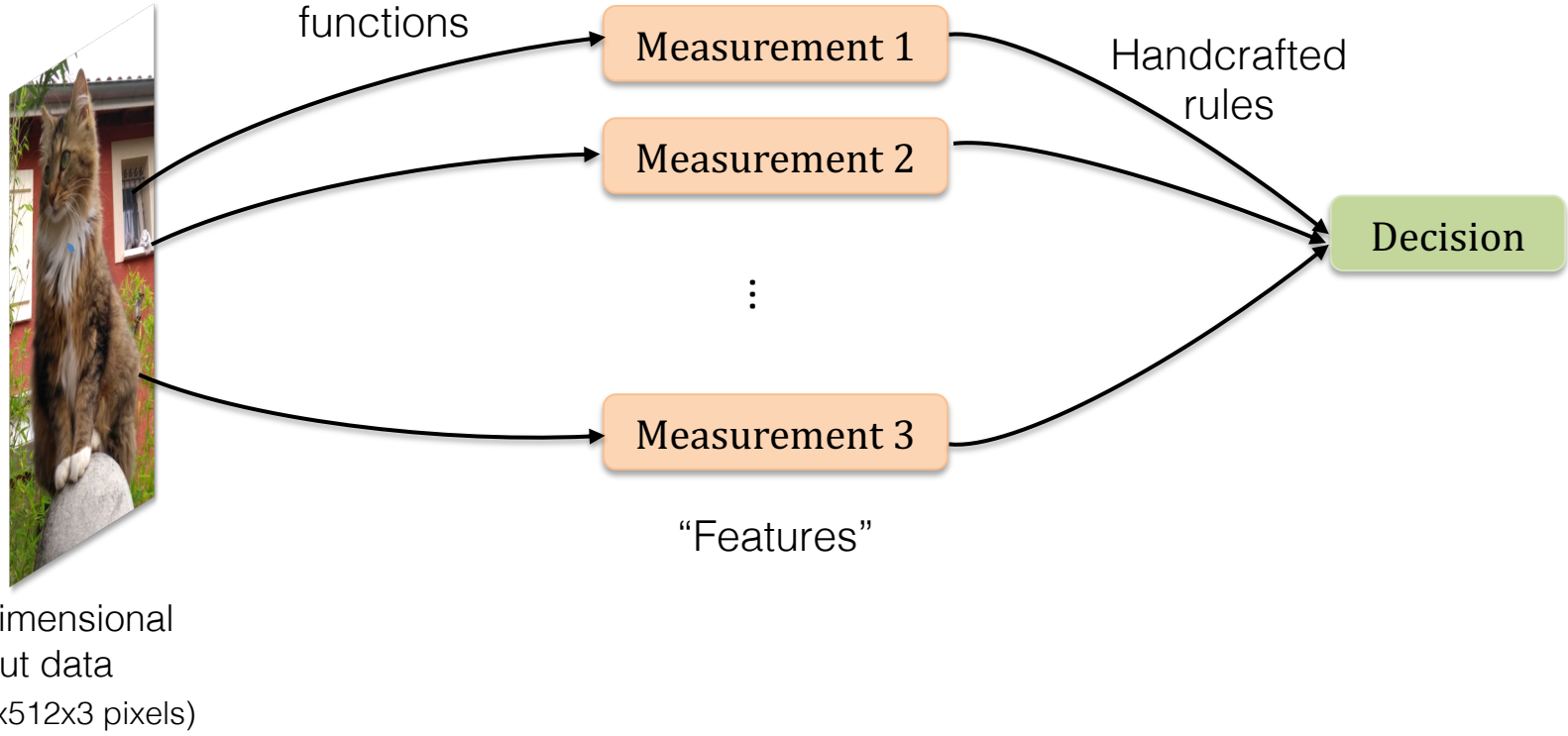
"A blue parrot with a yellow belly sitting on a branch in a forest"

Decision taking

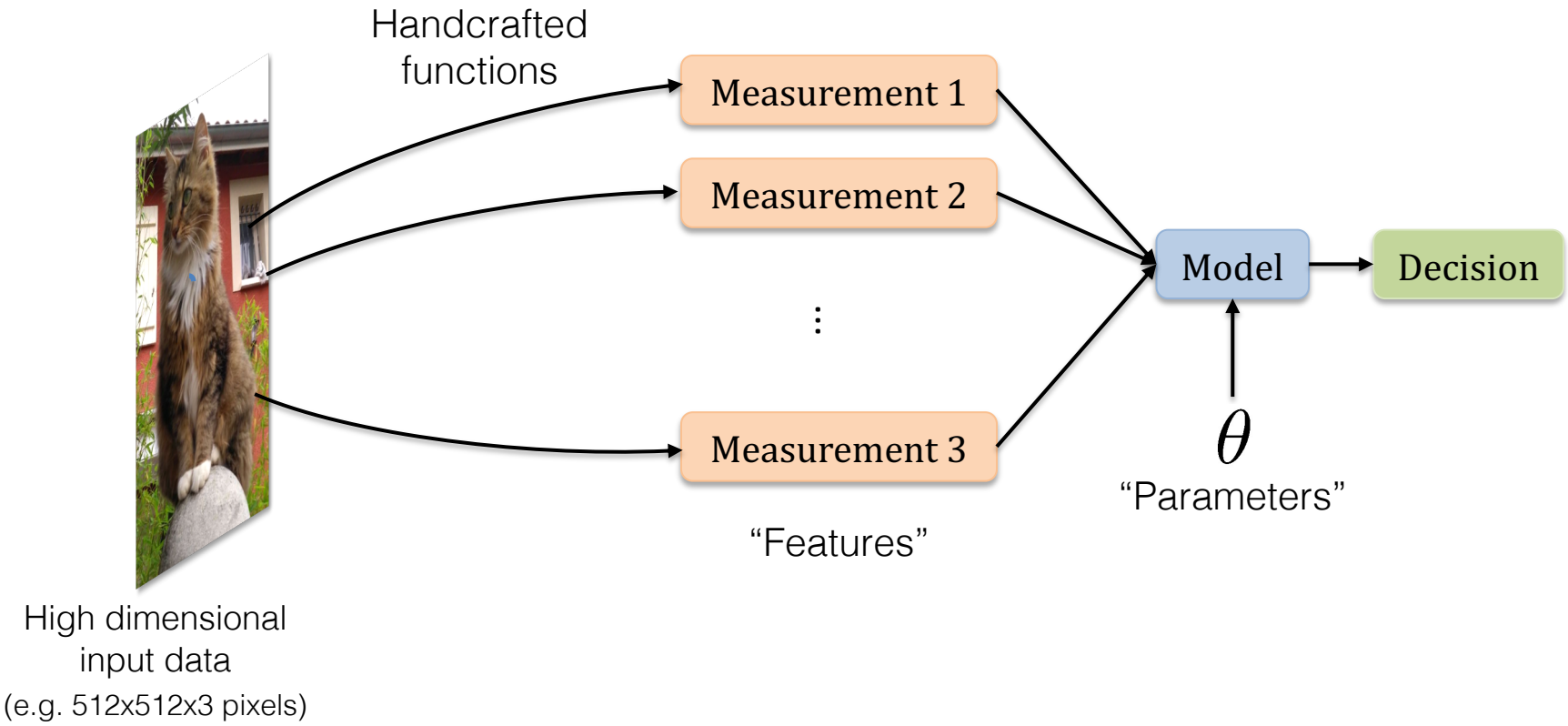


High dimensional
input data
(e.g. 512x512x3 pixels)

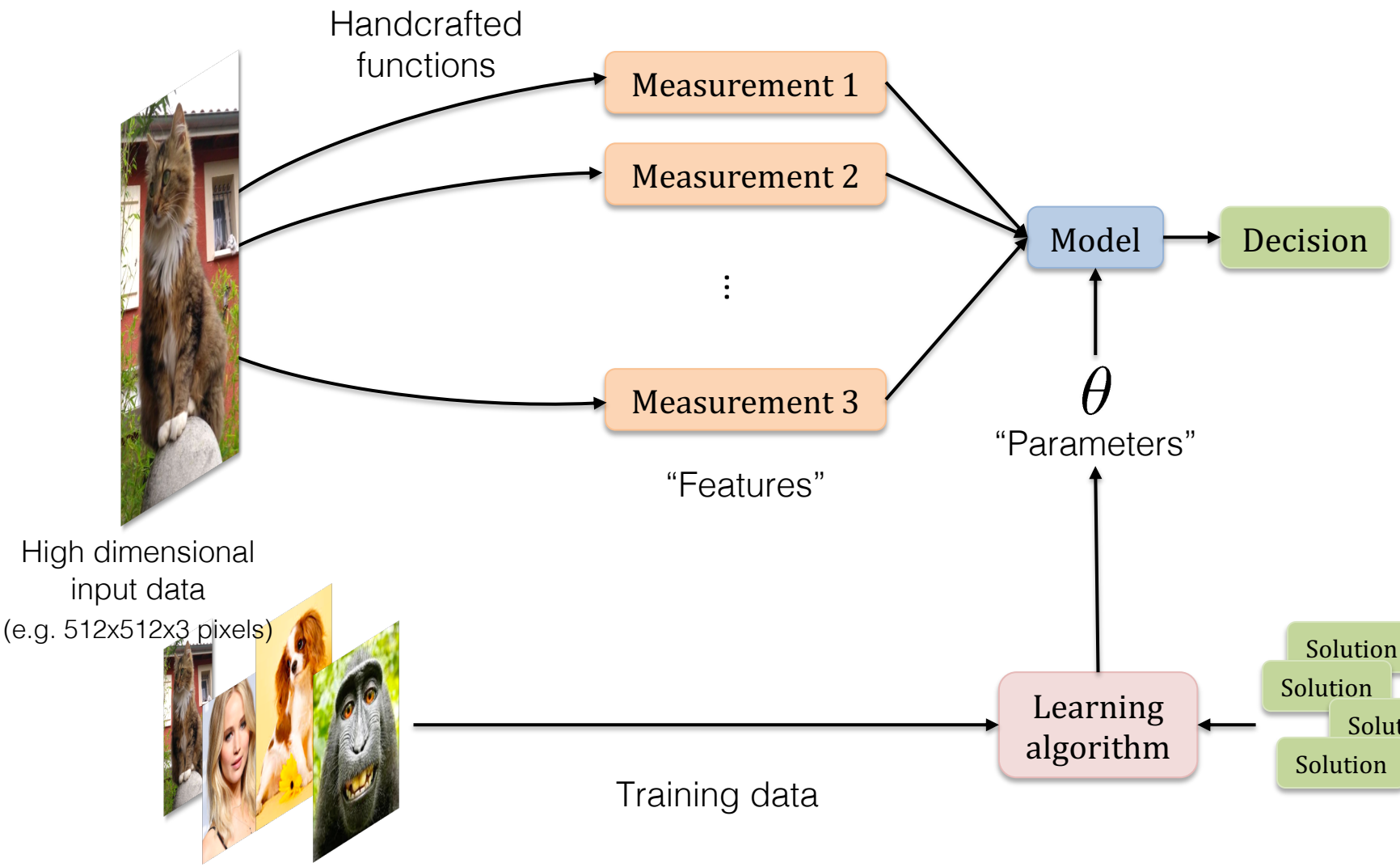
Decision taking: expert knowledge



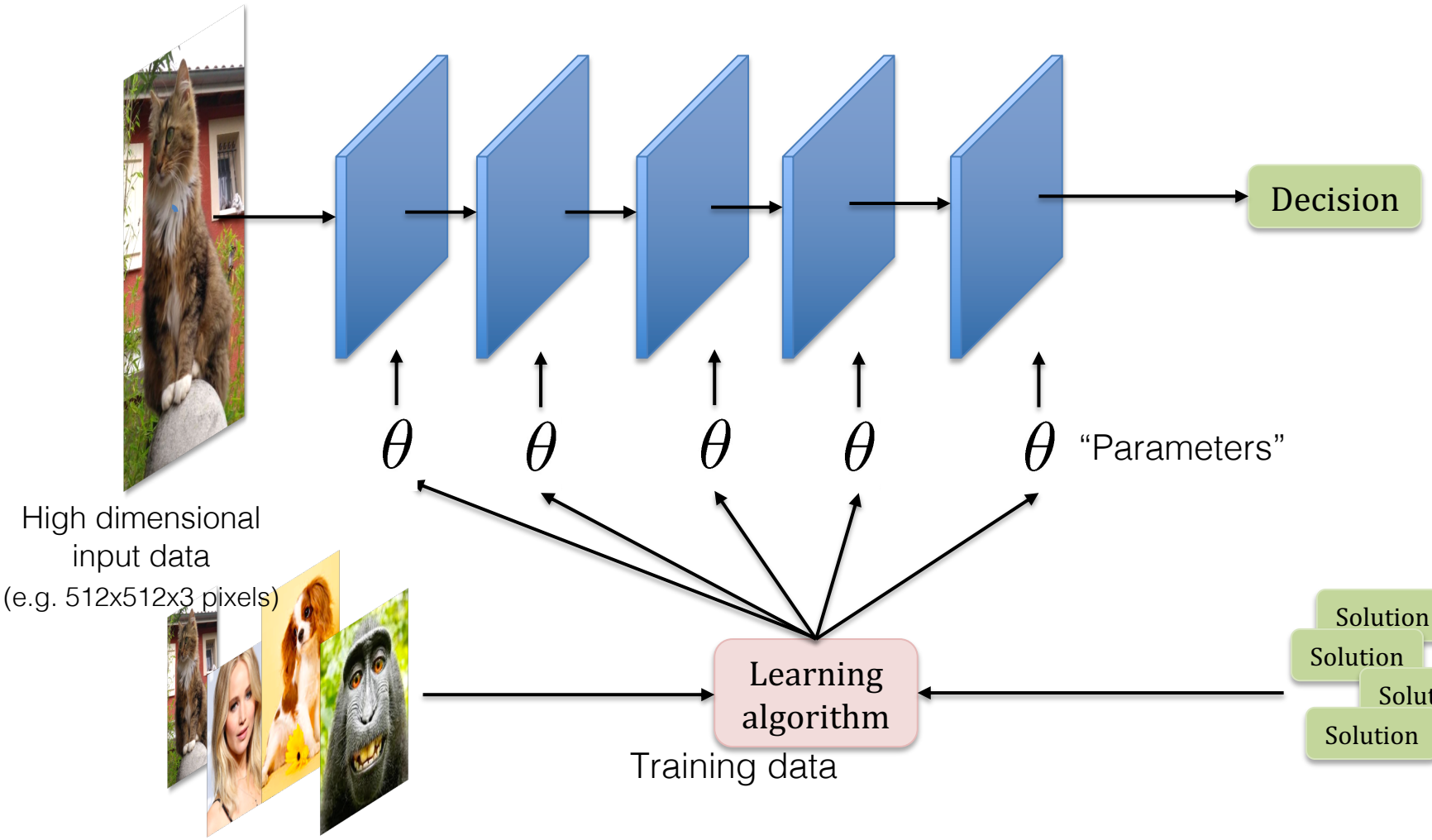
Decision taking: adding learning



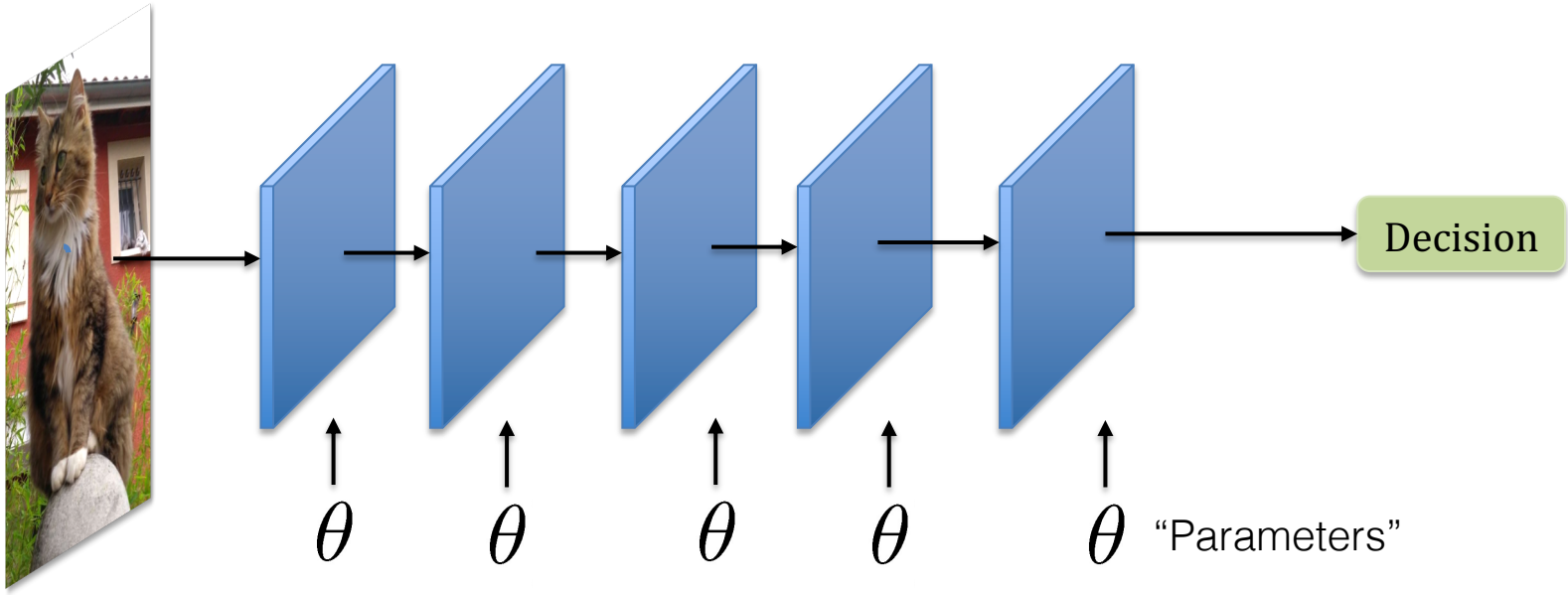
Adding machine learning



Decision taking: deep learning



Decision taking: deep learning



High dimensional input data
(e.g. 512x512x3 pixels)

Deep Learning:

- Learning from raw signals
- Hierarchical, layered representation
- Different levels of abstraction
- Learning from massive amounts of data, requiring massive compute