EECS 498-007 / 598-005
Deep Learning for Computer Vision

Lecture 1: Introduction

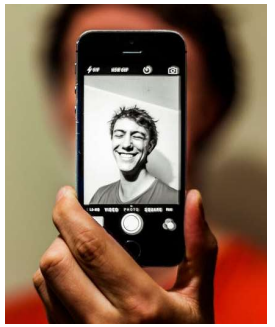
Deep Learning for Computer Vision

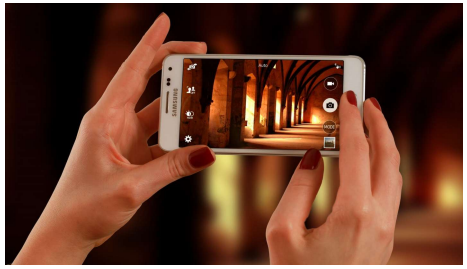
Deep Learning for Computer Vision

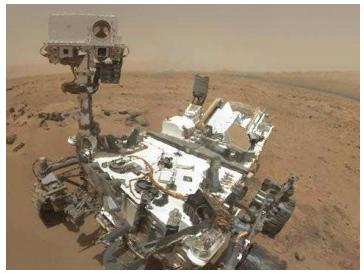
Building artificial systems
that process, perceive, and
reason about visual data

Computer Vision is everywhere!



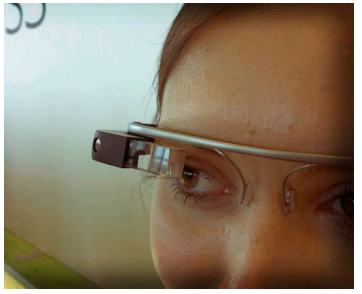











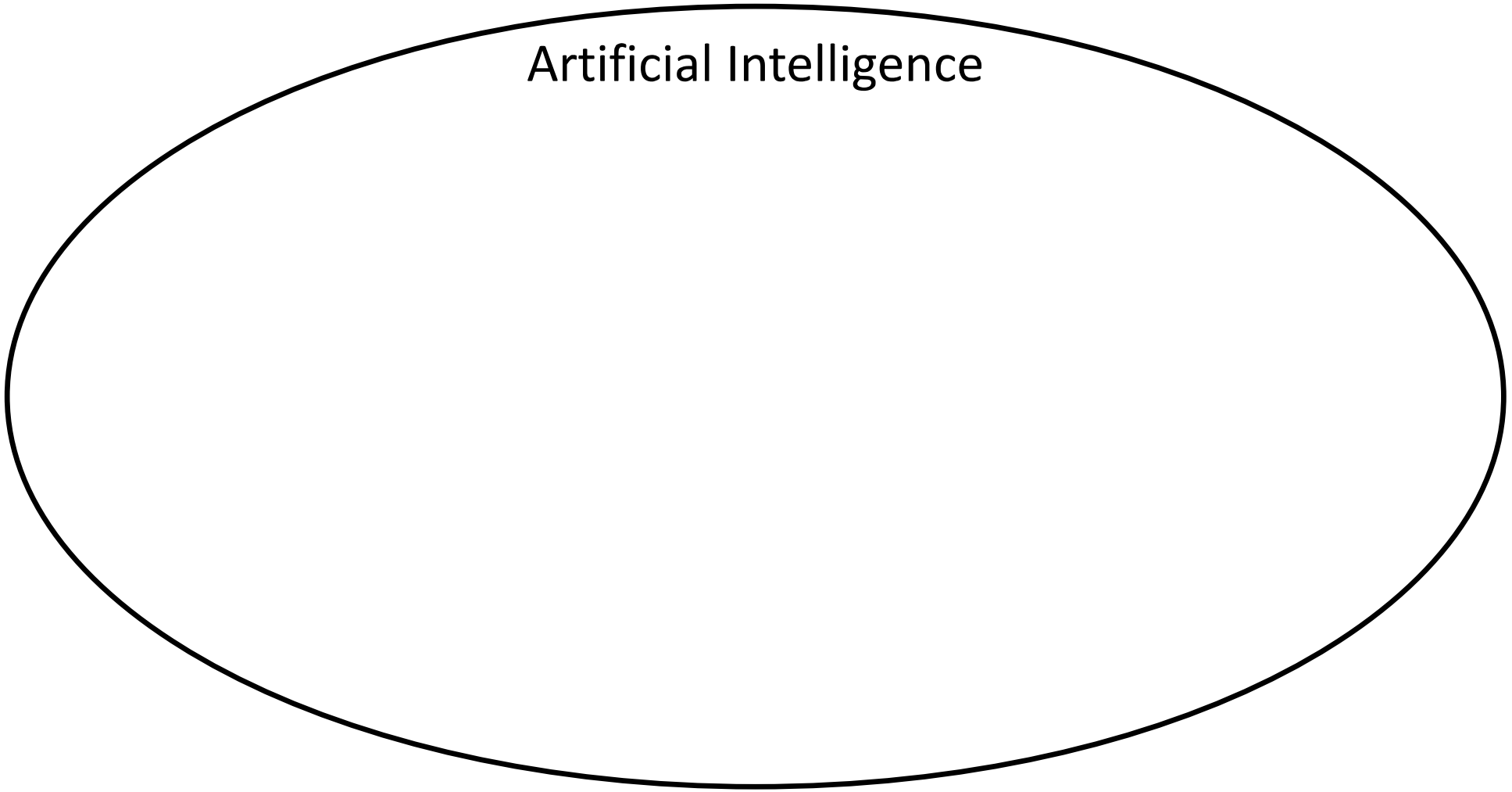


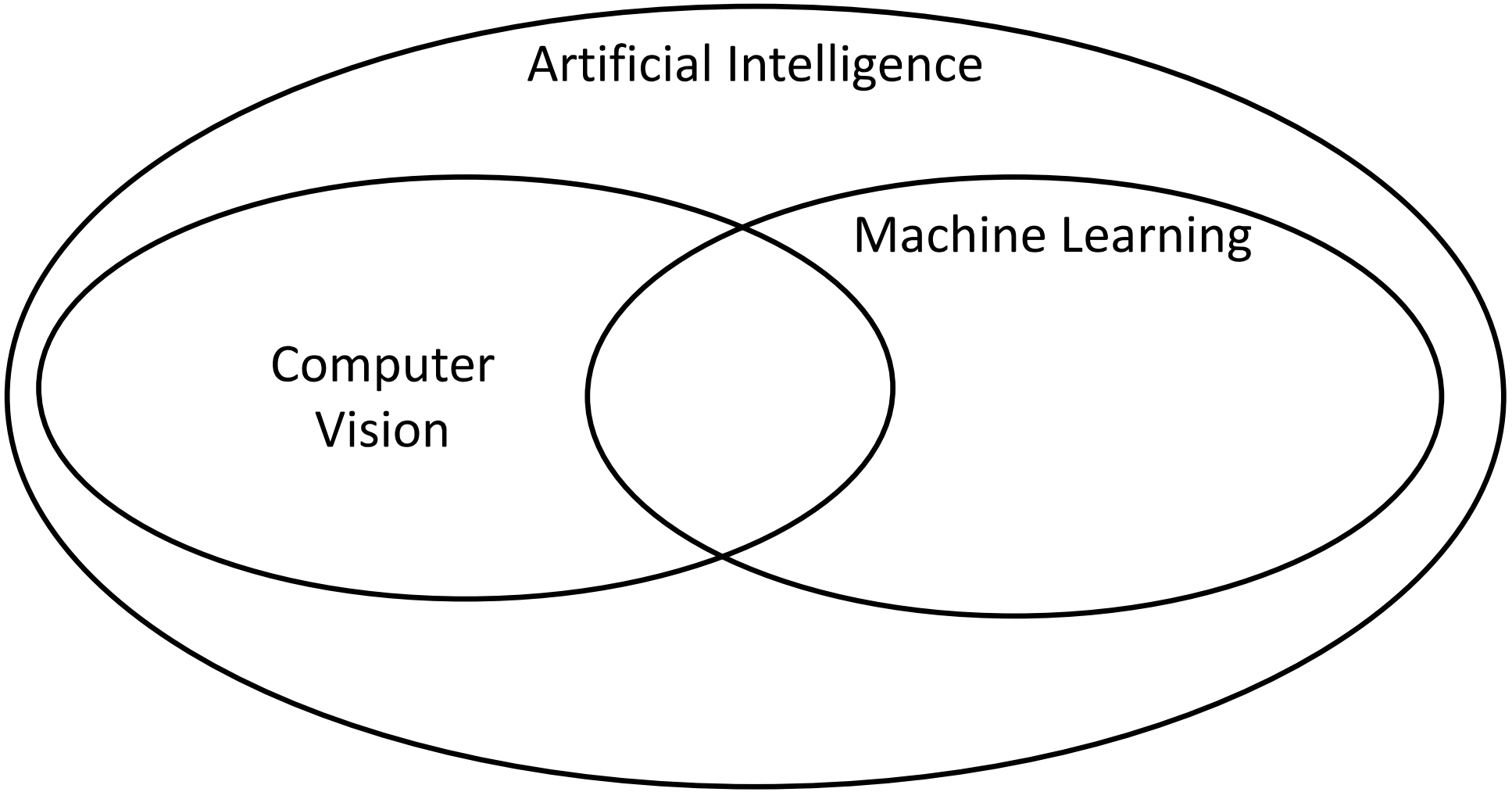
Deep Learning for Computer Vision

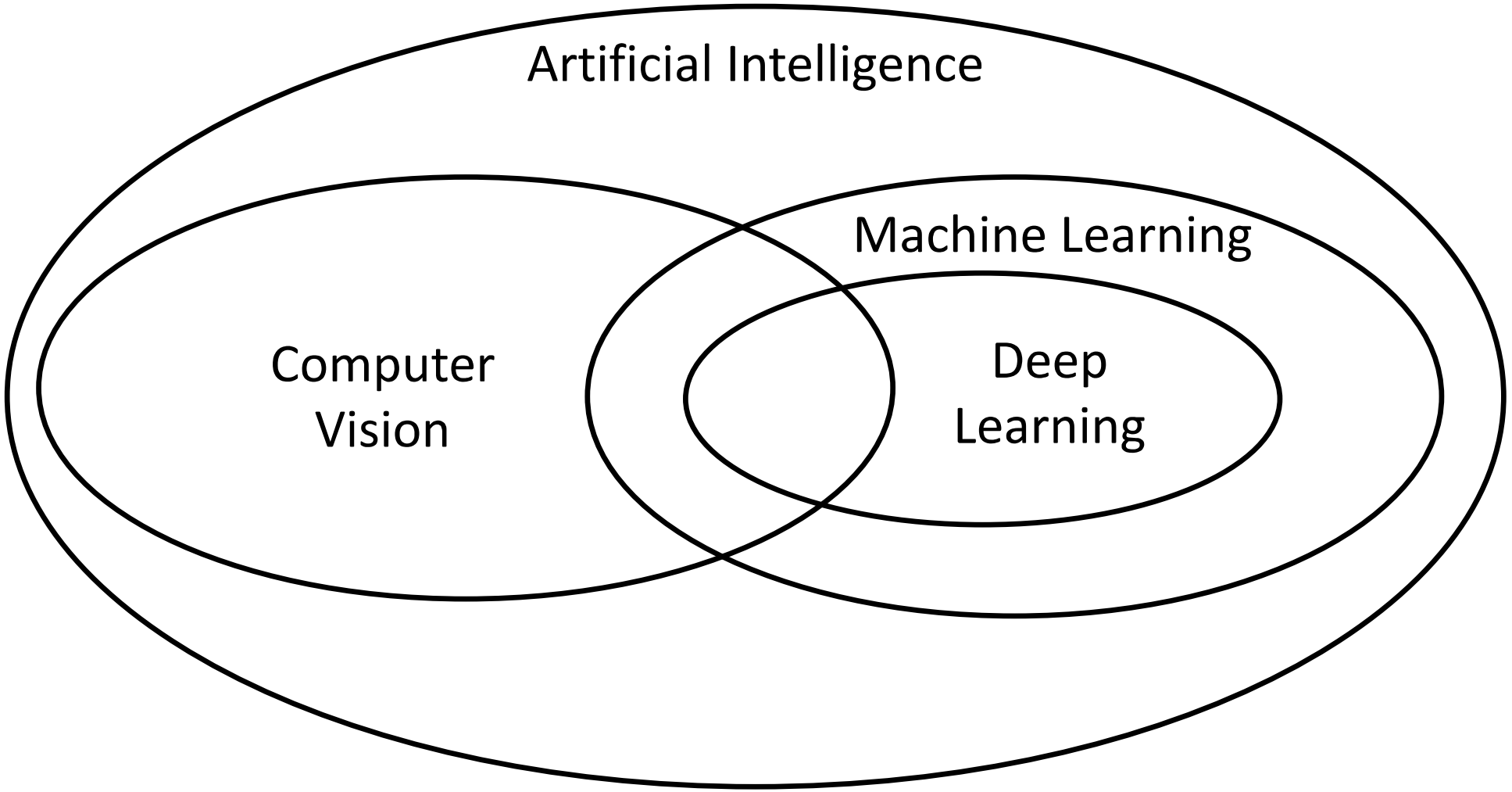
Building artificial systems that
learn from data and experience

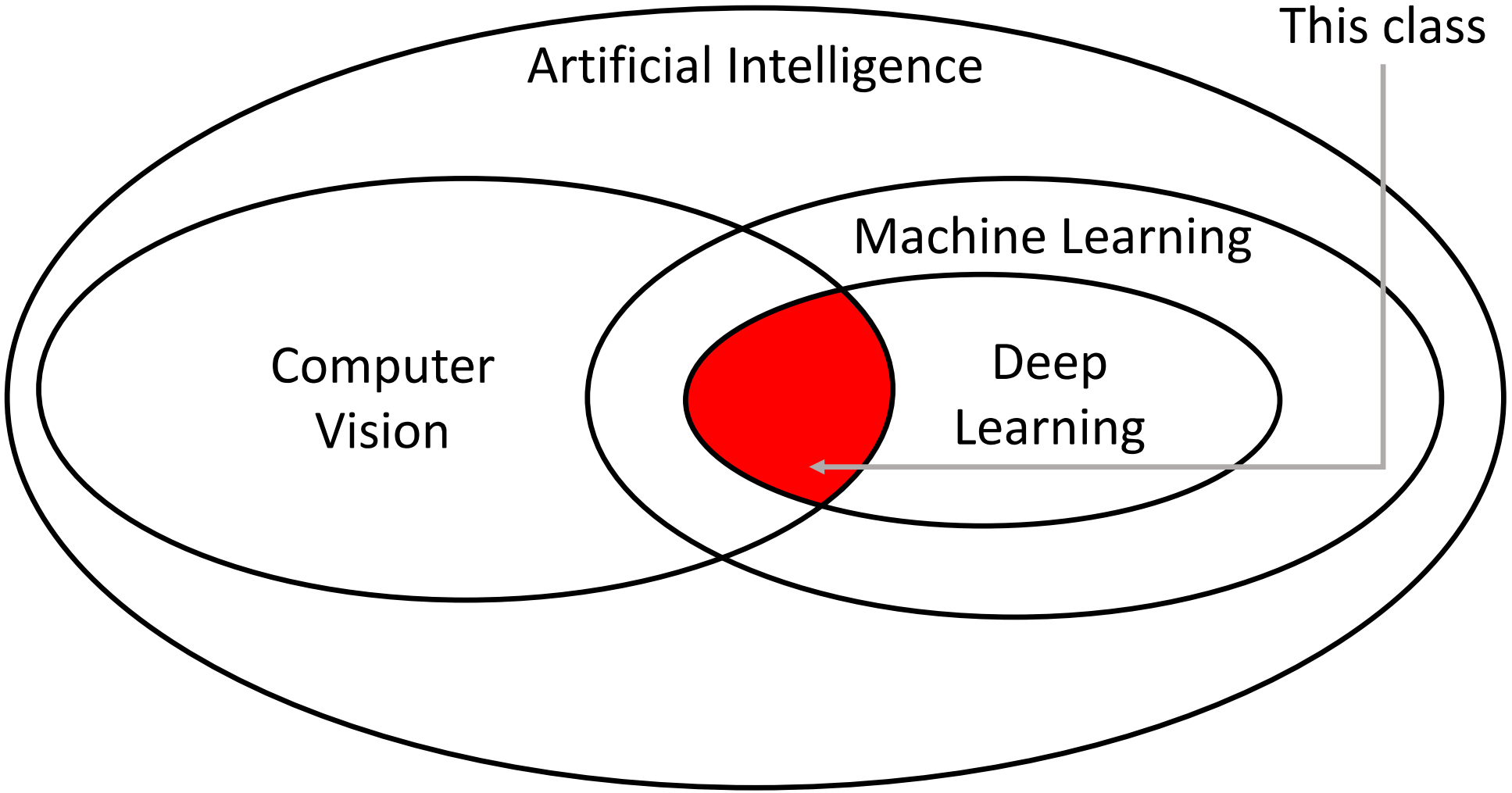
Deep Learning for Computer Vision

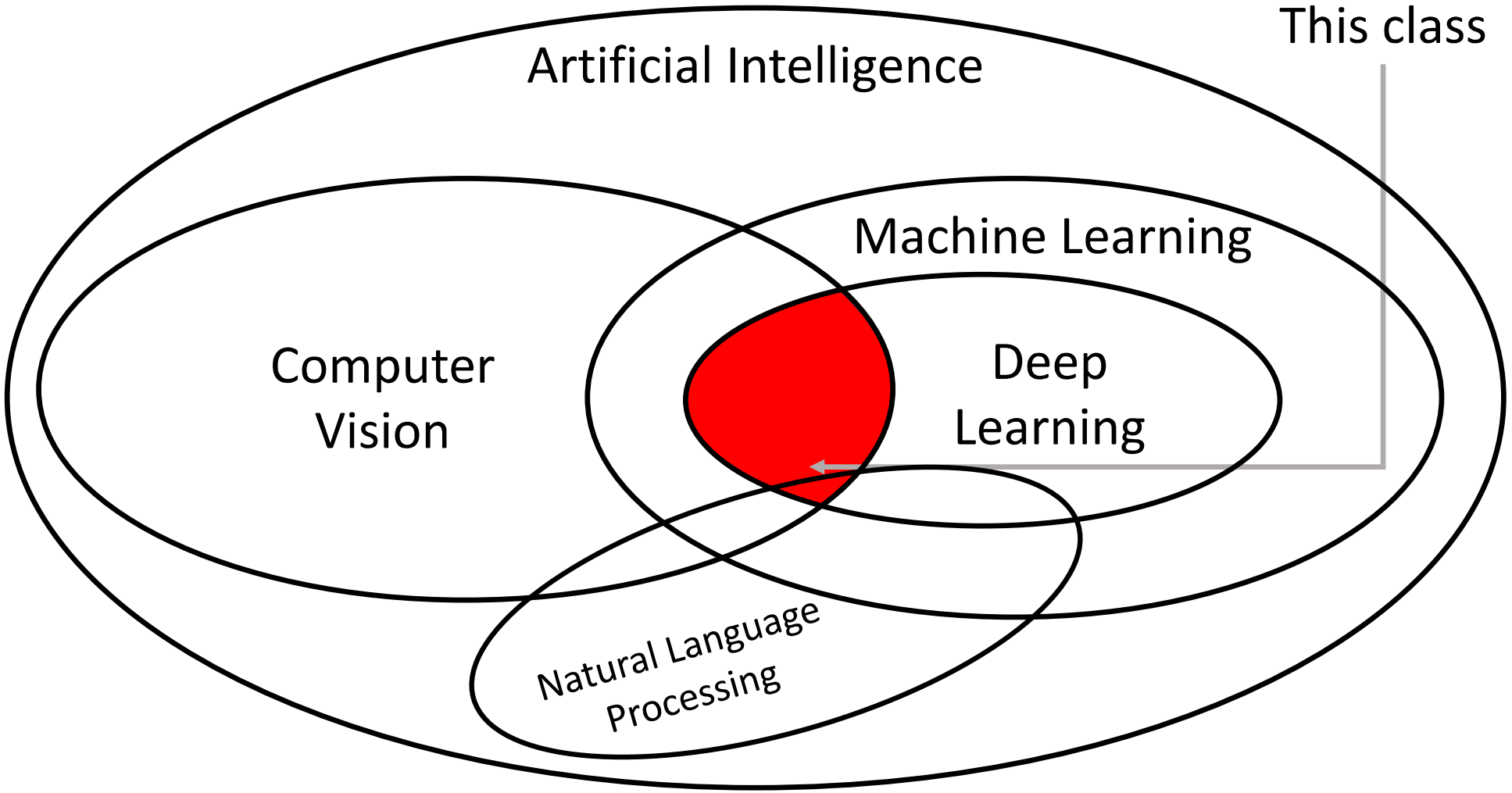
Hierarchical learning algorithms with many “layers”, (very) loosely inspired by the brain

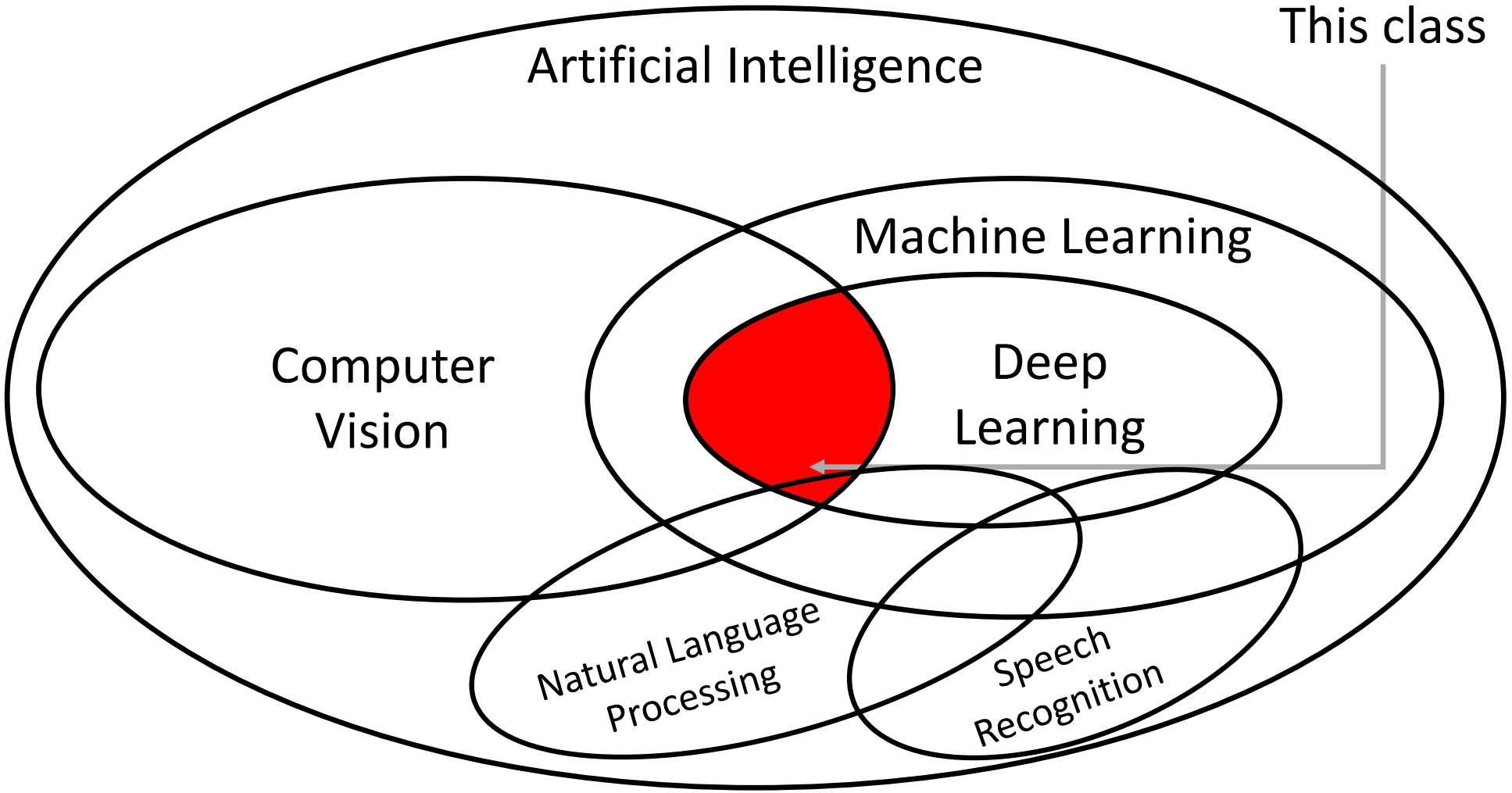

Artificial Intelligence

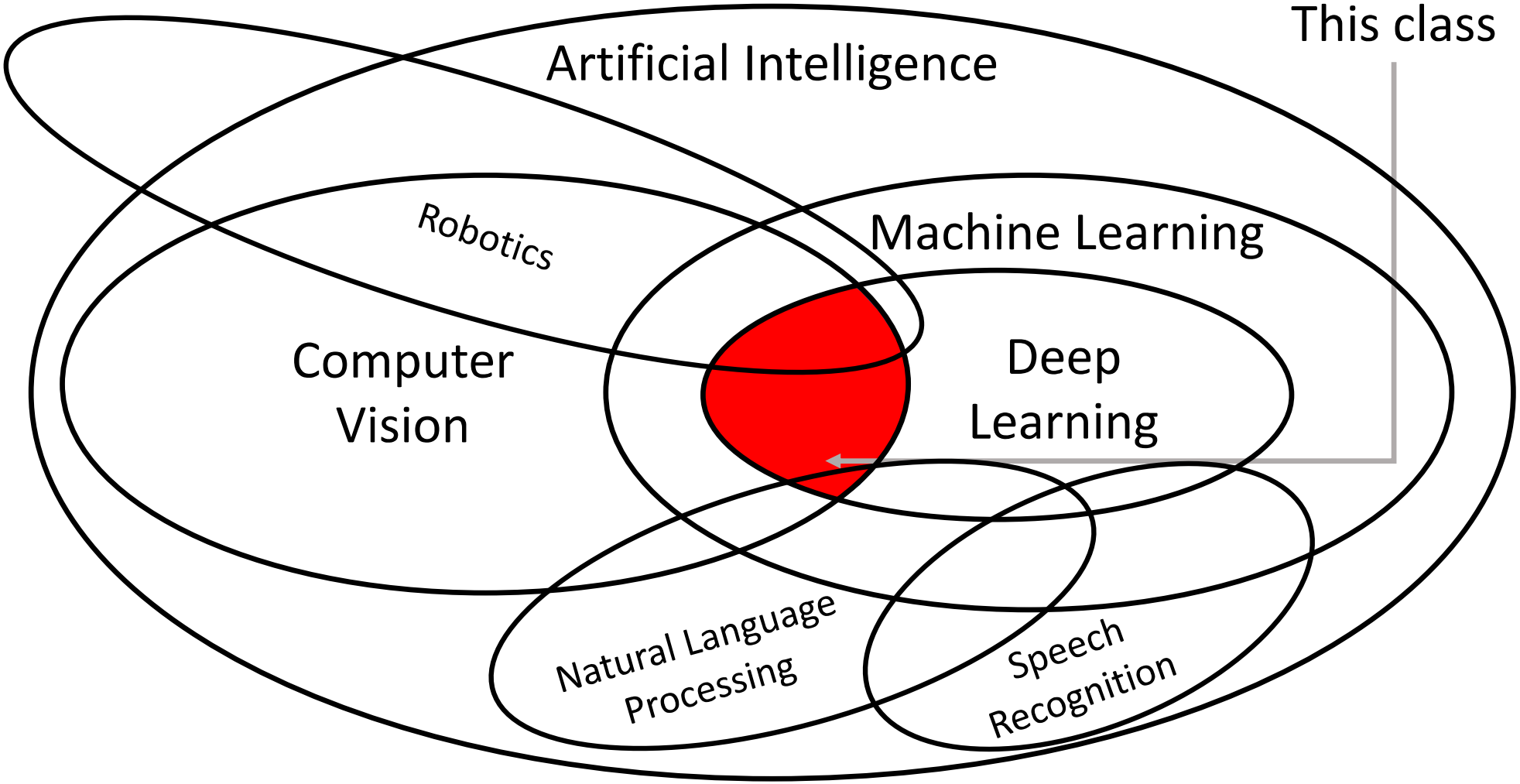

Artificial Intelligence
Machine Learning
Computer
Vision


Artificial Intelligence
Machine Learning
Computer Vision
Deep Learning


This class
Artificial Intelligence
Machine Learning
Computer Vision
Deep Learning


This class
Artificial Intelligence
Machine Learning
Computer Vision
Deep Learning
Natural Language Processing


This class
Artificial Intelligence
Machine Learning
Computer Vision
Deep Learning
Natural Language Processing
Speech Recognition


This class
Artificial Intelligence
Robotics
Machine Learning
Computer Vision
Deep Learning
Natural Language Processing
Speech Recognition

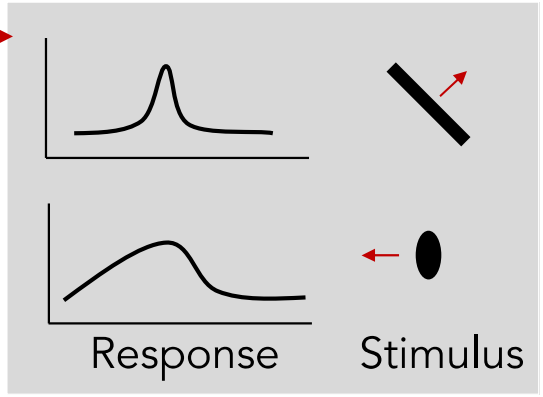
Today's Agenda

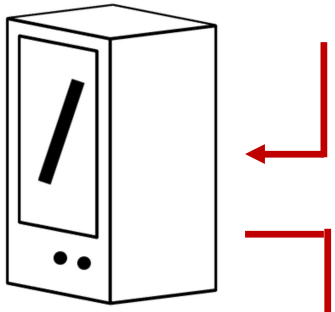
- A brief history of computer vision and deep learning
- Course overview and logistics

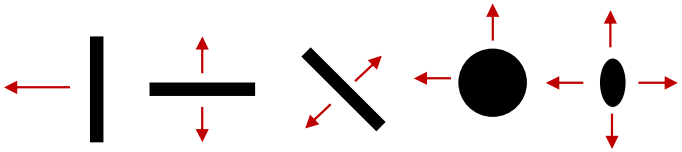
Hubel and Wiesel, 1959

Measure brain activity

1959
Hubel & Wiesel



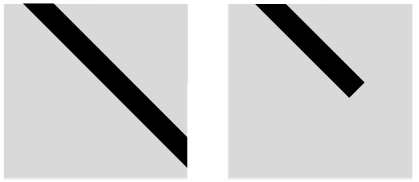




Simple cells:
Response to light orientation

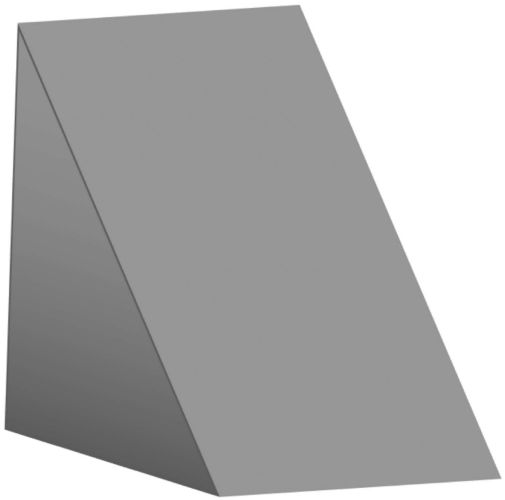
Complex cells:
Response to light orientation and movement

Hypercomplex cells:
response to movement with an end point

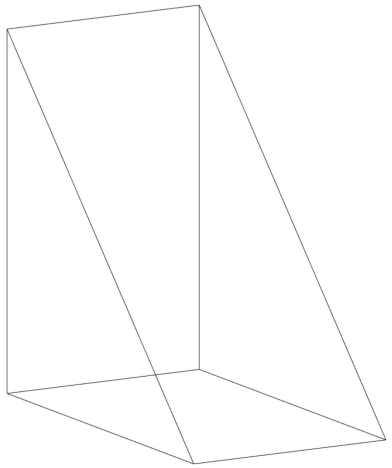


No response

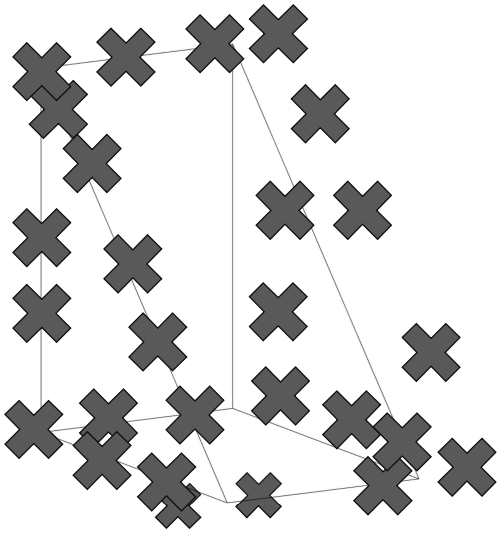
Larry Roberts, 1963



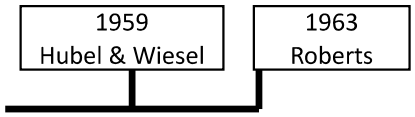
(a) Original picture



(b) Differentiated picture



(c) Feature points selected



MASSACHUSETTS INSTITUTE OF TECHNOLOGY

PROJECT MAC

Artificial Intelligence Group — July 7, 1966
Vision Memo. No. 100.

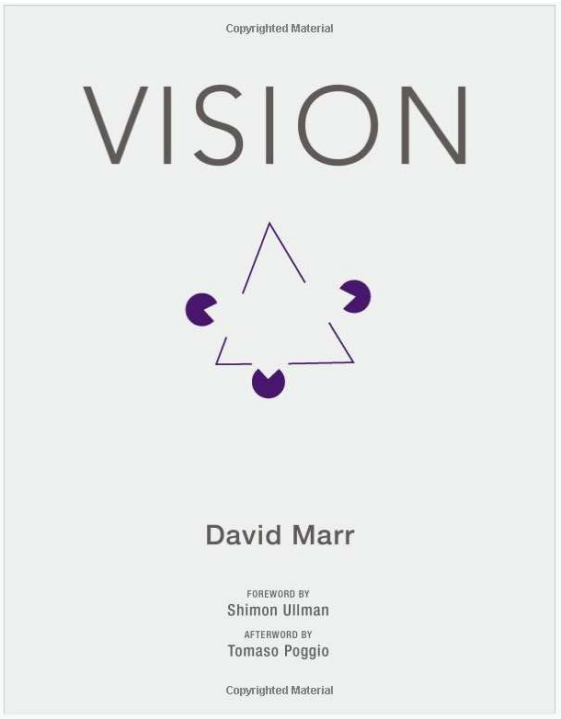
THE SUMMER VISION PROJECT

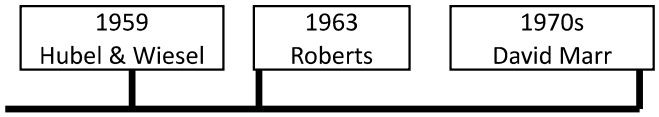
Seymour Papert

The summer vision project is an attempt to use our summer workers effectively in the construction of a significant part of a visual system. The particular task was chosen partly because it can be segmented into sub-problems which will allow individuals to work independently and yet participate in the construction of a system complex enough to be a real landmark in the development of "pattern recognition".

1959 Hubel & Wiesel

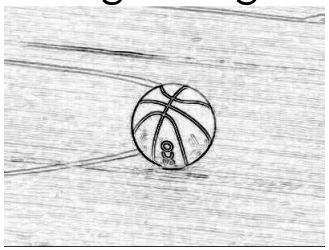
1963 Roberts



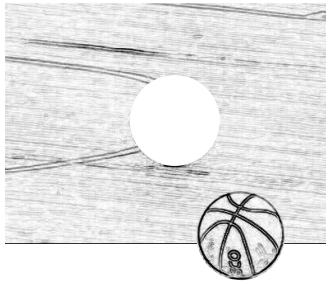


Input image

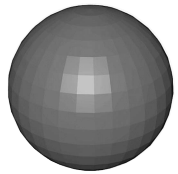
Edge image

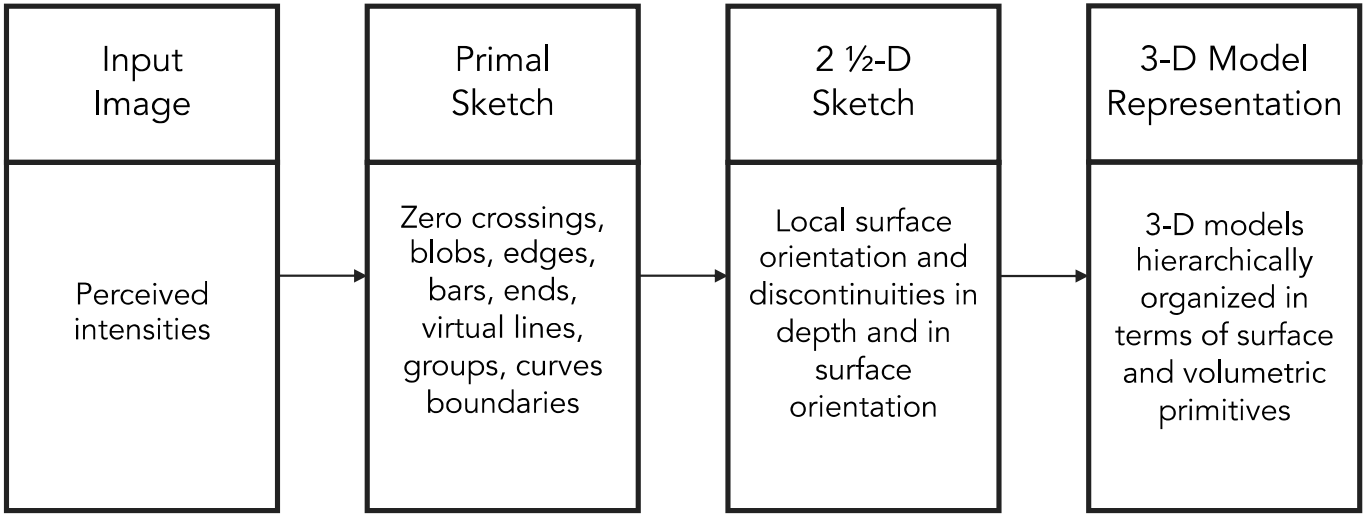


2 ½-D sketch



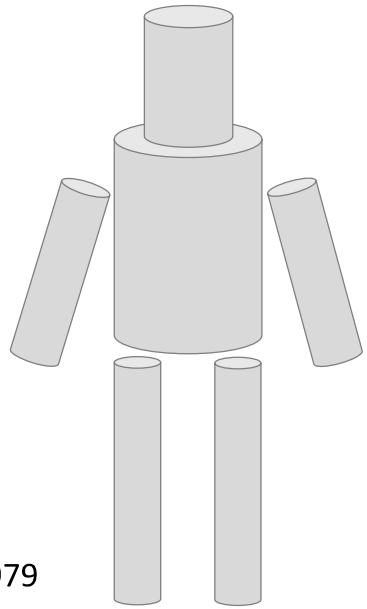
3-D model



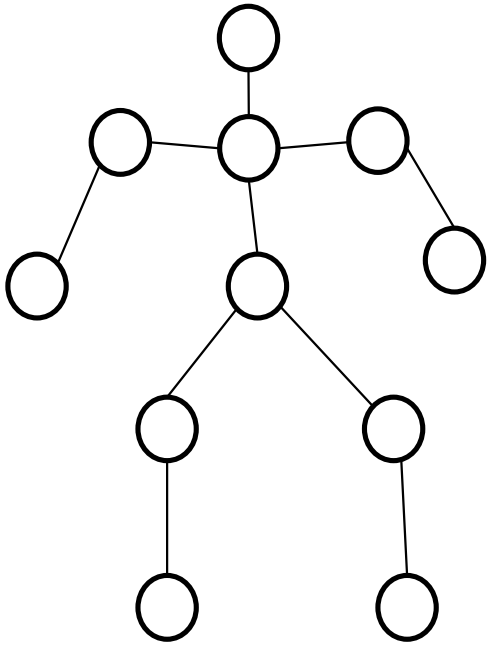


Stages of Visual Representation, David Marr, 1970s

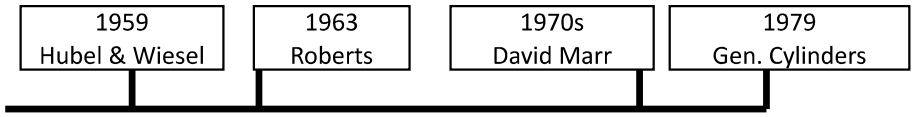
Recognition via Parts (1970s)



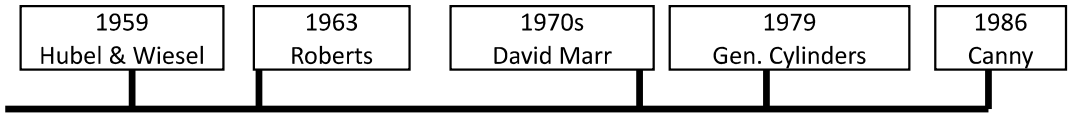
Generalized Cylinders,
Brooks and Binford, 1979



Pictorial Structures,
Fischler and Elshlager, 1973

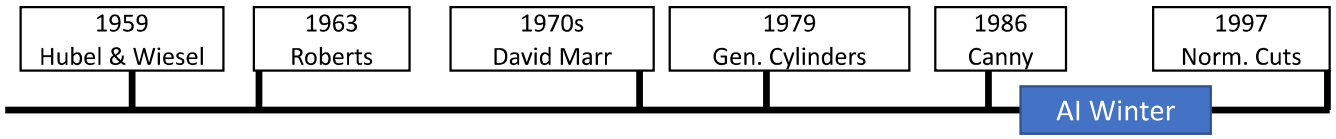


Recognition via Edge Detection (1980s)



John Canny, 1986
David Lowe, 1987

Recognition via Grouping (1990s)

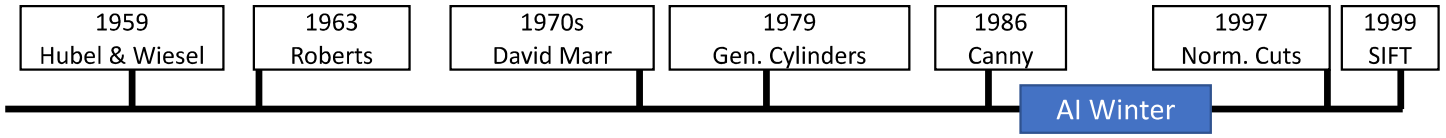


Normalized Cuts, Shi and Malik, 1997

Recognition via Matching (2000s)

Image is public domain

Image is public domain

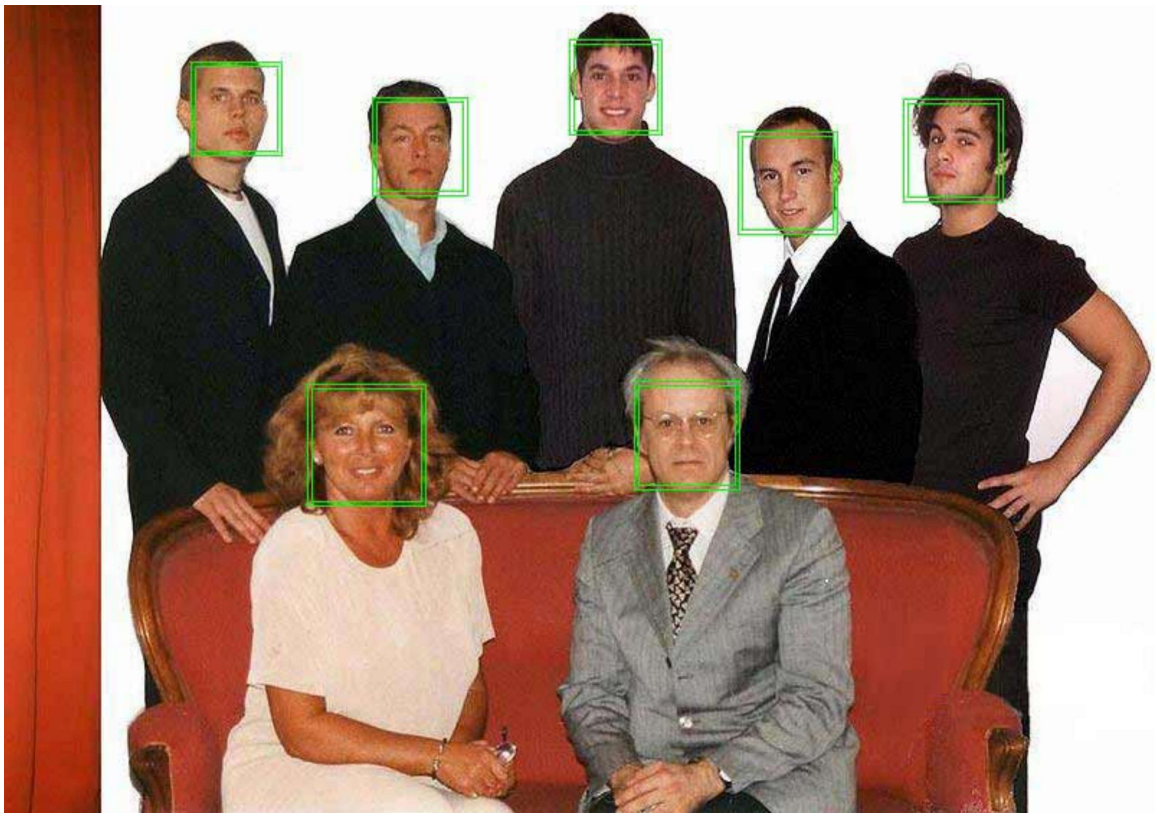


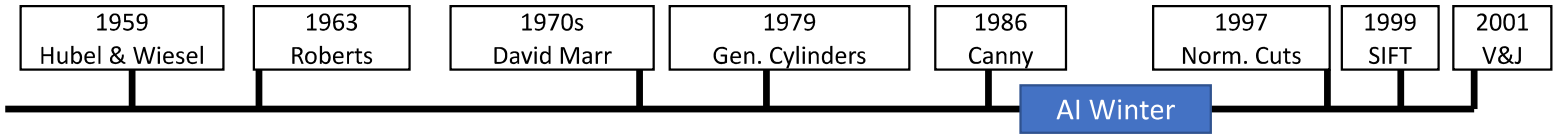
SIFT, David Lowe, 1999

Face Detection

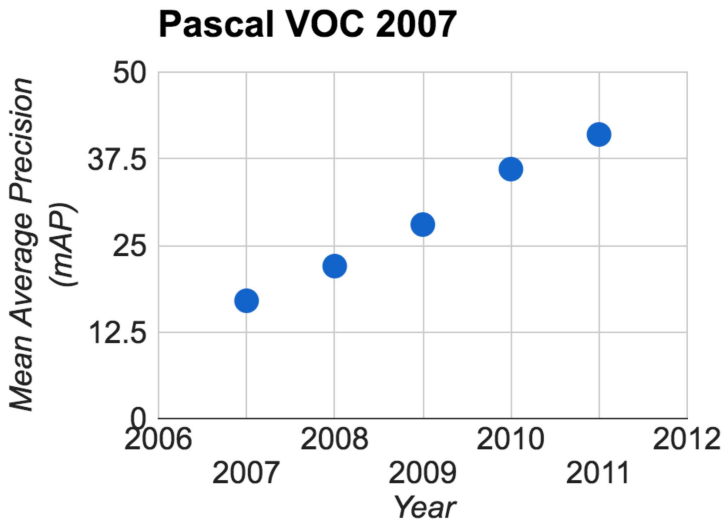
Viola and Jones, 2001

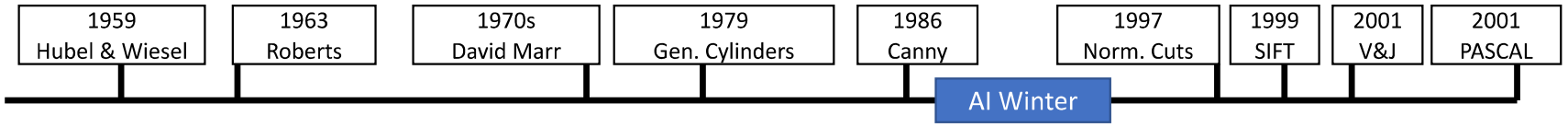
One of the first successful applications of machine learning to vision





PASCAL Visual Object Challenge





IMAGENET Large Scale Visual Recognition Challenge

The Image Classification Challenge:

1,000 object classes

1,431,167 images

Output:

Scale

T-shirt

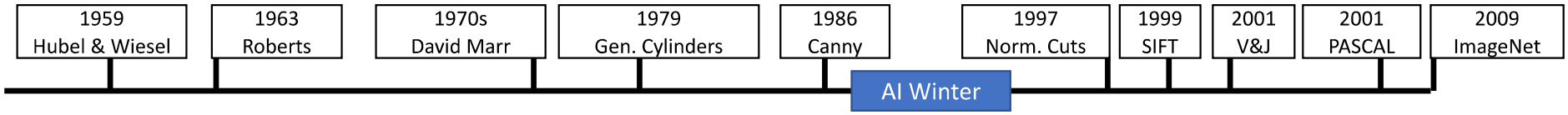
Steel drum

Drumstick

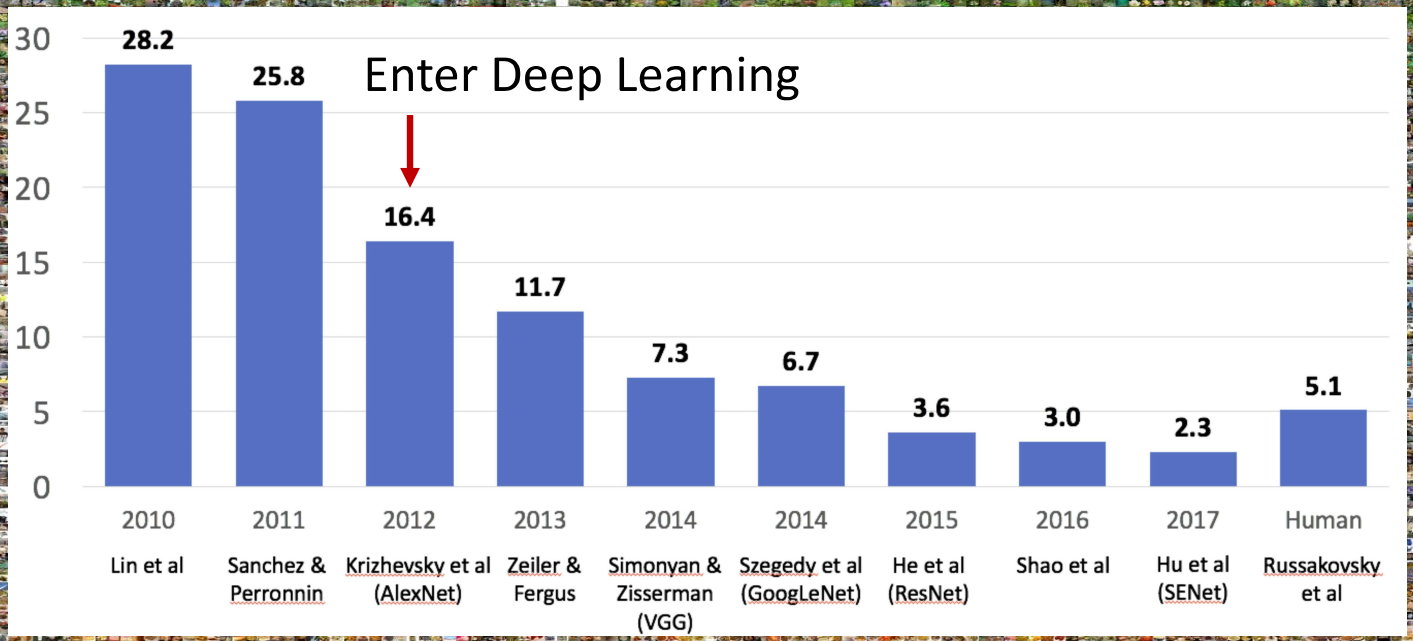
Mud turtle

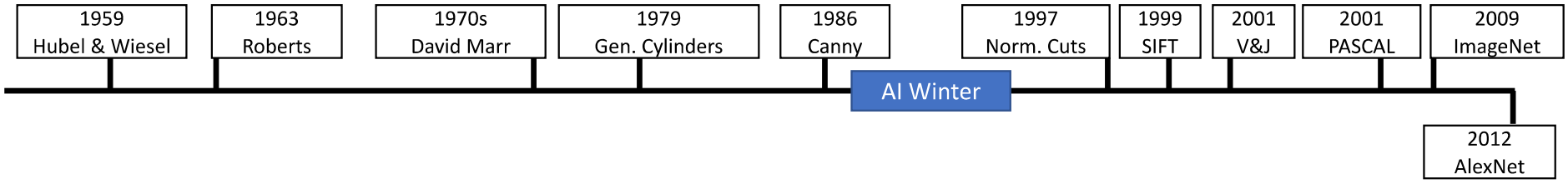
Deng et al, 2009

Russakovsky et al. IJCV 2015

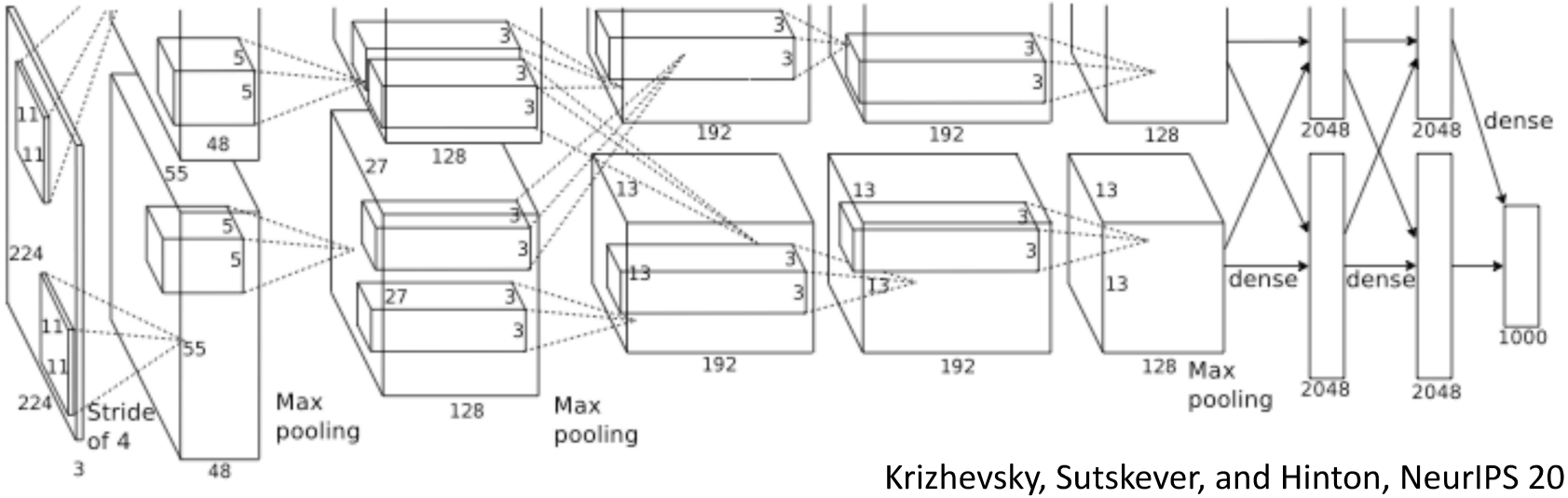


IMAGENET Large Scale Visual Recognition Challenge

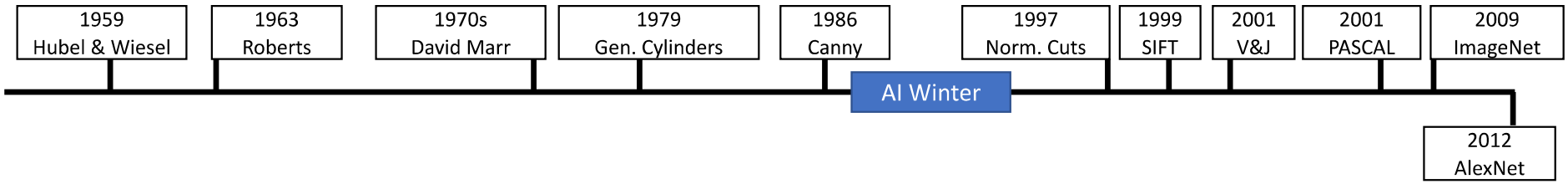




AlexNet: Deep Learning Goes Mainstream



Krizhevsky, Sutskever, and Hinton, NeurIPS 2012



Perceptron

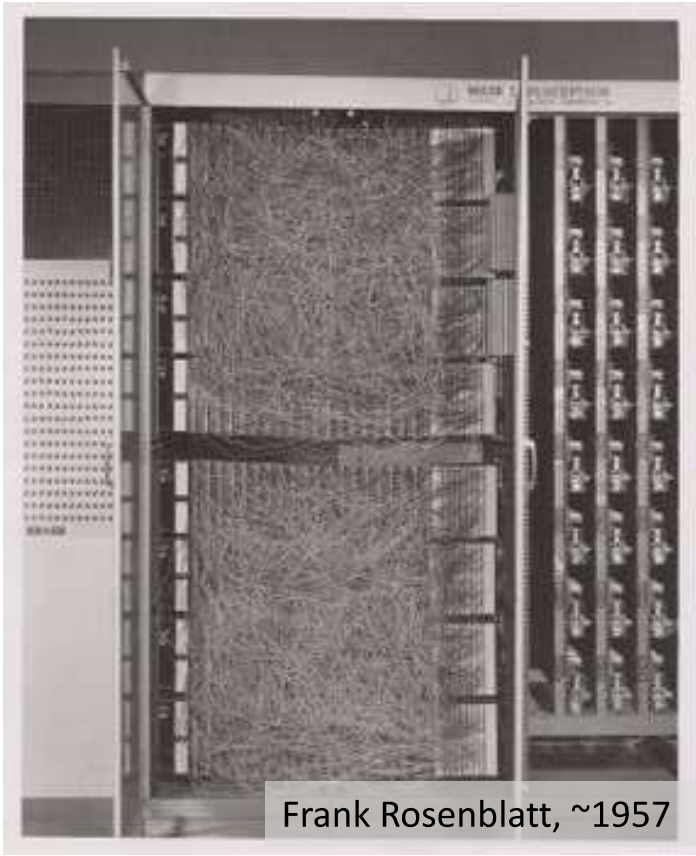
One of the earliest algorithms that could learn from data

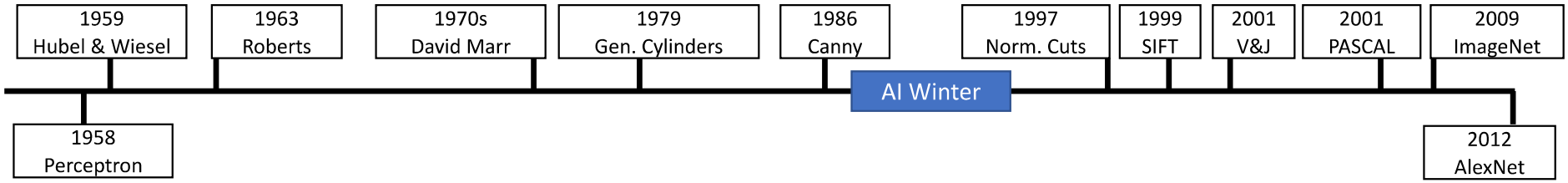
Implemented in hardware! Weights stored in potentiometers, updated with electric motors during learning

Connected to a camera that used 20x20 cadmium sulfide photocells to make a 400-pixel image

Could learn to recognize letters of the alphabet

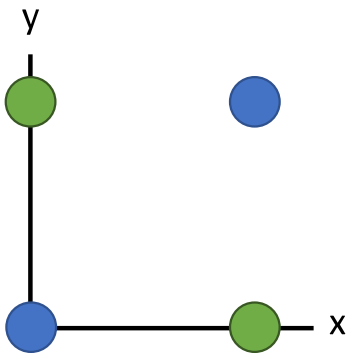
Today we would recognize it as a **linear classifier**

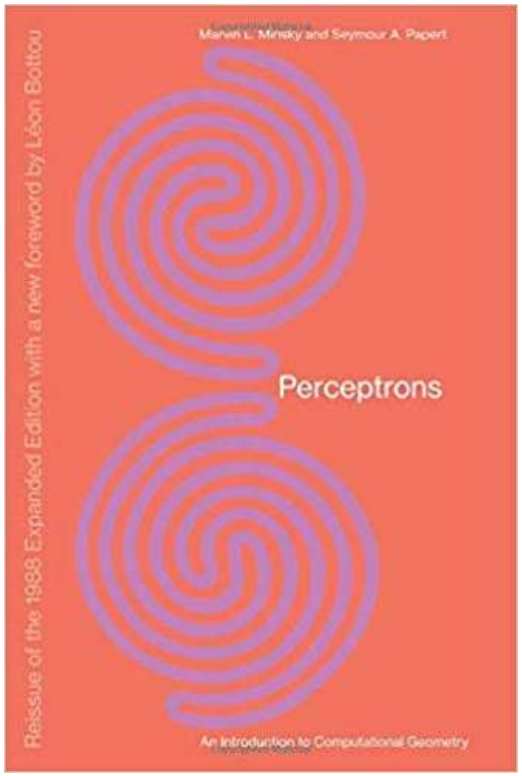

Frank Rosenblatt, ~1957



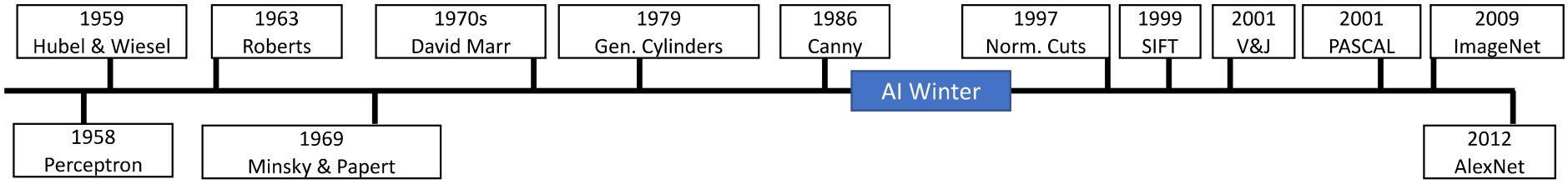
Minsky and Papert, 1969

X	Y	F(x,y)
0	0	0
0	1	1
1	0	1
1	1	0





Showed that Perceptrons could not learn the XOR function
Caused a lot of disillusionment in the field

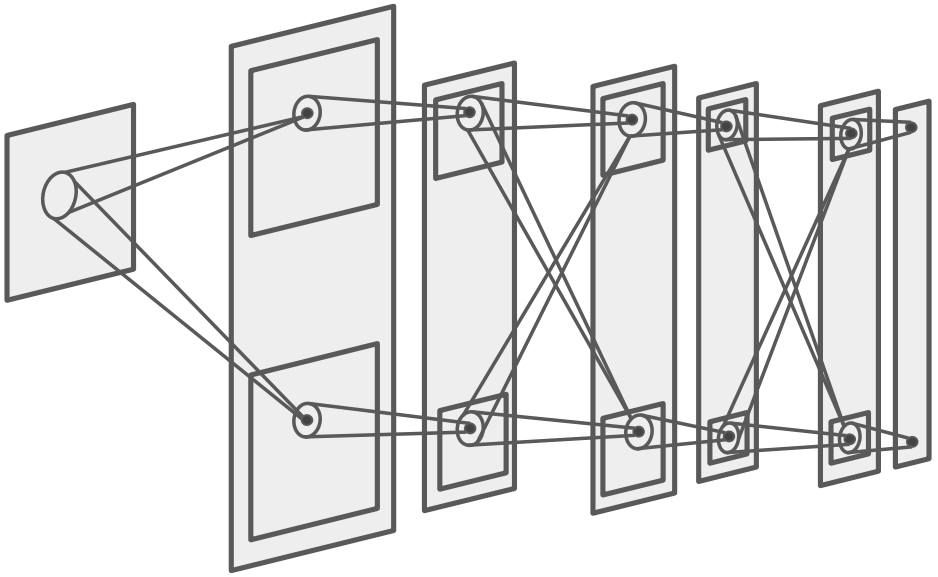


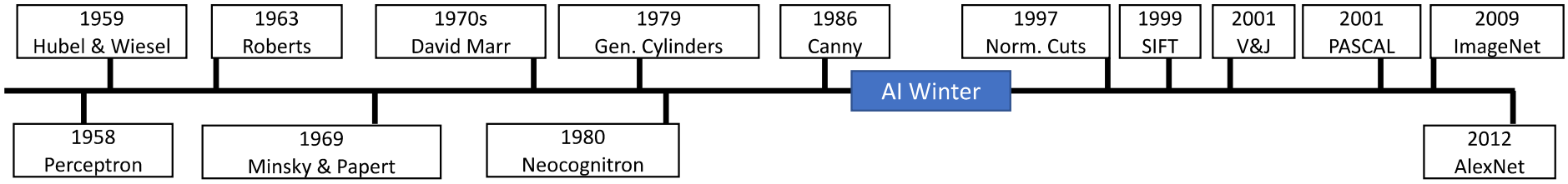
Neocognitron: Fukushima, 1980

Computational model the visual system, directly inspired by Hubel and Wiesel's hierarchy of complex and simple cells

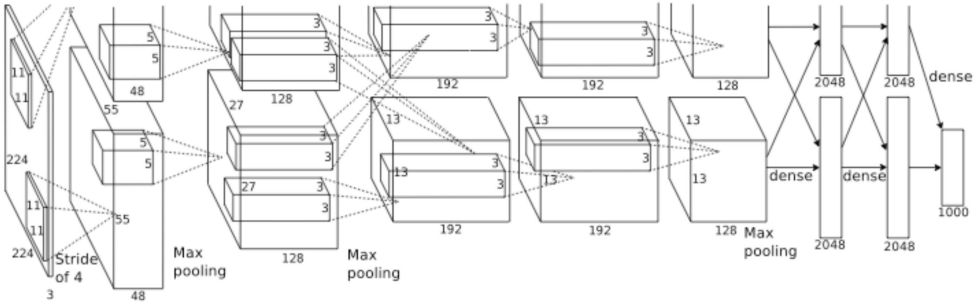
Interleaved simple cells (convolution) and complex cells (pooling)

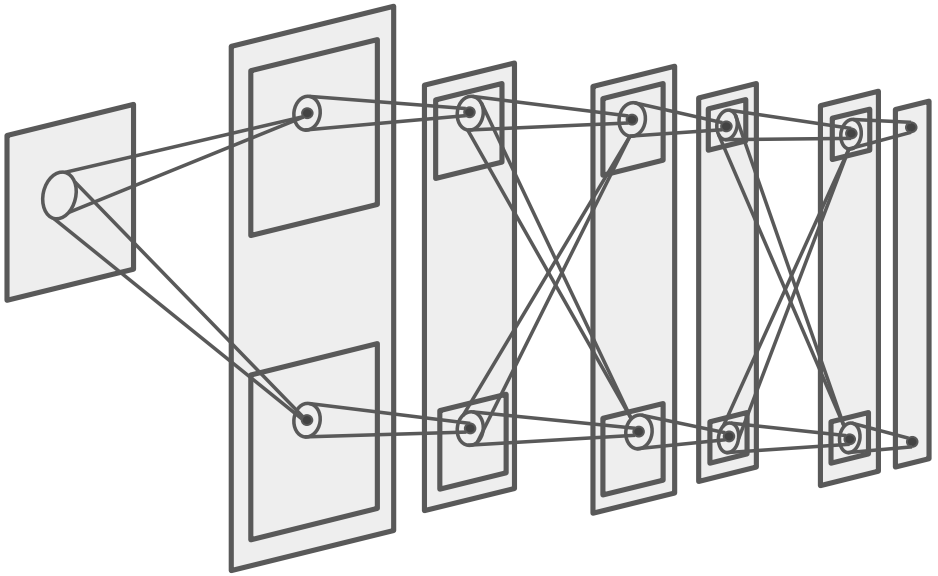
No practical training algorithm



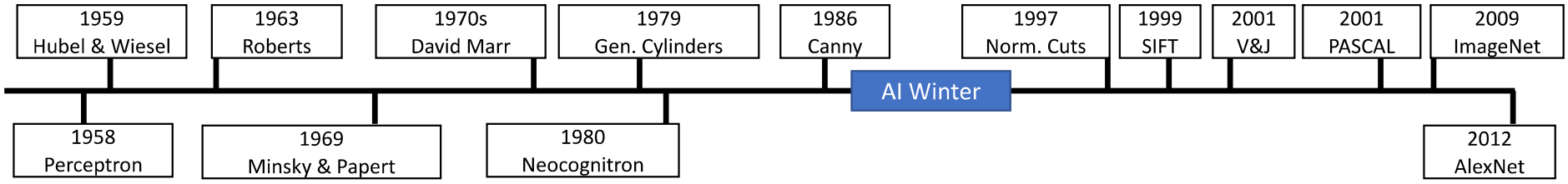


Neocognitron: Fukushima, 1980





Looks a lot like AlexNet
more than 32 years later!



Backprop: Rumelhart, Hinton, and Williams, 1986

Introduced backpropagation for computing gradients in neural networks

Successfully trained perceptrons with multiple layers

$$\frac{\partial E_p}{\partial w_{ji}} = \frac{\partial E_p}{\partial o_{pj}} \frac{\partial o_{pj}}{\partial w_{ji}}$$

recognizable math

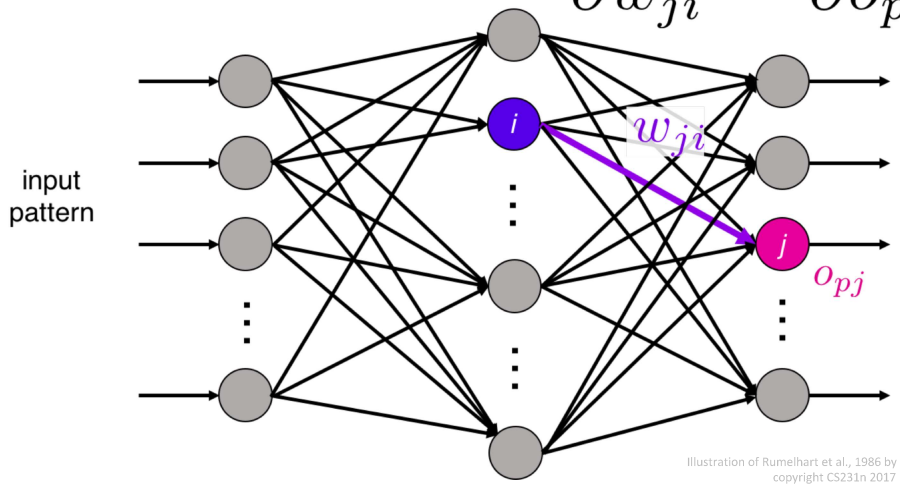
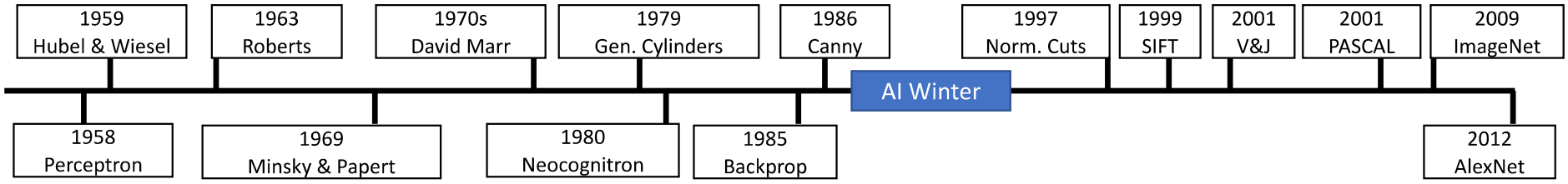
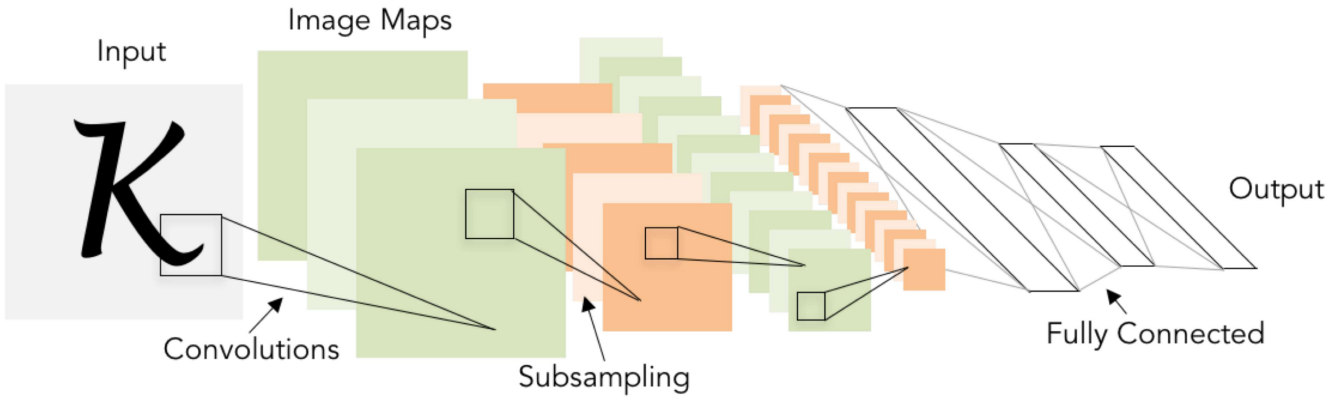


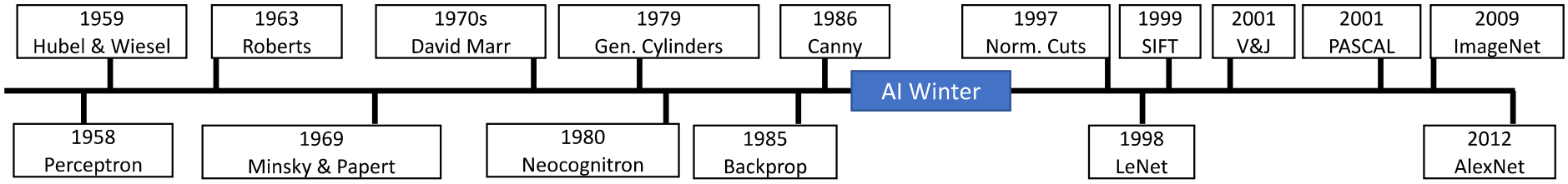
Illustration of Rumelhart et al., 1986 by Lane McIntosh, copyright CS231n 2017



Convolutional Networks: LeCun et al, 1998



Applied backprop algorithm to a Neocognitron-like architecture
Learned to recognize handwritten digits
Was deployed in a commercial system by NEC, processed handwritten checks
Very similar to our modern convolutional networks!

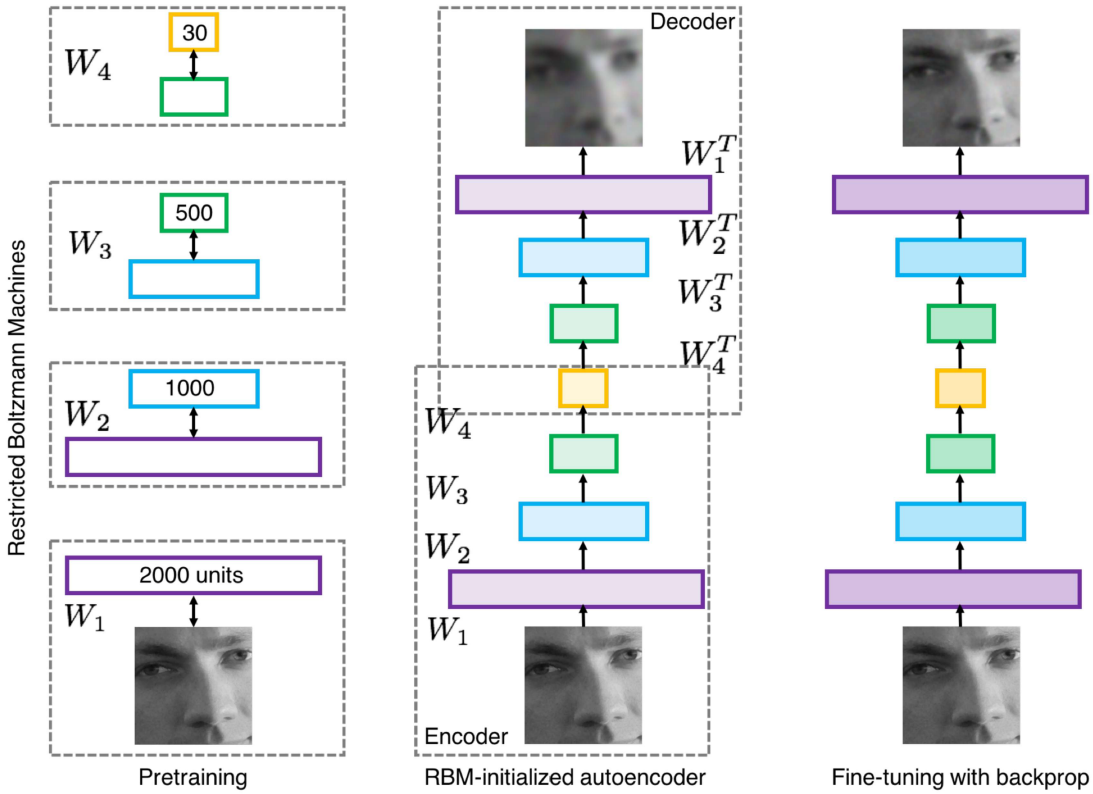


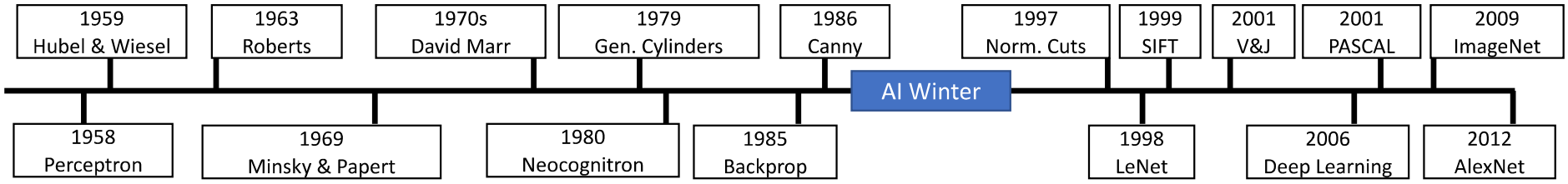
2000s: “Deep Learning”

People tried to train neural networks that were deeper and deeper

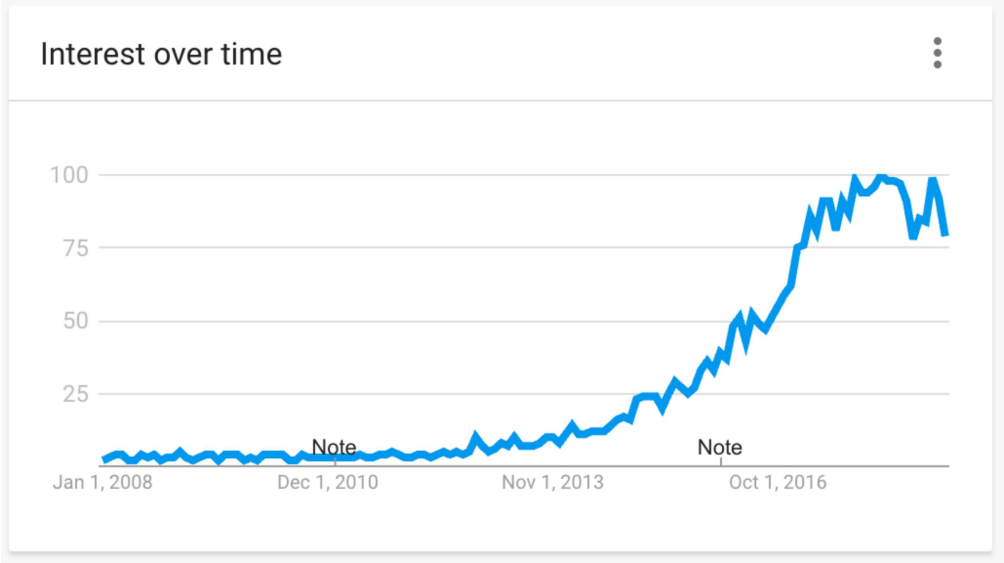
Not a mainstream research topic at this time

Hinton and Salakhutdinov, 2006
Bengio et al, 2007
Lee et al, 2009
Glorot and Bengio, 2010

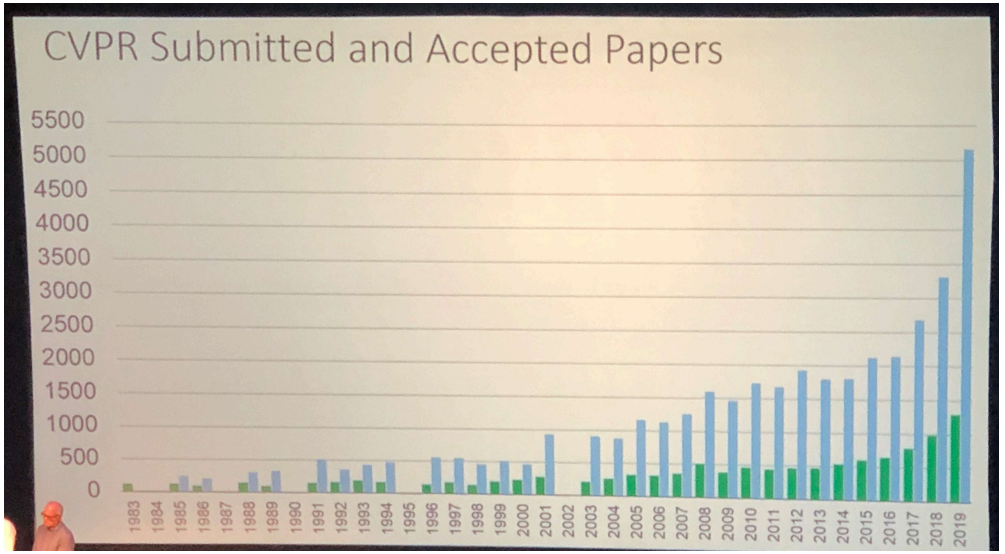




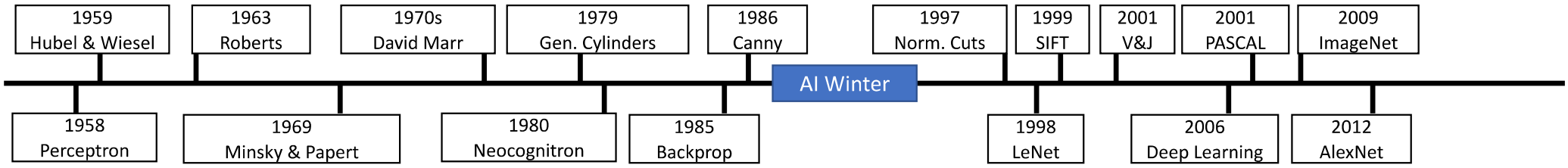
2012 to Present: Deep Learning Explosion



Google Trends: "Deep Learning"



Publications at top Computer Vision conference



2012 to Present: ConvNets are everywhere

Image Classification

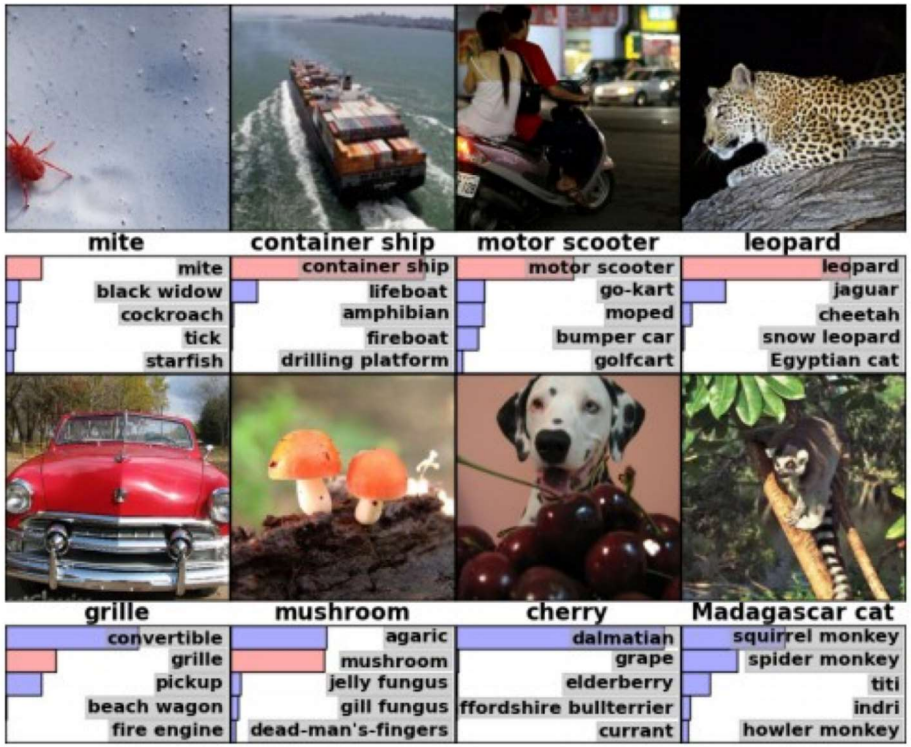
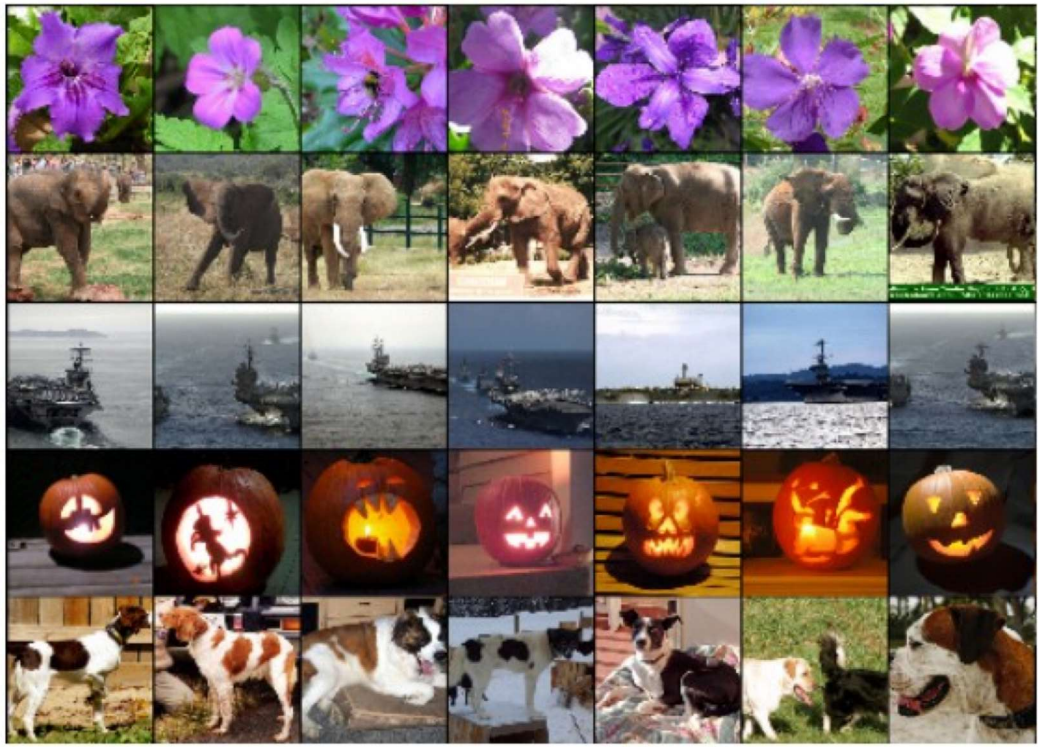
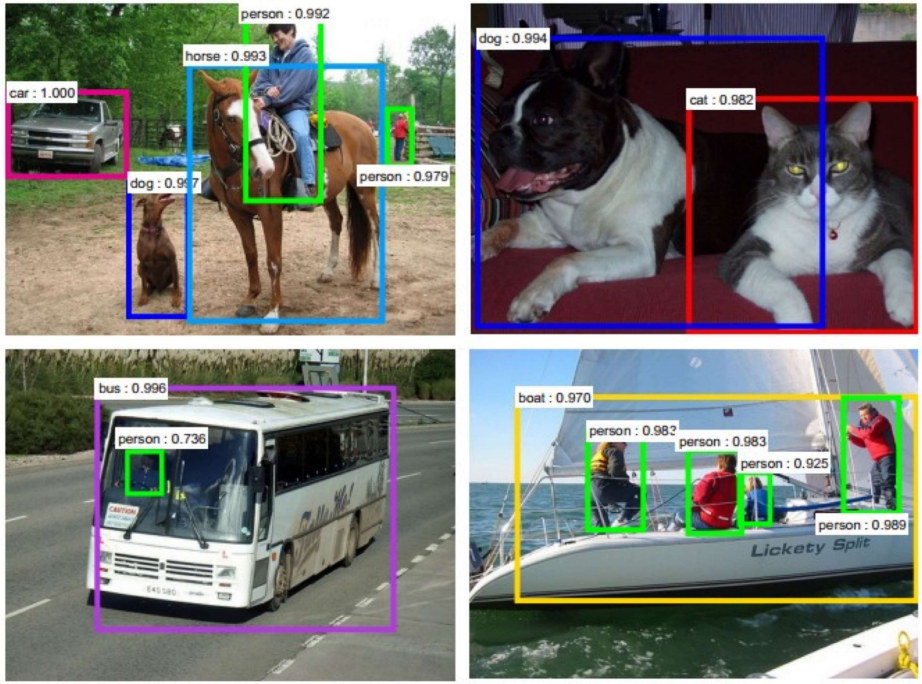


Image Retrieval



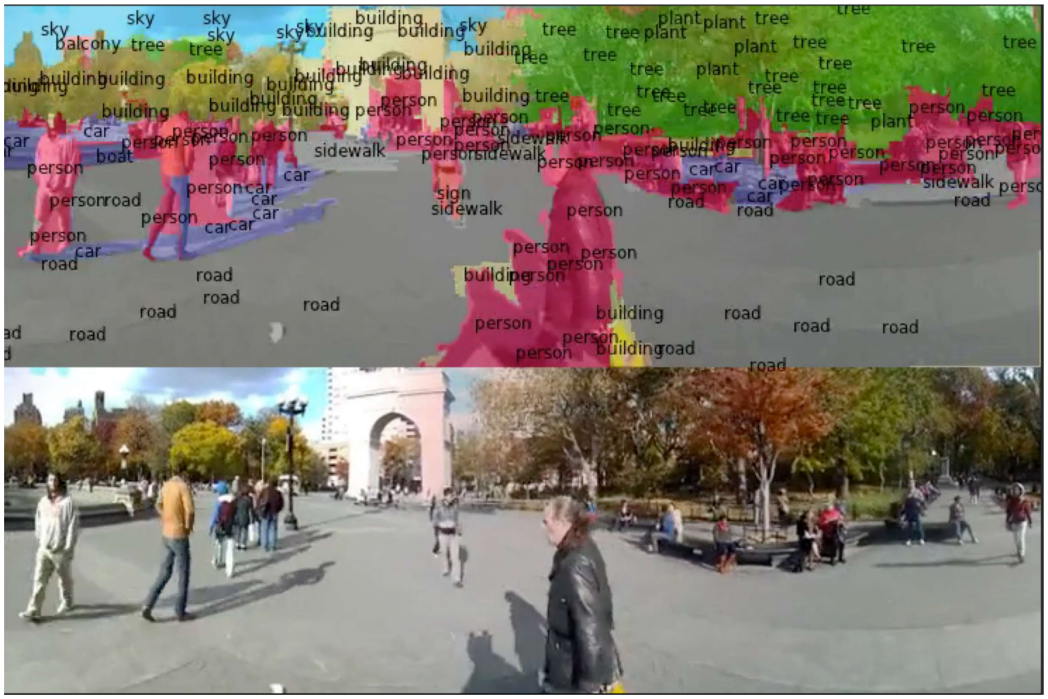
2012 to Present: ConvNets are everywhere

Object Detection



Ren, He, Girshick, and Sun, 2015

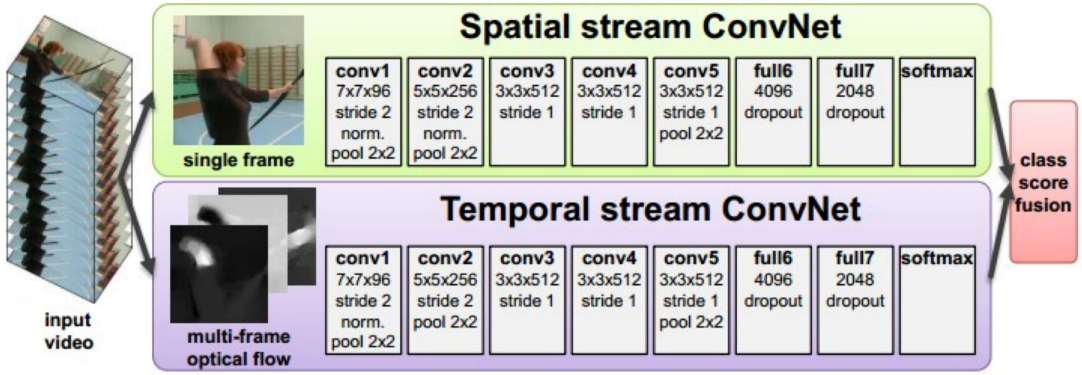
Image Segmentation



Fabaret et al, 2012

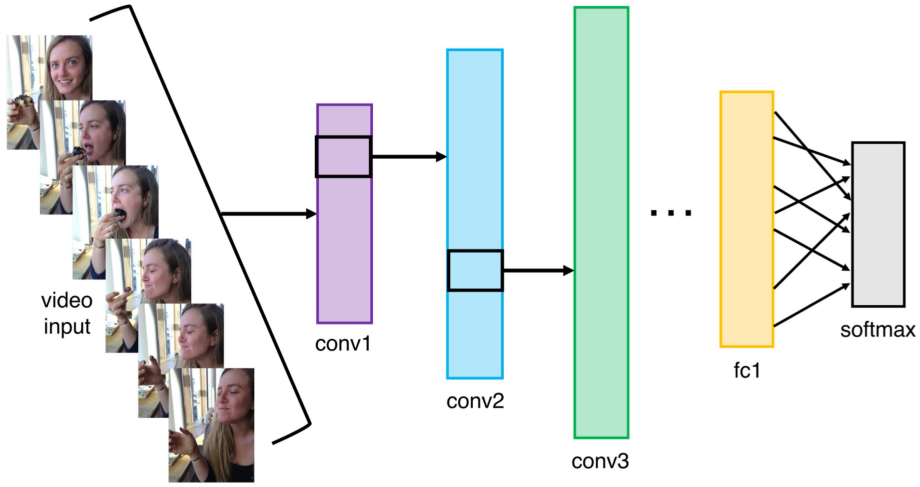
2012 to Present: ConvNets are everywhere

Video Classification



Simonyan et al, 2014

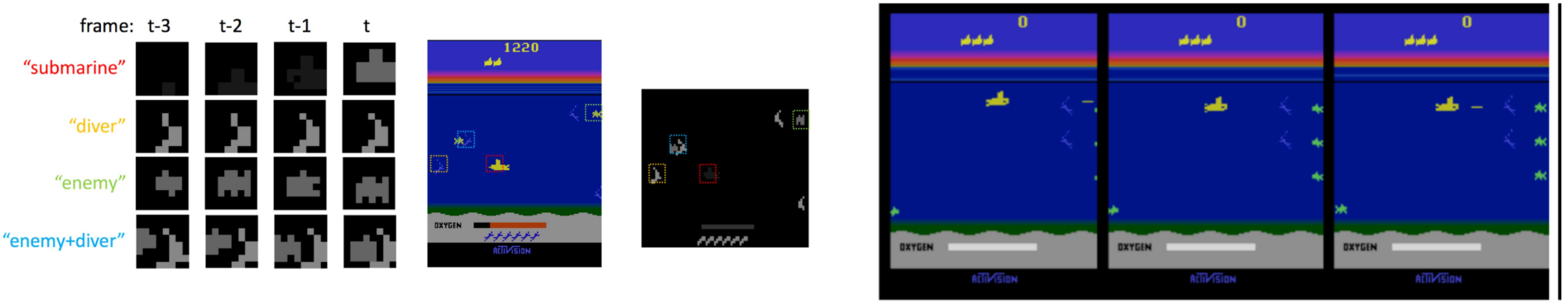
Activity Recognition



2012 to Present: ConvNets are everywhere

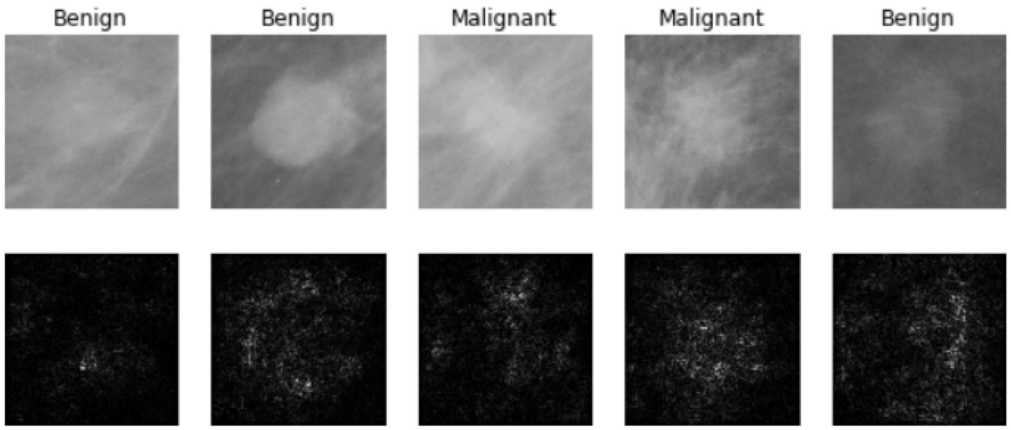
Pose Recognition (Toshev and Szegedy, 2014)

Playing Atari games (Guo et al, 2014)



2012 to Present: ConvNets are everywhere

Medical Imaging



Levy et al, 2016 Figure reproduced with permission

Galaxy Classification

Dieleman et al, 2014

From left to right: public domain by NASA, usage permitted by ESA/Hubble, public domain by NASA, and public domain.

Whale recognition

Kaggle Challenge

This image by Christin Khan is in the public domain and originally came from the U.S. NOAA.

2012 to Present: ConvNets are everywhere

Image Captioning

Vinyals et al, 2015
Karpathy and Fei-Fei, 2015

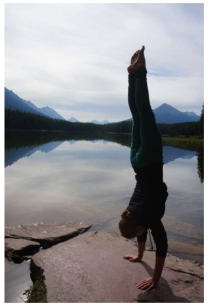
A white teddy bear sitting in the grass

A man in a baseball uniform throwing a ball

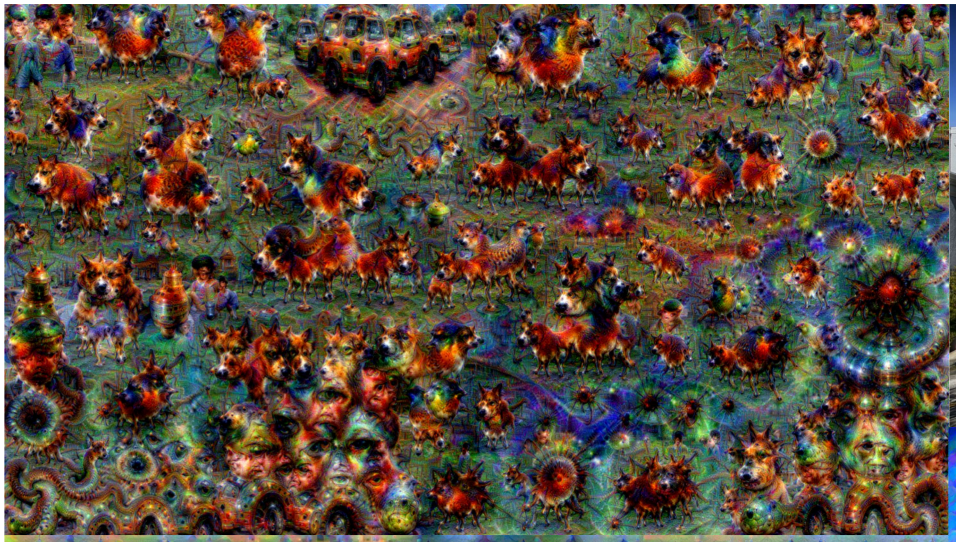
A woman is holding a cat in her hand

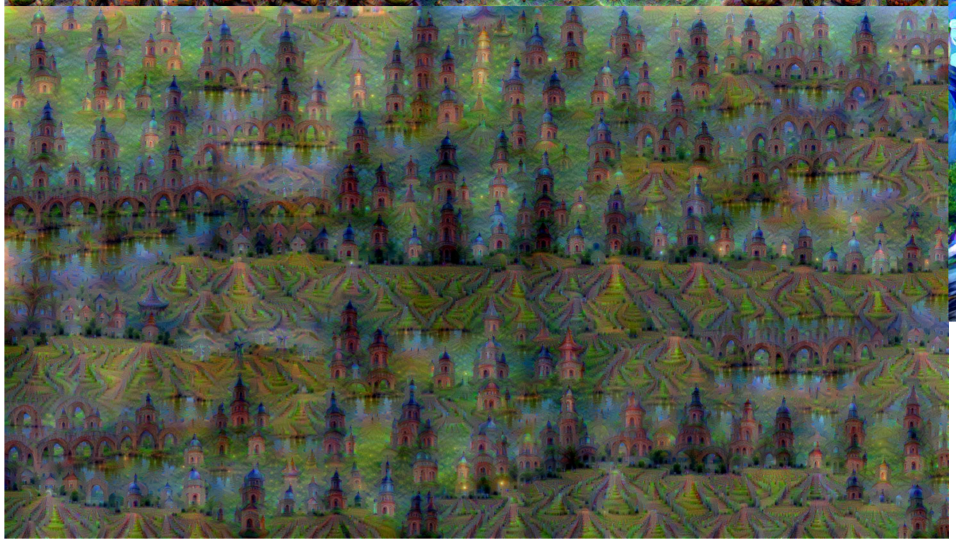
A man riding a wave on top of a surfboard

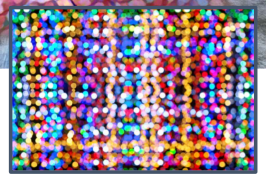
A cat sitting on a suitcase on the floor

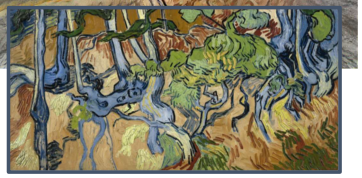


A woman standing on a beach holding a surfboard







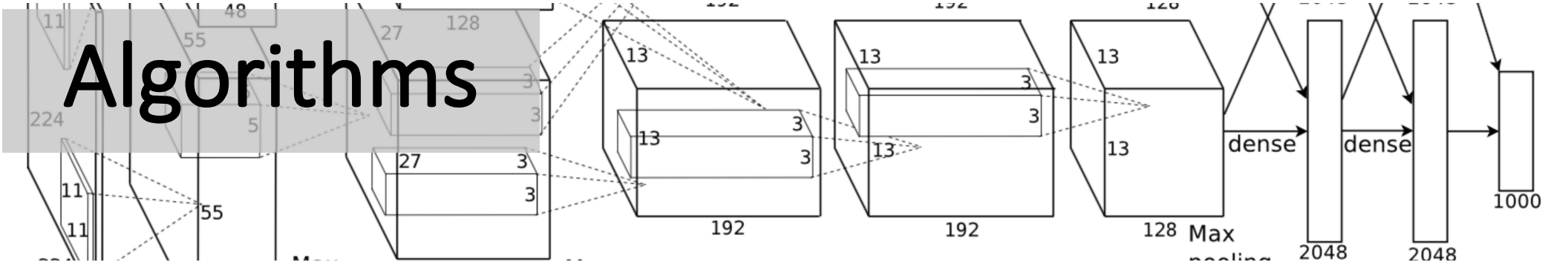


Figures copyright Justin Johnson, 2015. Reproduced with permission. Generated using the Inceptionism approach from a blog post by Google Research.

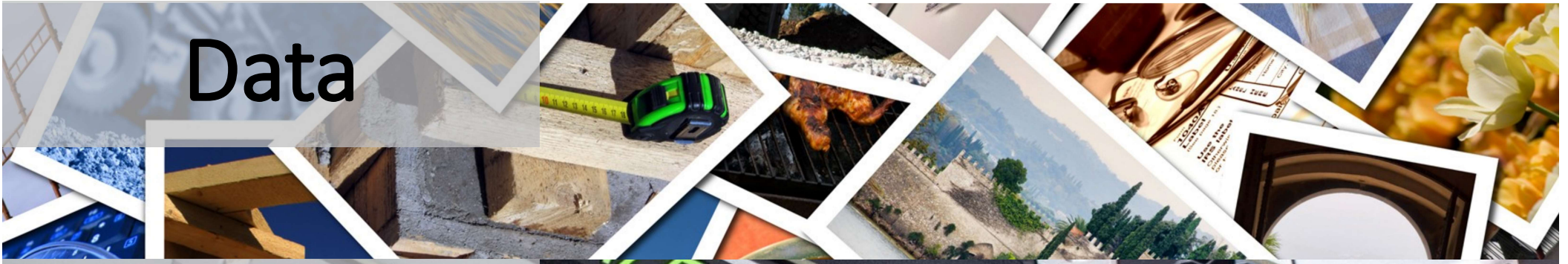
Original image is CC0 public domain
Starry Night and Tree Roots by Van Gogh are in the public domain
Bokeh image is in the public domain
Stylized images copyright Justin Johnson, 2017;
reproduced with permission

Mordvinsev et al, 2015
Gatys et al, 2016

Algorithms

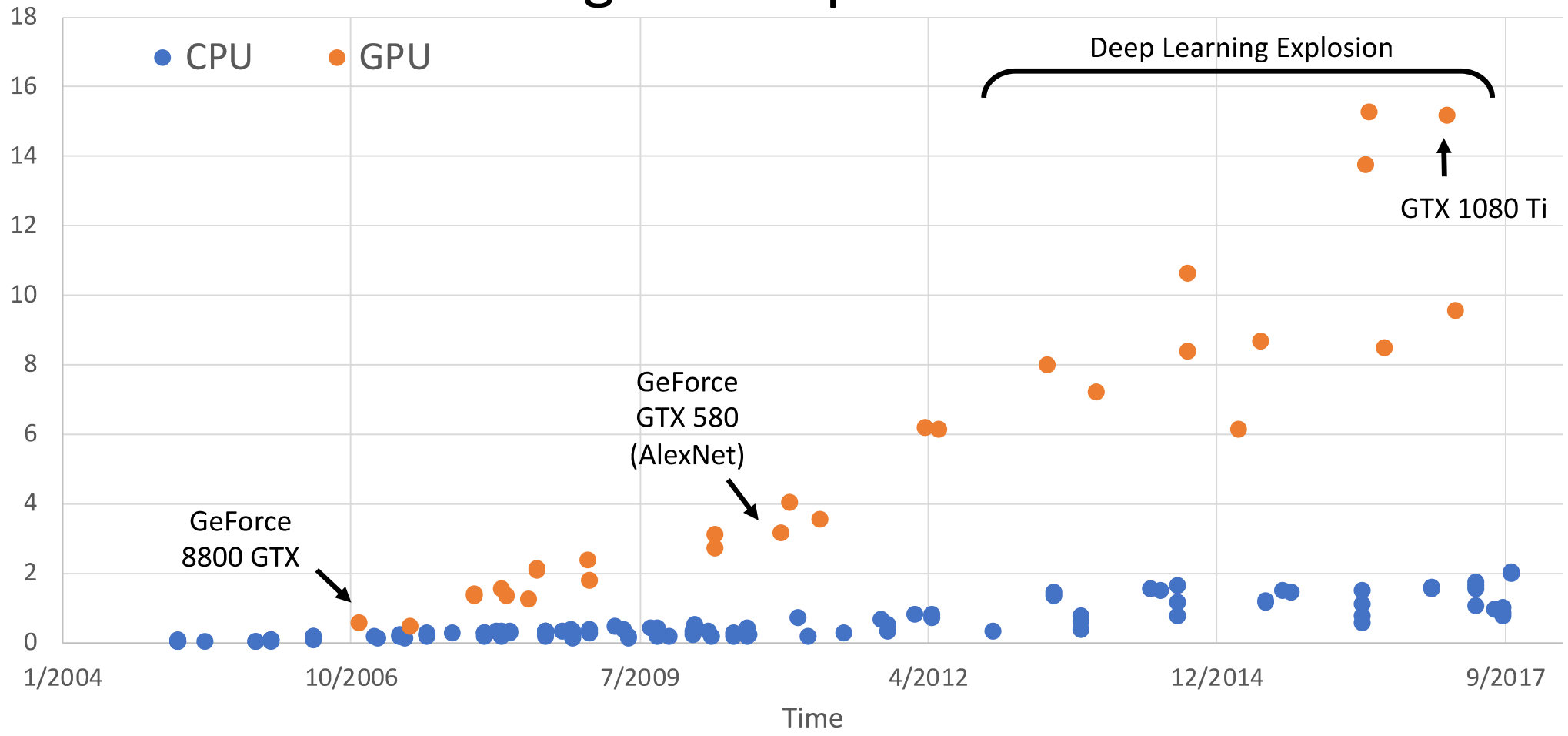


Data

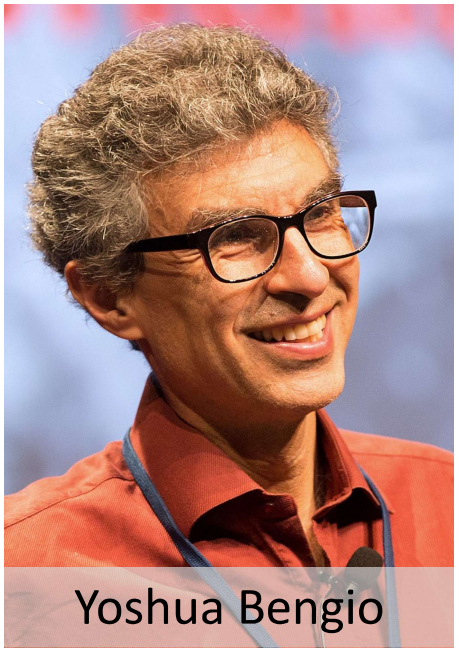


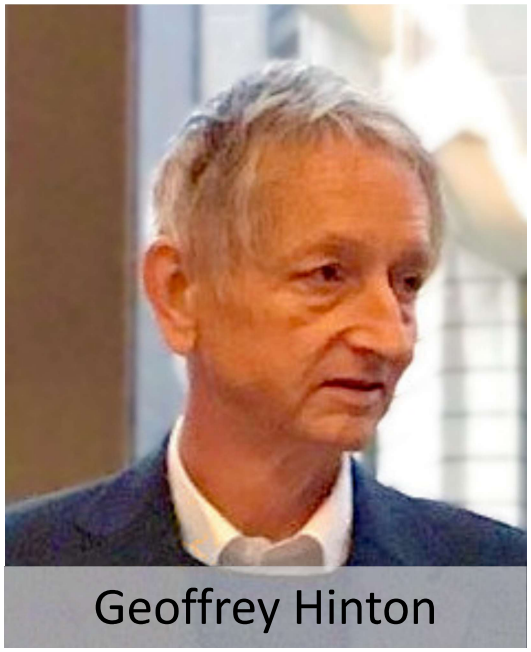
Computation

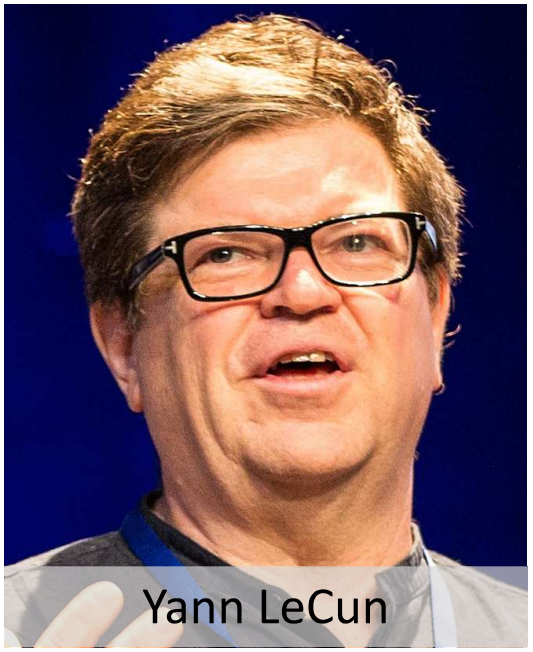
GigaFLOPs per Dollar

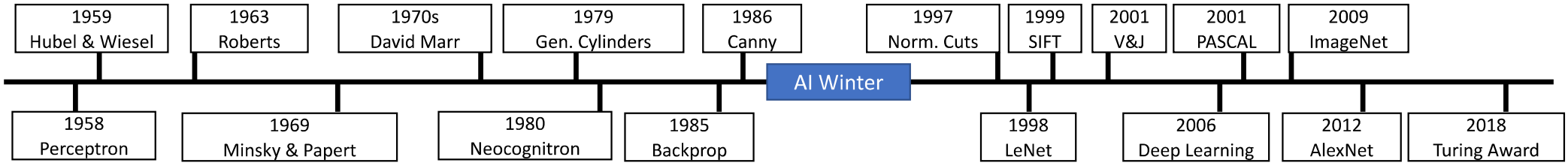


2018 Turing Award


Yoshua Bengio


Geoffrey Hinton


Yann LeCun



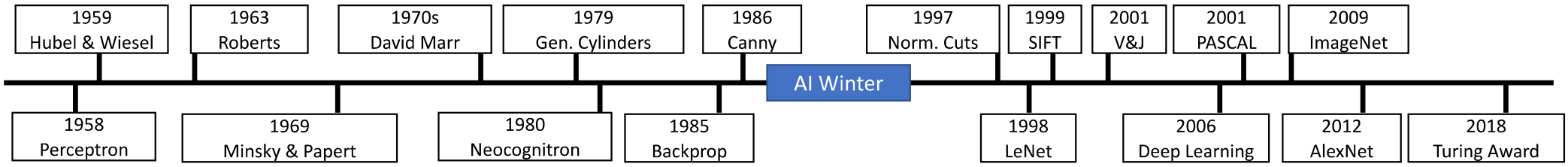
Despite our success, computer vision still has a long way to go…

Example credit:
Andrej Karpathy


Computer Vision Technology
Can Better Our Lives

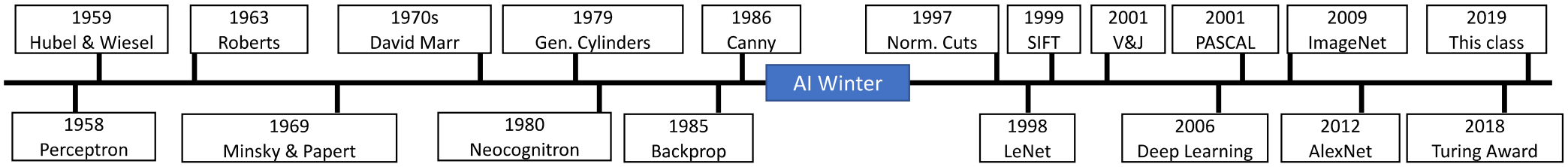
Today's Agenda

- A brief history of computer vision and deep learning
- Course overview and logistics



Today's Agenda

- A brief history of computer vision and deep learning
- Course overview and logistics



Course Staff

Instructor

Justin Johnson
Assistant Professor, CSE

Graduate Student Instructors

Yunseok Jang
PhD student, CSE

Video understanding,
Generative models

Kibok Lee
PhD student, CSE

Robustness,
Generalization

Luowei Zhao
PhD student, RI

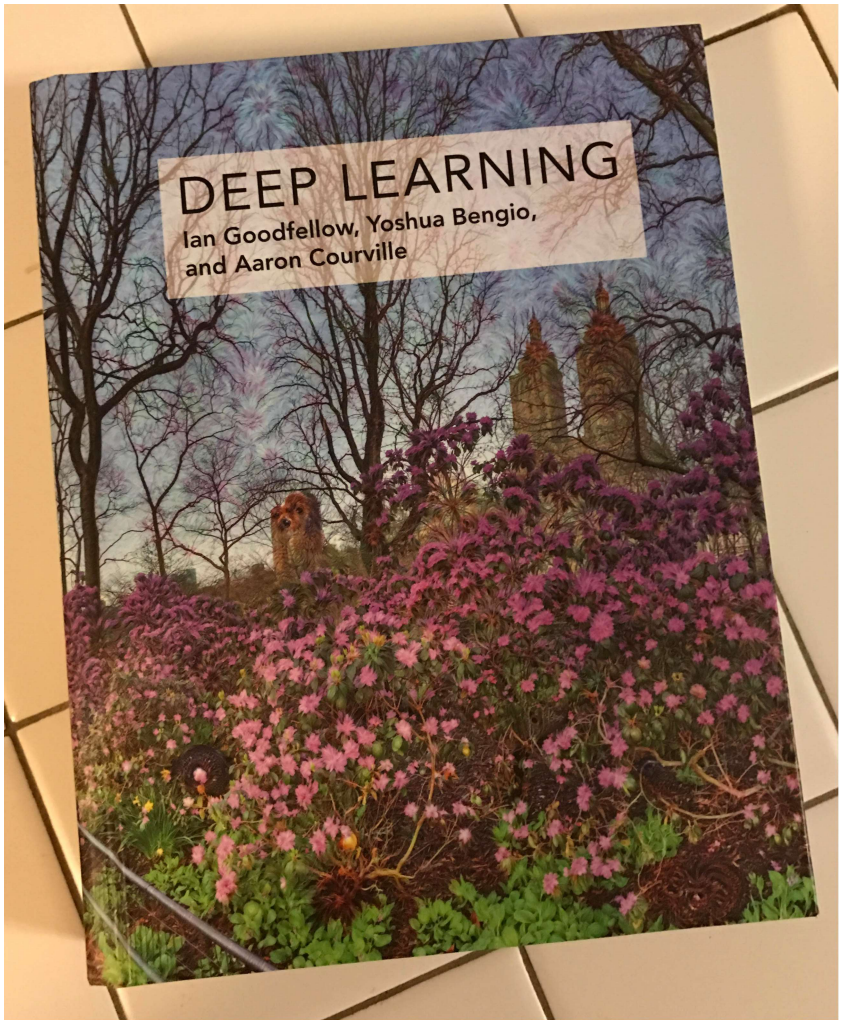
Vision & Language

How to contact us

- Course Website: https://web.eecs.umich.edu/~justincj/teaching/eecs498/
  - Syllabus, schedule, assignments, slides, lecture videos, etc
- Piazza: https://piazza.com/class/k01uvwqmf8c4nb
  - (Almost) all questions about the course should go here!
  - We will also use Piazza to communicate with you
  - Use private questions if you want to post code
- Canvas:
  - For turning in homework assignments
- Google Calendar: For office hours (starting next week)
- Email: Only for sensitive, confidential issues

Optional Textbook

- *Deep Learning* by Goodfellow, Bengio, and Courville
- Free online



Course Content and Grading

- 6 programming assignments (10% each)
  - Homework assignments will use Python, PyTorch, and Google Colab
- Midterm Exam (20%)
- Final Exam (20%)
- Late policy
  - 3 free late days to use on assignments
  - Once free late days are exhausted, 25% penalty per day

Course Content and Grading

- 6 programming assignments (10% each)
  - Homework assignments will use Python, PyTorch, and Google Colab
- Midterm Exam (20%)
- Final Exam (20%)
- Late policy
  - 3 free late days to use on assignments
  - Once free late days are exhausted, 25% penalty per day

Collaboration Policy

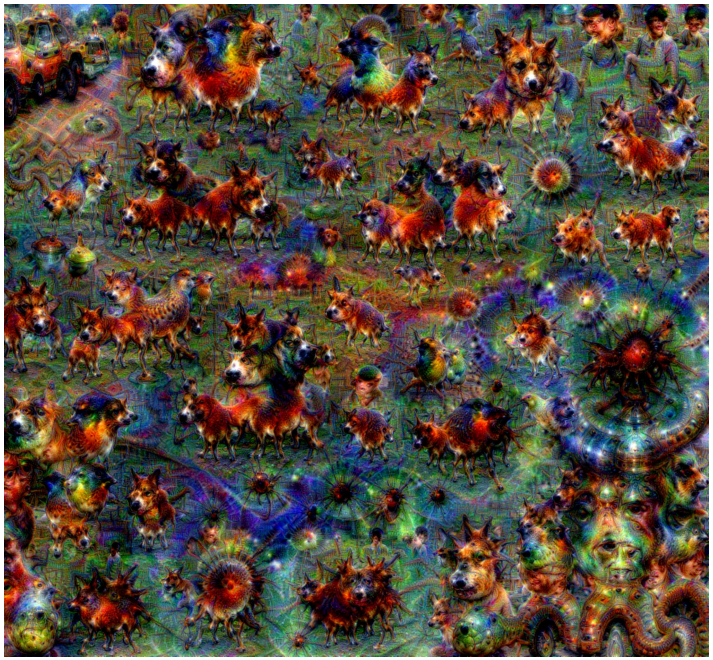
- **Rule 1**: Don't look at solutions or code that are not your own; everything you submit should be your own work
- **Rule 2**: Don't share your solution code with others; however discussing ideas or general strategies is fine and encouraged
- **Rule 3**: Indicate in your submissions anyone you worked with
- Turning in something late / incomplete is better than violating the honor code

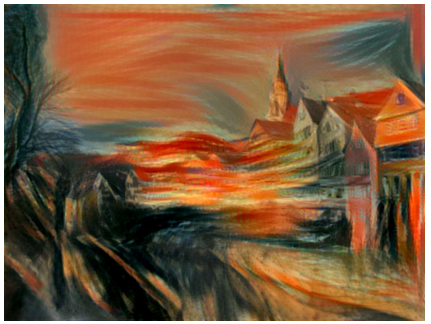
Course Philosophy

- Thorough and Detailed.
  - This not “Learn PyTorch in 90 days”, nor “Deep Learning in 10 lines of code”
  - Understand how to write from scratch, debug, and train convolutional and other types of deep neural networks
  - We prefer to write from scratch, rather than rely on existing implementations
- Practical
  - Focus on practical techniques for training and debugging neural networks
  - Will use state-of-the-art software tools like PyTorch and TensorFlow
- State of the art
  - Most material we cover is research published in the last 5 years

Course Philosophy

- Will also cover some fun topics:
  - Image captioning (with RNNs)
  - DeepDream, Artistic Style Transfer





Course Structure

- First half: Fundamentals
  - Details of how to implement and train different types of networks
  - Fully-connected networks, convolutional networks, recurrent networks
  - How to train and debug, very detailed
- Second half: Applications and “Researchy” topics
  - Object detection, image segmentation, 3D vision, videos
  - Attention, Transformers
  - Vision and Language
  - Generative models: GANs, VAEs, etc
  - Less detailed: provide overview and references, but skip some details

First homework assignment

- Will be released over the weekend
- Due one week after release
- Monday's lecture will be enough to complete it

Next time: Image Classification