# EECS 498-008 / 598-008
# Deep Learning for Computer Vision

## Lecture 1: Introduction

# Logistics:

- Remote for first two weeks (Lectures 1 – 3)

- After that, in-person lecture in Chrysler 220

# 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!




Left to right:
Image by Roger H Goun is licensed under CC BY 2.0
Image is CC0 1.0 public domain
Image is CC0 1.0 public domain
Image is CC0 1.0 public domain




Left to right:
Image is free to use
Image is CC0 1.0 public domain
Image by NASA is licensed under CC BY 2.0
Image is CC0 1.0 public domain



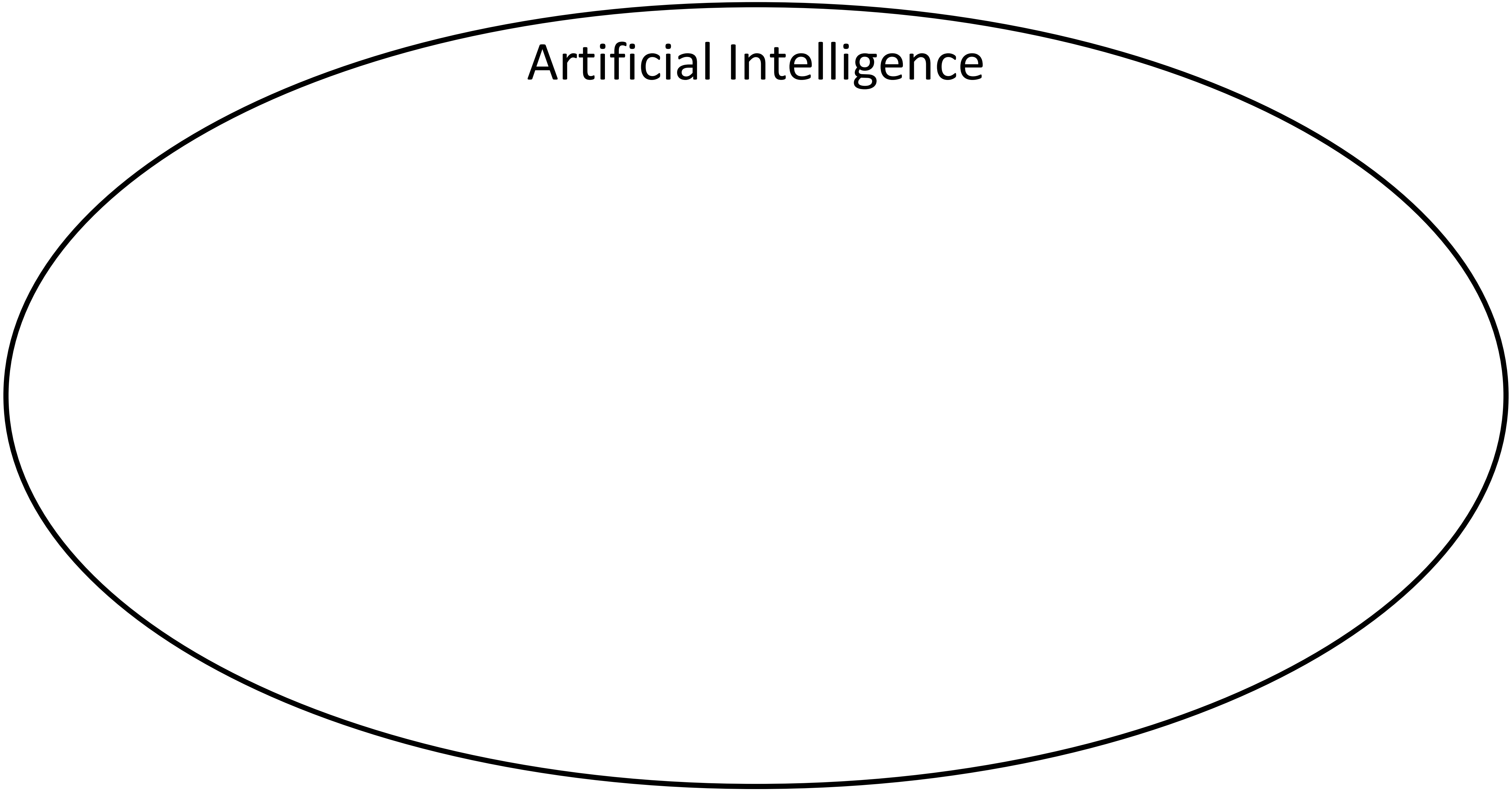

Bottom row, left to right
Image is CC0 1.0 public domain
Image by Derek Keats is licensed under CC BY 2.0; changes made
Image is public domain
Image is licensed under CC-BY 2.0; changes made

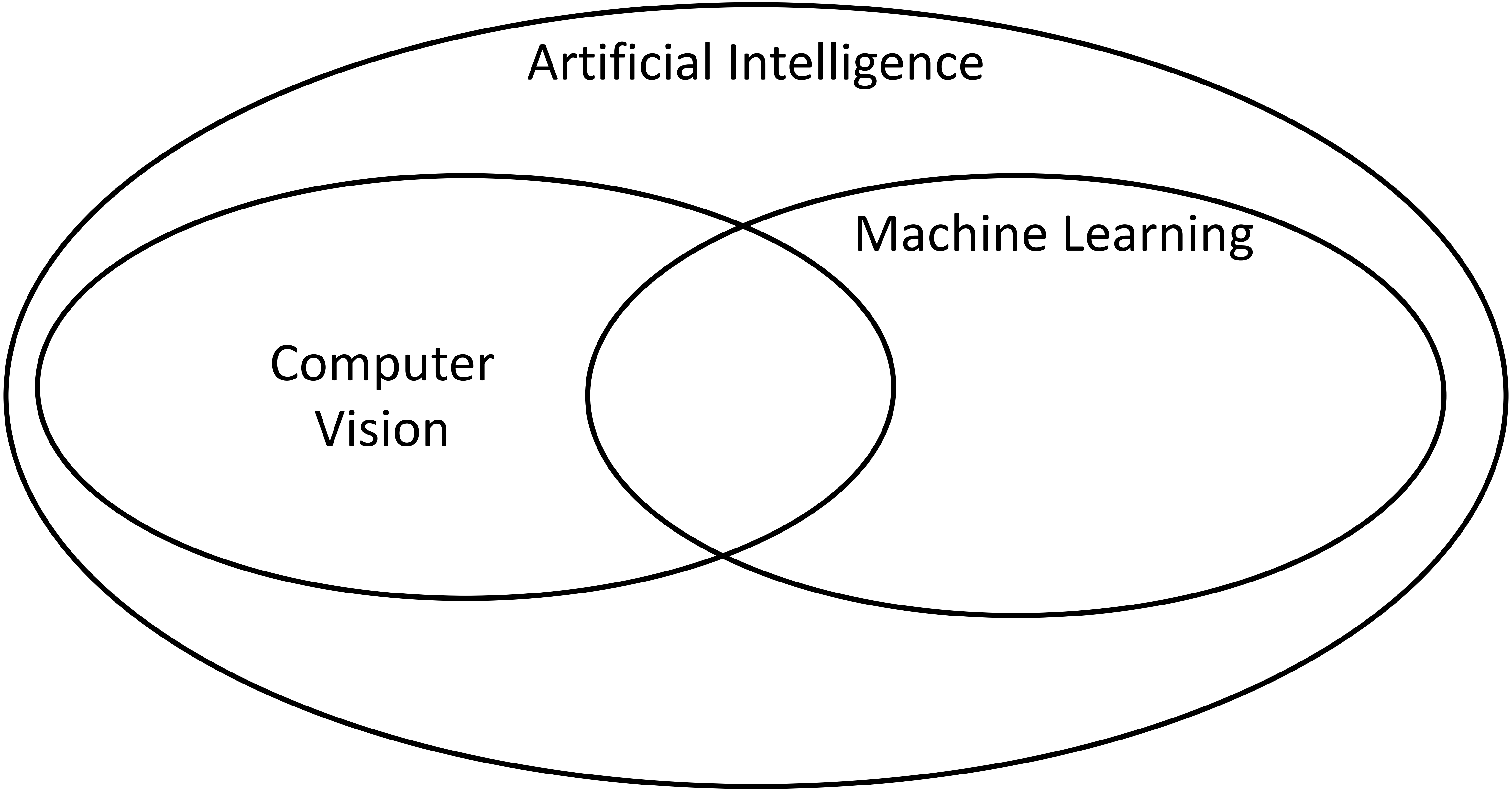
# Deep Learning for Computer Vision

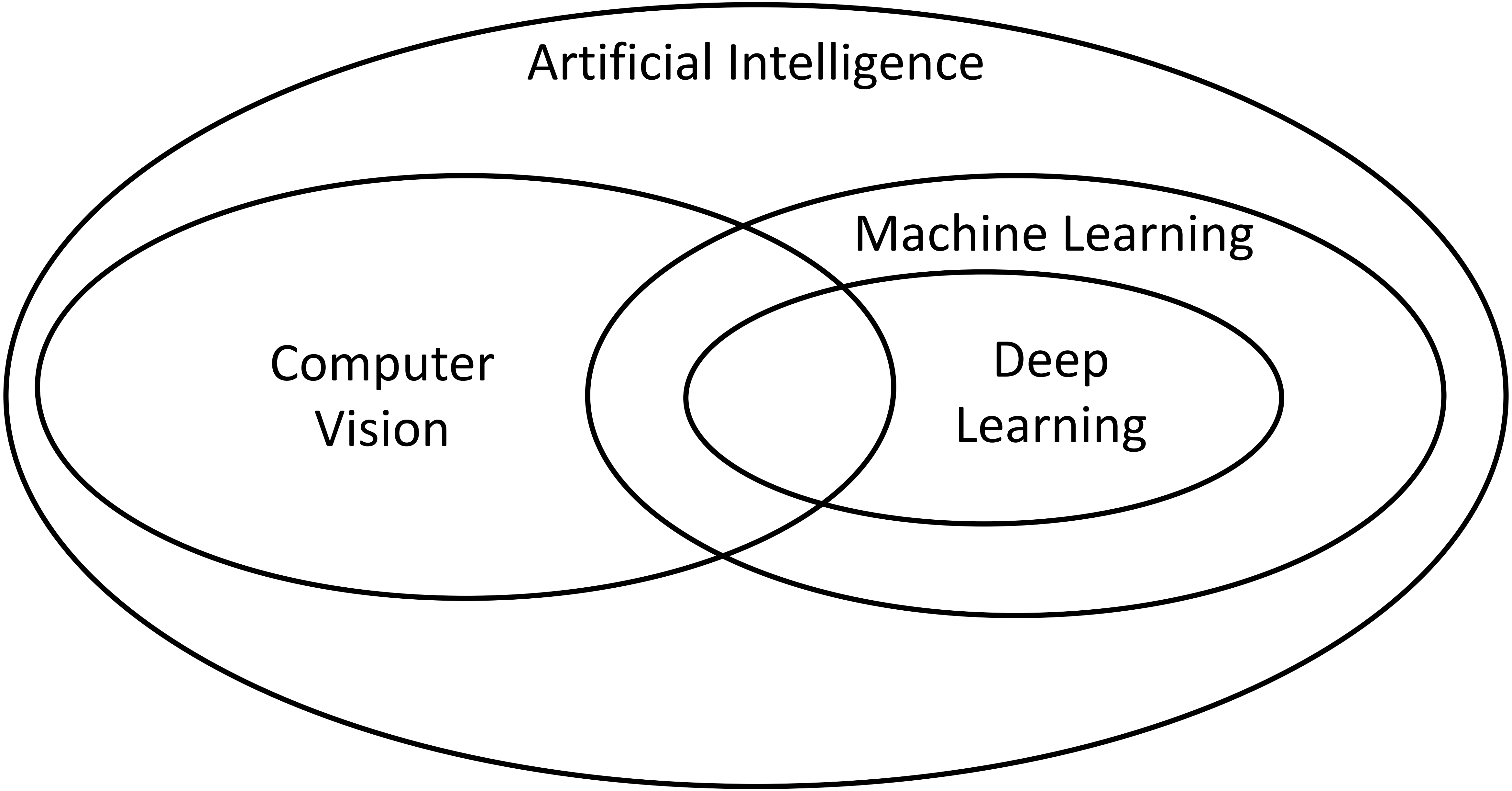
Building artificial systems that
learn from data and experience

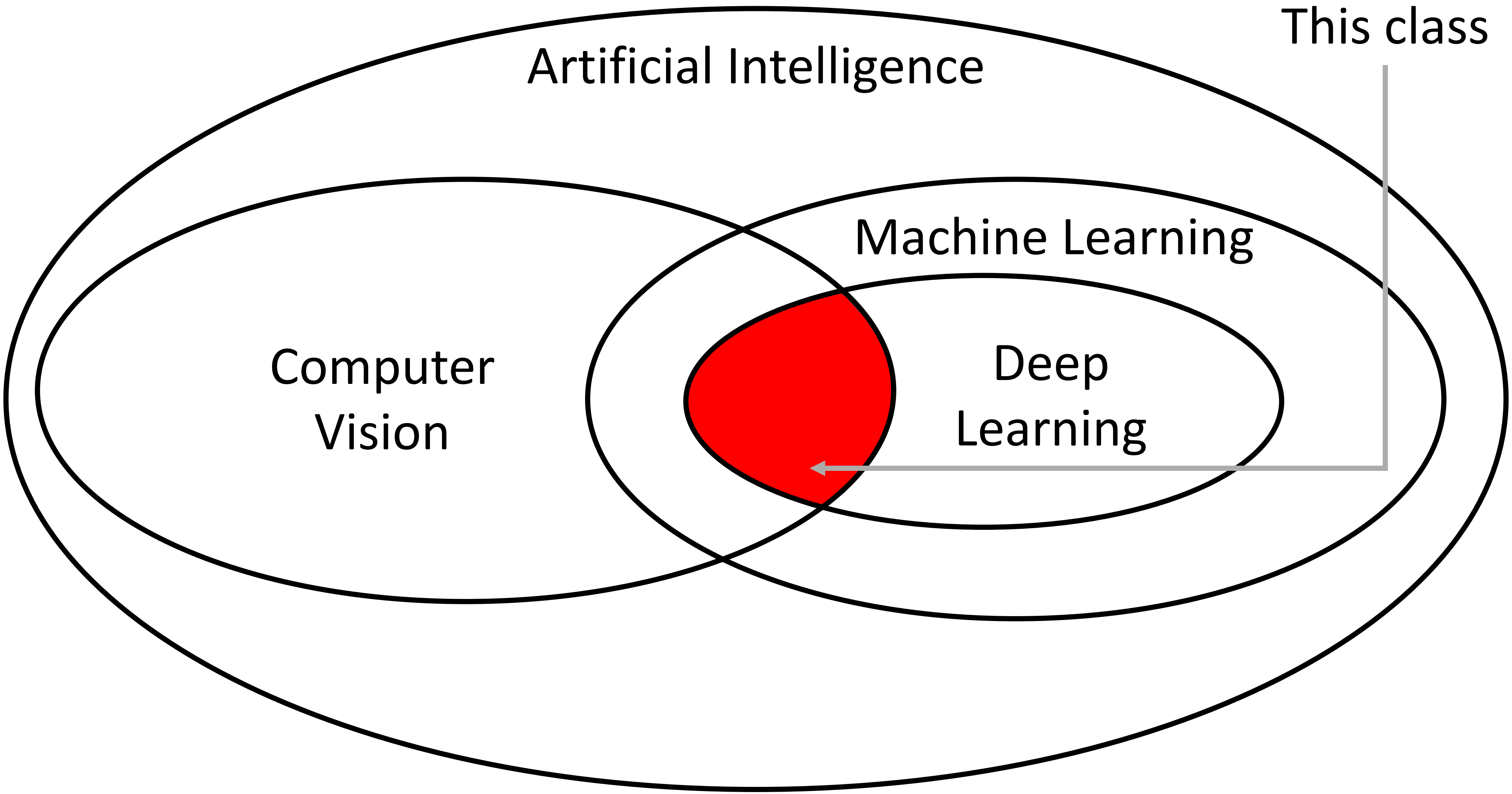
# Deep Learning for Computer Vision

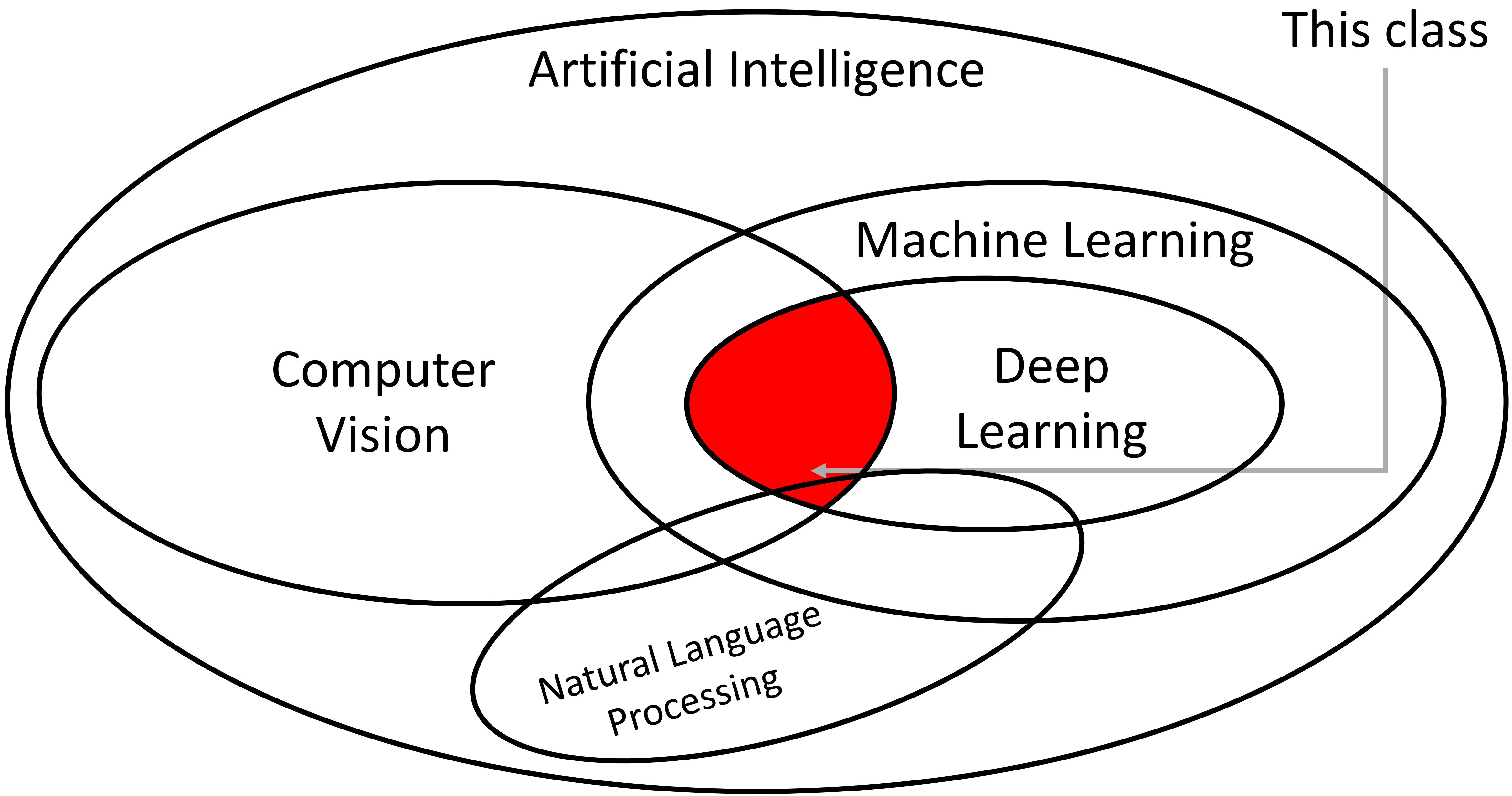
Hierarchical learning algorithms with many “layers”, (very) loosely inspired by the brain

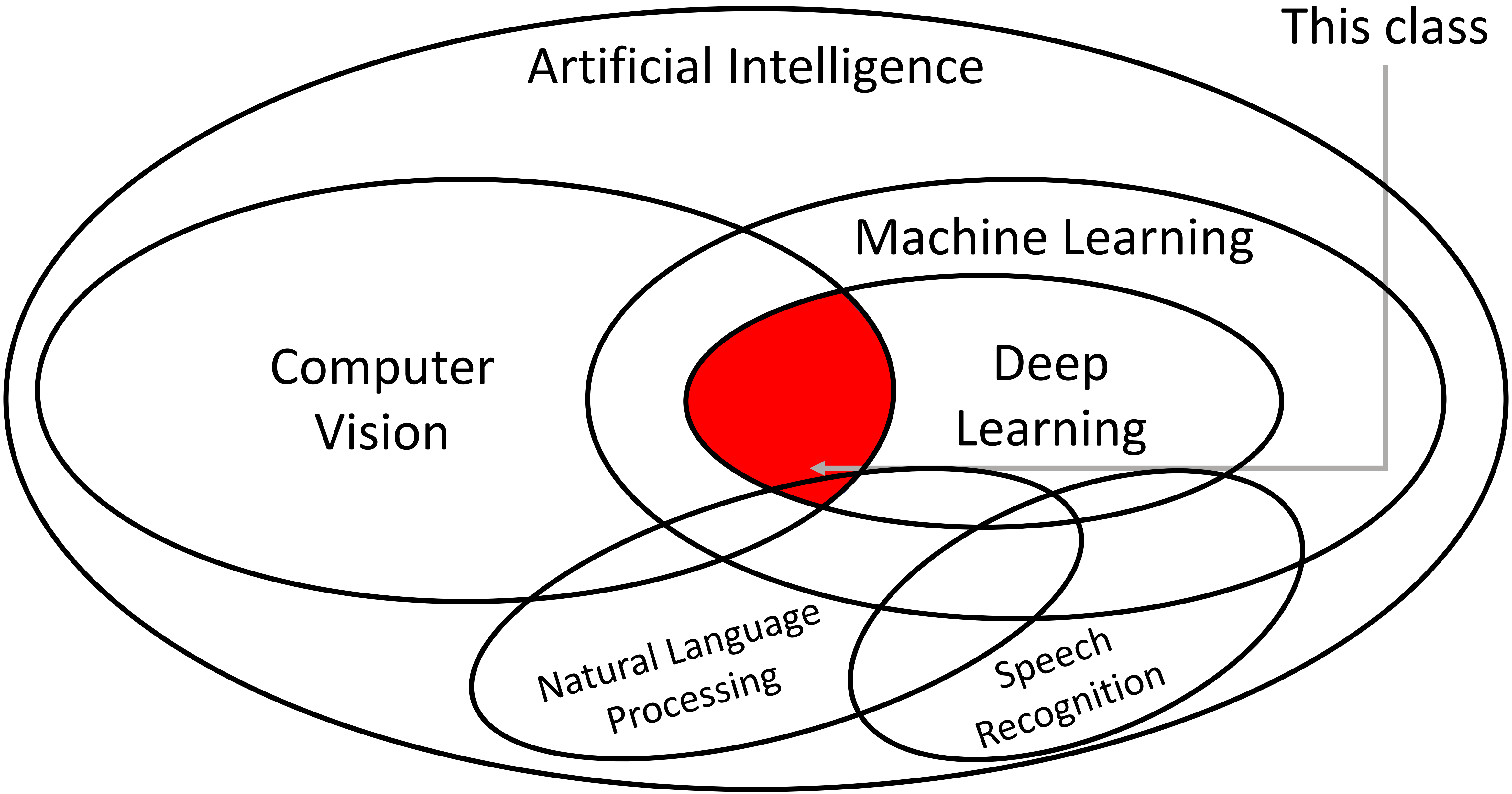

Artificial Intelligence

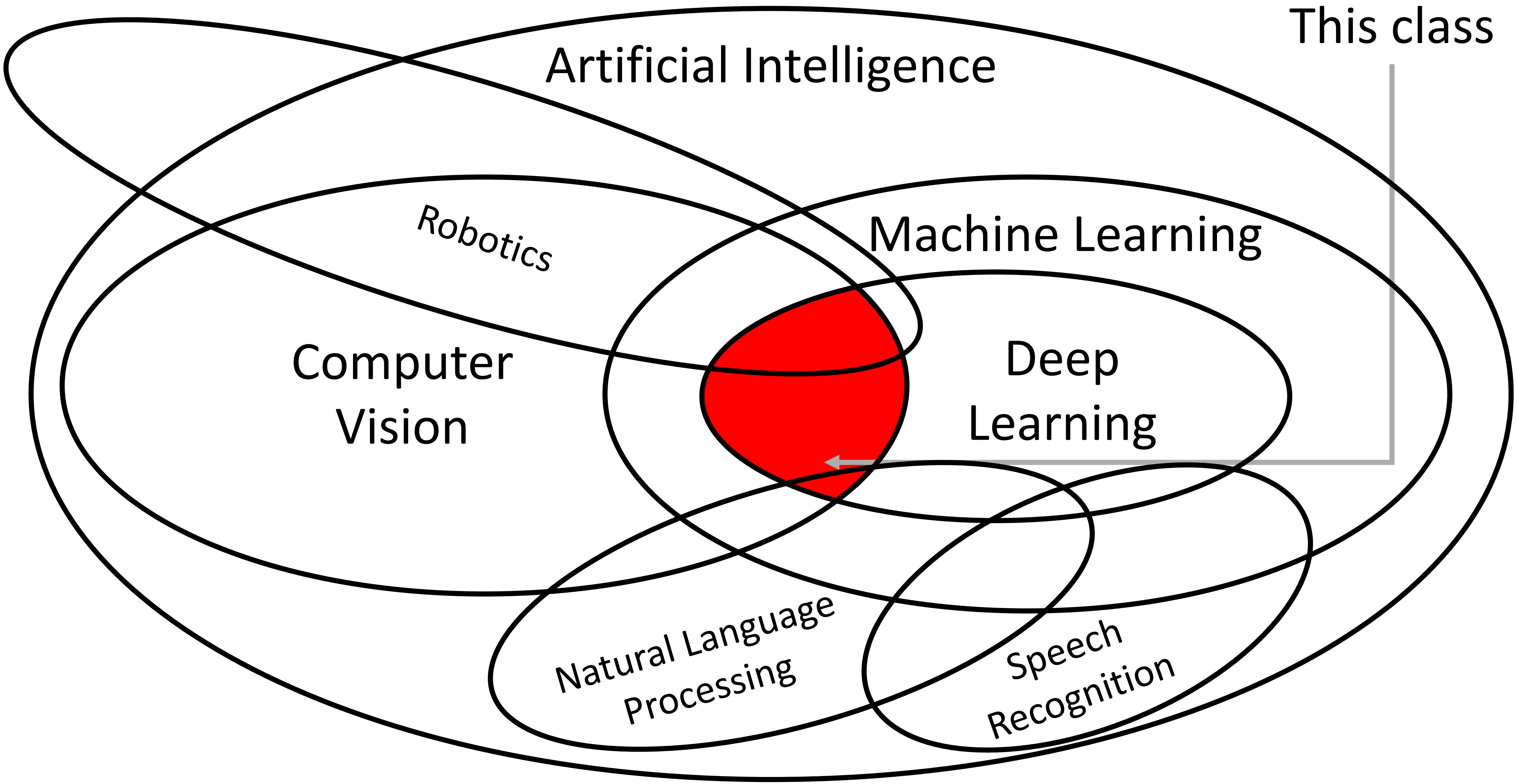

Artificial Intelligence
Machine Learning
Computer
Vision


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

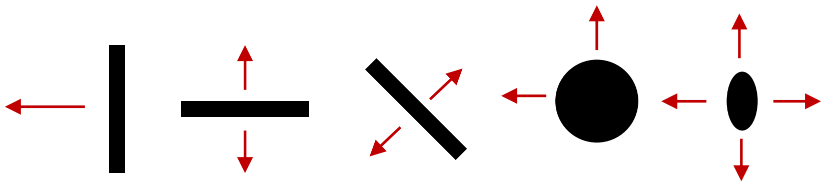

This class
Artificial Intelligence
Machine Learning
Computer Vision
Deep Learning
Natural Language Processing
Speech Recognition


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

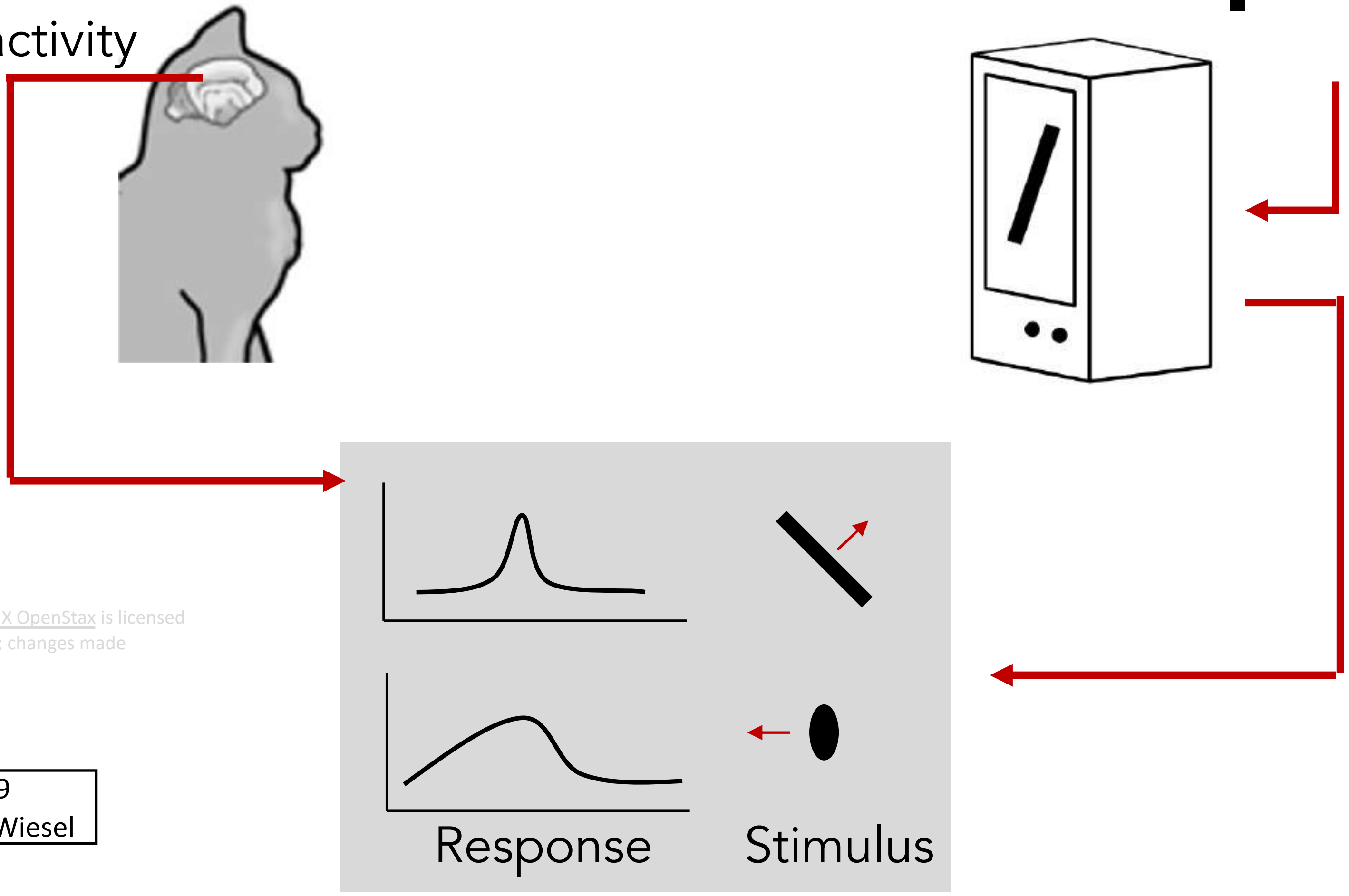
# Today's Agenda

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

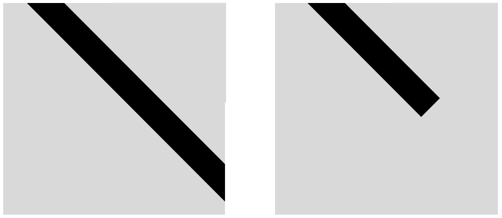
# Hubel and Wiesel, 1959



Measure
brain activity




**Simple cells**:
Response to light
orientation



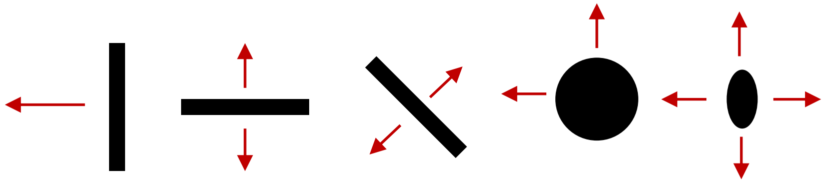
No response

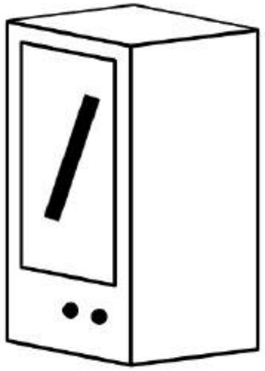


1959
Hubel & Wiesel

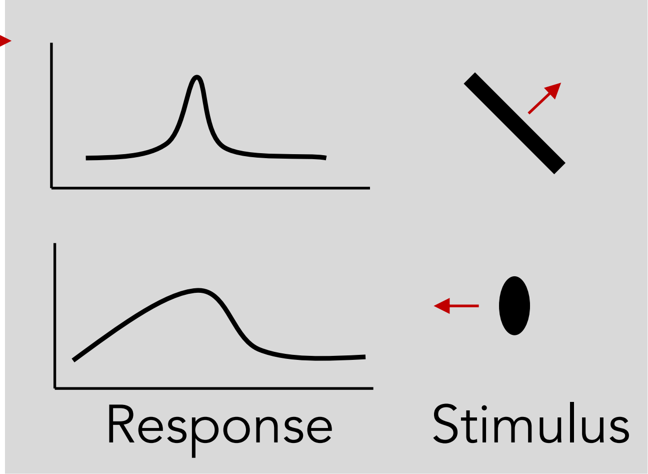
# Hubel and Wiesel, 1959

Measure brain activity





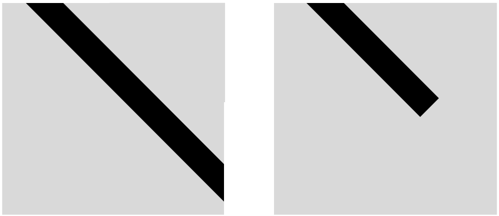
**Simple cells**:
Response to light orientation




**Complex cells**:
Response to light orientation and movement

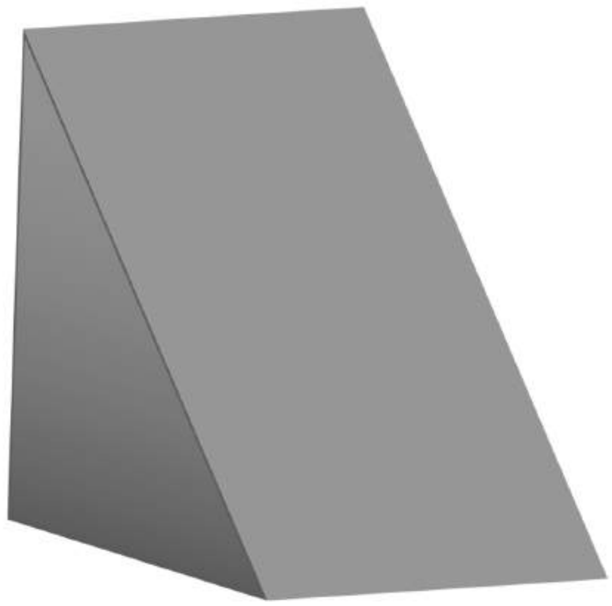


1959
Hubel & Wiesel

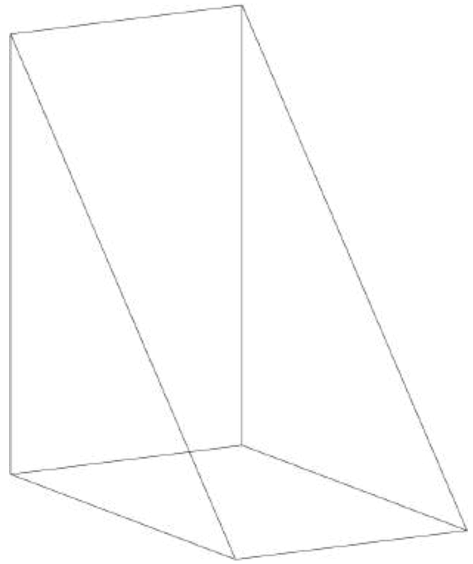


No response

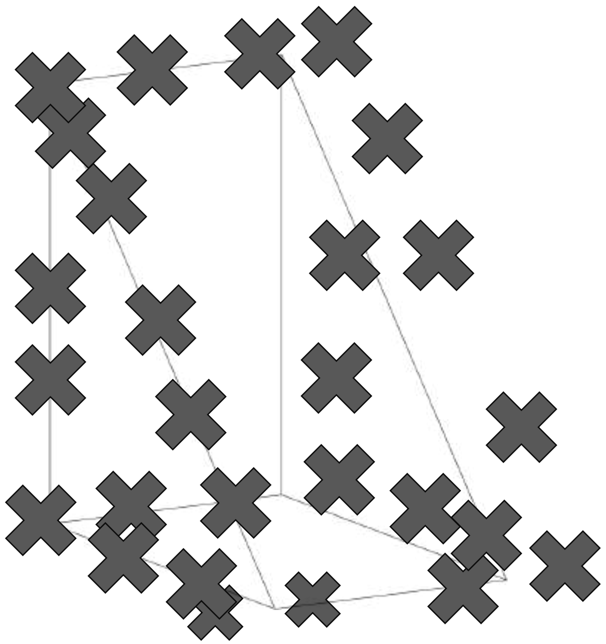
# Larry Roberts, 1963



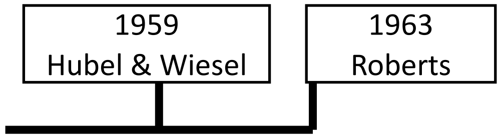
(a) Original picture



(b) Differentiated picture



(c) Feature points selected




Lawrence Gilman Roberts, “Machine Perception of Three-Dimensional Solids”, 1963

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

PROJECT MAC

Artificial Intelligence Group — July 7, 1966

Vision Memo. No. 100.

# THE SUMMER VISION PROJECT

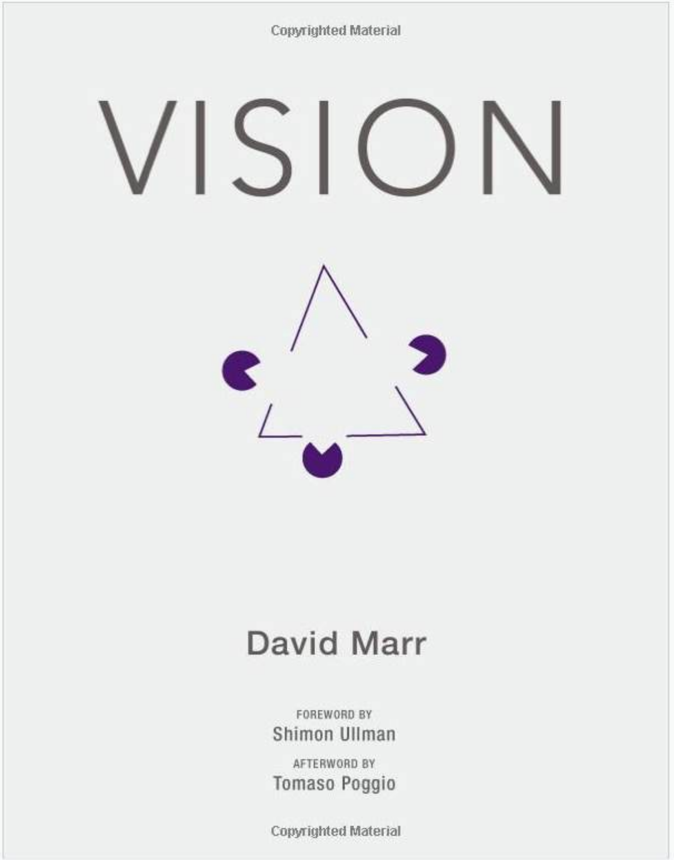
Seymour Papert

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

https://dspace.mit.edu/handle/1721.1/6125




1959
Hubel & Wiesel

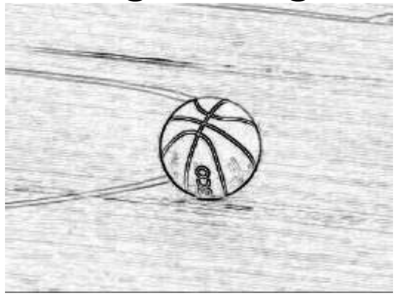
1963
Roberts

1970s
David Marr

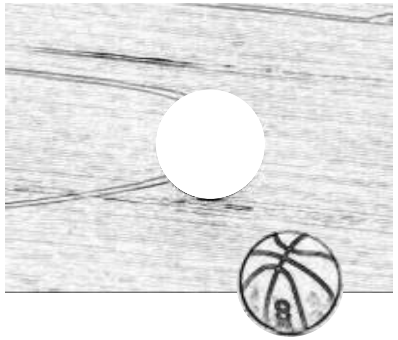
Input image

This image is CC0 1.0 public domain

Edge image



2 ½-D sketch



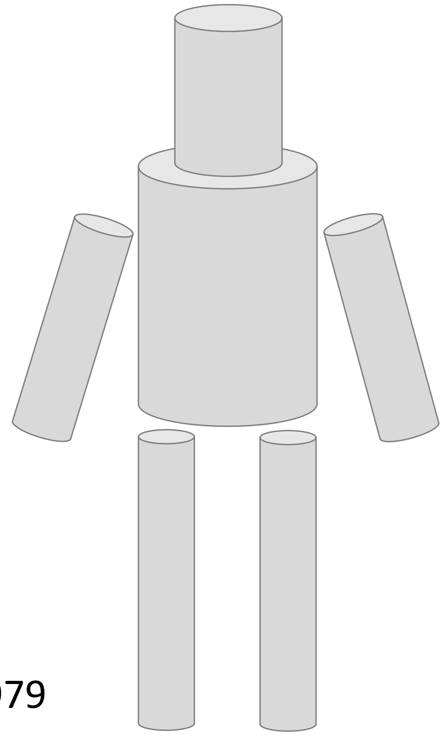
3-D model

This image is CC0 1.0 public domain

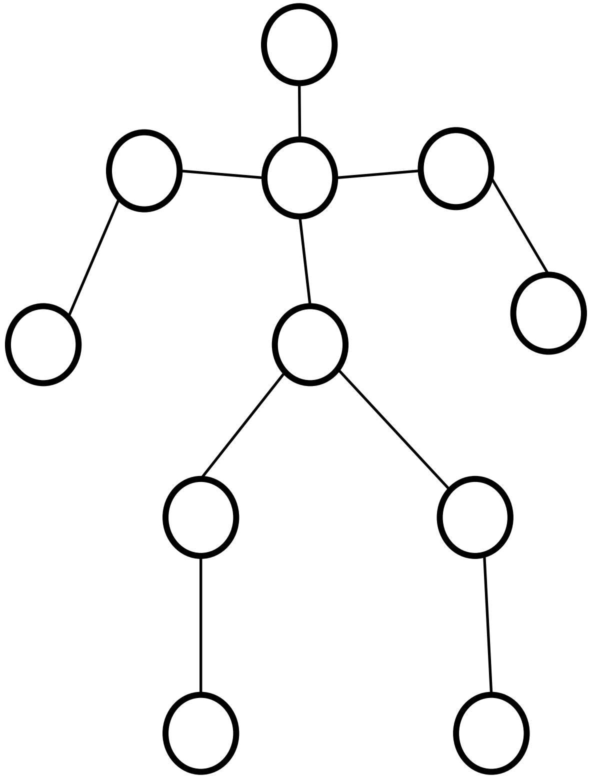
| Input Image | Primal Sketch | 2 ½-D Sketch | 3-D Model Representation |
|---|---|---|---|
| Perceived intensities | Zero crossings, blobs, edges, bars, ends, virtual lines, groups, curves boundaries | Local surface orientation and discontinuities in depth and in surface orientation | 3-D models hierarchically organized in terms of surface and volumetric primitives |

Stages of Visual Representation, David Marr, 1970s

# Recognition via Parts (1970s)



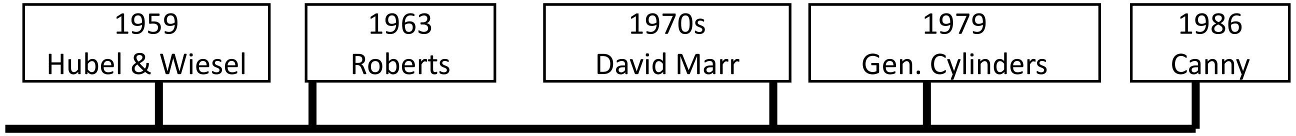
Generalized Cylinders,
Brooks and Binford, 1979



Pictorial Structures,
Fischler and Elshlager, 1973

| 1959 Hubel & Wiesel | 1963 Roberts | 1970s David Marr | 1979 Gen. Cylinders |
|---|---|---|---|

# Recognition via Edge Detection (1980s)




John Canny, 1986
David Lowe, 1987

# Recognition via Grouping (1990s)

1959
Hubel & Wiesel

1963
Roberts

1970s
David Marr

1979
Gen. Cylinders

1986
Canny

1997
Norm. Cuts

AI Winter

Normalized Cuts, Shi and Malik, 1997

# Recognition via Matching (2000s)


Image is public domain


Image is public domain

1959 Hubel & Wiesel | 1963 Roberts | 1970s David Marr | 1979 Gen. Cylinders | 1986 Canny | AI Winter | 1997 Norm. Cuts | 1999 SIFT

SIFT, David Lowe, 1999

# Face Detection

Viola and Jones, 2001

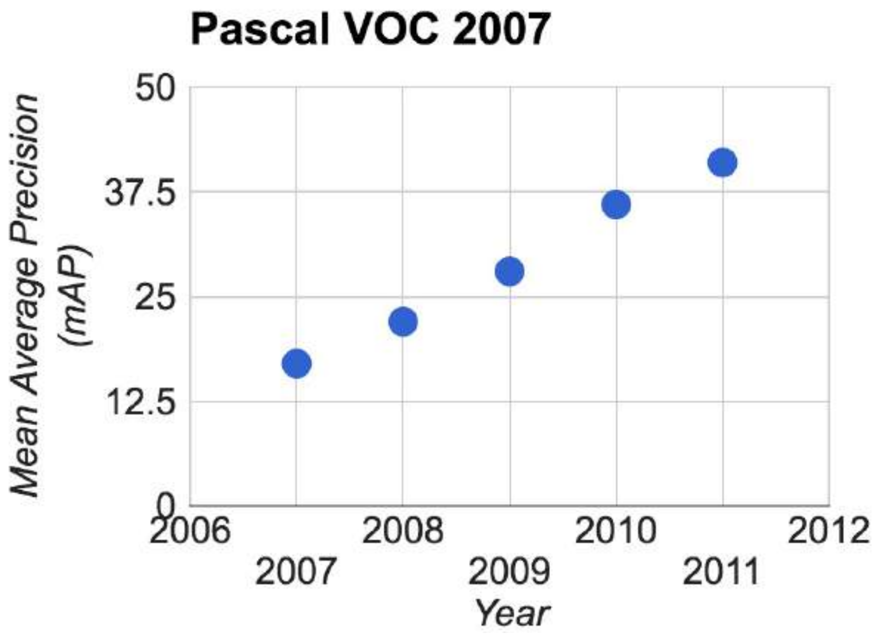
One of the first successful applications of machine learning to vision

1959 Hubel & Wiesel | 1963 Roberts | 1970s David Marr | 1979 Gen. Cylinders | 1986 Canny | AI Winter | 1997 Norm. Cuts | 1999 SIFT | 2001 V&J

# PASCAL Visual Object Challenge

Image is CC0 1.0 public domain


Image is CC0 1.0 public domain




1959 Hubel & Wiesel | 1963 Roberts | 1970s David Marr | 1979 Gen. Cylinders | 1986 Canny | AI Winter | 1997 Norm. Cuts | 1999 SIFT | 2001 V&J | 2007 PASCAL

# IMAGENET Large Scale Visual Recognition Challenge

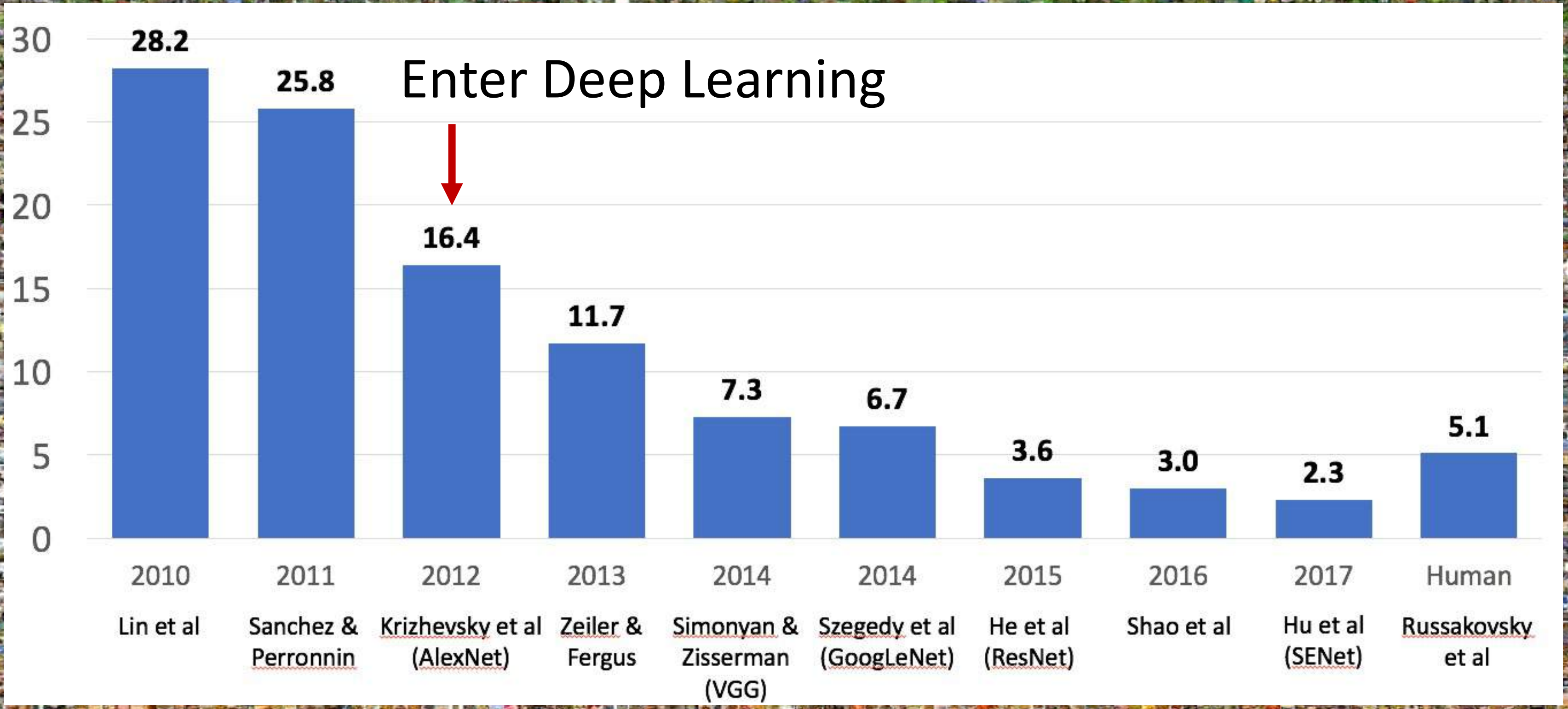
The Image Classification Challenge:
1,000 object classes
1,431,167 images

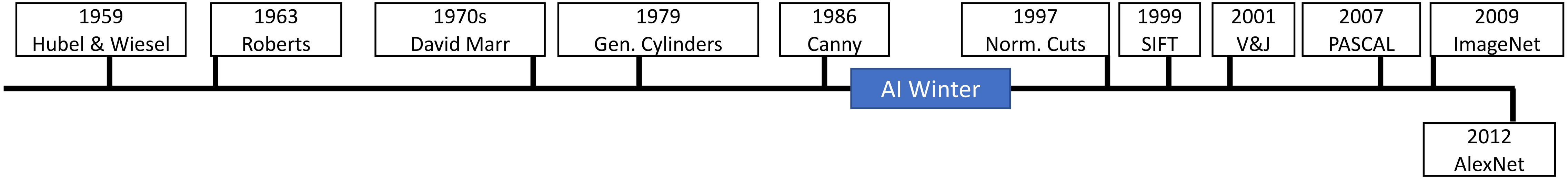
**Output:**
Scale
T-shirt
<u>Steel drum</u>
Drumstick
Mud turtle

Deng et al, 2009
Russakovsky et al. IJCV 2015

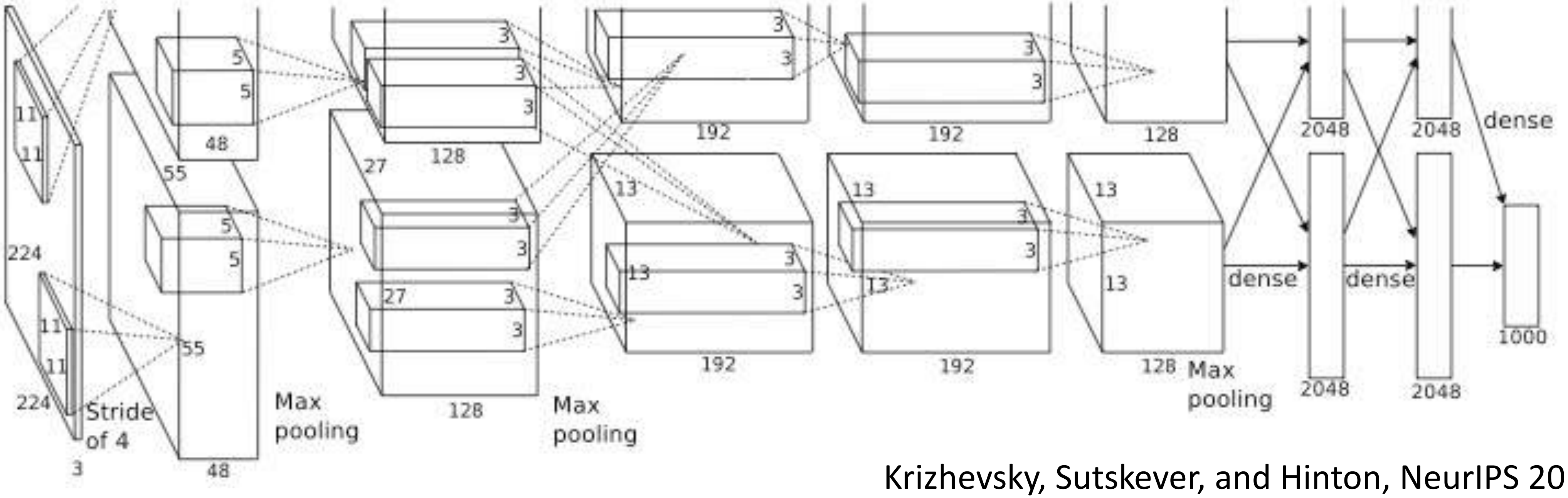
| 1959 Hubel & Wiesel | 1963 Roberts | 1970s David Marr | 1979 Gen. Cylinders | 1986 Canny | AI Winter | 1997 Norm. Cuts | 1999 SIFT | 2001 V&J | 2007 PASCAL | 2009 ImageNet |
|---|---|---|---|---|---|---|---|---|---|---|

# IMAGENET Large Scale Visual Recognition Challenge



# AlexNet: Deep Learning Goes Mainstream




Krizhevsky, Sutskever, and Hinton, NeurIPS 2012

1959 Hubel & Wiesel

1963 Roberts

1970s David Marr

1979 Gen. Cylinders

1986 Canny

AI Winter

1997 Norm. Cuts

1999 SIFT

2001 V&J

2007 PASCAL

2009 ImageNet

2012 AlexNet

# Perceptron

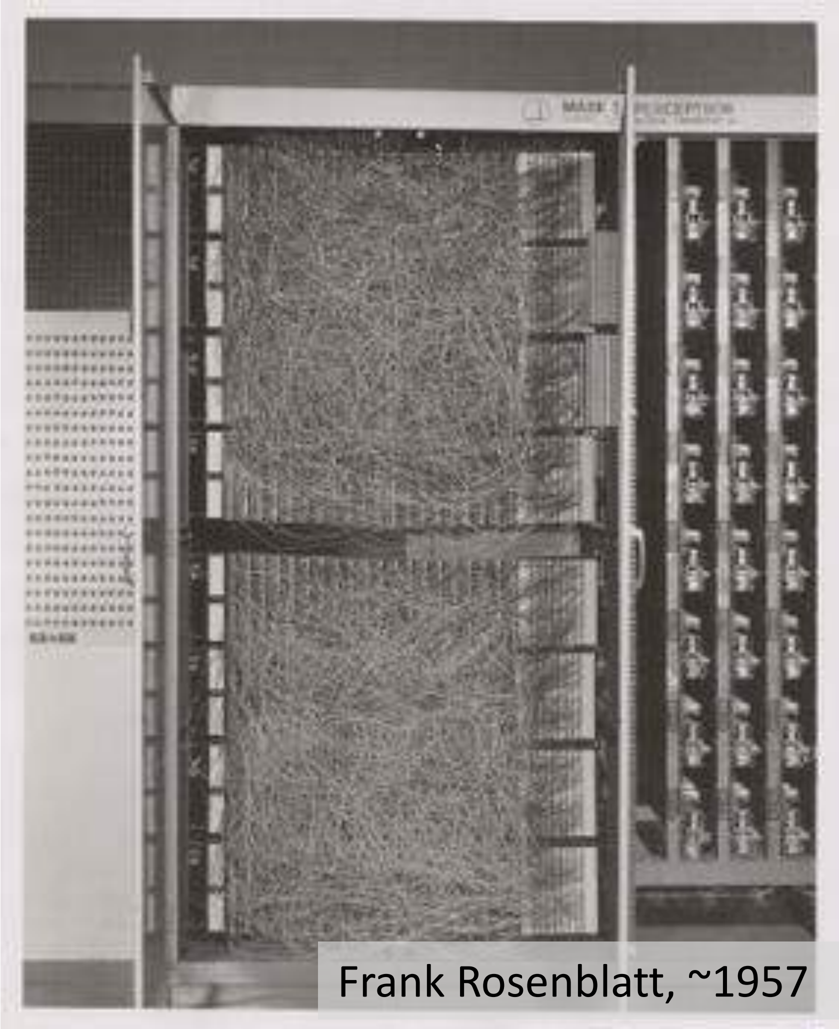
One of the earliest algorithms that could learn from data

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

1959 Hubel & Wiesel

1963 Roberts

1970s David Marr

1979 Gen. Cylinders

1986 Canny

1997 Norm. Cuts

1999 SIFT

2001 V&J

2007 PASCAL

2009 ImageNet

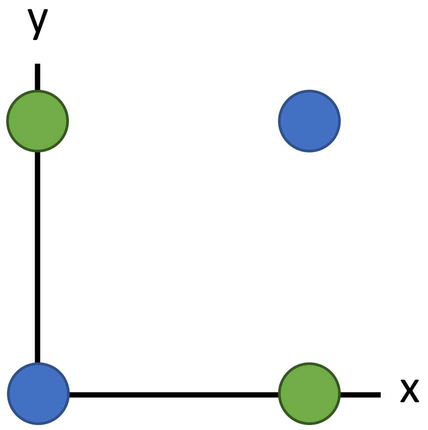
AI Winter

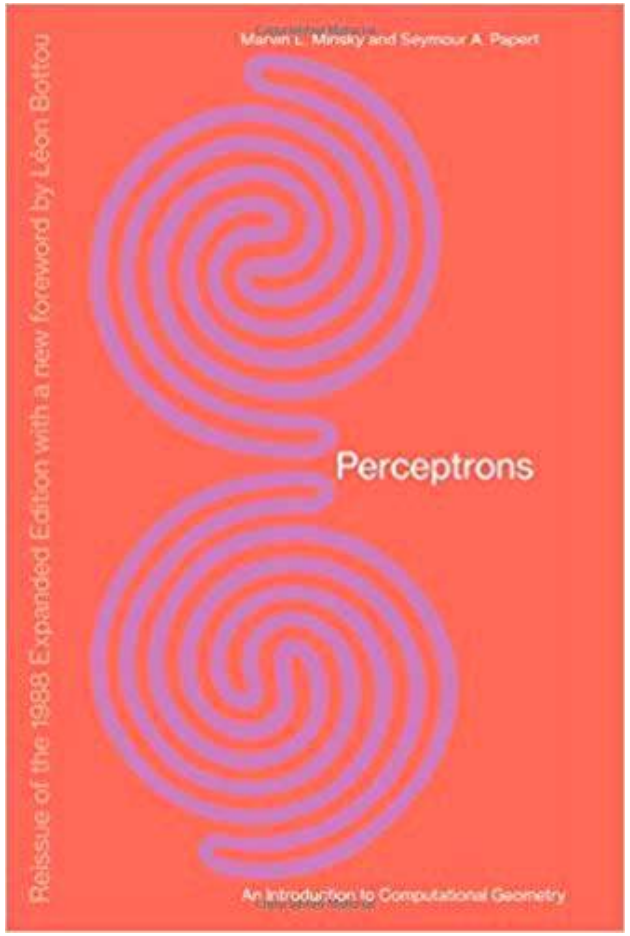
1958 Perceptron

2012 AlexNet

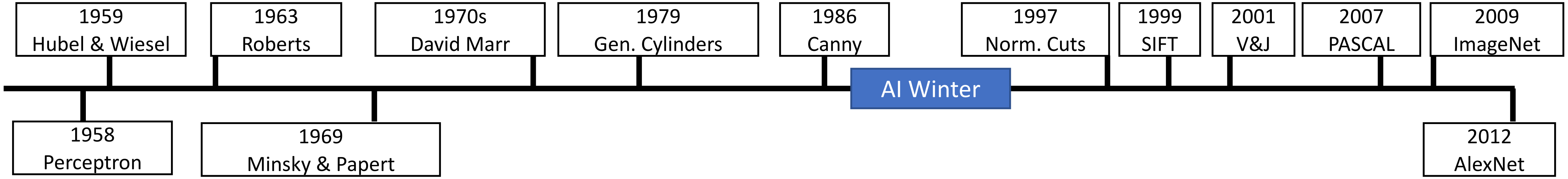
# Minsky and Papert, 1969

| 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

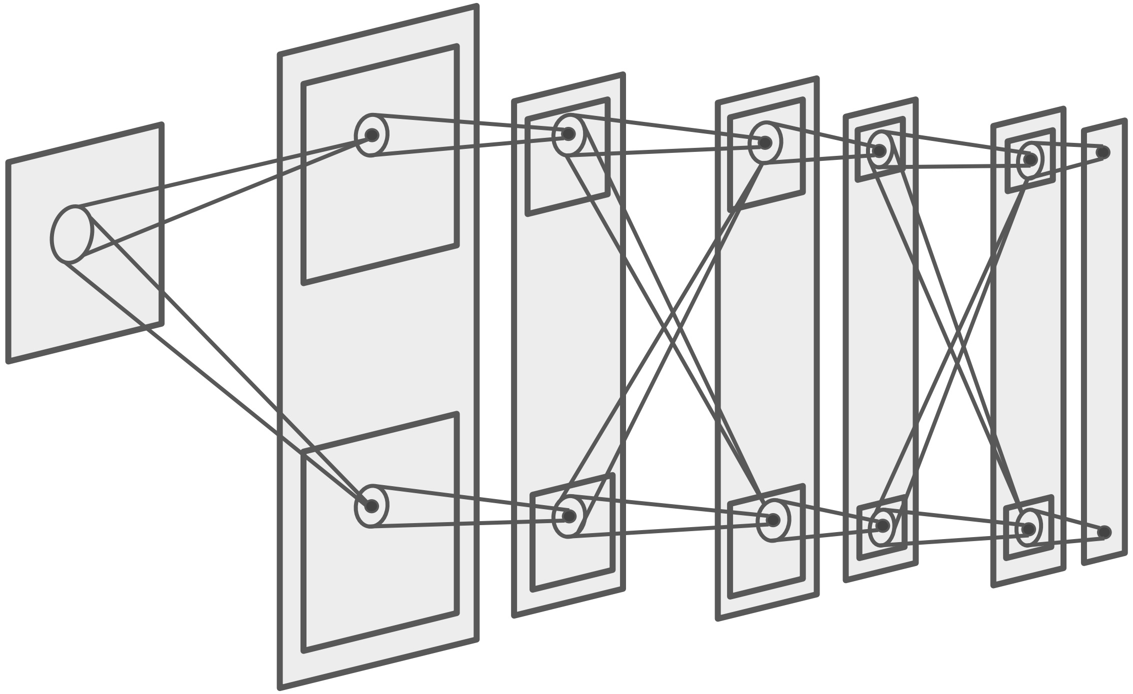


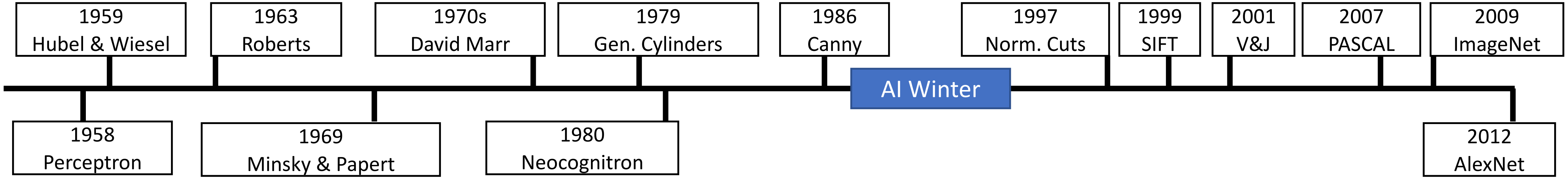
# Neocognitron: Fukushima, 1980

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

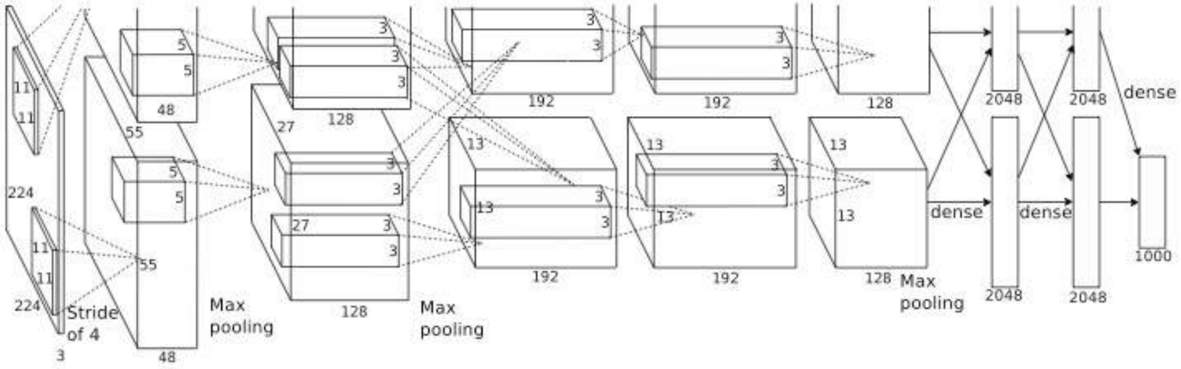
Interleaved simple cells (convolution) and complex cells (pooling)

No practical training algorithm

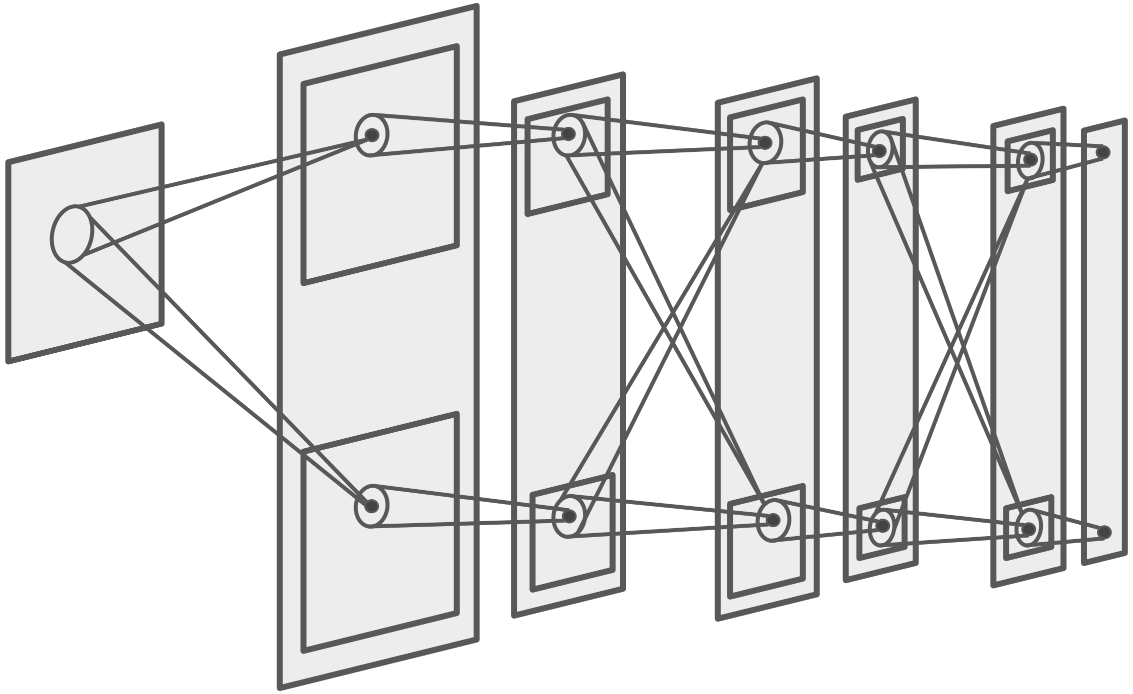




# Neocognitron: Fukushima, 1980




Looks a lot like AlexNet
more than 32 years later!



1959 Hubel & Wiesel | 1963 Roberts | 1970s David Marr | 1979 Gen. Cylinders | 1986 Canny | 1997 Norm. Cuts | 1999 SIFT | 2001 V&J | 2007 PASCAL | 2009 ImageNet

AI Winter

1958 Perceptron | 1969 Minsky & Papert | 1980 Neocognitron | 2012 AlexNet

# 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

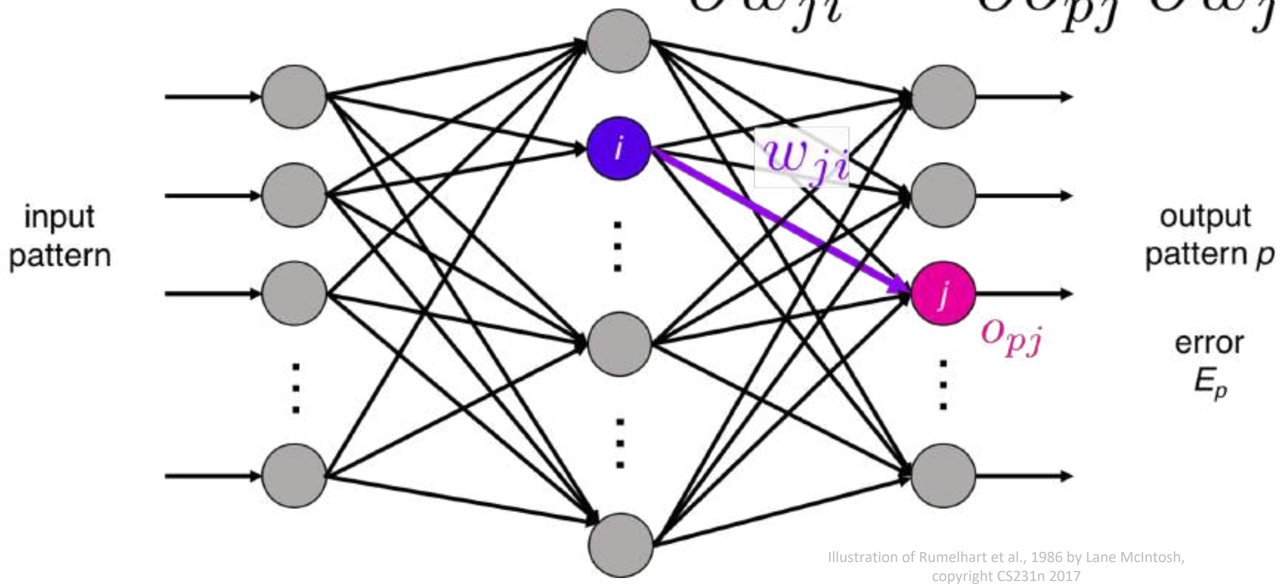
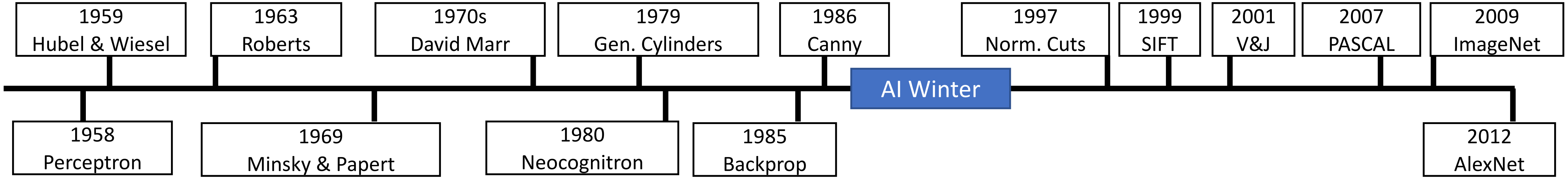
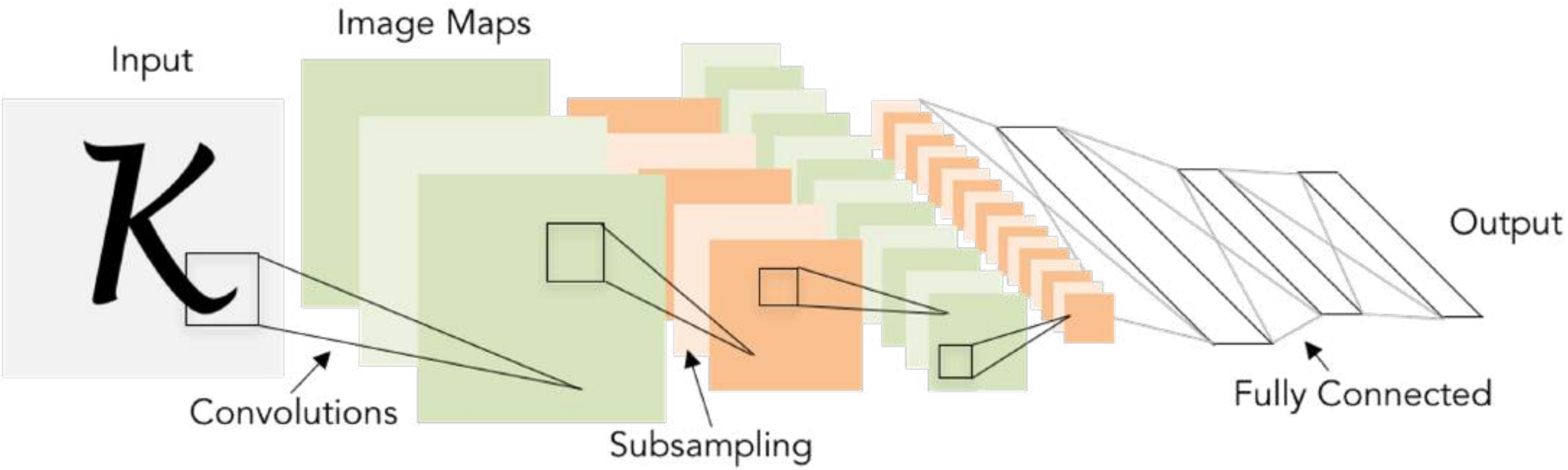


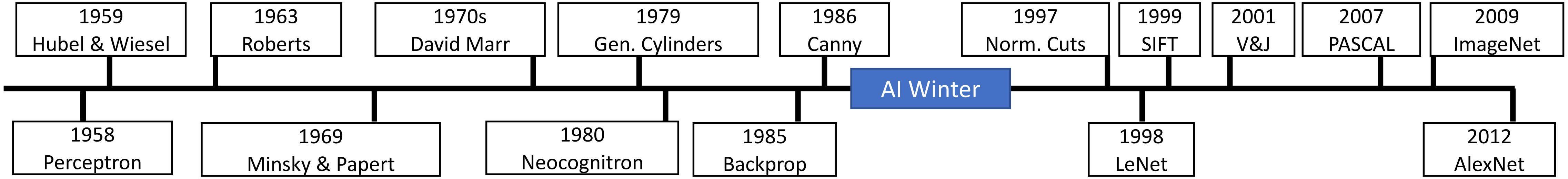

Illustration of Rumelhart et al., 1986 by Lane McIntosh, copyright CS231n 2017



# 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!

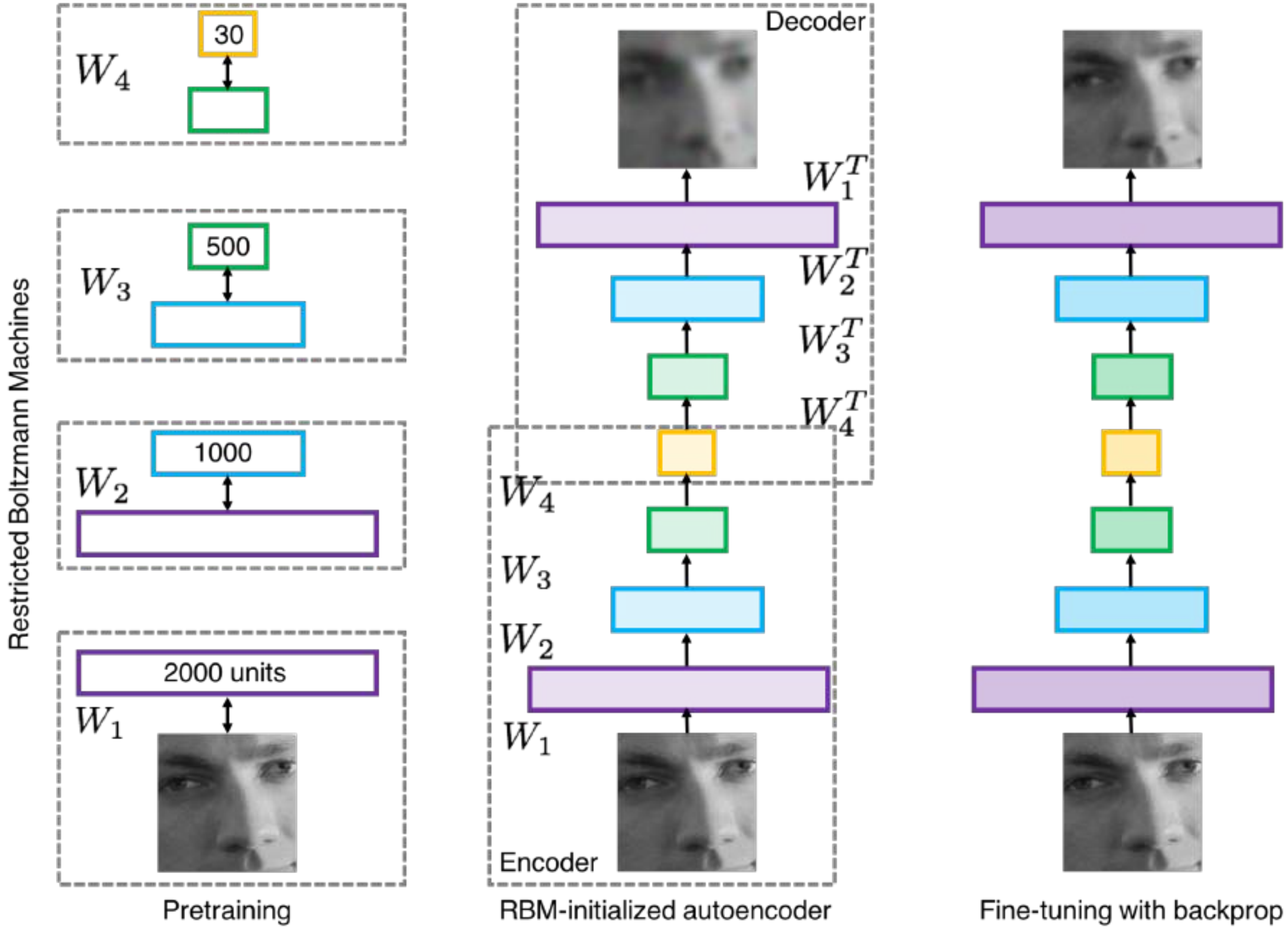


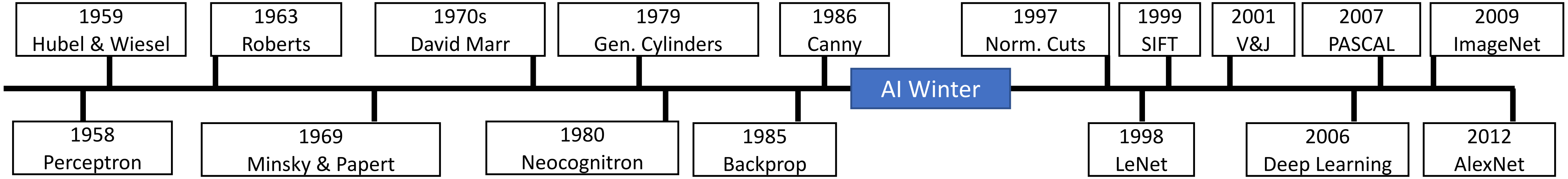
# 2000s: "Deep Learning"

People tried to train neural networks that were deeper and deeper

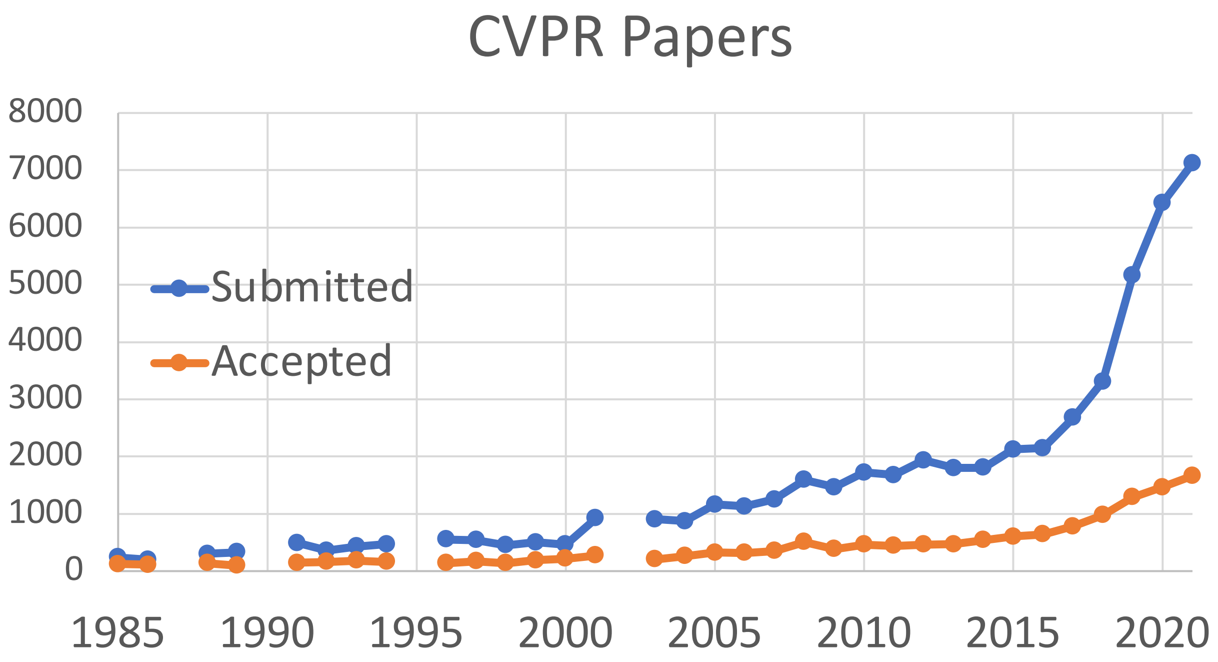
Not a mainstream research topic at this time

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

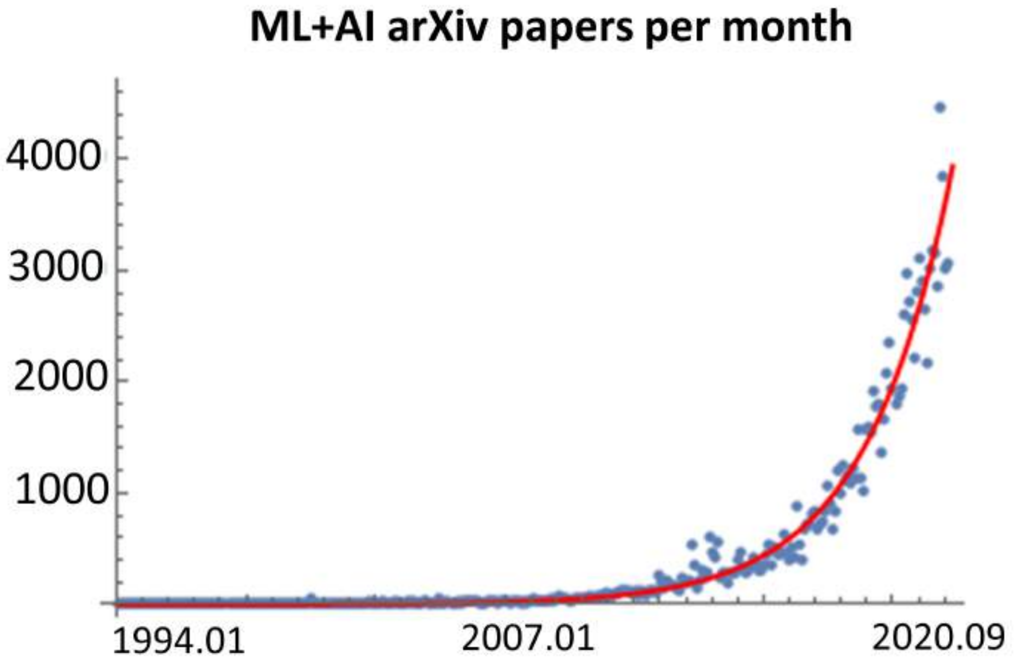


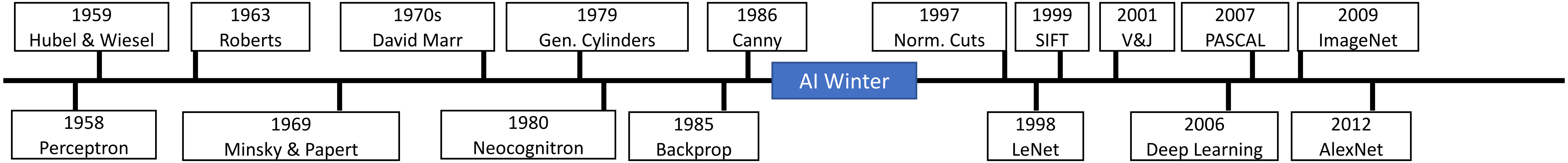
# 2012 to Present: Deep Learning Explosion




Publications at top Computer Vision conference




arXiv papers per month (source)



# 2012 to Present: Deep Learning is Everywhere

Image Classification

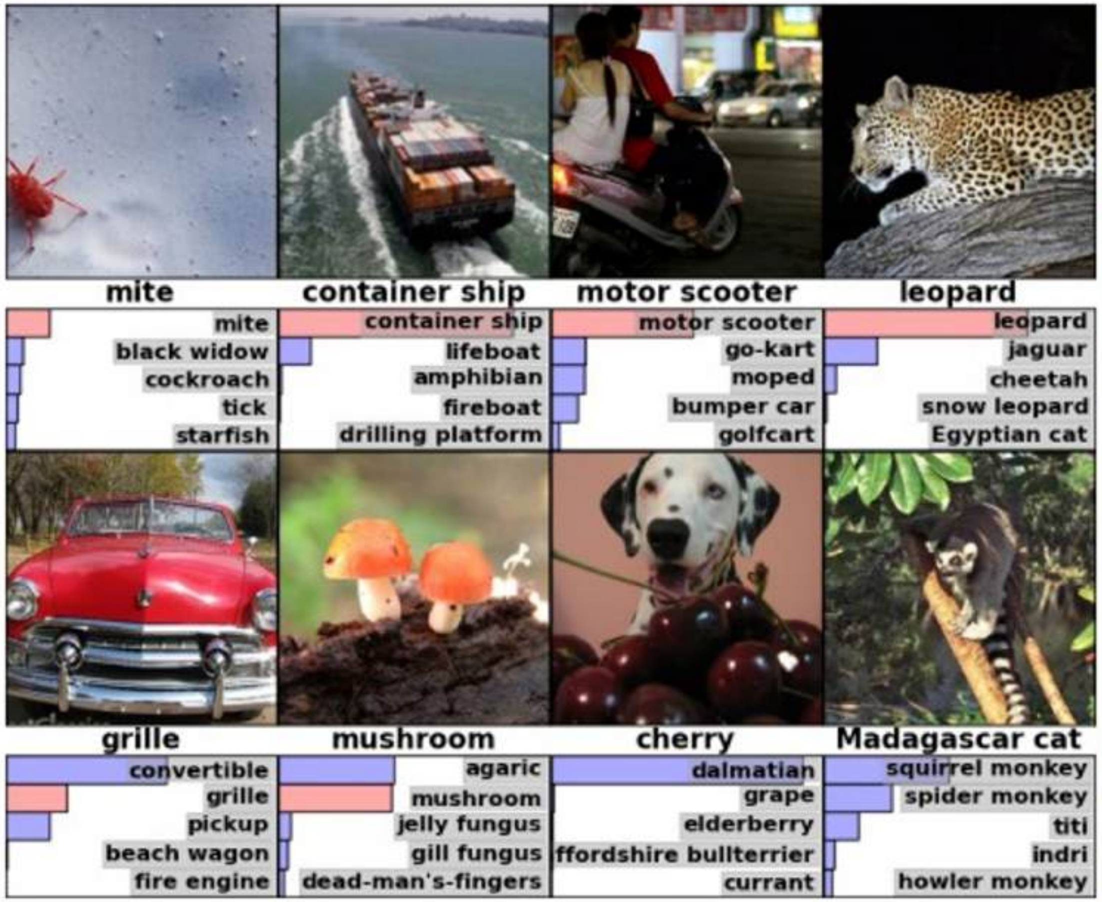
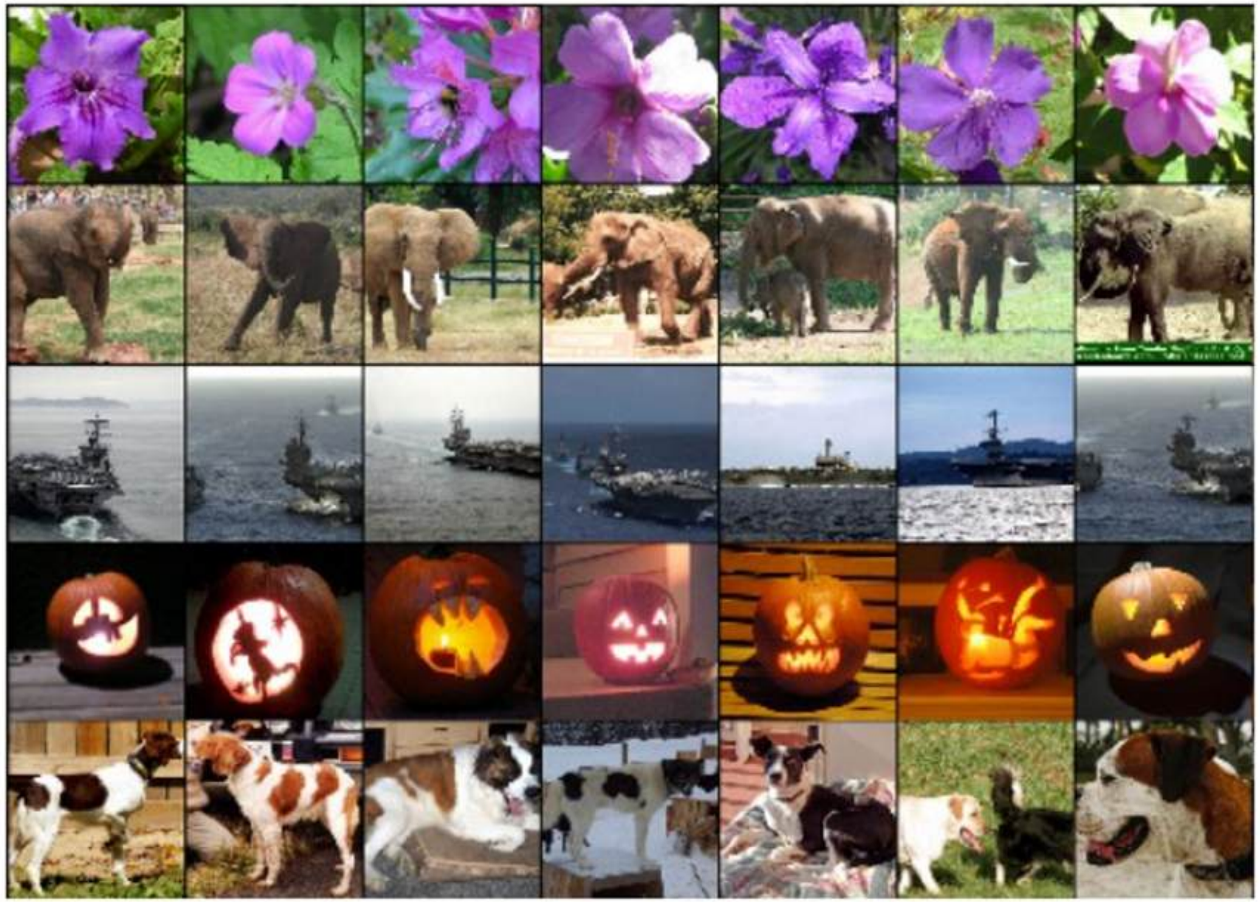
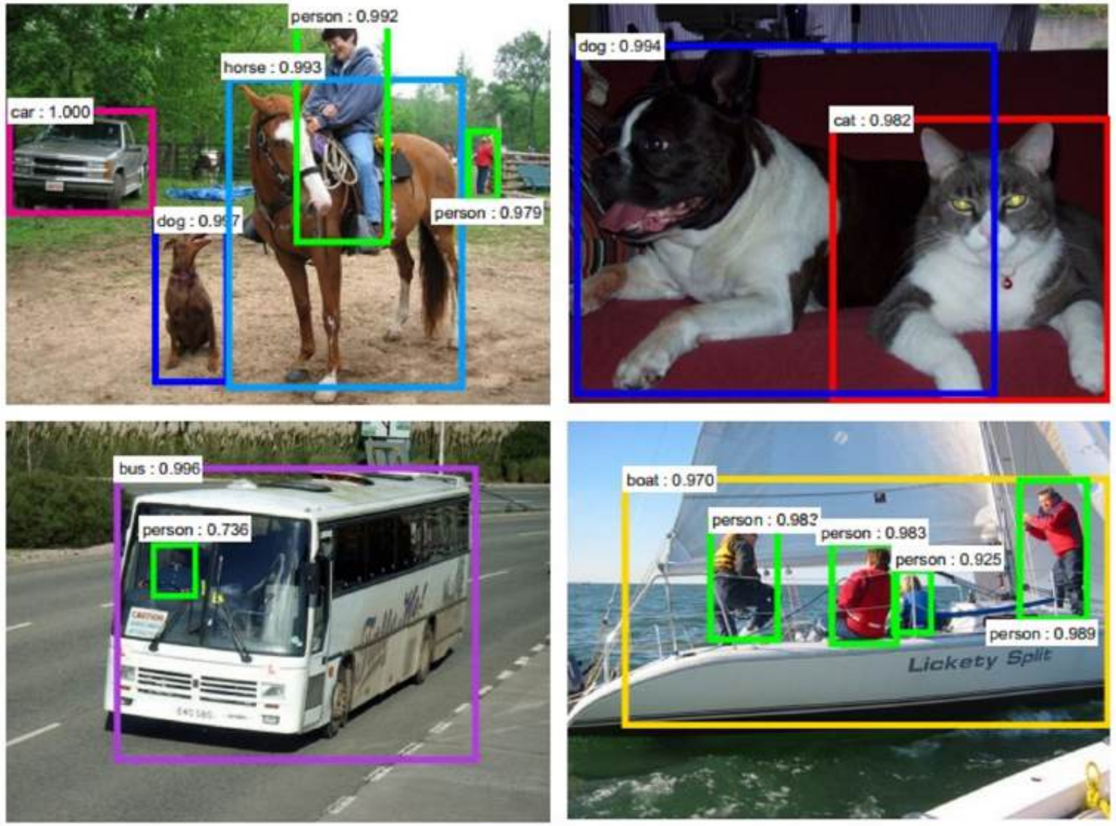



Image Retrieval



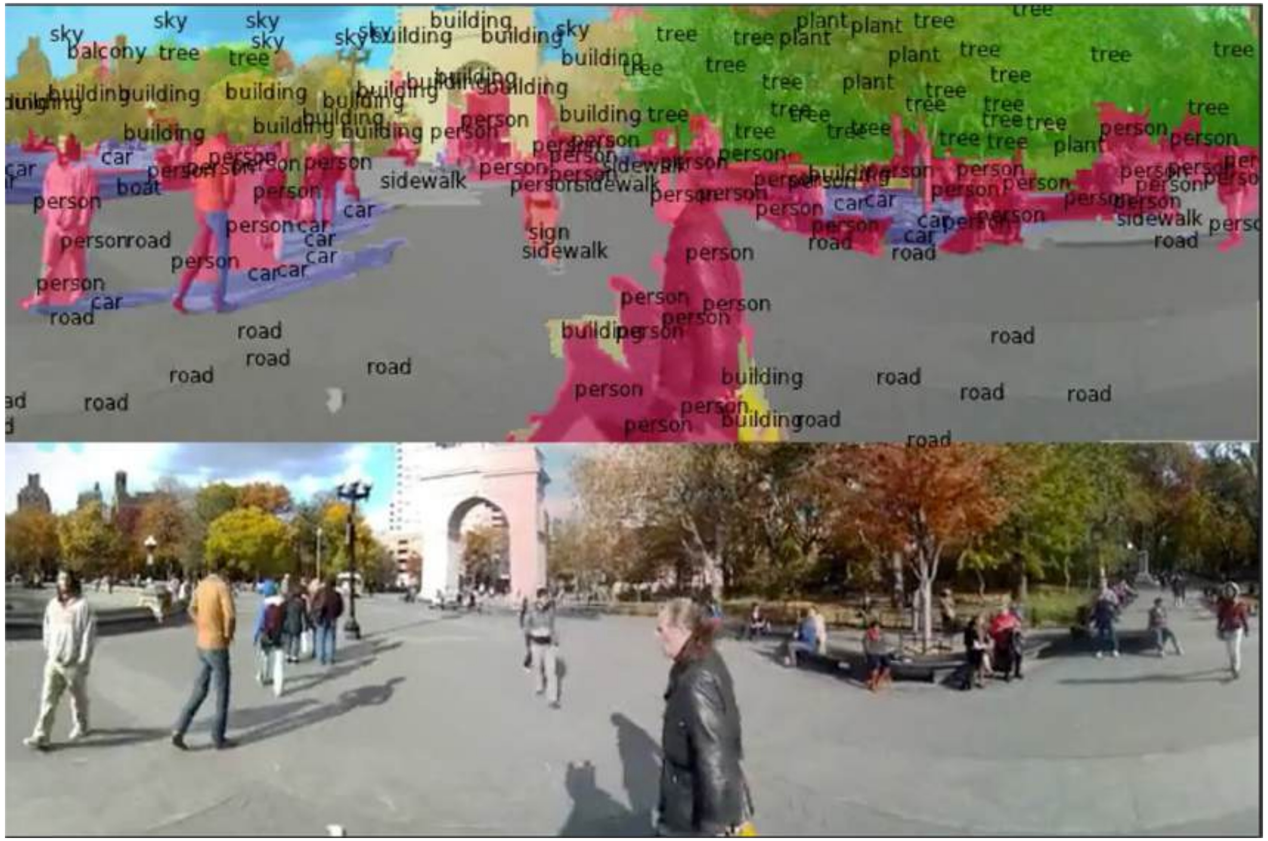
# 2012 to Present: Deep Learning is Everywhere

Object Detection




Ren, He, Girshick, and Sun, 2015

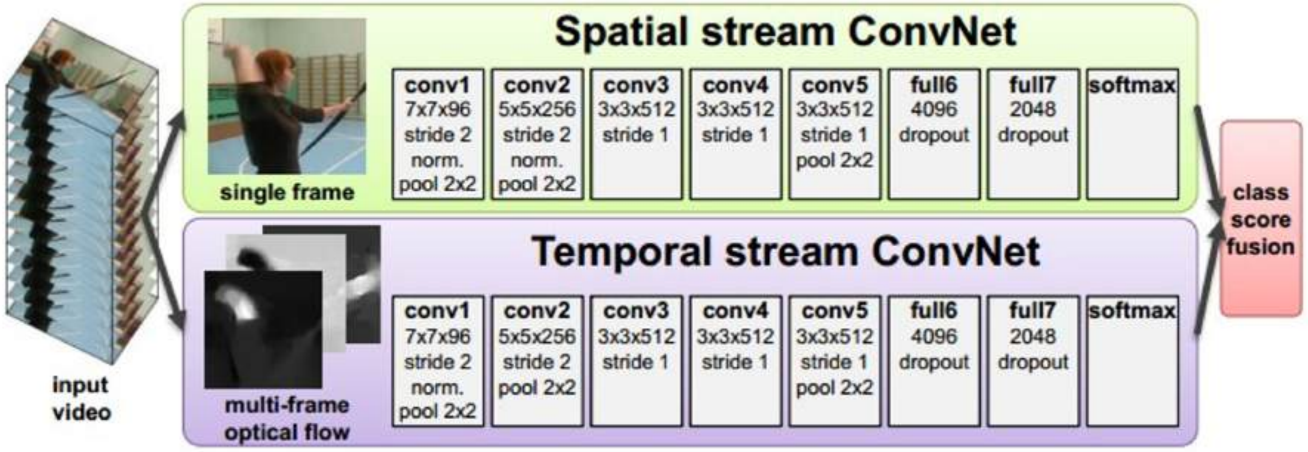
Image Segmentation



Fabaret et al, 2012

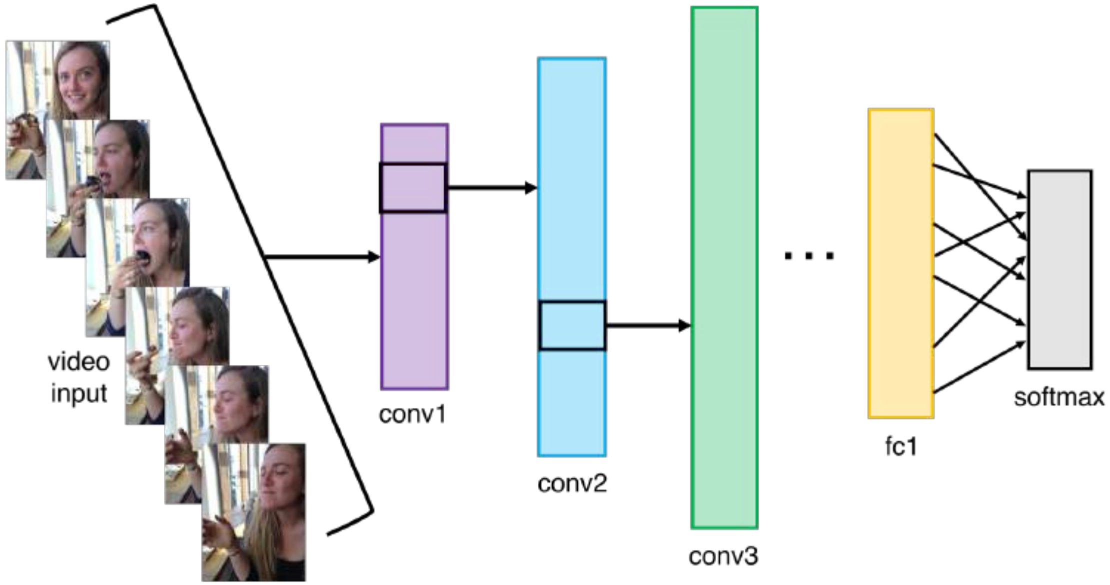
# 2012 to Present: Deep Learning is Everywhere

Video Classification




Simonyan et al, 2014

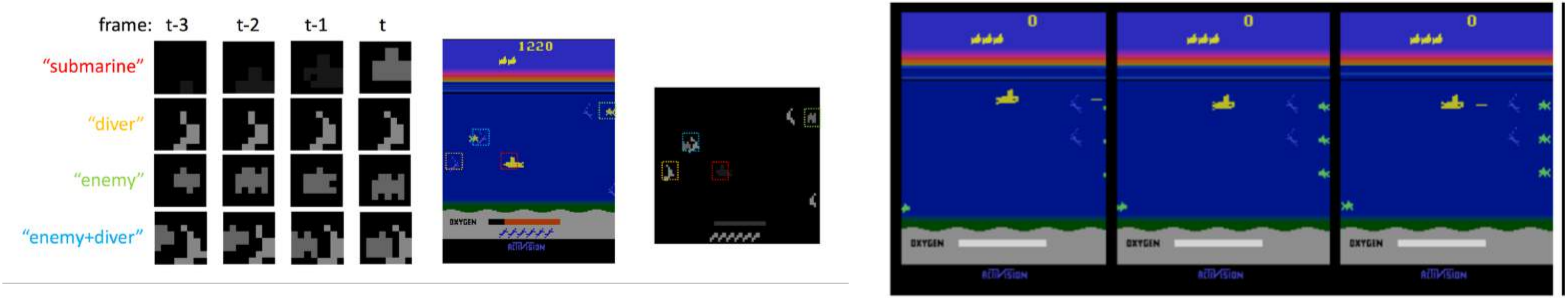
Activity Recognition



# 2012 to Present: Deep Learning is Everywhere

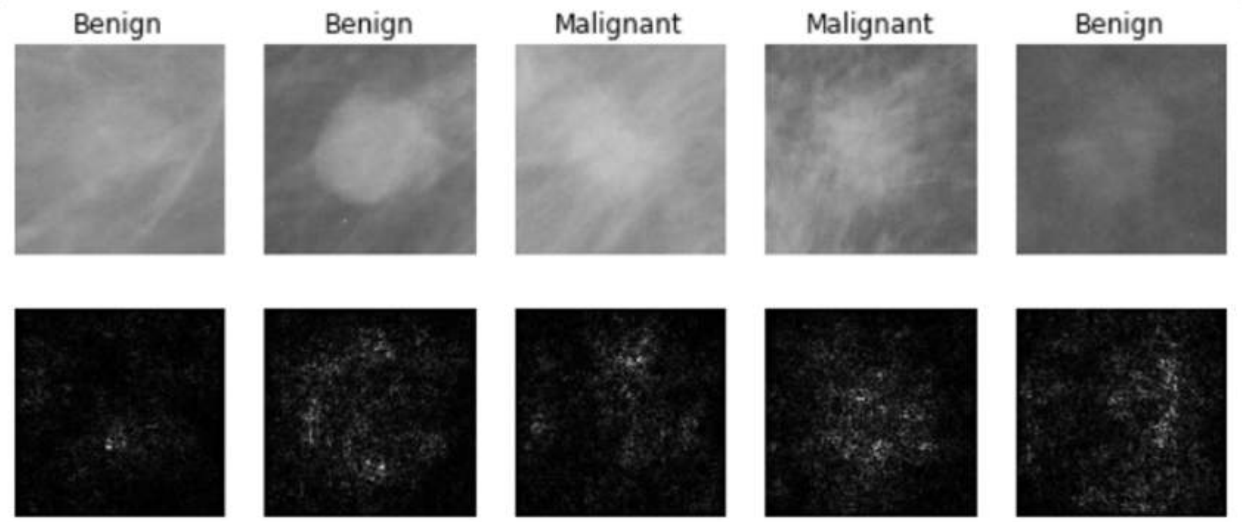
Pose Recognition (Toshev and Szegedy, 2014)

Playing Atari games (Guo et al, 2014)



# 2012 to Present: Deep Learning is 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: Deep Learning is Everywhere

## Image Captioning

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

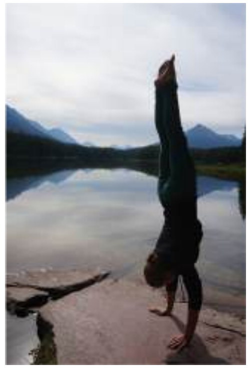
*A white teddy bear sitting in the grass*

*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*

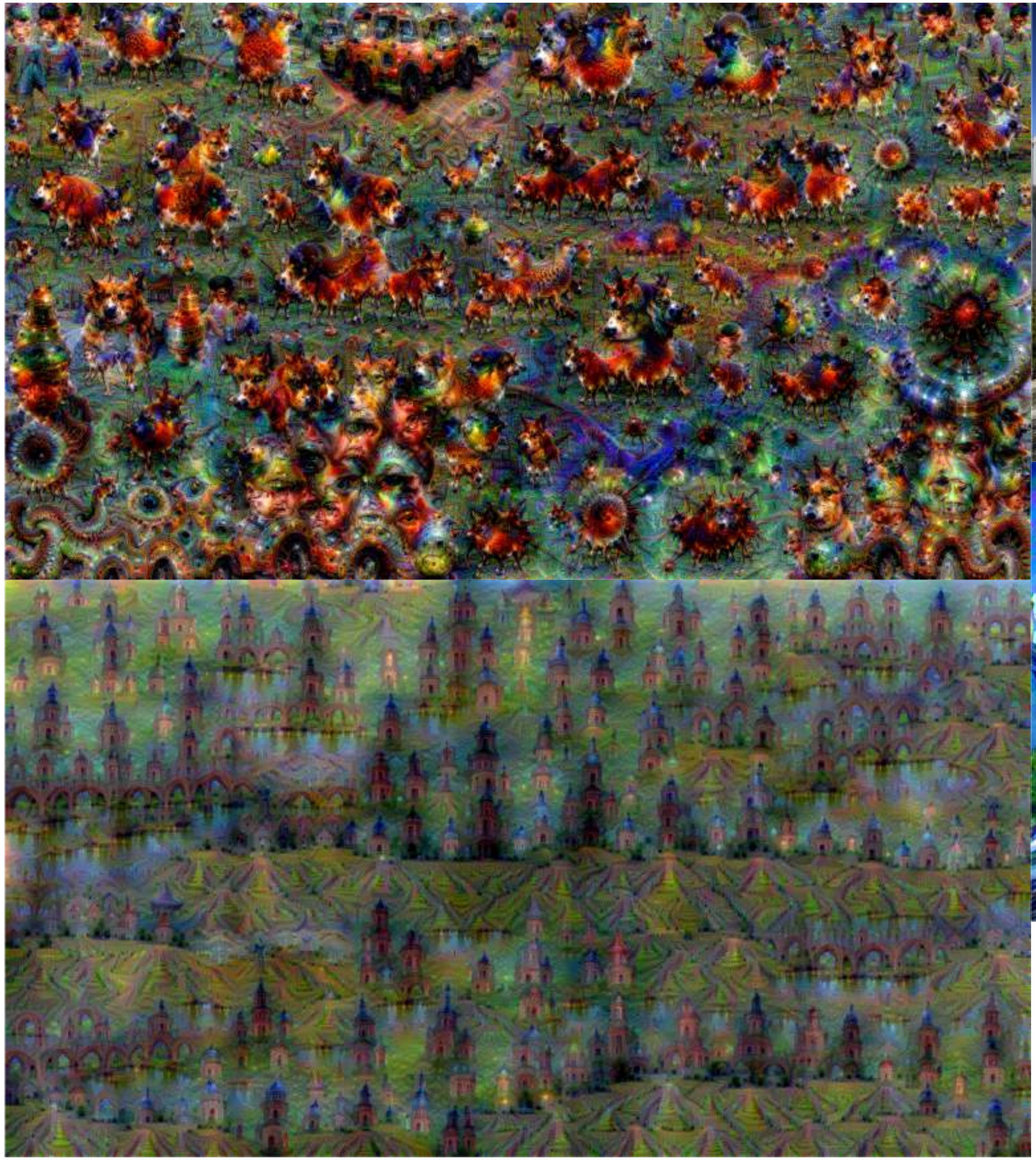
*A cat sitting on a suitcase on the floor*



*A woman standing on a beach holding a surfboard*



Captions generated by Justin Johnson using Neuraltalk2

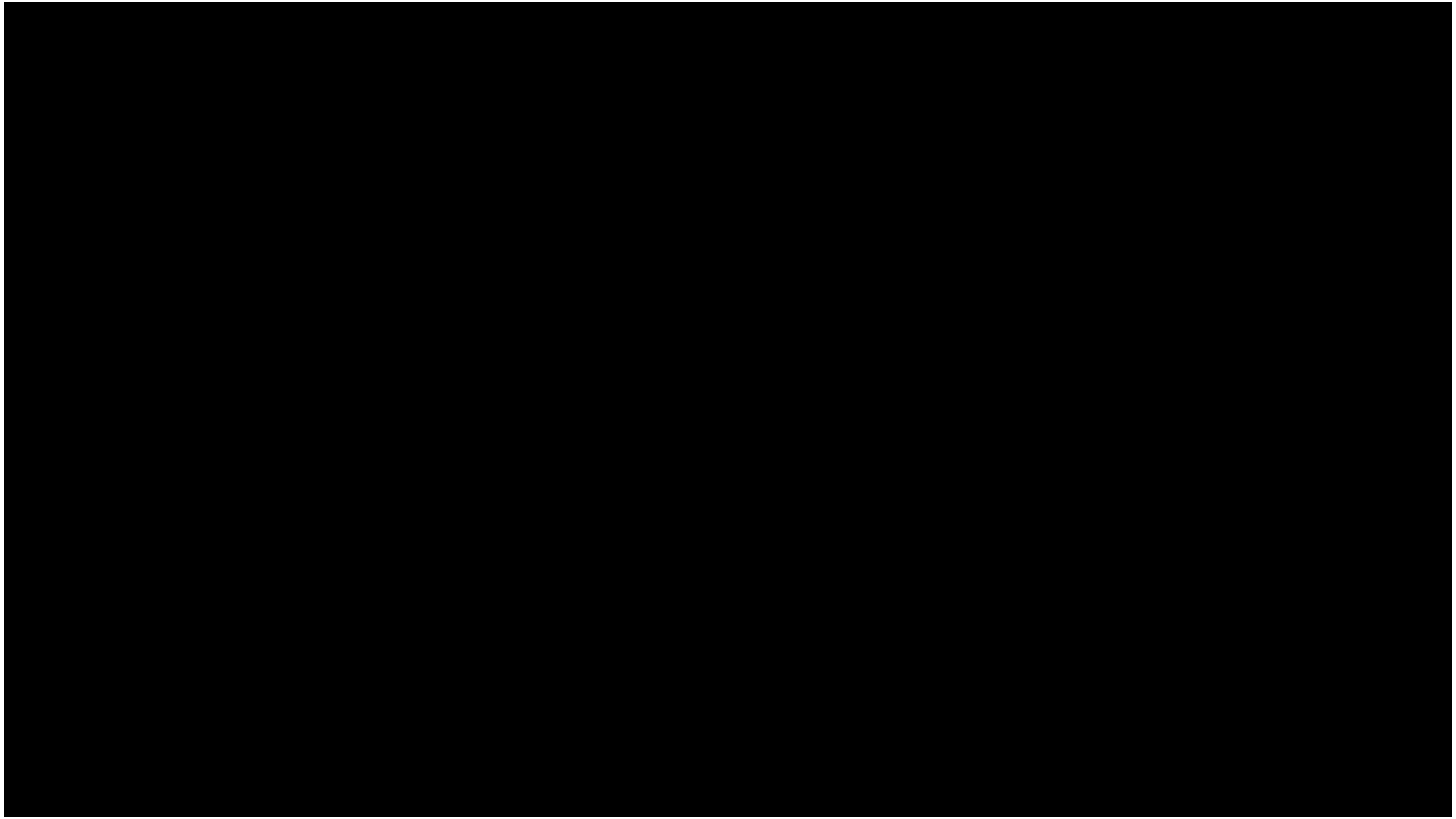






Mordvinsev et al, 2015
Gatys et al, 2016

# 2012 to Present: Deep Learning is Everywhere

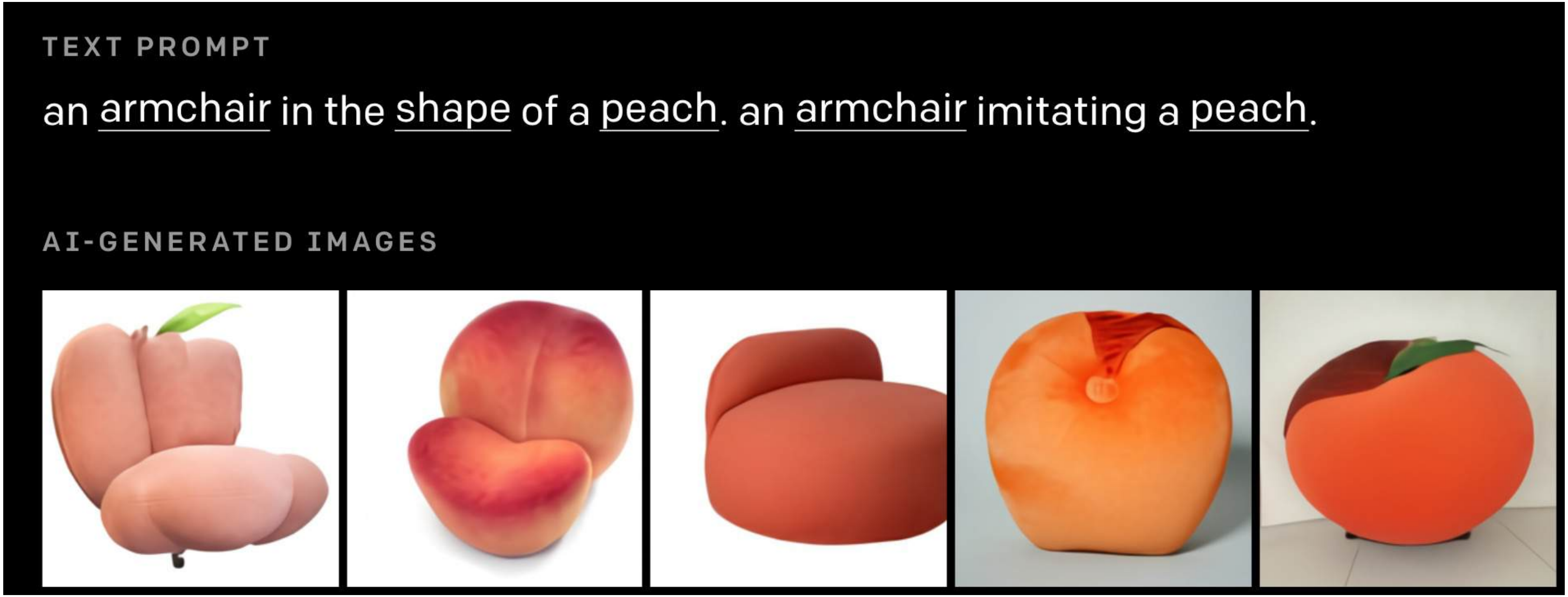


Karras et al, "Progressive Growing of GANs for Improved Quality, Stability, and Variation", ICLR 2018

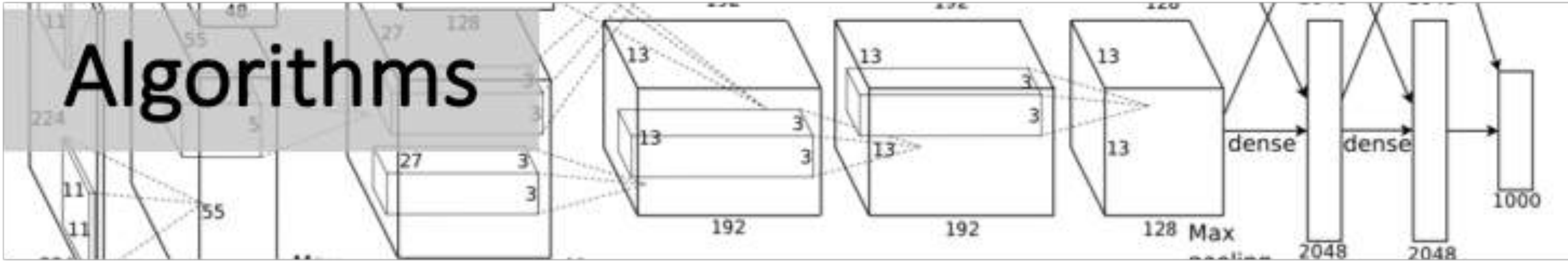
# 2012 to Present: Deep Learning is Everywhere


Ramesh et al, “DALL·E: Creating Images from Text”, 2021. https://openai.com/blog/dall-e/

# 2012 to Present: Deep Learning is Everywhere



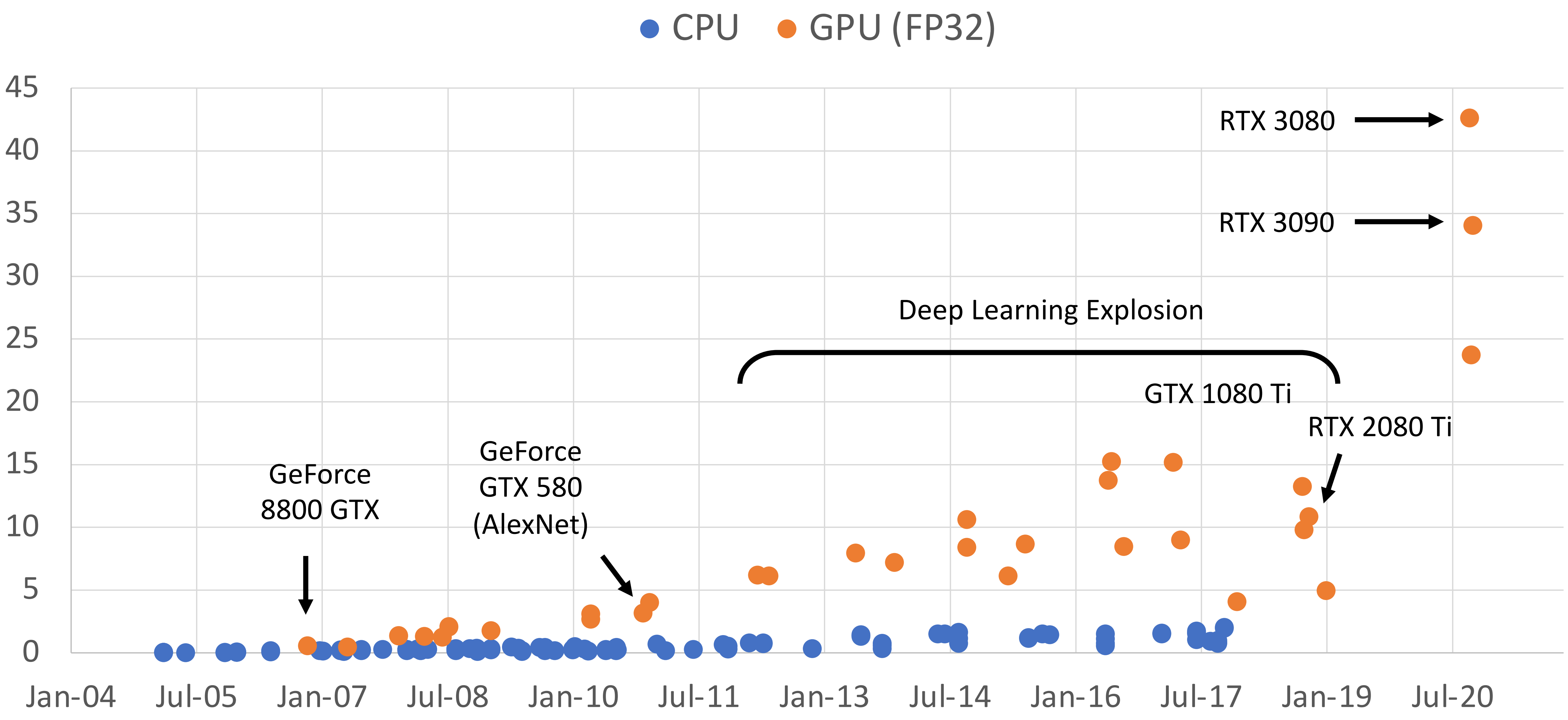

Ramesh et al, “DALL·E: Creating Images from Text”, 2021. https://openai.com/blog/dall-e/


Algorithms
13
13
3
3
192
13
13
3
3
192
13
13
128
Max
dense
dense
2048
2048
1000
27
3
3
11
11
55

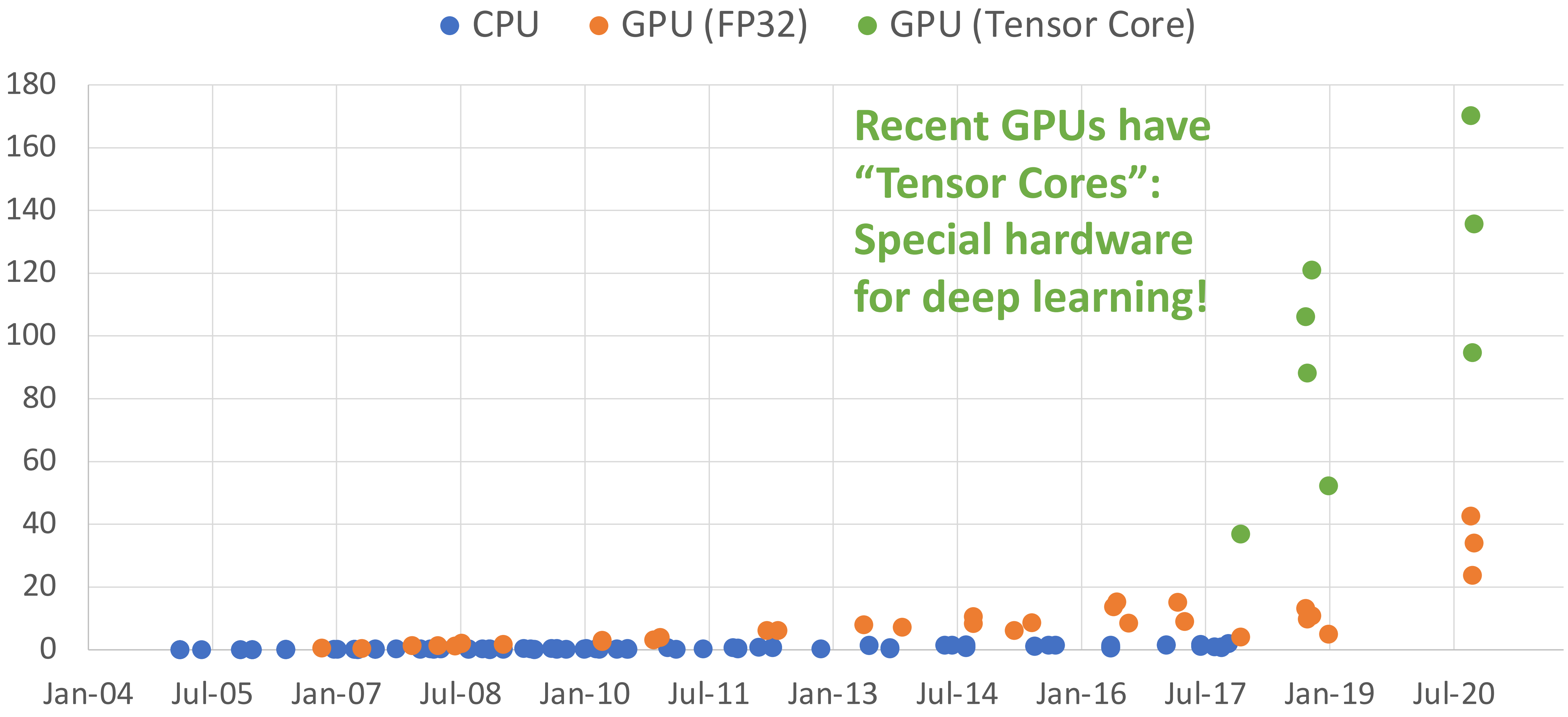

Data


Computation
GTX
GTX

# GFLOP per Dollar



# GFLOP per Dollar



# 2018 Turing Award

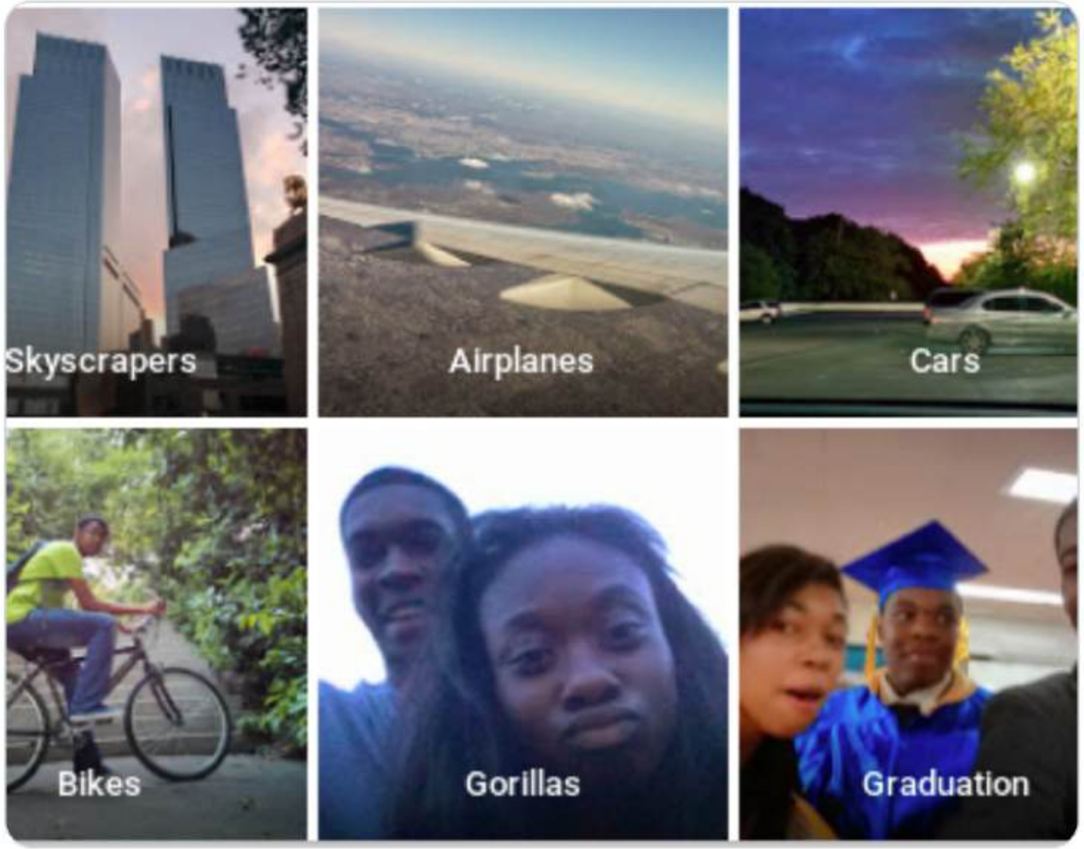



- 1958 Perceptron
- 1959 Hubel & Wiesel
- 1963 Roberts
- 1969 Minsky & Papert
- 1970s David Marr
- 1979 Gen. Cylinders
- 1980 Neocognitron
- 1985 Backprop
- 1986 Canny
- AI Winter
- 1997 Norm. Cuts
- 1998 LeNet
- 1999 SIFT
- 2001 V&J
- 2006 Deep Learning
- 2007 PASCAL
- 2009 ImageNet
- 2012 AlexNet
- 2018 Turing Award

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

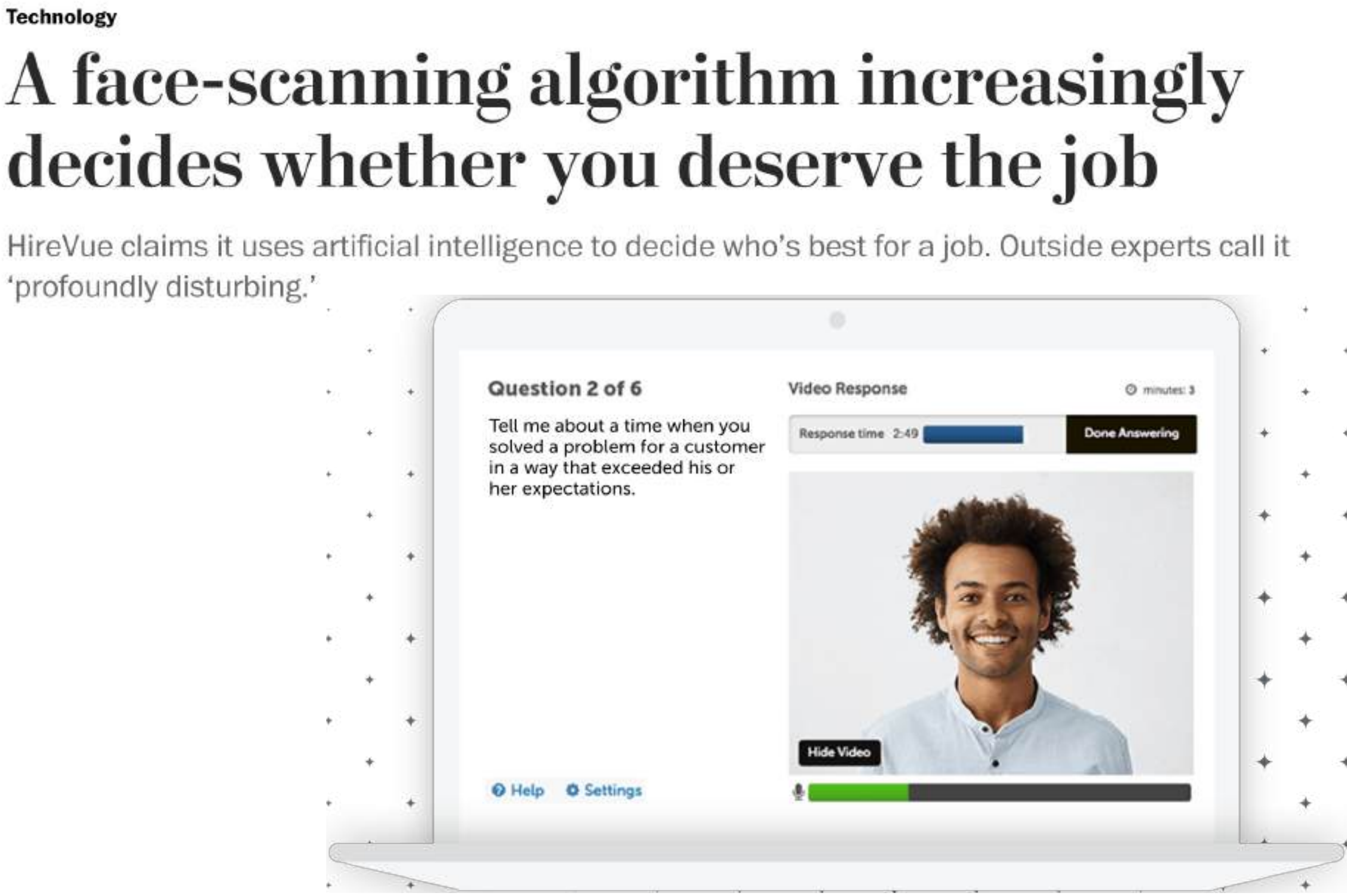
# Computer Vision can cause harm

## Harmful Stereotypes




Barocas et al, "The Problem With Bias: Allocative Versus Representational Harms in Machine Learning", SIGCIS 2017
Kate Crawford, "The Trouble with Bias", NeurIPS 2017 Keynote
Source: https://twitter.com/jackyalcine/status/615329515909156865 (2015)

## Affect people's lives




Source: https://www.washingtonpost.com/technology/2019/10/22/ai-hiring-face-scanning-algorithm-increasingly-decides-whether-you-deserve-the-job/
https://www.hirevue.com/platform/online-video-interviewing-software
Example Credit: Timnit Gebru



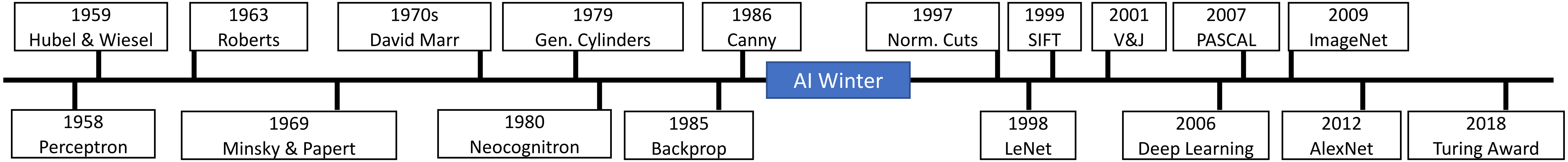
Example credit: Andrej Karpathy

# Computer Vision Technology Can Better Our Lives

# Today's Agenda

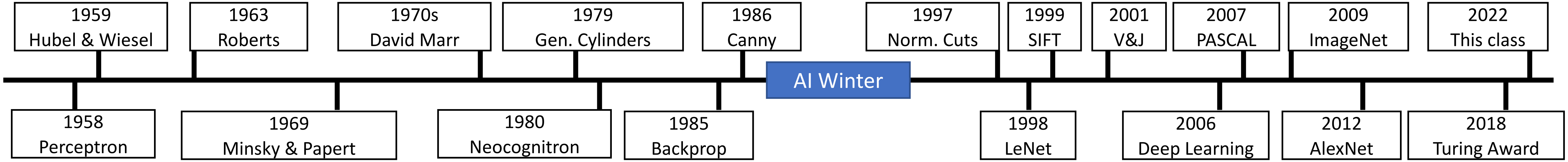
- A brief history of computer vision and deep learning

- 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

## GSIs / IAs

Karan Desai (KD)

Janpreet Singh (JS)

Jim Yang

Wallace Sui (WS)

Gaurav Kaul

# 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/kxtai72amx34p0
  - (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
- EECS Autograder: https://autograder.io/web/course/151
  - For turning in homework assignments
- Google Calendar: For office hours (starting next week)
- Email: Only for sensitive, confidential issues

# Course Website: Check the Schedule!

EECS 498.008 / 598.008
Deep Learning for Computer Vision
Winter 2022

## Schedule

Lectures are Mondays and Wednesdays, 4:30pm to 6pm. Attendance is not required. Recordings will be posted after each lecture in case you are unable the attend the scheduled time.

Some lectures have reading drawn from the course notes of Stanford CS 231n, written by Andrej Karpathy.

Some lectures have optional reading from the book *Deep Learning* by Ian Goodfellow, Yoshua Bengio, and Aaron Courville (GBC for short). The entire text of the book is available for free online so you don't need to buy a copy.

| Event | Date | Description | Course Materials |
|---|---|---|---|
| Lecture 1 | Wednesday January 5 | **Course Introduction**<br>Computer vision overview<br>Historical context<br>Course logistics | [slides]<br>[FA2019 video (public)]<br>[Python tutorial]<br>[GBC Sec 1.2]<br>[GBC Sec 6.6] |
| Lecture 2 | Monday January 10 | **Image Classification**<br>Data-driven approach<br>K-Nearest Neighbor<br>Hyperparameters<br>Cross-validation | [slides]<br>[FA2019 video (public)]<br>[231n Image Classification] |
| Lecture 3 | Wednesday January 12 | **Linear Classifiers**<br>Algebraic / Visual / Geometric viewpoints<br>Softmax / SVM classifiers | [slides]<br>[FA2019 video (public)]<br>[231n Linear Classification] |
| A1 Due | Friday January 14 | **Assignment 1 Due**<br>PyTorch warmup<br>kNN Classifier | [Assignment 1] |
| | Monday January 17 | **No class**<br>MLK Day | |

https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/schedule.html

# Piazza Etiquette

- Post only short snippets of code (< 20-30 lines)
- Ask a specific, concrete question
- Explain what you have tried so far, and what happened
- See StackOverflow guide on asking good questions: https://stackoverflow.com/help/how-to-ask

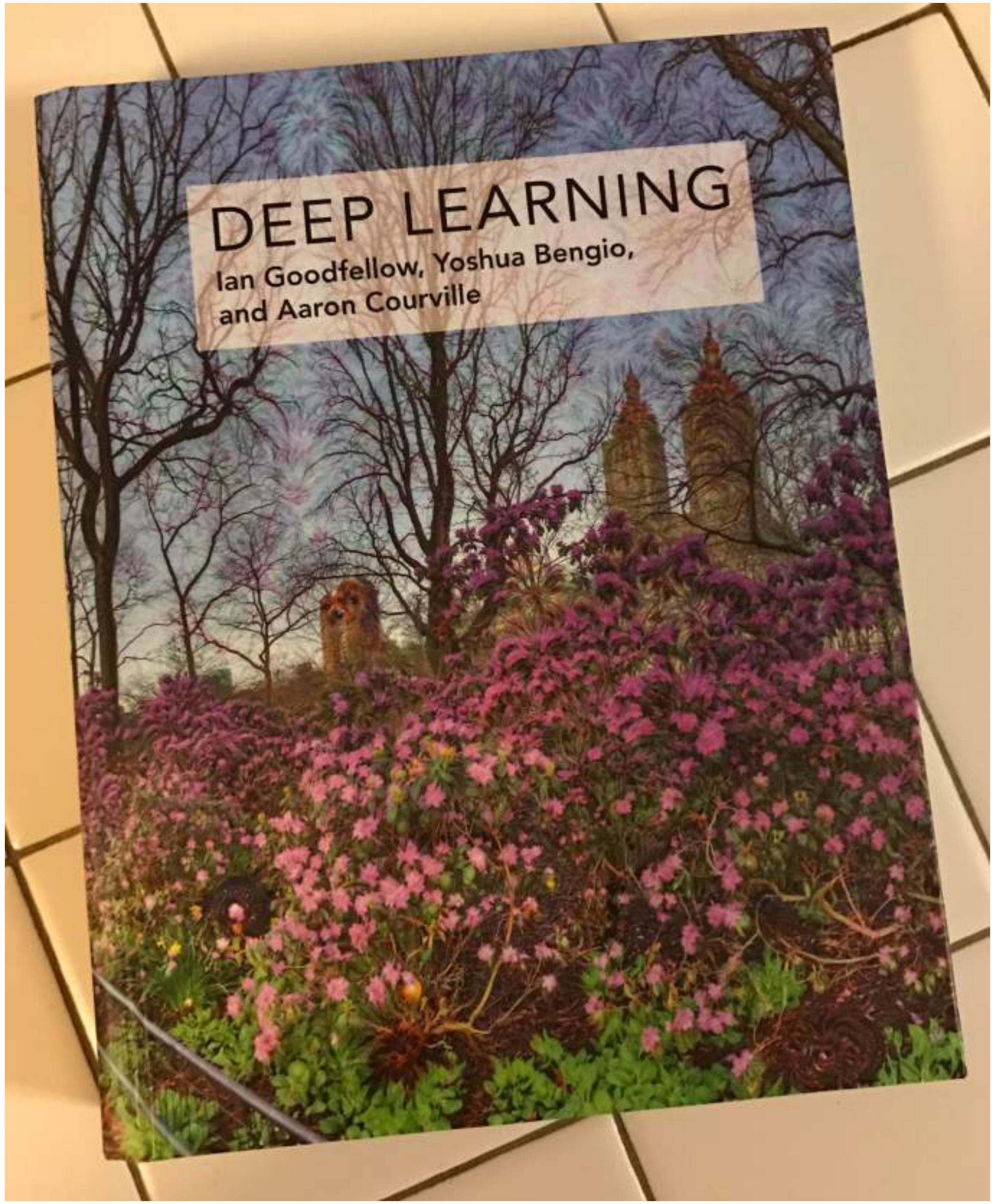
# Piazza Etiquette

- Post only short snippets of code (< 20-30 lines)
- Ask a specific, concrete question
- Explain what you have tried so far, and what happened
- See StackOverflow guide on asking good questions: https://stackoverflow.com/help/how-to-ask

- Don't expect and answer within 30 minutes of posting
- Monday – Friday, 10am – 6pm EST we'll try to answer within 2 hours
- Other times, we'll try to answer within 12 hours

# Optional Textbook

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



# Course Content and Grading

- 6 programming assignments (10% each)
- Midterm Exam (20%)
- Mini-Project (20%)
- Late policy
  - 3 free late days to use on assignments
  - Once free late days are exhausted, 25% penalty per day

# Programming Assignments

- Python, PyTorch, will use Google Colab
- “Earn your wings” – implement things “from scratch” in early assignments, then use PyTorch in later assignments
- ”Challenge Questions”
  - Go above and beyond the basic expectations of each assignment
  - Much higher time/points ratio than other parts of the assignment
  - Not necessary to get an A: will be 5% or less of each assignment

# Midterm Exam

- Written exam testing basic concepts from first half of course
- True / False, Multiple choice, short answer
- We will provide a practice exam to get a sense of the format of the questions (but not necessarily the length or difficulty)

# Mini-Project

- New this year!
- Work in teams of 1 to 3
- Homework assignments: We provide starter code, you “fill in the blank”
- Mini-Project: We provide high-level written description of an algorithm to implement, you do the whole thing “from scratch”
- We will give ~3 project descriptions, you pick one of those
- Deliverable: zip of code, and Colab notebook that walks through your implementation and main results (should be like notebooks from homework!)

# Collaboration Policy

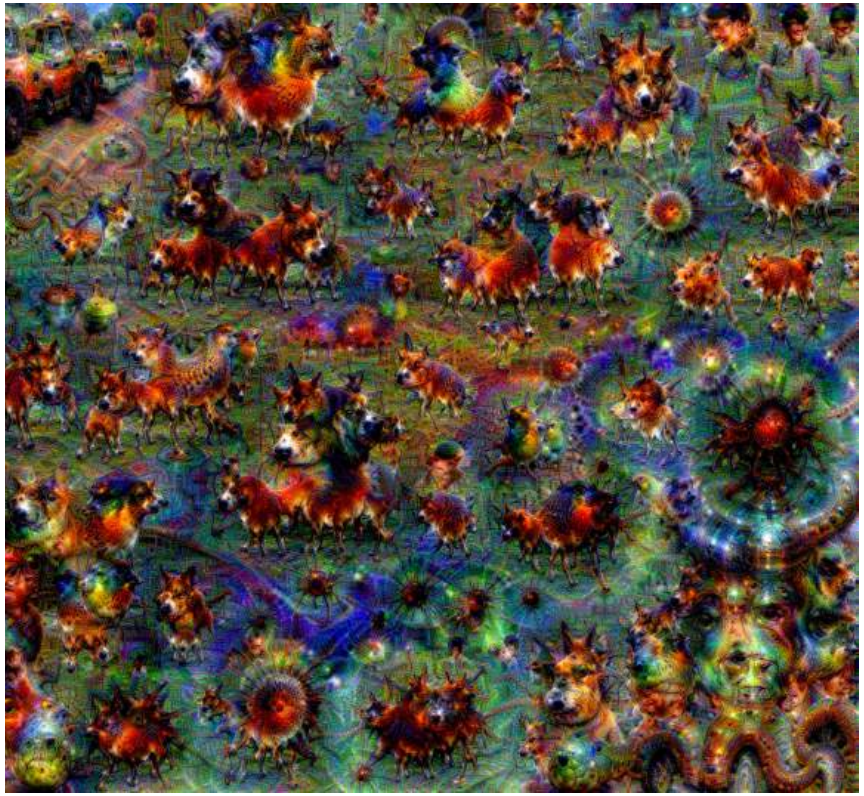
- **Rule 1**: Don't look at solutions or code that are not your own; everything you submit should be your own work
- **Rule 2**: Don't share your solution code with others; however discussing ideas or general strategies is fine and encouraged
- **Rule 3**: Indicate in your submissions anyone you worked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

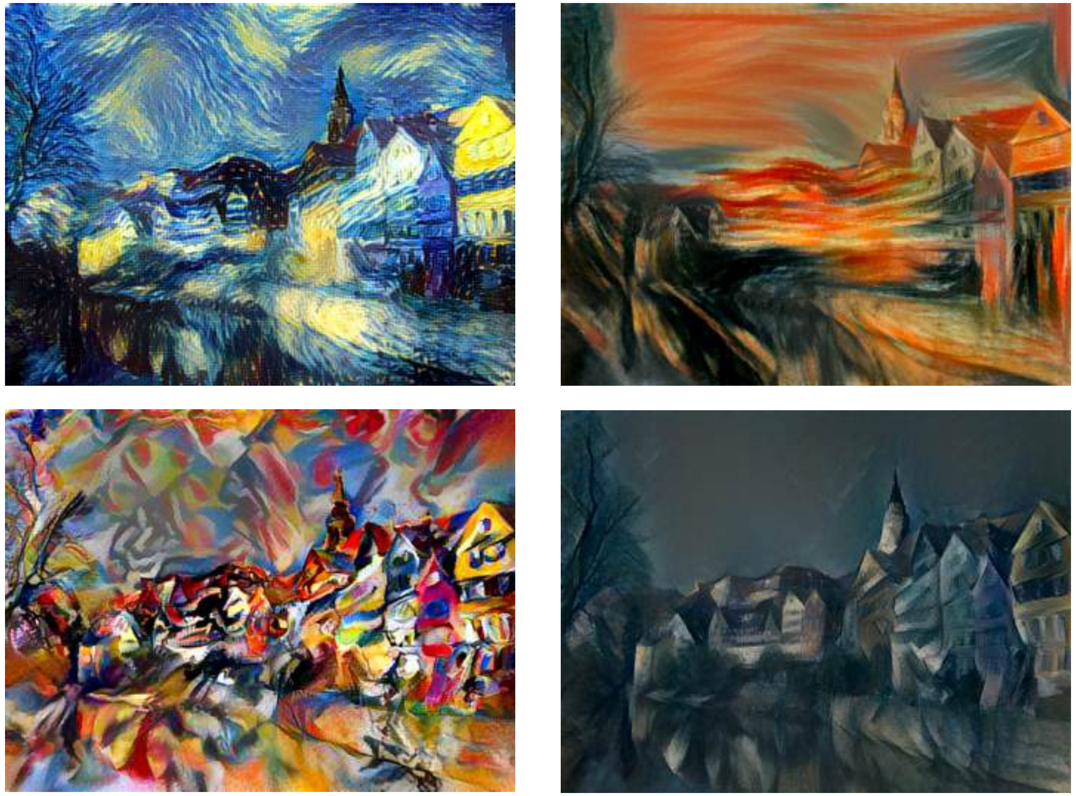
# Course Philosophy

- 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
  - 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

# New Topics since FA2020

- Modern CNN architectures
  - SENets, MobileNets, NAS
  - EfficientNets, RegNets, NFNets
- Vision Transformers
  - Architectures: ViT, DeiT, Swin, MViT
  - Applications: DETR
  - MLP-like architectures
- Vision + Language
  - Language-based pretraining: CLIP, ALIGN
- Self-Supervised Learning
  - Contrastive learning
  - Masked autoencoding

# First homework assignment

- Will be released by today or tomorrow
- Due Friday 1/14/2022
- Next lecture will be enough to complete it

# Next time: Image Classification