Deep Learning

Lecture 1 - Introduction

Who is Ranjay?

Ranjay Krishna (Assistant Professor at UW CSE)

- PhD from Stanford
- I worked with Fei-Fei Li (**AI**)
- And with Michael Bernstein (**HCI**)

I conduct two types of **research inquiries**:

- I study emergent **human behaviors** when they interact with AI systems
- I develop better **AI** (specifically **computer vision**) systems with these insights

Past courses:

- UW CSE 493G1 [2023]: Deep learning
- UW CSE 599H [2023]: Artificial intelligence vs intelligence augmentation
- Stanford CS 231N [2020, 2021]: Convolutional neural networks for computer vision
- Stanford CS 131 [2017, 2018, 2019]: Computer vision fundamentals and applications

Who is Sarah?

Sarah Pratt (Ph.D. candidate at UW CSE)

- Undergrad at Brown
- Currently working with Ali Farhadi in RAIVN lab at UW

Research

- My research examines the intersection of **vision** and **language** for deep learning systems.

Past courses:

- UW CSE 493G1 [2023]: Deep learning (TA)
- Brown 0220: [2017, 2018] Discrete Structures and Probability (TA)
- Brown 0150 [2016]: Introduction to Object Oriented programming (TA)

Are you in the right place?

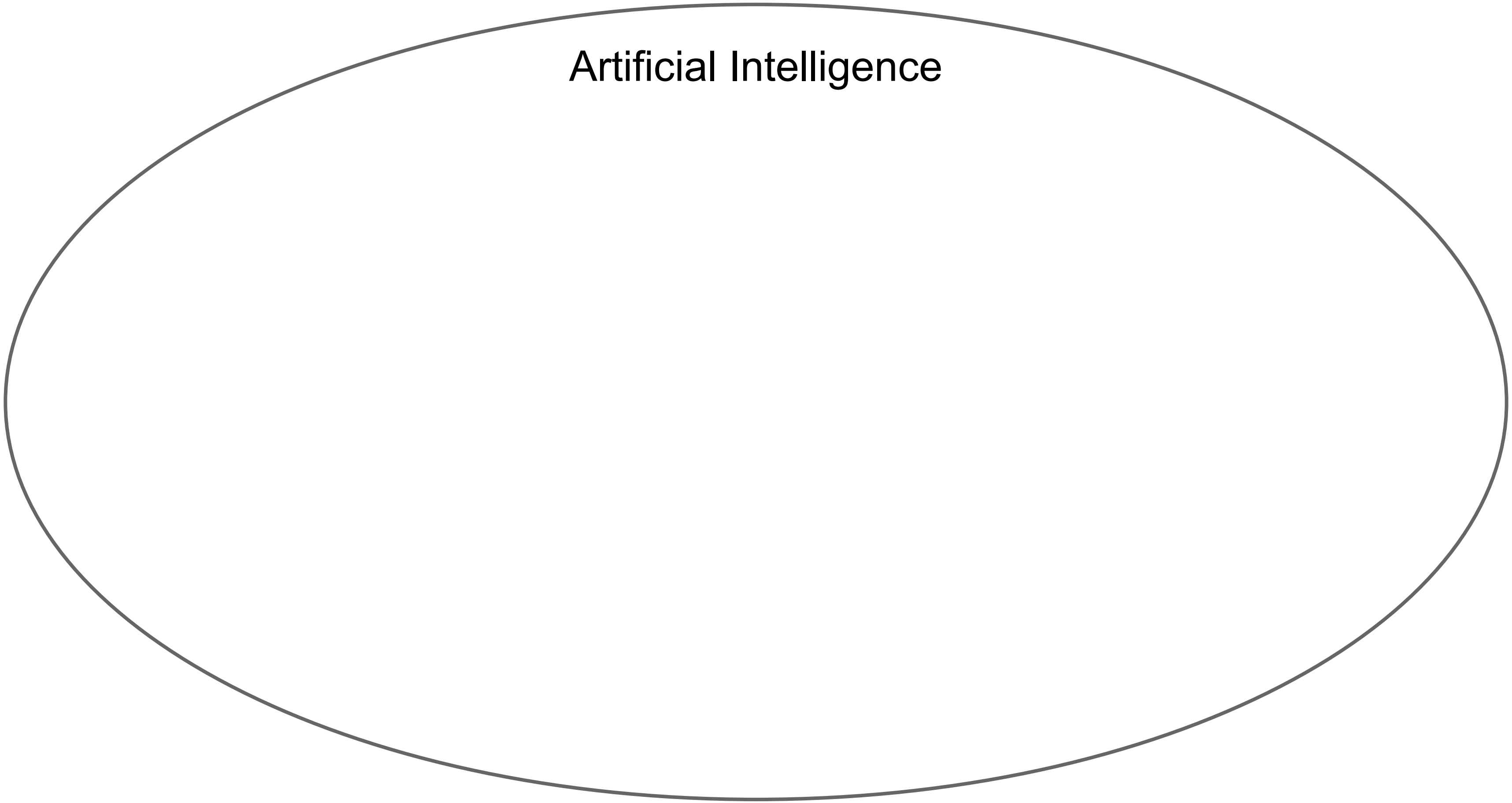
Location: SIG 134
Lectures: Tuesdays and Thursdays @ 10-11:20am
Recitations: Fridays
Canvas: https://canvas.uw.edu/courses/1694426
Gradescope: https://www.gradescope.com/courses/687869
Website: https://courses.cs.washington.edu/courses/cse493g1/24wi/
EdStem: https://edstem.org/us/courses/50490

What is ~~Deep~~ Learning?

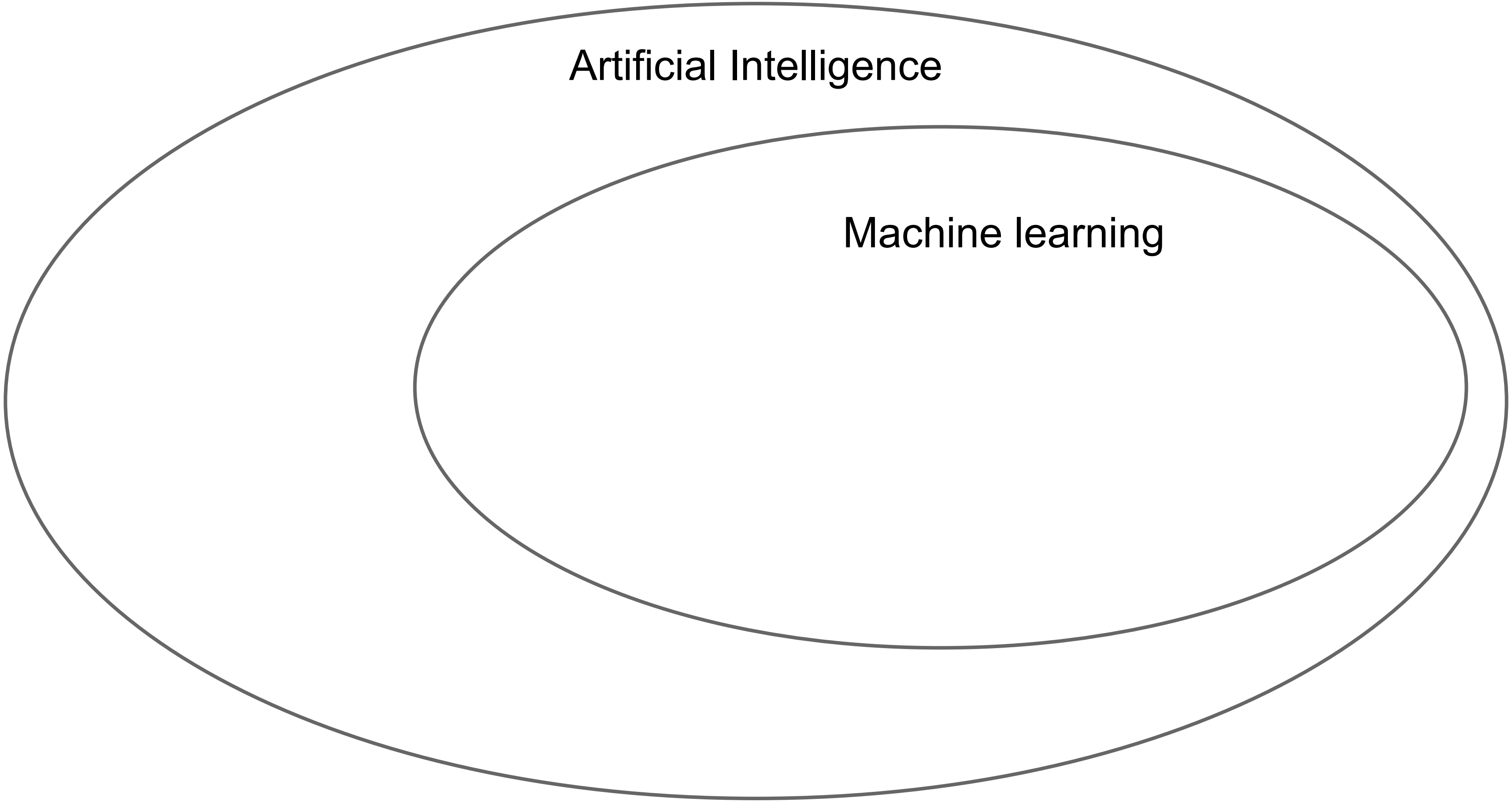
Building artificial systems that learn from data and experience

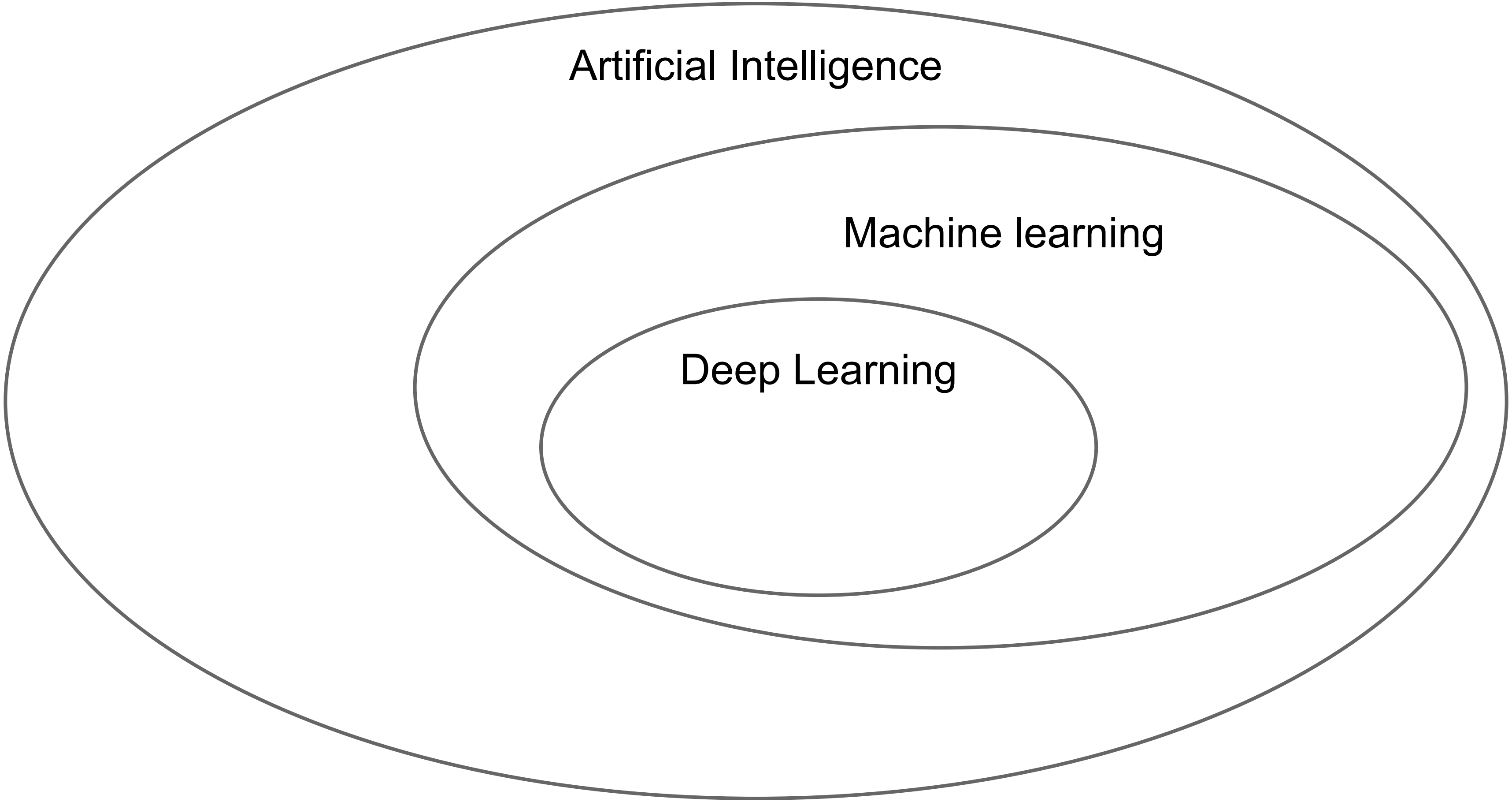
What is Deep Learning?

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

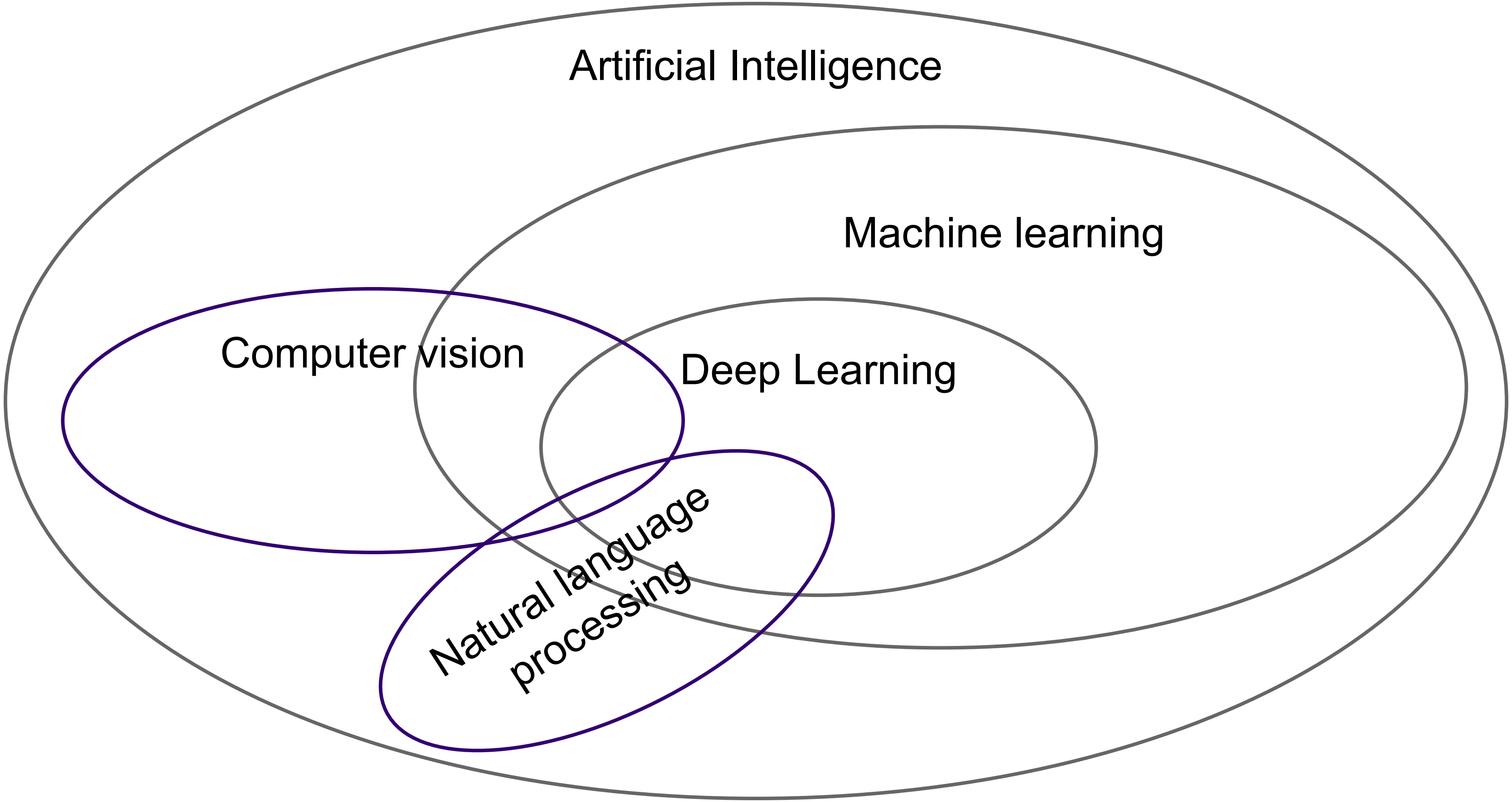


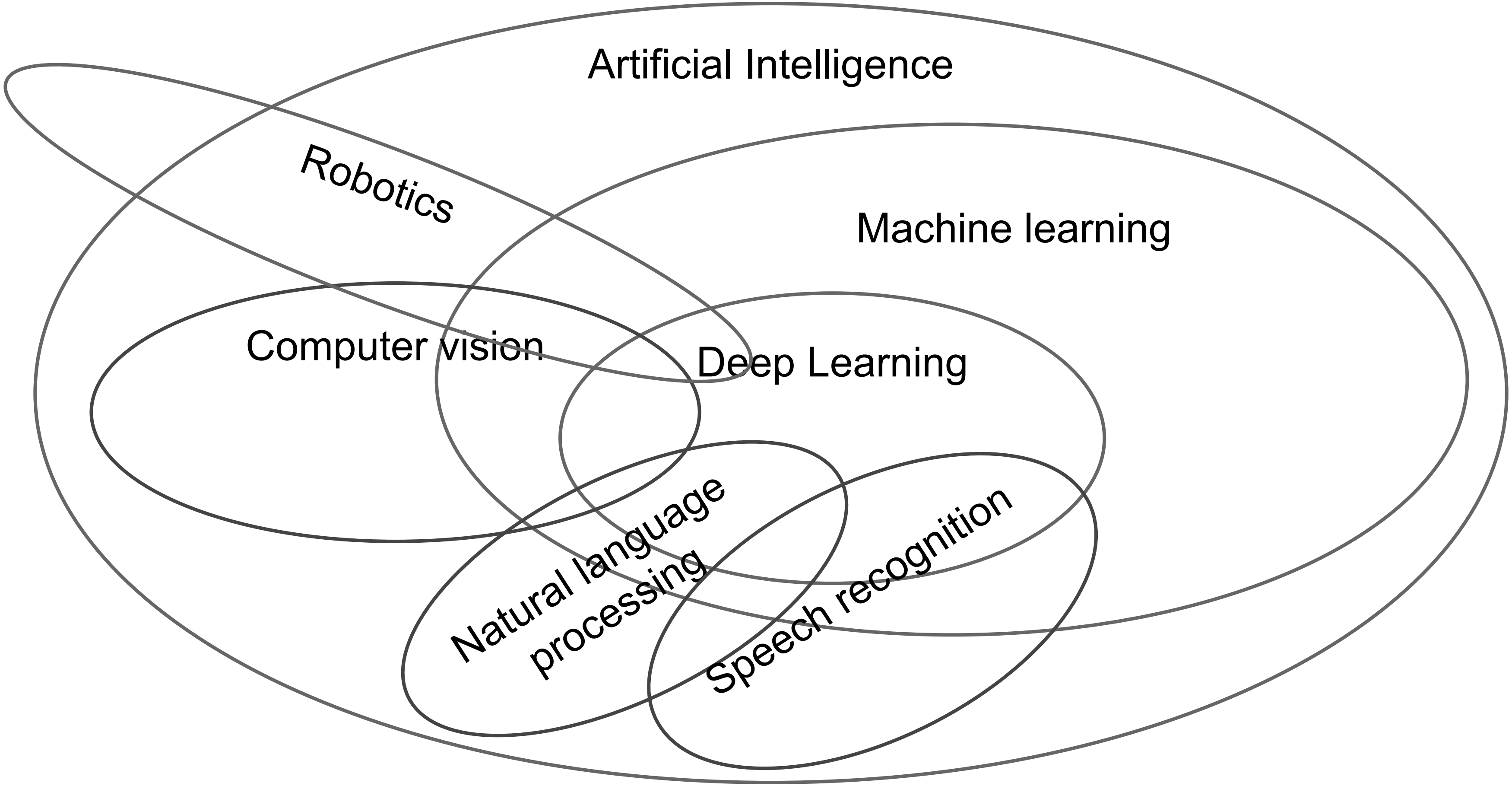
Artificial Intelligence

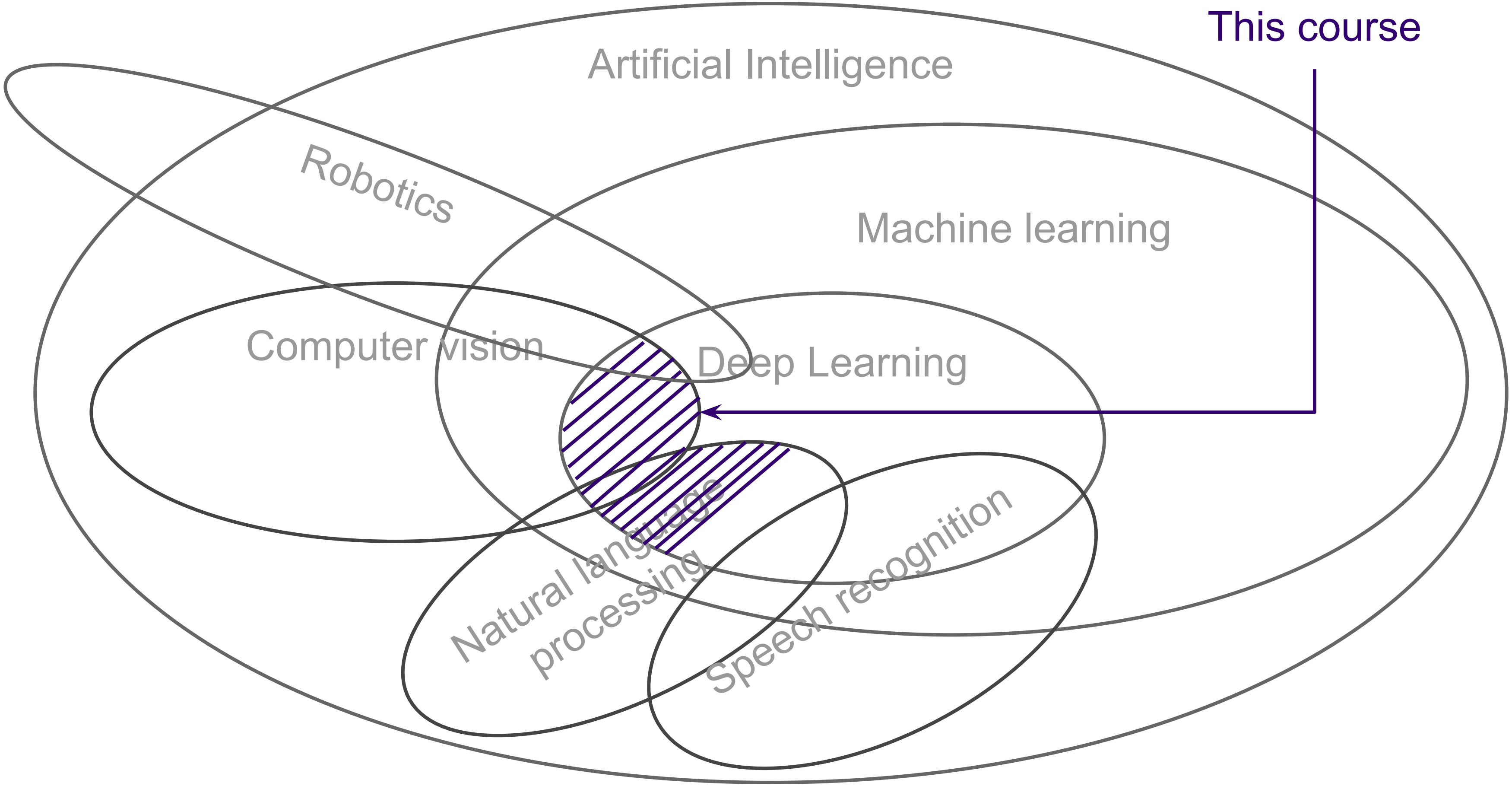

Artificial Intelligence
Machine learning

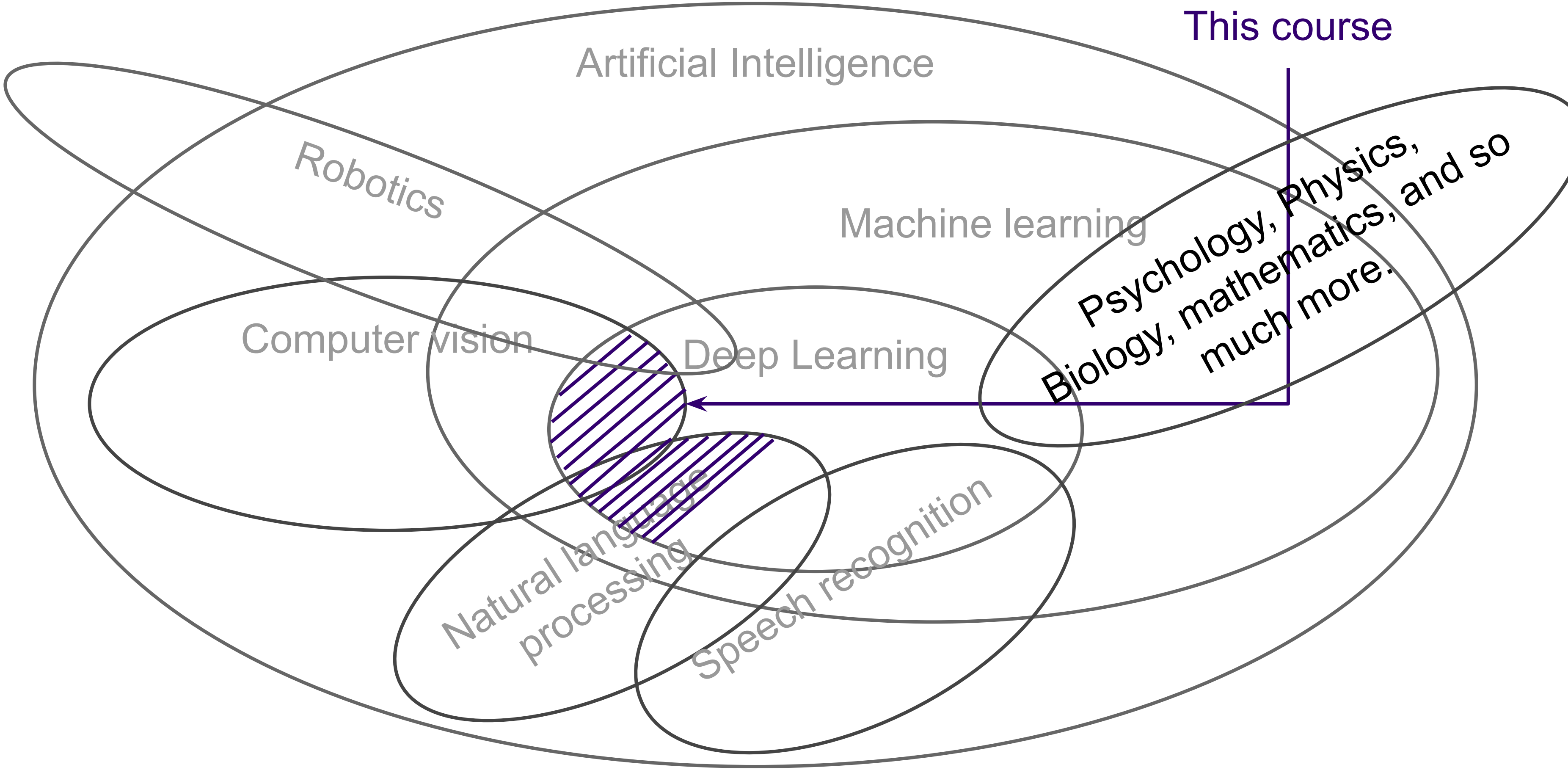


Artificial Intelligence
Machine learning
Deep Learning


Artificial Intelligence
Machine learning
Computer vision
Deep Learning
Natural language processing


Artificial Intelligence
Robotics
Machine learning
Computer vision
Deep Learning
Natural language processing
Speech recognition


This course
Artificial Intelligence
Robotics
Machine learning
Computer vision
Deep Learning
Natural language processing
Speech recognition


This course
Artificial Intelligence
Robotics
Machine learning
Psychology, Physics, Biology, mathematics, and so much more.
Computer vision
Deep Learning
Natural language processing
Speech recognition

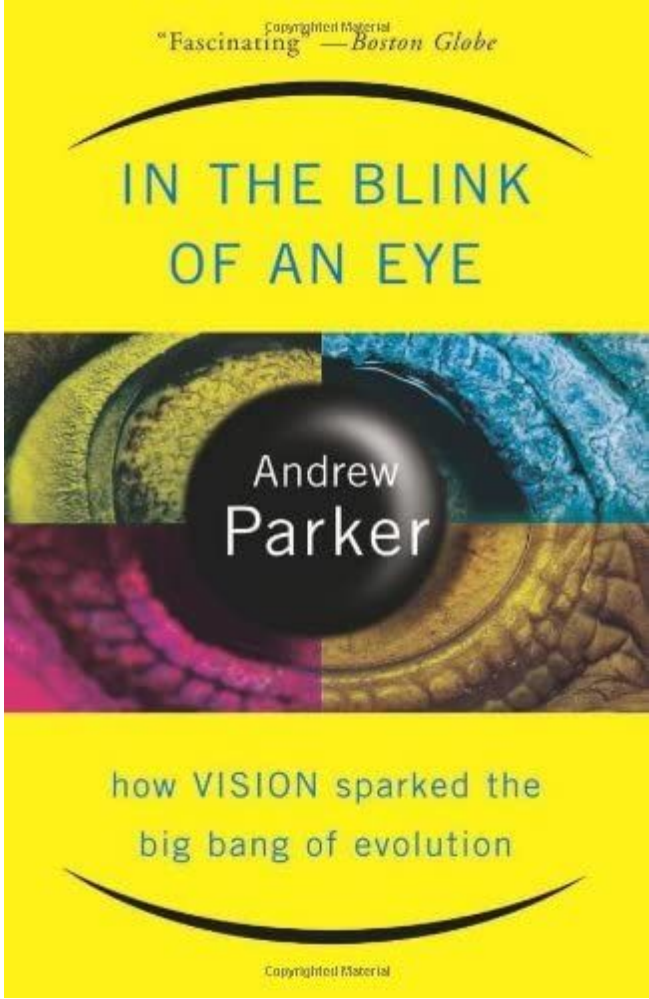
Today’s agenda

- A brief history of deep learning
- CSE 493G1 overview

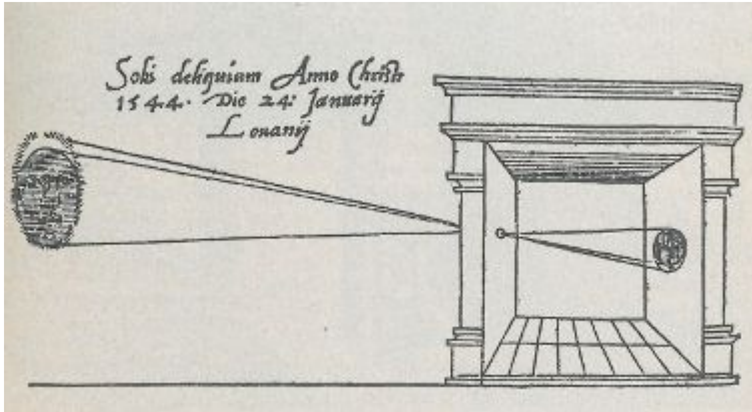
The European Camera of the Year 1983.
Fortunately, fine photography is a universal language.
PENTAX
PENTAX

Vision is core to the evolution of intelligence

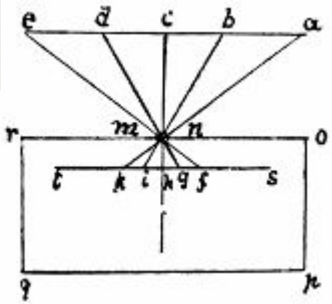
543 million years ago.


"Fascinating" —Boston Globe
IN THE BLINK OF AN EYE
Andrew Parker
how VISION sparked the big bang of evolution
Copyrighted Material

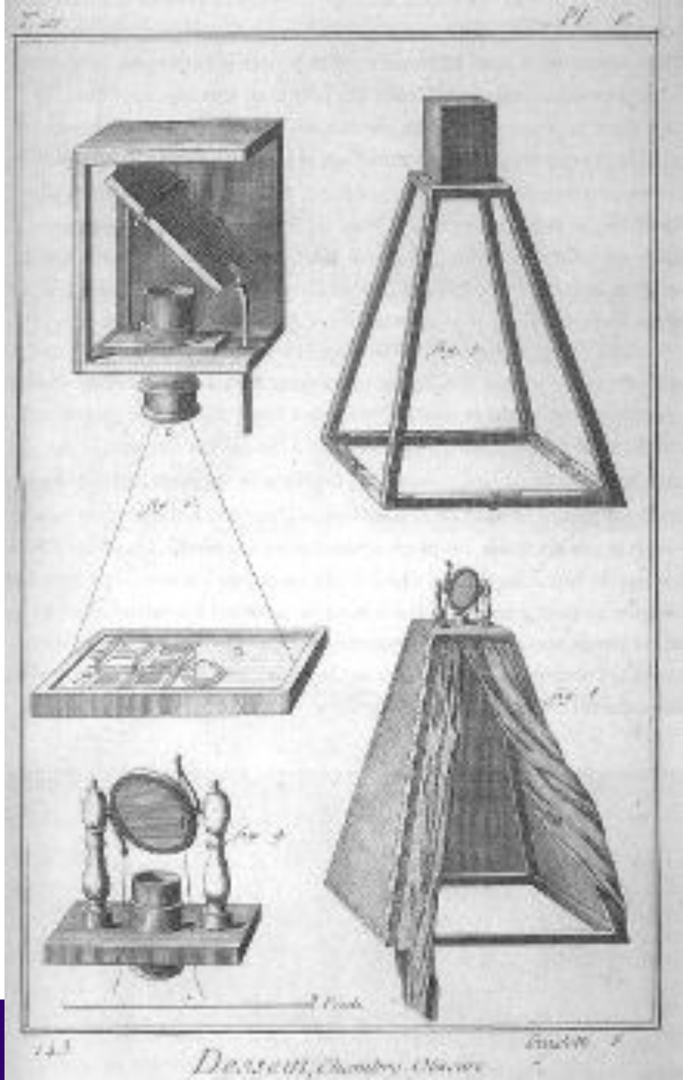
The first attempts at capturing the visual world



Camera obscura by Gemma Frisius, 1545



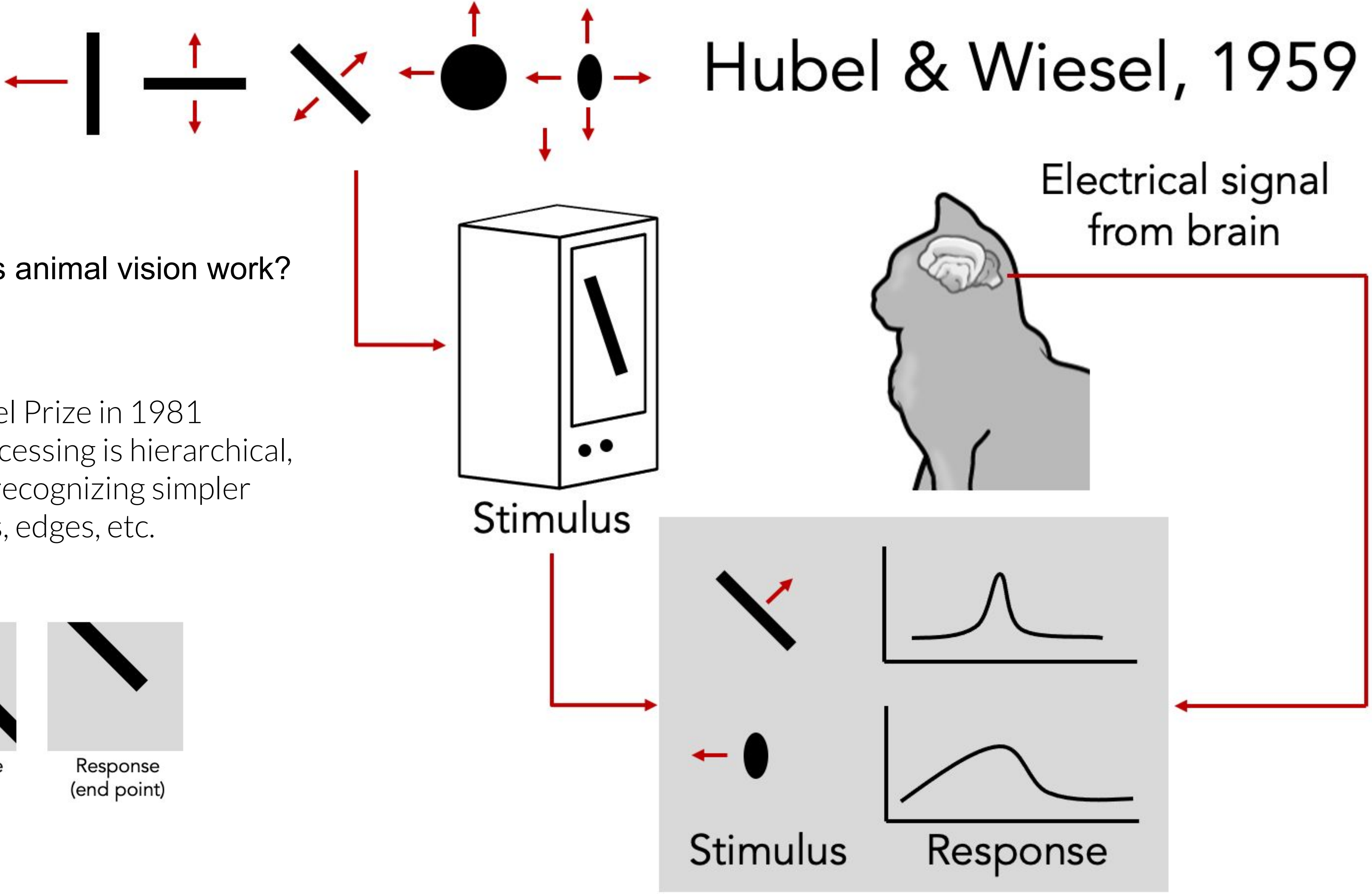
Inspired Leonardo da Vinci,
16th Century AD



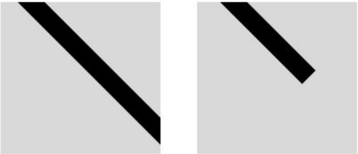
Examples from 18th century Encyclopedia

Hubel & Wiesel, 1959

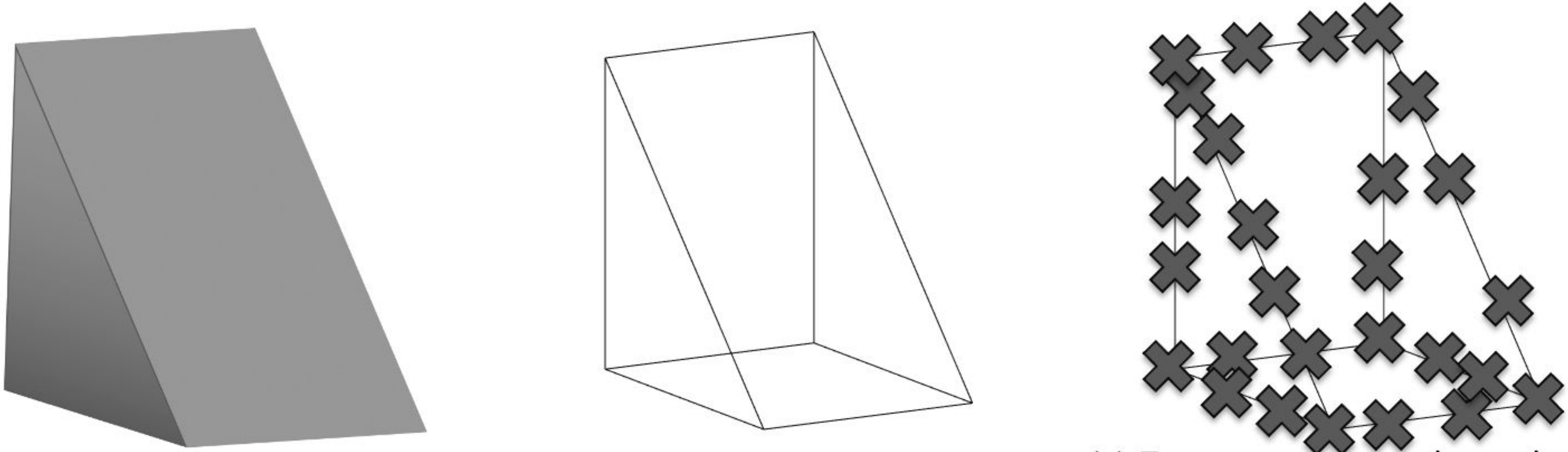
How does animal vision work?



Won Nobel Prize in 1981
Visual processing is hierarchical, involving recognizing simpler structures, edges, etc.



Larry Roberts - Father of computer vision



(a) Original picture

(b) Differentiated picture

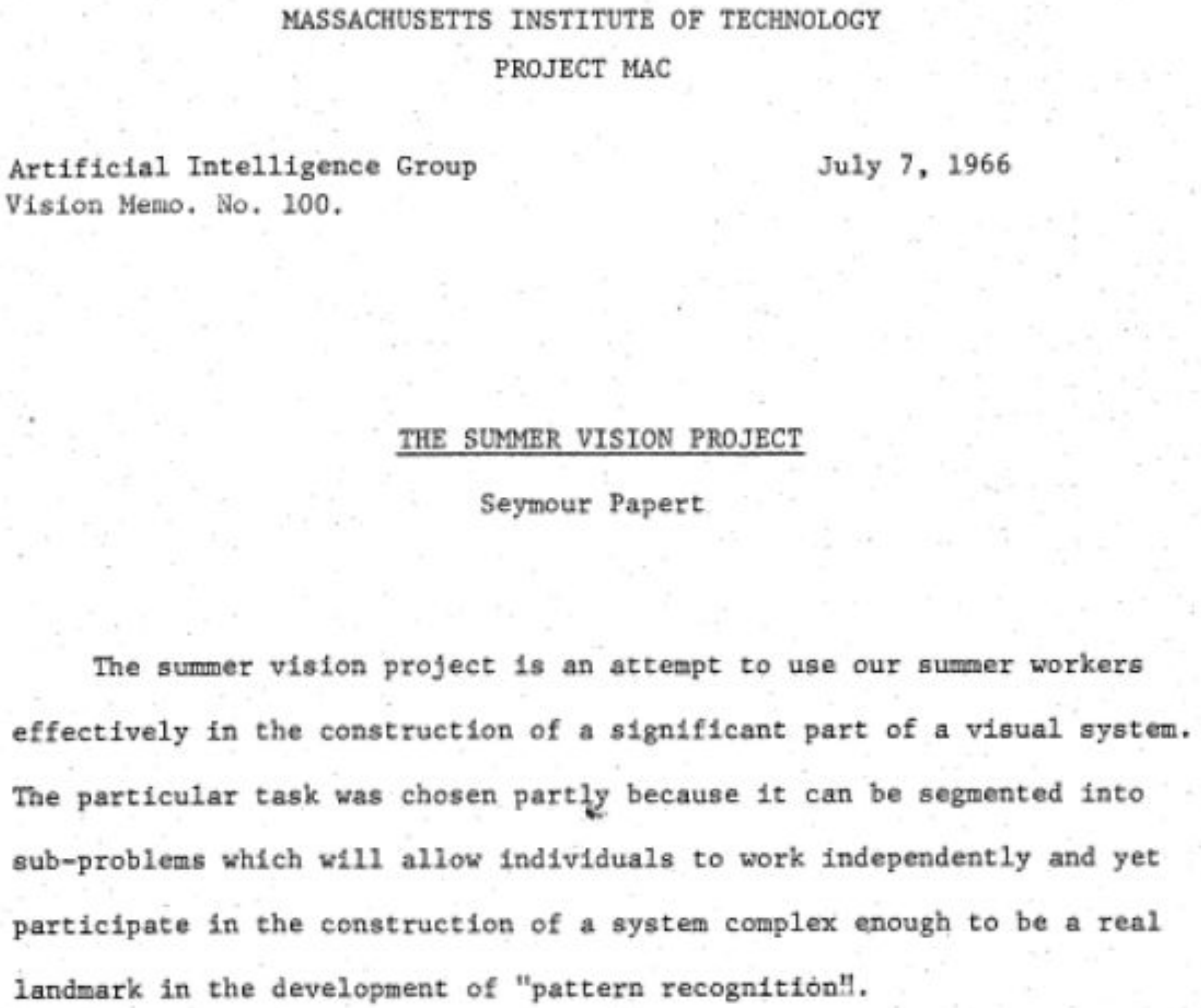
(c) Feature points selected

Synthetic images, building up the visual world from simpler structures

The summer vision project

Organized by Seymour Papert

Computer vision was meant to be just a simple summer intern project



MASSACHUSETTS INSTITUTE OF TECHNOLOGY

PROJECT MAC

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

THE SUMMER VISION PROJECT

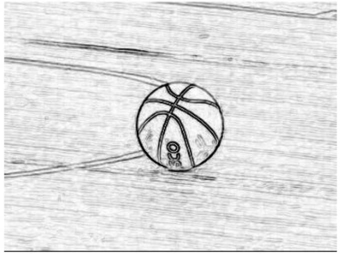
Seymour Papert

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".

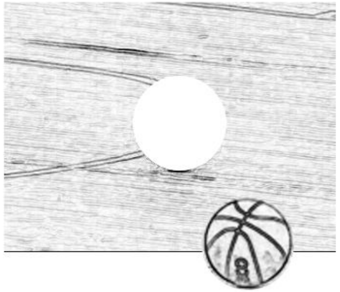
Input image

This image is CC0 1.0 public domain

Edge image

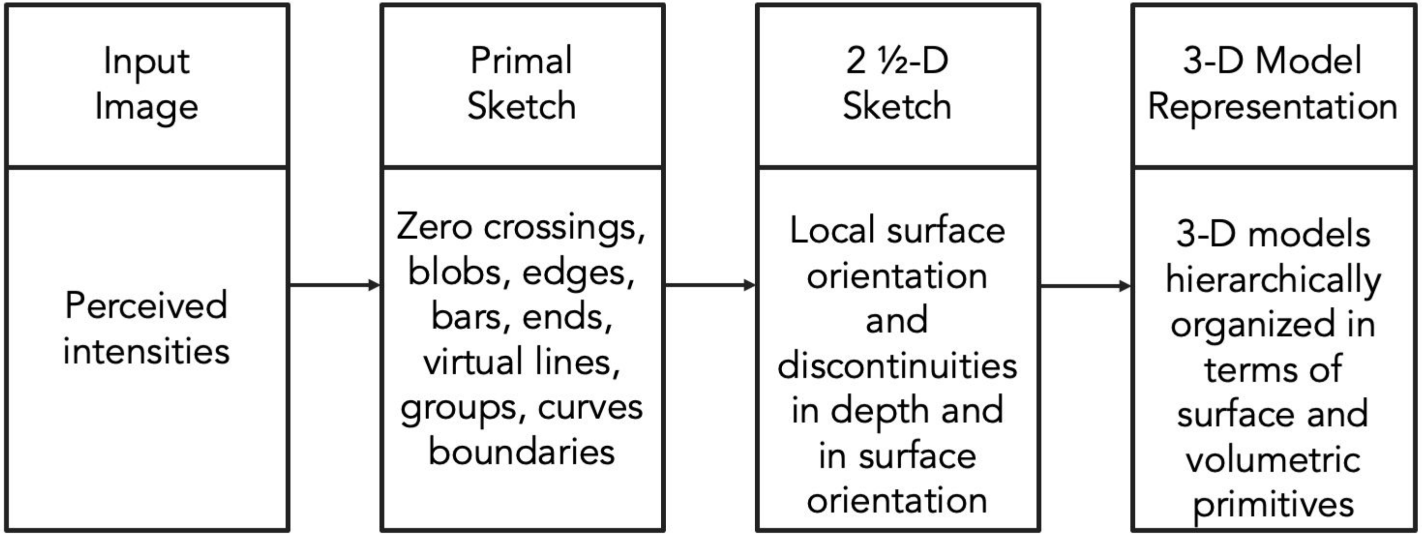


2 ½-D sketch



3-D model

This image is CC0 1.0 public domain



Input Image

Perceived intensities

Primal Sketch

Zero crossings, blobs, edges, bars, ends, virtual lines, groups, curves boundaries

2 ½-D Sketch

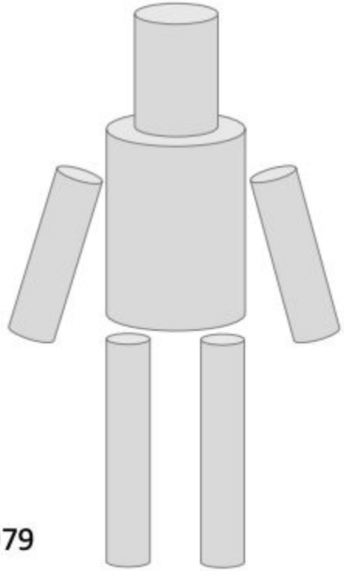
Local surface orientation and discontinuities in depth and in surface orientation

3-D Model Representation

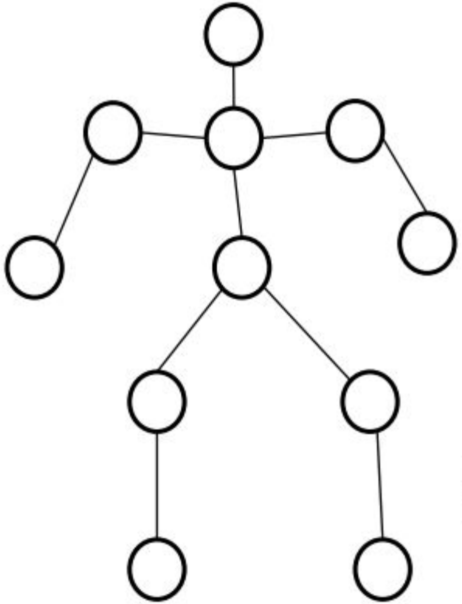
3-D models hierarchically organized in terms of surface and volumetric primitives

David Marr,
Stages of Visual Representation,
1970

Recognition via parts (1970s)

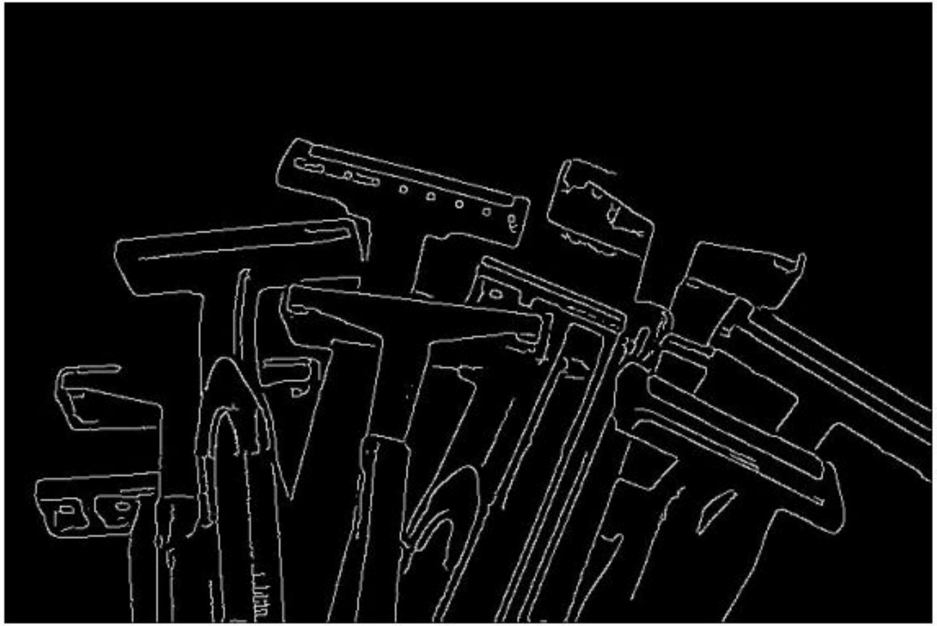


Generalized Cylinders,
Brooks and Binford, 1979



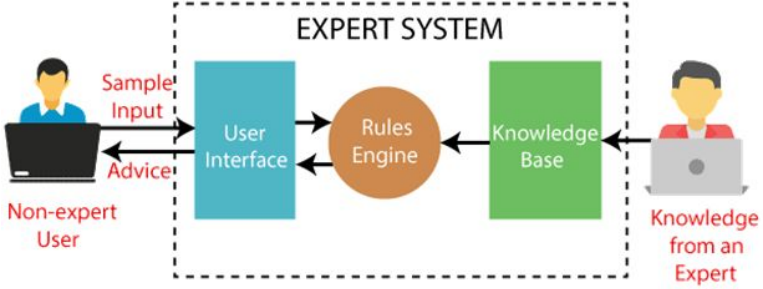
Pictorial Structures,
Fischler and Elshlager, 1973

Recognition via edge detection (1980s)

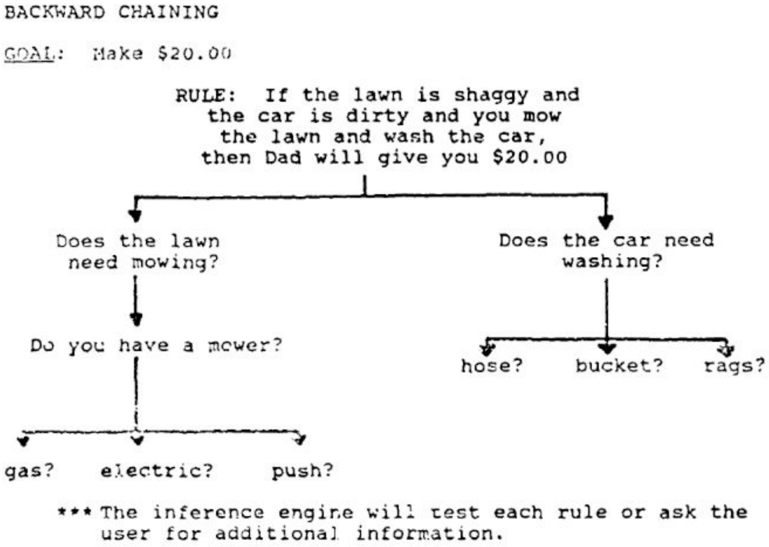


John Canny, 1986 David Lowe, 1987

1980s caused one of the larger AI winters (the second AI winter)



Originally called heuristic programming project.



RULE: If the lawn is shaggy and the car is dirty and you mow the lawn and wash the car, then Dad will give you $20.00

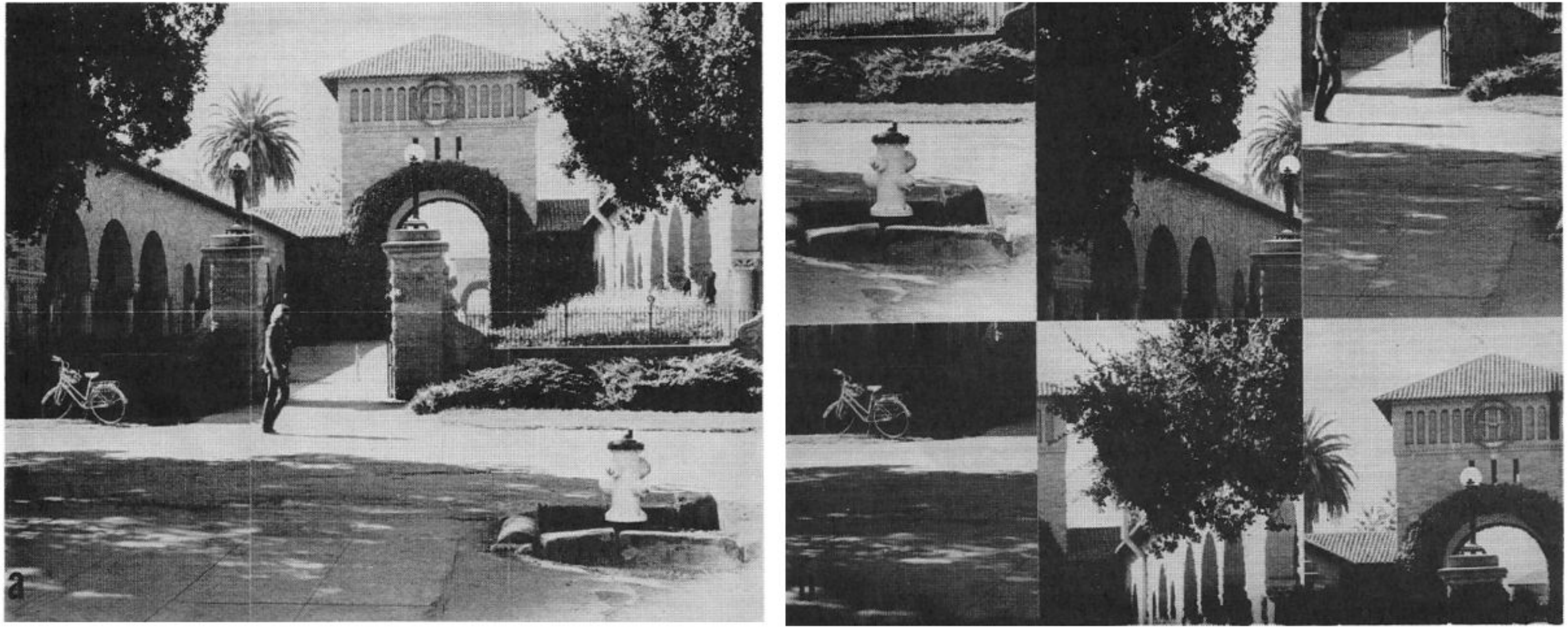
*** The inference engine will test each rule or ask the user for additional information.

- Enthusiasm (and funding!) for AI research dwindled
- “Expert Systems” failed to deliver on their promises
- But subfields of AI continued to grow
    - Computer vision, NLP, robotics, compbio, etc.

In the meantime…seminal work in cognitive and neuroscience

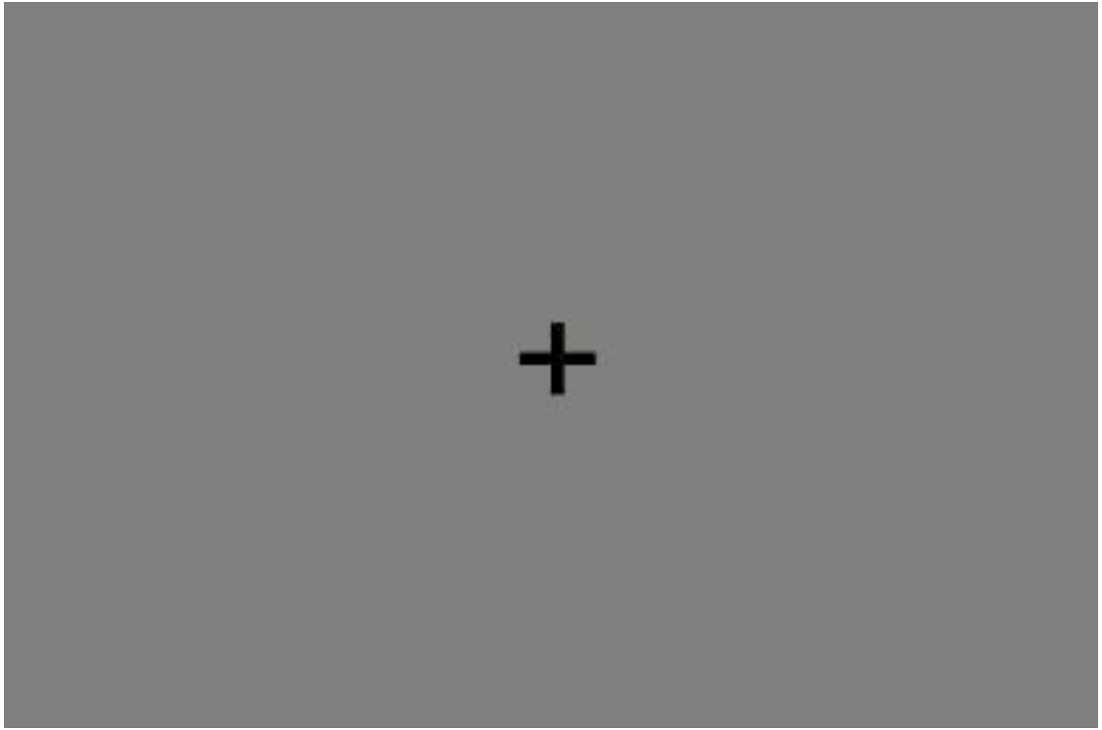
Perceiving real-world scenes

Irving Biederman



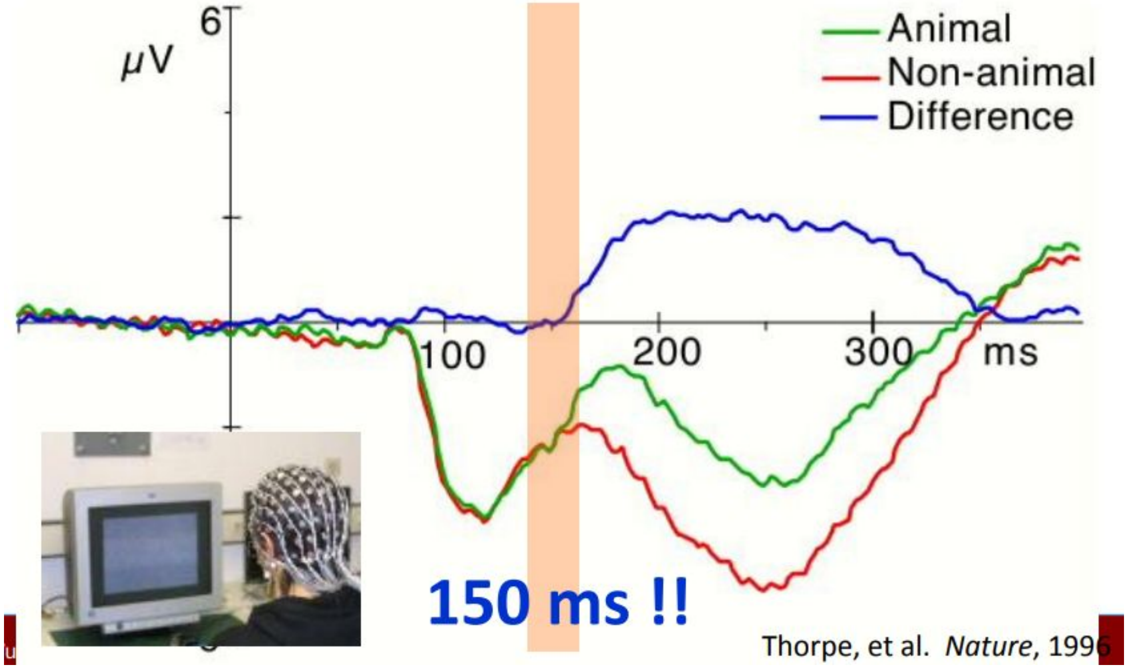
I. Biederman, *Science*, 1972

Rapid Serial Visual Perception (RSVP)



Potter, etc. 1970s

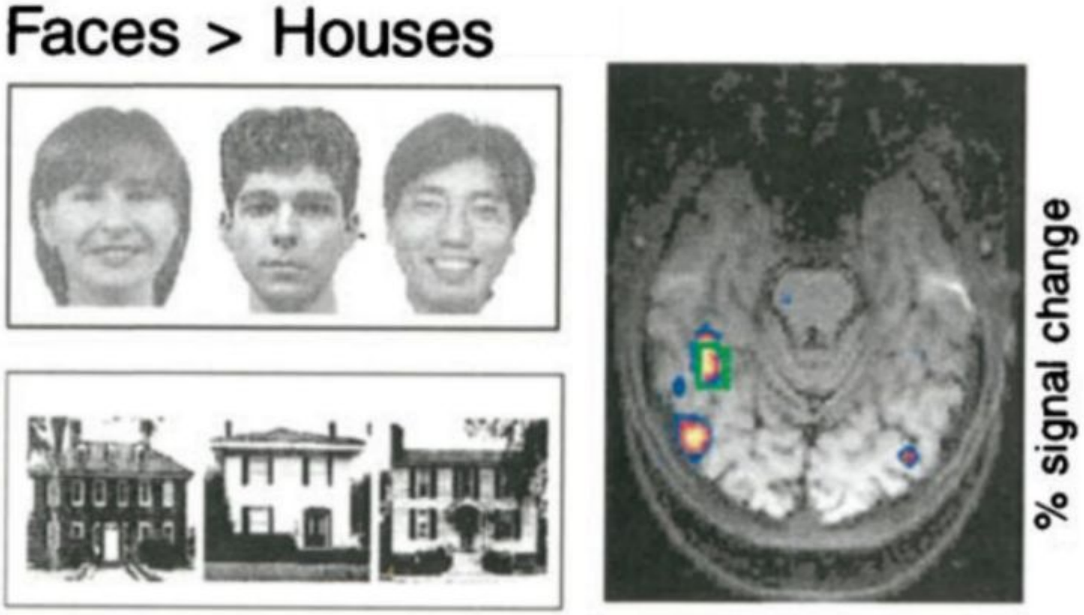
Speed of processing in the human visual system (Thorpe et al. Nature 1996)



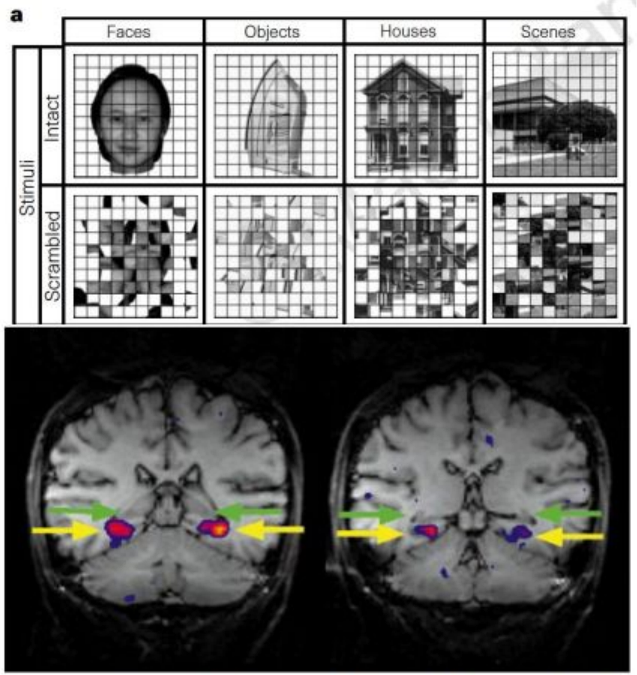
150 ms !!

Thorpe, et al. *Nature*, 1996

Neural correlates of object & scene recognition

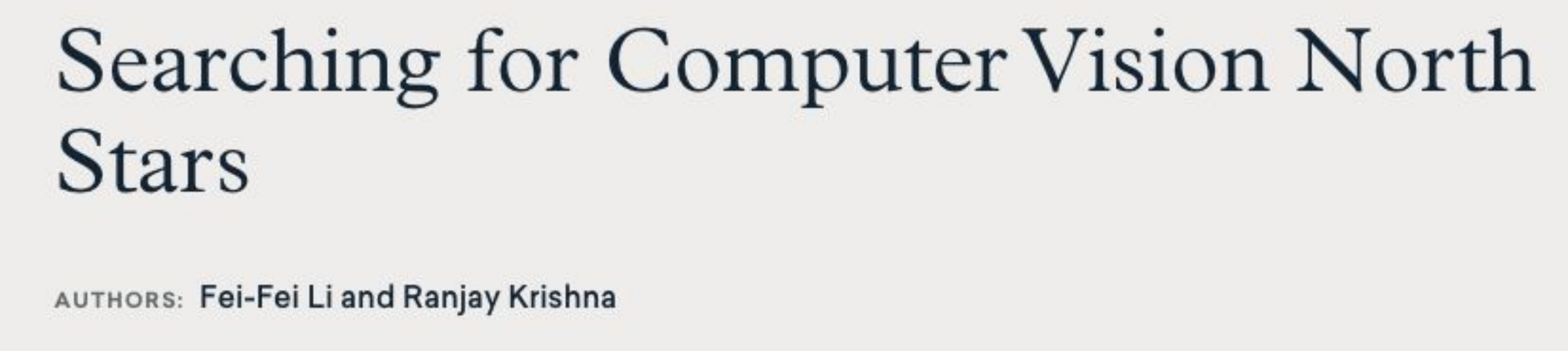


Kanwisher et al. J. Neuro. 1997



Epstein & Kanwisher, Nature, 1998

Visual recognition is a fundamental to intelligence

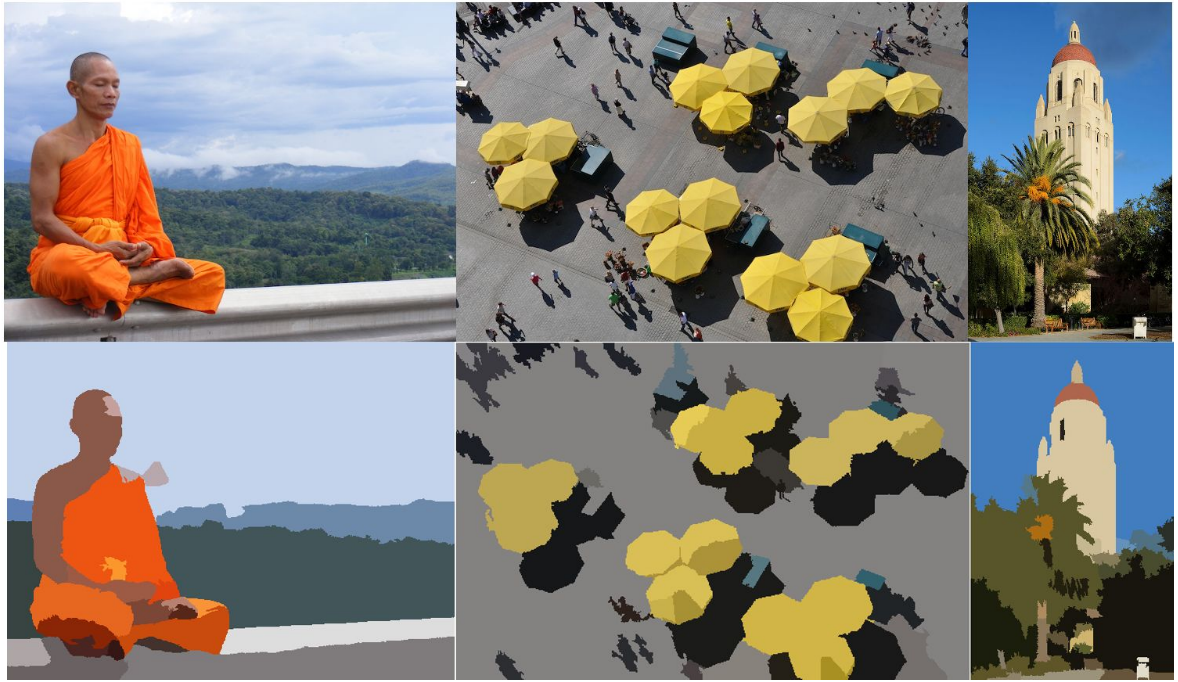


Searching for Computer Vision North Stars

AUTHORS: Fei-Fei Li and Ranjay Krishna

Until the 90s,
computer vision was not broadly applied to real world images

The focus was on algorithms! Recognition via Grouping (1990s)



Shi & Malik,
Normalized Cut, 1997

Recognition via Matching (2000s)

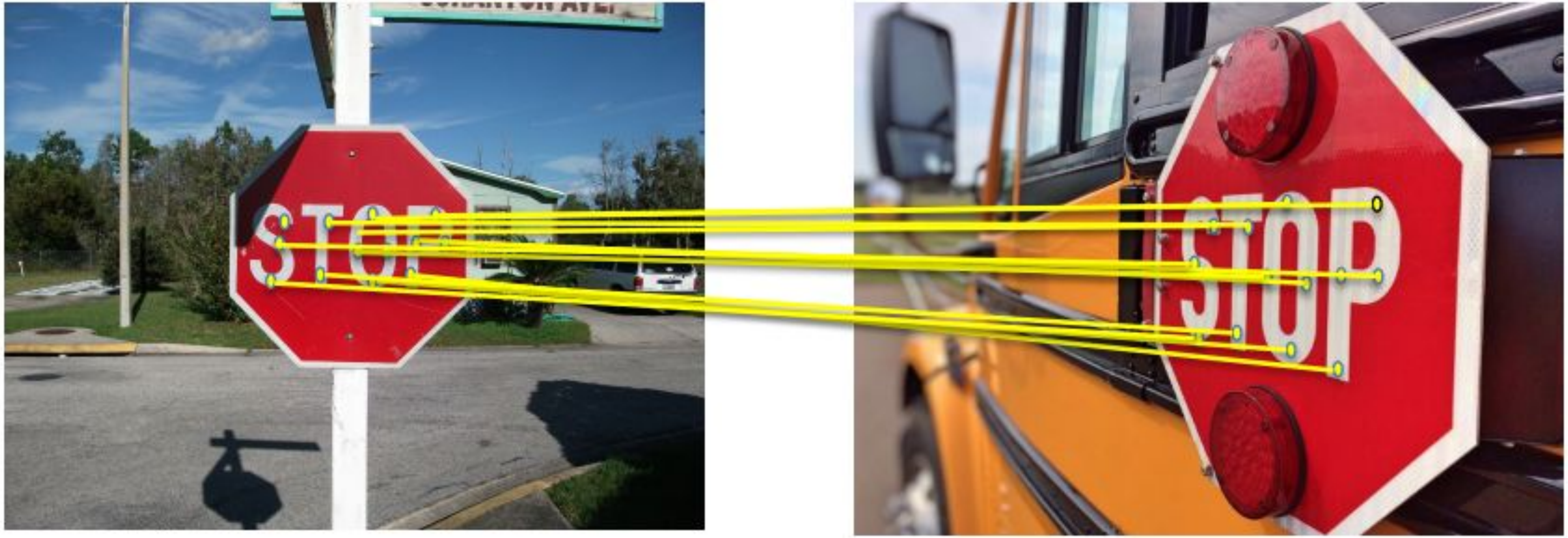


Image is public domain

Image is public domain

SIFT, David Lowe, 1999

First commercial success of computer vision

It came from embracing machine learning in 2001.

Does anyone know what it was?

First commercial success of computer vision

Real time face detection using using an algorithm by Viola and Jones, 2001

- Fujifilm face detection in cameras
- HP patent immediately

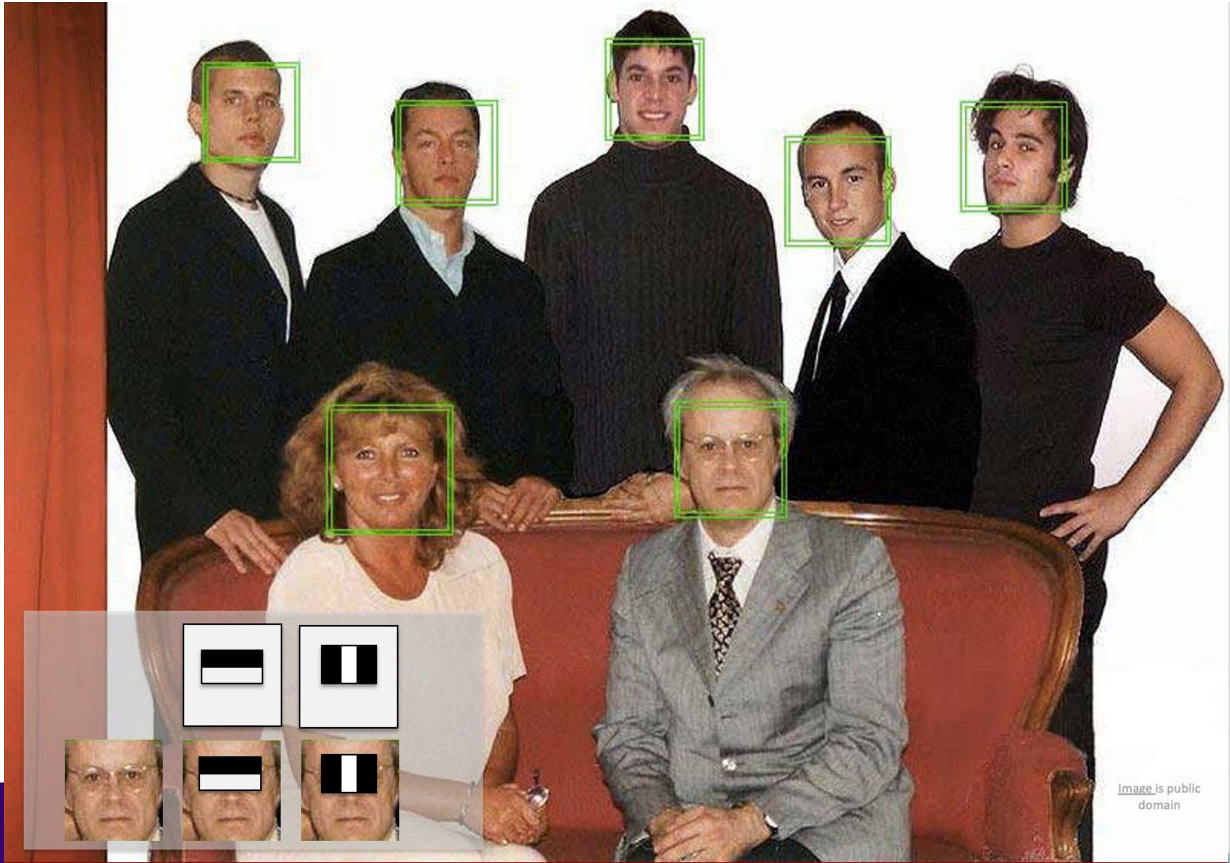


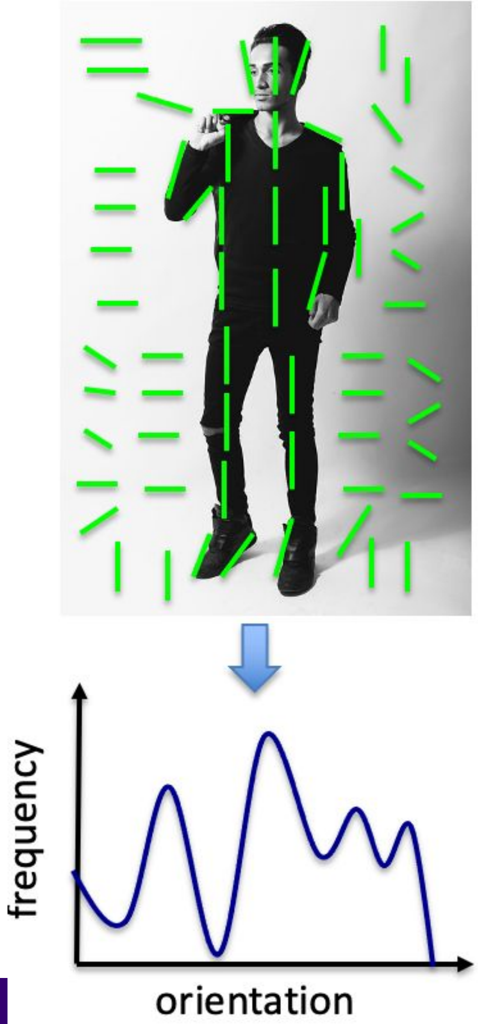
Image is public domain

Designing better feature extraction became the focus

HoG features

- Histogram of oriented gradients
- Handcrafted

[Dalal & Triggs, HoG. 2005]



Caltech 101 images

PASCAL Visual Object Challenge

Image is CC0 1.0 public domain

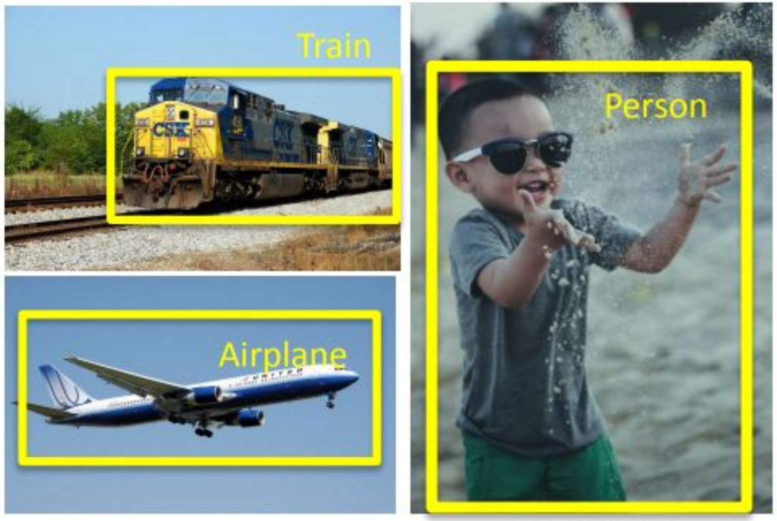


Image is CC0 1.0 public domain

IMAGENET

www.image-net.org

22K categories and 14M images

- Animals
  - Bird
  - Fish
  - Mammal
  - Invertebrate
- Plants
  - Tree
  - Flower
- Food
- Materials
- Structures
- Artifact
  - Tools
  - Appliances
  - Structures
- Person
- Scenes
  - Indoor
  - Geological Formations
- Sport Activities

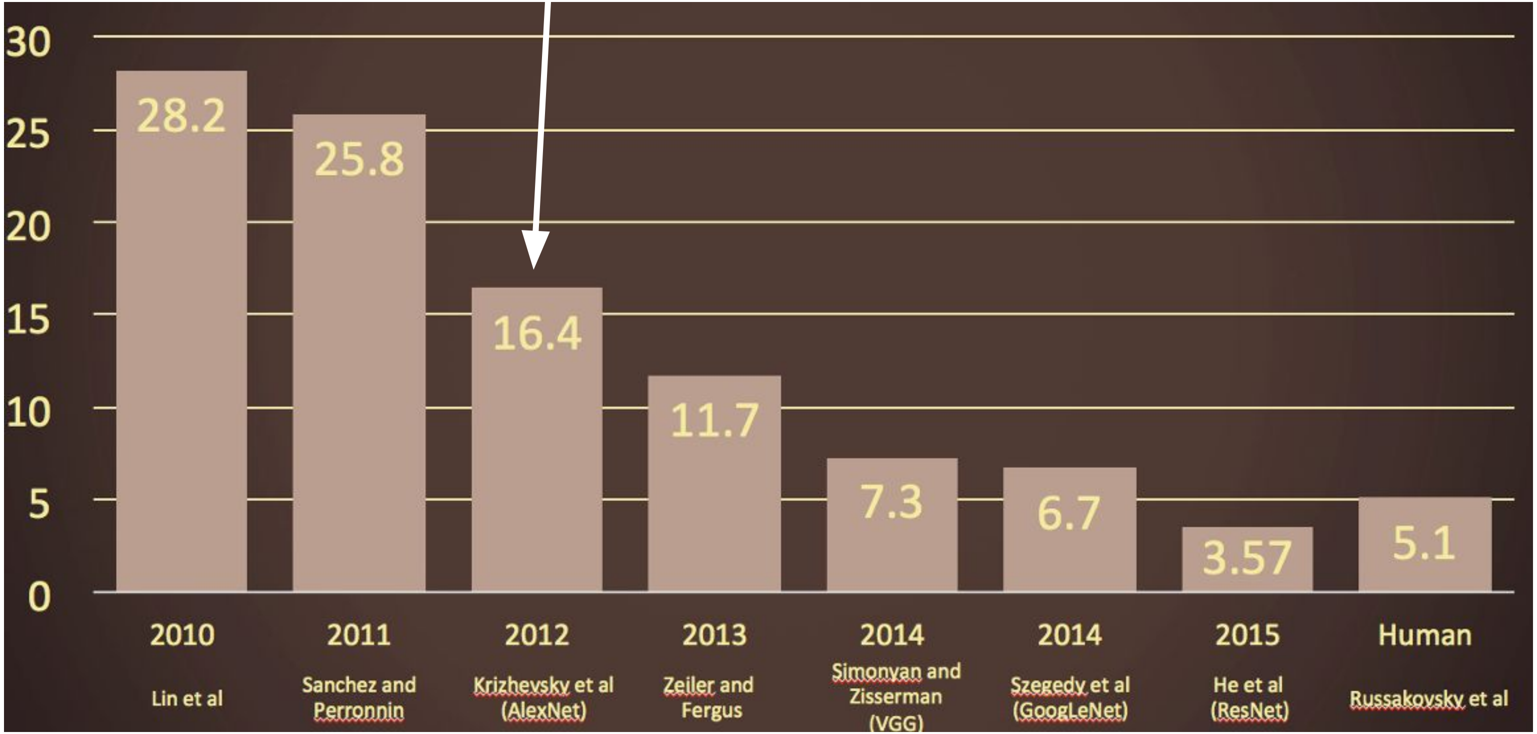
Deng, Dong, Socher, Li, Li, & Fei-Fei, 2009

Hypothesis behind ImageNet

- A child sees nearly 3K unique objects by the age of 6
- Calculated by Irving Biederman
    - [Biederman. Recognition-by-components: a theory of human image understanding. 1983]

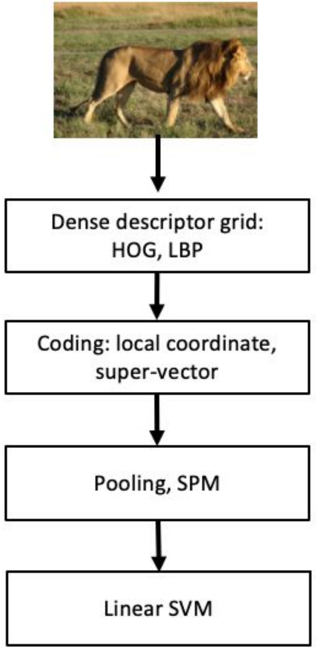
- But computer vision algorithms are trained on a handful of objects.

Object recognition accuracy drops by half in 2012 (Enter **deep learning**)



Year 2010

NEC-UIUC



Dense descriptor grid: HOG, LBP

Coding: local coordinate, super-vector

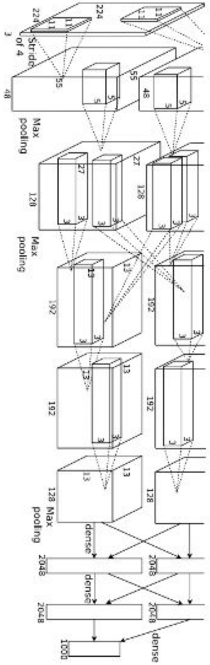
Pooling, SPM

Linear SVM

[Lin CVPR 2011]

Year 2012

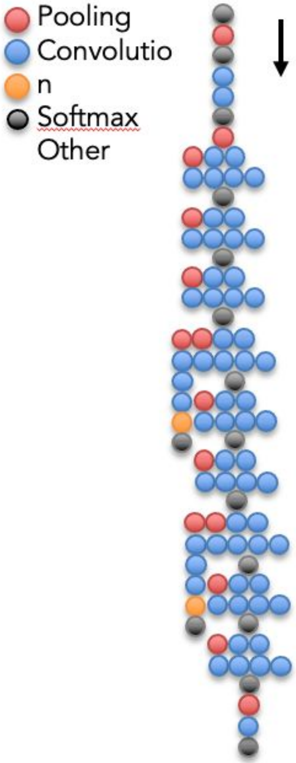
SuperVision



[Krizhevsky NIPS 2012]

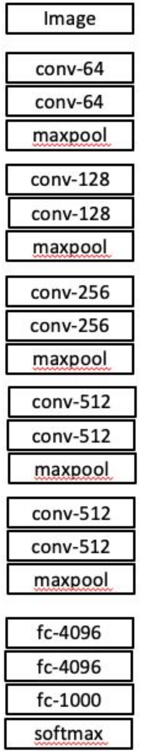
Year 2014

GoogLeNet



[Szegedy arxiv 2014]

VGG



Image

conv-64

conv-64

maxpool

conv-128

conv-128

maxpool

conv-256

conv-256

maxpool

conv-512

conv-512

maxpool

conv-512

conv-512

maxpool

fc-4096

fc-4096

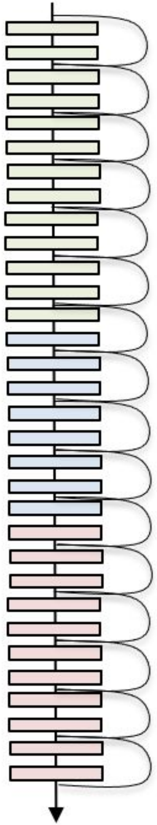
fc-1000

softmax

[Simonyan arxiv 2014]

Year 2015

MSRA



[He ICCV 2015]

AlexNet goes mainstream across computer vision

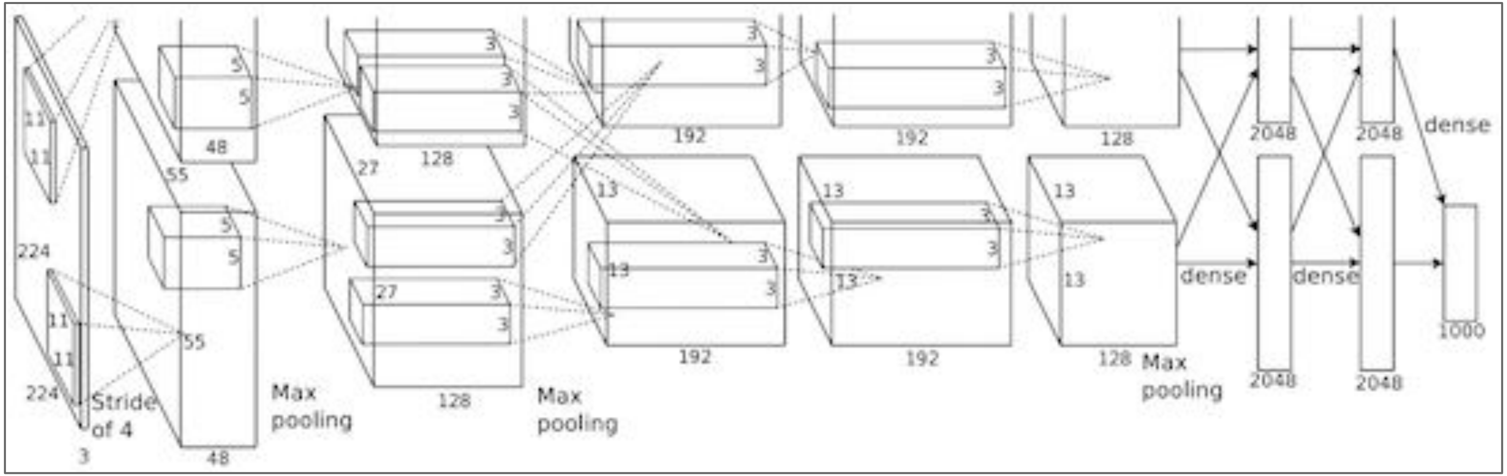


Figure copyright Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton, 2012. Reproduced with permission.

“AlexNet”

Core ideas go back many decades!

The **Mark I Perceptron** machine was the first implementation of the perceptron algorithm.

The machine was connected to a camera that used 20×20 cadmium sulfide photocells to produce a 400-pixel image.

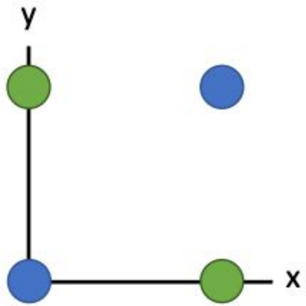
recognized
letters of the alphabet

Frank Rosenblatt, ~1957: Perceptron

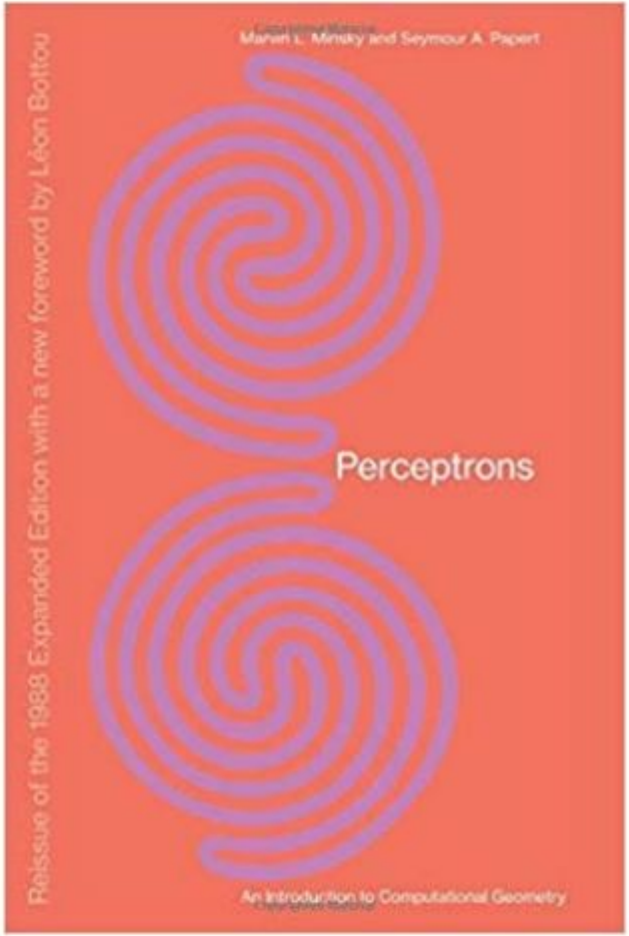
This image by Rocky Acosta is licensed under CC-BY 3.0

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

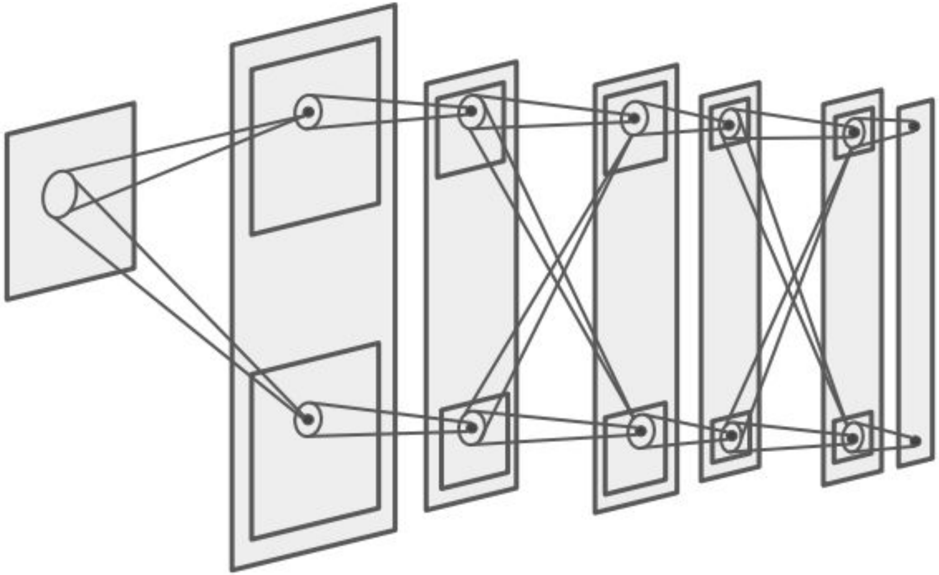

Marvin L. Minsky and Seymour A. Papert
Perceptrons
An Introduction to Computational Geometry
Reissue of the 1988 Expanded Edition with a new foreword by Léon Bottou

Neocognitron: Fukushima, 1980

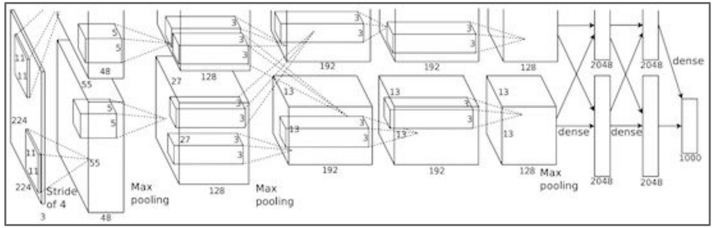
Computational model the visual system, directly inspired by Hubel and Wiesel's hierarchy of complex and simple cells

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

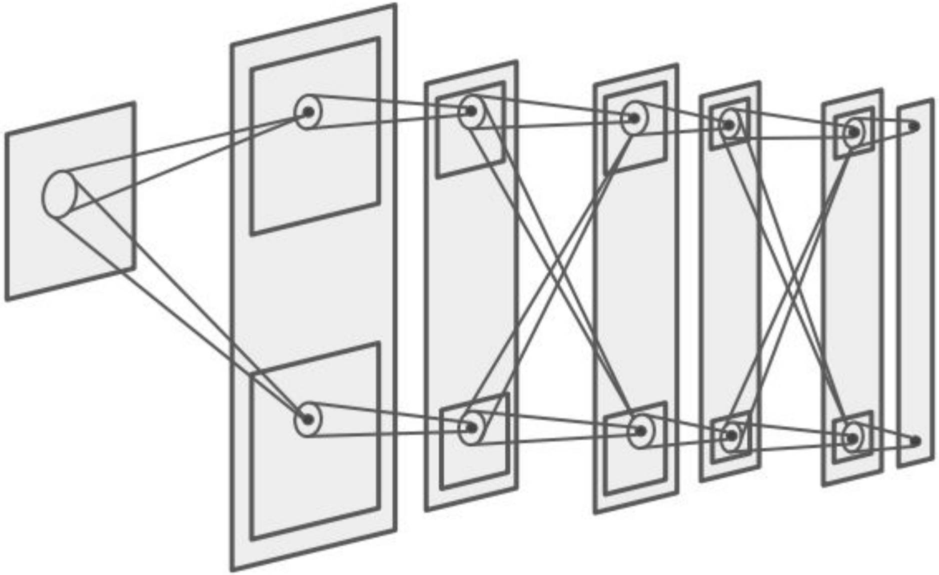
No practical training algorithm



A lot like AlexNet today



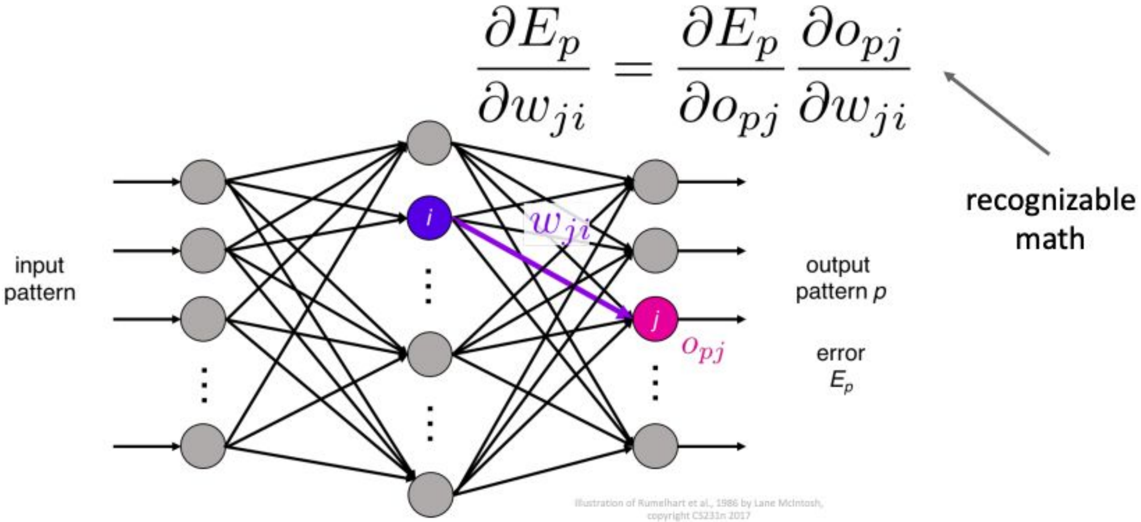
“AlexNet”



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

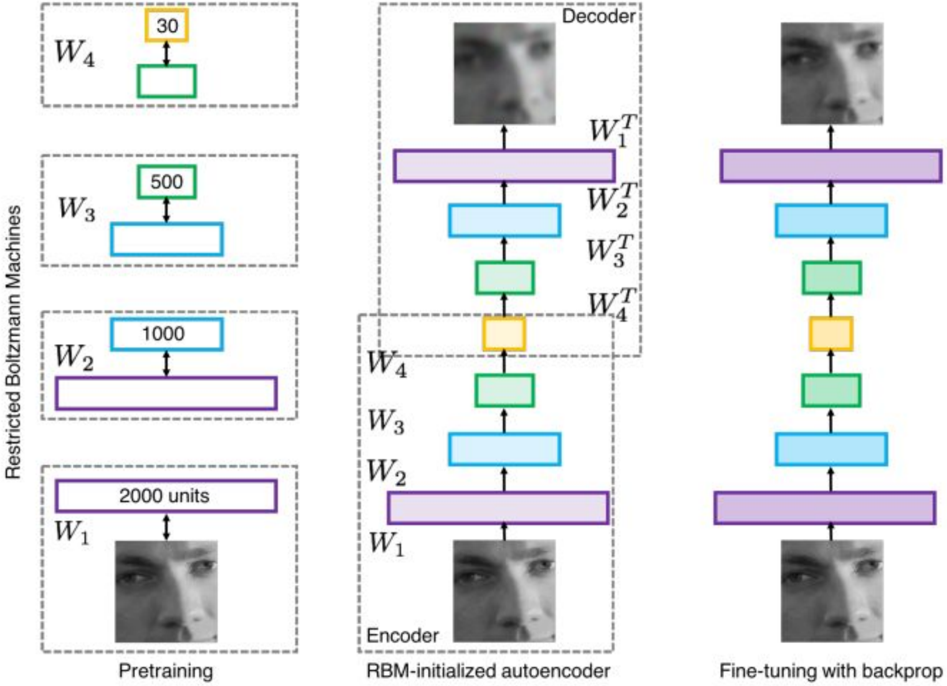
Illustration of Rumelhart et al., 1986 by Lane McIntosh, copyright CS231n 2017

2000s: “Deep Learning”

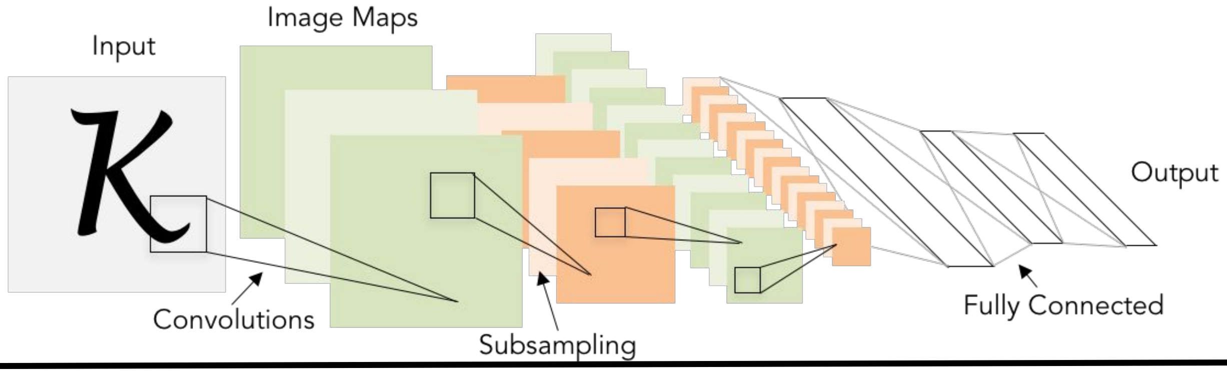
People tried to train neural networks that were deeper and deeper

Not a mainstream research topic at this time

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



1998 LeCun et al.



# of transistors

10^6

# of pixels used to train:

10^7 NIST

2012 Krizhevsky et al.

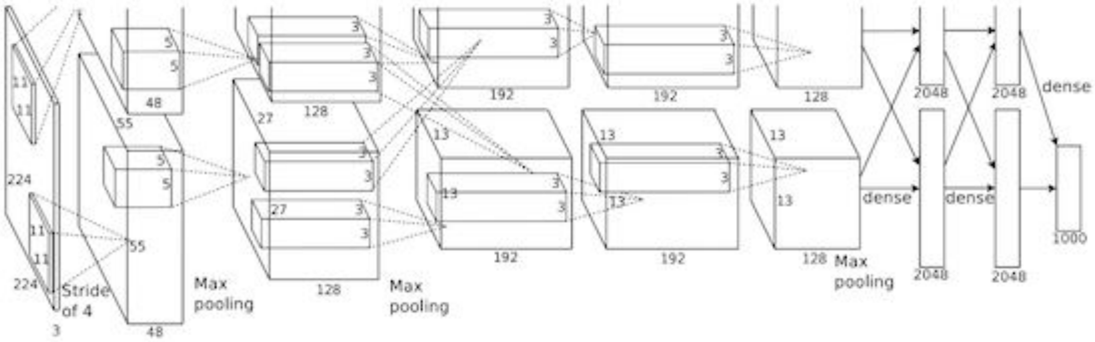


Figure copyright Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton, 2012. Reproduced with permission.

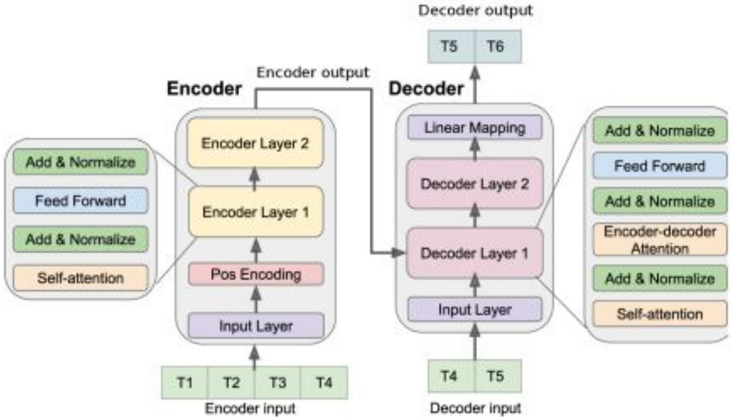
# of transistors

10^9

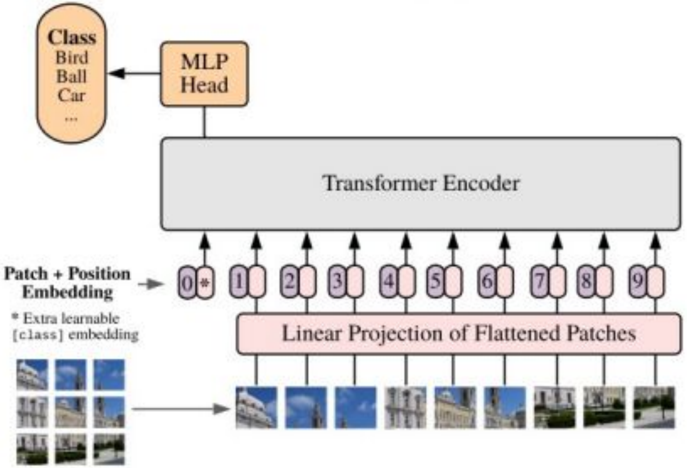
# of pixels used to train:

10^{14} IMAGENET

Today: Homogenization of Deep Learning
Same models for GPT-4 and image recognition



Transformer Models originally designed for NLP


Class
Bird
Ball
Car
...
MLP Head
Transformer Encoder
Patch + Position Embedding
* Extra learnable [class] embedding
0
*
1
2
3
4
5
6
7
8
9
Linear Projection of Flattened Patches

Almost identical model (Visual Transformers) can be applied to Computer Vision tasks

2012 to present: deep learning is everywhere

Image Classification

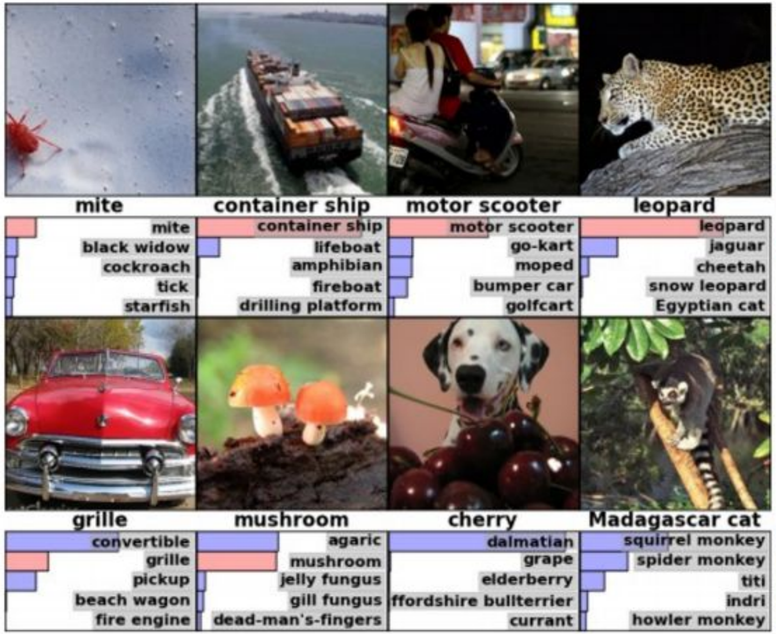
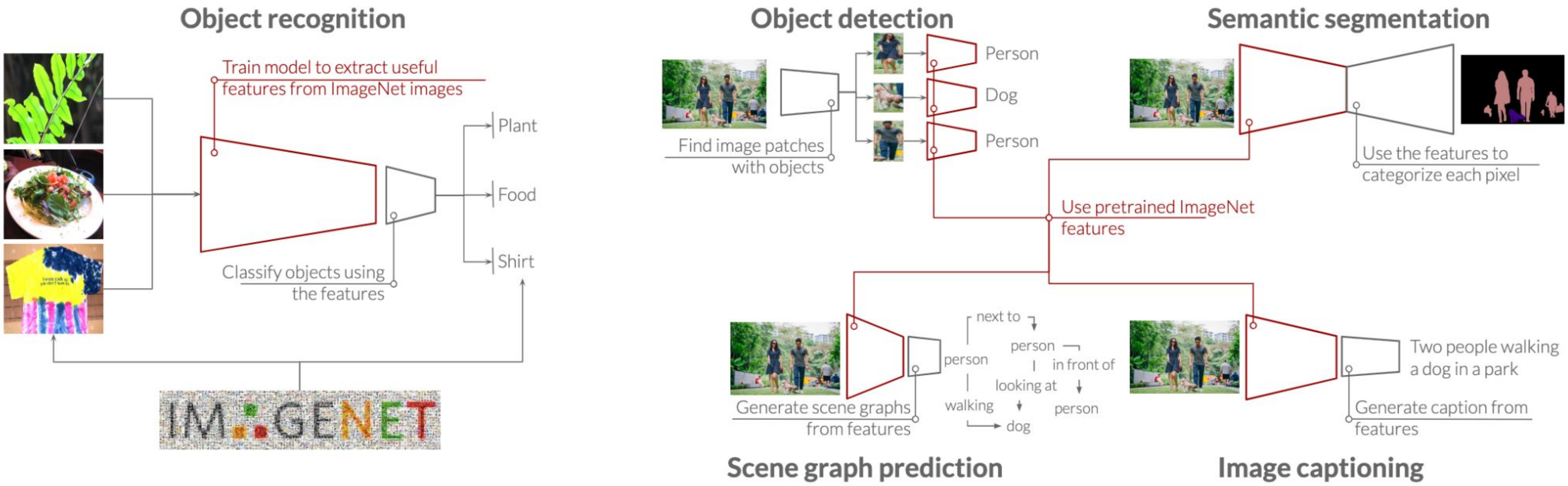


Image Retrieval

Data hungry machine learning models are now everywhere

Pretraining on ImageNet for object classification → Transfer ImageNet features for many other tasks:



Object recognition

Train model to extract useful features from ImageNet images

Plant

Food

Shirt

Classify objects using the features

IMAGENET

Object detection

Person

Dog

Person

Find image patches with objects

Semantic segmentation

Use the features to categorize each pixel

Use pretrained ImageNet features

next to

person

in front of

person

looking at

walking

person

dog

Generate scene graphs from features

Scene graph prediction

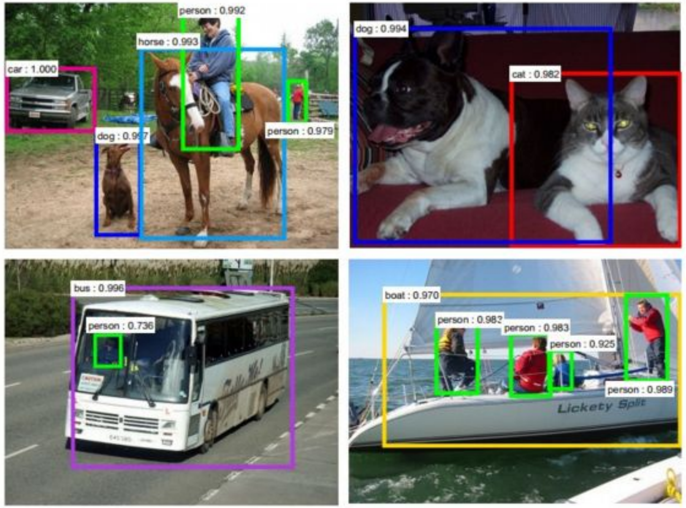
Two people walking a dog in a park

Generate caption from features

Image captioning

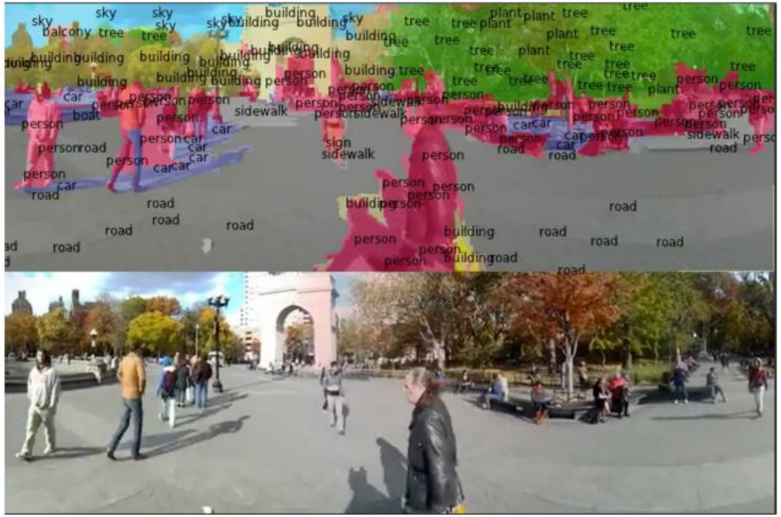
2012 to Present: Deep Learning is Everywhere

Object Detection



Ren, He, Girshick, and Sun, 2015

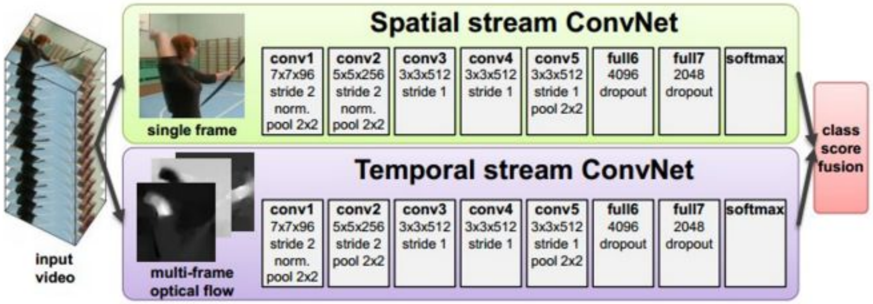
Image Segmentation



Fabaret et al, 2012

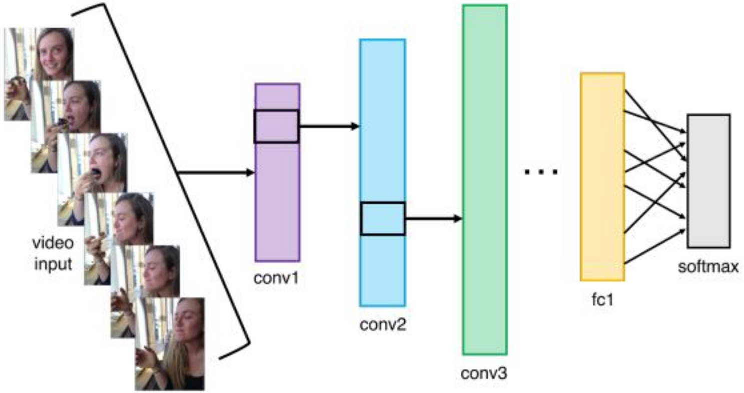
2012 to Present: Deep Learning is Everywhere

Video Classification



Simonyan et al, 2014

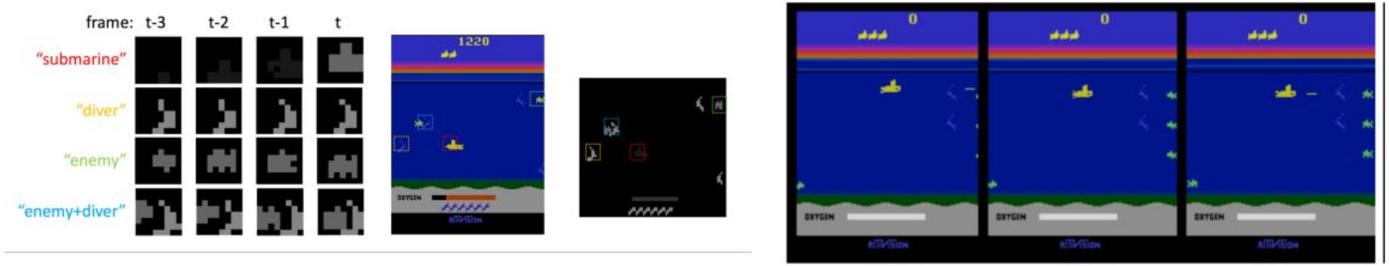
Activity Recognition



2012 to Present: Deep Learning is Everywhere

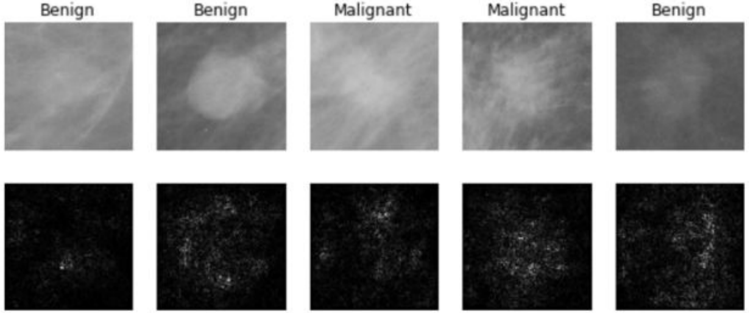
Pose Recognition (Toshev and Szegedy, 2014)

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

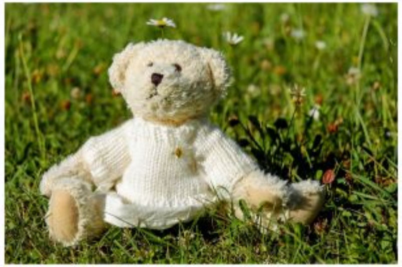
Kaggle Challenge

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

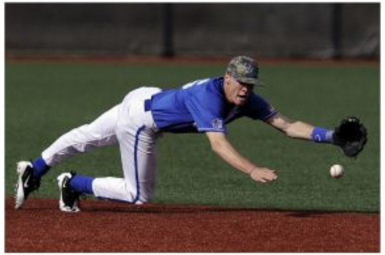
2012 to Present: Deep Learning is Everywhere

Image Captioning

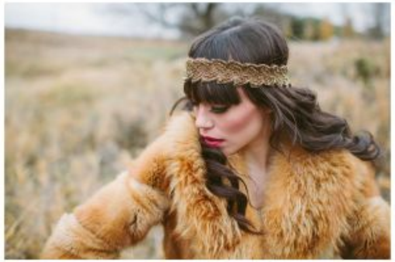
Vinyals et al, 2015
Karpathy and Fei-Fei, 2015



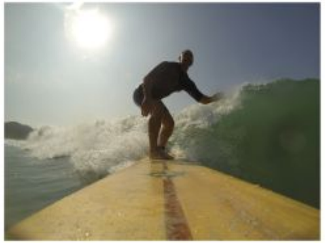
A white teddy bear sitting in the grass



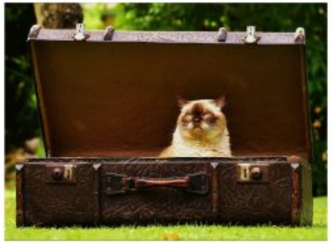
A man in a baseball uniform throwing a ball



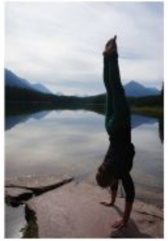
A woman is holding a cat in her hand



A man riding a wave on top of a surfboard



A cat sitting on a suitcase on the floor

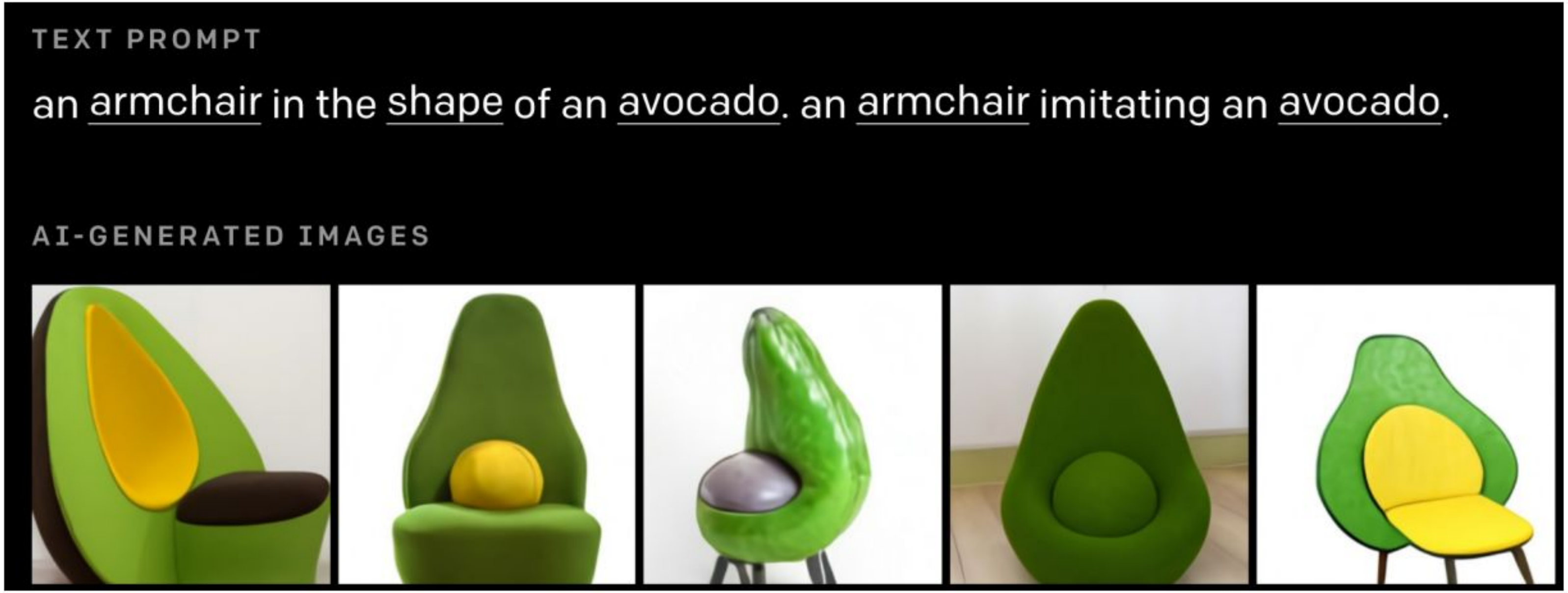


A woman standing on a beach holding a surfboard

All images are CC0 Public domain:
https://pixabay.com/en/luggage-antique-cat-1643010/
https://pixabay.com/en/teddy-plush-bears-cute-teddy-bear-1623436/
https://pixabay.com/en/surf-wave-summer-sport-litoral-1668716/
https://pixabay.com/en/woman-female-model-portrait-adult-983967/
https://pixabay.com/en/handstand-lake-meditation-496008/
https://pixabay.com/en/baseball-player-shortstop-infield-1045263/

Captions generated by Justin Johnson using Neuraltalk2

2012 to Present: Deep Learning is Everywhere



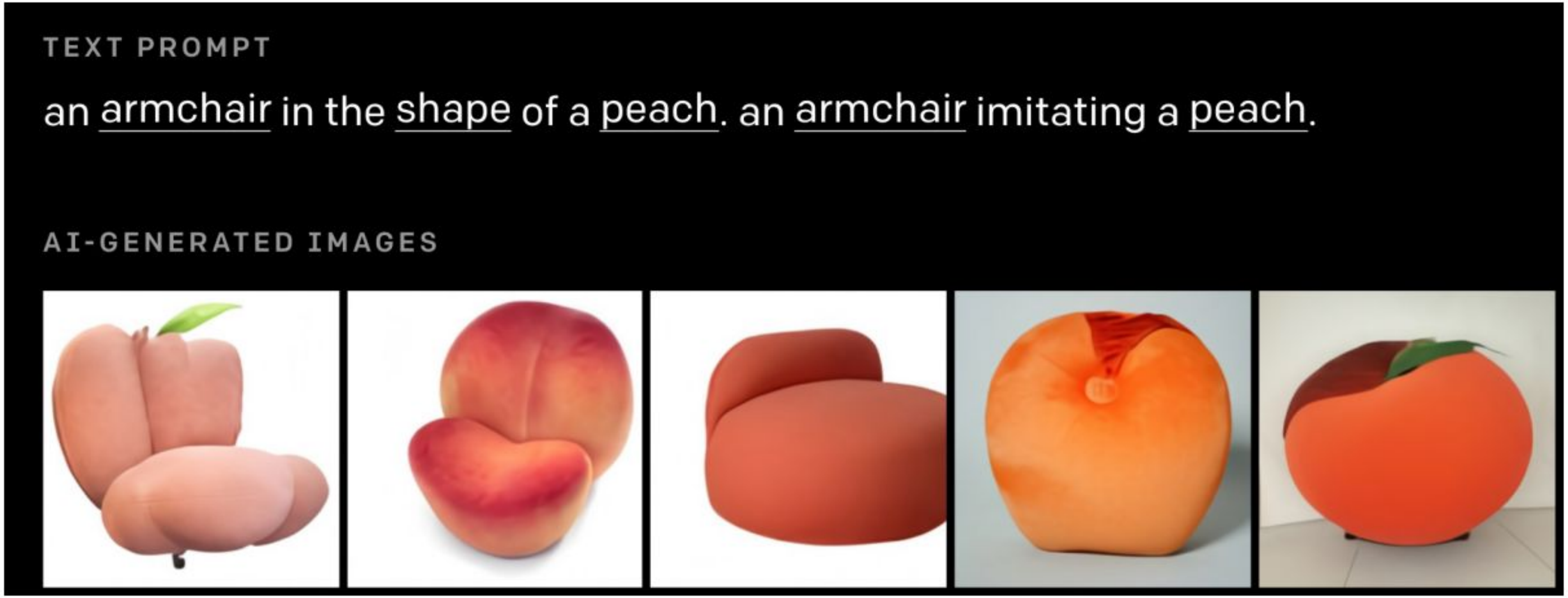
TEXT PROMPT

an armchair in the shape of an avocado. an armchair imitating an avocado.

AI-GENERATED IMAGES

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

2012 to Present: Deep Learning is Everywhere



TEXT PROMPT

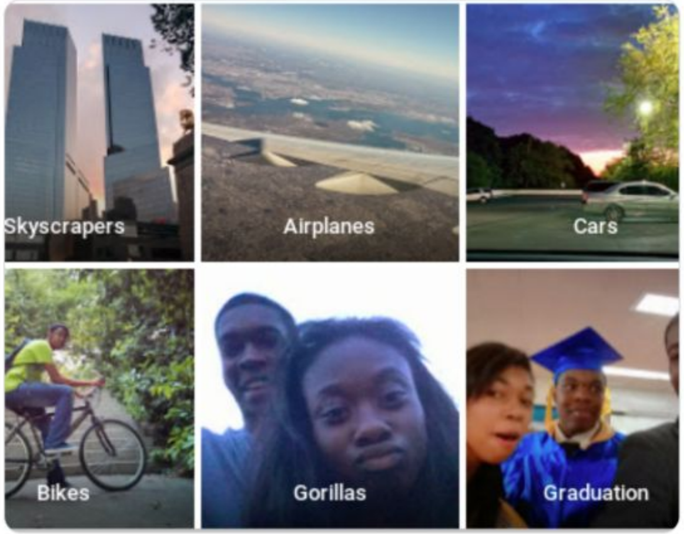
an armchair in the shape of a peach. an armchair imitating a peach.

AI-GENERATED IMAGES

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

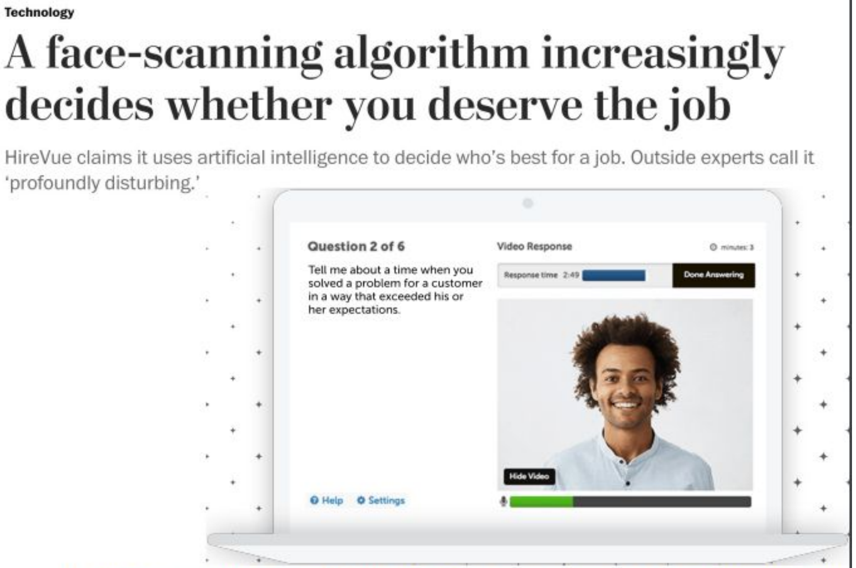
Despite progress, deep learning can be harmful

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


Technology

A face-scanning algorithm increasingly decides whether you deserve the job

HireVue claims it uses artificial intelligence to decide who's best for a job. Outside experts call it 'profoundly disturbing.'

Question 2 of 6

Tell me about a time when you solved a problem for a customer in a way that exceeded his or her expectations.

Video Response

minutes: 3

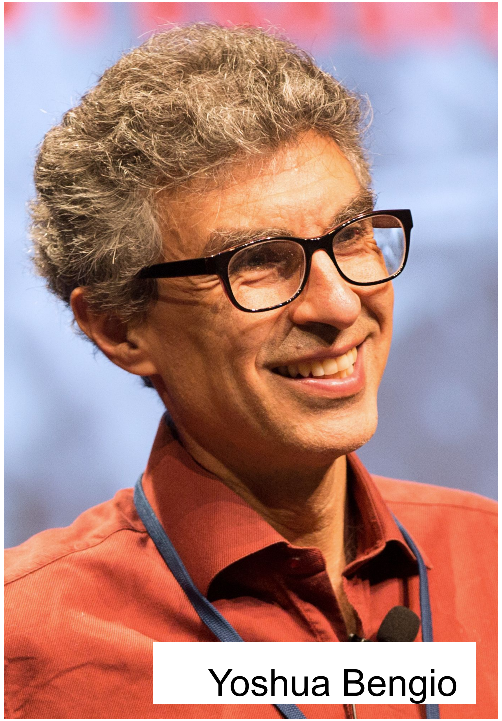
Response time 2:49

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

2018 Turing Award for deep learning

most prestigious technical award, is given for major contributions of lasting importance to computing.

This image is CC0 public domain



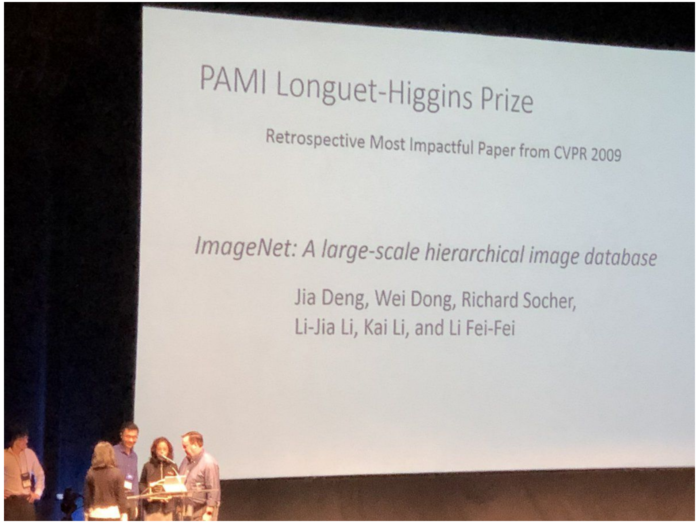
This image is CC0 public domain

This image is CC0 public domain

IEEE PAMI Longuet-Higgins Prize

Award recognizes ONE Computer Vision paper from **ten years ago** with **significant impact on computer vision** research.

In 2019, it was awarded to the 2009 original ImageNet paper



PAMI Longuet-Higgins Prize

Retrospective Most Impactful Paper from CVPR 2009

ImageNet: A large-scale hierarchical image database

Jia Deng, Wei Dong, Richard Socher,
Li-Jia Li, Kai Li, and Li Fei-Fei

Algorithms
13
13
3
3
192
13
13
3
3
192
13
13
128
Max
dense
dense
2048
2048
1000
224
11
11
55
27
3
3
Data
Computation
GTX
GTX

In this course, we will study these algorithms and architectures starting from a grounding in Visual Recognition

A fundamental and general problem in Computer Vision, that has roots in Cognitive Science

Image Classification: A core task in Computer Vision

This image by Nikita is licensed under CC-BY 2.0

→ cat

Image by US Army is licensed under CC BY 2.0

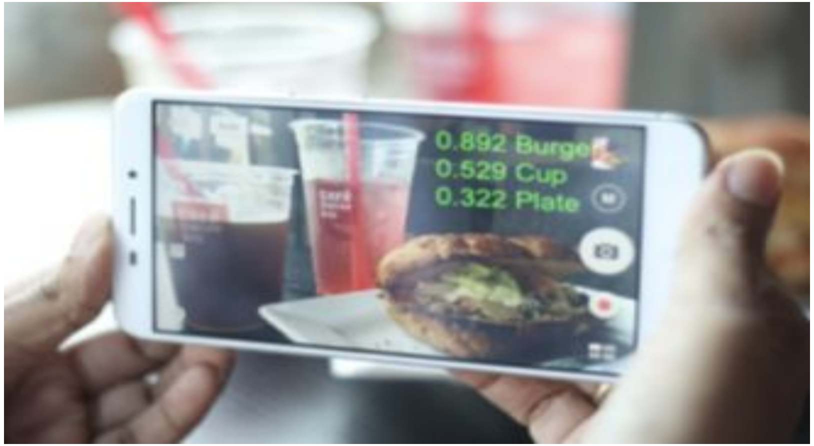


Image is CC0 1.0 public domain

Image by Kippelboy is licensed under CC BY-SA 3.0

Image by Christina C. is licensed under CC BY-SA 4.0

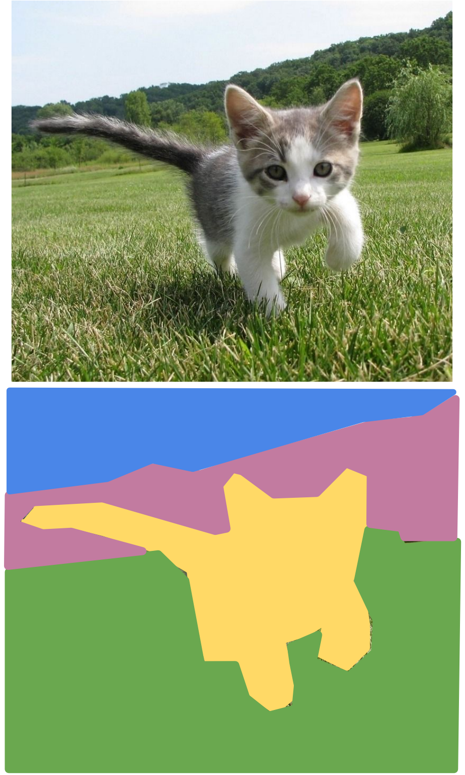
Object detection
car

Action recognition
bicycling

Scene graph prediction
<person - holding - hammer>

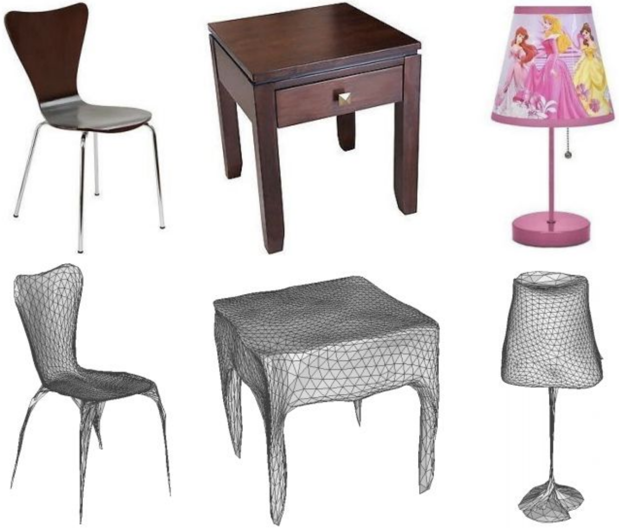
Captioning:
a person holding a hammer

Beyond recognition: Segmentation, 2D/3D Generation



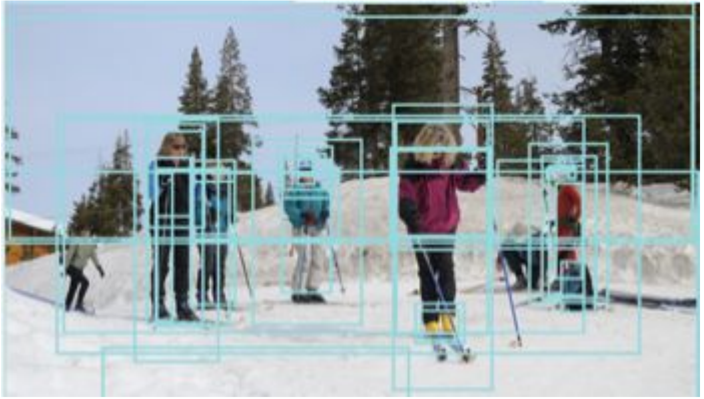
This image is CC0 public domain

Progressive GAN, Karras 2018.



Wang et al, “Pixel2Mesh: Generating 3D Mesh Models from Single RGB Images”, ECCV 2018

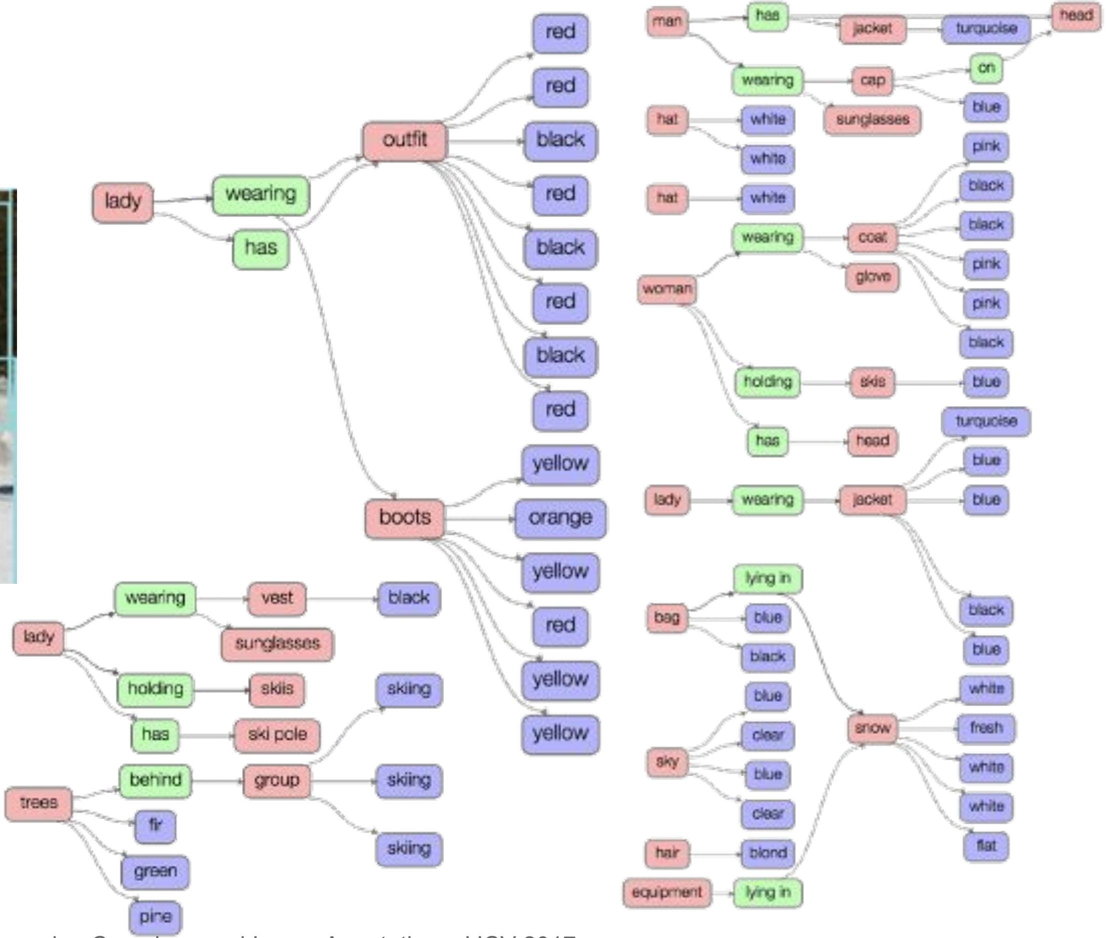
Scene Graphs



This image is CC0 public domain

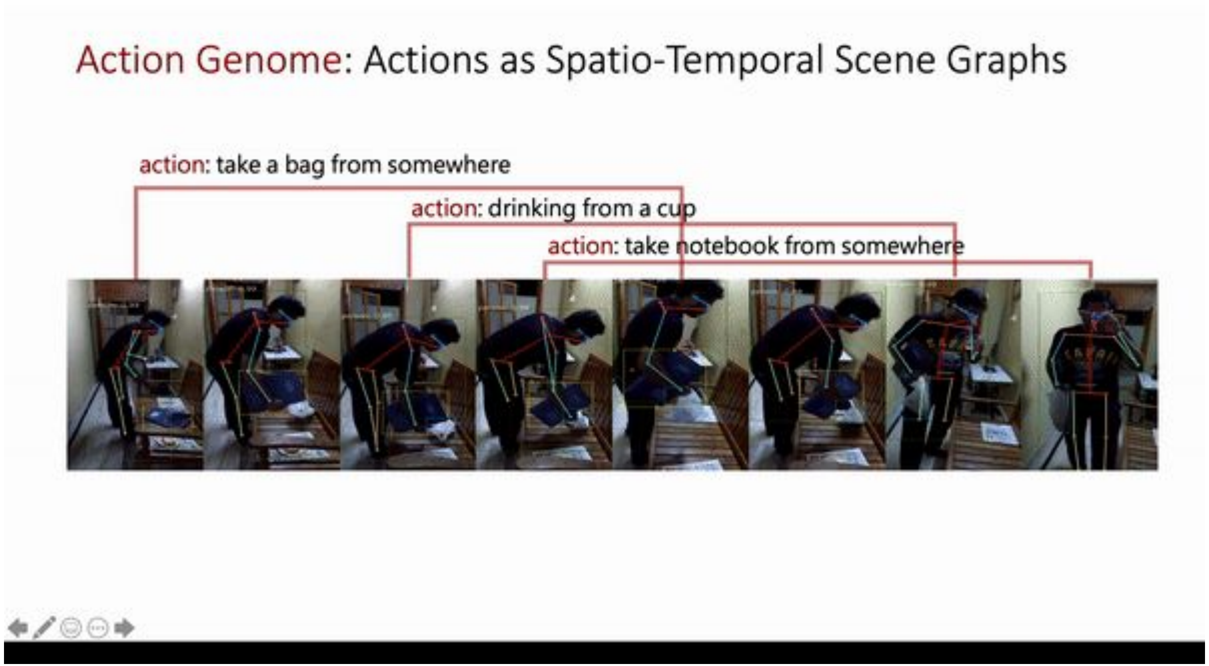
Three Ways Computer Vision Is Transforming Marketing

- Forbes Technology Council



Krishna et al., Visual Genome: Connecting Vision and Language using Crowdsourced Image Annotations, IJCV 2017

Spatio-temporal scene graphs

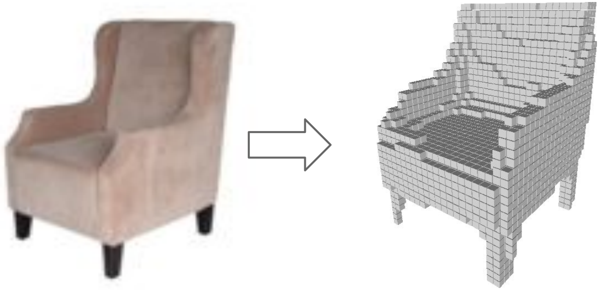


Action Genome: Actions as Spatio-Temporal Scene Graphs

action: take a bag from somewhere
action: drinking from a cup
action: take notebook from somewhere

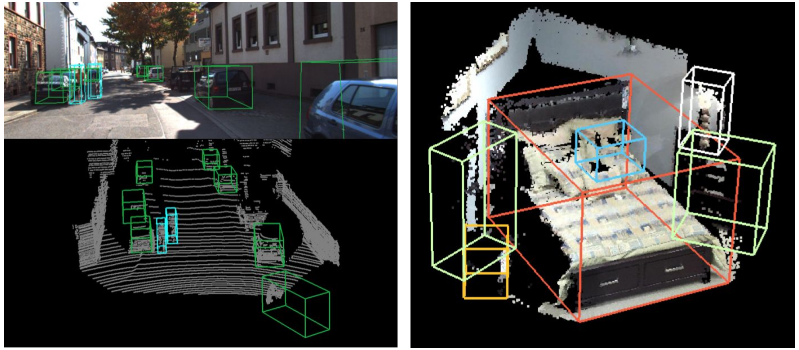
Ji, Krishna et al., Action Genome: Actions as Composition of Spatio-temporal Scene Graphs, CVPR 2020

3D Vision & Robotic Vision

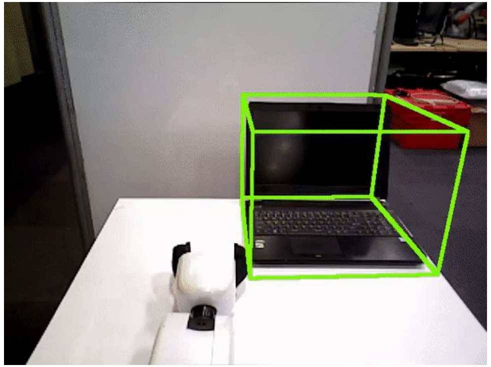


Choy et al., 3D-R2N2: Recurrent Reconstruction Neural Network (2016)

Mandlekar and Xu et al., Learning to Generalize Across Long-Horizon Tasks from Human Demonstrations (2020)



Xu et al., PointFusion: Deep Sensor Fusion for 3D Bounding Box Estimation (2018)



Wang et al., 6-PACK: Category-level 6D Pose Tracker with Anchor-Based Keypoints (2020)

PT = 500ms

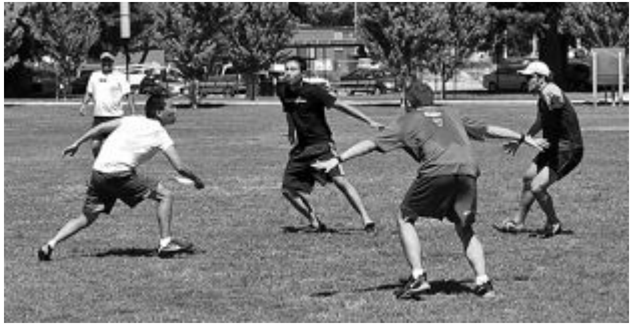


Image is licensed under CC BY-SA 3.0; changes made

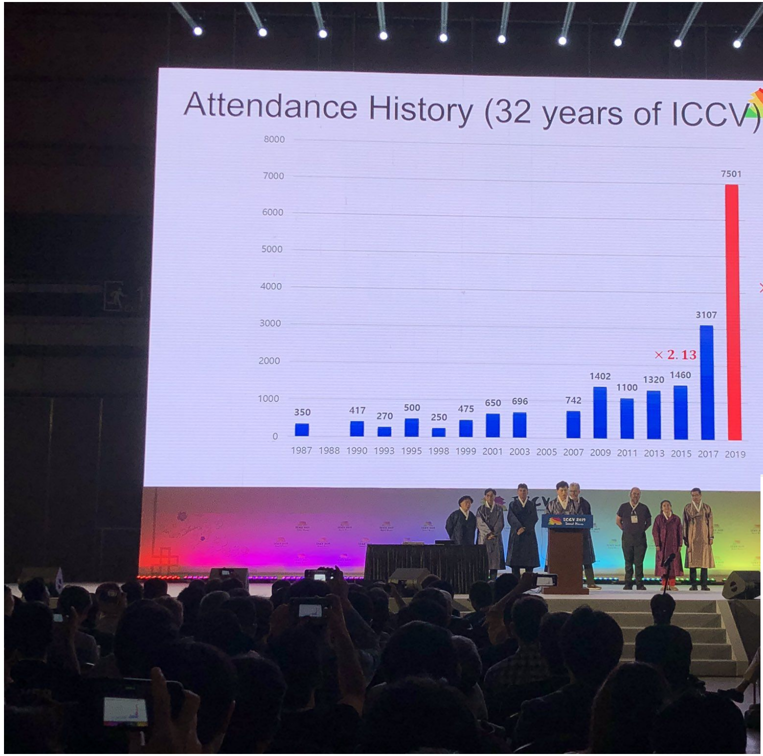
Some kind of game or fight. Two groups of two men? The man on the left is throwing something. Outdoors seemed like because i have an impression of grass and maybe lines on the grass? That would be why I think perhaps a game, rough game though, more like rugby than football because they pairs weren't in pads and helmets, though I did get the impression of similar clothing. maybe some trees? in the background.

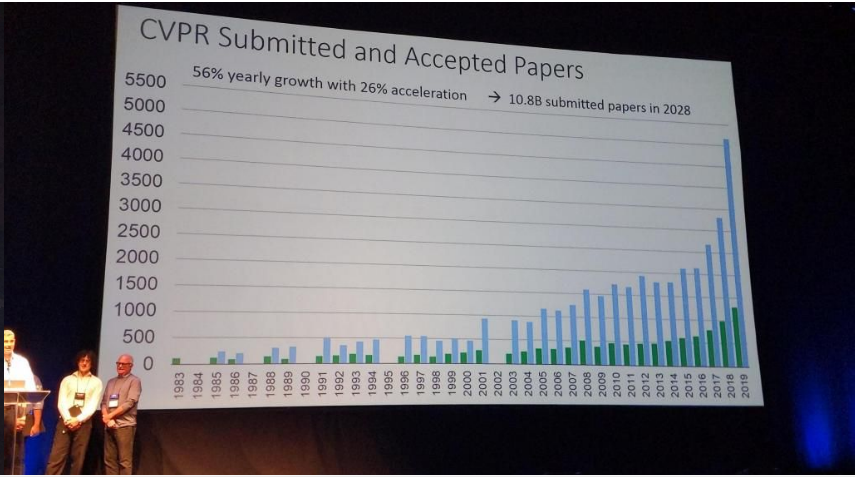
Fei-Fei, Iyer, Koch, Perona, *JoV,* 2007

And there is a lot we don’t know how to do

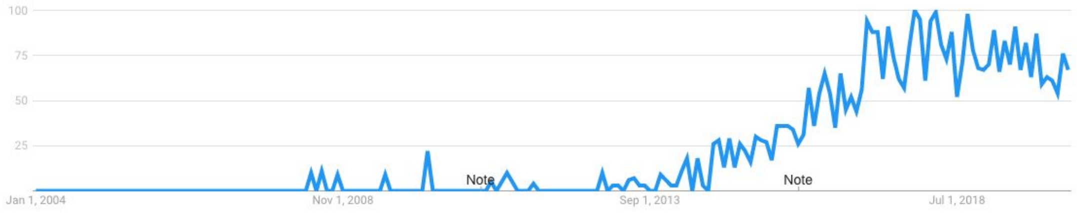
https://fedandfit.com/wp-content/uploads/2020/06/summer-activities-for-kids_optimized-scaled.jpeg

Why is deep learning its own course?



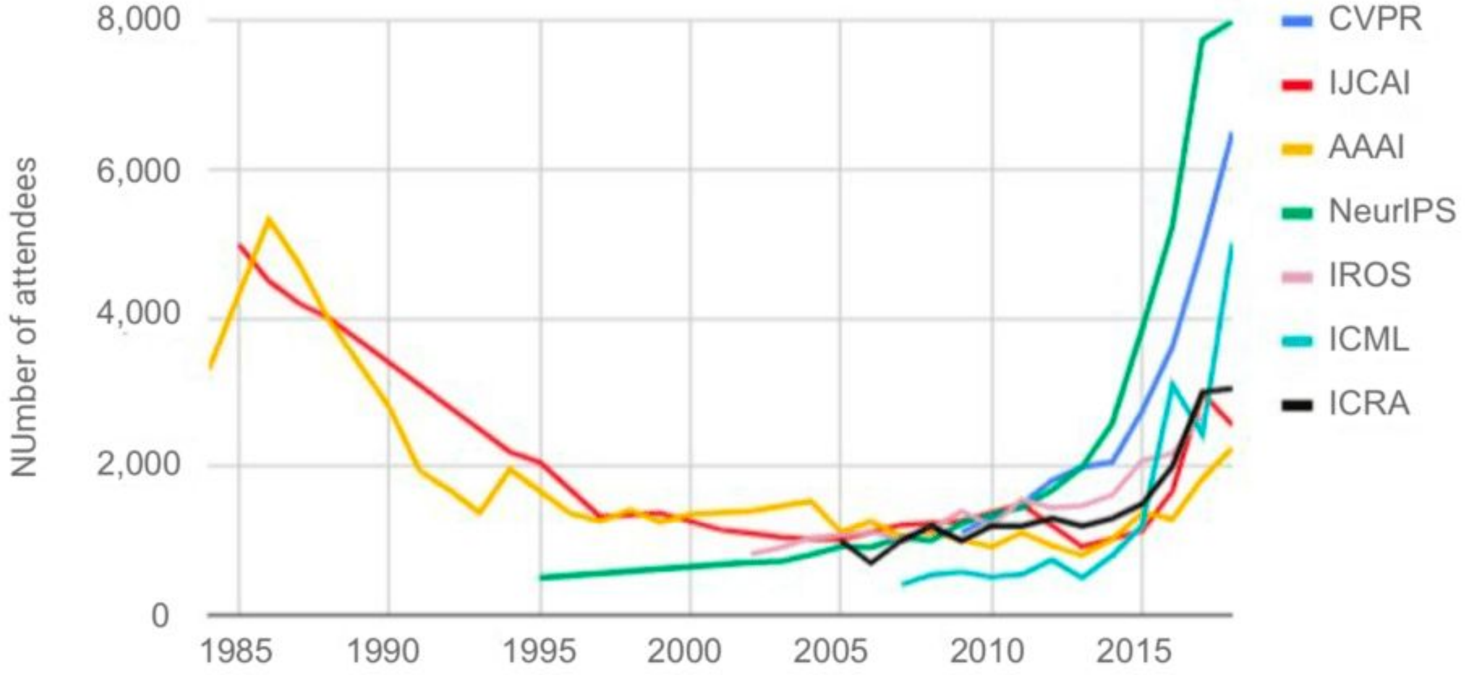


Google search trends for convolutional neural networks



Attendance at large conferences (1984—2018)

Source: Conference provided data



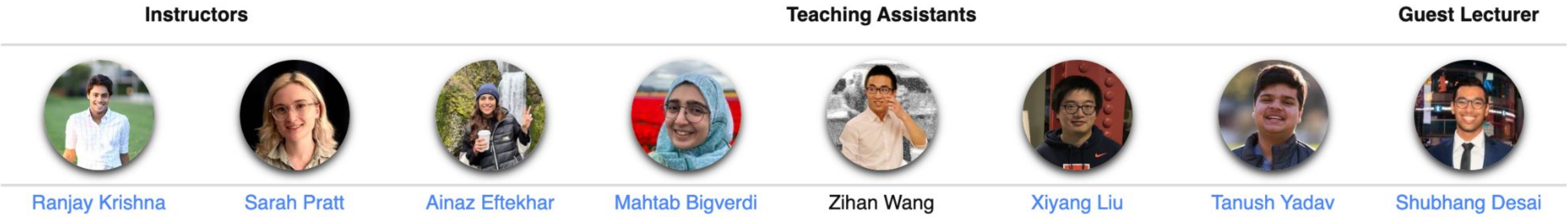
Today's agenda

- A brief history of computer vision
- CSE 493G1/ 599 overview

Survey - A show of hands

Undergrad?
M.S.?
Ph.D.?

CSE / EE?
Other Engineering?
Math / Natural Science?
Others?



Instructors		Teaching Assistants					Guest Lecturer
Ranjay Krishna	Sarah Pratt	Ainaz Eftekhar	Mahtab Bigverdi	Zihan Wang	Xiyang Liu	Tanush Yadav	Shubhang Desai

Syllabus

Deep learning Fundamentals

Data-driven approaches
Linear classification & kNN
Loss functions
Optimization
Backpropagation
Multi-layer perceptrons
Neural Networks
Convolutions
RNNs / LSTMs
Transformers

Practical training skills

Pytorch 1.4 / Tensorflow 2.0
Activation functions
Batch normalization
Transfer learning
Data augmentation
Momentum / RMSProp / Adam
Architecture design

Applications

Image captioning
Interpreting machine learning
Generative AI
Fairness & ethics
Data-centric AI
Deep reinforcement learning
Self-supervised learning
Diffusion
LLMs

Lectures

In person in Gates building: SIG 134

- Zoom links and recordings will be shared via canvas:
  - Due to security reasons, please do not share zoom links publicly
- **Tuesdays** and **Thursdays** between **10am to 11:20am**
  - To watch the lectures later, you must login to canvas. We highly recommend coming in person
- Slides posted to our website:
  - https://courses.cs.washington.edu/courses/cse493g1/24wi/

Friday recitation sections

Fridays

- Two recitation sections:
    - 9:30-10:30am (CSE2 G10)
    - 12:30-1:30pm (JHN 174)

Hands-on concepts, some tutorials, more practical details than tuesday/thursday lectures

Check the syllabus page for more information on what is going to be covered when.

This Friday: Python / numpy / Google Cloud (Presenter: ???)

Quizzes

Goal: Evaluate individual understanding of concepts from assignments and lecture

Will consist of multiple choice and short answer questions and will take place during recitation (except for quiz 5).

It will cover all concepts covered up till the Tuesday lecture before each quiz.

EdStem discussions

For questions about assignments, midterm, projects, logistics, etc, use EdStem!

SCPD students: Use your @uw.edu address to register for EdStem;

Office Hours

See course webpage for schedule.

- Add your name to a queue when you arrive for a particular office hours
- TAs will usually conduct 1-1 conversations in front of the whole group unless otherwise requested for a private conversation.

Optional textbook resources

- *Deep Learning*
    - by Goodfellow, Bengio, and Courville
    - Here is a free version
- Mathematics of deep learning
    - Chapters 5, 6 7 are useful to understand vector calculus and continuous optimization
    - Free online version
- Dive into deep learning
    - An interactive deep learning book with code, math, and discussions, based on the NumPy interface.
    - Free online version

Grading

All assignments, coding and written portions, will be submitted via Gradescope.

We use an **auto-grading system**

- A consistent grading scheme,
- Public tests:
    - Students see results of public tests immediately
- Private tests
    - Generalizations of the public tests to thoroughly test your implementation

Grading

5 Assignments (A1-A5): 8% each = 40%

A0 is worth 0%

5 Quizzes on Fridays: 6% each = 24% (we will drop your lowest quiz score)

Course Project: 36%

- Project Proposal: 2%
- Milestone: 4%
- Final report: 20%
- Poster presentation: 10%

Participation **Extra Credit** in lectures: up to 5%

Grading

Late policy

- 5 free late days
- Can use at most 2 per assignment (or proposal or milestone)
- Afterwards, 25% penalty per day late
- No late days for project report
- Weekends count as 1 day.
  - So using 1 late day for a Friday 11:59pm deadline means you can submit by Sunday 11:59pm

Overview on communication

Course Website: https://courses.cs.washington.edu/courses/cse493g1/24wi/

- Syllabus, lecture slides, links to assignment downloads, etc

EdStem:

- Use this for most communication with course staff
- Ask questions about assignments, grading, logistics, etc
- Use private questions if you want to post code

Gradescope:

- For turning in homework and receiving grades

Canvas:

- For watching lecture videos

Assignments

All assignments will be completed using **Google Colab**

- We have a tutorial for how to use Google Colab on the website

Assignment 0 IS OUT!!!, due 1/11 by 11:59pm

- Easy assignment
- Hardest part is learning how to use colab and how to submit on gradescope
- Worth 0% of your grade
- Used to evaluate how prepared you are to take this course

Assignments

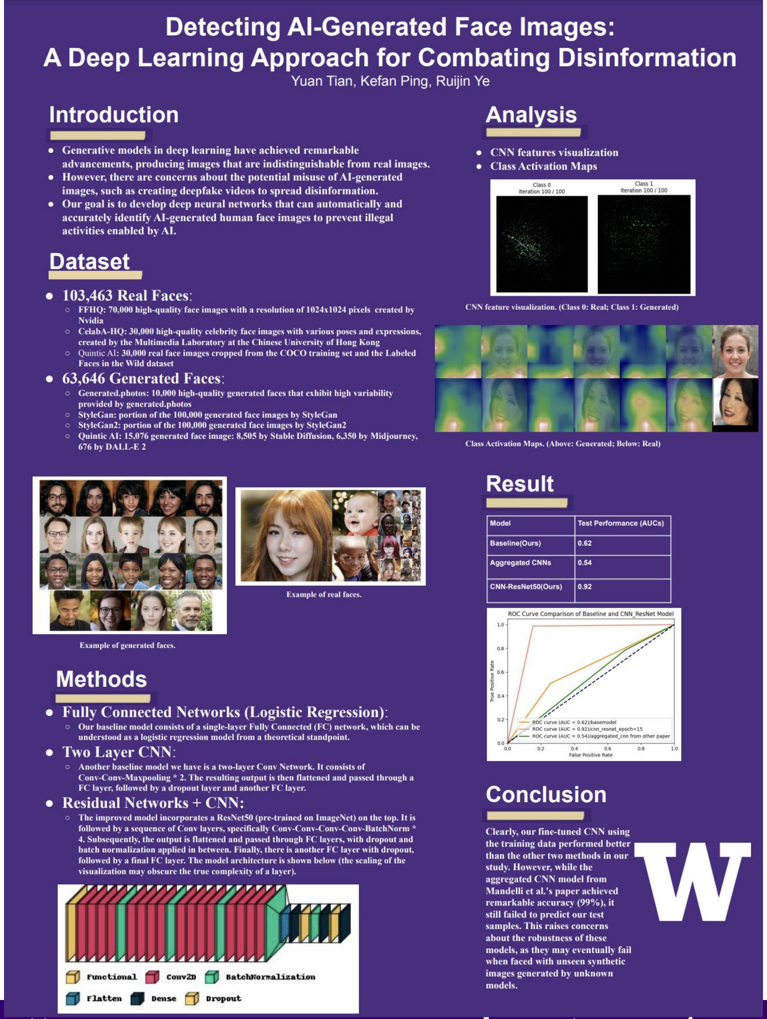
Assignment 1 will be released this weekend!!!, due 1/18 by 11:59pm

- K-Nearest Neighbor
- Linear classifiers: SVM, Softmax

Final project

- Groups of up to 3
- You can form groups yourselves
    - For students looking for groups, we will help assign you
- Anything related to deep learning

Example final project


Class 0
Iteration 100 / 100
Class 1
Iteration 100 / 100
ROC Curve Comparison of Baseline and CNN_ResNet Model
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
ROC curve (AUC = 0.62)/basemodel
ROC curve (AUC = 0.92)/cnn_resnet_epoch=15
ROC curve (AUC = 0.54)/aggregated_cnn from other paper
Functional
Conv2D
BatchNormalization
Flatten
Dense
Dropout

Detecting AI-Generated Face Images: A Deep Learning Approach for Combating Disinformation

Yuan Tian, Kefan Ping, Ruijin Ye

Introduction

- Generative models in deep learning have achieved remarkable advancements, producing images that are indistinguishable from real images.
- However, there are concerns about the potential misuse of AI-generated images, such as creating deepfake videos to spread disinformation.
- Our goal is to develop deep neural networks that can automatically and accurately identify AI-generated human face images to prevent illegal activities enabled by AI.

Dataset

- 103,463 Real Faces:
  - FFHQ: 70,000 high-quality face images with a resolution of 1024x1024 pixels created by Nvidia
  - CelabA-HQ: 30,000 high-quality celebrity face images with various poses and expressions, created by the Multimedia Laboratory at the Chinese University of Hong Kong
  - Quintic AI: 30,000 real face images cropped from the COCO training set and the Labeled Faces in the Wild dataset
- 63,646 Generated Faces:
  - Generated.photos: 10,000 high-quality generated faces that exhibit high variability provided by generated.photos
  - StyleGan: portion of the 100,000 generated face images by StyleGan
  - StyleGan2: portion of the 100,000 generated face images by StyleGan2
  - Quintic AI: 15,076 generated face image: 8,505 by Stable Diffusion, 6,350 by Midjourney, 676 by DALL-E 2

Example of generated faces.

Example of real faces.

Methods

- Fully Connected Networks (Logistic Regression):
  - Our baseline model consists of a single-layer Fully Connected (FC) network, which can be understood as a logistic regression model from a theoretical standpoint.
- Two Layer CNN:
  - Another baseline model we have is a two-layer Conv Network. It consists of Conv-Conv-Maxpooling * 2. The resulting output is then flattened and passed through a FC layer, followed by a dropout layer and another FC layer.
- Residual Networks + CNN:
  - The improved model incorporates a ResNet50 (pre-trained on ImageNet) on the top. It is followed by a sequence of Conv layers, specifically Conv-Conv-Conv-Conv-BatchNorm * 4. Subsequently, the output is flattened and passed through FC layers, with dropout and batch normalization applied in between. Finally, there is another FC layer with dropout, followed by a final FC layer. The model architecture is shown below (the scaling of the visualization may obscure the true complexity of a layer).

Analysis

- CNN features visualization
- Class Activation Maps

CNN feature visualization. (Class 0: Real; Class 1: Generated)

Class Activation Maps. (Above: Generated; Below: Real)

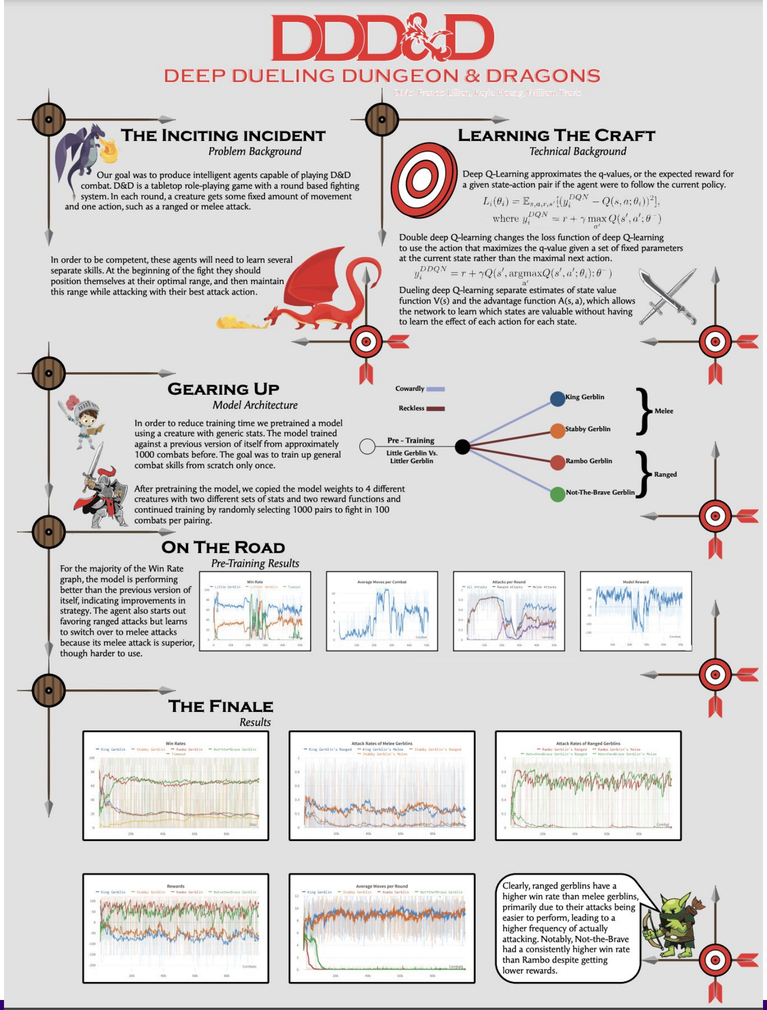
Result

Model	Test Performance (AUCs)
Baseline(Ours)	0.62
Aggregated CNNs	0.54
CNN-ResNet50(Ours)	0.92

Conclusion

Clearly, our fine-tuned CNN using the training data performed better than the other two methods in our study. However, while the aggregated CNN model from Mandelli et al.'s paper achieved remarkable accuracy (99%), it still failed to predict our test samples. This raises concerns about the robustness of these models, as they may eventually fail when faced with unseen synthetic images generated by unknown models.

Example final project



DDD&D

DEEP DUELING DUNGEON & DRAGONS

DMs: Francis Lillian, Kayla Hoang, William Travis

The Inciting Incident

Problem Background

Our goal was to produce intelligent agents capable of playing D&D combat. D&D is a tabletop role-playing game with a round based fighting system. In each round, a creature gets some fixed amount of movement and one action, such as a ranged or melee attack.

In order to be competent, these agents will need to learn several separate skills. At the beginning of the fight they should position themselves at their optimal range, and then maintain this range while attacking with their best attack action.

Learning The Craft

Technical Background

Deep Q-Learning approximates the q-values, or the expected reward for a given state-action pair if the agent were to follow the current policy.

$$L_i(\theta_i) = \mathbb{E}_{s,a,r,s'}[(y_i^{DQN} - Q(s,a;\theta_i))^2],$$
$$\text{where } y_i^{DQN} = r + \gamma \max_{a'} Q(s',a';\theta^-)$$

Double deep Q-learning changes the loss function of deep Q-learning to use the action that maximizes the q-value given a set of fixed parameters at the current state rather than the maximal next action.

$$y_i^{DDQN} = r + \gamma Q(s', \underset{a'}{\text{argmax}} Q(s',a';\theta_i); \theta^-)$$

Dueling deep Q-learning separate estimates of state value function V(s) and the advantage function A(s, a), which allows the network to learn which states are valuable without having to learn the effect of each action for each state.

Gearing Up

Model Architecture

In order to reduce training time we pretrained a model using a creature with generic stats. The model trained against a previous version of itself from approximately 1000 combats before. The goal was to train up general combat skills from scratch only once.

After pretraining the model, we copied the model weights to 4 different creatures with two different sets of stats and two reward functions and continued training by randomly selecting 1000 pairs to fight in 100 combats per pairing.

On The Road

Pre-Training Results

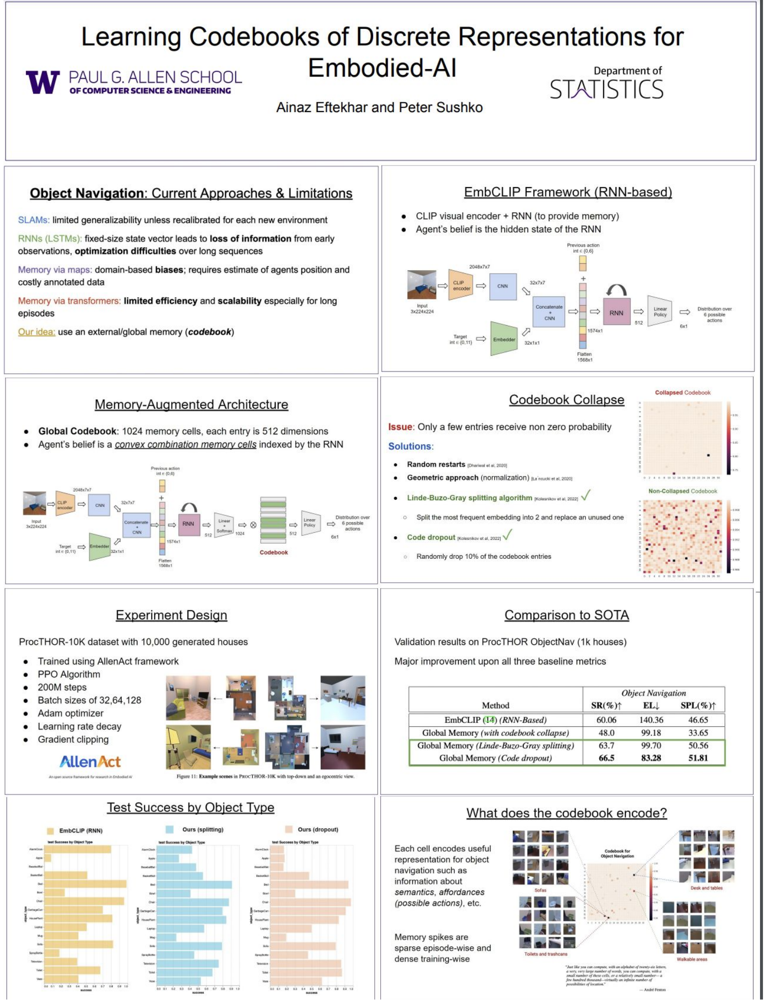
For the majority of the Win Rate graph, the model is performing better than the previous version of itself, indicating improvements in strategy. The agent also starts out favoring ranged attacks but learns to switch over to melee attacks because its melee attack is superior, though harder to use.

The Finale

Results

Clearly, ranged gerblins have a higher win rate than melee gerblins, primarily due to their attacks being easier to perform, leading to a higher frequency of actually attacking. Notably, Not-the-Brave had a consistently higher win rate than Rambo despite getting lower rewards.

Example final project



Learning Codebooks of Discrete Representations for Embodied-AI

PAUL G. ALLEN SCHOOL OF COMPUTER SCIENCE & ENGINEERING

Ainaz Eftekhar and Peter Sushko

Department of STATISTICS

Object Navigation: Current Approaches & Limitations

SLAMs: limited generalizability unless recalibrated for each new environment

RNNs (LSTMs): fixed-size state vector leads to **loss of information** from early observations, **optimization difficulties** over long sequences

Memory via maps: domain-based **biases**; requires estimate of agents position and costly annotated data

Memory via transformers: **limited efficiency** and **scalability** especially for long episodes

Our idea: use an external/global memory (***codebook***)

EmbCLIP Framework (RNN-based)

- CLIP visual encoder + RNN (to provide memory)
- Agent's belief is the hidden state of the RNN

Memory-Augmented Architecture

- **Global Codebook**: 1024 memory cells, each entry is 512 dimensions
- Agent's belief is a *convex combination memory cells* indexed by the RNN

Codebook Collapse

Issue: Only a few entries receive non zero probability

Solutions:

- **Random restarts** [Dhariwal et al, 2020]
- **Geometric approach** (normalization) [La´ncucki et al, 2020]
- **Linde-Buzo-Gray splitting algorithm** [Kolesnikov et al, 2022] ✓
  - Split the most frequent embedding into 2 and replace an unused one
- **Code dropout** [Kolesnikov et al, 2022] ✓
  - Randomly drop 10% of the codebook entries

Experiment Design

ProcTHOR-10K dataset with 10,000 generated houses

- Trained using AllenAct framework
- PPO Algorithm
- 200M steps
- Batch sizes of 32,64,128
- Adam optimizer
- Learning rate decay
- Gradient clipping

AllenAct
An open source framework for research in Embodied AI

Figure 11: **Example scenes** in PROCTHOR-10K with top-down and an egocentric view.

Comparison to SOTA

Validation results on ProcTHOR ObjectNav (1k houses)

Major improvement upon all three baseline metrics

	Object Navigation		
Method	SR(%)↑	EL↓	SPL(%)↑
EmbCLIP (14) *(RNN-Based)*	60.06	140.36	46.65
Global Memory *(with codebook collapse)*	48.0	99.18	33.65
Global Memory *(Linde-Buzo-Gray splitting)*	63.7	99.70	50.56
Global Memory *(Code dropout)*	**66.5**	**83.28**	**51.81**

Test Success by Object Type

What does the codebook encode?

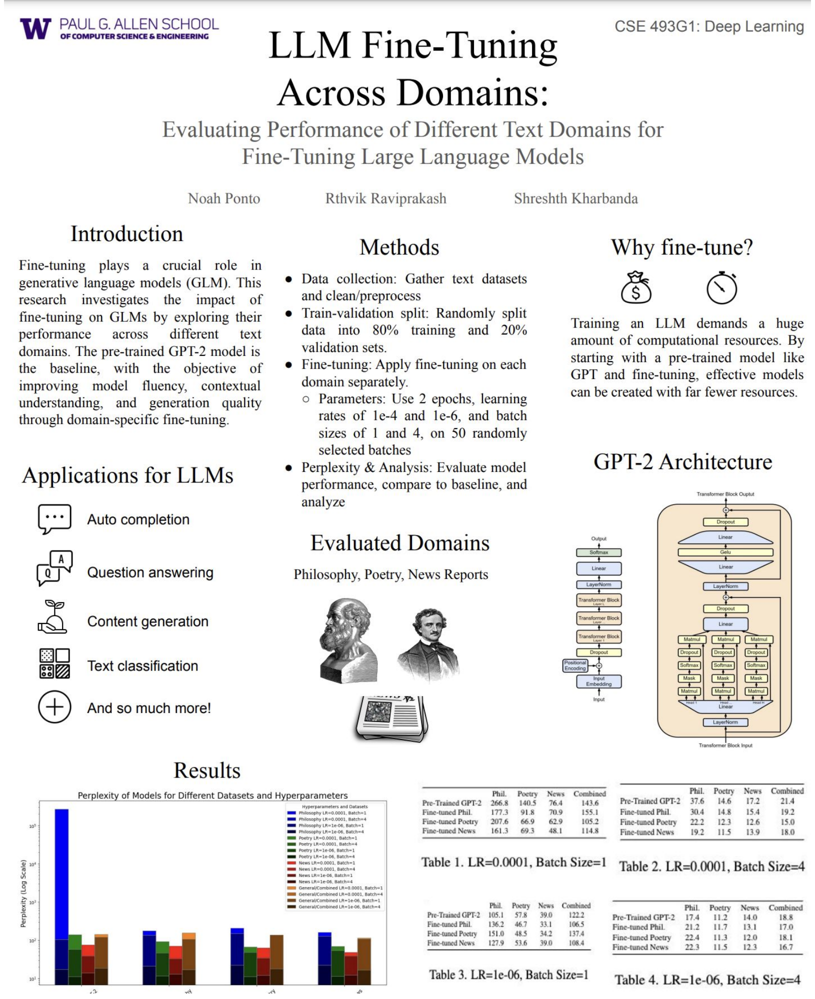
Each cell encodes useful representation for object navigation such as information about *semantics*, *affordances (possible actions)*, etc.

Memory spikes are sparse episode-wise and dense training-wise

"Just like you can compute, with an alphabet of twenty-six letters, a very, very large number of words, you can compute, with a small number of these cells, or a relatively small number— a few hundred thousand—virtually an infinite number of possibilities of location."

— André Fenton

Example final project



PAUL G. ALLEN SCHOOL OF COMPUTER SCIENCE & ENGINEERING

CSE 493G1: Deep Learning

LLM Fine-Tuning Across Domains:

Evaluating Performance of Different Text Domains for Fine-Tuning Large Language Models

Noah Ponto   Rthvik Raviprakash   Shreshth Kharbanda

Introduction

Fine-tuning plays a crucial role in generative language models (GLM). This research investigates the impact of fine-tuning on GLMs by exploring their performance across different text domains. The pre-trained GPT-2 model is the baseline, with the objective of improving model fluency, contextual understanding, and generation quality through domain-specific fine-tuning.

Applications for LLMs

- Auto completion
- Question answering
- Content generation
- Text classification
- And so much more!

Methods

- Data collection: Gather text datasets and clean/preprocess
- Train-validation split: Randomly split data into 80% training and 20% validation sets.
- Fine-tuning: Apply fine-tuning on each domain separately.
    - Parameters: Use 2 epochs, learning rates of 1e-4 and 1e-6, and batch sizes of 1 and 4, on 50 randomly selected batches
- Perplexity & Analysis: Evaluate model performance, compare to baseline, and analyze

Evaluated Domains

Philosophy, Poetry, News Reports

Why fine-tune?

Training an LLM demands a huge amount of computational resources. By starting with a pre-trained model like GPT and fine-tuning, effective models can be created with far fewer resources.

GPT-2 Architecture

Results

Perplexity of Models for Different Datasets and Hyperparameters

Perplexity (Log Scale)
Hyperparameters and Datasets
Philosophy LR=0.0001, Batch=1
Philosophy LR=0.0001, Batch=4
Philosophy LR=1e-06, Batch=1
Philosophy LR=1e-06, Batch=4
Poetry LR=0.0001, Batch=1
Poetry LR=0.0001, Batch=4
Poetry LR=1e-06, Batch=1
Poetry LR=1e-06, Batch=4
News LR=0.0001, Batch=1
News LR=0.0001, Batch=4
News LR=1e-06, Batch=1
News LR=1e-06, Batch=4
General/Combined LR=0.0001, Batch=1
General/Combined LR=0.0001, Batch=4
General/Combined LR=1e-06, Batch=1
General/Combined LR=1e-06, Batch=4

	Phil.	Poetry	News	Combined
Pre-Trained GPT-2	266.8	140.5	76.4	143.6
Fine-tuned Phil.	177.3	91.8	70.9	155.1
Fine-tuned Poetry	207.6	66.9	62.9	105.2
Fine-tuned News	161.3	69.3	48.1	114.8

Table 1. LR=0.0001, Batch Size=1

	Phil.	Poetry	News	Combined
Pre-Trained GPT-2	37.6	14.6	17.2	21.4
Fine-tuned Phil.	30.4	14.8	15.4	19.2
Fine-tuned Poetry	22.2	12.3	12.6	15.0
Fine-tuned News	19.2	11.5	13.9	18.0

Table 2. LR=0.0001, Batch Size=4

	Phil.	Poetry	News	Combined
Pre-Trained GPT-2	105.1	57.8	39.0	122.2
Fine-tuned Phil.	136.2	46.7	33.1	106.5
Fine-tuned Poetry	151.0	48.5	34.2	137.4
Fine-tuned News	127.9	53.6	39.0	108.4

Table 3. LR=1e-06, Batch Size=1

	Phil.	Poetry	News	Combined
Pre-Trained GPT-2	17.4	11.2	14.0	18.8
Fine-tuned Phil.	21.2	11.7	13.1	17.0
Fine-tuned Poetry	22.4	11.3	12.0	18.1
Fine-tuned News	22.3	11.5	12.3	16.7

Table 4. LR=1e-06, Batch Size=4

Pre-requisites

Proficiency in Python

- All class assignments will be in Python (and use numpy)
- Later in the class, you will be using Pytorch and TensorFlow
- We will go over a Python tutorial on this Friday’s recitation.

You need to know:

- **College Calculus,**
- **Linear Algebra,**
- **experience with Python**

No longer need Machine Learning as a prerequisite

Collaboration policy

Please follow UW student code of conduct – read it!

Here are our course specific rules:

- **Rule 1**: Don’t look at solutions or code that are not your own; everything you submit should be your own work. We have automatic tools that detect plagiarism.
- **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 code

Plagiarism and Collaboration

We will run all assignments through plagiarism software.

Additionally, you may use online resources to understand concepts, but not to complete the coding portion of your assignments. This includes Stack Overflow and ChatGPT.

We will compare all student solutions to ChatGPT generated solutions. If we detect plagiarism in your assignments, you will get a 0 on the assignment and we will have no choice but to report to the university.

**** It is much better to turn in an incomplete assignment than to turn in code that is not your own! ****

Learning objectives

Formalize deep learning applications into tasks

- Formalize inputs and outputs for vision-related problems
- Understand what data and computational requirements you need to train a model

Develop and train deep learning models

- Learn to code, debug, and train convolutional neural networks.
- Learn how to use software frameworks like TensorFlow and PyTorch

Gain an understanding of where the field is and where it is headed

- What new research has come out in the last 0-9 years
- What are open research challenges?
- What ethical and societal considerations should we consider before deployment?

What you should expect from us

Fun: We will discuss fun applications like image captioning, GPT, generative AI

What we expect from you

Patience.

- This is new for us as much as it is new for you
- Things will break; we will experience technical difficulties
- Bear with us and trust us to listen to you

Contribute

- Build a community with your peers
- Help one another - discuss topics you enjoy
- Give us (anonymous) feedback

Why should you take this class?

Become a deep learning researcher (an incomplete list of conferences)

- Get involved with research at UW: apply using this form.

Conferences:

- CVPR 2023, ACL 2023, NeurIPS 2023, ICML 2023

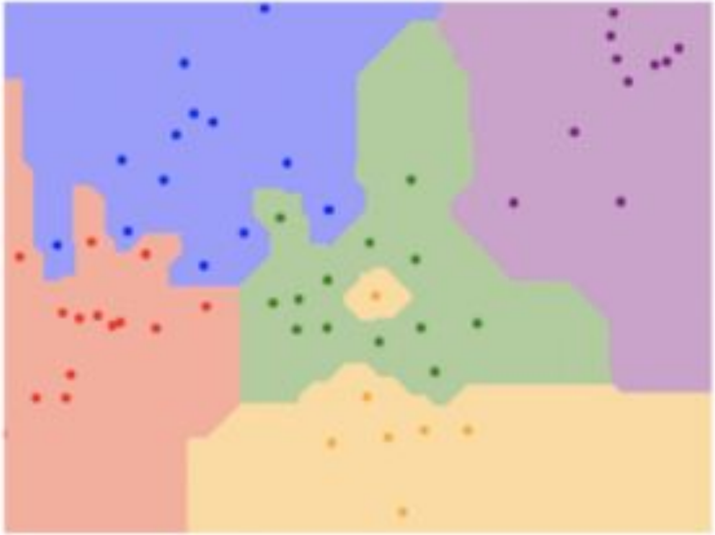
Become a deep learning engineer in industry (an incomplete list of industry teams)

- Brain team at Google AI
- OpenAI
- Meta’s Fundamental AI research team
- Microsoft’s AI research team

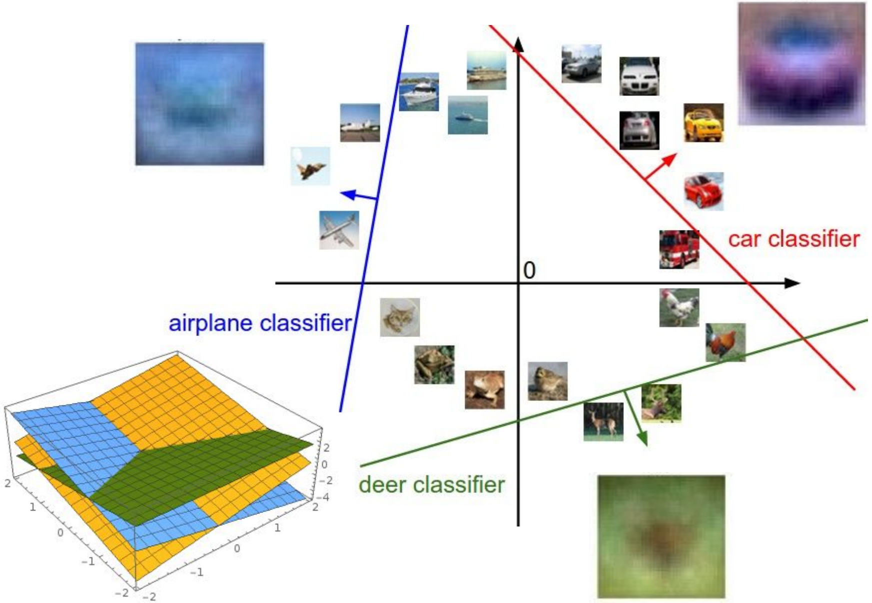
General interest

Next time: Image classification

k- nearest neighbor



Linear classification



Plot created using Wolfram Cloud

References

- Dalal, Navneet, and Bill Triggs. "Histograms of oriented gradients for human detection." Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on. Vol. 1. IEEE, 2005. [PDF]
- Felzenszwalb, Pedro, David McAllester, and Deva Ramanan. "A discriminatively trained, multiscale, deformable part model." Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on. IEEE, 2008 [PDF]
- Everingham, Mark, et al. "The pascal visual object classes (VOC) challenge." International Journal of Computer Vision 88.2 (2010): 303-338. [PDF]
- Deng, Jia, et al. "Imagenet: A large-scale hierarchical image database." Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. IEEE, 2009. [PDF]
- Russakovsky, Olga, et al. "Imagenet Large Scale Visual Recognition Challenge." arXiv:1409.0575. [PDF]
- Lin, Yuanqing, et al. "Large-scale image classification: fast feature extraction and SVM training." Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on. IEEE, 2011. [PDF]
- Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." Advances in neural information processing systems. 2012. [PDF]
- Szegedy, Christian, et al. "Going deeper with convolutions." arXiv preprint arXiv:1409.4842 (2014). [PDF]
- Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556 (2014). [PDF]
- He, Kaiming, et al. "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition." arXiv preprint arXiv:1406.4729 (2014). [PDF]
- LeCun, Yann, et al. "Gradient-based learning applied to document recognition." Proceedings of the IEEE 86.11 (1998): 2278-2324. [PDF]
- Fei-Fei, Li, et al. "What do we perceive in a glance of a real-world scene?." Journal of vision 7.1 (2007): 10. [PDF]

References

•Dalal, Navneet, and Bill Triggs. "Histograms of oriented gradients for human detection." Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on. Vol. 1. IEEE, 2005. [PDF]

•Felzenszwalb, Pedro, David McAllester, and Deva Ramanan. "A discriminatively trained, multiscale, deformable part model." Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on. IEEE, 2008 [PDF]

•Everingham, Mark, et al. "The pascal visual object classes (VOC) challenge." International Journal of Computer Vision 88.2 (2010): 303-338. [PDF]

•Deng, Jia, et al. "Imagenet: A large-scale hierarchical image database." Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. IEEE, 2009. [PDF]

•Russakovsky, Olga, et al. "Imagenet Large Scale Visual Recognition Challenge." arXiv:1409.0575. [PDF]

•Lin, Yuanqing, et al. "Large-scale image classification: fast feature extraction and SVM training." Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on. IEEE, 2011. [PDF]

•Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." Advances in neural information processing systems. 2012. [PDF]

•Szegedy, Christian, et al. "Going deeper with convolutions." arXiv preprint arXiv:1409.4842 (2014). [PDF]

•Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556 (2014). [PDF]

•He, Kaiming, et al. "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition." arXiv preprint arXiv:1406.4729 (2014). [PDF]

•LeCun, Yann, et al. "Gradient-based learning applied to document recognition." Proceedings of the IEEE 86.11 (1998): 2278-2324. [PDF]

•Fei-Fei, Li, et al. "What do we perceive in a glance of a real-world scene?." Journal of vision 7.1 (2007): 10. [PDF]