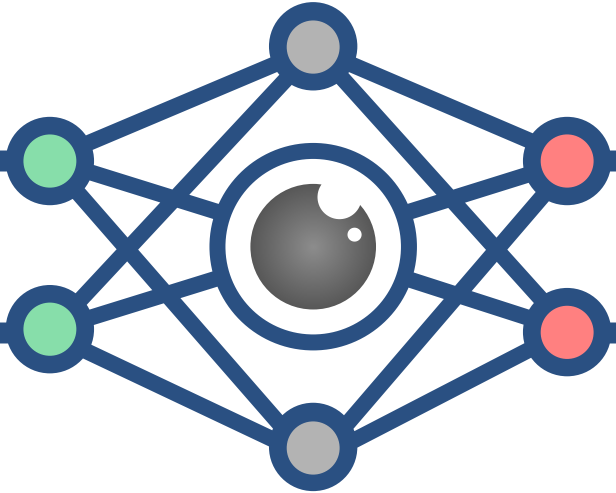
CS396

Deep Learning for Computer Vision



Lec 1: Introduction to Computer Vision

Course Staff

- **Instructor**: Jeova Farias Sales Rocha Neto (can call me Jeova or Prof. Farias)
  - Email: jfariassal@haverford.edu
  - Office Hours: Tues 4-5:30pm at L303 or just come by!
- **TA:** Isaac Wasserman (iwasserman@haverford.edu).
- **Instructor Bio:**
  - PhD, Comp. Engineering. (Brown University)
  - MSc, Applied Math (Brown University), CS (University of Nice, France)
- **Research interests:**
  - Statistical Machine Learning and Discrete Optimization,
  - Image Processing and Computer Vision.
  - More recently: Unsupervised Deep Learning.

Learning Objectives

- At of the course, the students are expected to:
  a. Understand what **Computer Vision** is and how it's been applied in the industry and academia.
  b. Know what neural networks are and how that led to the development of **Deep Learning**.
  c. Have a concise knowledge of how Deep Learning's development and the problems it brought solutions to.
  d. Have enough working skills on **PyTorch** to developer Deep Learning applications for, and more broadly than, Computer Vision.
  e. Be able to read and understand the **recent literature** in Deep Learning and Computer Vision and present their content to a wider audience.

What you can expect from this course

- You can expect from me:
  - Organized lectures and lab instructions.
    - OBS: Some lectures may not cover the whole 1.5 hours.
  - I’ll use Moodle as our main tool for communication and file storage.
    - You’ll find the lecture slides, syllabus, recordings, exams, etc. there.
    - **Pay attention** to the emails I’ll send.
  - Weekly office hours and individual help:
    - Make appointments whenever you feel the need,
    - When my door is open (it usually is), you can just walk in and chat.
  - **I’ll do my best in these**: clear lectures schedule and exams and timely assignment grading.
- I expect from you:
  - Courtesy with your peers when they ask questions,
  - Timely assignments turn-ins and (sometimes) patience with grades delivery.
  - We’ll do in-class exercises: **focus on them**. We’ll have breaks too.

What you can expect from this course

- Prerequisites:
  - In this course, we'll use **math** in some classes, especially calculus and linear algebra.
  - I also assume you understand the basics of machine learning, such as: training and testing data, supervised learning, overfitting and underfitting, etc.
  - In any case, I'll provide quick reviews as we go.
  - And, more importantly: **Feel free to ask questions about these topics!** It's ok if you forgot some of these things or never learned them well. I'll try my best not to help you on them too.
- We'll go through the very basics of Deep Learning, so, for those who may already know some stuff: **be patient** (and also contribute to the class, if possible).
- Some classes will **have faster pace** than others. Let me know if they are too fast, tho!
- Most of modern Deep Learning literature is still being written and it is not concise, so
  - We may use for different sources other than the textbook,
  - I may ask you to do paper readings, especially on the second half of the course.

Software and Hardware

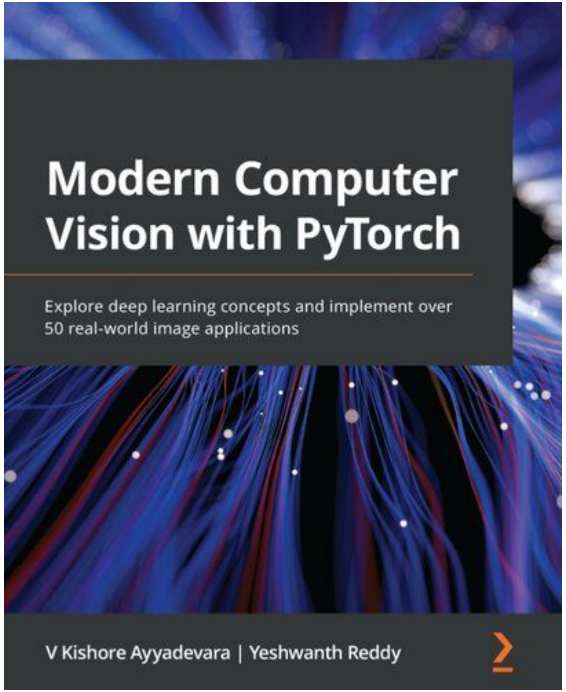
- One big disadvantage when doing/teaching Deep Learning: it requires expensive hardware, i.e., Graphics Processing Units (GPUs).
- While the College has GPUs available, our usage of them will (potentially) be limited.
- **Google Colab** is an option for codes in class, homeworks and you final project:
  - Uses Python notebooks (.ipynb).
  - It is free for any one with a Gdrive account,
  - Provides GPUs for us to run our codes on,
- One problem: in the free accounts, it may stop providing GPU access after a lot of usage on it.
- Over the semester, we’ll evaluate this drawback and find viable solutions for it.

Syllabus: Textbook

- It is hard to find a textbook for this course: the field changes extremely fast.
- However, the book we will most use in this course is:

  Modern Computer Vision with PyTorch, by V Kishore Ayyadevara and Yeshwanth Reddy. Packt Publishing Ltd, 2020.

  It's not an academic book, more like a "hands-on" learning material.
- Most of the code will be inspired from it.
- For the theory, use the slides + (potential) papers I'll send to you.



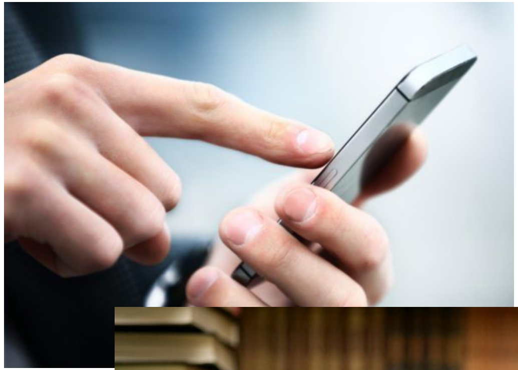
Syllabus: Final grade

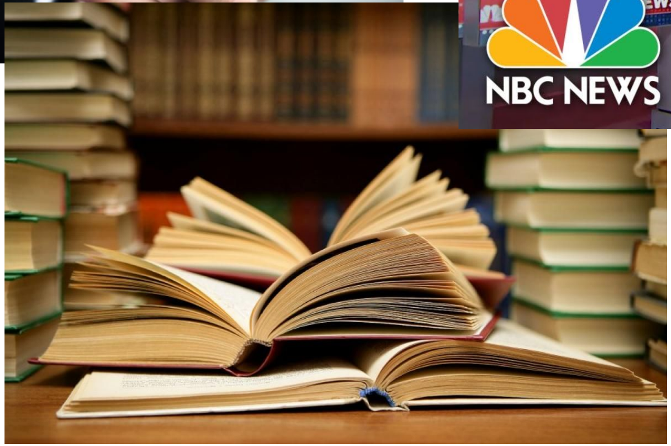
- **Labs (30 %):**
  - Every Thursday from 3-4p, 6-7 total, every 1-2 weeks.
  - You'll be given **4 late days for the semester** (budget your days wisely)
  - Deliverables will be reports **in Latex** (or else, for 80% of the grade), starting from Lab 2. Suggestion: Use Overleaf for starting off with Latex.
  - (*Tentative*) Students will pair-up in each lab and deliverable. *I may change this strategy later.*
- **Midterms (30 %):**
  - Two midterms (15% each), on October/6th and November/22nd. *Dates may change, if needed.*
- **Final Project (30 %):**
  - **Theme**: anything that deep learning as a solver **or** a review of two or more papers with code.
  - A proposal will be required by October/20th.
  - Deliverables: code and presentation. Teams of 2-4 people.
- **Participation (10 %):**
  - Includes attendance in class and labs and participating in class.

Syllabus: final thoughts

- **What counts as participation?**
  - **Asking** and **answering** questions in class (very important!)
    - Raise your hand and wait your turn;
    - Will call on groups, but only after giving you a few minutes to think/discuss.
  - Actively participating in in-class activities and labs.
  - Attending office hours.
- The lectures will be mainly **in person**, but I can live stream them if there is need.
- The all of this info (and a PDF of it) is on Moodle.
  - Most of your questions about logistics will be answered there.
  - **Make sure to read it!**

Let's get started: Data and the modern world





A DAY IN DATA

The exponential growth of data is undisputed, but the numbers behind this explosion – fuelled by internet of things and the use of connected devcies – are hard to comprehend, particularly when looked at in the context of one day

500m

tweets are sent every day

Twitter

4PB

of data created by Facebook, including

350m photos

100m hours of video watch time

Facebook Research

320bn

emails to be sent each day by 2021

294bn

billion emails are sent

Radicati Group

306bn

emails to be sent each day by 2020

3.9bn

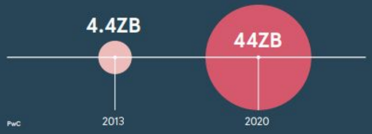
people use emails

4TB

of data produced by a connected car

Intel

ACCUMULATED DIGITAL UNIVERSE OF DATA



PwC

DEMYSTIFIYING DATA UNITS

From the more familiar 'bit' or 'megabyte', larger units of measurement are more frequently being used to explain the masses of data

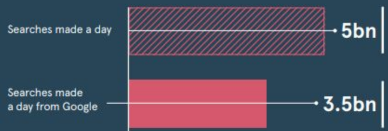
Unit		Value	Size
b	bit	0 or 1	1/8 of a byte
B	byte	8 bits	1 byte
KB	kilobyte	1,000 bytes	1,000 bytes
MB	megabyte	$1{,}000^2$ bytes	1,000,000 bytes
GB	gigabyte	$1{,}000^3$ bytes	1,000,000,000 bytes
TB	terabyte	$1{,}000^4$ bytes	1,000,000,000,000 bytes
PB	petabyte	$1{,}000^5$ bytes	1,000,000,000,000,000 bytes
EB	exabyte	$1{,}000^6$ bytes	1,000,000,000,000,000,000 bytes
ZB	zettabyte	$1{,}000^7$ bytes	1,000,000,000,000,000,000,000 bytes
YB	yottabyte	$1{,}000^8$ bytes	1,000,000,000,000,000,000,000,000 bytes

*A lowercase "b" is used as an abbreviation for bits, while an uppercase "B" represents bytes.

65bn

messages sent over WhatsApp and two billion minutes of voice and video calls made

Facebook



Smart insights

463EB

of data will be created every day by 2025

IDC

95m

photos and videos are shared on Instagram

Instagram Business

28PB

to be generated from wearable devices by 2020

Statista

RACONTEUR

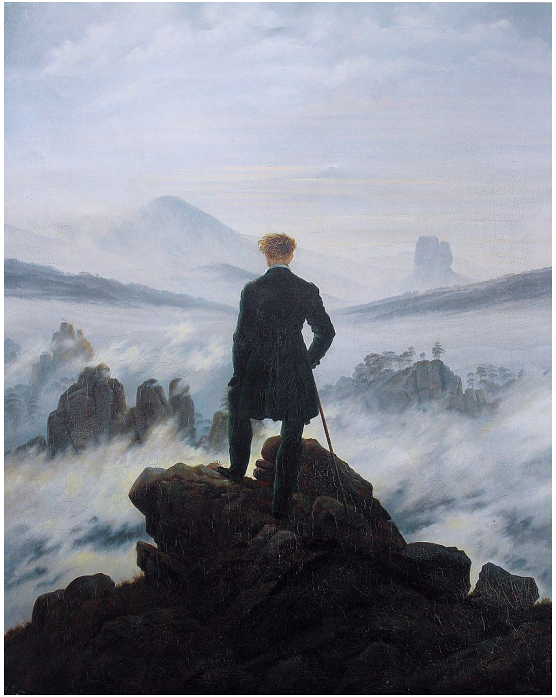
The richest data

- Arguably, the visual data (image and videos) is the **richest** among all types of data!
- In a simple image, we have have so much information:

1. A couple is drinking wine during the day.
2. The man has a beard, long hair and is wearing a white shirt.
3. The woman also has long hair and is wearing a typical French cap.
4. They seem in love with each other.

What else?

- The visual data is so rich that, we perceive **up to 80%** of all impressions by means of our sight.
- **The vision system is our main door to the world!**

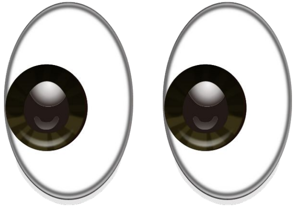


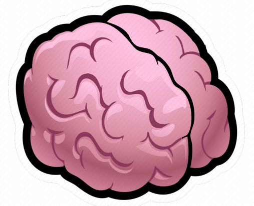
Caspar David Friedrich's *Wanderer above the Sea of Fog* (1817)

(Oversimplified) Vision pipeline

- In simple terms, our visual system proceeds as following when it sees an image:

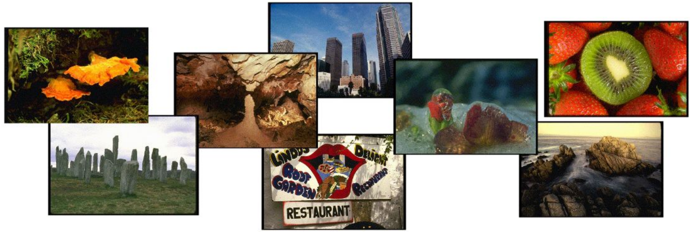
Image (or video) → Sensing device → Interpreting device → Interpretation





"A baby and a cat being extremely adorable"

- And it works very well: on average, we only need 150ms to detect whether there is an animal in a picture!



Computer Vision

- As Computer Scientists, we aim to do Computer Vision:

  Computer Vision (CV) is the study of how computers can gain high-level understanding from **visual data**, such as images and videos.

- Here, we seek to **understand and automate** tasks proper to the human visual system.
- **Our desire pipeline is:**

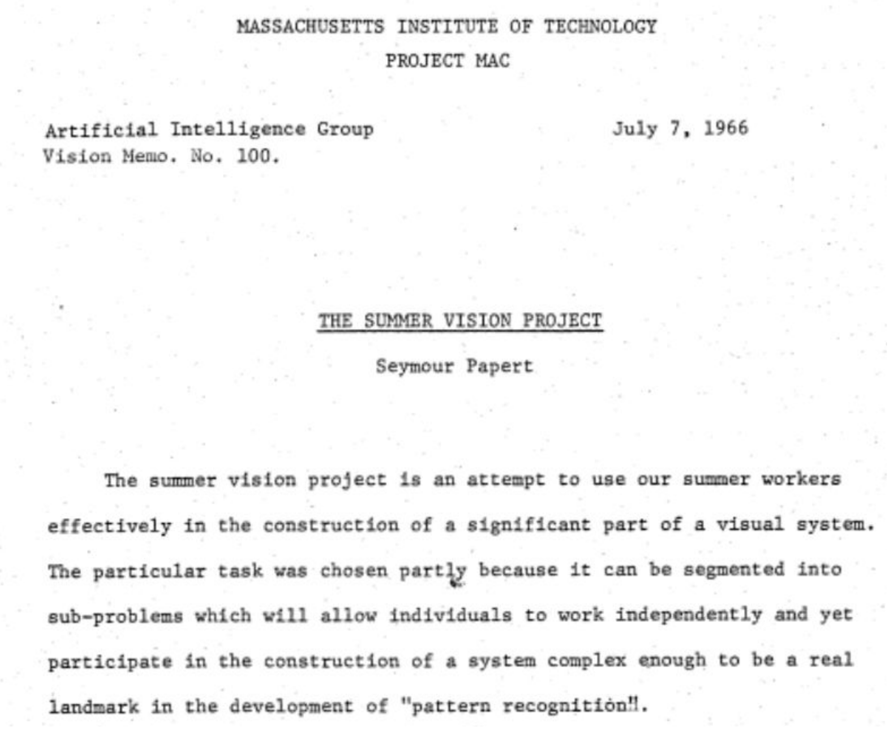
Image (or video) → **Sensing device** → **Interpreting device** → **Interpretation**

"A baby and a cat being extremely adorable"

- Computer Vision is then about finding the right algorithms for the interpreting device.

Easy peasy

- At first, Computer Vision seems like an easy problem to solve: “*we humans do it so easily since we were children*”!
- It seemed so easy that its study started with an MIT undergraduate **summer project** in 1966.
- The goal was very modest: find a model that rightly mimics our visual system.
- Unfortunately (or fortunately!) vision is much harder problem to solve and a summer wasn’t enough time.



MASSACHUSETTS INSTITUTE OF TECHNOLOGY
PROJECT MAC

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

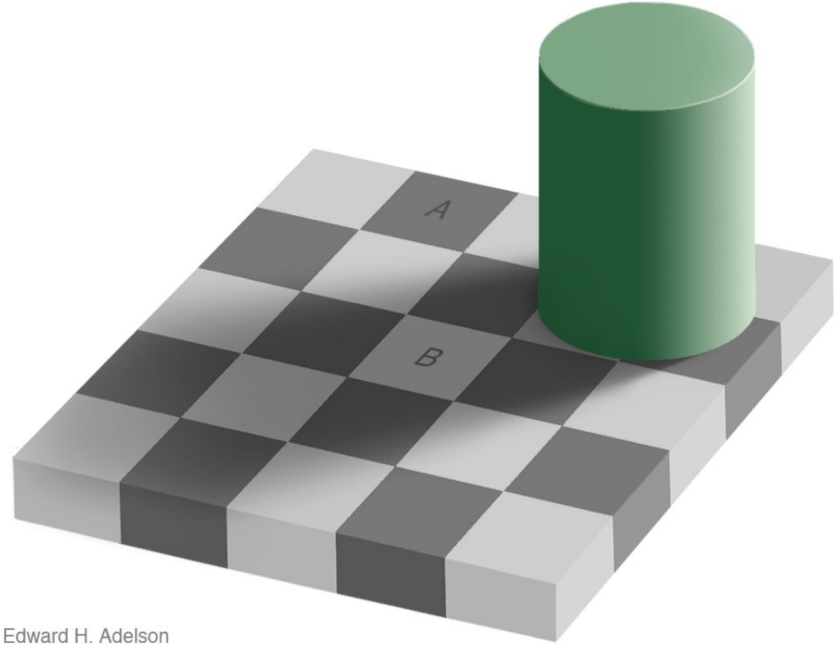
THE SUMMER VISION PROJECT

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".

Example of why vision is complicated

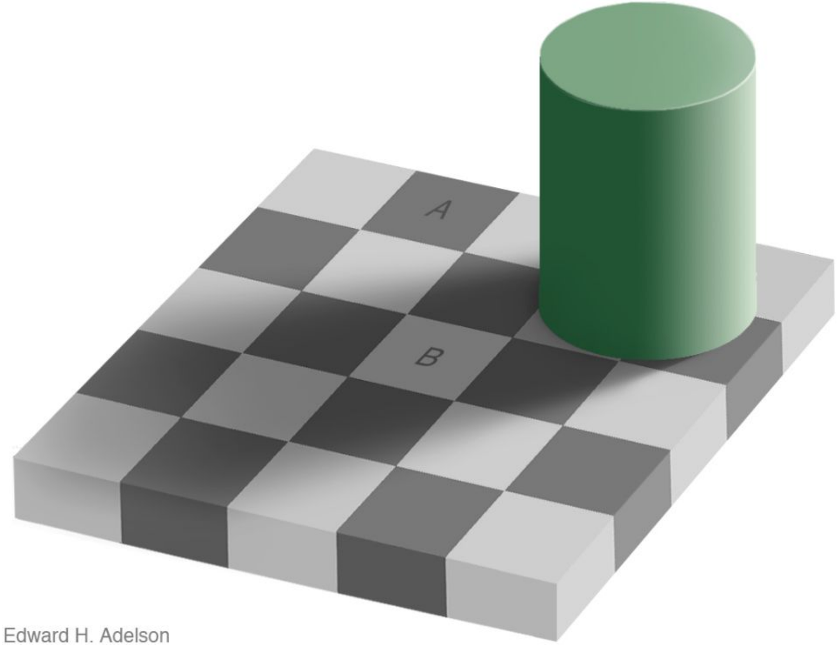
- Which region is darker in the image below, A or B?

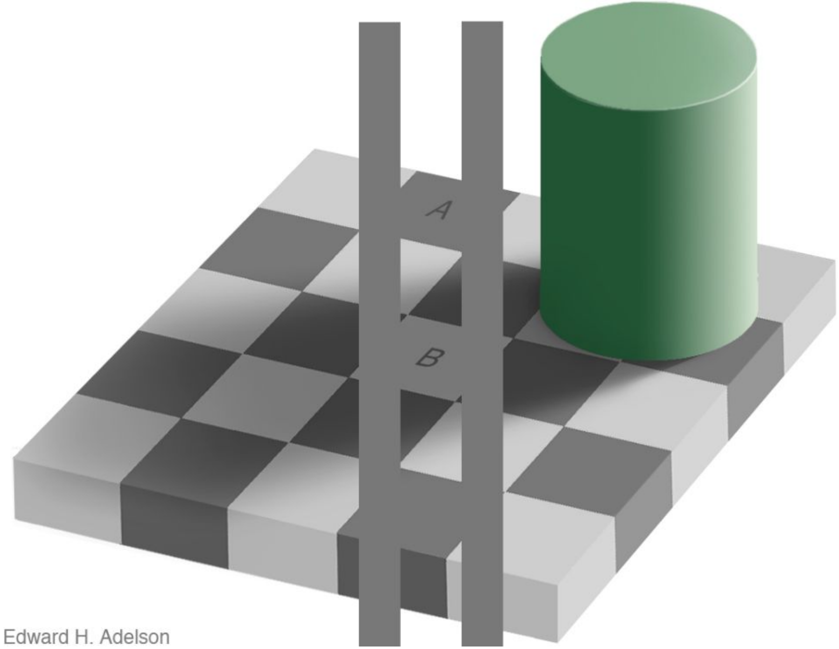


Edward H. Adelson

Example of why vision is complicated

- Believe it or not, but A and B are of the same shade of gray!



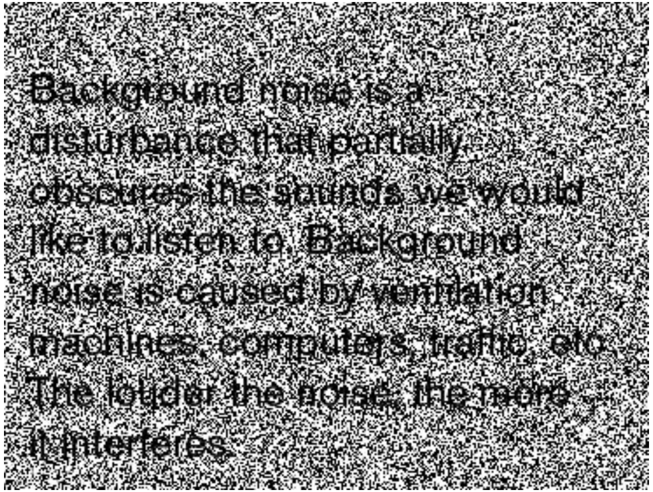


Exercise (*In pairs*)

- Answer the questions for the following images:

How many objects are there in this image?

Which object is the largest? What color is the sky?



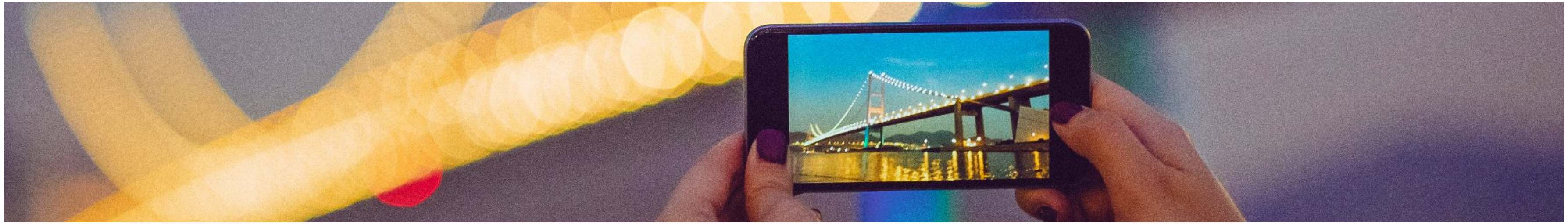
What is written in the above image?

Why vision is complicated

- Computer Vision is hard for many reasons, like:
  - **Our visual system is limited in some tasks**:
    - Our vision is prone to illusions that lead to misinterpretation of certain phenomena.
    - That raises the question of whether we want to necessarily mimic it using CV.
  - **Many vision problems are ambiguous**:
    - There is usually not an objective solution for some problems, as they may depend on human subjective reasonings.
    - That makes the modeling of those problems harder.
  - **Images remove important aspects of the observed scene:**
    - Usually the distance from the camera to the objects is unavailable, making the distance between the objects in a scene wrong.
    - Also, some color information may not match to what the real scene's colors are.
  - **The sensing device is imperfect and the images are noisy:**
    - It is usually impossible to capture a real world scene without adding noise to it.

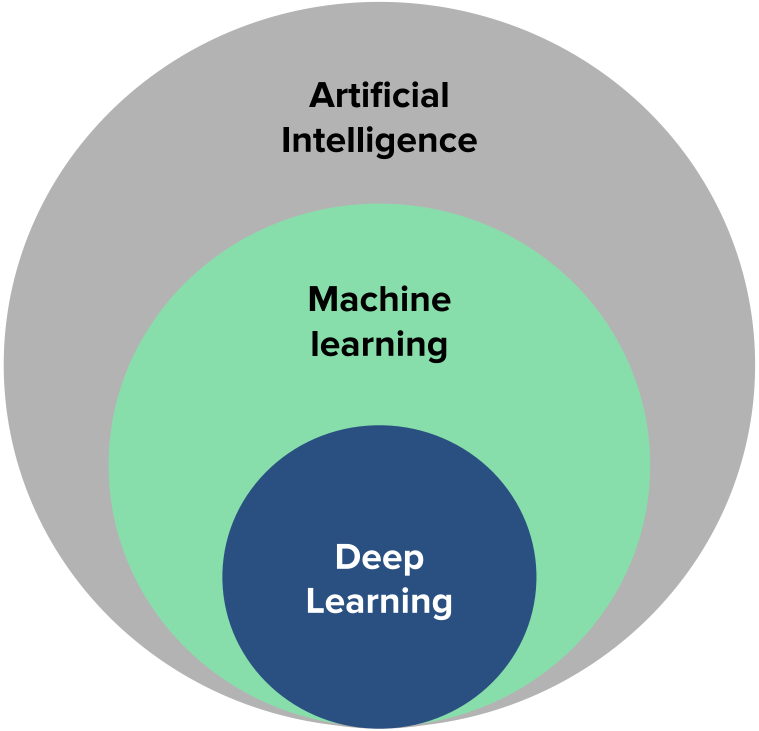
Why is Computer Vision so Important Now

- Despite these issues, technology is progressed at an exponential pace and computer vision solutions found numerous solutions and applications in modern society.
- This is due to the following factors:
  - **Computing Power**: Hardware designed for computer vision has become cheaper, faster, better and easily accessible.
  - **Big Data**: Availability of large training data sets due to mobile technology with built-in cameras saturating the world with photos and videos
  - **Open Source**: Availability of new algorithmic frameworks that can take advantage of hardware and software capabilities



Deep Learning and Computer Vision

- Largely, the great success of Computer Vision is due to recent use of **Deep Learning** as an “algorithmic framework” (the interpreting device).
- Based on Neural Networks, it a **supervised learning model** inspired by the way the brain propagates information.
- It was initially mainly used for image classification, it has been extended to **regression** and even **unsupervised tasks**.
- In fact, it has been extended to solve problems way beyond Computer Vision!



Deep Learning in the news

Deep learning helps predict traffic crashes before they happen

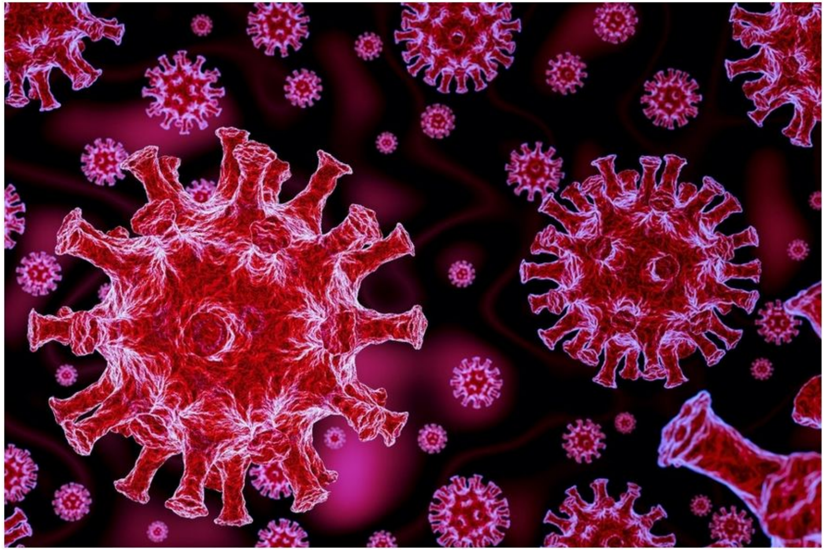
A deep model was trained on historical crash data, road maps, satellite imagery, and GPS to enable high-resolution crash maps that could lead to safer roads.

Rachel Gordon | MIT CSAIL
October 12, 2021

Deep learning helps predict new drug combinations to fight Covid-19

Neural network identifies synergistic drug blends for treating viruses like SARS-CoV-2.

Rachel Gordon | MIT CSAIL
September 24, 2021



Deep Learning in the news

OCTOBER 27, 2021

Discovering exoplanets using artificial intelligence

by University of Geneva

WILL KNIGHT BUSINESS 04.23.2021 07:00 AM

Now for AI's Latest Trick: Writing Computer Code

Programs such as GPT-3 can compose convincing text. Some people are using the tool to automate software development and hunt for bugs.

Machine learning refines earthquake detection capabilities

New methodology enables the detection of ground deformation automatically at a global scale

Date: November 11, 2021

How to do Deep Learning

- Over the years, many libraries in Python were implemented for developing applications using deep learning.
- Currently, the most famous ones (both open-sourced) are:

Tensorflow

- Created in 2015 by Google Brain department (Version 2.0 launched in 2019).
- Steep learning curve.
- Usually used in pair with Keras, an easy interface for it
- Large community, many tutorials.
- Use to be DL's main library.

Pytorch

- Developed in 2016 by Facebook's AI Research lab.
- Based on a library called Torch, written in Lua,
- Highly "Pythonic", easy to learn.
- Gives you more control.
- Small (but growing) community.
- Research preferable library.

- In our course, we'll use **PyTorch** and we'll have a lecture solely on it.

Applications of Computer Vision

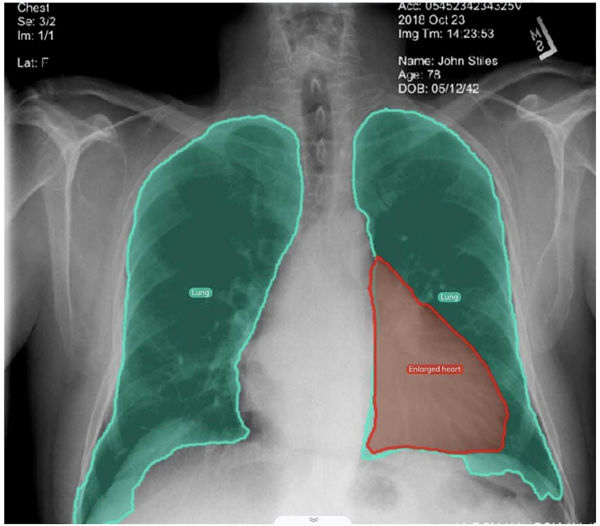
- **Shopping** will never be the same without Computer Vision!

Cashierless Shopping	Customer tracking and Flow	Gaze detection
		

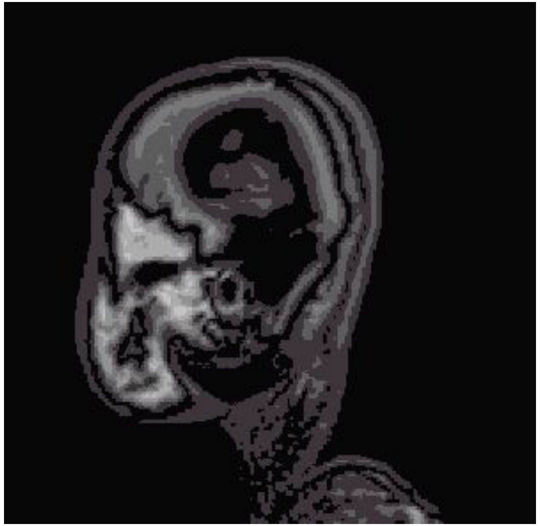
Applications of Computer Vision

- Without the Computer Vision, **healthcare** professionals would forced to spend hours manually analyzing patient data.

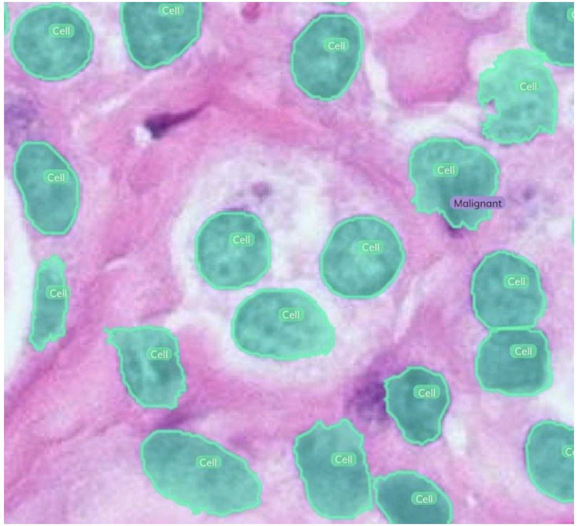
X-Ray image segmentation



MRI understanding



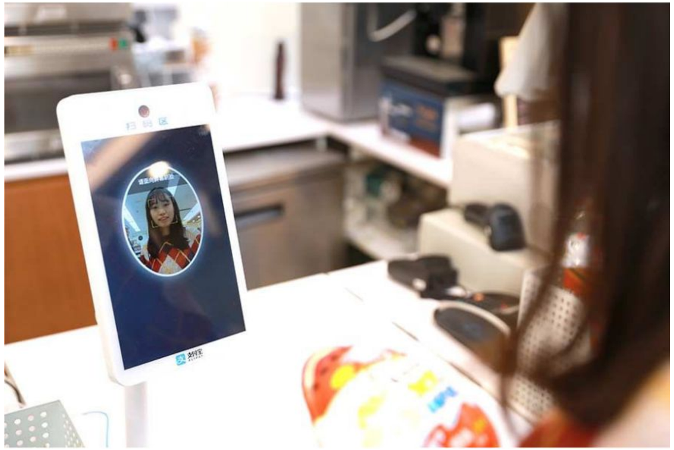
Pathology detection

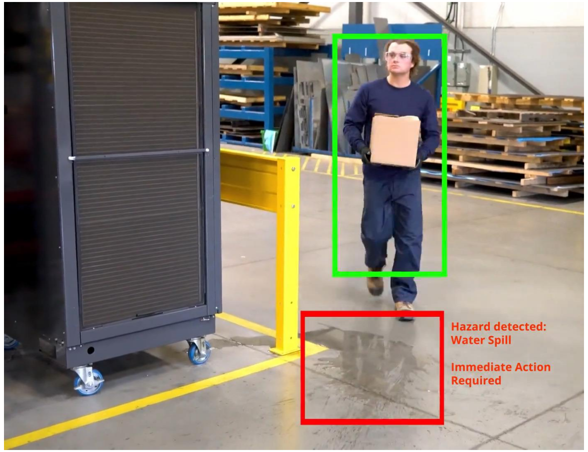


Applications of Computer Vision

- Computer vision enables many possibilities in **security and surveillance**!

Face Recognition	Hazard Detection	Crowd Counting/Control



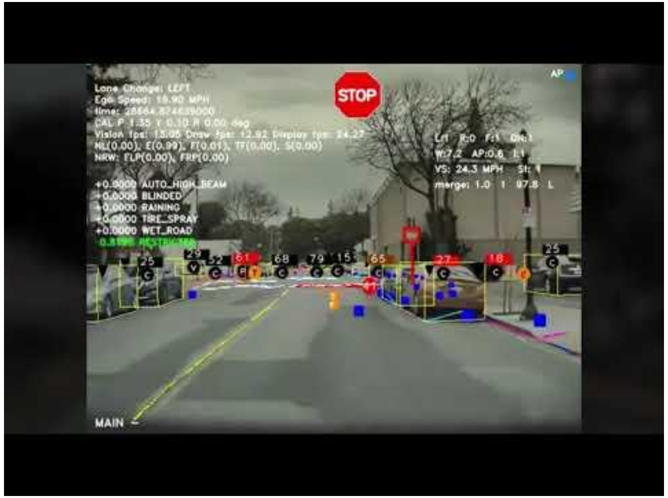


- Good news for public safety: helping police and first responders more easily spot crimes and accidents.

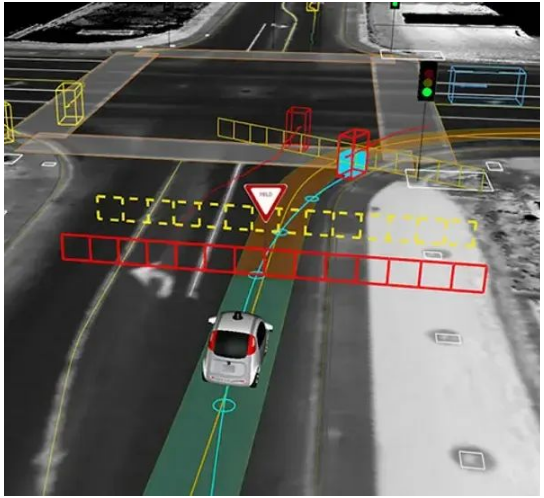
Applications of Computer Vision

- The future of **self-driving cars** depends on Computer Vision:

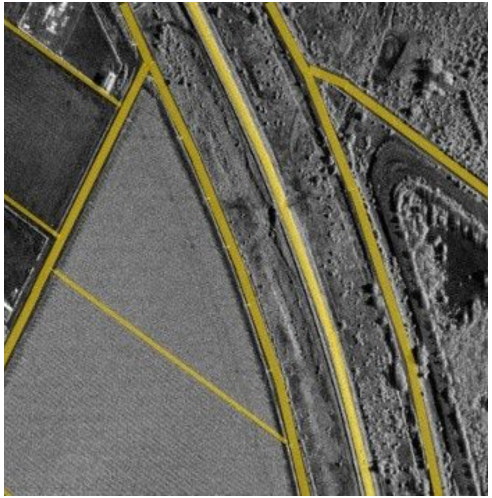
Car and pedestrian detection



Trajectory Planning



Road Detection



- And this all is already implemented on Tesla cars.

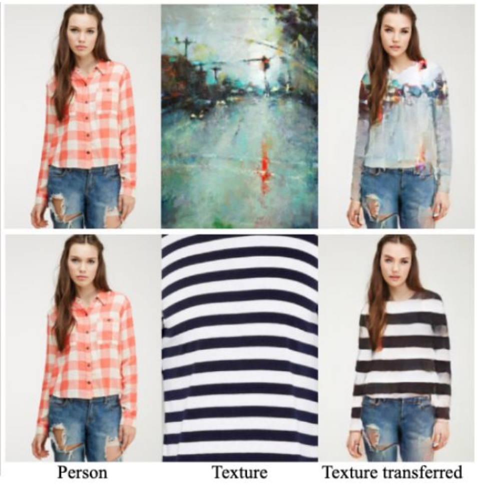
Applications of Computer Vision

- Also, many applications in the **fashion** industry:

Virtual Try-on

Garment Layering

Clothing editing



Source and more: cuiaiyu.github.io/dressing-in-order/

(Tentative) Lecture Roadmap

Basics of Deep Learning

1. Intro to Computer Vision
2. Linear Classifiers and Perceptron
3. Multilayer Perceptron and Intro to Deep Learning
4. Optimization and Regularization
5. Pytorch
6. Convolutional Neural Networks
7. Data Augmentation and Deep CNNs
8. Transfer Learning and Residual Nets
9. Inception Net and what CNNs learn
10. Adversarial Examples and Self-supervision

Computer Vision Tasks

1. Intro to Object Detection
2. Fast Object Detection
3. Intro to Image Segmentation
4. Applications of Detection and Segmentation
5. Image Manipulation and Autoencoders
6. Image Denoising and Style Transfer
7. Image Generation with GANs
8. Advanced GANs
9. Image Generation by Prompt

Video: *The Deep Learning Revolution*