Computer Vision

Dr. Nabeel Mohammed

Course Information

Course Code: CSE553/EEE660/CSE468

Course Title: Computer Vision

Credits: 3

Faculty: Dr. Nabeel Mohammed (NbM)

Email: nabeel.mohammed@northsouth.edu

Class Time: Tuesday 7pm - 10pm

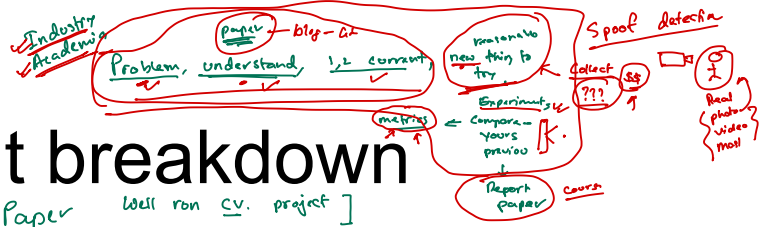
Assessment breakdown

Project - 40%

Assignments - 20%

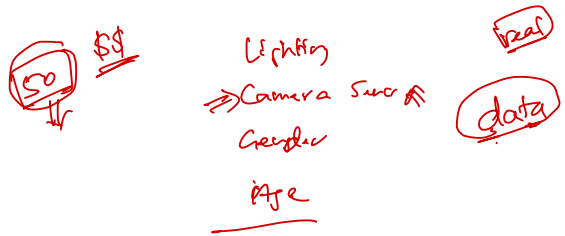
Midterm - 20%

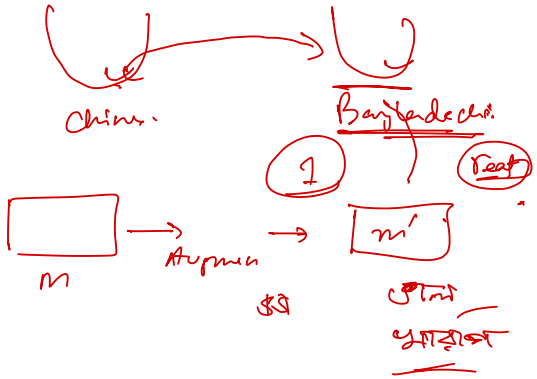
Final - 20% (Extra questions for MS Students)



(Group Project. Between 3-4 members)

Spoof





Prerequisite knowledge

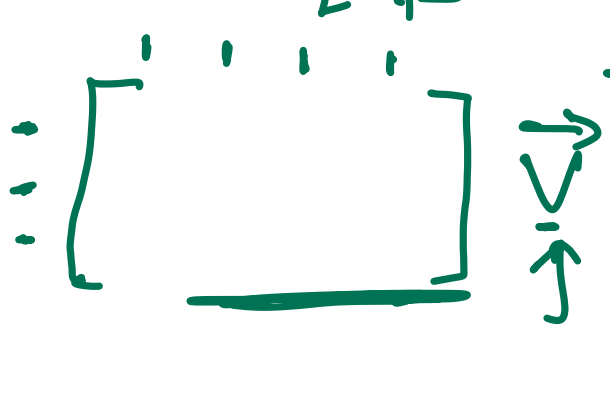
- Programming
- Data Structures
- Linear Algebra
- Calculus of multiple variables
- Some machine learning will be useful
  - First three chapters of http://neuralnetworksanddeeplearning.com/
  - Terminologies: Supervised, Unsupervised, Accuracy, Train-Validation-Test slit, Regression, Classification, Mini-batch, loss/error function, backpropagation etc..
- Opencv (or other image processing libraries)
- PyTorch

Math

Scalar value (Single number) $\mathbb{R}$

Vectors more than one number $\mathbb{R}^d$
(Magnitude, Direction)

$$\vec{v} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix} \quad \mathbb{R}^4 \qquad \vec{v}^T = [v_1\ v_2\ v_3\ v_4]$$



$$\vec{v} = \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}$$

p

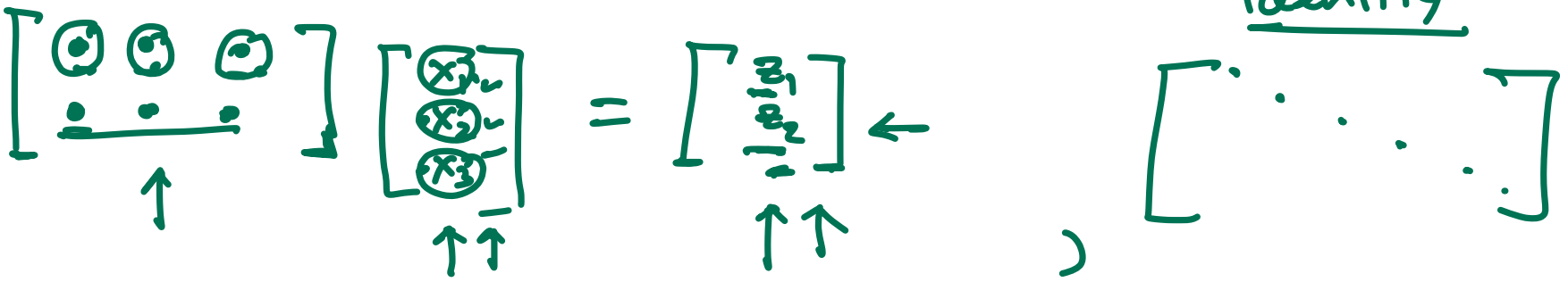
k

p cols
k rows

I

Matrix

identify



$$\vec{v} = \begin{bmatrix} 2 \\ 1 \\ 4 \end{bmatrix}$$

L_2-norm $\|\vec{v}\|_2 = ?$

L_1-norm $\|\vec{v}\|_1 = ?$

$$L_2\text{-norm} = \sqrt{2^2 + 1^2 + 4^2} = \sqrt{4+1+16} = \sqrt{21}$$

$$L_1\text{-norm} = \sqrt{121 + 111 + 141} = \sqrt{2+1+4} = \sqrt{7}$$

$\vec{v}$, $\|\vec{v}\|_2 = \sqrt{21}$

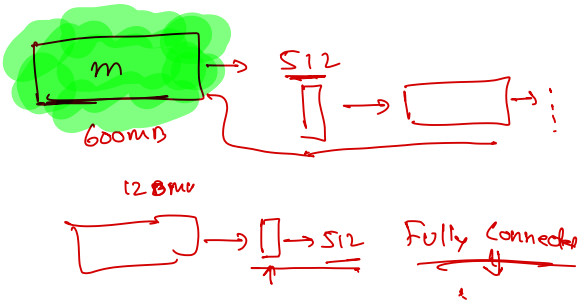
$$\left\| \frac{\vec{v}}{\|\vec{v}\|_2} \right\|_2 = 1$$

$$\frac{\vec{v}}{\|\vec{v}\|_2} = \begin{bmatrix} 2/\sqrt{21} \\ 1/\sqrt{21} \\ 4/\sqrt{21} \end{bmatrix}$$

$$\sqrt{\left(\frac{2}{\sqrt{21}}\right)^2 + \left(\frac{1}{\sqrt{21}}\right)^2 + \left(\frac{4}{\sqrt{21}}\right)^2}$$

$$= \sqrt{\frac{4}{21} + \frac{1}{21} + \frac{16}{21}}$$

$$= \sqrt{\frac{21}{21}} = 1$$



$$\vec{v} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix}$$

$$W\vec{v} = \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}$$

W —
Col — 4
Row - 3

$$W = \begin{bmatrix} \vec{w}_1^T \\ \vec{w}_2^T \\ \vec{w}_3^T \end{bmatrix}$$

$$\vec{w}_1 = \begin{bmatrix} w_{1,1} \\ w_{1,2} \\ w_{1,3} \\ w_{1,4} \end{bmatrix}$$

$$t_1 = \text{dot}(\vec{w}_1, \vec{v})$$

$$= \|\vec{w}_1\| \, \|\vec{v}\| \cos\theta$$

$$= \cos\theta$$

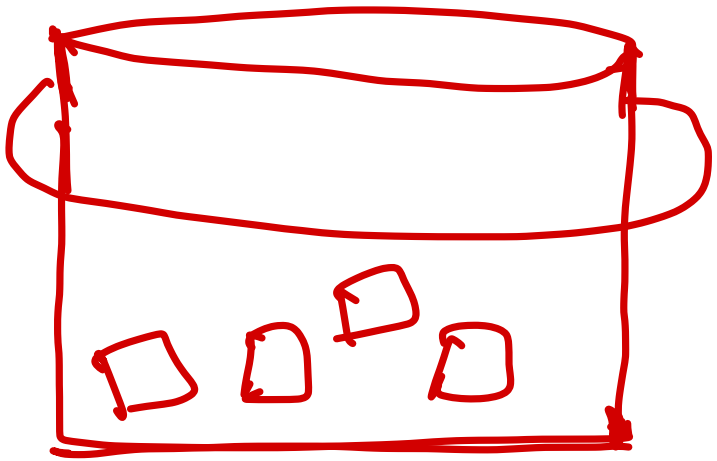
(θ is the angle between $\vec{w}, \vec{v}$)

$\|\vec{w}_1\| = 1$

$\|\vec{v}\| = 1$

$$-1 \le t_1 \le 1$$

Probability



০, ১, ২, ৩,
৪, ৫, ৬, ৭, ৮
৯

P.D has 10 probability values

$p_1 \cdots\cdots p_{10}$

$$0 \leq p_i \leq 1$$

$$\sum_{k=1}^{10} p_k = 1$$

15 kg – 0.2

60 kg – 0.5

90 kg – 0.3

Average weight = $\frac{15*0.2 + 60*0.5 + 90*0.3}{60}$

= 60

15 kg – 2

60 kg – 5

90 kg – 3

Average Weight = $\frac{(15+15+60+60+60+60+60+90+90+90)}{10}$

$= \frac{15*2 + 60*5 + 90*3}{10}$

$= \frac{2}{10}.15 + \frac{5}{10}.60 + \frac{3}{10}.90$

Computer Science

$$k = 2x^2 + 3x \qquad (x = 2)$$

$$= 2.(4) + 3.(2)$$

$$= 8 + 6 = 14.$$

$$\frac{dk}{dx} = 4x + 3$$

$$= 4.2 + 3$$

$$= 11$$

$$t = k^2 + 4 \quad = 14^2 + 4$$

$$= 196 + 4 = 200$$

$$\frac{dt}{dk} = 2k = 2.14 = 28.$$

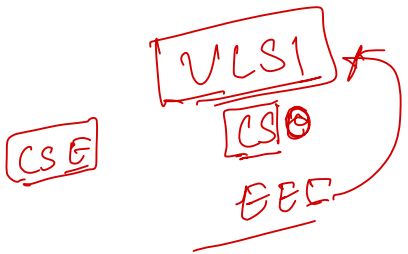
$$\frac{dt}{dx} = \frac{dt}{dk} \cdot \frac{dk}{dx} = 28 \times 11 = 308$$

Topics for this course (this semester)

- **Classification**
  - **This is crucial for understanding the rest**
- Detection
- Segmentation
- Transfer learning
- Adversarial Samples
- Generative Models
  - Generative Adversarial Networks
- Open Set Problems
- Search
- Other interesting topics (if time permits)

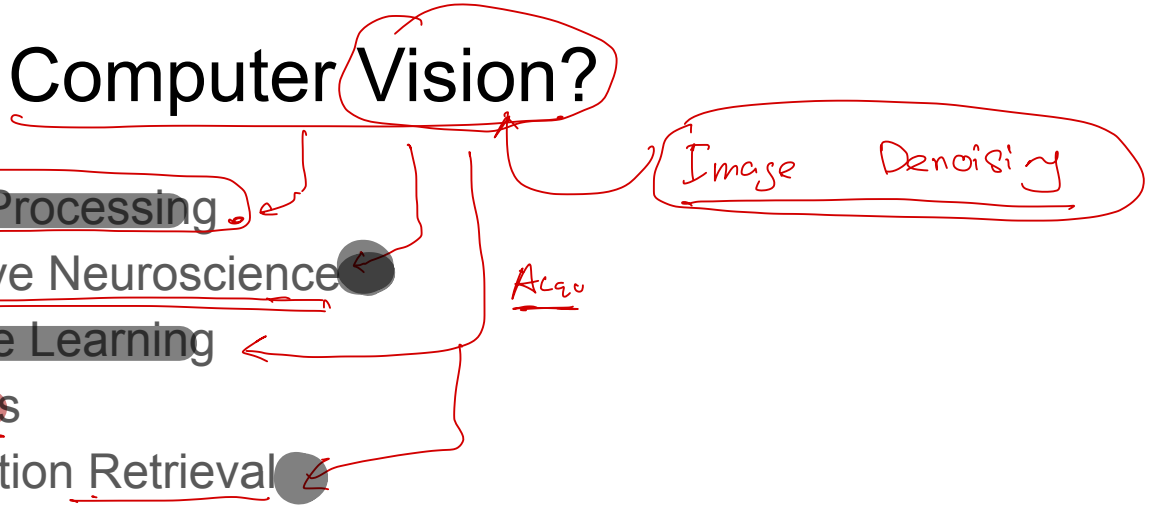
This Lecture

- What is computer vision?
  - How is it separate from image processing?
- History
- Image classification
  - Naive non-parametric approach



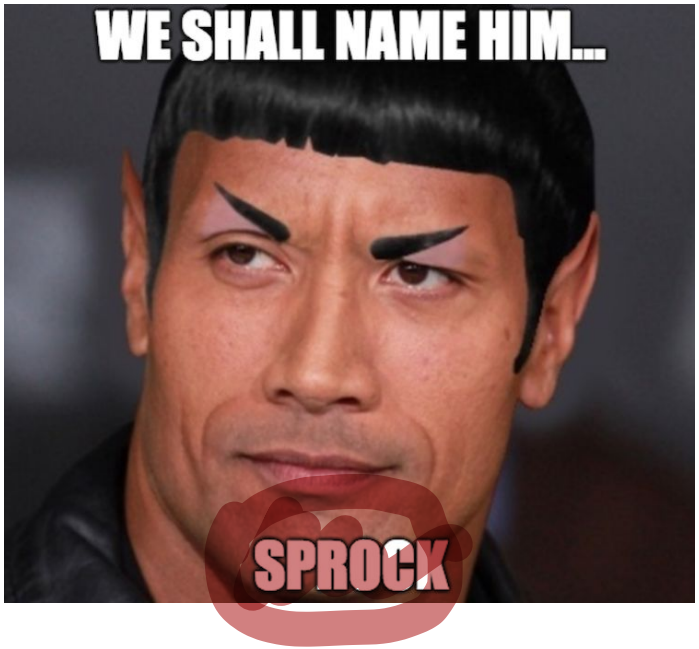
What is Computer Vision?

- Image Processing
- Cognitive Neuroscience
- Machine Learning
- Robotics
- Information Retrieval
- Graphics
- Algorithms
- Mathematics
- And much more



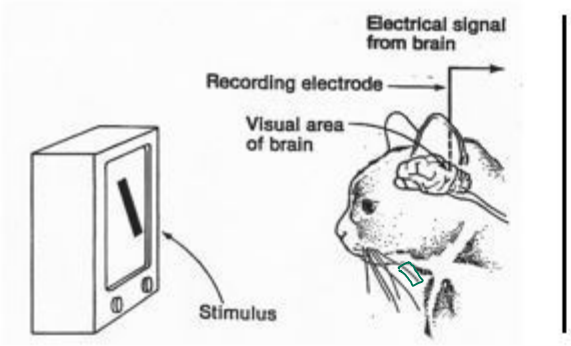
Consider this image

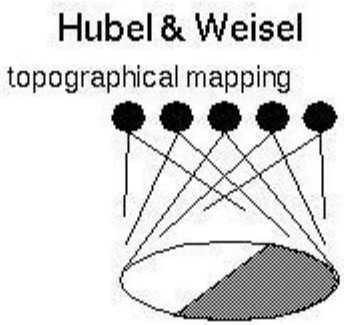
Consider this image

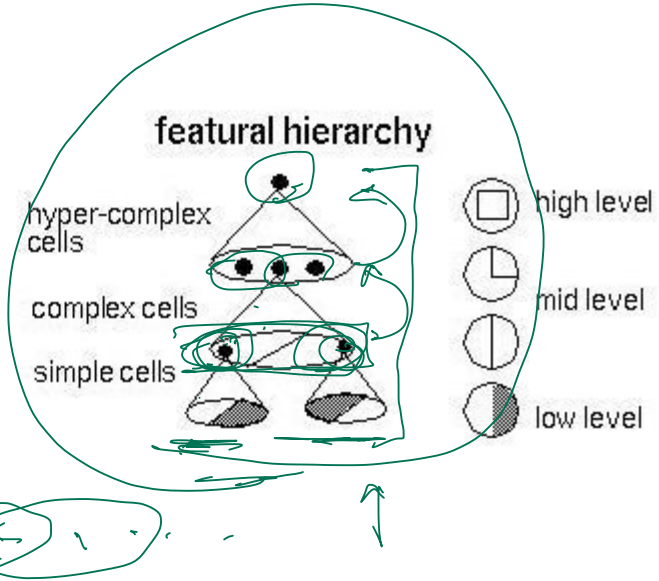


Consider this image

History

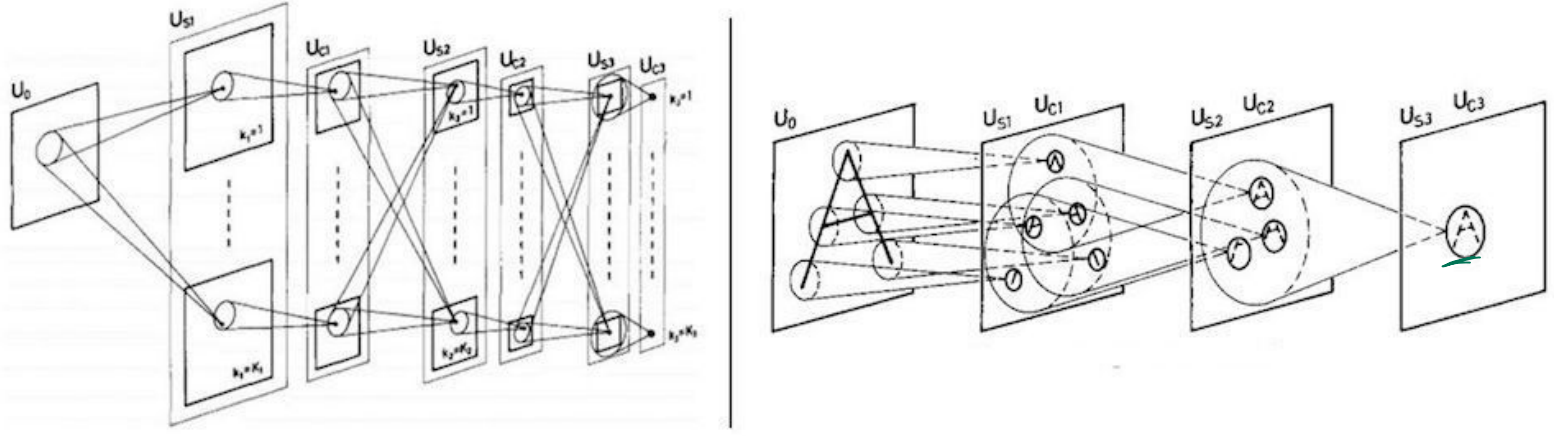

Electrical signal
from brain
Recording electrode
Visual area
of brain
Stimulus


Hubel & Weisel
topographical mapping


featural hierarchy
hyper-complex
cells
complex cells
simple cells
high level
mid level
low level

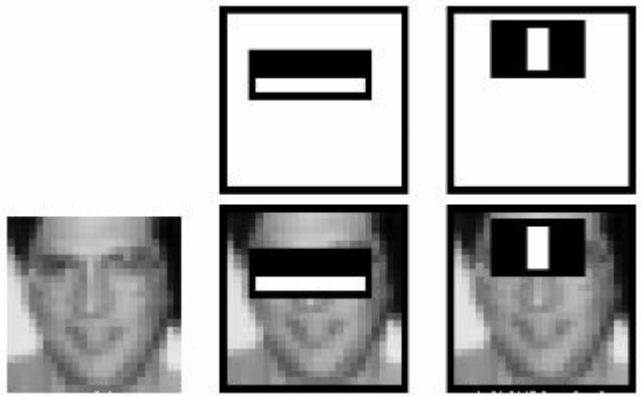
History

Fukushima's Neocognitron (1982)



History

Viola-Jones 2001

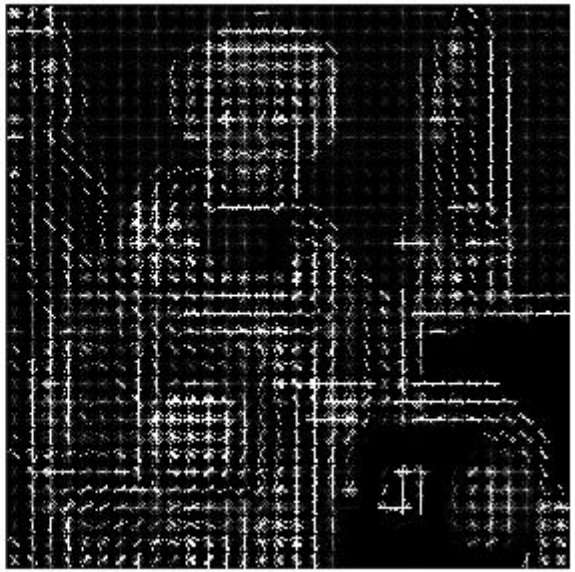


History

Histogram of Oriented Gradients, Dalal and Triggs (2005)

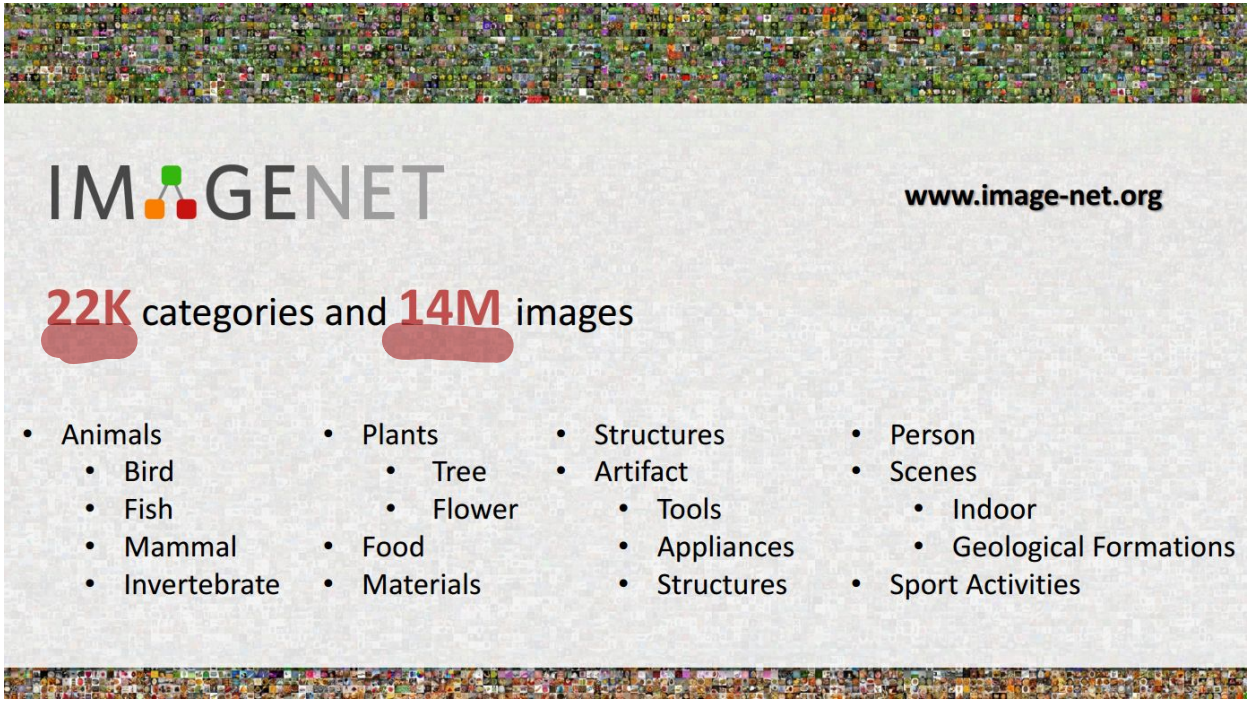
Input image

Histogram of Oriented Gradients



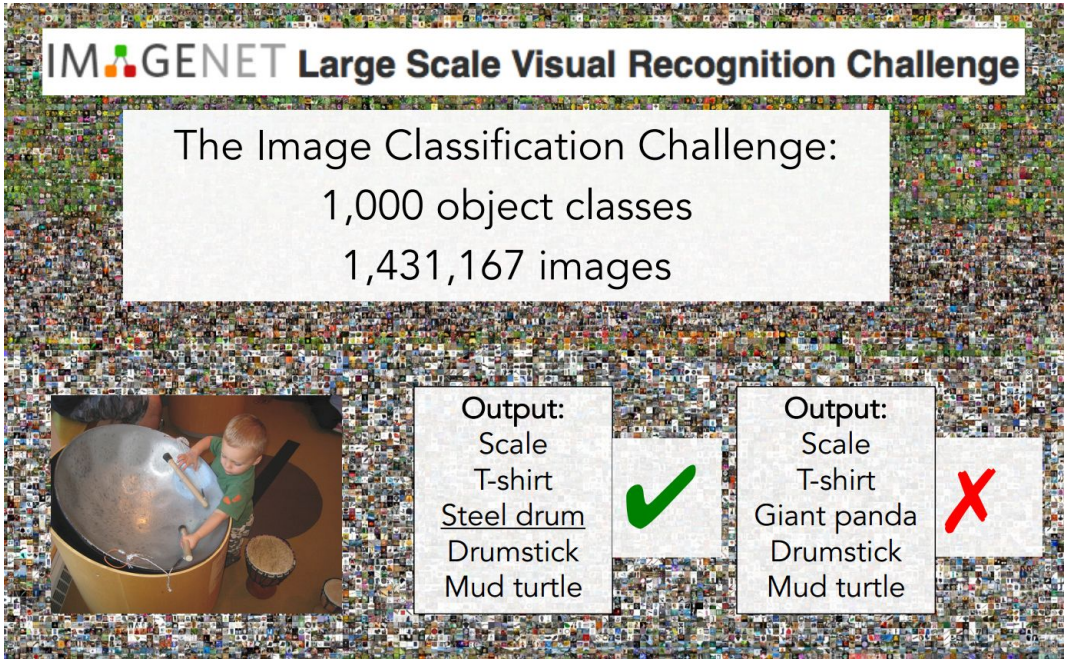
History

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

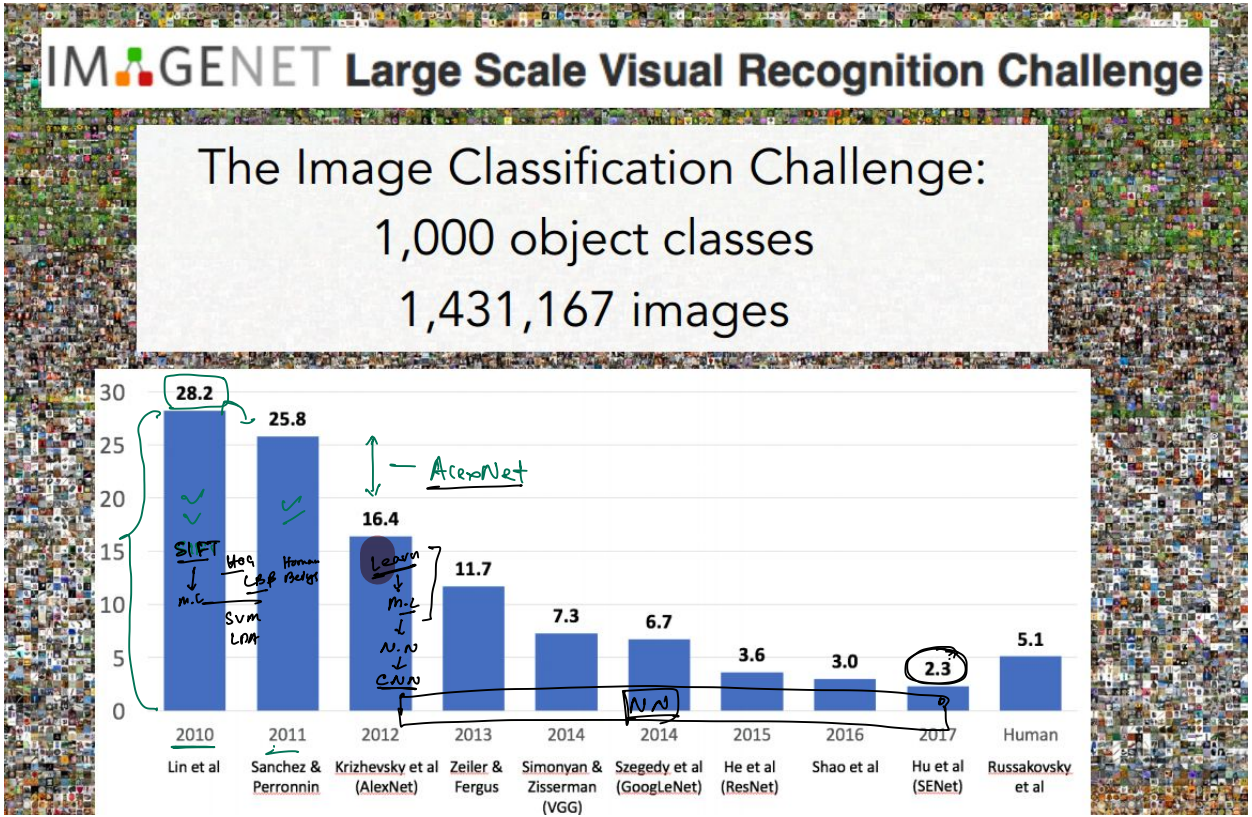


History

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



History - Context



Historical perspective

For our purpose we will simplify into two parts

Pre-2012

- Hand crafted features used with classifiers/search modules

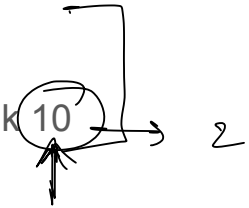
Post-2012

- Learn features
  - Feature engineering is less of a concern, network engineering seems to be more useful

NN

In this course

- We will mostly concentrate of Deep Learning-based architectures **and techniques.**
- Initially we will look at a non-parametric approach
- Also, some non-deep learning techniques
  - These are good to know for completeness of knowledge
- We will cover theory in course
- You will need to pick up a Deep Learning framework
- Pick a **suitable** project
  - 3-4 member teams
  - Paper must be ready by week 10



Project

- Fit your computational budget (capacity)
  Google Colab ☺ / Kaggle Kernels

- Do not attempt to collect your data.].

- Where there is scope for improvement.

- Follow (initially) your mentor's direction.

Image Classification

Image Classification

- 

Wicket

Image Classification

- Is a core part of Computer Vision
- Extremely Challenging
- Semi formalised version is only slightly less challenging
  - Given a finite set of labels **L = {l_1, l_2, l_3...l_n}** and an Image *M,* ascribe to *M* one or more labels from **L**
  - We are not worrying about the “do not know” class