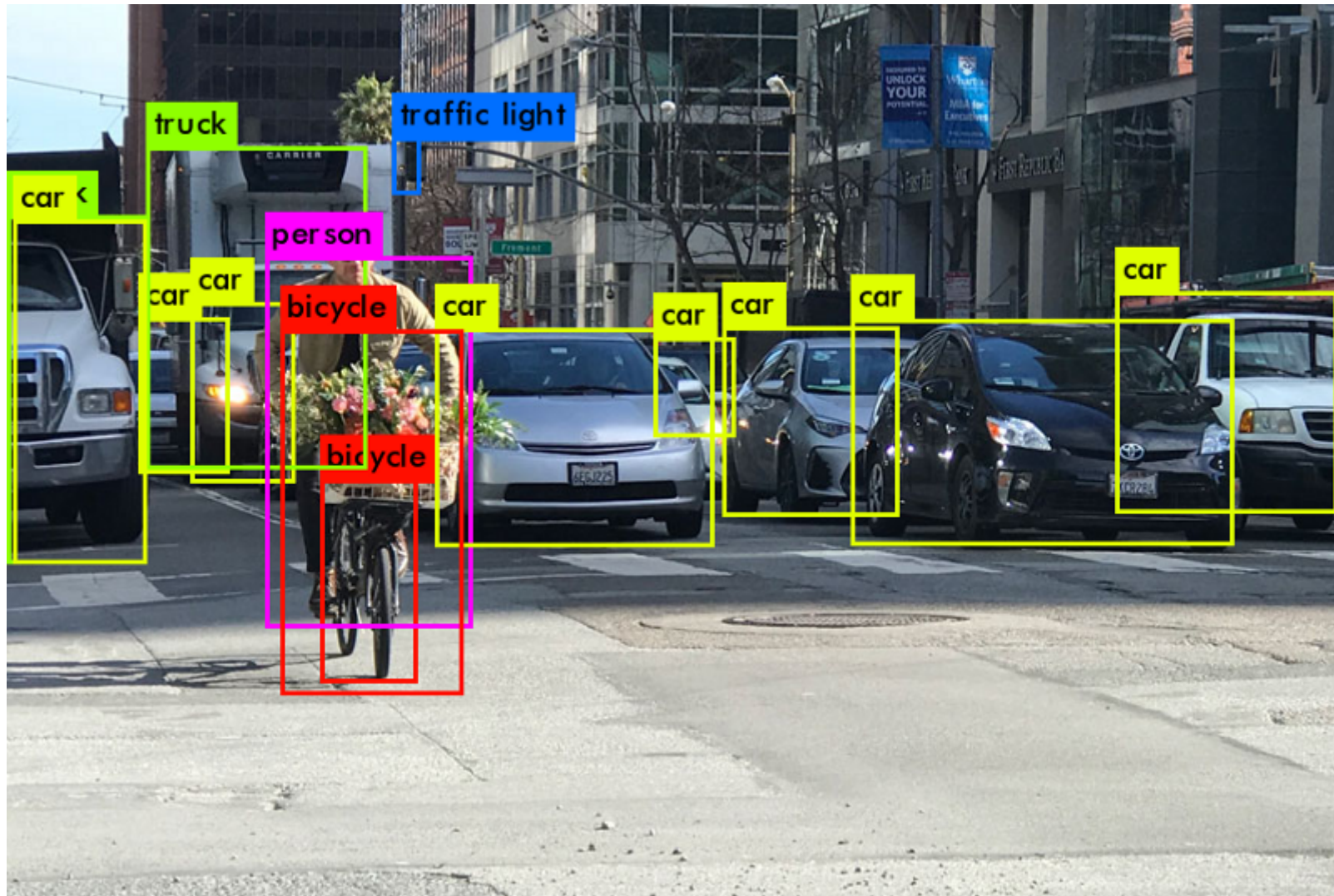


Deep Computer Vision



Who am I?



- PhD student – astronomy, instrumentation.
- MOOC junkie.
- Deep into deep learning.
- Offering training, workshops on DL.

Vikram Mark Radhakrishnan
radhakrishnan@strw.leidenuniv.nl

<https://github.com/VikramRadhakrishnan/DeepCV>

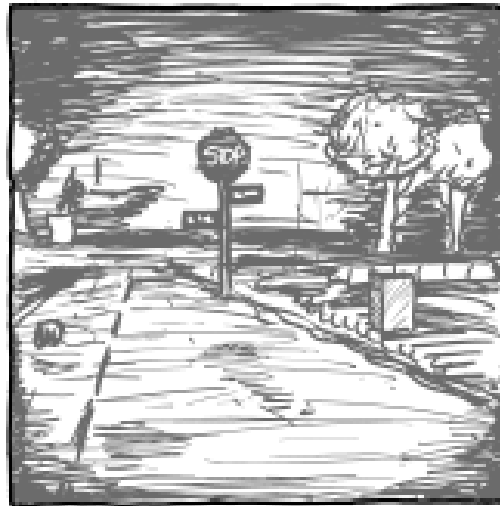
<https://www.kaggle.com/c/digit-recognizer/kernels>

Make computers see what we see



“Get” the picture

TO COMPLETE YOUR REGISTRATION, PLEASE TELL US
WHETHER OR NOT THIS IMAGE CONTAINS A STOP SIGN:

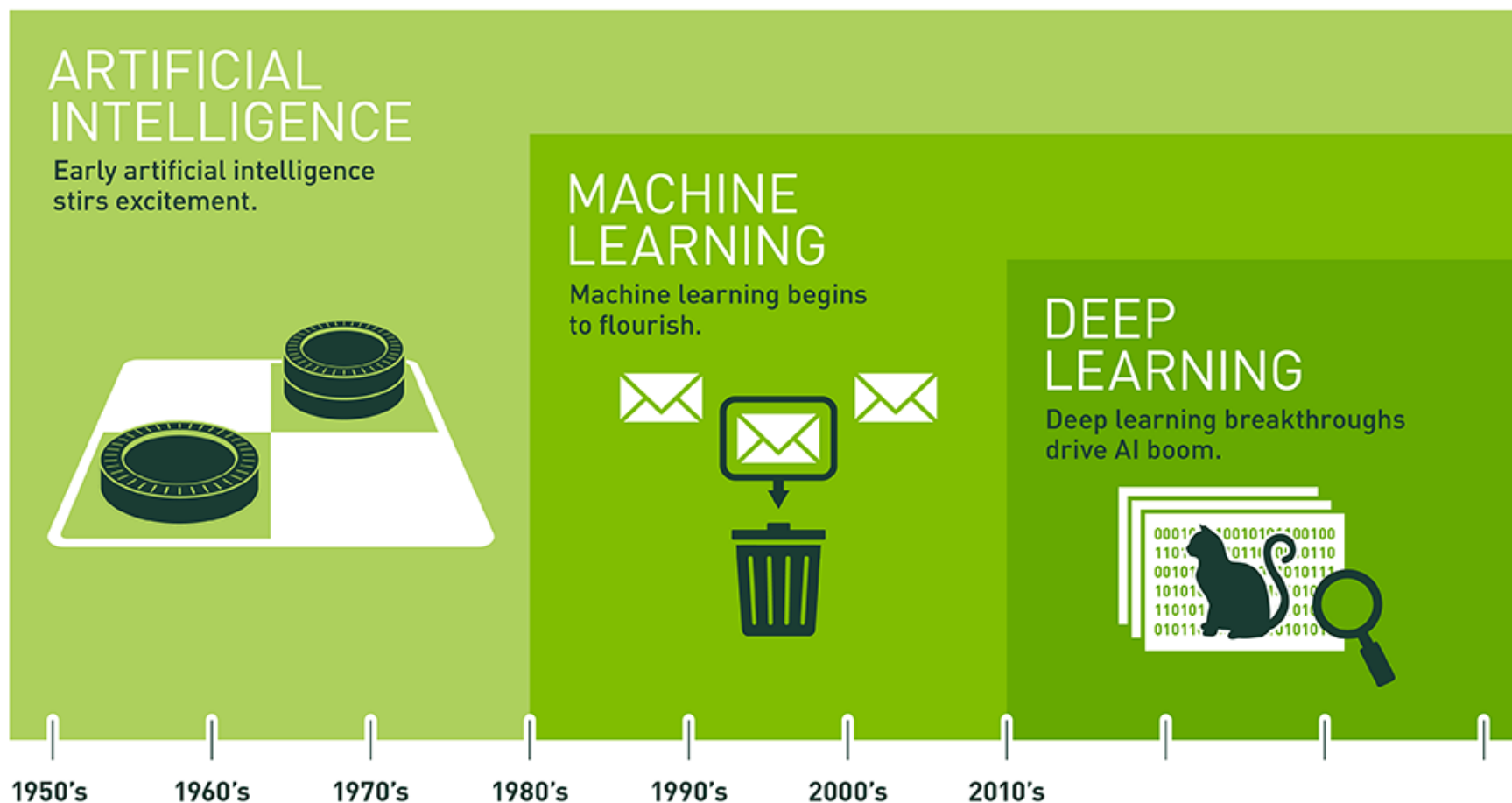


NO YES

ANSWER QUICKLY—OUR SELF-DRIVING
CAR IS ALMOST AT THE INTERSECTION.

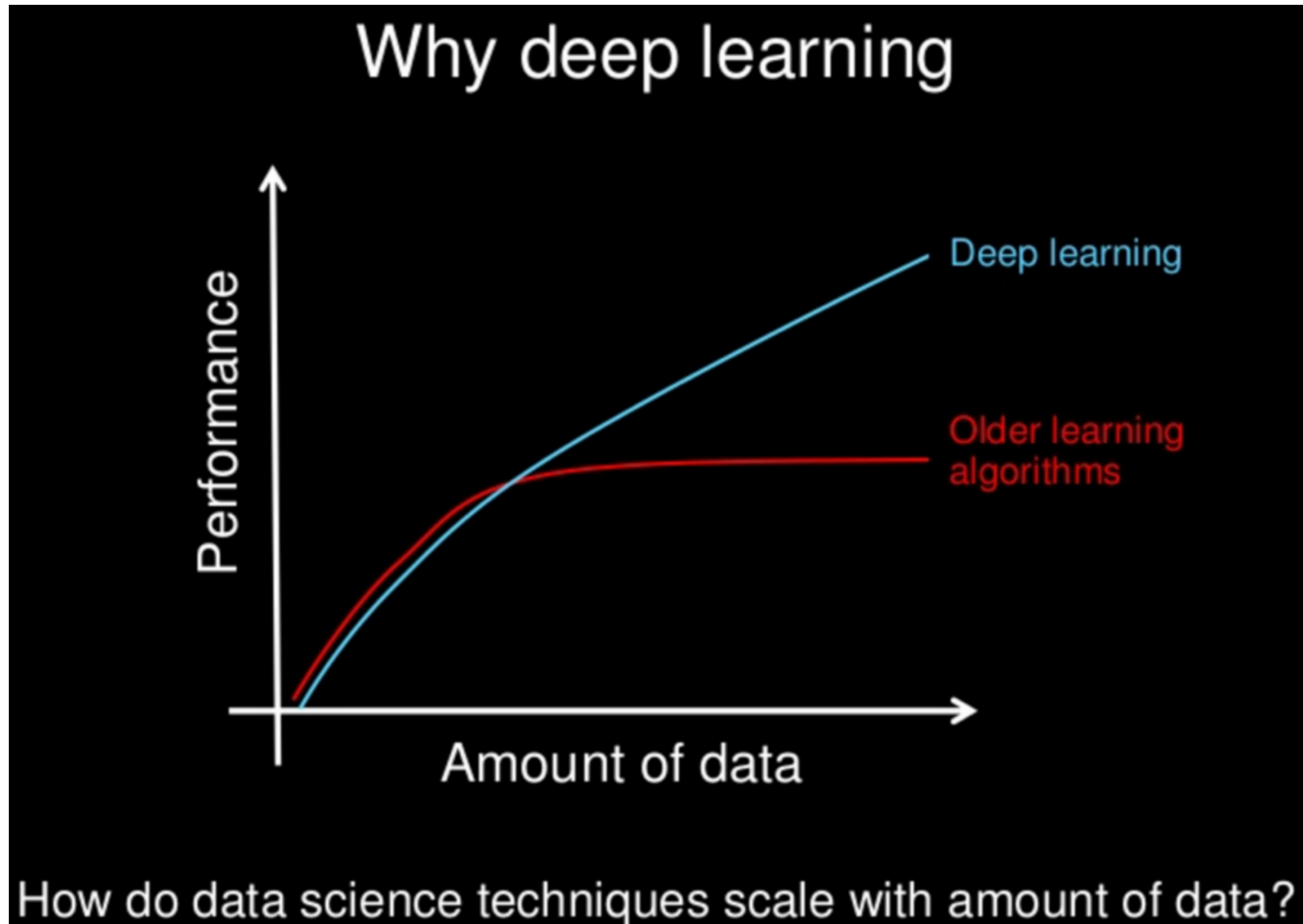
SO MUCH OF “AI” IS JUST FIGURING OUT WAYS
TO OFFLOAD WORK ONTO RANDOM STRANGERS.

A.I. can be our visual cortex

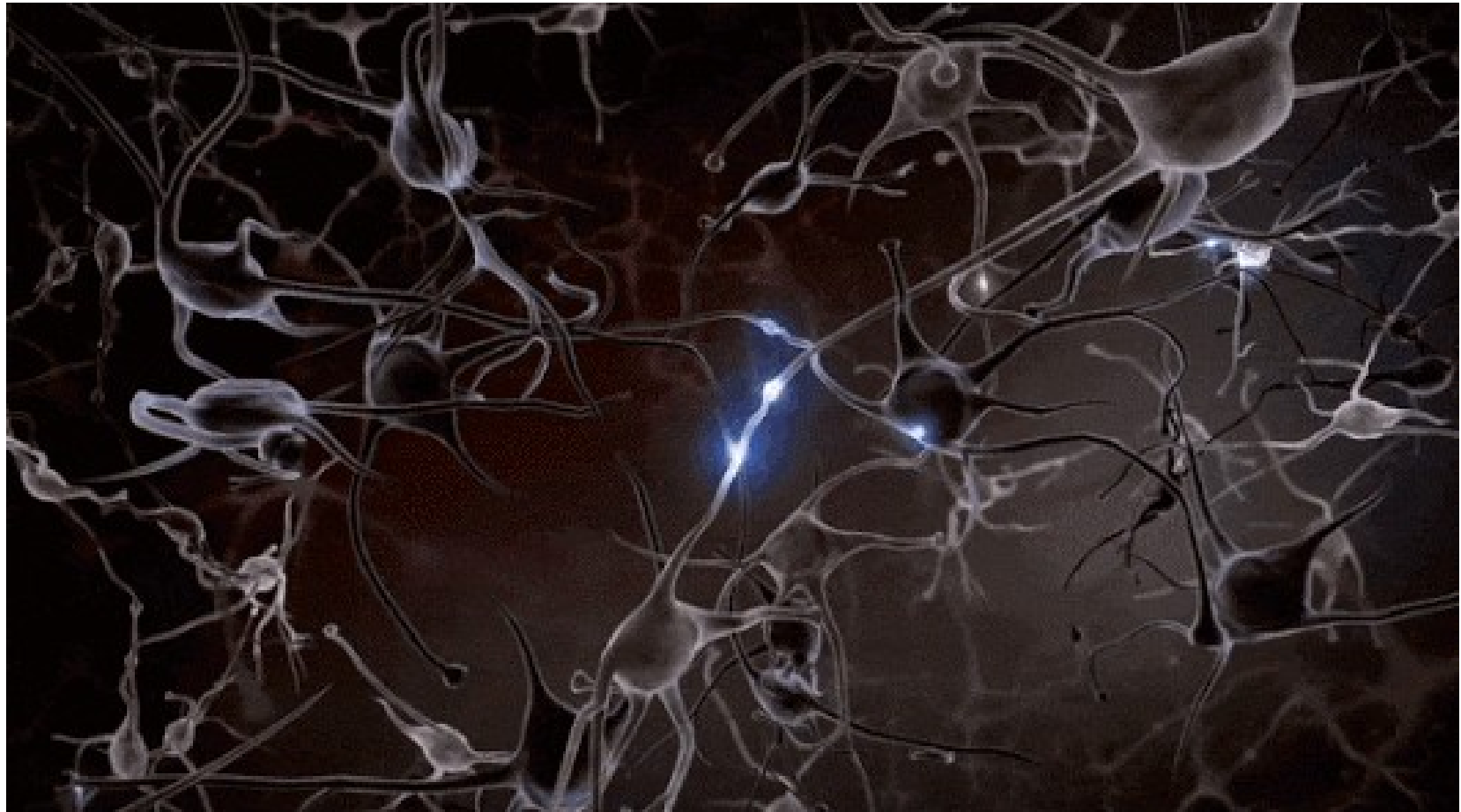


Since an early flush of optimism in the 1950s, smaller subsets of artificial intelligence – first machine learning, then deep learning, a subset of machine learning – have created ever larger disruptions.

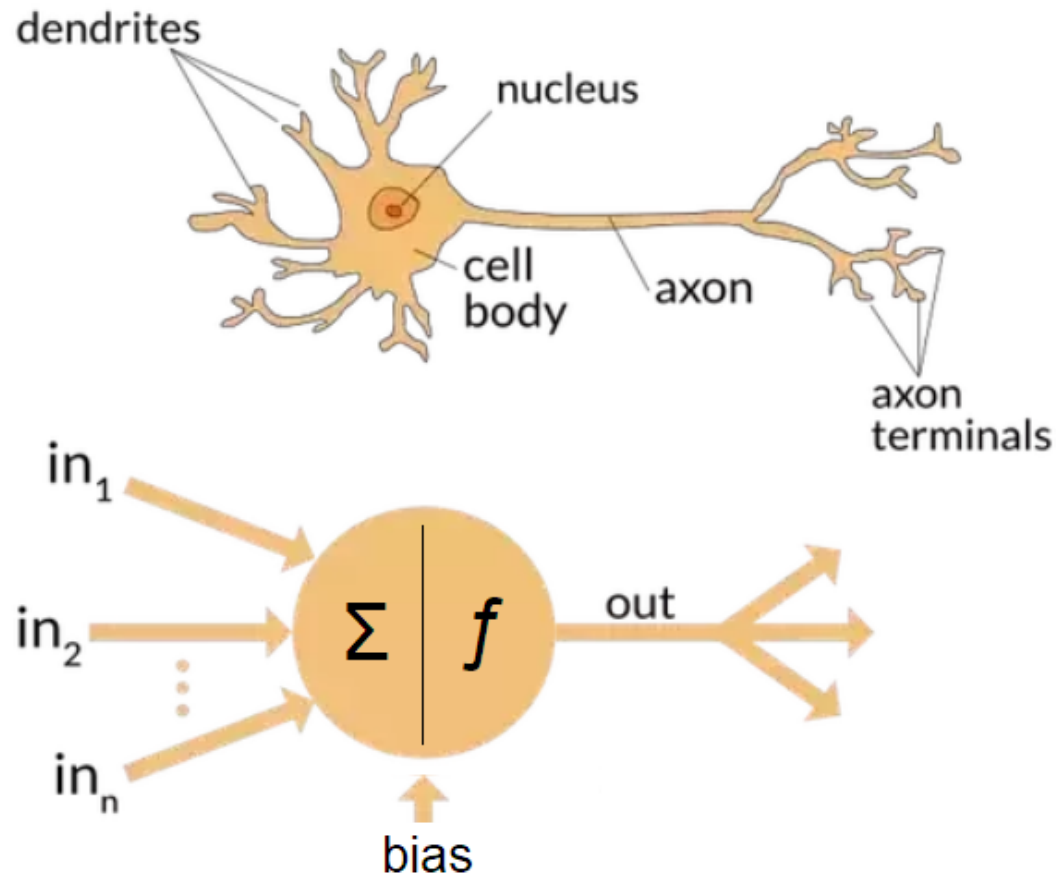
The dawn of deep learning



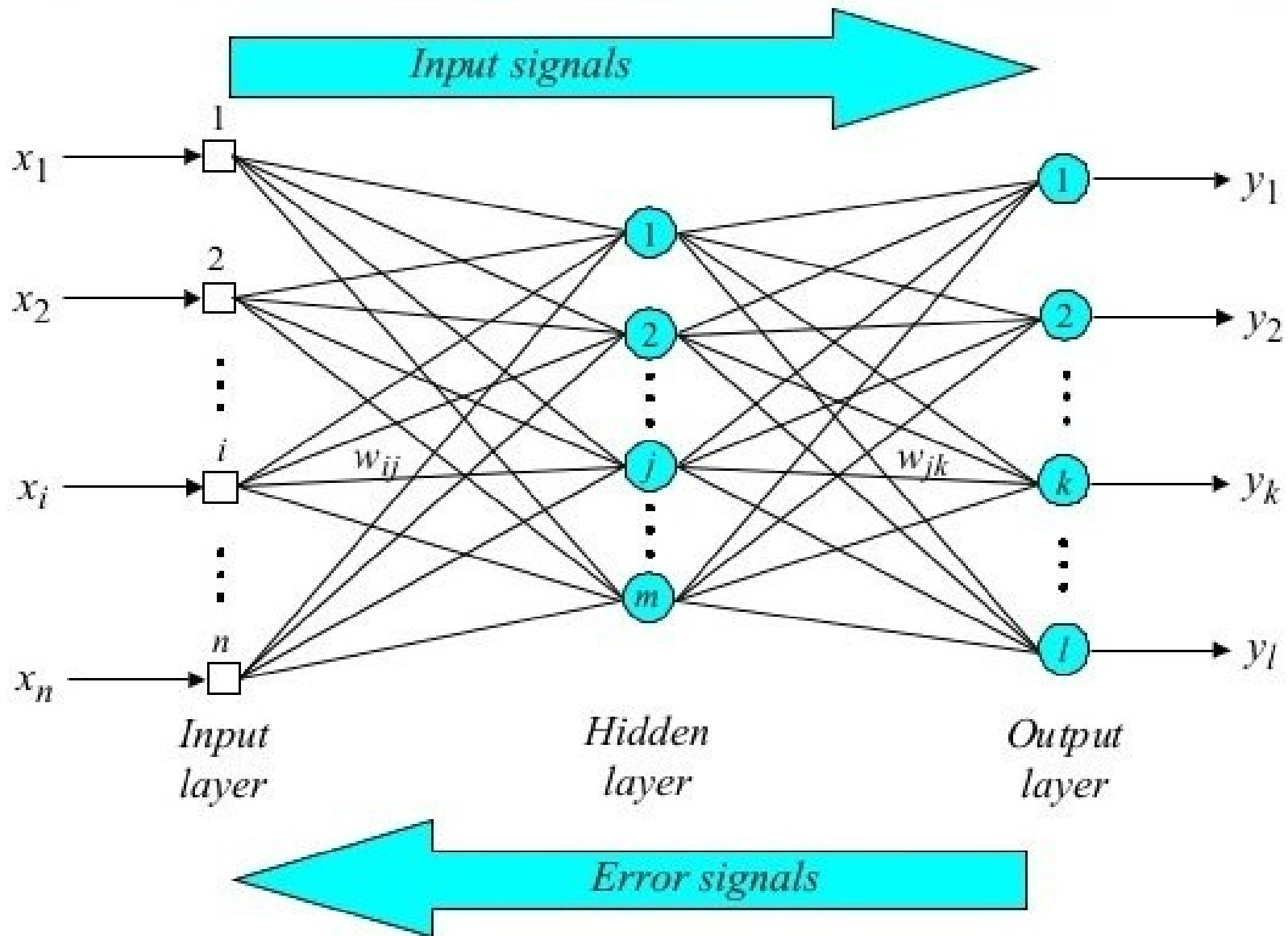
Based on biology



Neural networks - Home grown and in the wild



Fully connected neurons

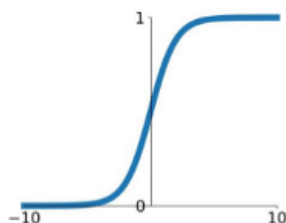


Activation functions

Activation Functions

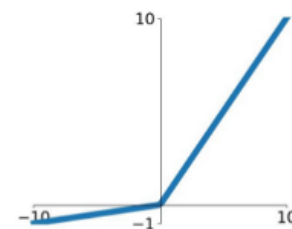
Sigmoid

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



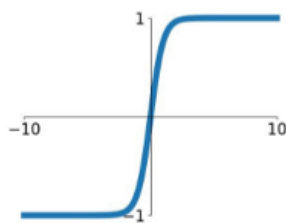
Leaky ReLU

$$\max(0.1x, x)$$



tanh

$$\tanh(x)$$

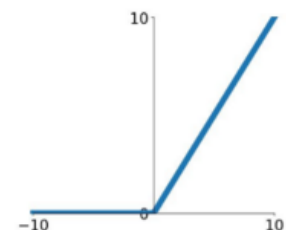


Maxout

$$\max(w_1^T x + b_1, w_2^T x + b_2)$$

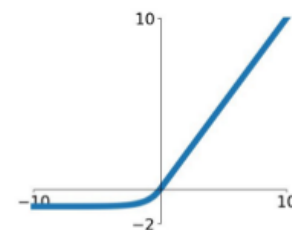
ReLU

$$\max(0, x)$$

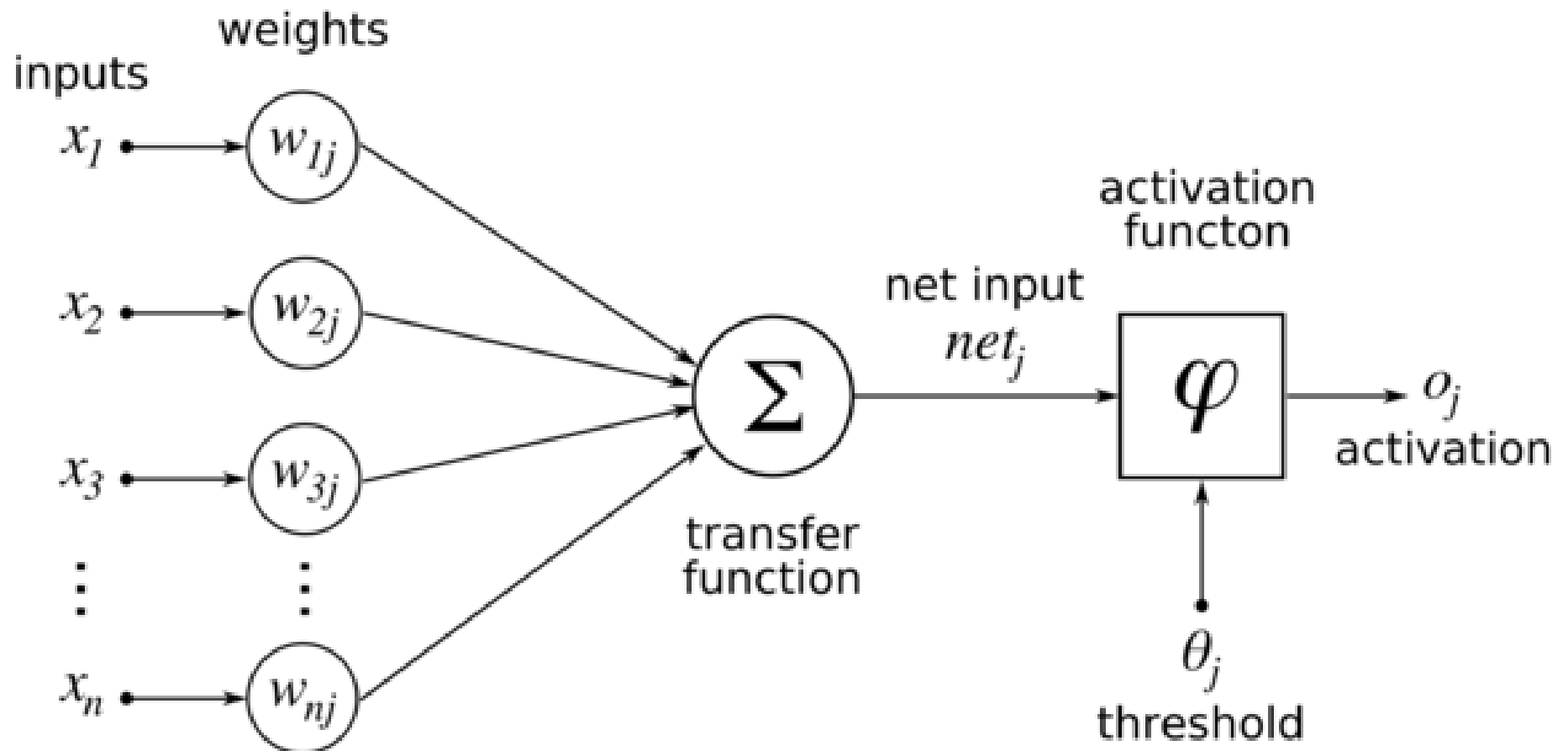


ELU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



Linear \rightarrow Non-linear



Convolution

Input Volume (+pad 1) (7x7x3)

$x[:, :, 0]$

0	0	0	0	0	0	0
0	0	0	1	0	2	0
0	1	0	2	0	1	0
0	1	0	2	2	0	0
0	2	0	0	2	0	0
0	2	1	2	2	0	0
0	0	0	0	0	0	0

$x[:, :, 1]$

0	0	0	0	0	0	0
0	2	1	2	1	1	0
0	2	1	2	0	1	0
0	0	2	1	0	1	0
0	1	2	2	2	2	0
0	0	1	2	0	1	0
0	0	0	0	0	0	0

$x[:, :, 2]$

0	0	0	0	0	0	0
0	2	1	1	2	0	0
0	1	0	0	1	0	0
0	0	1	0	0	0	0
0	1	0	2	1	0	0
0	2	2	1	1	1	0
0	0	0	0	0	0	0

Filter W0 (3x3x3)

$w0[:, :, 0]$

-1	0	1
0	0	1
1	-1	1

$w0[:, :, 1]$

-1	0	1
1	-1	1
0	1	0

$w0[:, :, 2]$

-1	1	1
1	1	0
0	-1	0

Bias b0 (1x1x1)

$b0[:, :, 0]$

1

Filter W1 (3x3x3)

$w1[:, :, 0]$

0	1	-1
0	-1	0
0	-1	1

$w1[:, :, 1]$

-1	0	0
1	-1	0
1	-1	0

$w1[:, :, 2]$

-1	1	-1
0	-1	-1
1	0	0

Bias b1 (1x1x1)

$b1[:, :, 0]$

0

Output Volume (3x3x2)

$o[:, :, 0]$

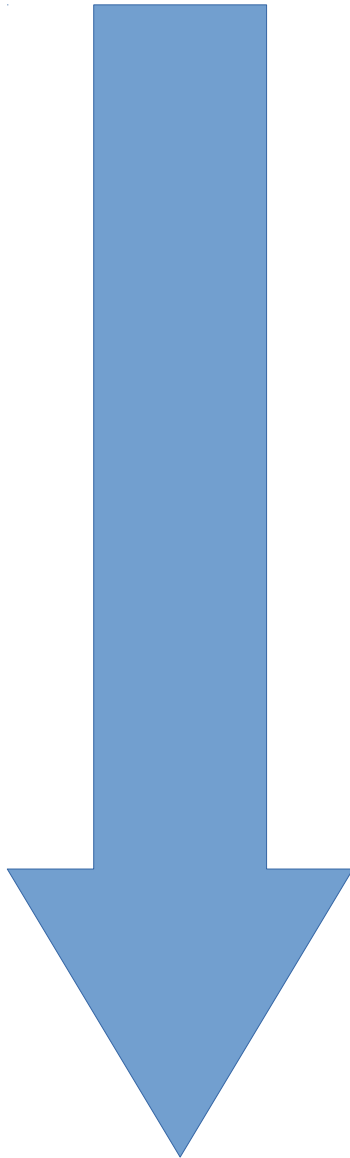
2	3	3
3	7	3
8	10	-3

$o[:, :, 1]$

-8	-8	-3
-3	1	0
-3	-8	-5

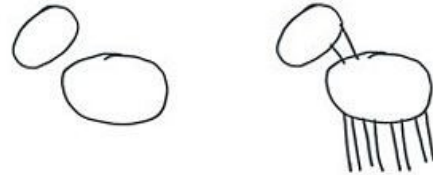
toggle movement

How our brains process images

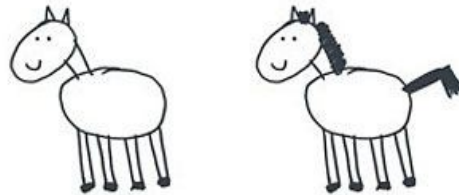


HOW TO: DRAW A HORSE

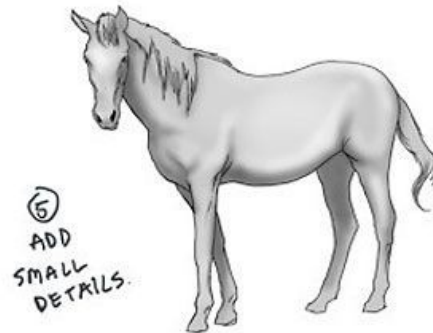
BY VAN OKTOP



① DRAW 2 CIRCLES ② DRAW THE LEGS



③ DRAW THE FACE ④ DRAW THE HAIR



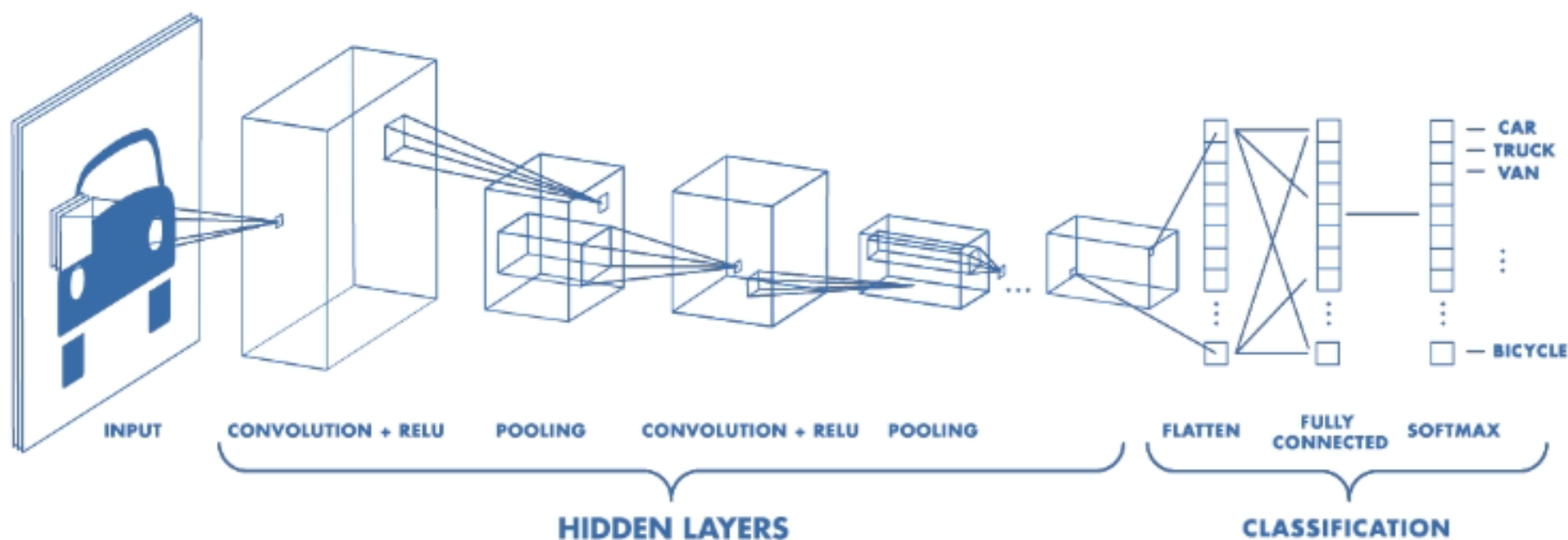
⑤
ADD
SMALL
DETAILS.

Pooling

1x1	1x0	1x1	0	0
0x0	1x1	1x0	1	0
0x1	0x0	1x1	1	1
0	0	1	1	0
0	1	1	0	0

4		

Putting them together

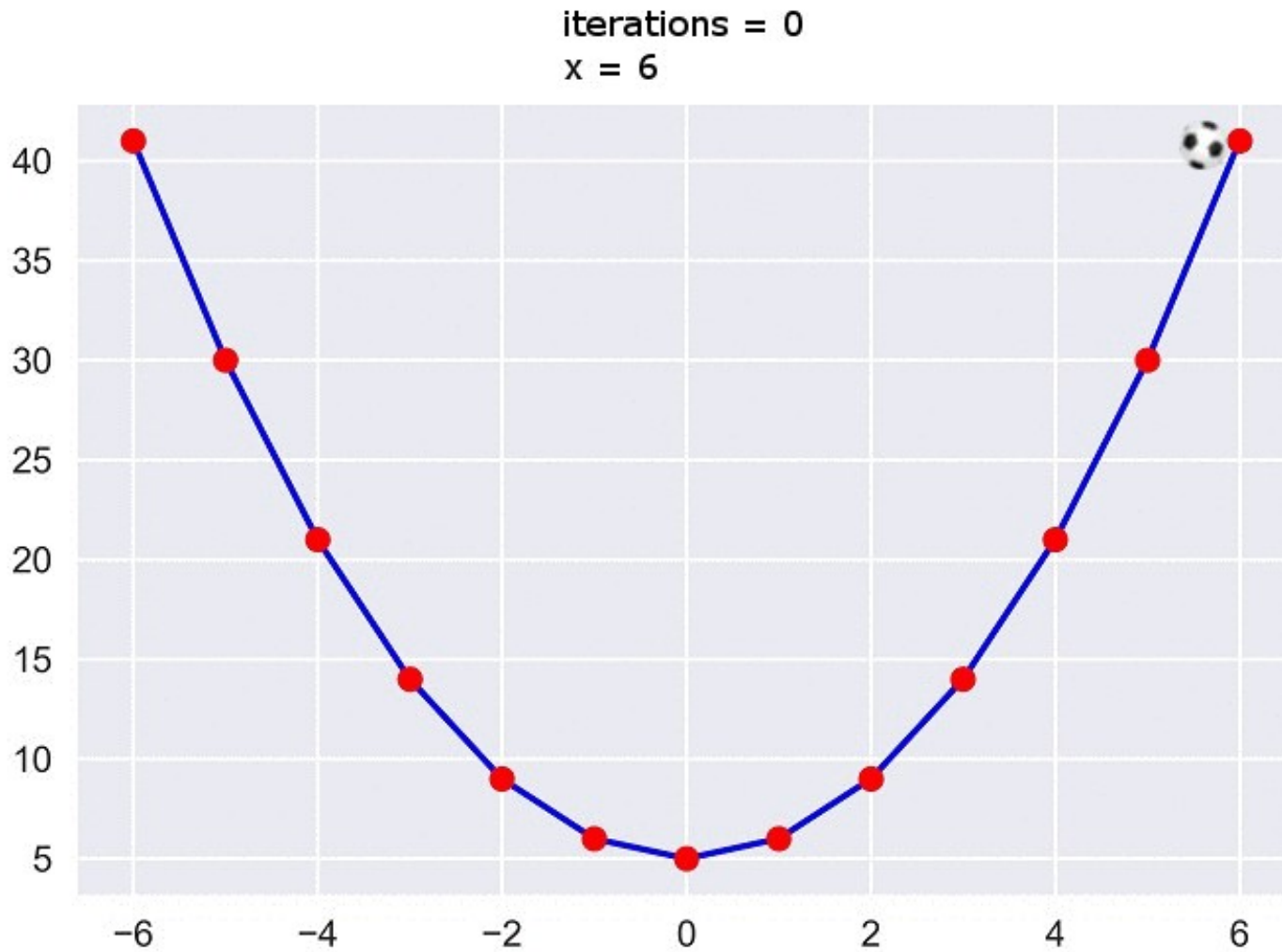


Learning – keep doing a better job

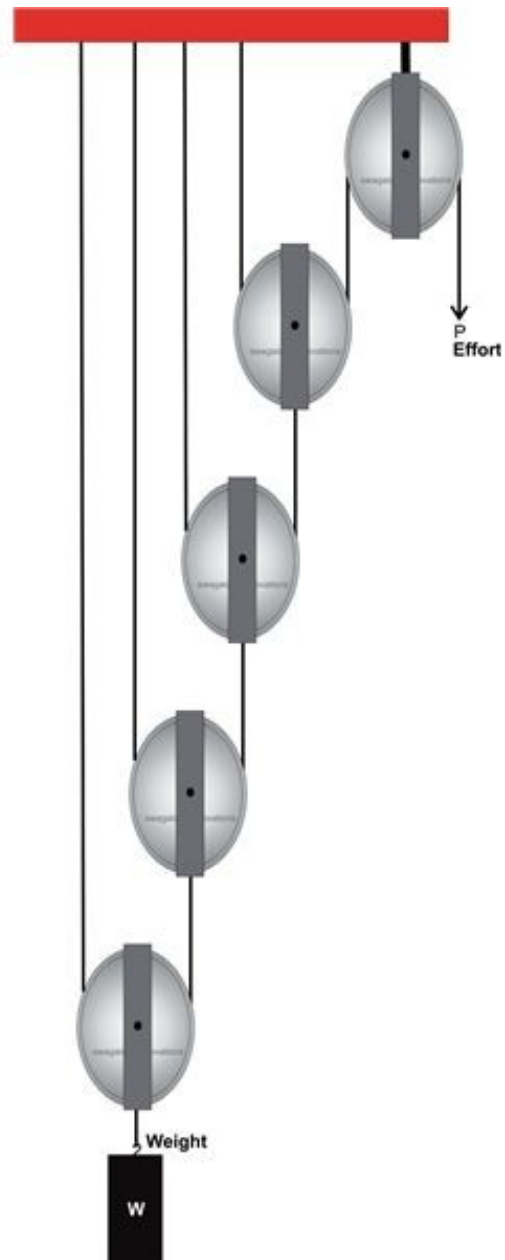


k0085653 www.fotosearch.com

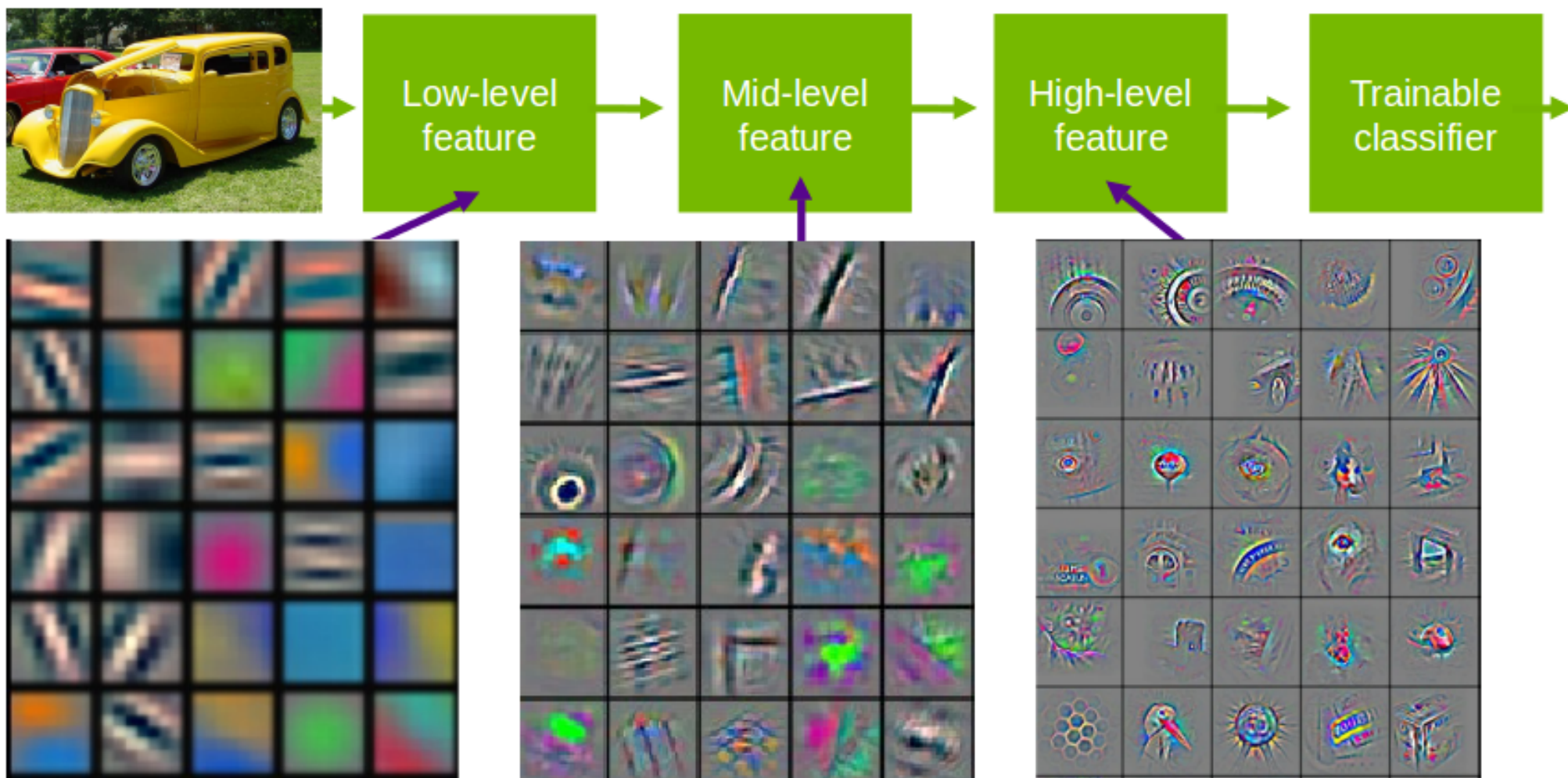
Gradient descent



Backprop!

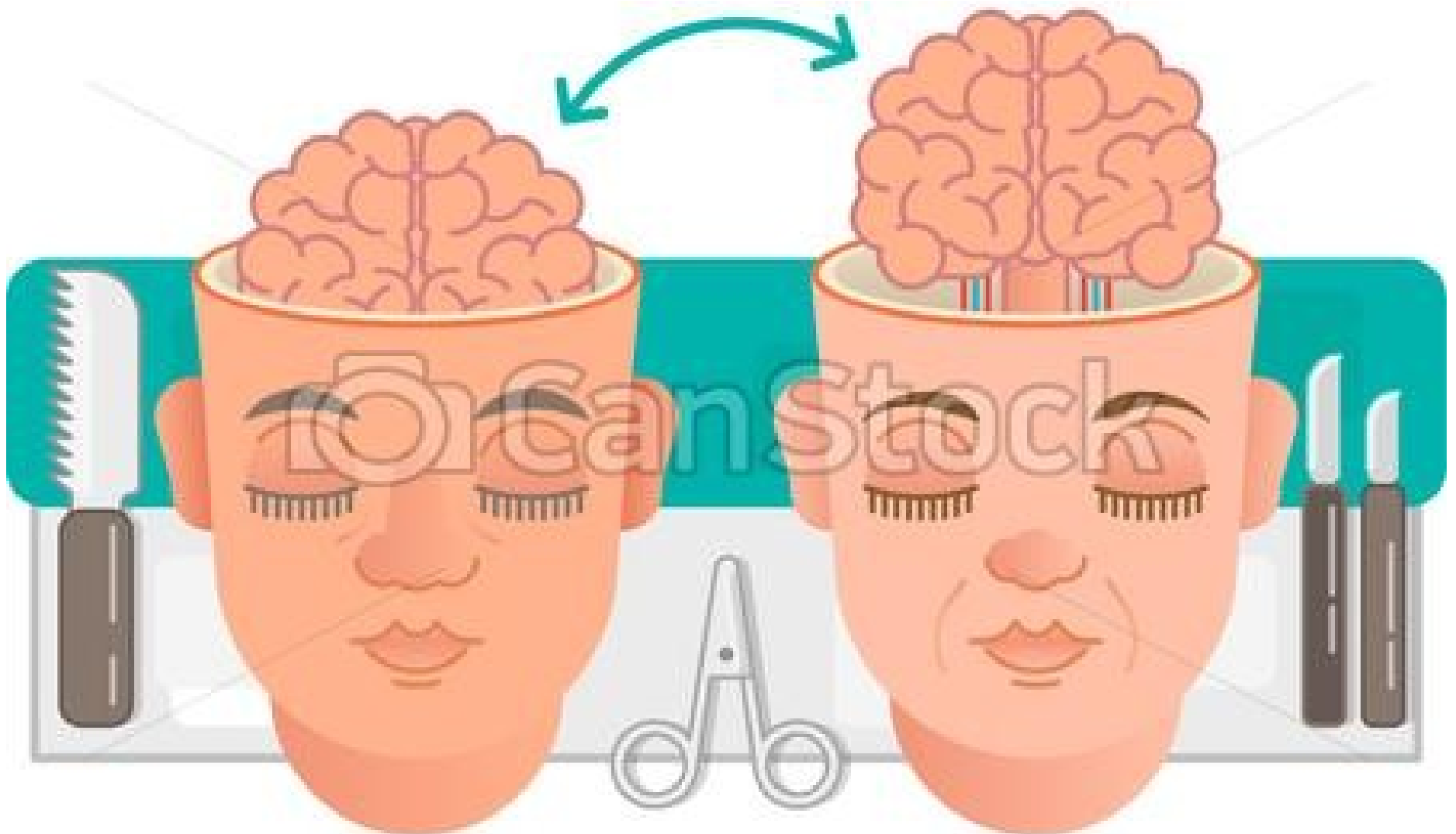


Feature extraction with CNN



Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]

Transfer learning



What can you do with this?

Convolutional neural networks

Transfer learning

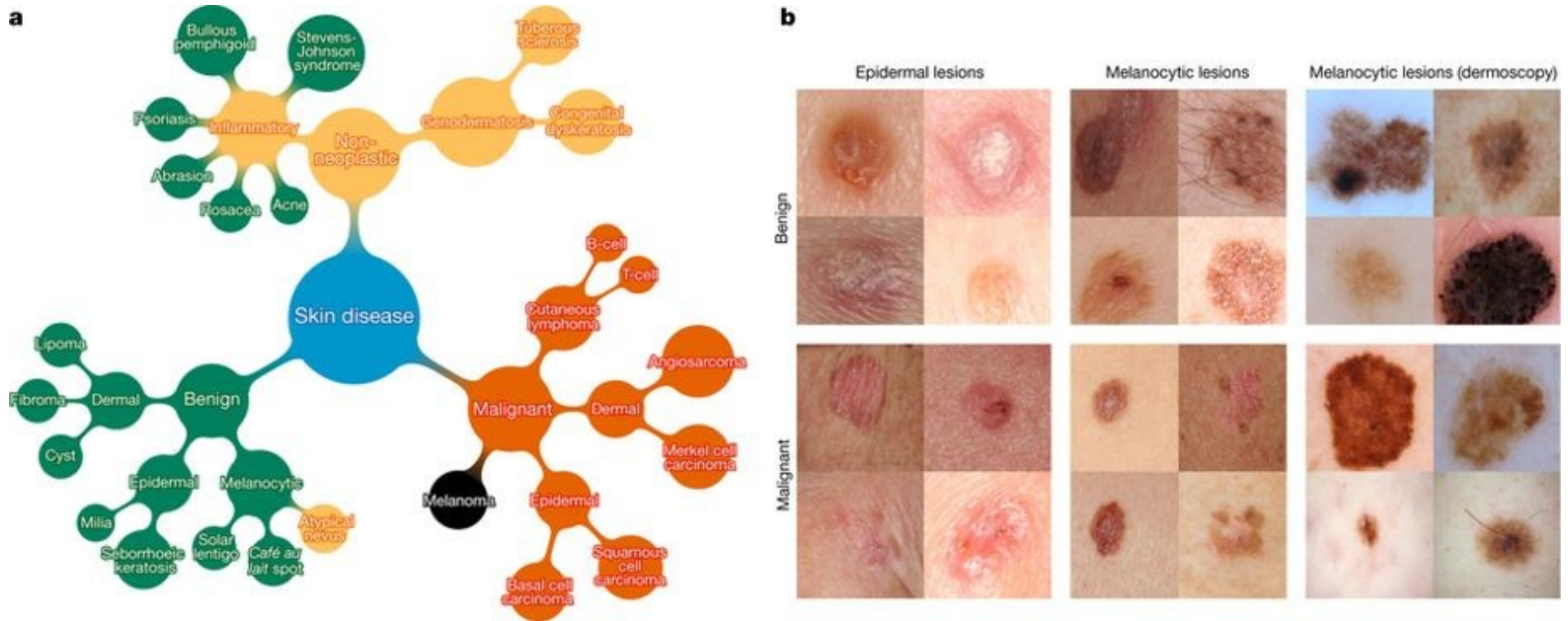
Advanced gradient descent algorithms

User friendly coding libraries (Keras, Fastai, Caffe2)

Access to powerful GPU hardware

Detect cancer better than a doctor

Accuracy of deep learning model > accuracy of 21 board-certified dermatologists



Esteva et al(2017) - Dermatologist-level classification of skin cancer with deep neural networks

<https://www.nature.com/articles/nature21056>

Create ridiculously expensive art



\$432,000

Classification



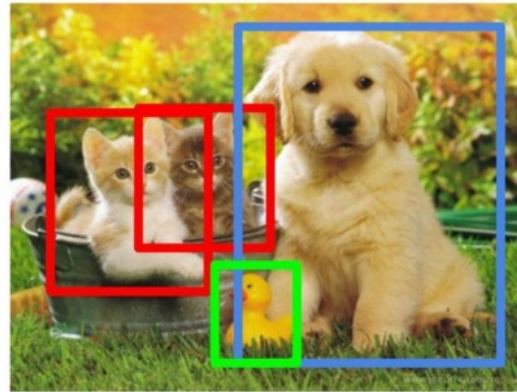
CAT

Classification + Localization



CAT

Object Detection



CAT, DOG, DUCK

Instance Segmentation



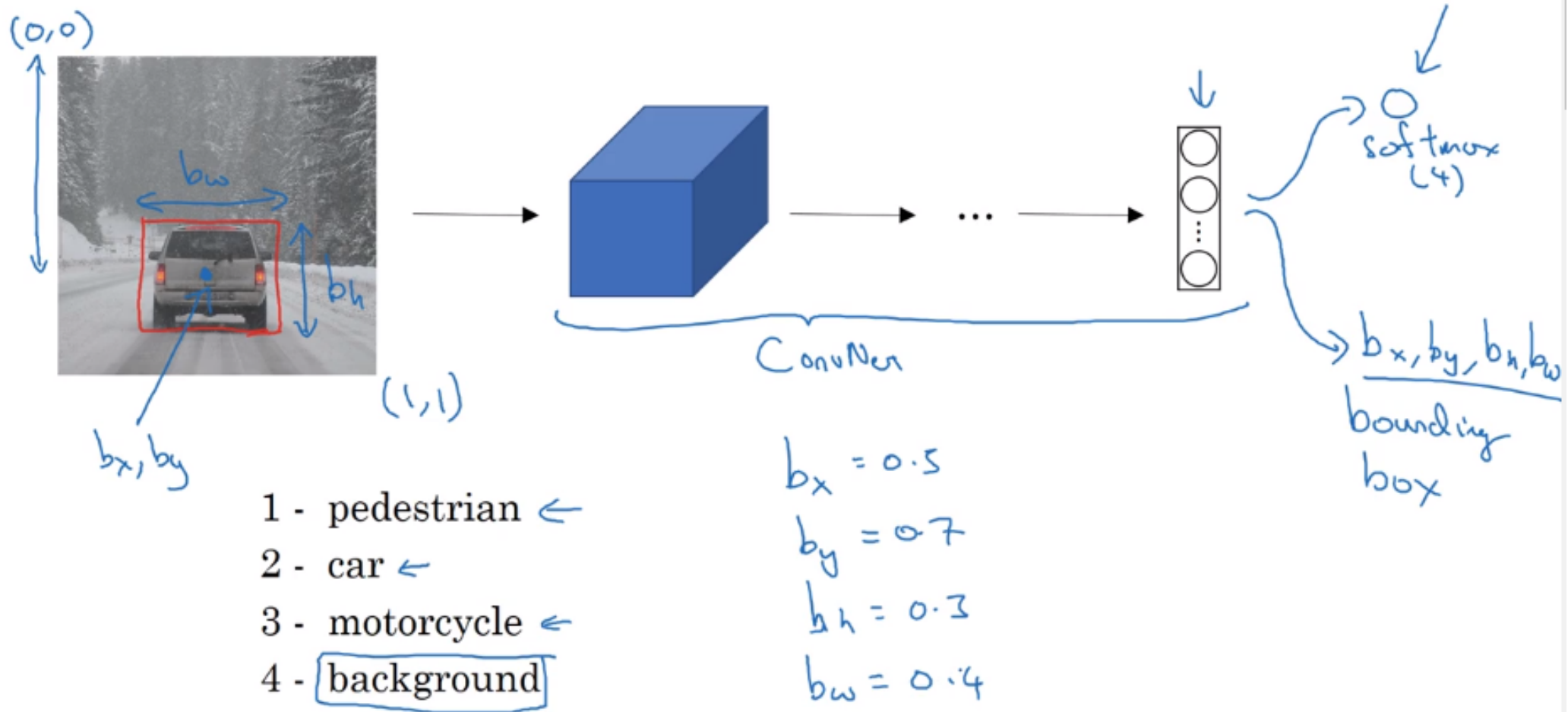
CAT, DOG, DUCK

Single object

Multiple objects

Arthur Ouaknine -
<https://medium.com/comet-app/>

Bounding boxes



Credits to Andrew Ng

Bounding box examples

$$\gamma = \begin{bmatrix} p_c \\ b_x \\ b_y \\ b_h \\ b_w \\ c_1 \\ c_2 \\ c_3 \end{bmatrix}$$

Detection

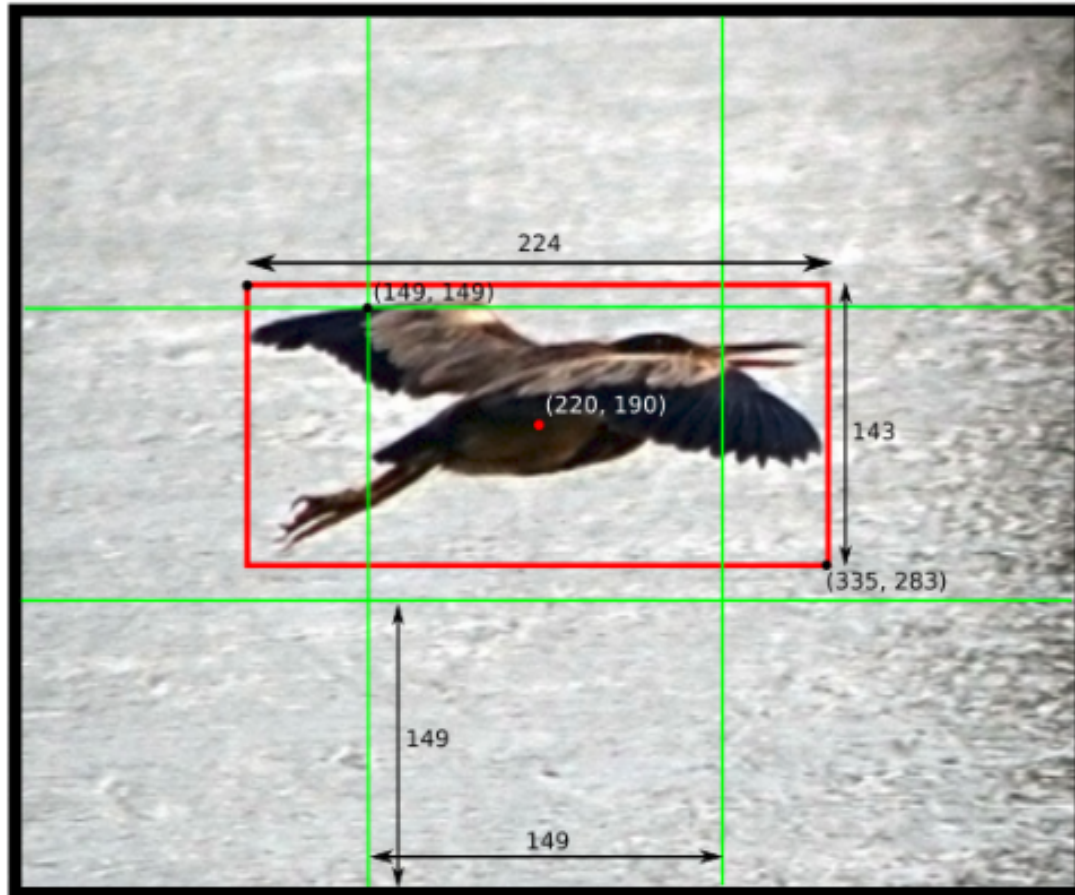
$$\begin{bmatrix} 1 \\ b_x \\ b_y \\ b_h \\ b_w \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

No detection

$$\begin{bmatrix} 0 \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \end{bmatrix}$$

Calculating box coordinates

(0, 0)



(447, 447)

$$x = (220 - 149) / 149 = 0.48$$

$$y = (190 - 149) / 149 = 0.28$$

$$w = 224 / 448 = 0.50$$

$$h = 143 / 448 = 0.32$$

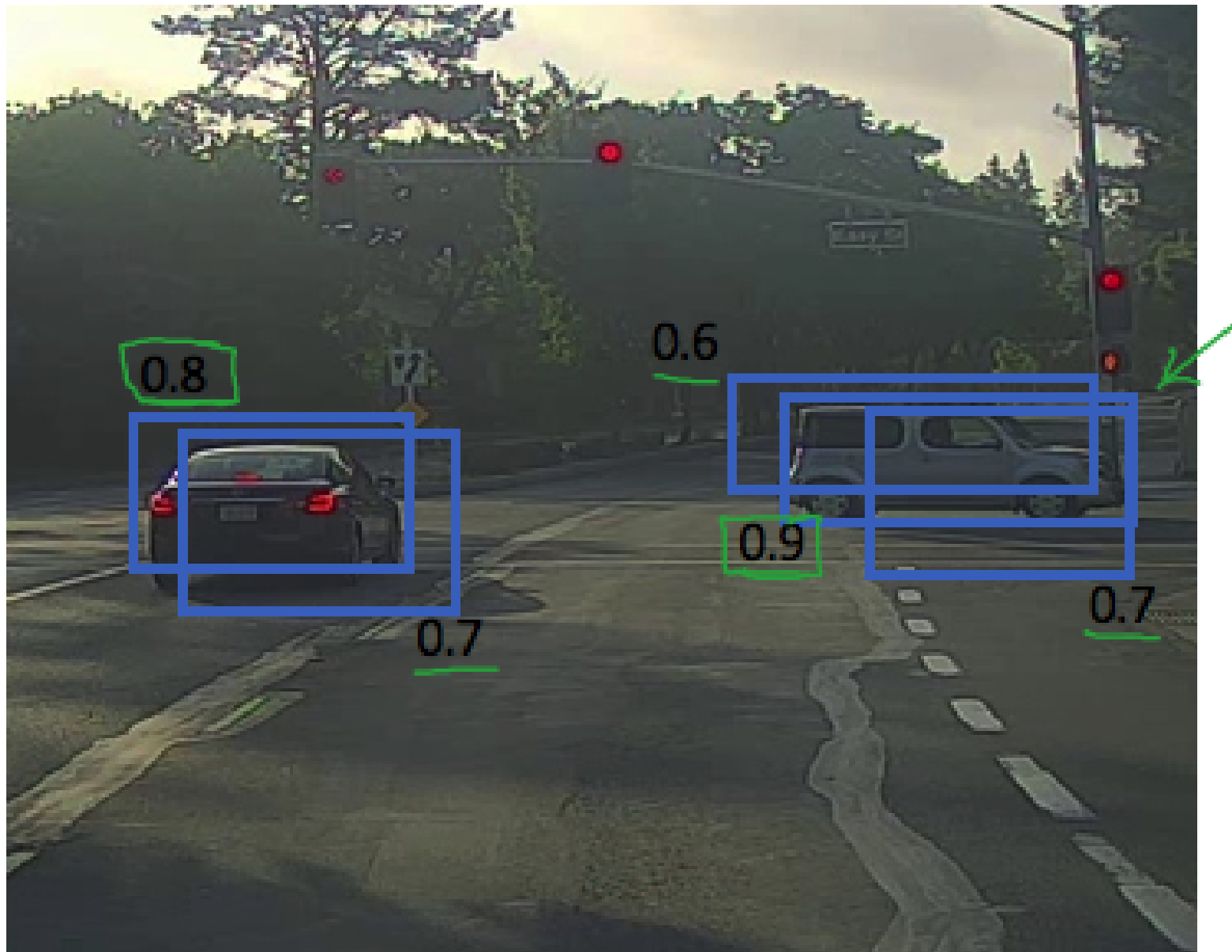
Example of how to calculate box coordinates in a 448x448 image with S=3. Note how the (x,y) coordinates are calculated relative to the center grid cell

IOU

$$IoU = \frac{\text{size of the intersection area}}{\text{size of the union area}}$$

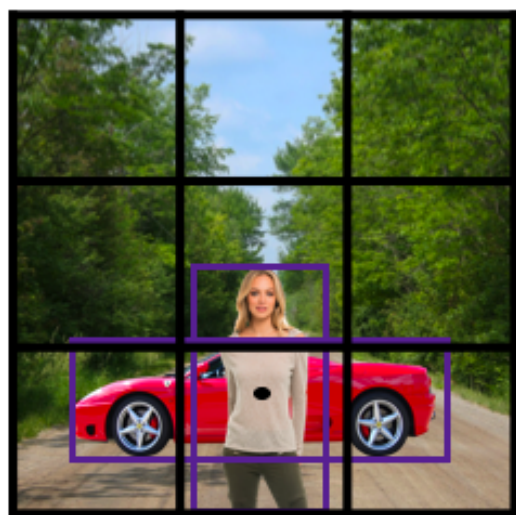


Non max suppression



Anchor boxes

$$Y = \begin{bmatrix} p_c \\ b_x \\ b_y \\ b_h \\ b_w \\ c_1 \\ c_2 \\ c_3 \\ p'_c \\ b'_x \\ b'_y \\ b'_h \\ b'_w \\ c'_1 \\ c'_2 \\ c'_3 \end{bmatrix}$$



$$y = \begin{bmatrix} p_c \\ b_x \\ b_y \\ b_h \\ b_w \\ c_1 \\ c_2 \\ c_3 \end{bmatrix}$$

Green arrow points to b_x, b_y, b_h, b_w . Blue arrow points to c_1, c_2, c_3 .

Anchor box 1:



Anchor box 2:



$$y = \begin{bmatrix} p_c \\ b_x \\ b_y \\ b_h \\ b_w \\ c_1 \\ c_2 \\ c_3 \\ p_c \\ b_x \\ b_y \\ b_h \\ b_w \\ c_1 \\ c_2 \\ c_3 \end{bmatrix}$$

Green box around $p_c, b_x, b_y, b_h, b_w, c_1, c_2, c_3$ labeled "Anchor box 1".
Orange box around $p_c, b_x, b_y, b_h, b_w, c_1, c_2, c_3$ labeled "Anchor box 2".