Getting Started with Computer Vision

By

Navaneeth Malingan

AI and IoT Researcher, Developer, Educator

Nivu Academy, Nunnari Labs

Computer vision is the field of computer science that focuses on replicating parts of the complexity of the human vision system and enabling computers to identify and process objects in images and videos in the same way that humans do.

At an abstract level, the goal of computer vision problems is to use the observed image data to infer something about the world.

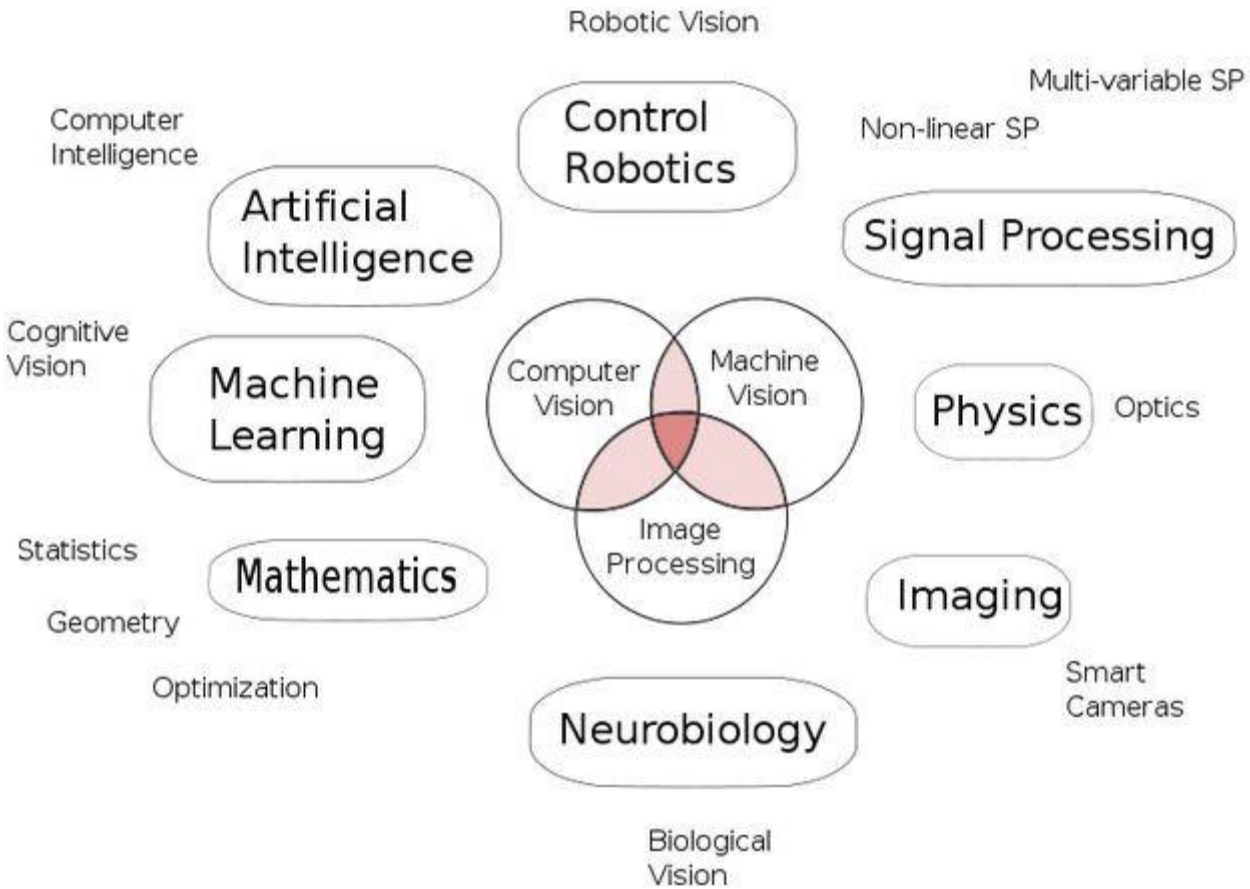
Working on Computer Vision is equivalent to working on millions of calculations in the blink of an eye with almost same accuracy as that of a human eye.

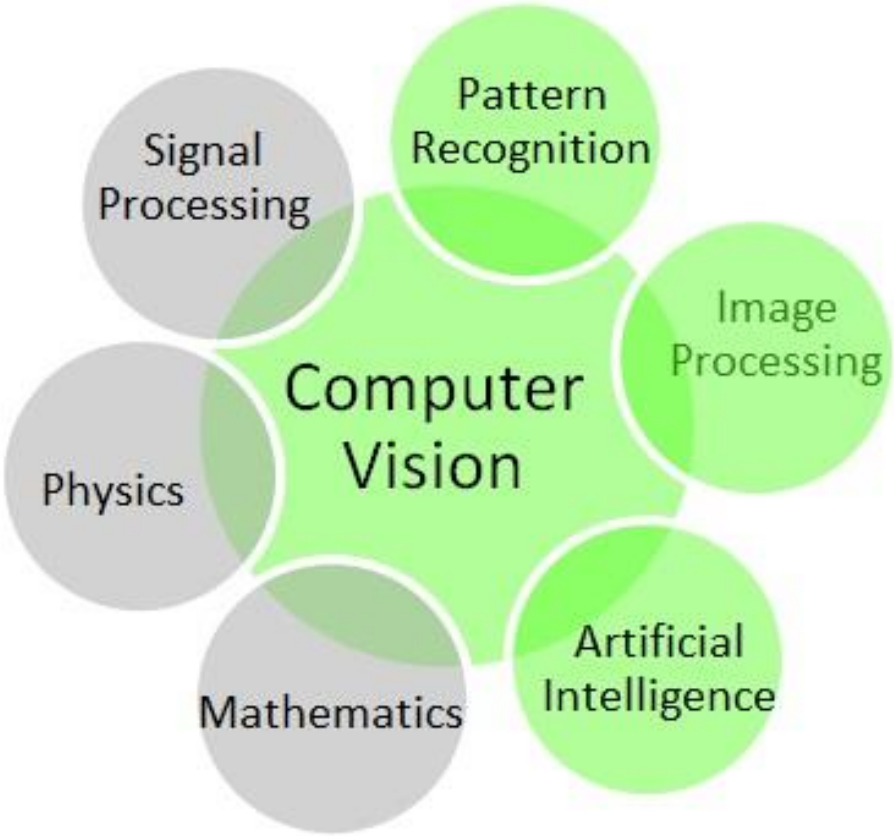

If We Want Machines to Think, We Need to Teach Them to See.
- Fei Fei Li,
Director of Stanford AI Lab and Stanford Vision Lab

Agenda

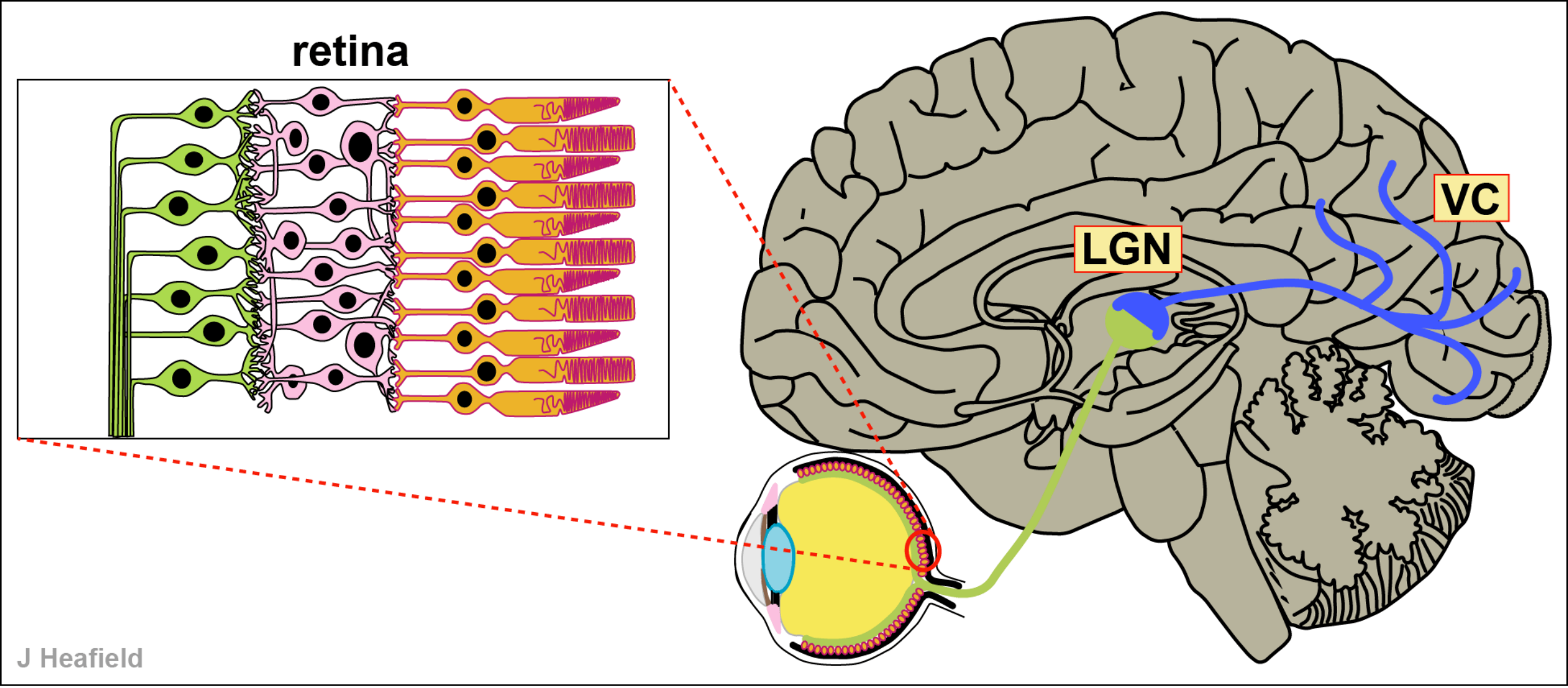
- Images and its Properties
- Traditional Computer Vision
- Deep Learning
- CNN

Areas that deal with Images


Robotic Vision
Multi-variable SP
Computer Intelligence
Control Robotics
Non-linear SP
Artificial Intelligence
Signal Processing
Cognitive Vision
Computer Vision
Machine Vision
Machine Learning
Physics
Optics
Image Processing
Statistics
Mathematics
Imaging
Geometry
Optimization
Smart Cameras
Neurobiology
Biological Vision


Pattern Recognition
Signal Processing
Image Processing
Computer Vision
Physics
Artificial Intelligence
Mathematics

How Human Vision Works?



Vision begins with the eyes, but it truly takes place in the brain.

Author - Navaneeth Malingan, Nivu Academy & Nunnari Labs

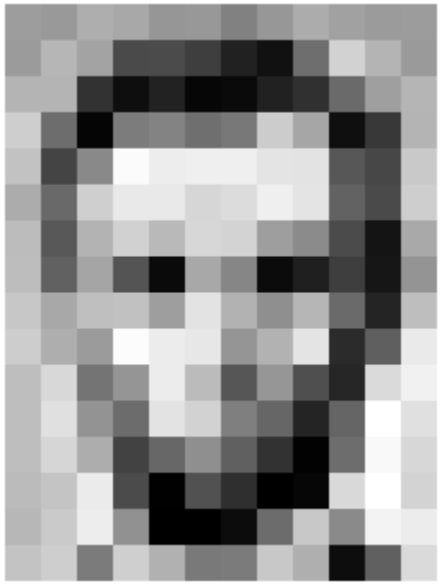
How do Machines see?

Author - Navaneeth Malingan, Nivu Academy & Nunnari Labs

Data

- Images
- Videos
- 3D Models

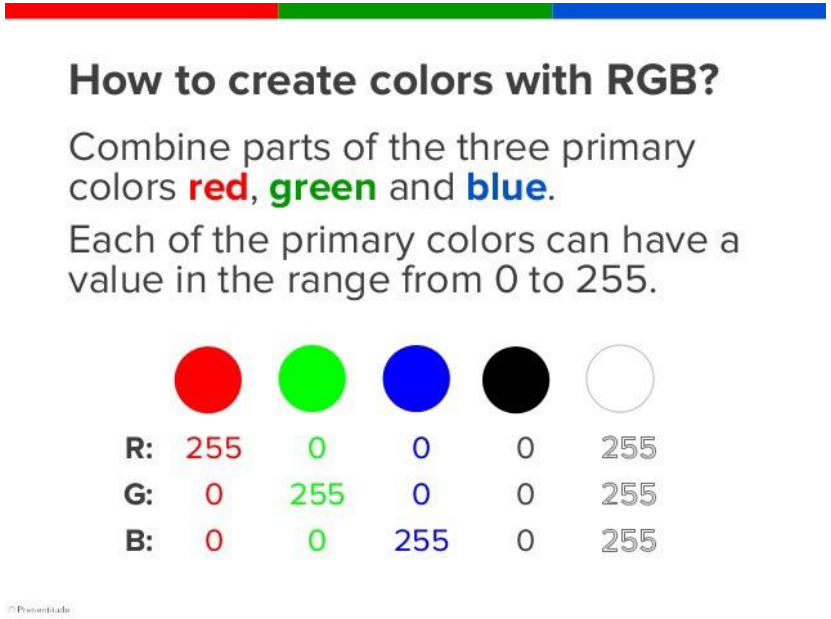
Images as Pixels

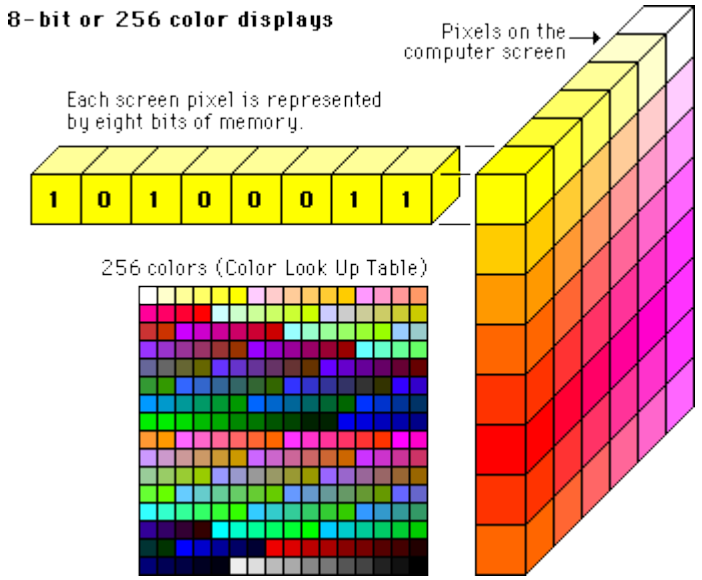


157	153	174	168	150	152	129	151	172	161	155	156
155	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	34	6	10	33	48	106	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	105	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	106	36	190
205	174	155	252	236	231	149	178	228	43	95	234
190	216	116	149	236	187	86	150	79	38	218	241
190	224	147	108	227	210	127	102	36	101	255	224
190	214	173	66	103	143	96	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
195	206	123	207	177	121	123	200	175	13	96	218

157	153	174	168	150	152	129	151	172	161	155	156
155	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	34	6	10	33	48	106	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	105	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	106	36	190
205	174	155	252	236	231	149	178	228	43	95	234
190	216	116	149	236	187	86	150	79	38	218	241
190	224	147	108	227	210	127	102	36	101	255	224
190	214	173	66	103	143	96	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
195	206	123	207	177	121	123	200	175	13	96	218

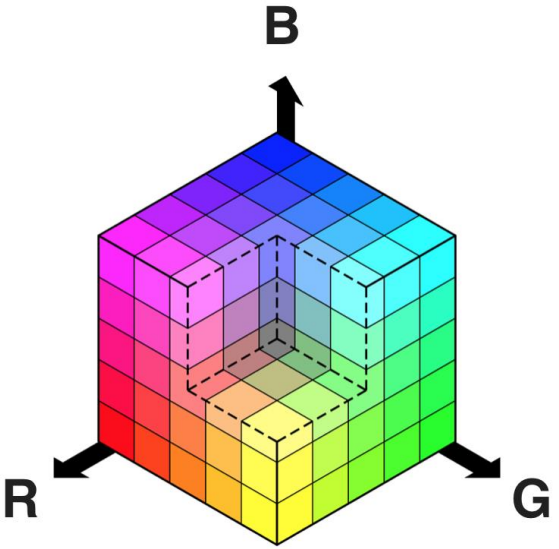
Represent colors by numbers: In computer science, each color is represented by a specified HEX value. That is how machines are programmed to understand what colors the image pixels are made up. Whereas as humans we have an inherited knowledge to differ between the shades.





Color Spaces

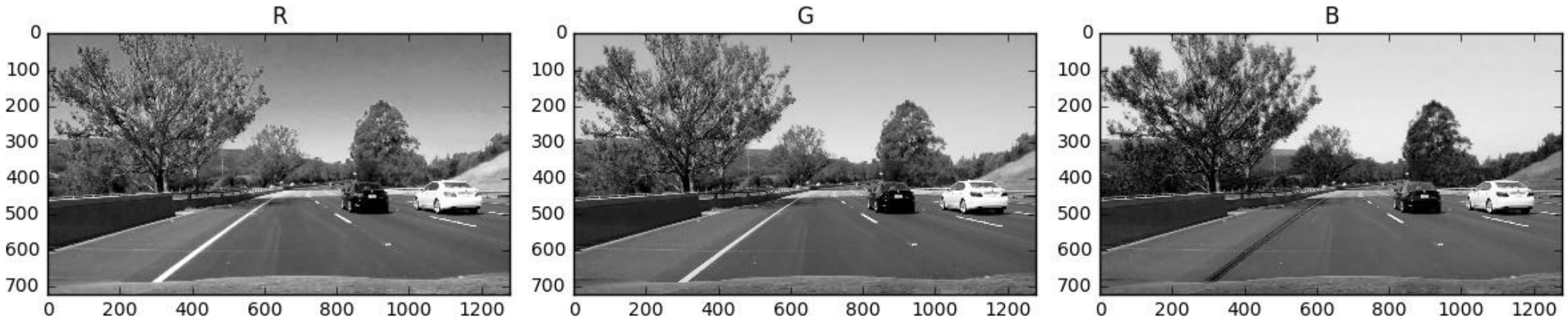
- RGB
- HSV
- HSL



- There is also **HSV** color space (hue, saturation, and value), and **HLS** space (hue, lightness, and saturation). These are some of the most commonly used color spaces in image analysis.

HLS Color Space

Yellow Line



Many popular computer vision applications involve trying to recognize things in photographs; for example:

- **Image Classification**: What broad category of object is in this photograph?
- **Object Identification**: Which type of a given object is in this photograph?
- **Object Verification**: Is the object in the photograph?
- **Object Detection**: Where are the objects in the photograph?
- **Object Landmark Detection**: What are the key points for the object in the photograph?
- **Image Segmentation**: What pixels belong to the object in the image?
- **Object Recognition**: What objects are in this photograph and where are they?
- **Object Tracking**: Where is the object in the video frames?
- **Image Generation**: Generate new images
- **OCR**: Image to Text

Outside of just recognition, other methods of analysis include:

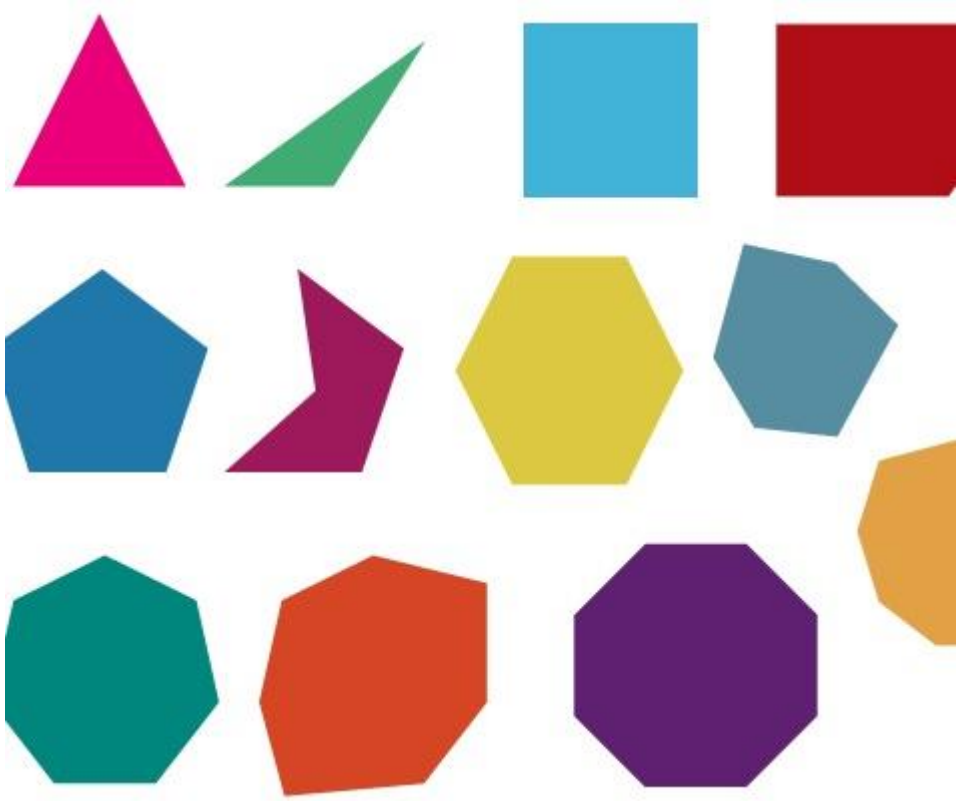
- **Video motion analysis** uses computer vision to estimate the velocity of objects in a video, or the camera itself.
- In **image segmentation**, algorithms partition images into multiple sets of views.
- **Scene reconstruction** creates a 3D model of a scene inputted through images or video.
- In **image restoration**, noise such as blurring is removed from photos using Machine Learning based filters.

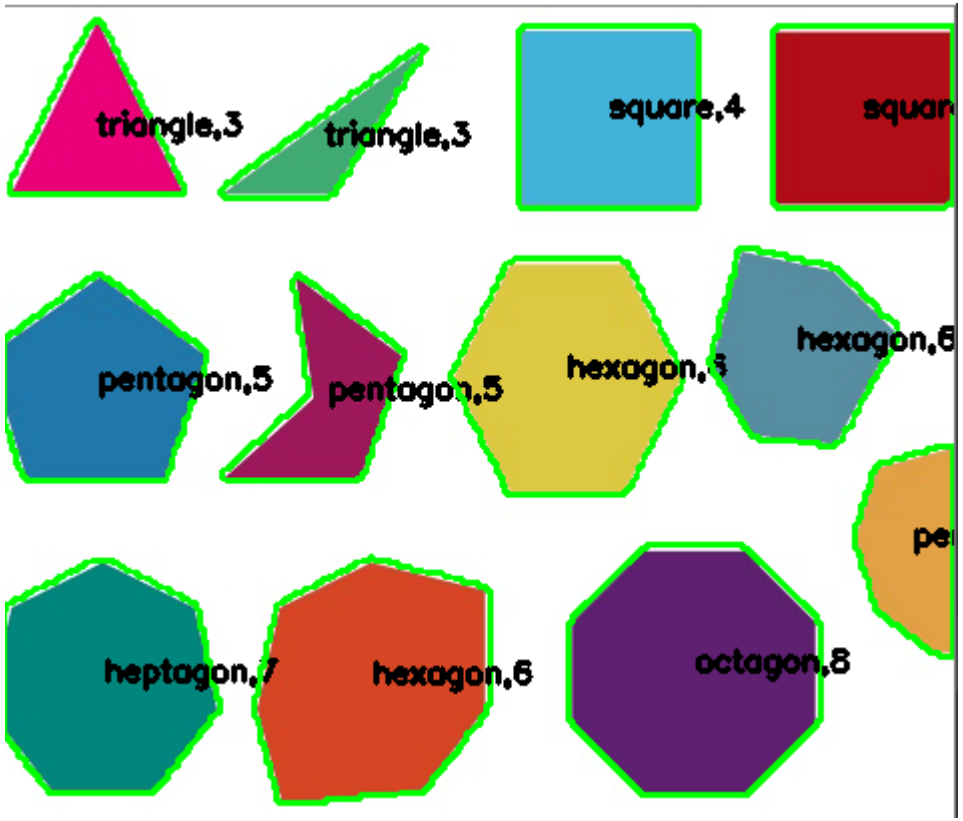
SoAI CV Curriculum

- Low Level Vision (Image to Image)
  - Basic Image Processing
  - Optical Flow
- Mid Level Vision (Image to Features)
  - Basic Segmentation
  - Fitting
- Multiple Views
  - Multiple Images
  - 3D Scenes
- High Level Vision (Features to Analysis)
  - Object Detection and Classification
  - Modern Deep Learning

1. Low Level Vision : Image Brightness

2. Mid Level Vision : Contour Detection

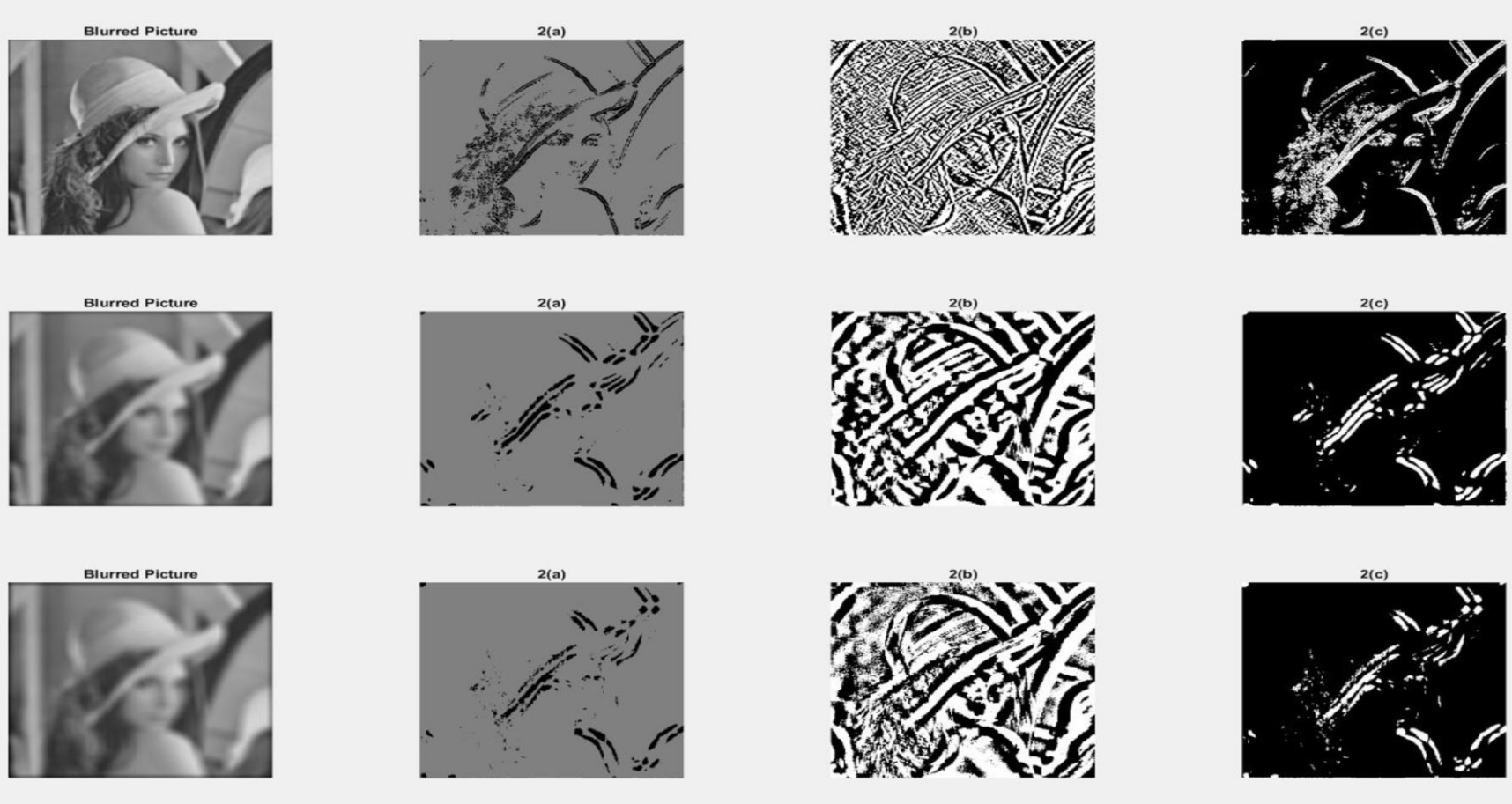




Number of polygons 12

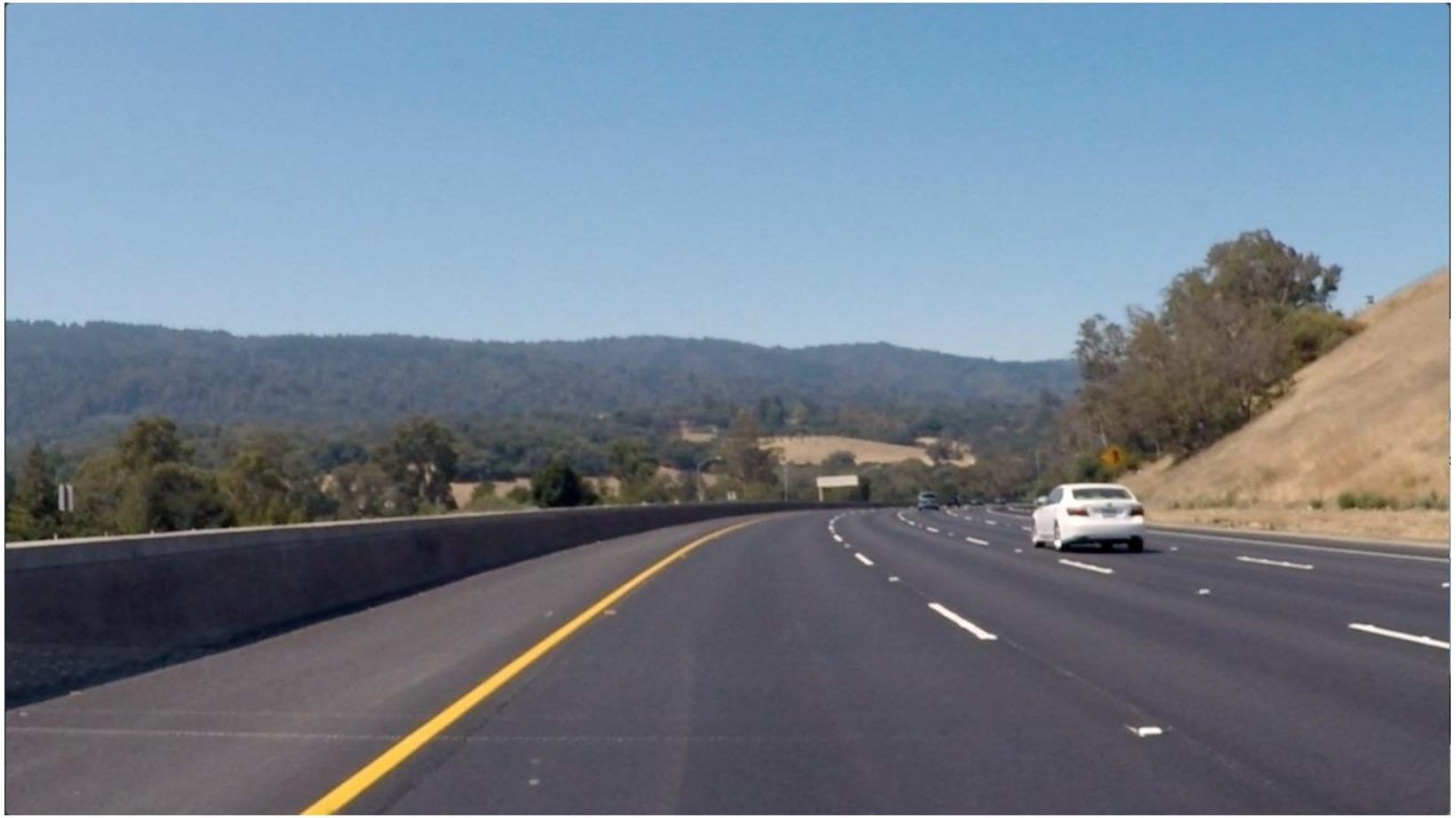
No. of vertices 62

Mid Level Vision : Edge Detection



Author - Navaneeth Malingan, Nivu Academy & Nunnari Labs

Mid Level Vision : Edge Detection
If we apply the Sobel x and y operators to this image:



Sobel x

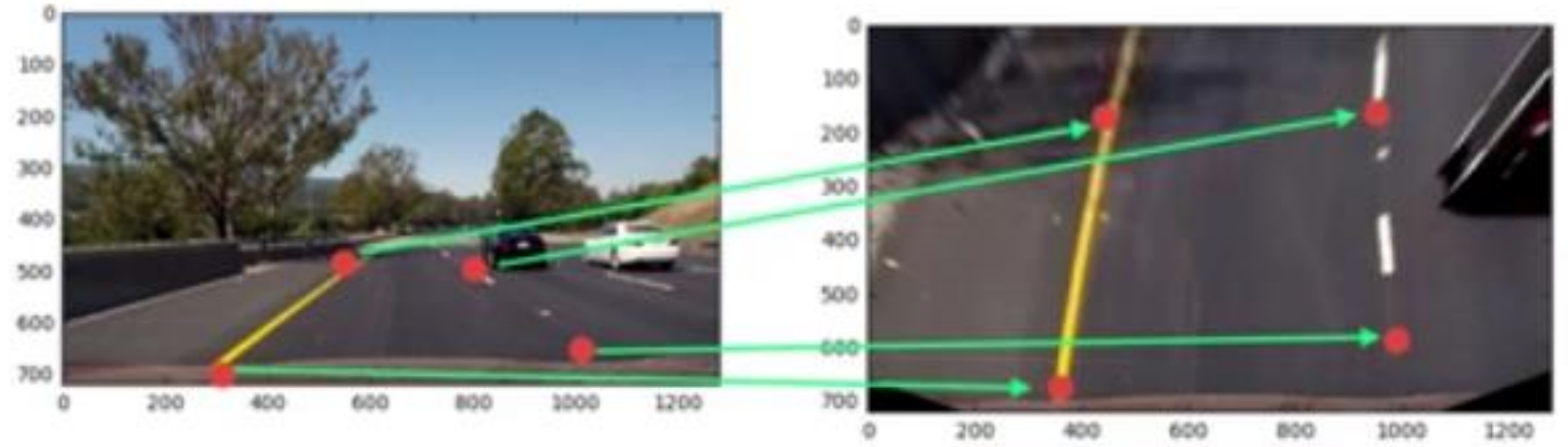
Sobel y

3. Multiple Views : Image Stitching

Multiple Views : Perspective Transform

Author - Navaneeth Malingan, Nivu Academy & Nunnari Labs

Multiple Views : Roads (Birds Eye View)



SDC Camera Calibration

Extras

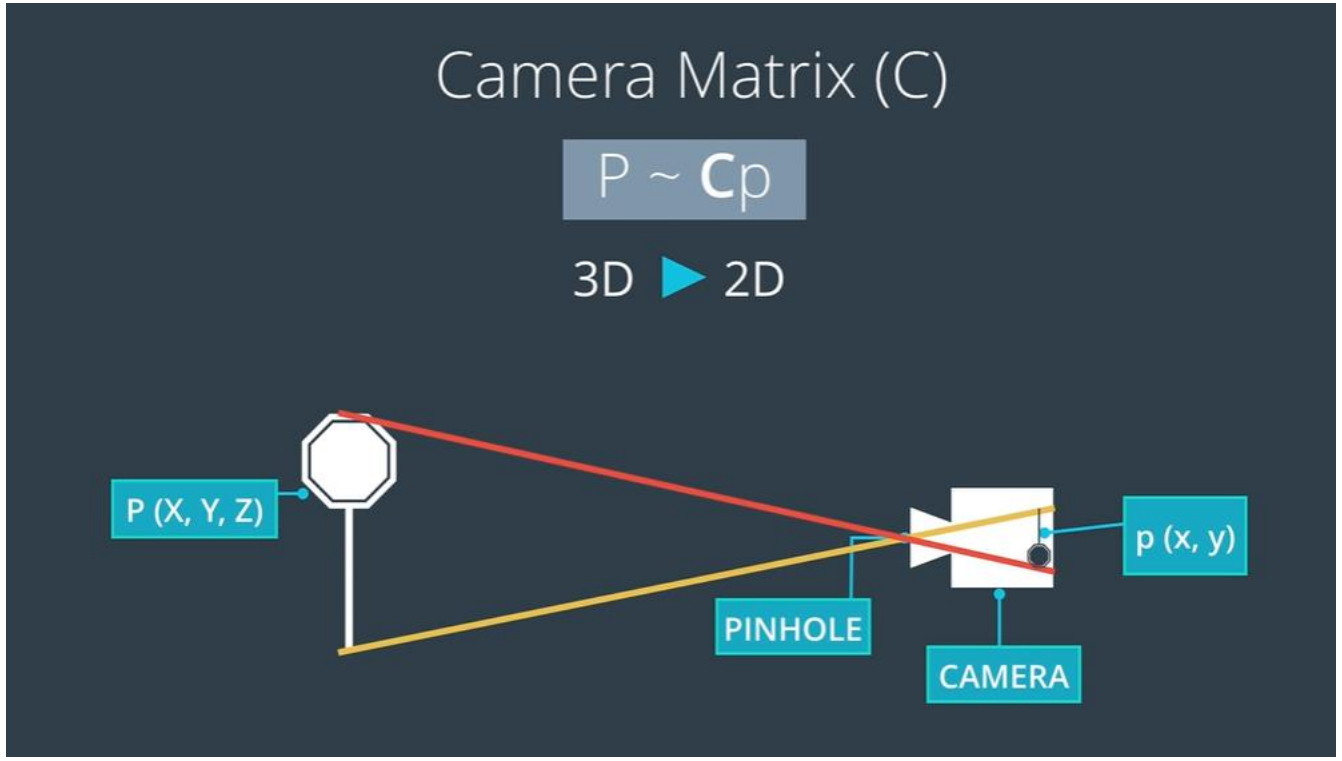
Author - Navaneeth Malingan, Nivu Academy & Nunnari Labs

Sensors	Spatial Resolution	3D	Cost
Radar + Lidar	Low	Yes	$$$
Single Camera	High	No	$

- **Distortion**
- Image distortion occurs when a camera looks at 3D objects in the real world and transforms them into a 2D image; this transformation isn’t perfect. Distortion actually changes what the shape and size of these 3D objects appear to be. So, the first step in analyzing camera images, is to undo this distortion so that you can get correct and useful information out of them.

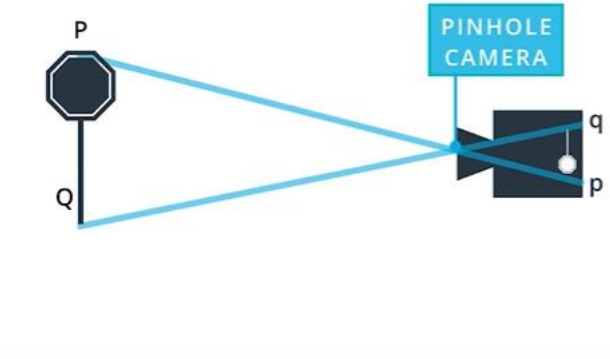
Why is it important to correct for image distortion?

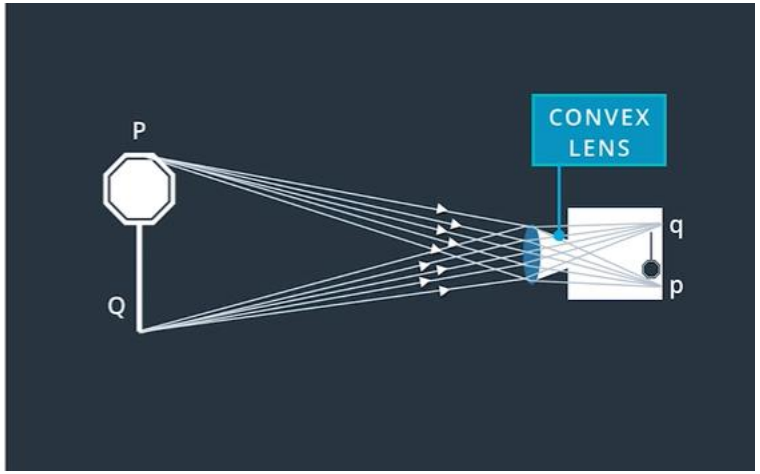
- Distortion can change the apparent size of an object in an image.
- Distortion can change the apparent shape of an object in an image.
- Distortion can cause an object's appearance to change depending on where it is in the field of view.
- Distortion can make objects appear closer or farther away than they actually are.



Author - Navaneeth Malingan, Nivu Academy & Nunnari Labs

Light rays bend at edges of camera lens

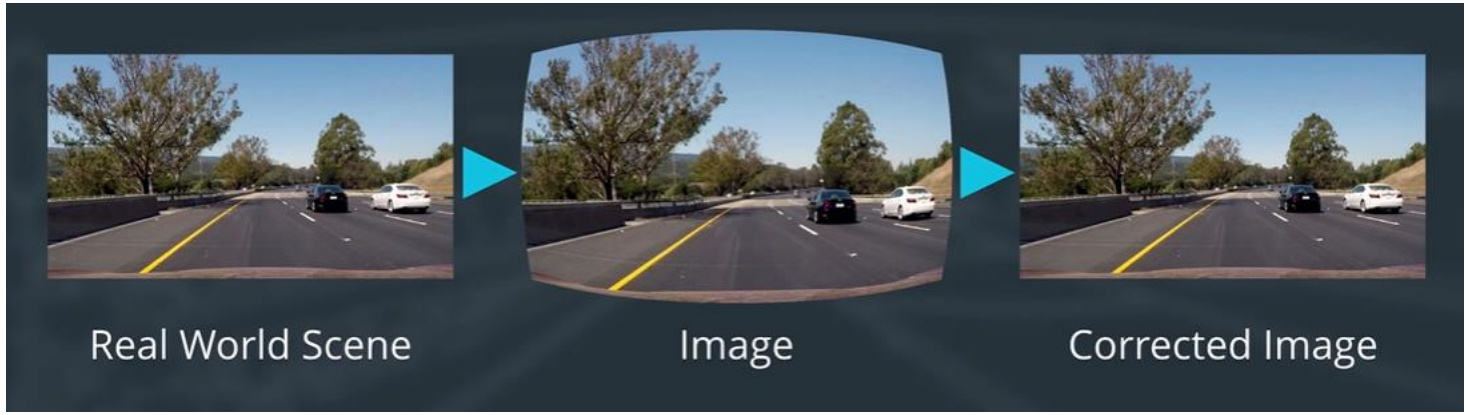





Radial Distortion

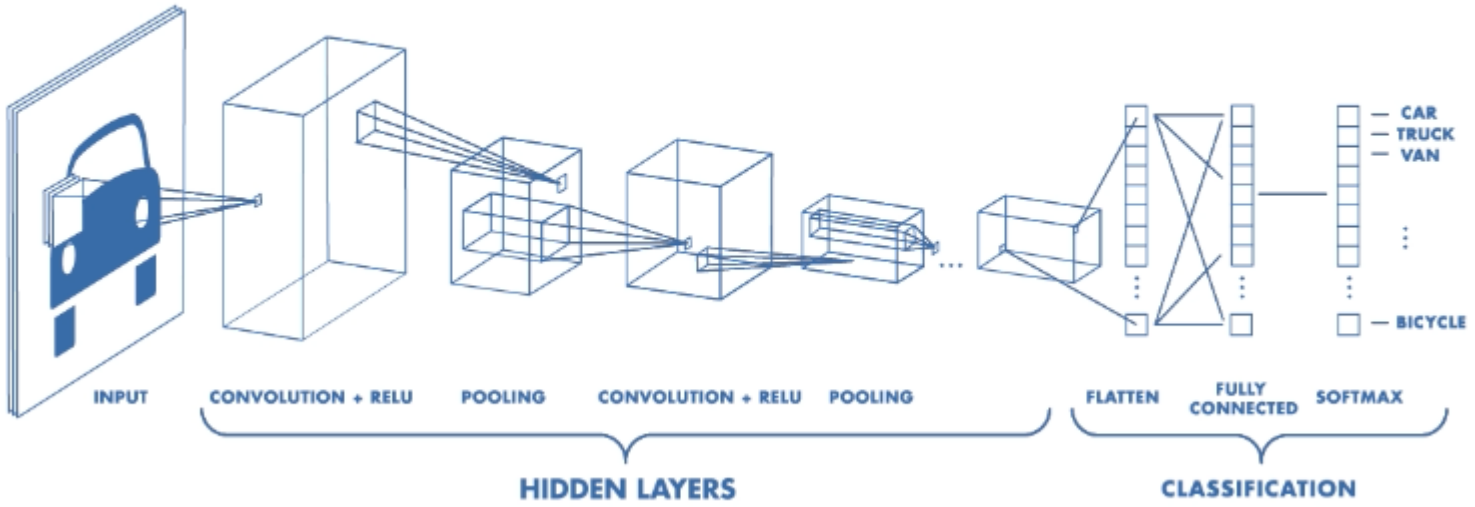

Tangential Distortion

Using these distortion coefficients we can undistorted camera images



- Radial Distortion
- Tangential Distortion
- Purposeful distortions like (fish eye, wide-angle)

4. High Level Vision : Deep Learning (CNN)



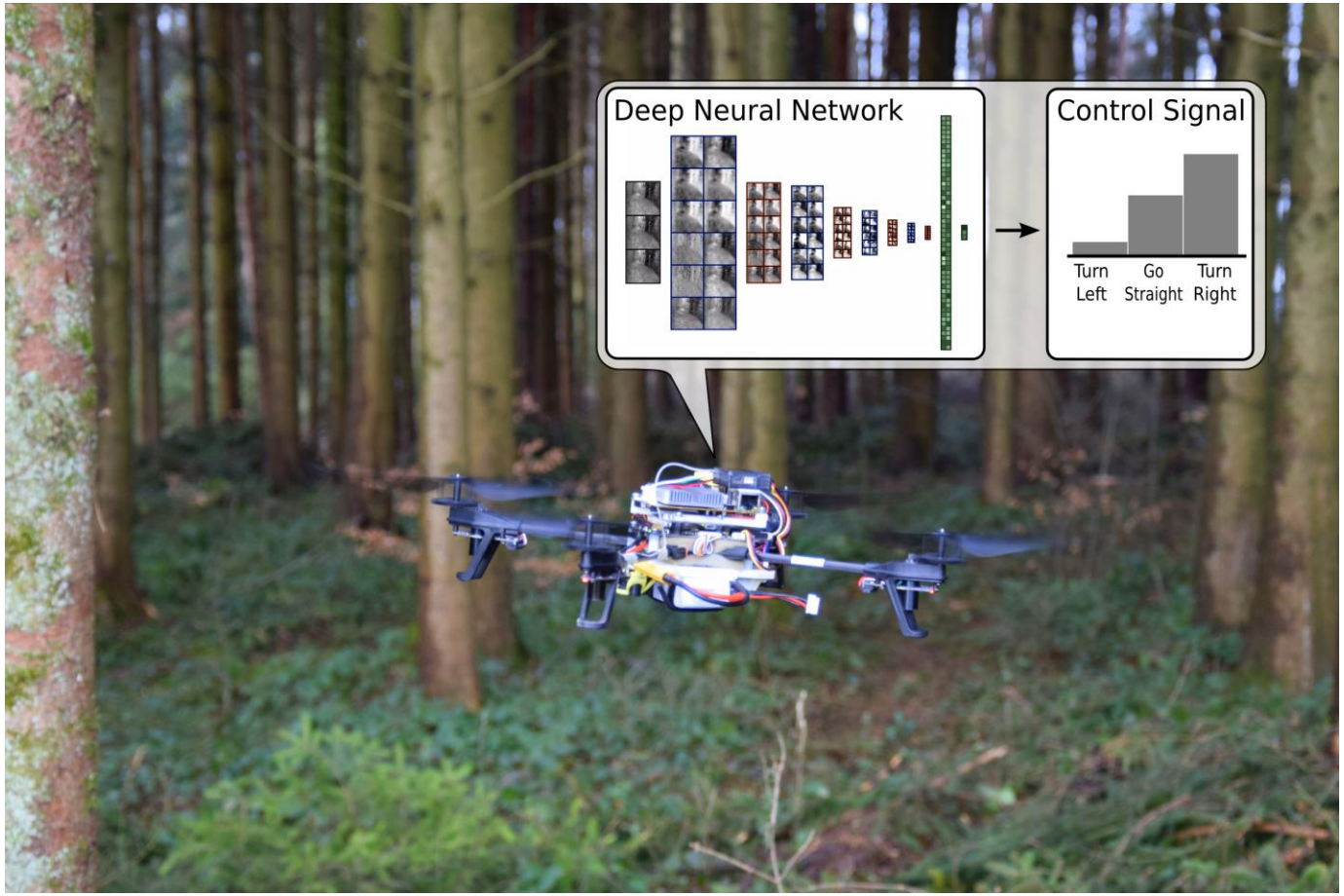
Computer Vision Learning Path

- Learn Computer Vision
  - https://www.youtube.com/watch?v=FSe_02FpJas
  - https://github.com/llSourcell/Learn_Computer_Vision
- Computer Vision, PyImageSearch
  - https://www.pyimagesearch.com/start-here/
  - https://www.pyimagesearch.com/pyimagesearch-gurus/

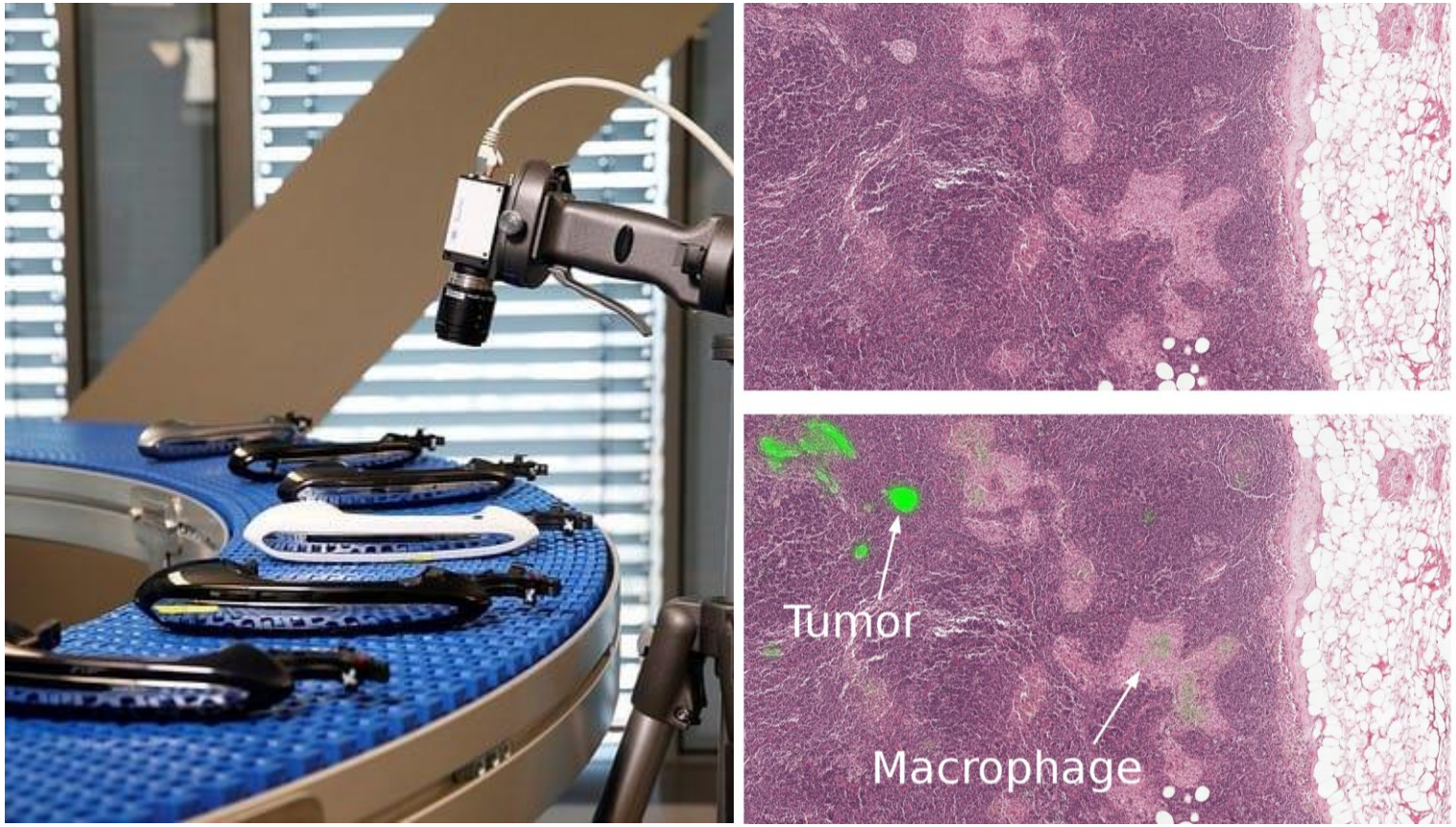
Applications

Self driving cars

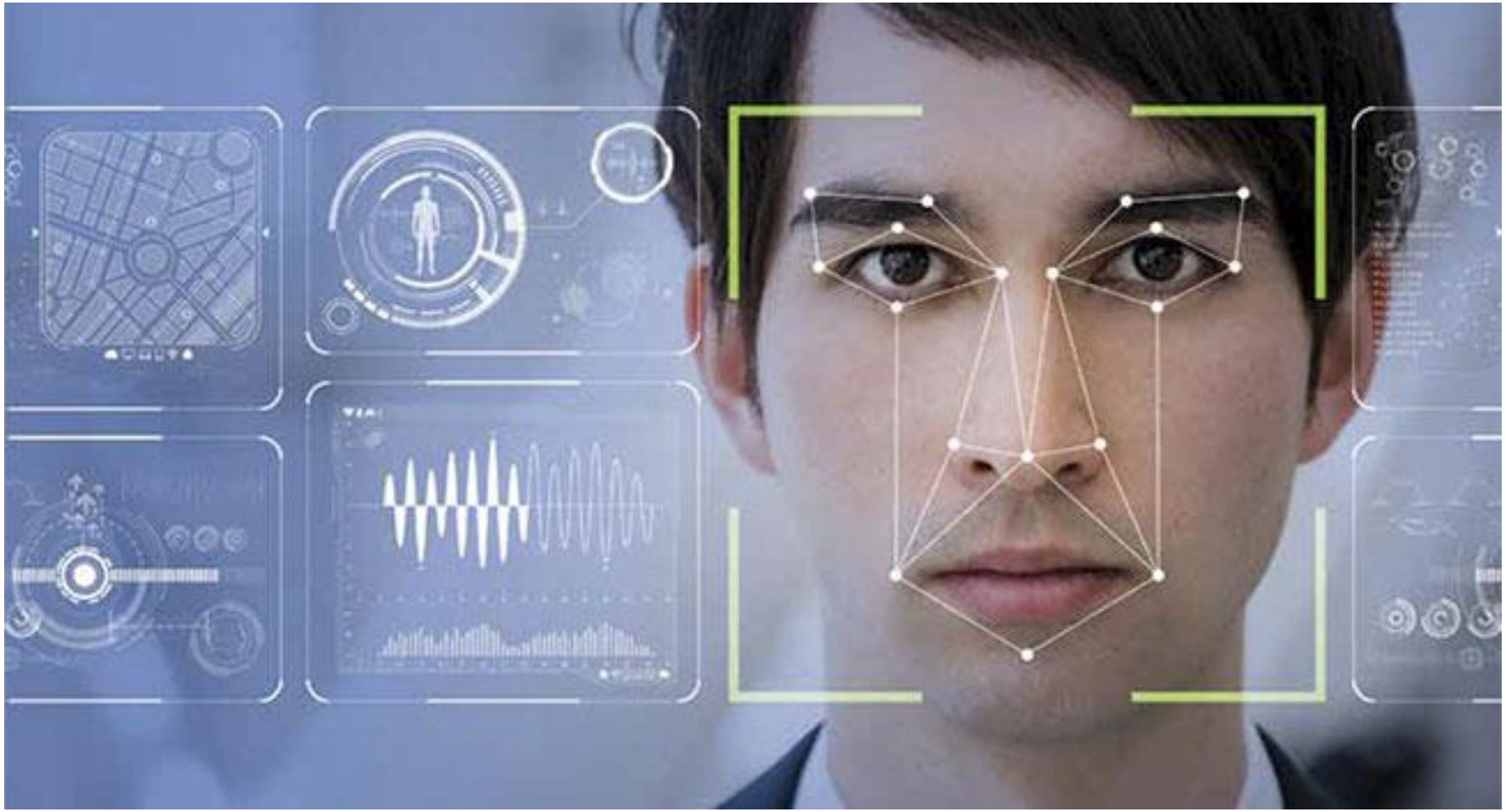
Autonomous Drones



Healthcare

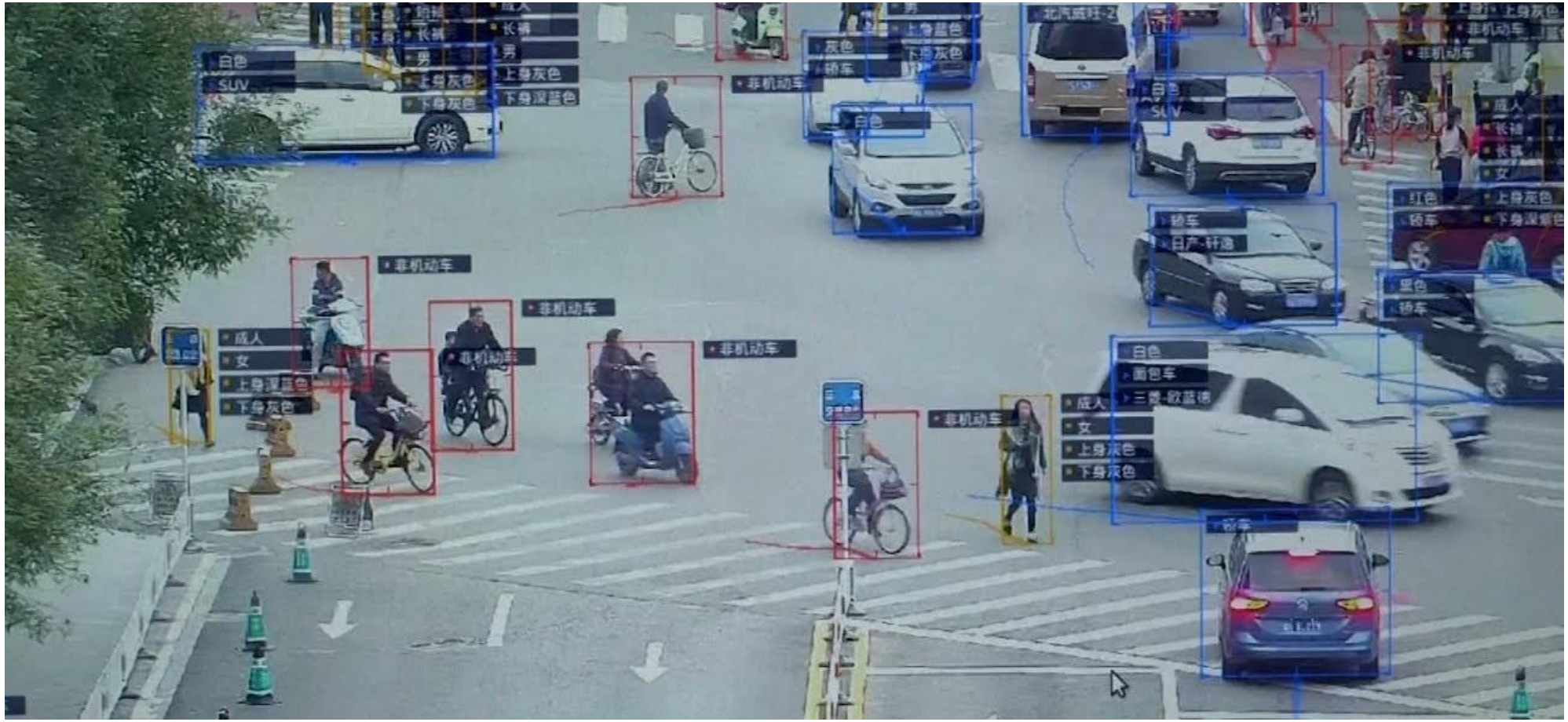


Face Recognition and Authentication



Author - Navaneeth Malingan, Nivu Academy & Nunnari Labs

Surveillance



Author - Navaneeth Malingan, Nivu Academy & Nunnari Labs

Other few Application

- Optical character recognition (OCR)
- Machine inspection
- Retail (e.g. automated checkouts)
- 3D model building (photogrammetry)
- Medical imaging
- Automotive safety
- Match move (e.g. merging CGI with live actors in movies)
- Motion capture (mocap)
- Surveillance
- Fingerprint recognition and biometrics

Other Cool Computer Vision Applications using Deep Learning

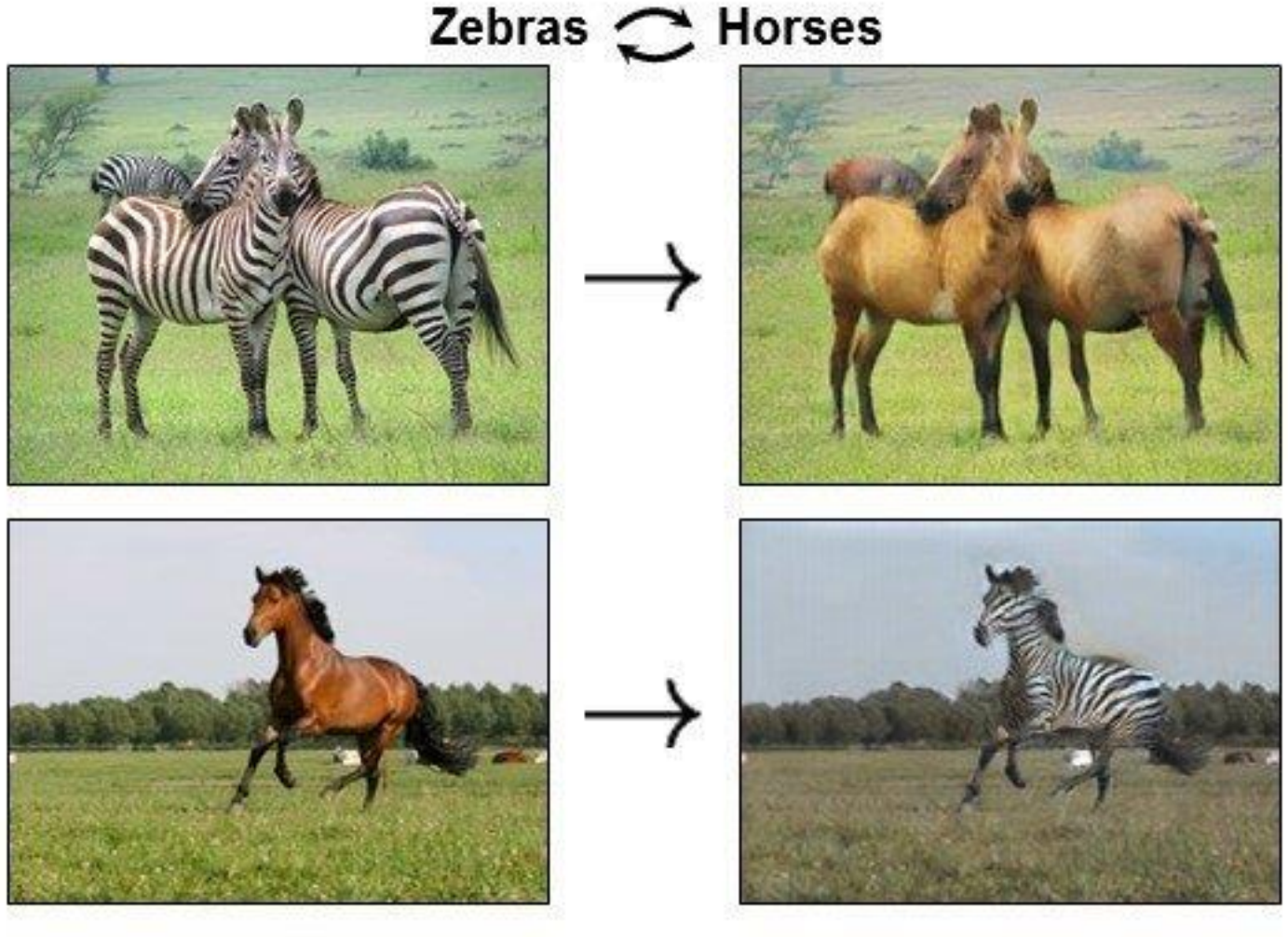
Author - Navaneeth Malingan, Nivu Academy & Nunnari Labs

Deep Fakes


Face2Face
Source Actor
Real-time Reenactment
Reenactment Result
BloombergQuickTake
Target Actor

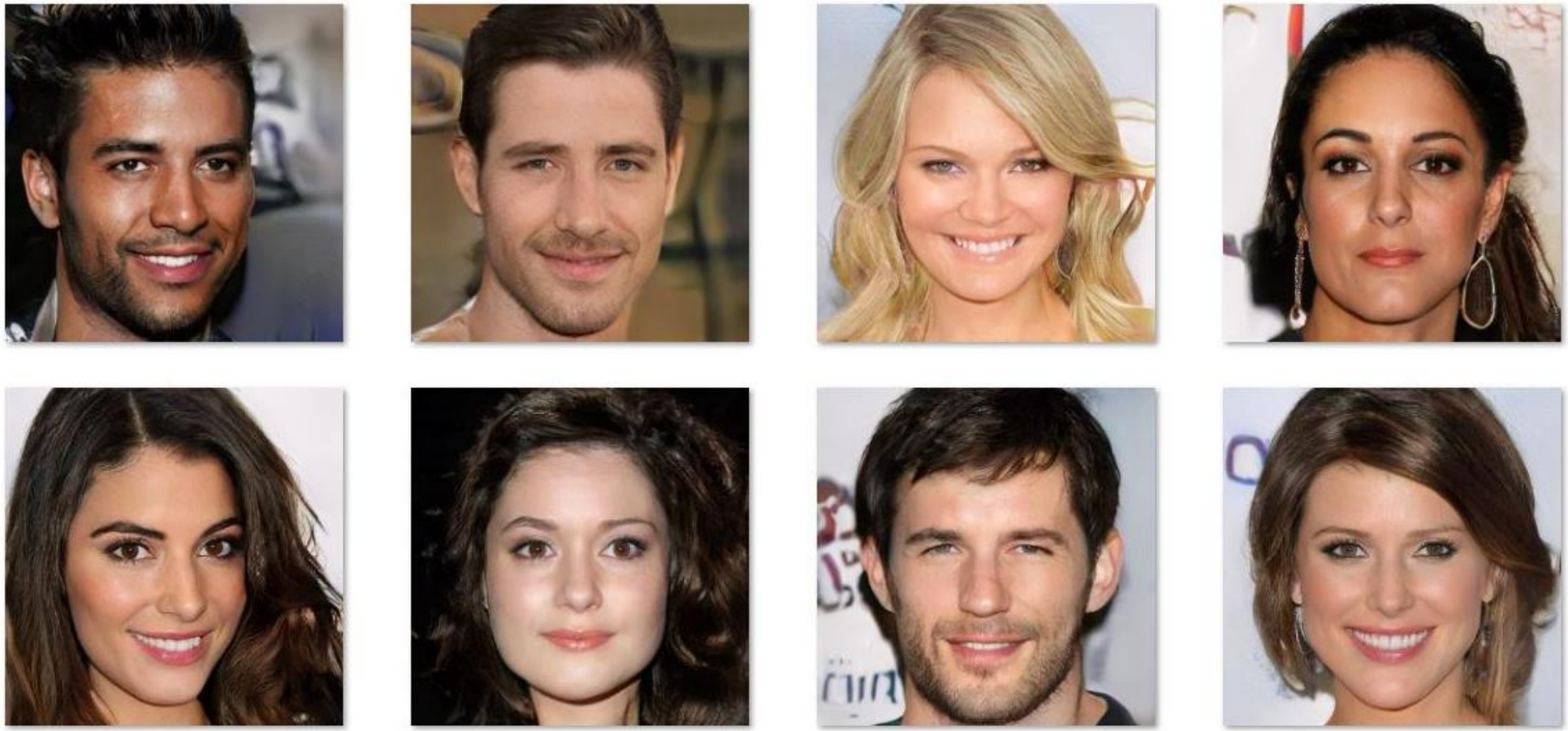
Pose Transfer

CycleGans are used to convert one source of domain to another source and vice versa



Style Transfer

GANs can create fake celebrity faces



Deep Learning Revolution

Author - Navaneeth Malingan, Nivu Academy & Nunnari Labs

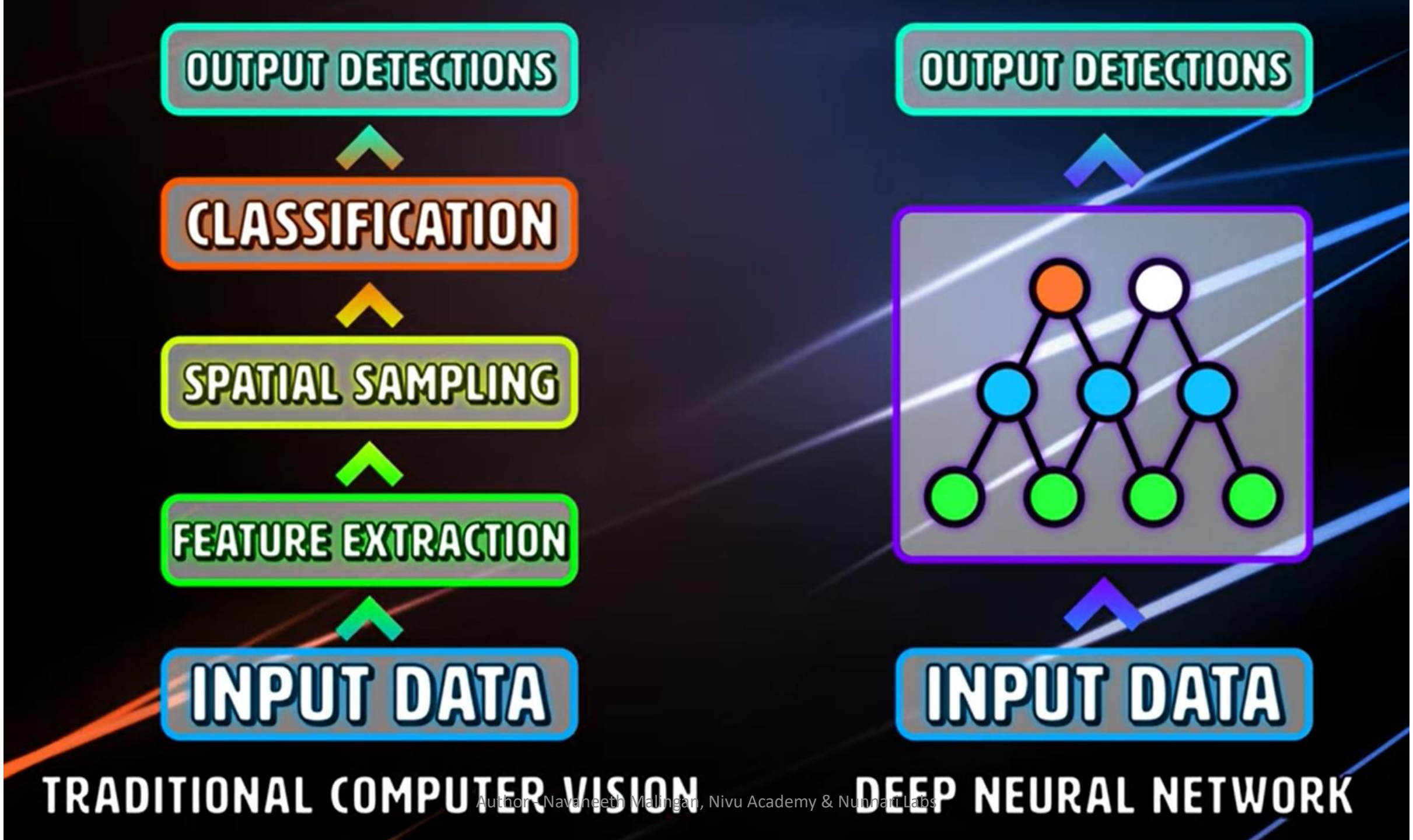
Deep Learning

Until recently, computer vision only worked in limited capacity.

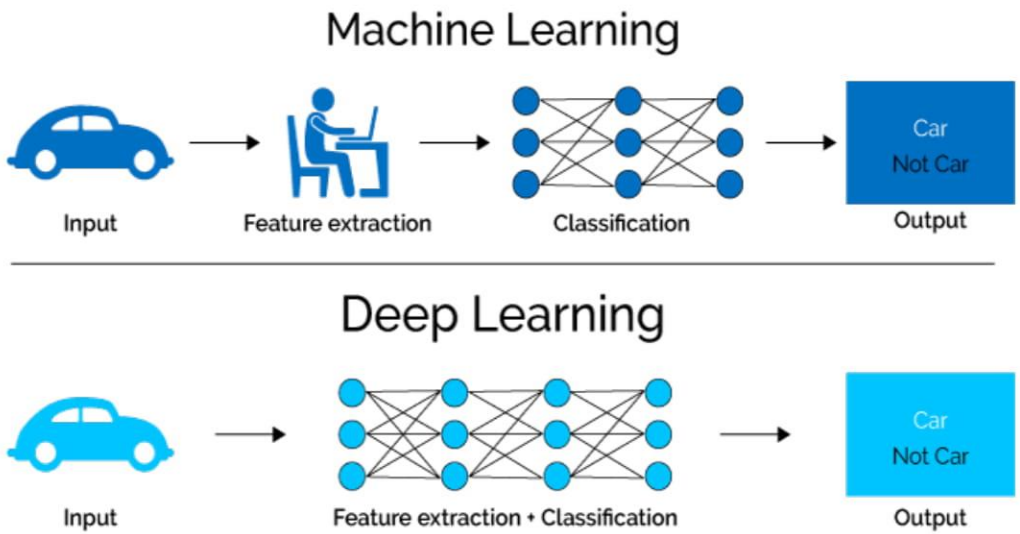
Thanks to advances in artificial intelligence and innovations in deep learning and neural networks, the field has been able to take great leaps in recent years and has been able to surpass humans in some tasks related to detecting and labeling objects.

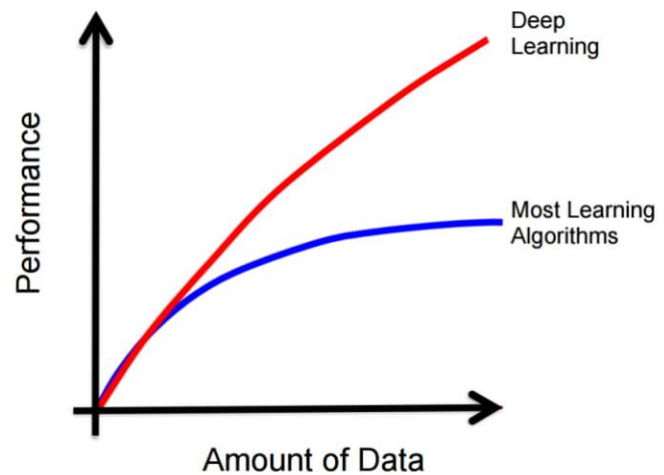
Traditional Computer Vision VS Deep Learning Approach

Author - Navaneeth Malingan, Nivu Academy & Nunnari Labs



Why Deep Learning? **Scalable** Machine Learning



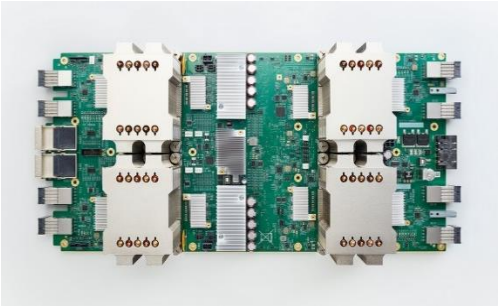


Why AI Now?

Author - Navaneeth Malingan, Nivu Academy & Nunnari Labs

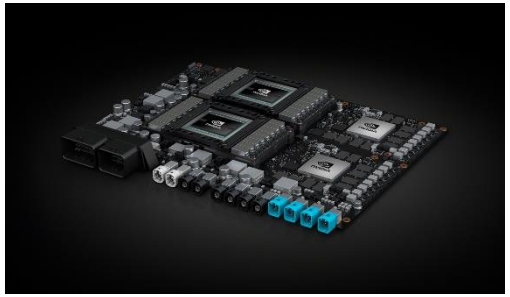
Author - Navaneeth Malingan, Nivu Academy & Nunnari Labs

ML Hardware



Google TPU

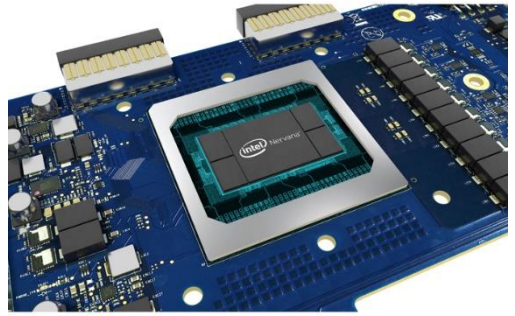
Cloud TPU is designed to run cutting-edge machine learning models with AI services on Google Cloud. And its custom high-speed network offers up to 11.5 petaflops of performance in a single pod.



Nvidia GPU

NVIDIA CUDA is the world leader in high-performance parallel computing on GPUs.

Even before the creation of CUDA, NVIDIA was a pioneer in the creation of innovative algorithms and applications to bring GPU Computing to the world.



Intel ML Kit

Harnessing silicon designed specifically for AI, end to end solutions that broadly span from the data center to the edge, and tools that enable customers to quickly deploy and scale up, Intel AI is inside AI and leading the next evolution of compute.

StatsModels
Statistics in Python

Seaborn

pandas

$$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$$

xarray

scikit learn

scikit-image
image processing in python

NumPy

SciPy

matplotlib

jupyter

python™

IP[y]:
IPython

Author - Navaneeth Malingan, Nivu Academy & Nunnari Labs

Computer Vision Tools and Library

- OpenCV
- Dlib
- Imutils
- TensorFlow
- PyTorch
- Matlab
- Scipy and Numpy

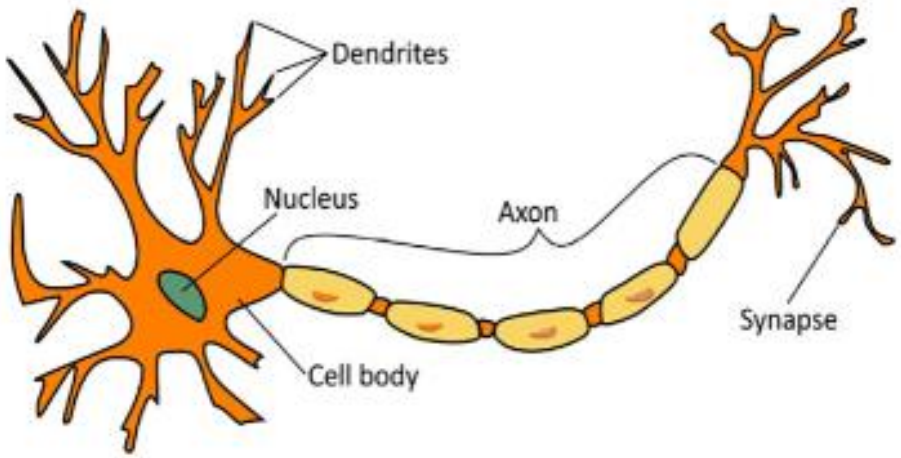
Computer Vision as Service:

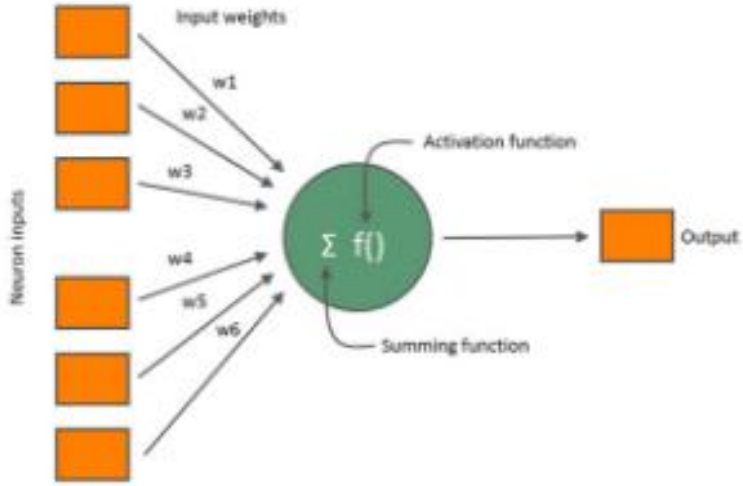
- **Google Cloud and Mobile Vision APIs**
- **Amazon Rekognition**
- **Microsoft Azure Computer Vision API**

- All these services enables developers to perform image processing by encapsulating powerful machine learning models in a simple REST API that can be called in an application to add image and video analysis to your applications.
- The service can identify objects, text, people, scenes and activities, and it can also detect inappropriate content, apart from providing highly accurate facial analysis and facial recognition for sentiment analysis and OCR.

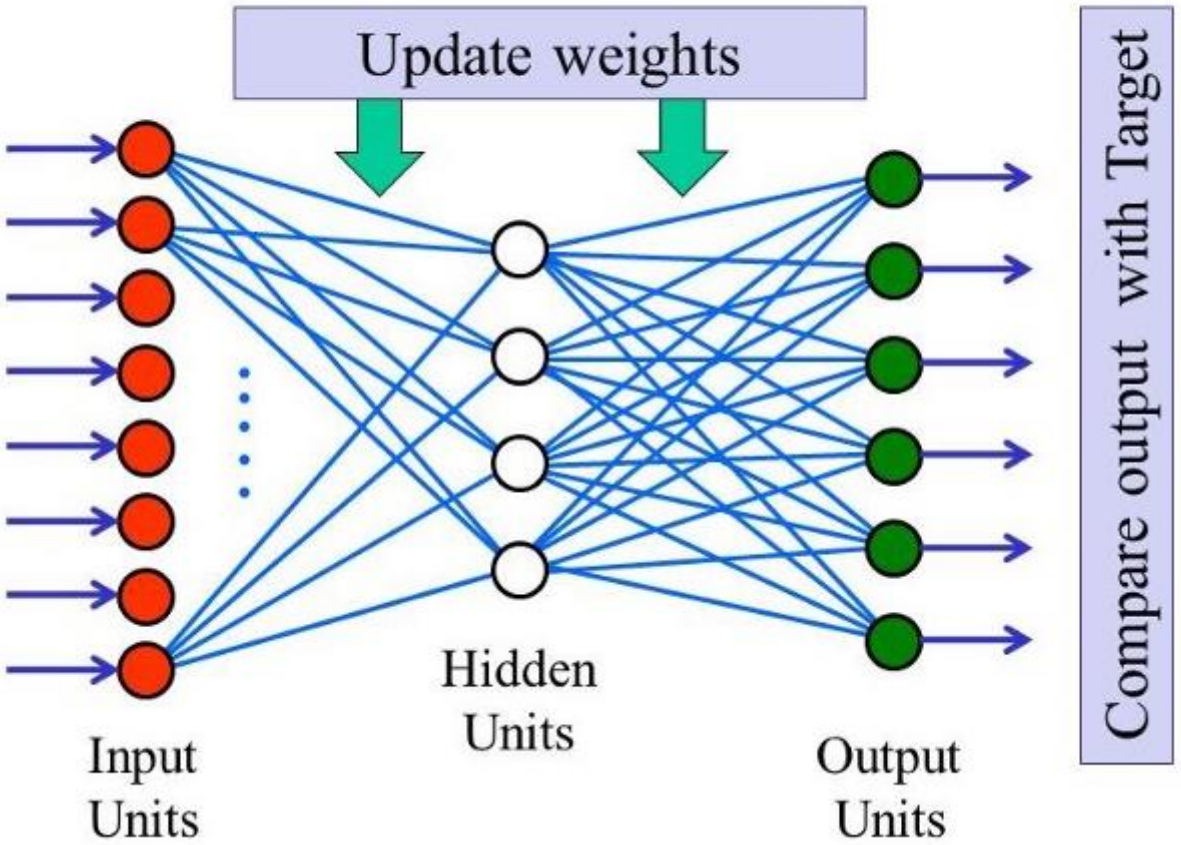
Neural Network

Author - Navaneeth Malingan, Nivu Academy & Nunnari Labs

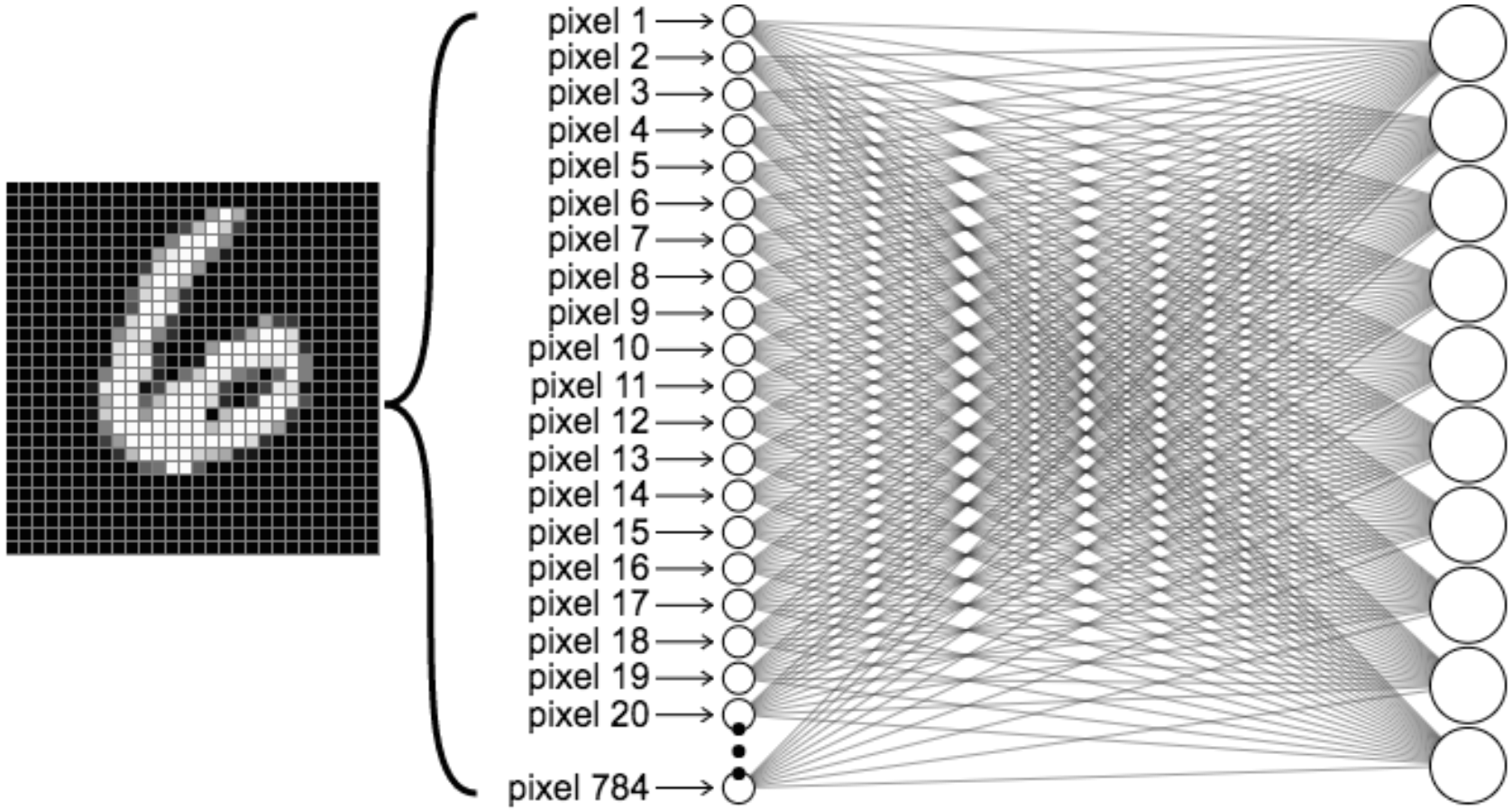

Dendrites
Nucleus
Axon
Synapse
Cell body


Input weights
w1
w2
w3
w4
w5
w6
Neuron inputs
Activation function
Σ f()
Output
Summing function

Artificial Neural Network



MNIST Feed Forward

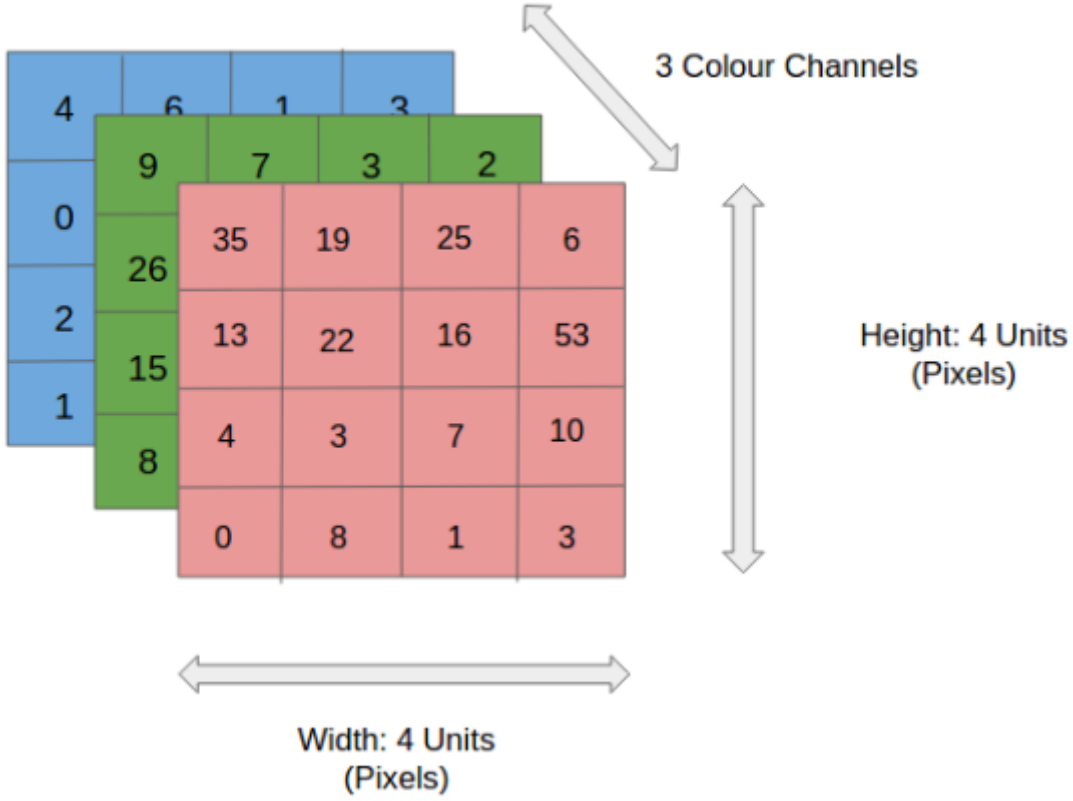


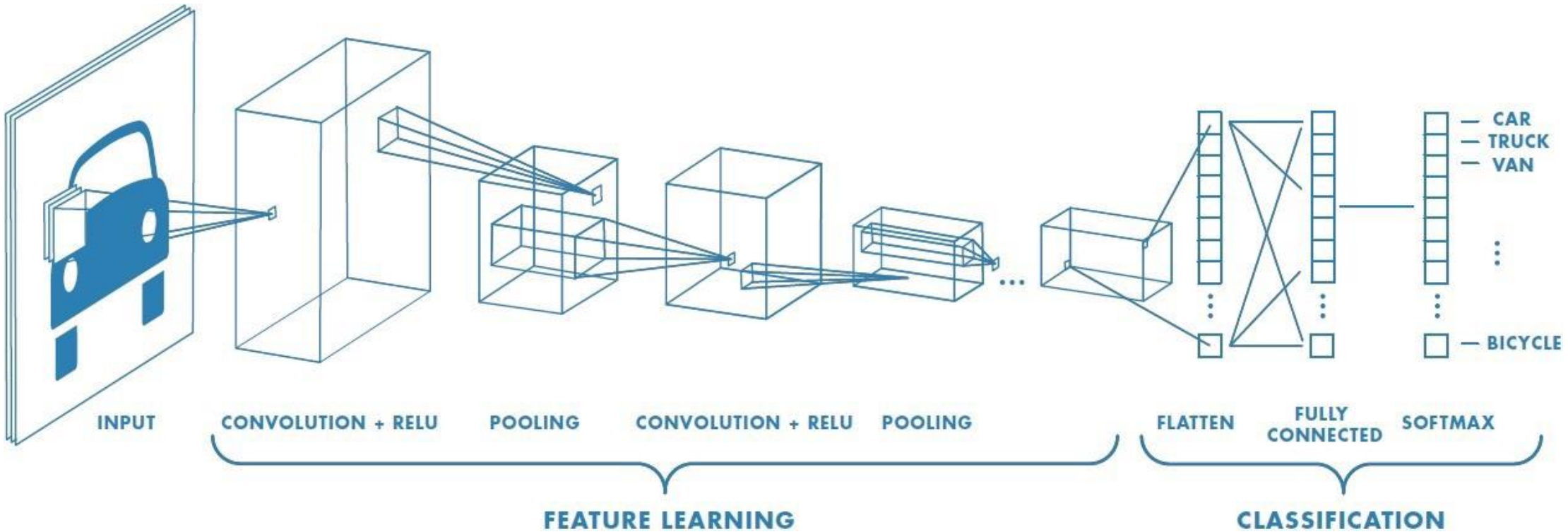
CNN

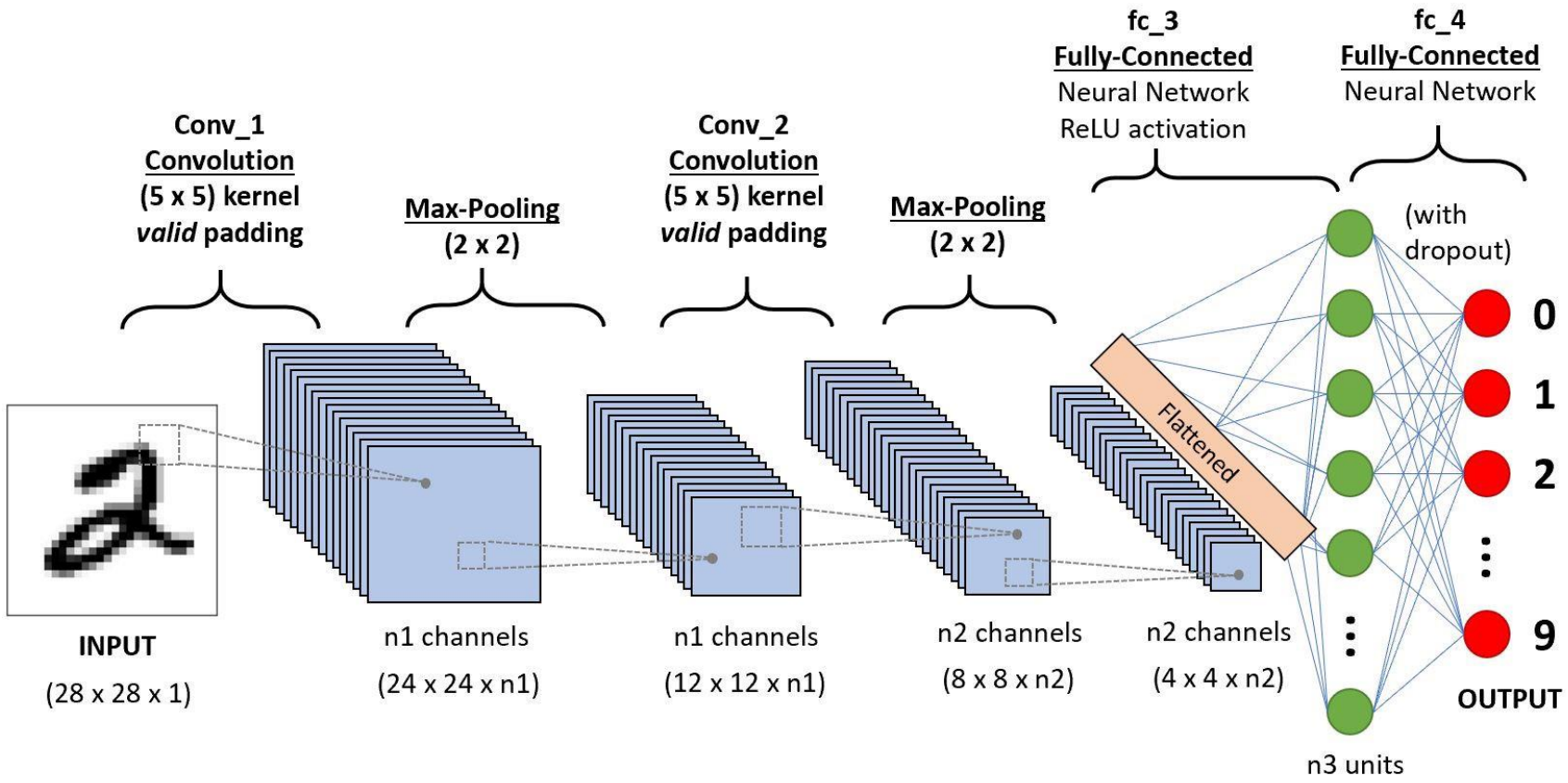
Convolutional Neural Network

Author - Navaneeth Malingan, Nivu Academy & Nunnari Labs

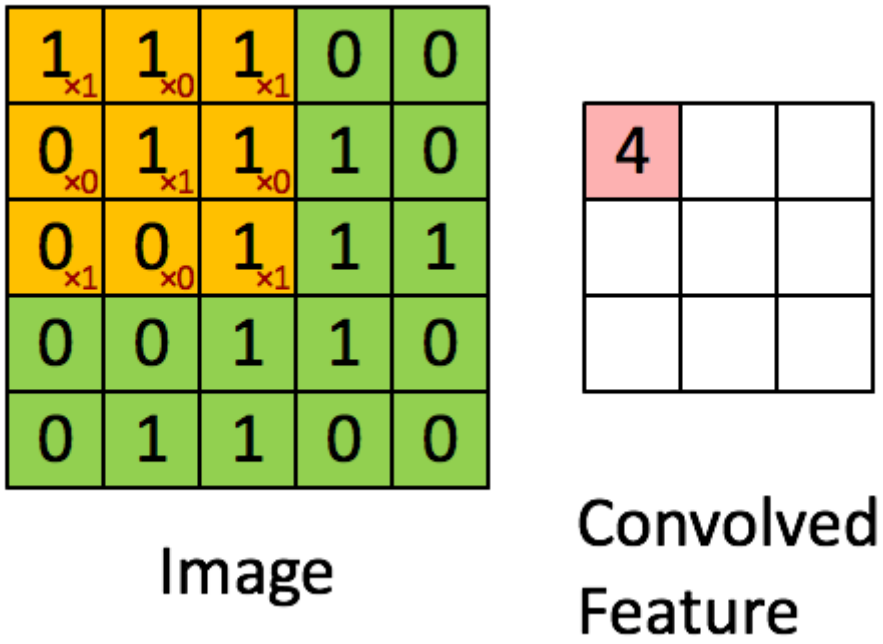
Color Spaces : Grayscale, RGB, HSV, CMYK, etc.



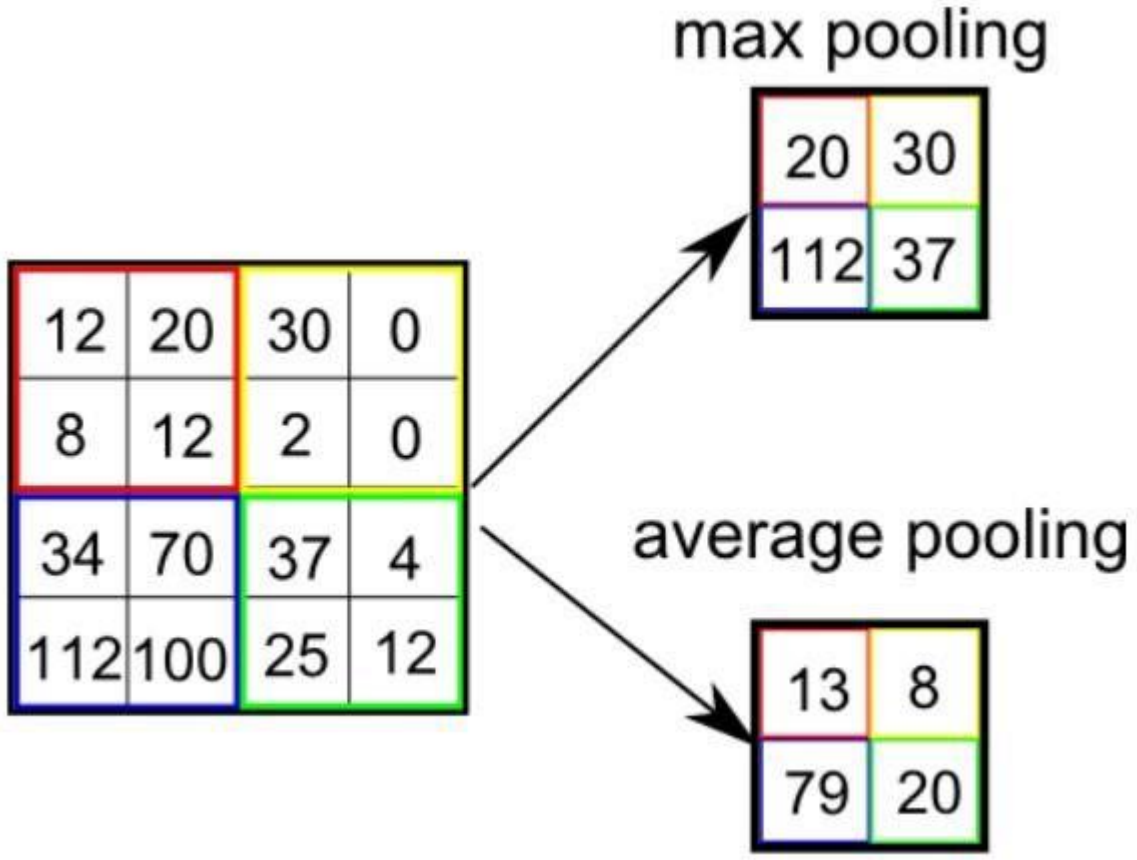

CAR
TRUCK
VAN
BICYCLE
INPUT
CONVOLUTION + RELU
POOLING
CONVOLUTION + RELU
POOLING
FLATTEN
FULLY CONNECTED
SOFTMAX
FEATURE LEARNING
CLASSIFICATION

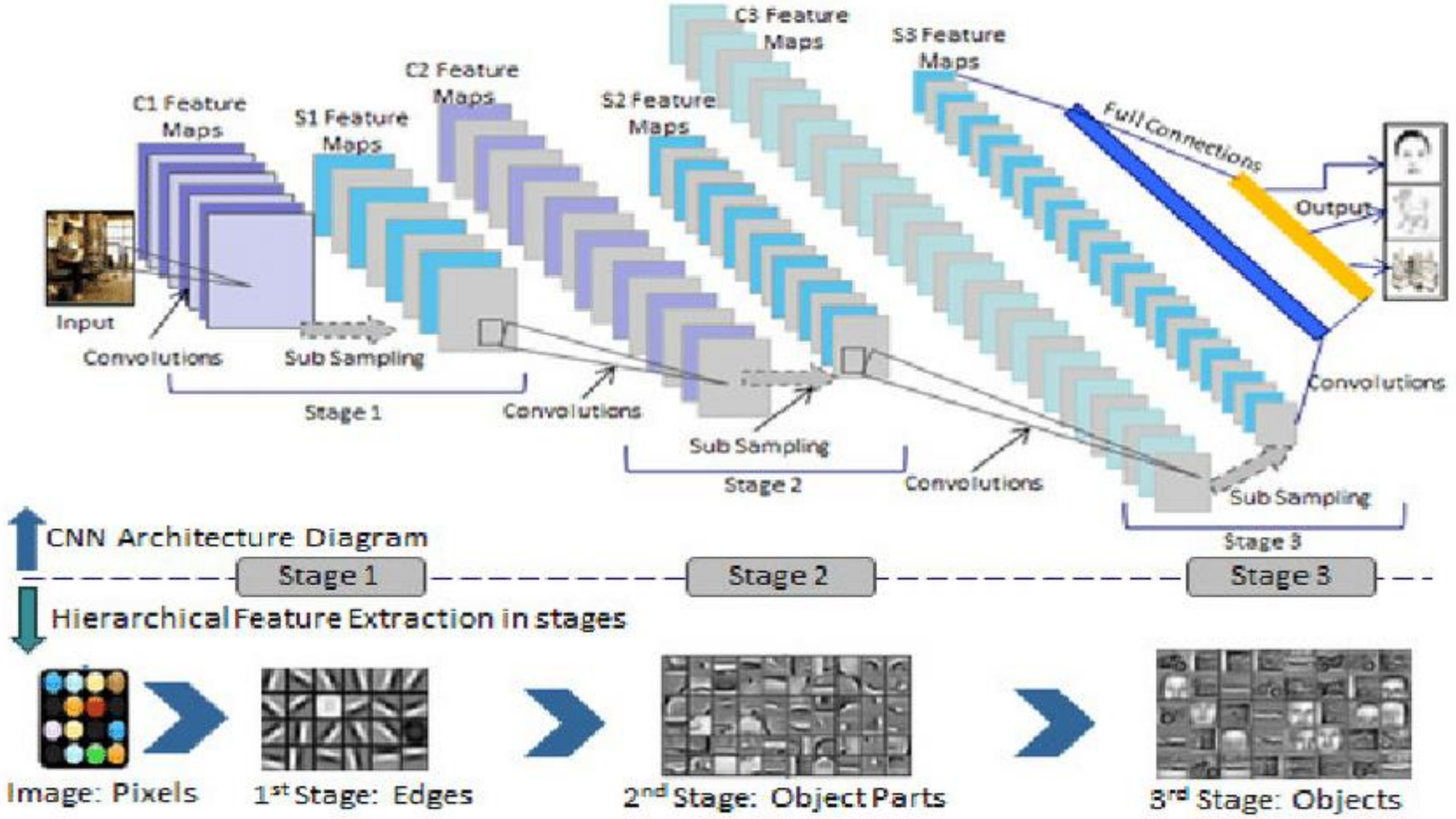

fc_3
Fully-Connected
Neural Network
ReLU activation
fc_4
Fully-Connected
Neural Network
Conv_1
Convolution
(5 x 5) kernel
valid padding
Max-Pooling
(2 x 2)
Conv_2
Convolution
(5 x 5) kernel
valid padding
Max-Pooling
(2 x 2)
(with
dropout)
Flattened
0
1
2
9
INPUT
(28 x 28 x 1)
n1 channels
(24 x 24 x n1)
n1 channels
(12 x 12 x n1)
n2 channels
(8 x 8 x n2)
n2 channels
(4 x 4 x n2)
n3 units
OUTPUT

Convolution Layer — The Kernel

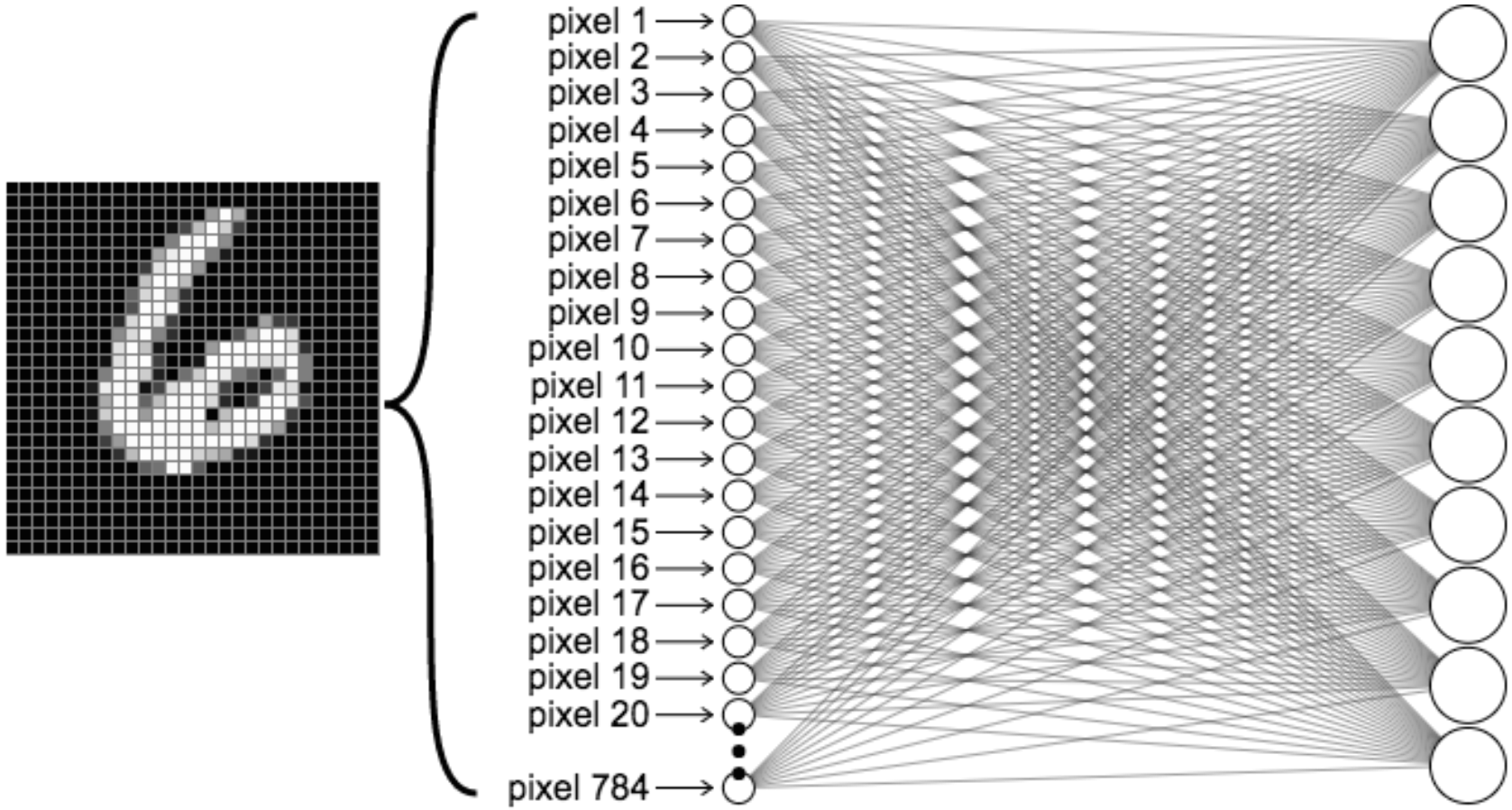


Pooling (Reduces Dimensions and Removes Noise)




C1 Feature Maps
S1 Feature Maps
C2 Feature Maps
S2 Feature Maps
C3 Feature Maps
S3 Feature Maps
Full Connections
Output
Input
Convolutions
Sub Sampling
Stage 1
Convolutions
Sub Sampling
Stage 2
Convolutions
Convolutions
Sub Sampling
Stage 3
CNN Architecture Diagram
Stage 1
Stage 2
Stage 3
Hierarchical Feature Extraction in stages
Image: Pixels
1st Stage: Edges
2nd Stage: Object Parts
3rd Stage: Objects

MNIST Feed Forward



MNIST CNN

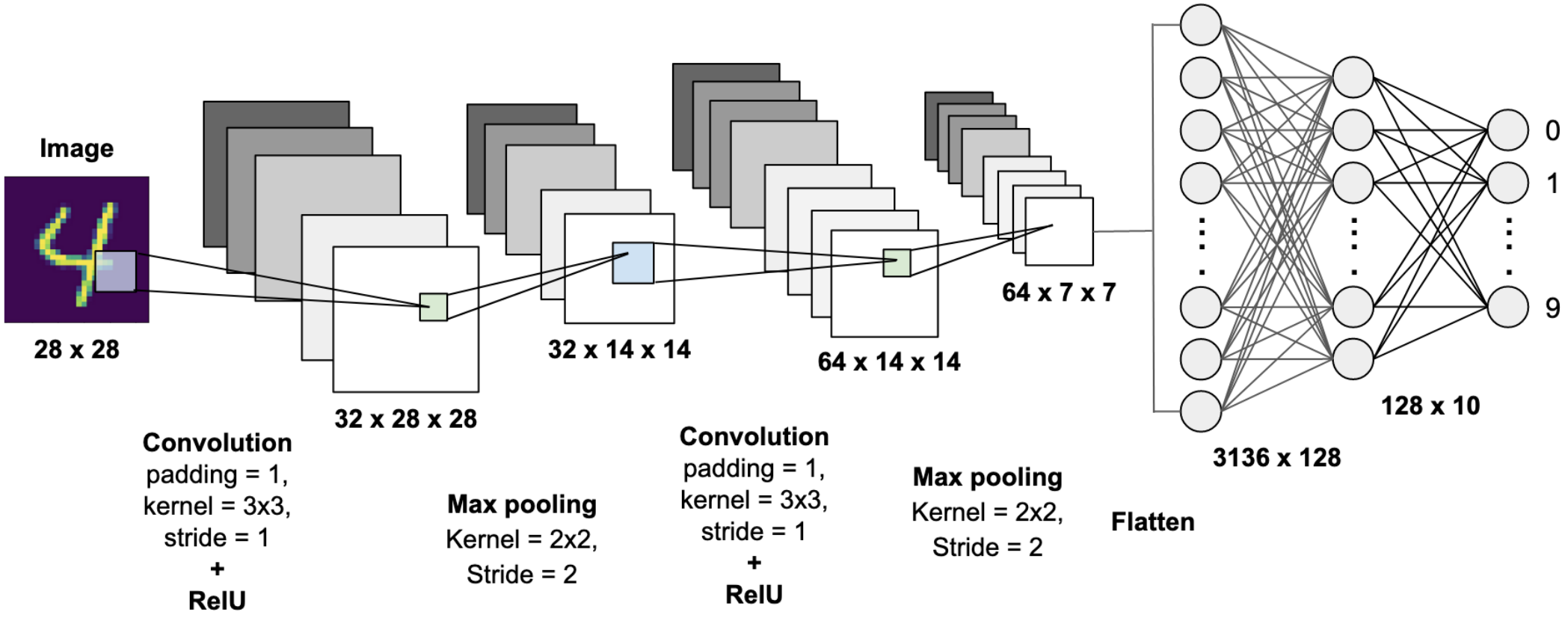
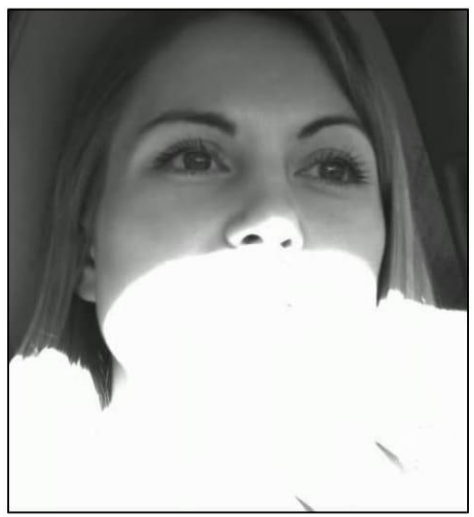


Image
28 x 28
32 x 28 x 28
Convolution
padding = 1,
kernel = 3x3,
stride = 1
+
RelU
32 x 14 x 14
Max pooling
Kernel = 2x2,
Stride = 2
64 x 14 x 14
Convolution
padding = 1,
kernel = 3x3,
stride = 1
+
RelU
64 x 7 x 7
Max pooling
Kernel = 2x2,
Stride = 2
3136 x 128
Flatten
128 x 10
0
1
9

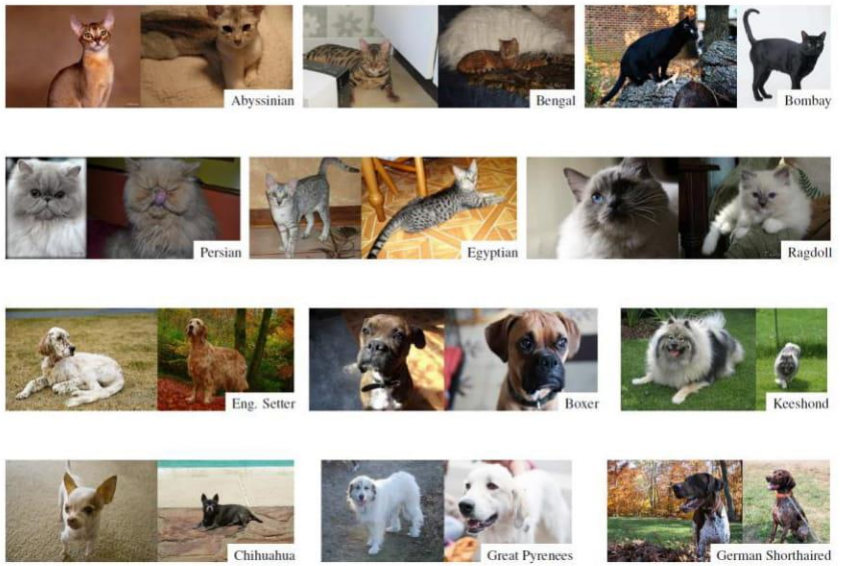
The Challenge of Deep Learning

- Ask the right question and know what the answer means:
  image classification ≠ scene understanding

- Select, collect, and organize the right data to train on:
  photos ≠ synthetic ≠ real-world video frames



Pure Perception is Hard



Visual Understanding is Harder

Examples of what we can't do well:

- Mirrors
- Sparse information
- 3D Structure
- Physics
- What's on peoples' minds?
- What happens next?
- Humor

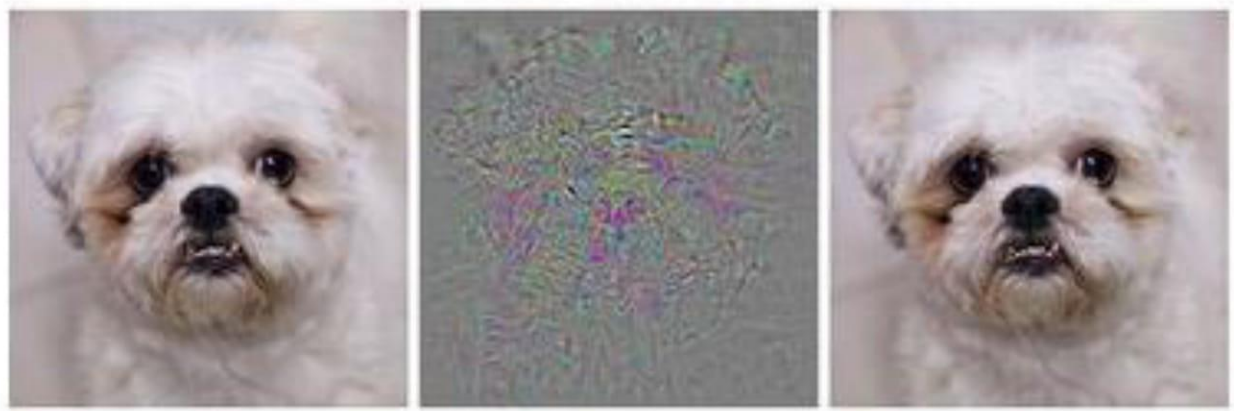
Deep Learning:

Our intuition about what's "hard" is flawed (in complicated ways)

Visual perception: 540,000,000 years of data

Bipedal movement: 230,000,000 years of data

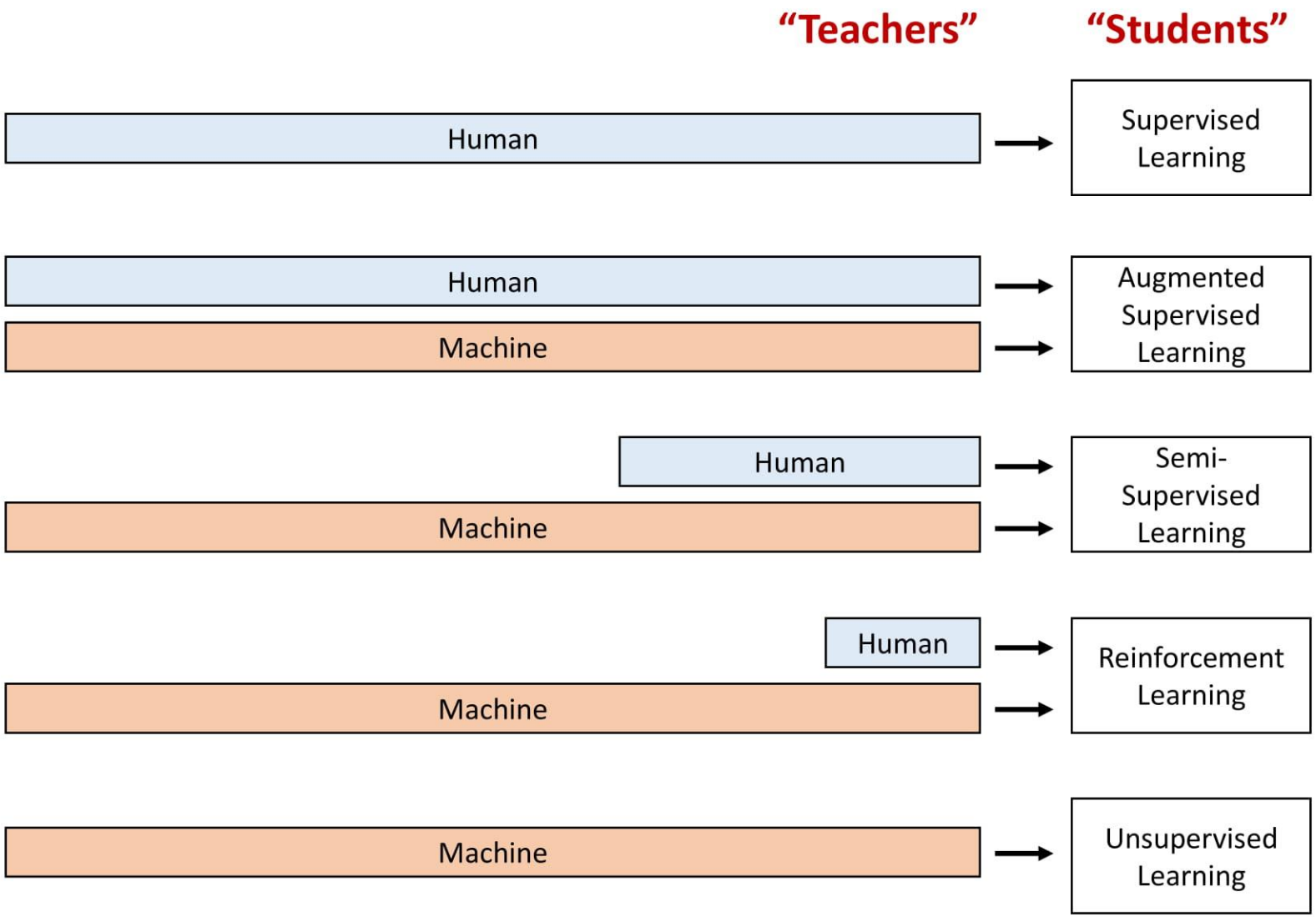
Abstract thought: 100,000 years of data



Prediction: **Dog** + Distortion Prediction: **Ostrich**

"Encoded in the large, highly evolve sensory and motor portions of the human brain is a **billion years of experience** about the nature of the world and how to survive in it…. Abstract thought, though, is a new trick, perhaps less than **100 thousand years** old. We have not yet mastered it. It is not all that intrinsically difficult; it just seems so when we do it."
- *Hans Moravec, Mind Children (1988)*

Deep Learning from Human and Machine



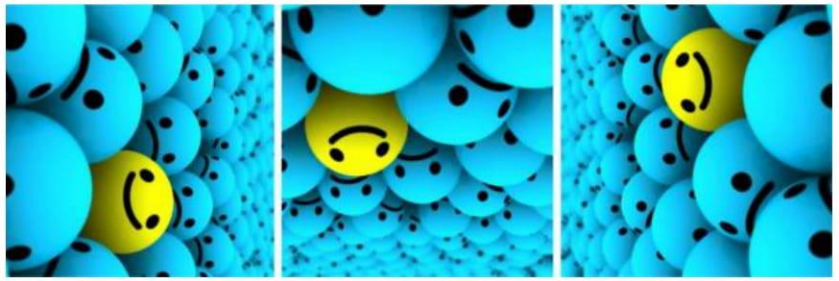
Data Augmentation

Crop:

Flip:

Scale:

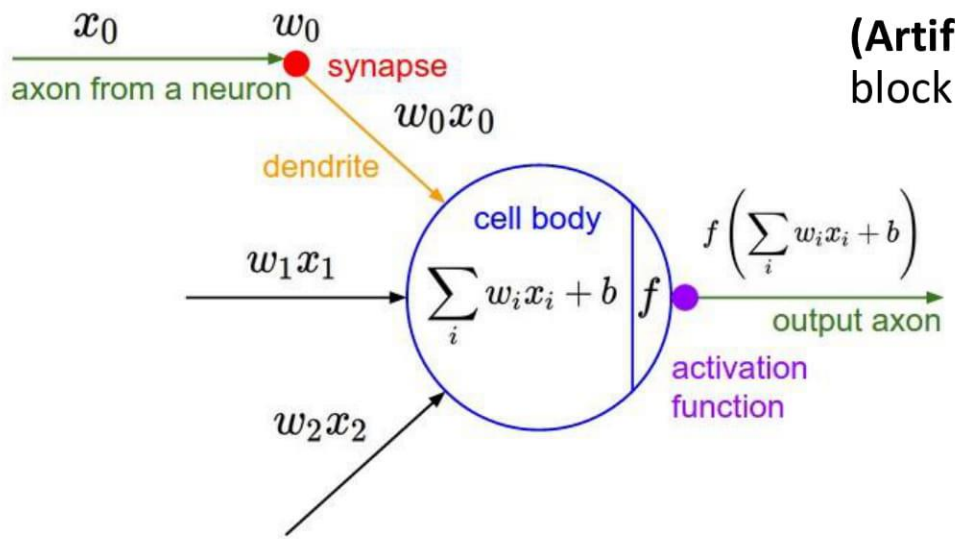
Rotate:



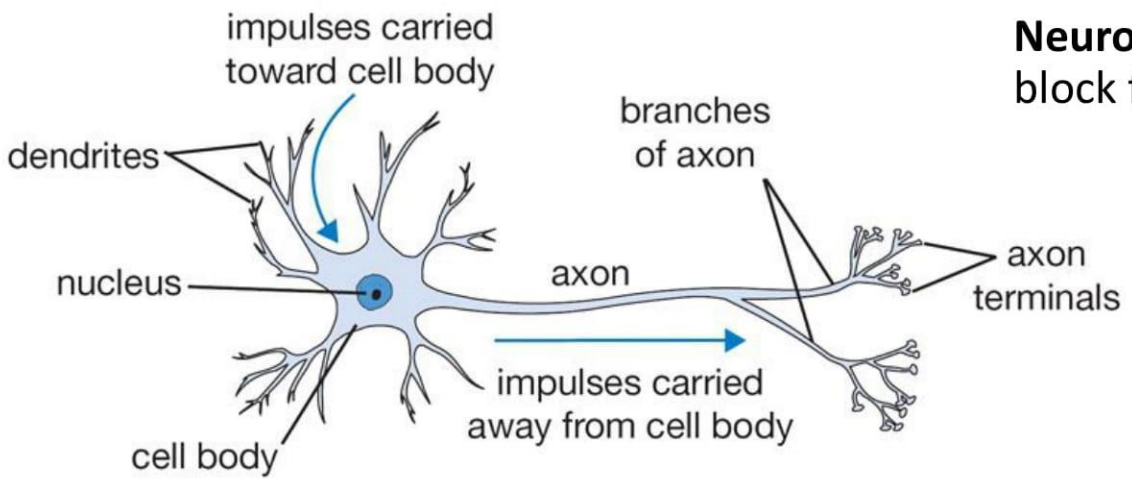
Translation:

Noise:

Neuron: Biological Inspiration for Computation

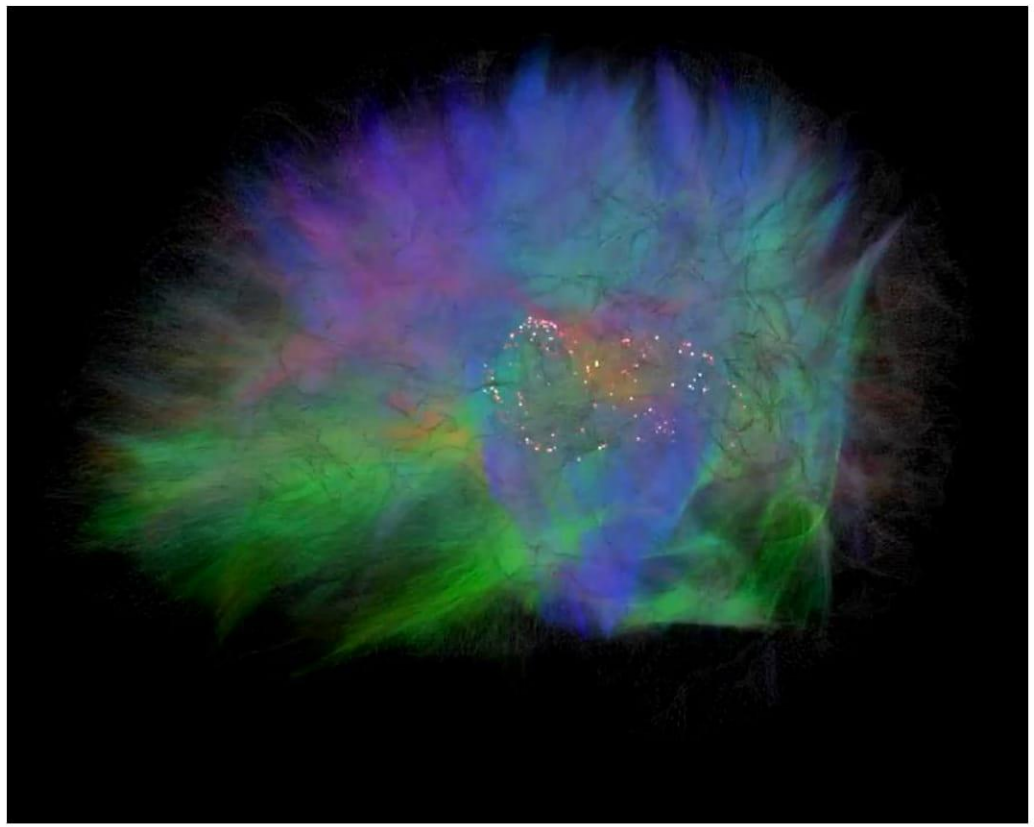


(Artificial) Neuron: computational building block for the "neural network"



Neuron: computational building block for the brain

Biological and Artificial Neural Networks



Human Brain

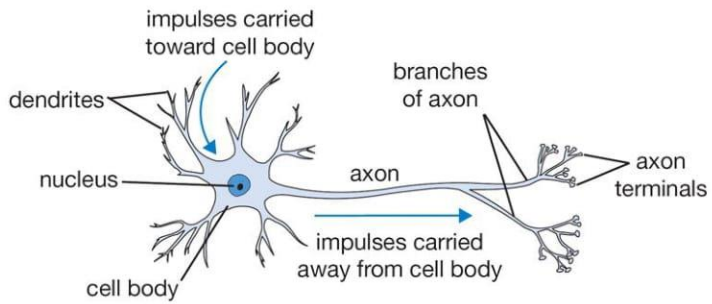
- Thalamocortical system:
  3 million neurons
  476 million synapses
- Full brain:
  100 billion neurons
  1,000 trillion synapses

Artificial Neural Network

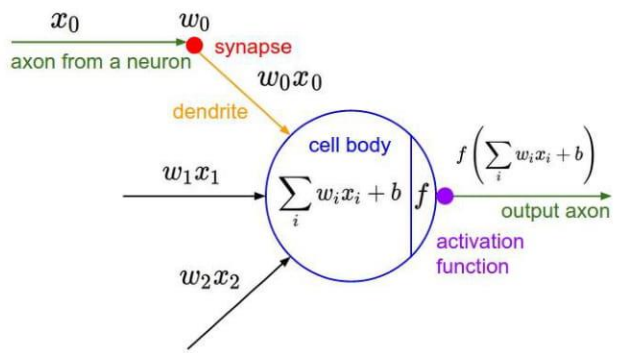
- ResNet-152:
  60 million synapses

Human brains have ~10,000,000 times synapses
than artificial neural networks.

Neuron: Biological Inspiration for Computation



- **Neuron:** computational building block for the brain



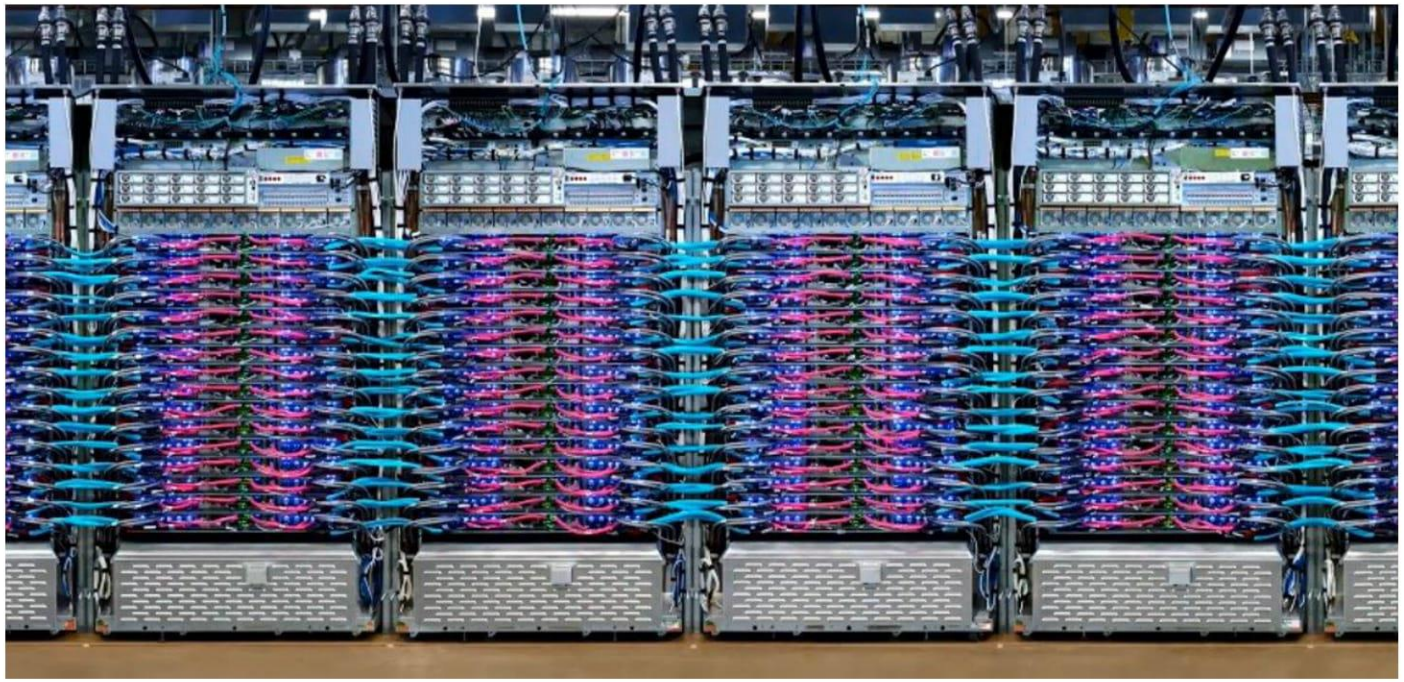
- **(Artificial) Neuron:** computational building block for the "neural network"

Key Difference:

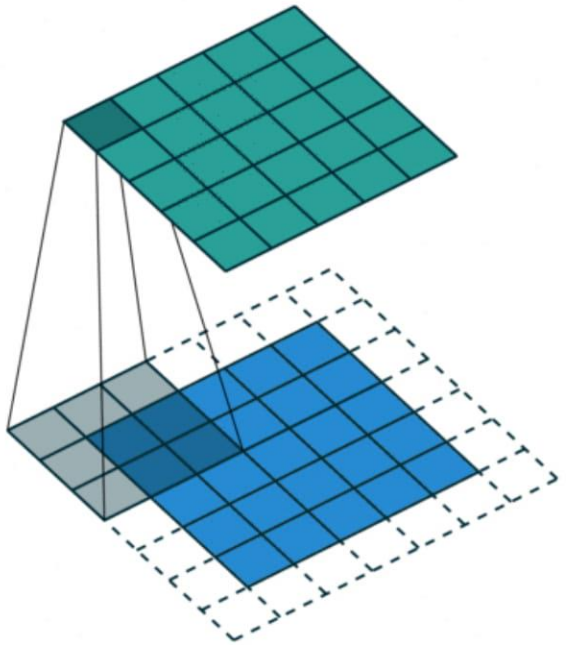
- **Parameters:** Human brains have ~10,000,000 times synapses than artificial neural networks.
- **Topology:** Human brains have no "layers". **Async:** The human brain works asynchronously, ANNs work synchronously.
- **Learning algorithm**: ANNs use gradient descent for learning. We don't know what human brains use
- **Power consumption:** Biological neural networks use very little power compared to artificial networks
- **Stages:** Biological networks usually never stop learning. ANNs first train then test.

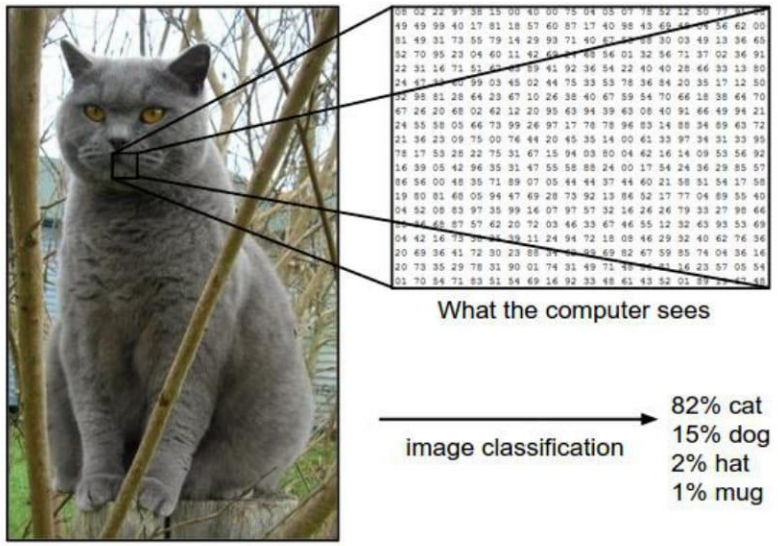
Compute Hardware

- **CPU** – serial, general purpose, everyone has one
- **GPU** – parallelizable, still general purpose
- **TPU** – custom ASIC (Application-Specific Integrated Circuit) by Google, specialized for machine learning, low precision

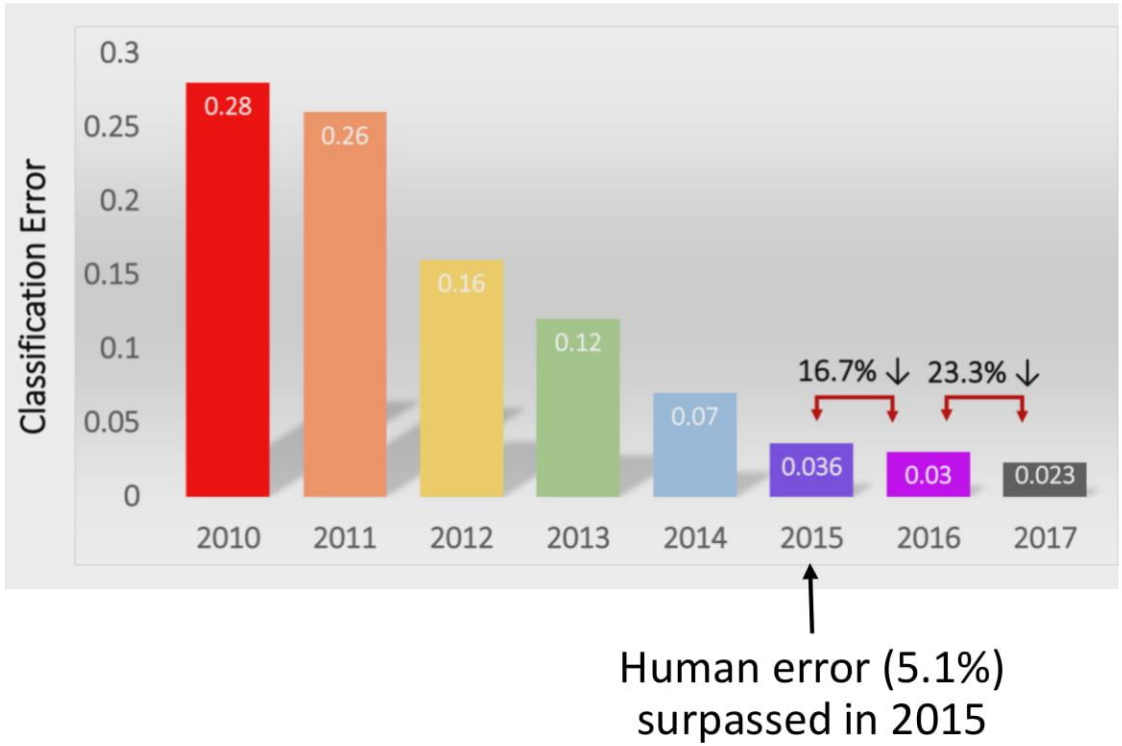


Convolutional Neural Networks: Image Classification





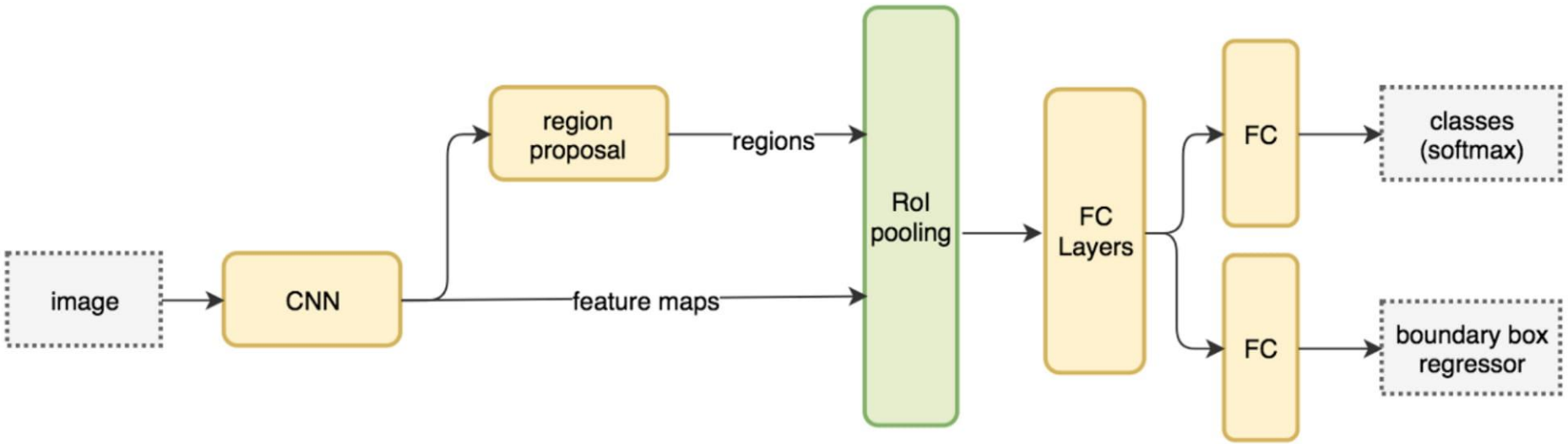
- Convolutional filters: take advantage of spatial invariance



- **AlexNet (2012): First CNN (15.4%)**
  - 8 layers
  - 61 million parameters
- **ZFNet (2013): 15.4% to 11.2%**
  - 8 layers
  - More filters. Denser stride.
- **VGGNet (2014): 11.2% to 7.3%**
  - Beautifully uniform:
    3x3 conv, stride 1, pad 1, 2x2 max pool
  - 16 layers
  - 138 million parameters
- **GoogLeNet (2014): 11.2% to 6.7%**
  - Inception modules
  - 22 layers
  - 5 million parameters
    (throw away fully connected layers)
- **ResNet (2015): 6.7% to 3.57%**
  - More layers = better performance
  - 152 layers
- **CUImage (2016): 3.57% to 2.99%**
  - Ensemble of 6 models
- **SENet (2017): 2.99% to 2.251%**
  - Squeeze and excitation block: network is allowed to adaptively adjust the weighting of each feature map in the convolutional block.

Object Detection / Localization

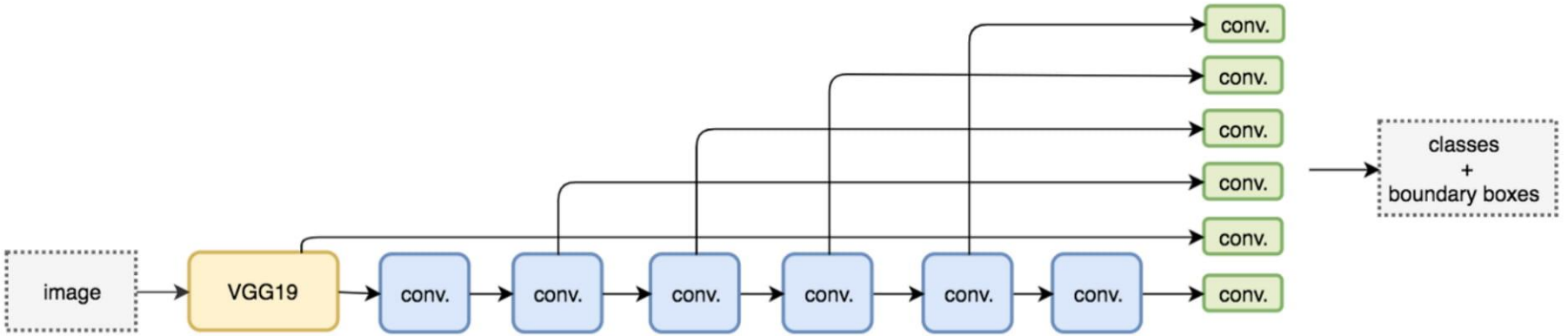
Region-Based Methods | Shown: Faster R-CNN

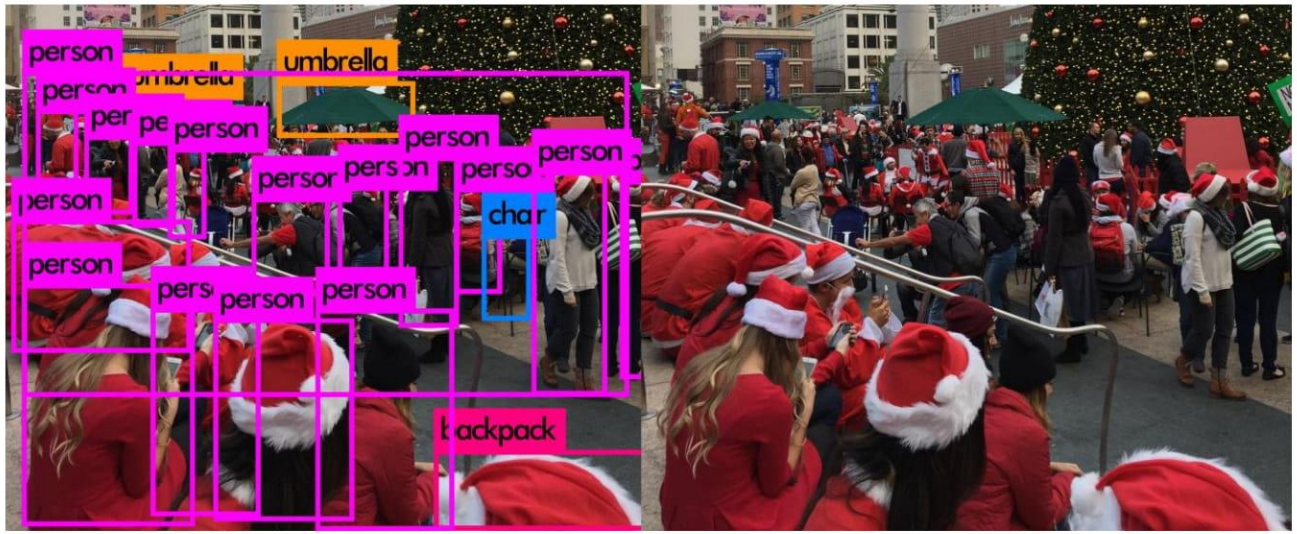



```
ROIs = region_proposal(image)
for ROI in ROIs
    patch = get_patch(image, ROI)
    results = detector(patch)
```

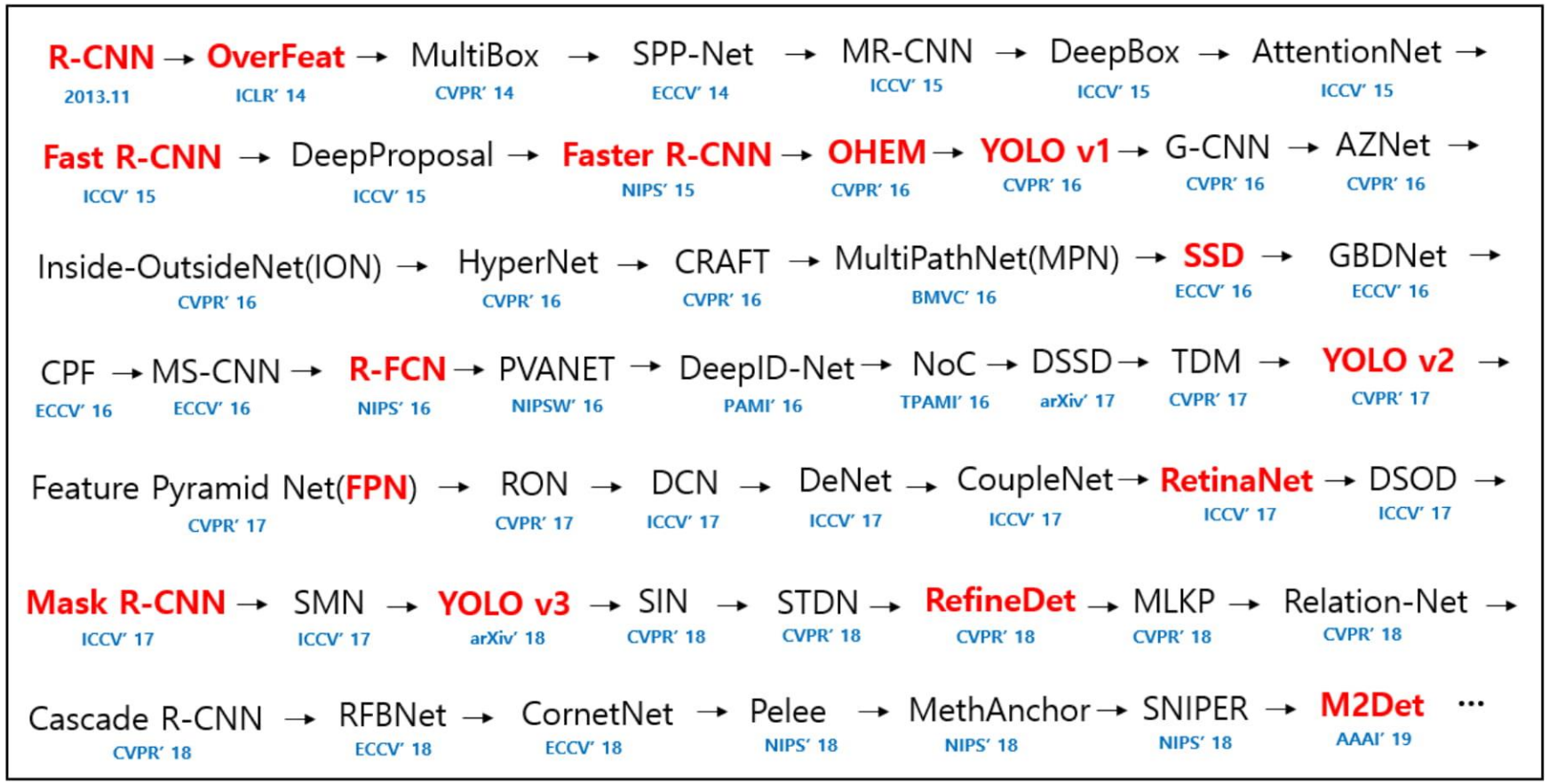
Object Detection / Localization

Single-Shot Methods | Shown: SSD

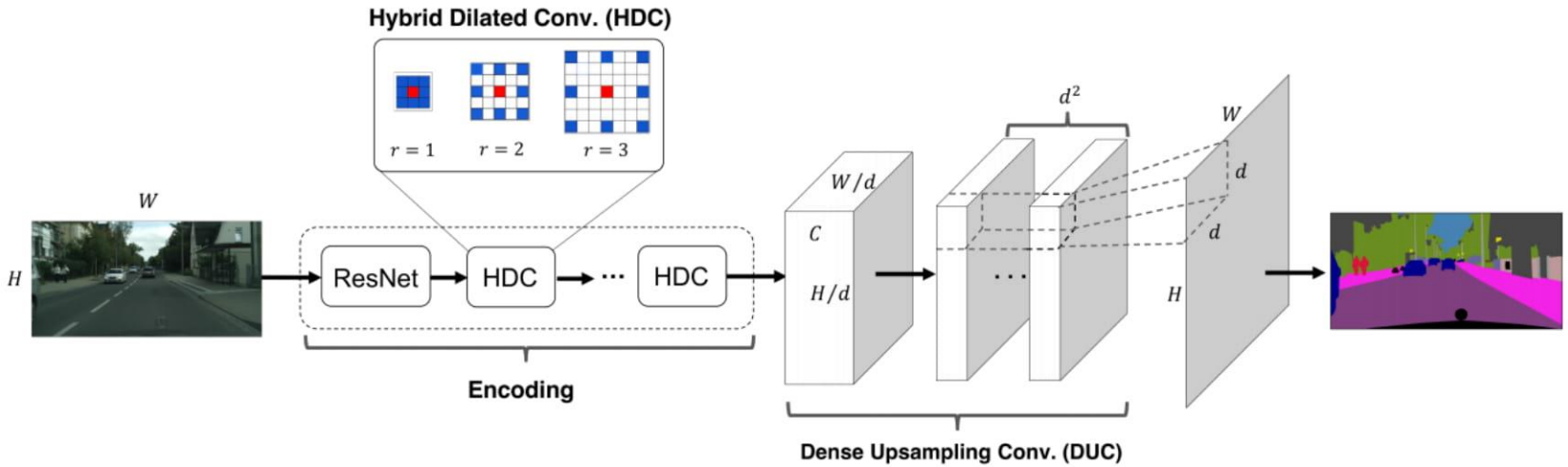




Object Detection: State of the Art Progress



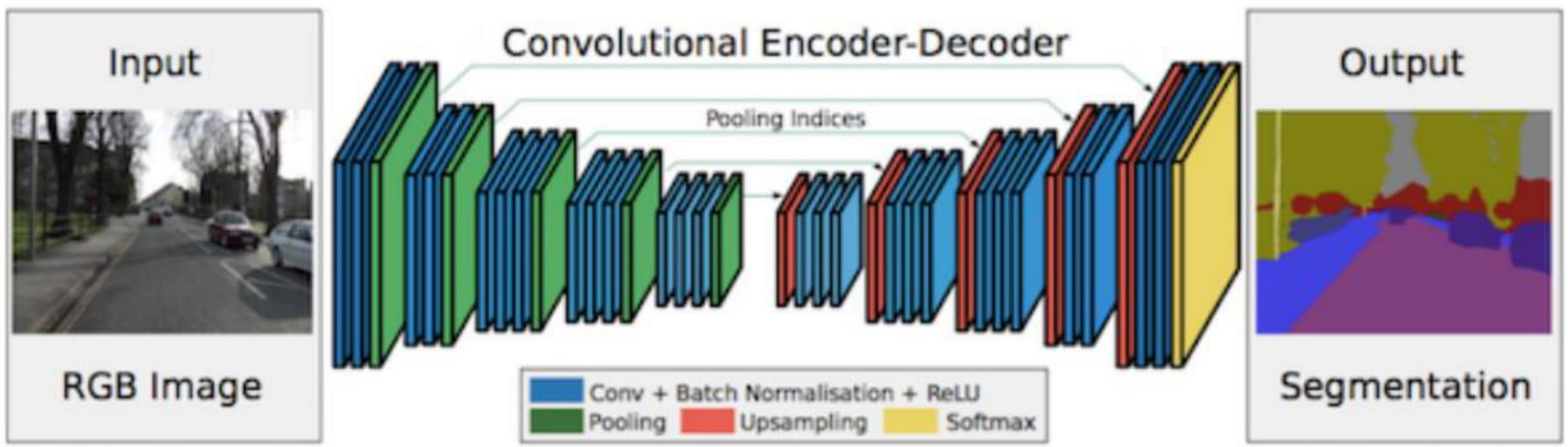
Semantic Segmentation



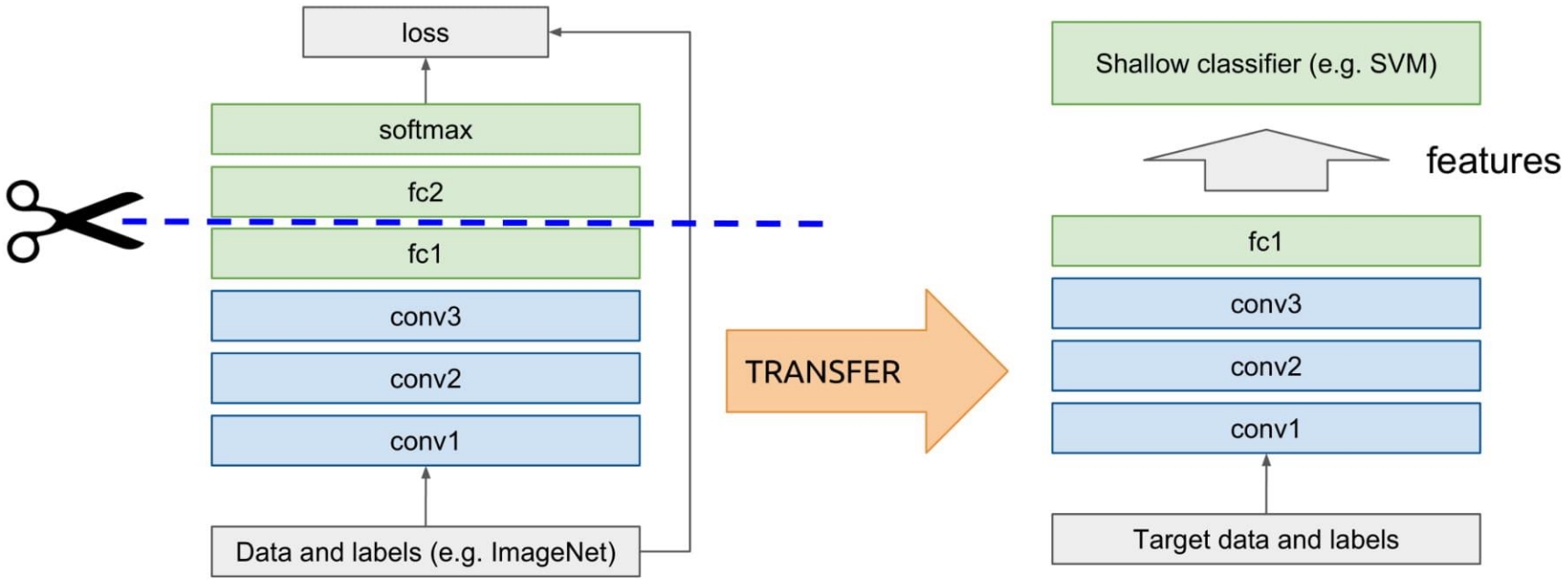
SegNet (Nov 2015)

Paper: "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation"

- Maxpooling indices transferred to decoder to improve the segmentation resolution.

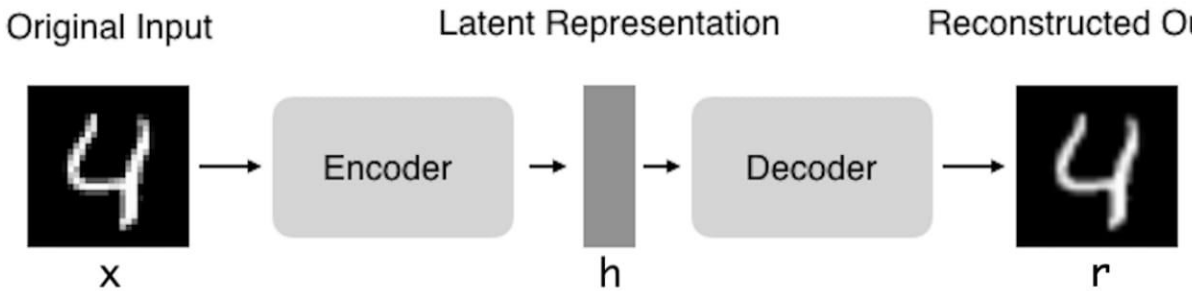


Transfer Learning

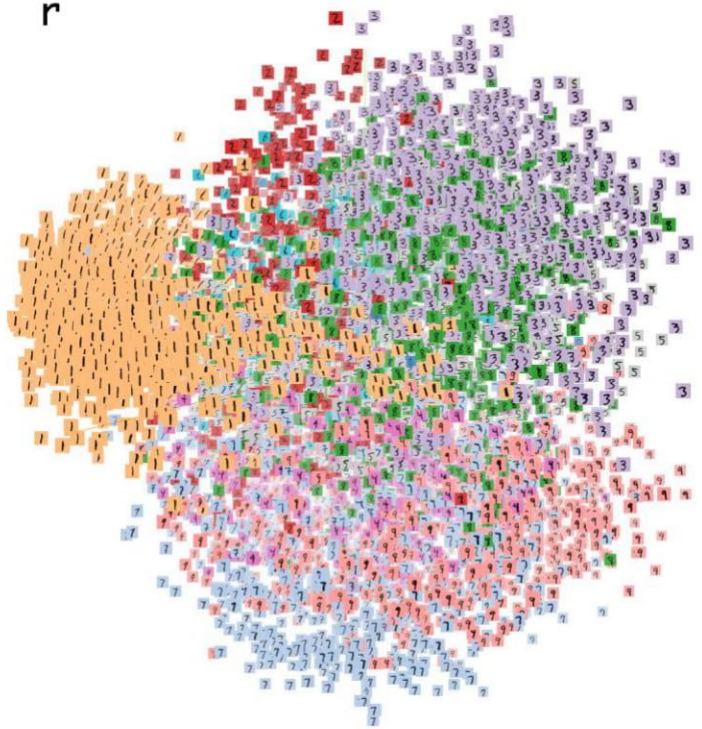


- Fine-tune a pre-trained model
- Effective in many applications: computer vision, audio, speech, natural language processing

Autoencoders



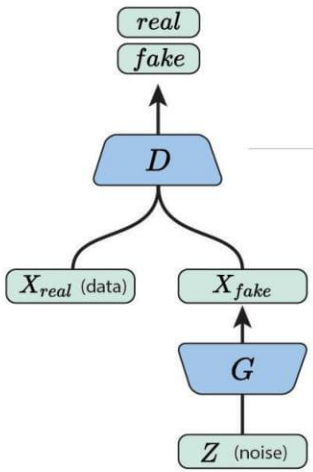
- Unsupervised learning
- Gives embedding
  - Typically better embeddings come from discriminative task



http://projector.tensorflow.org/

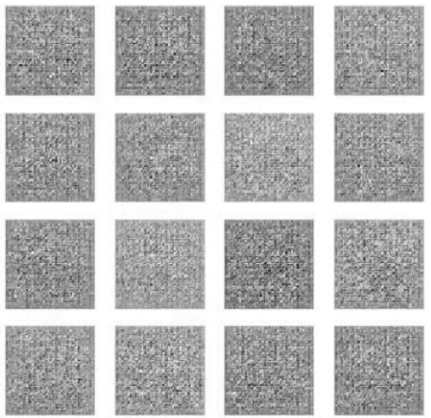
Generative Adversarial Network (GANs)

Generative Adversarial Networks (GANs) are a way to make a generative model by having two neural networks compete with each other.



The **discriminator** tries to distinguish genuine data from forgeries created by the generator.

The **generator** turns random noise into immitations of the data, in an attempt to fool the discriminator.



Progressive GAN
10/2017
1024 x 1024

How do I Start?

Best places to Start

https://github.com/soaicbe/ai_all_resources

An Ultimate Compilation of AI Resources for Mathematics, Machine Learning and Deep Learning.

An Ultimate Compilation of AI Resources for Mathematics, Machine Learning and Deep Learning

Knowledge Not Shared is wasted - Clan Jacobs

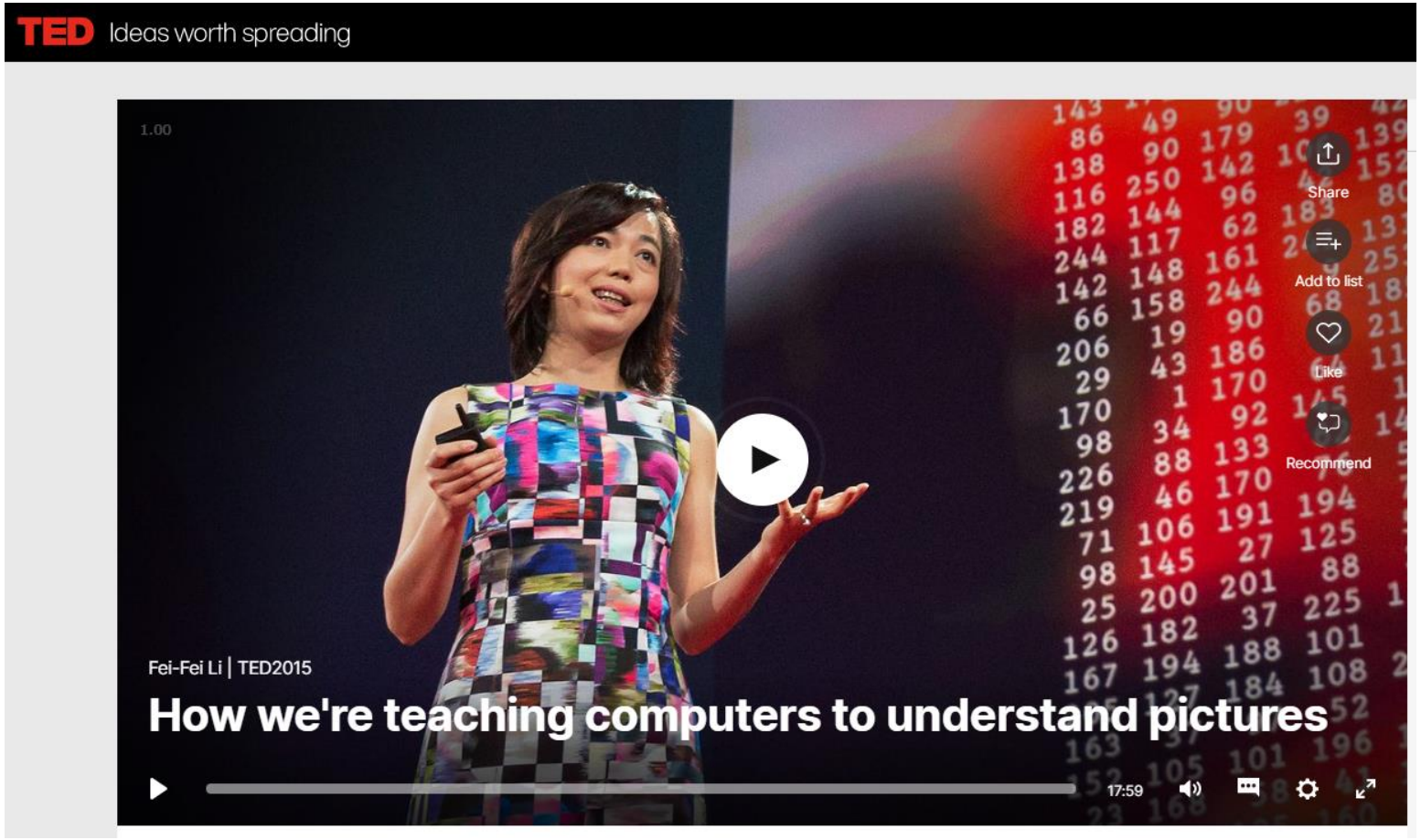
This collection is a compilation of Excellent ML and DL Tutorials created by the people below

- Andrej Karpathy blog
- Brandon Roher
- Andrew Trask
- Jay Alammar
- Sebastian Ruder
- Distill
- StatQuest with Josh Starmer
- sentdex
- Lex Fridman
- 3Blue1Brown
- Alexander Amini
- The Coding Train

Computer Vision Learning Path

- Learn Computer Vision
  - https://www.youtube.com/watch?v=FSe_02FpJas
  - https://github.com/llSourcell/Learn_Computer_Vision
- Computer Vision, PyImageSearch
  - https://www.pyimagesearch.com/start-here/
  - https://www.pyimagesearch.com/pyimagesearch-gurus/
- https://medium.com/readers-writers-digest/beginners-guide-to-computer-vision-23606224b720

https://www.ted.com/talks/fei_fei_li_how_we_re_teaching_computers_to_understand_pictures



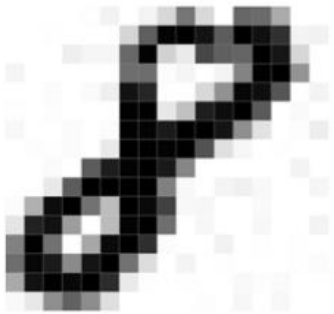
Shall I start Tomorrow?

Author - Navaneeth Malingan, Nivu Academy & Nunnari Labs


Code
hub
Newbie
Programmers
AI & ML
ALGORITHMS
DATA STRUCTURES
OOP
HELLO WORLD

Math for ML

Author - Navaneeth Malingan, Nivu Academy & Nunnari Labs



=

```
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 12, 0, 11, 39, 137, 37, 0, 152, 147, 84, 0, 0, 0, 0,
0, 1, 0, 0, 0, 41, 160, 250, 255, 235, 162, 255, 238, 206, 11, 13, 0, 0, 0, 0, 16, 9, 9, 150, 251, 45, 21, 184, 159, 154, 2
55, 233, 40, 0, 0, 10, 0, 0, 0, 0, 0, 145, 146, 3, 10, 0, 11, 124, 253, 255, 107, 0, 0, 0, 0, 3, 0, 4, 15, 236, 216, 0, 0,
38, 109, 247, 240, 169, 0, 11, 0, 1, 0, 2, 0, 0, 0, 253, 253, 23, 62, 224, 241, 255, 164, 0, 5, 0, 0, 6, 0, 0, 4, 0, 3, 252
, 250, 228, 255, 255, 234, 112, 28, 0, 2, 17, 0, 0, 2, 1, 4, 0, 21, 255, 253, 251, 255, 172, 31, 8, 0, 1, 0, 0, 0, 0, 0, 4,
 0, 163, 225, 251, 255, 229, 120, 0, 0, 0, 0, 0, 11, 0, 0, 0, 0, 21, 162, 255, 255, 254, 255, 126, 6, 0, 10, 14, 6, 0, 0, 9
, 0, 3, 79, 242, 255, 141, 66, 255, 245, 189, 7, 8, 0, 0, 5, 0, 0, 0, 0, 26, 221, 237, 98, 0, 67, 251, 255, 144, 0, 8, 0, 0
, 7, 0, 0, 11, 0, 125, 255, 141, 0, 87, 244, 255, 208, 3, 0, 0, 13, 0, 1, 0, 1, 0, 0, 145, 248, 228, 116, 235, 255, 141, 34
, 0, 11, 0, 1, 0, 0, 0, 1, 3, 0, 85, 237, 253, 246, 255, 210, 21, 1, 0, 1, 0, 0, 6, 2, 4, 0, 0, 0, 6, 23, 112, 157, 114, 32
, 0, 0, 0, 0, 2, 0, 8, 0, 7, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
```

Matrix Multiplication

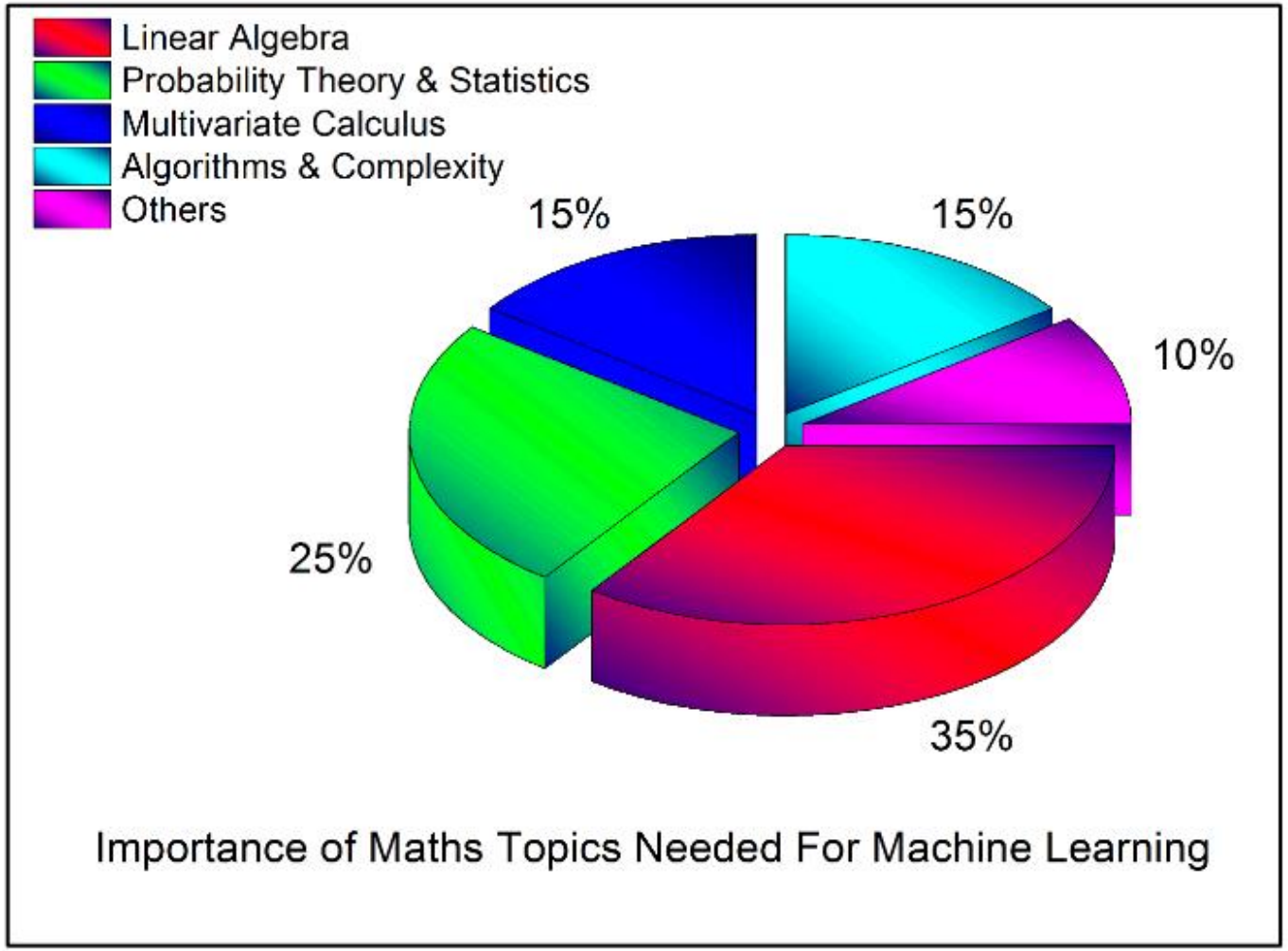
Neural Network

Author - Navaneeth Malingan, Nivu Academy & Nunnari Labs

Machine Learning theory is a field that intersects statistical, probabilistic, computer science and algorithmic aspects arising from learning iteratively from data and finding hidden insights which can be used to build intelligent applications.

Why need Math in ML?

- Why something works?
- Why one model is better than other?
- Selecting the right algorithm which includes giving considerations to accuracy, training time, model complexity, number of parameters and number of features.
- Choosing parameter settings and validation strategies.
- Identifying underfitting and overfitting by understanding the Bias-Variance tradeoff.
- Estimating the right confidence interval and uncertainty.


Linear Algebra
Probability Theory & Statistics
Multivariate Calculus
Algorithms & Complexity
Others
15%
15%
10%
25%
35%
Importance of Maths Topics Needed For Machine Learning

Math for ML Resources

https://nivu.me/posts/mathematics-for-machine-learning/

Reach out to me

Navaneeth Malingan

Mobile : 9047578585

Email: mail@nivu.me

Blog: https://nivu.me

LinkedIn: https://www.linkedin.com/in/nivu/

Twitter: https://twitter.com/nivu07

NUNNARI labs

NIVU

Author - Navaneeth Malingan, Nivu Academy & Nunnari Labs

Thank You

Author - Navaneeth Malingan, Nivu Academy & Nunnari Labs