University of Nottingham
UK | CHINA | MALAYSIA

COMP2005

Introduction to Image Processing

Lecture 11

Convolutional Neural Networks - CNN

University of Nottingham
UK | CHINA | MALAYSIA

COMP2005: Introduction to Image Processing
Week 33 – 10:00am Friday – 10 May 2024

valid for 65 minutes from 9:55am
generated 2024-01-20 03:09

Learning Outcomes

IDENTIFY

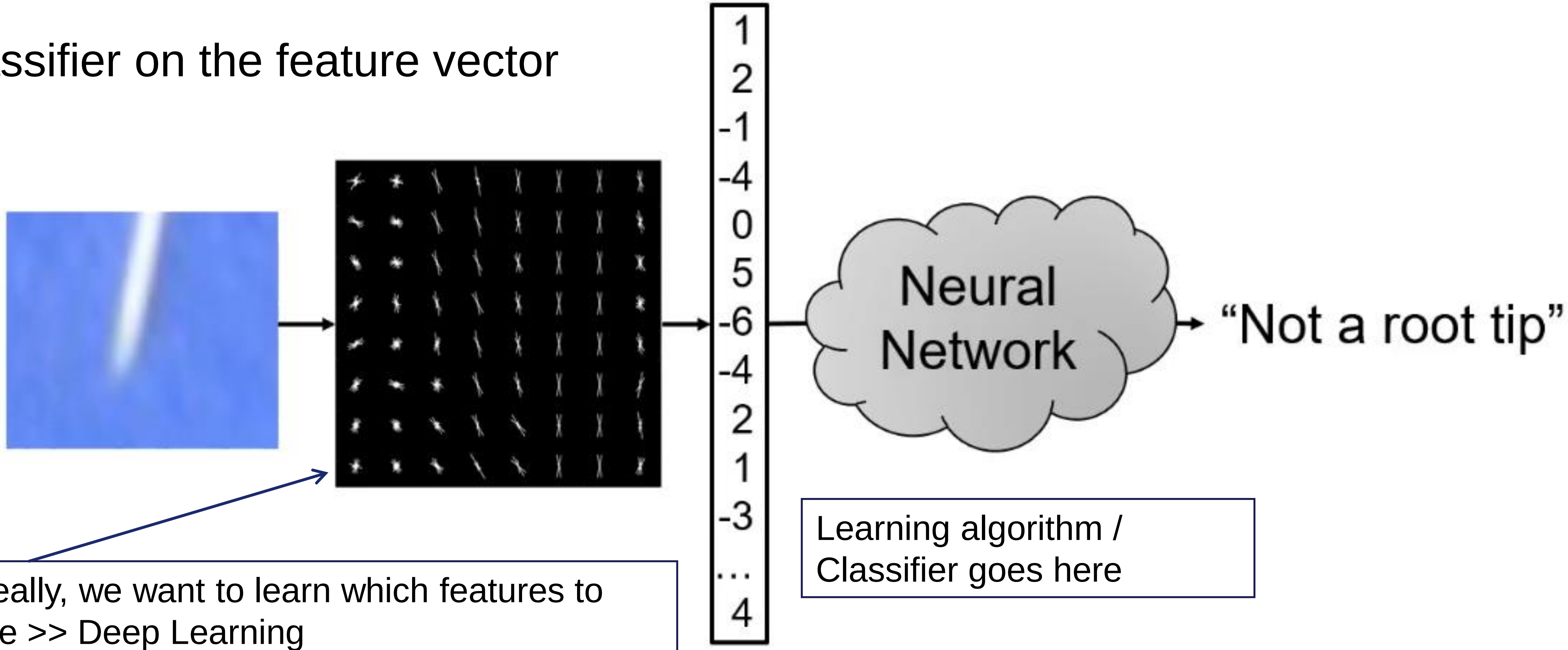
1. Background
2. Neural Networks
3. Convolutional Neural Network
   Convolutional layers
   Pooling Layers
   Activation Functions
   Final stage : Softmax
4. Applications
5. Future

University of Nottingham
UK | CHINA | MALAYSIA

Background

Traditional Pipeline for Machine Learning

A classifier on the feature vector



Learning algorithm / Classifier goes here

Ideally, we want to learn which features to use >> Deep Learning

What is Deep Learning

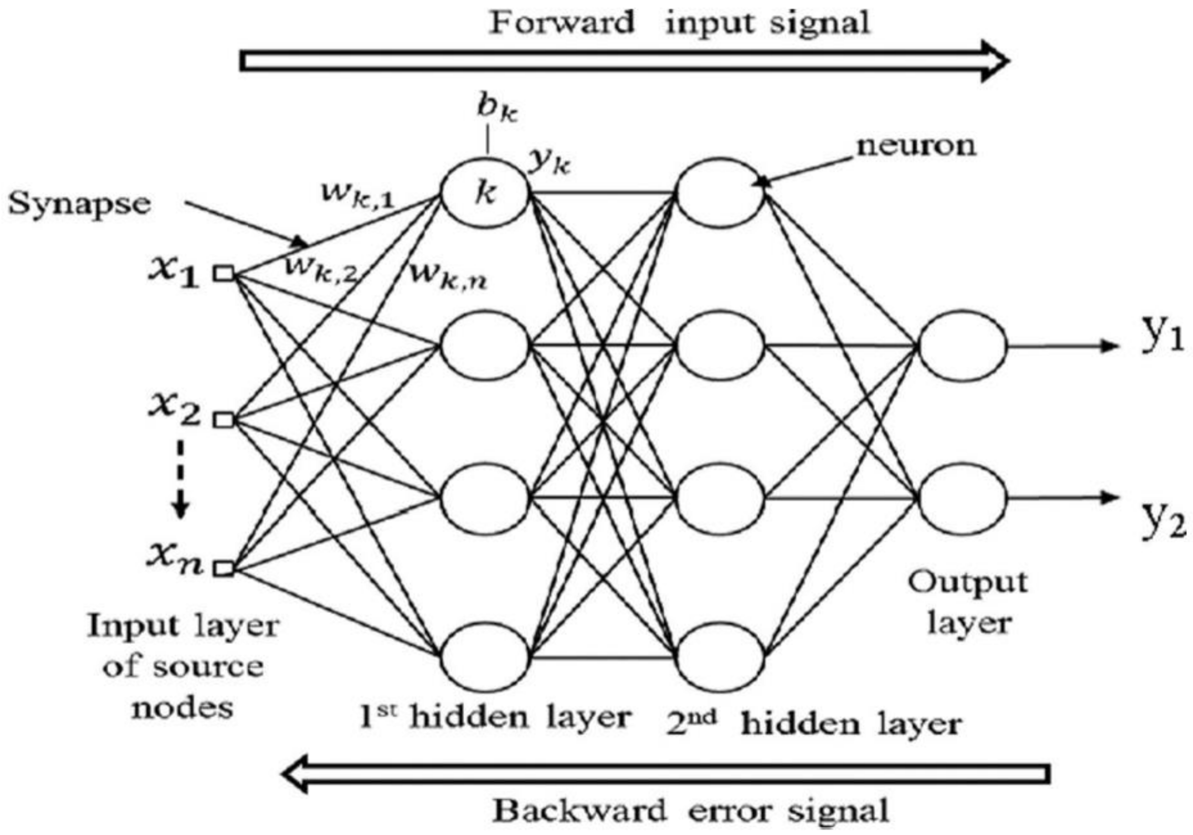
- Deep Learning is a popular AI technique
- Essentially a kind of Neural Network
- Deep refers to ability of having Many Layers in the network
  - Which was not possible in traditional Neural Networks

- What led to the development of deep learning ?
  - Several factors including :

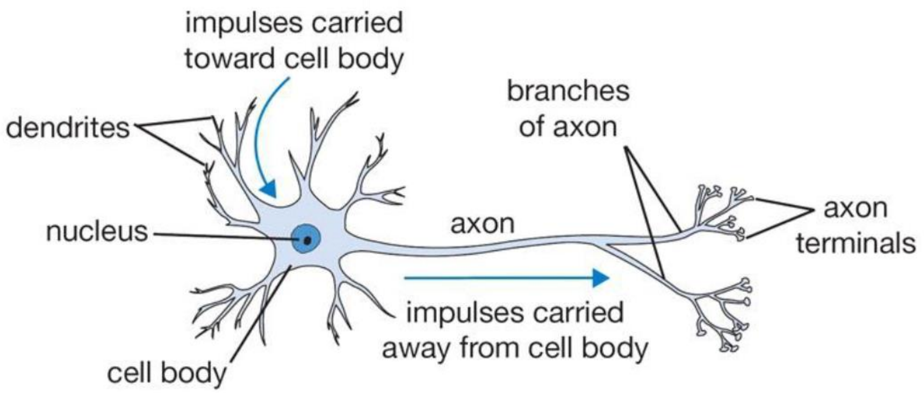
  - GPU development
  - Algorithm improvement
  - Availability of large training image sets
  - …..

- Looking into CNN as the main technique for deep learning

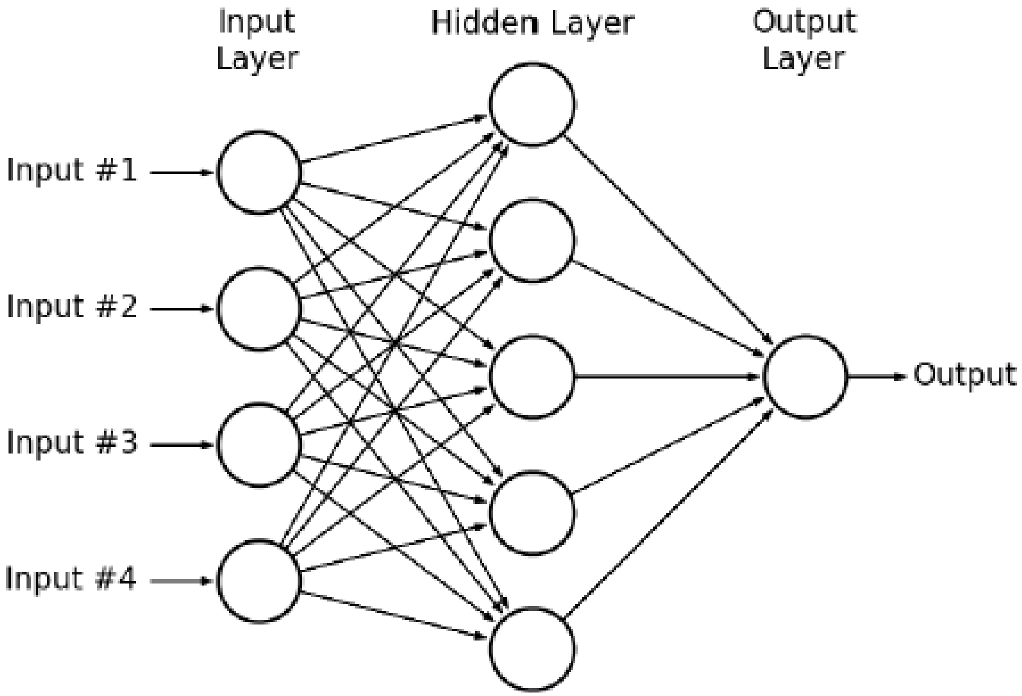
Classical Neural Network



Human Neutrons

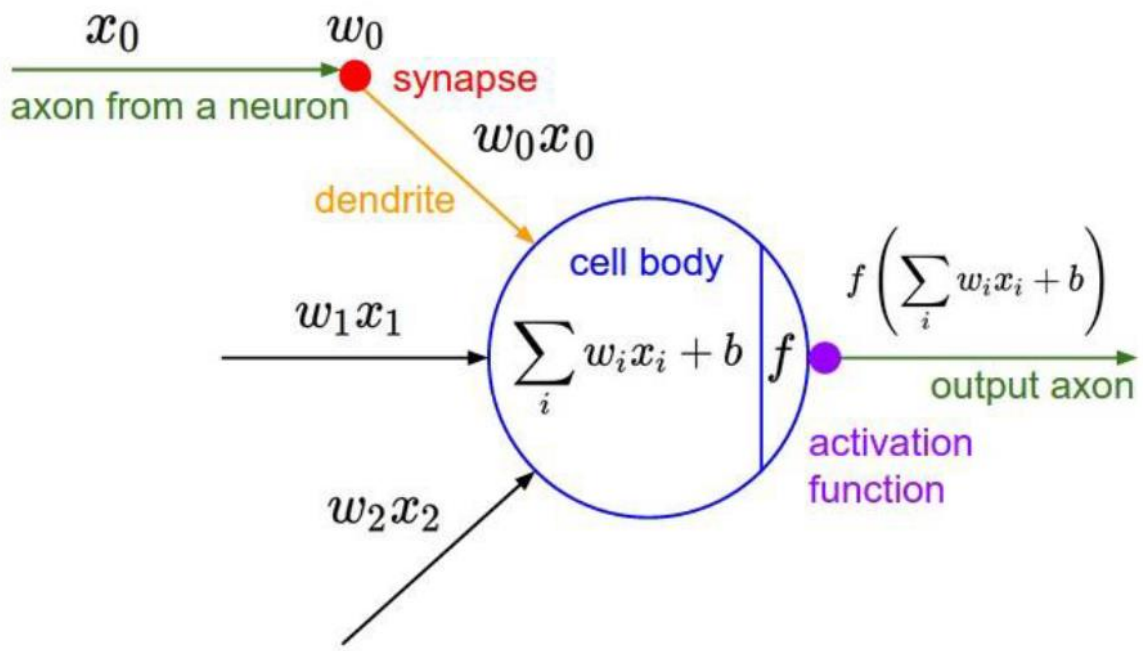


Humans have ~100-1,000 trillion connections in their brains


Input Layer
Hidden Layer
Output Layer
Input #1
Input #2
Input #3
Input #4
Output

Classical Neural Network

Modelling Neurons



Modern artificial networks tend to use 100k – 10b connections

Inspiration…

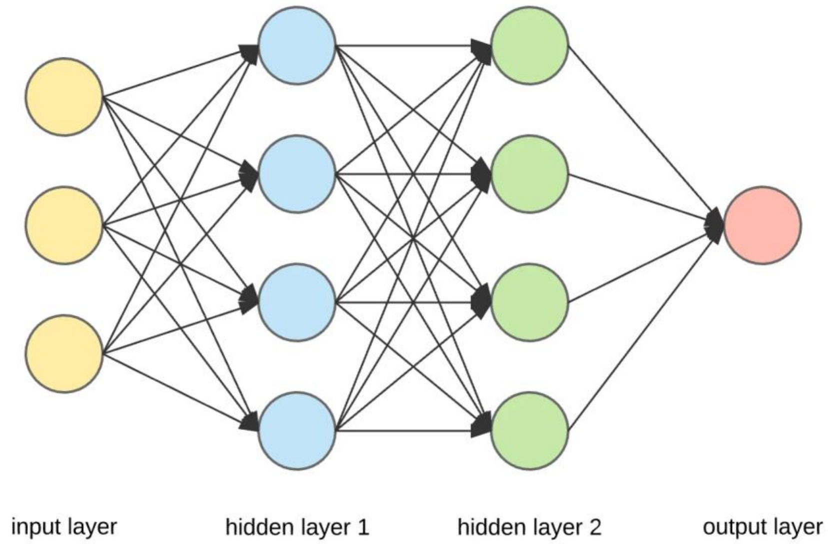
CNN

Convolutional Neural Network

Inspired by

DID YOU KNOW: Each of the convolutional layers perform the image processing techniques that we learned throughout this module. Techniques such as *convolution/filtering, re-sizing, noise removal* and *edge detection* to name a few.

Interesting FACT

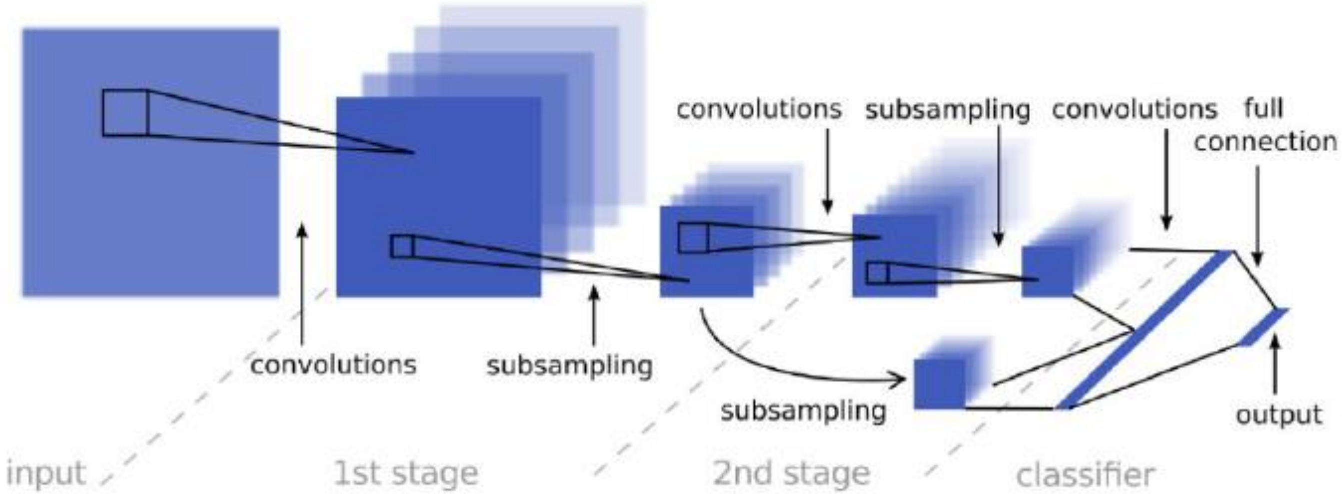


Artificial Neural Network[1]

[1] Dertat, A. (8 August 2017). *Applied Deep Learning – Part 1: Artificial Neural Network*. Towards Data Science. https://towardsdatascience.com/applied-deep-learning-part-1-artificial-neural-networks-d7834f67a4f6

Convolutional Neural Network

- Make the assumption the input is an *image*
- Neural networks that use convolution in place of general matrix multiplication in at least one of their layers.
- An end-to-end learned solution to many vision tasks
- Local analysis matches the natural structure of the most images
- Learn hierarchical models of image content



Multilayer Perceptron Network - MLP

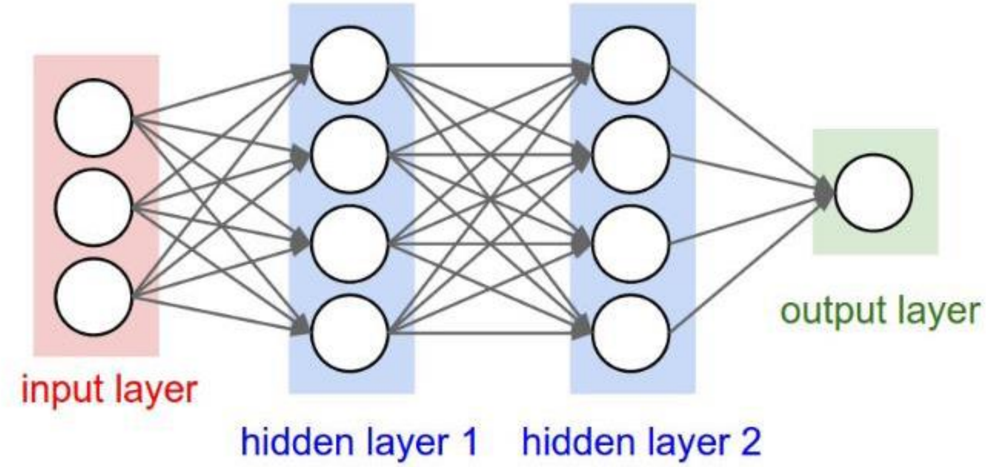
Wide application scenario–not just images

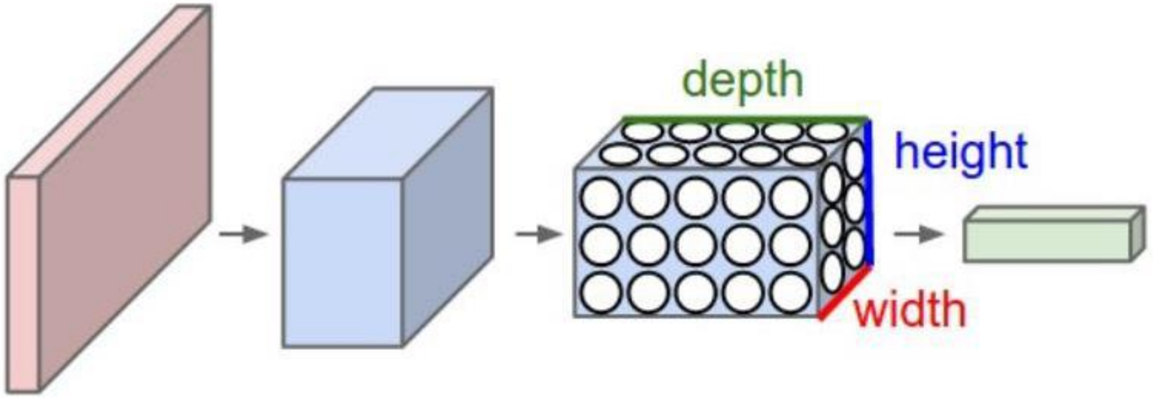
Neurons are **fully connected**–can't scale well to large size data (e.g.images)

Convolutional Neural Network - CNN

Neurons are arranged in'3D', each neuron is only connected to a small region of previous layer

Typical CNN structure: **Input-conv-activation-pool- fully connected-** output

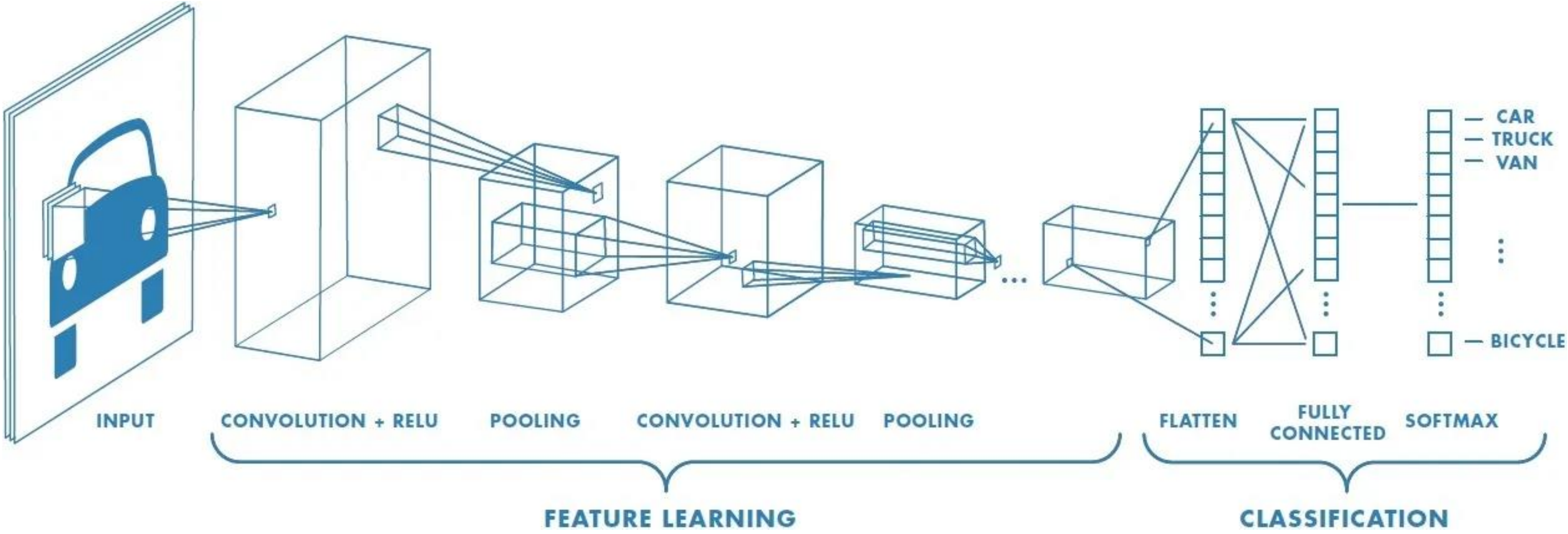




Multilayer Perceptron (MLP) Network
Convolutional Neural Network

Convolutional Neural Network

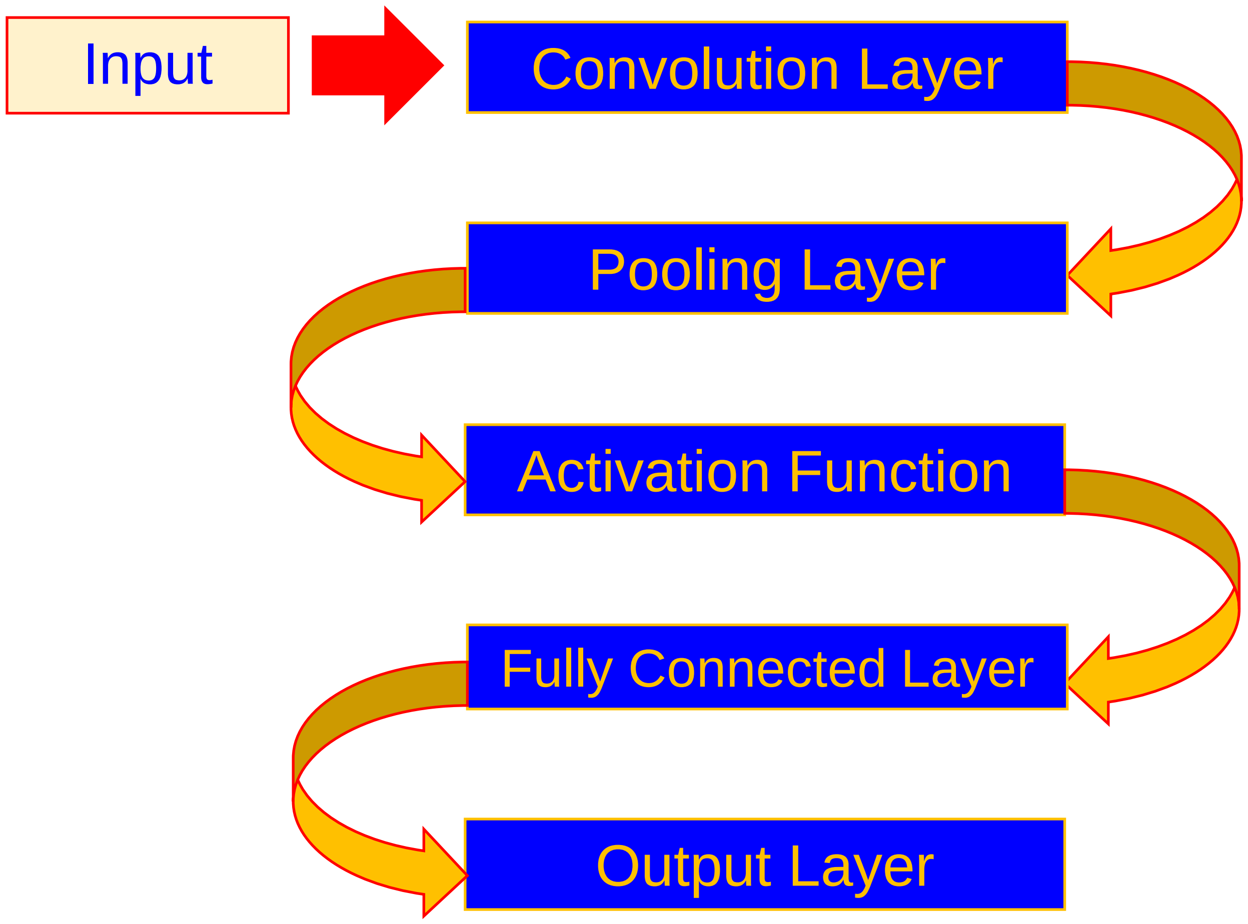
Convolutional Neural Network Architecture



Extracted from: Raghav, P. (4 May 2018). *Understanding of Convolutional Neural Network (CNN) – Deep Learning*. Medium. https://medium.com/@RaghavPrabhu/understanding-of-convolutional-neural-network-cnn-deep-learning-99760835f148

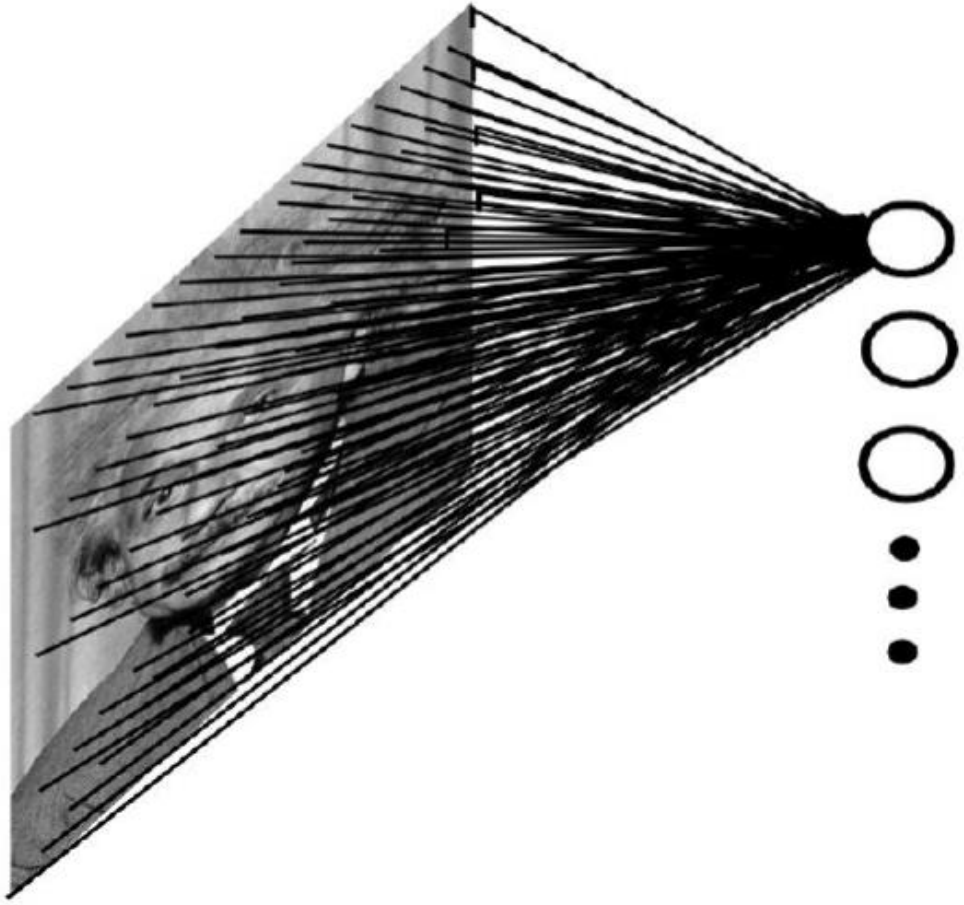
Components of CNN

CNN Components



Locally Connected Layers – Traditional NN

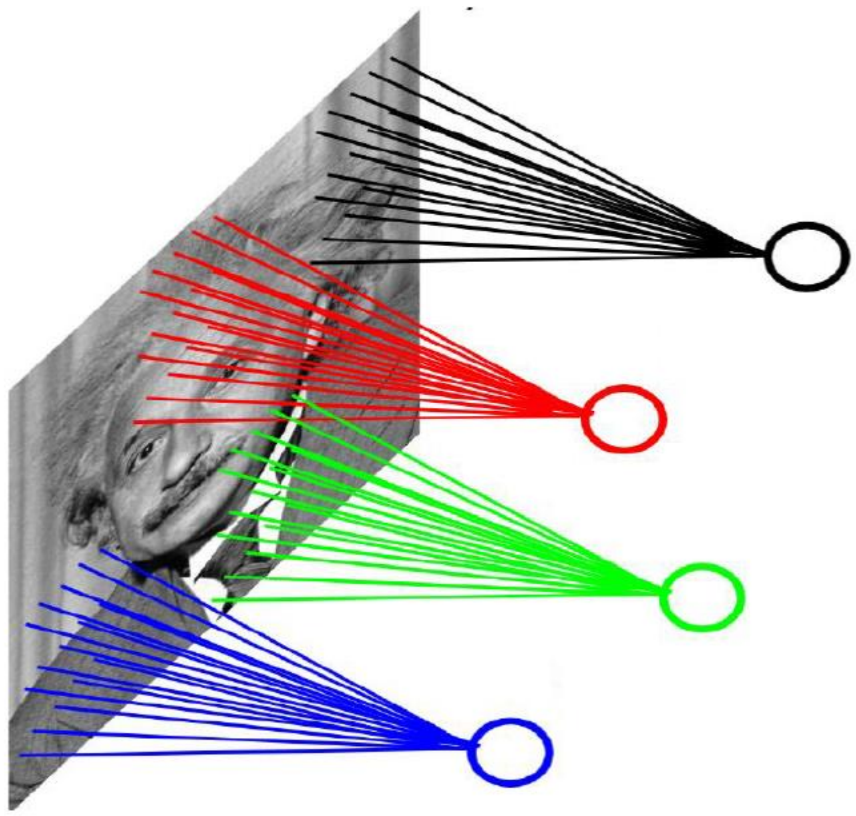
Classical NN with fully connected hidden layers



- 200x200 image, 40K hidden units (1 per pixel) means 1.6B weights to learn
- Waste of resources
  - Spatial correlation is local
  - most of the weights would be 0
- Would require an impractically large training set to learn this many weights

Locally Connected Layers

Classical NN with fully connected hidden layers



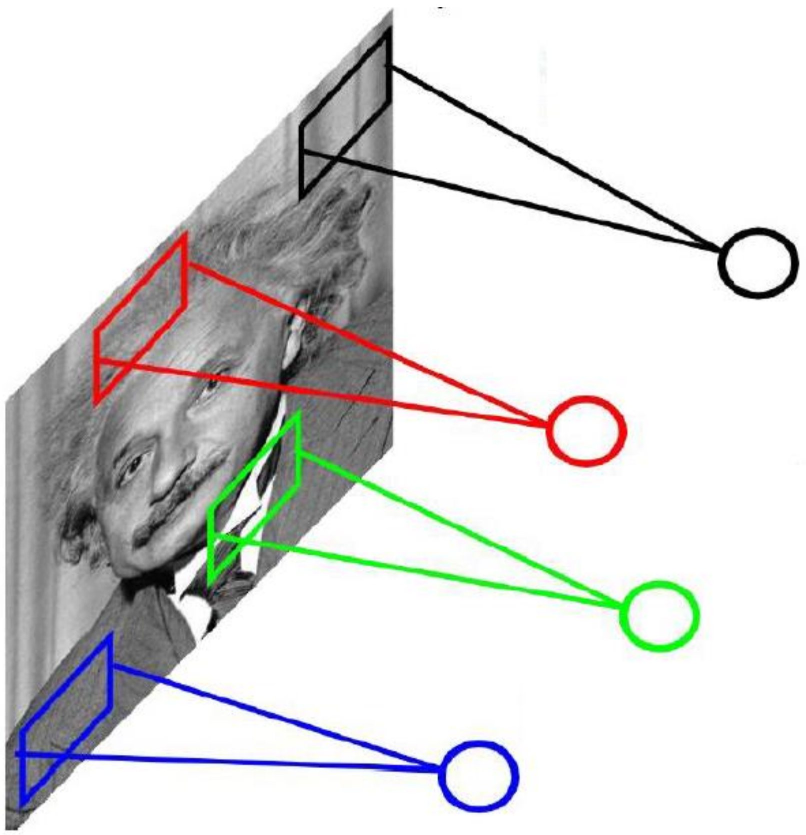
- CNNs' early layers are ***locally* connected**

e.g. 200x200 image, 40K hidden units, fed by 10 x 10 filters, means 4M weights to learn

= much fewer than 1.6B

Convolution Layers

Classical NN with fully connected hidden layers



- In image processing/vision we usually want to apply the ***same* convolution mask at each location**

- Each neuron has the same weights

e.g. 200x200 image, 40K hidden units, 10 x 10 mask means only 100 weights to learn

Convolutional Layer

IMPORTANT:

Parameters for **training process**

# of Filters

Defines the **depth** of the output feature map(s)

Stride

# of pixels the filtering window moves after each operation

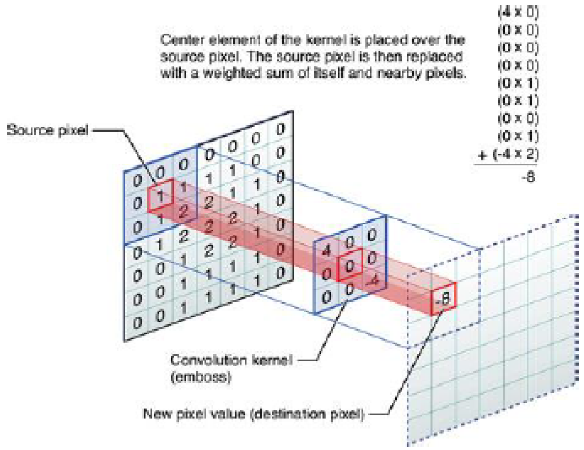
Zero-padding

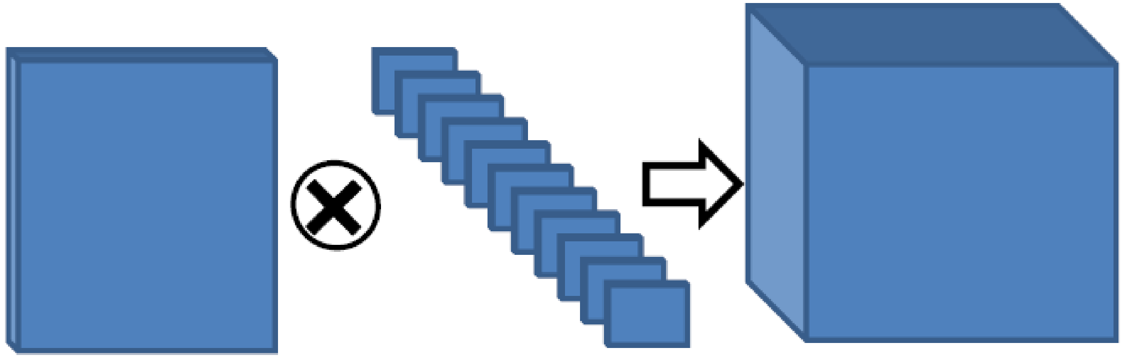
Adds zeroes to every side of the input boundaries – ensures filters fit the input image

Convolution

- We wish to *convolve* the input image with a set of ***learnable, small-size filters***
- size(W); filter size(F); zero padding(P); stride(S)
  - F, conv filter size,is like *are captive field*
- Output volume size calculation **(W-F+2P)/S+1**

Input | Learnable filters | Output





100 x 100 x3

10, 5x5x3
(75+1)*10 parameters

100x100x10
(100-5+2*2)/1+1=100

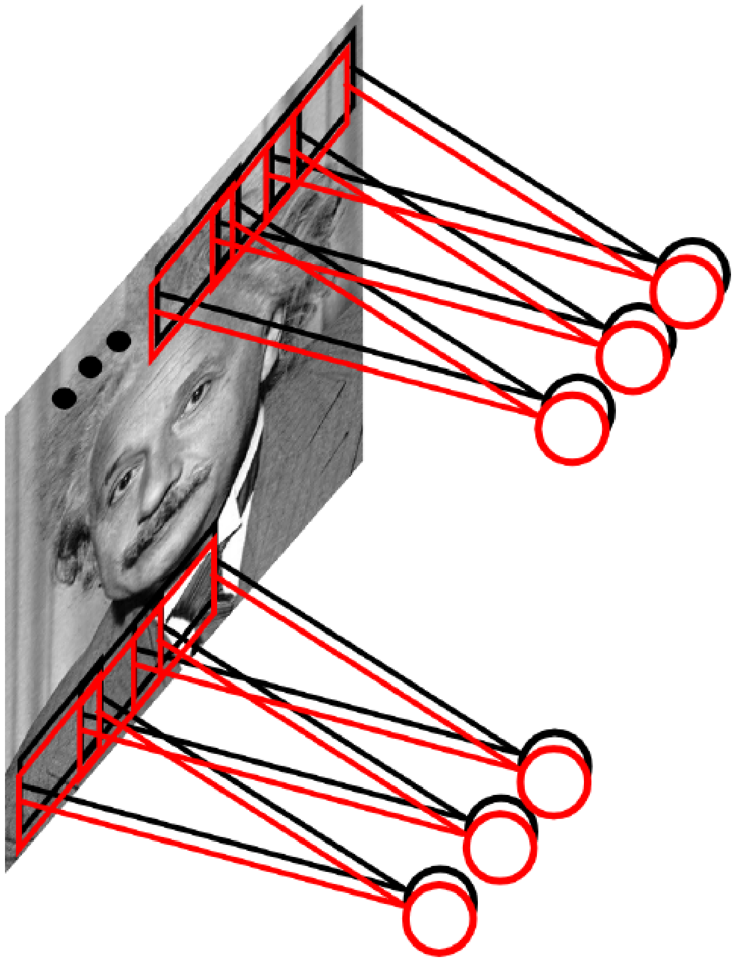
Does this look familiar?

Convolution Layer

- We can afford to learn multiple filters
  e.g. 100 10x10 masks is only 10K parameters

- *Convolutional layers*are filter banks performing convolutions
  with the learned kernels (masks)

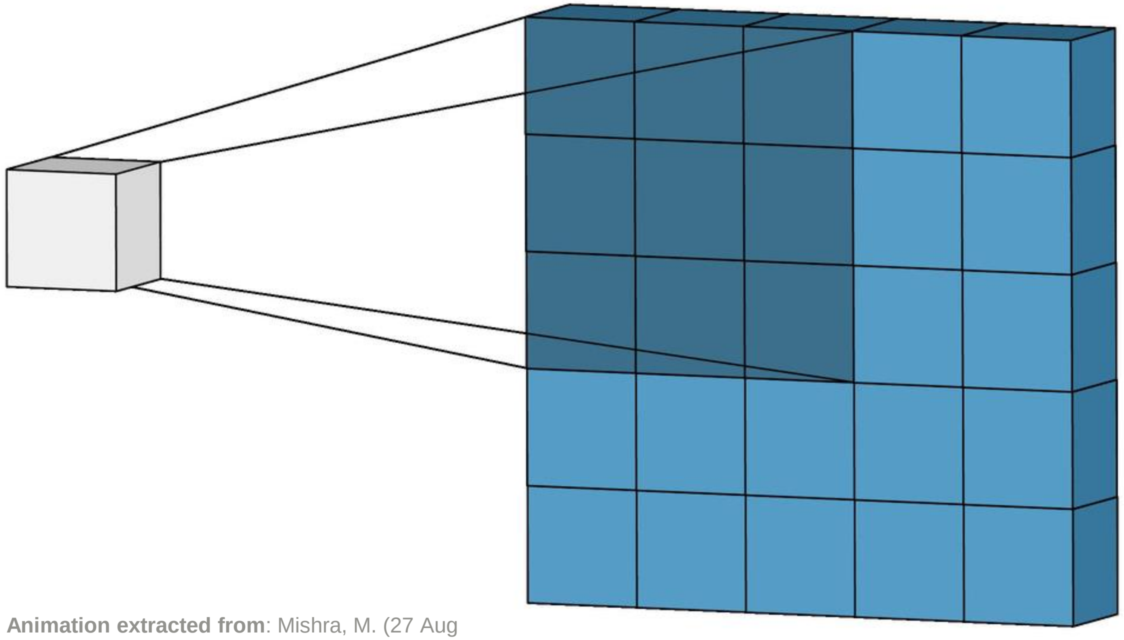
- Could be applied to all pixels, or have a small ‘stride’
  to spread them out



Convolutional Layer

- Most important layer
- Performs major computations
- Takes in input image, performs filtering which produces feature map(s)
- Filters using image processing techniques (e.g., **edge detection, blur** and **sharpen**)
- Filtering is performed using 3x3 kernels to perform the dot product

Resulting array [with weights] (*shown in grey in the animation below*) is known as feature map or activation map

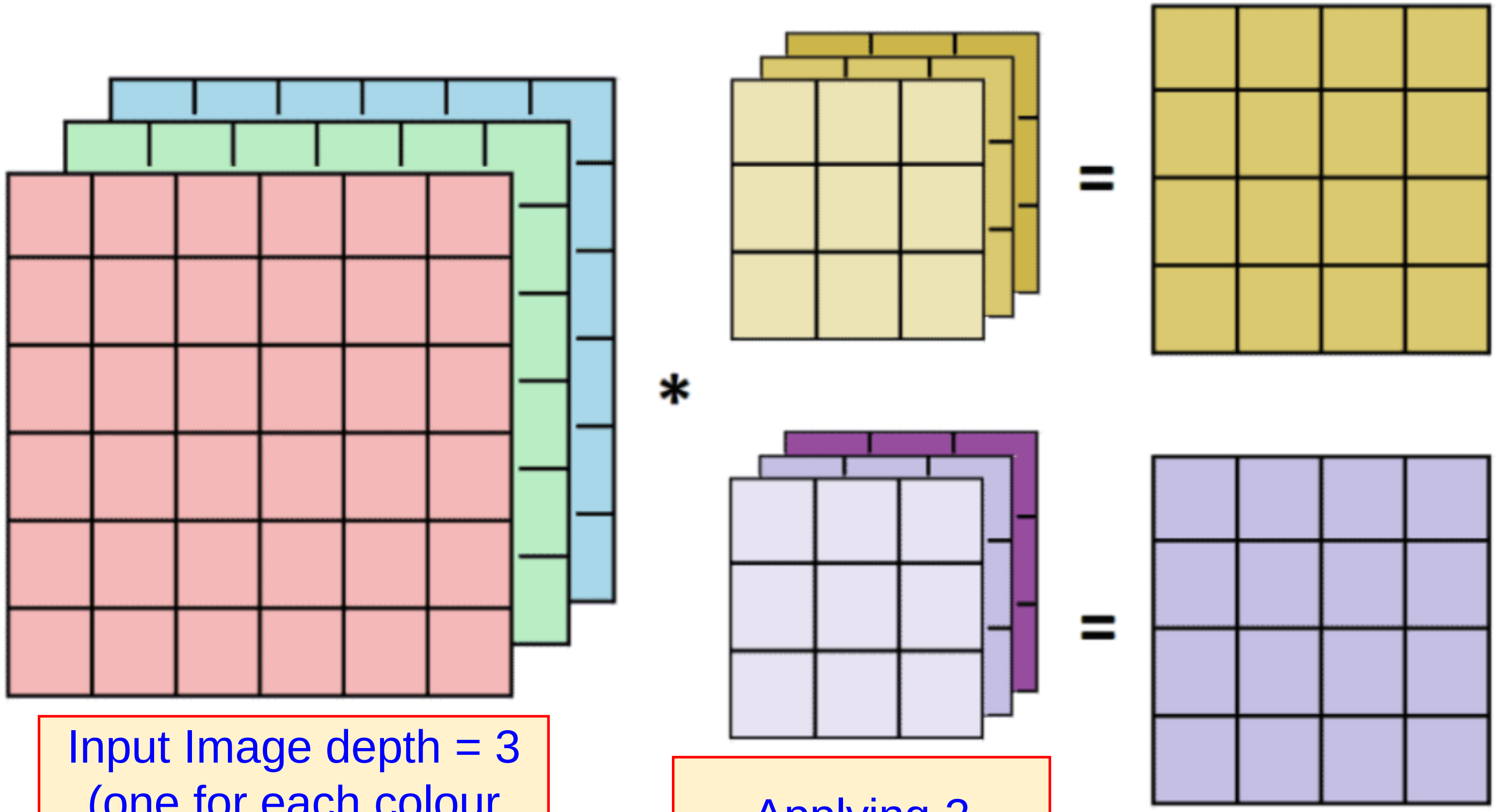


Animation extracted from: Mishra, M. (27 Aug 2020). *Convolutional Neural Networks, Explained*. Medium. https://towardsdatascience.com/convolutional-neural-networks-explained-9cc5188c4939

Filter weights are only **adjusted** via backpropagation & gradient decent during the training process

Note

Convolutional Layer



Input Image depth = 3 (one for each colour channel)

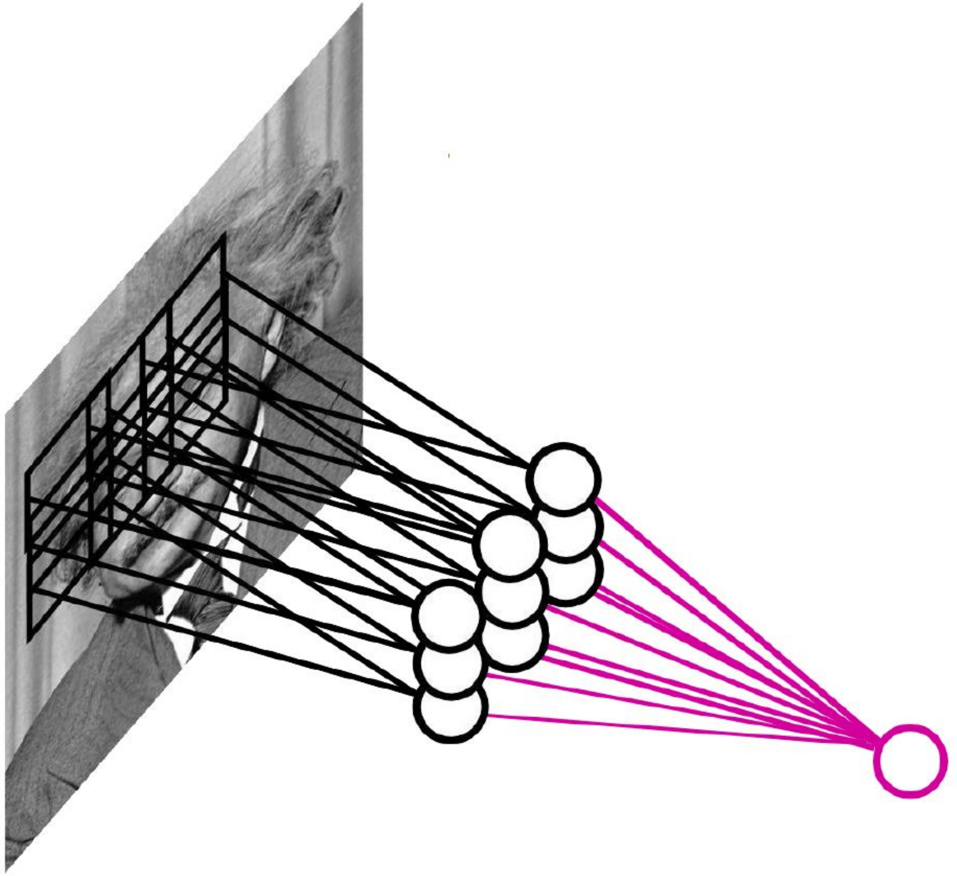
Applying 2 types of filters

Resulting Feature Maps

Illustration extracted from: Zvornicanin, E. (18 March 2024). *What is Depth in Covolutional Neural Network?* https://www.baeldung.com/cs/cnn-depth

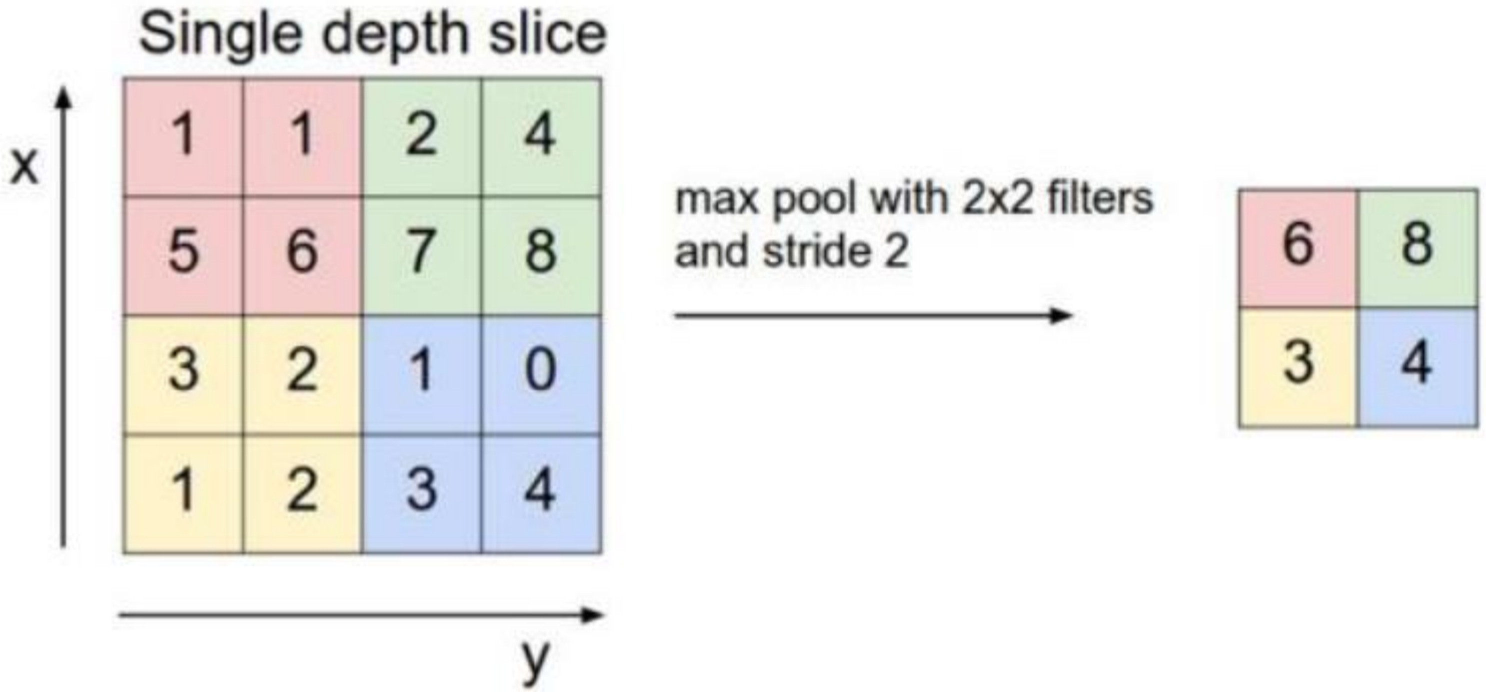
Pooling Layer

- Suppose one of our convolutions is an ***eye detector*** –how can we make the net robust to the exact location of the eye?
- By pooling (e.g. taking the max) filter responses at different locations
- Don’t pool different features



- A number of pooling methods exist, including subsampling
- Effect is to reduce the resolution of the filter outputs
- *Subsequent convolutional layers therefore access larger areas of the image*

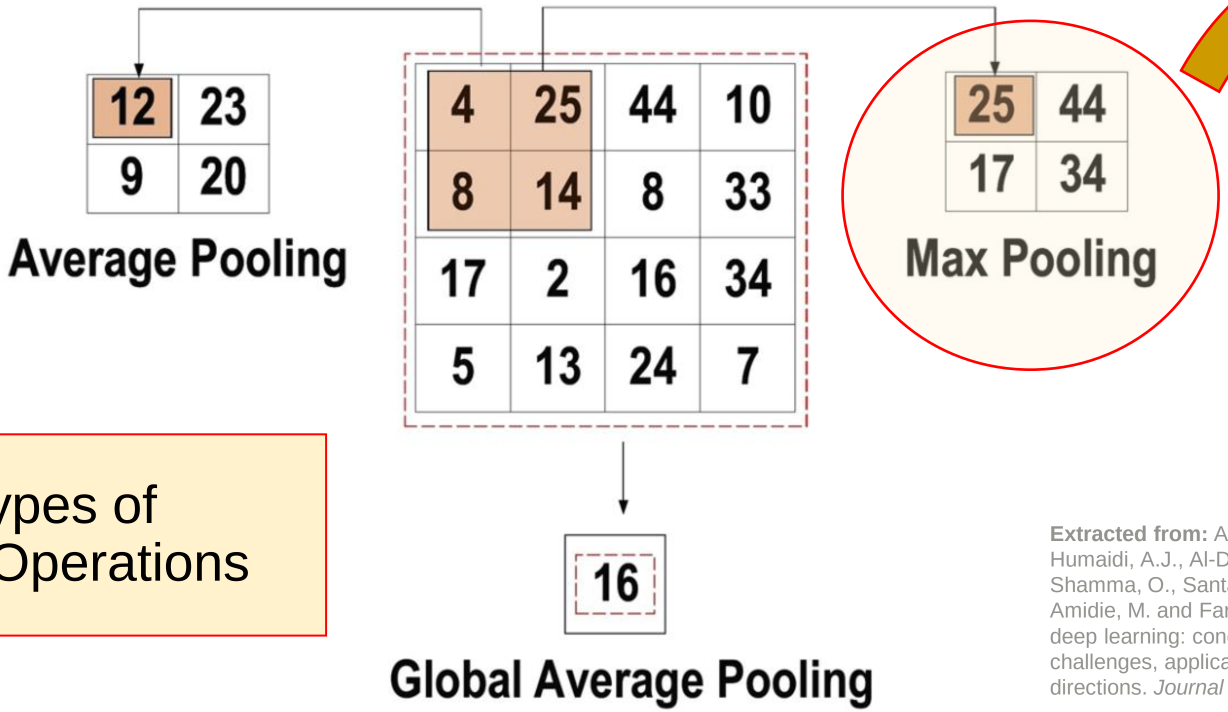
Name	Pooling formula
Average pool	$\frac{1}{s^2}\sum x_i$
Max pool	$\max\{x_i\}$
L2 pool	$\sqrt{\frac{1}{s^2}\sum x_i^2}$
L_p pool	$\left(\frac{1}{s^2}\sum \lvert x_i\rvert^p\right)^{\frac{1}{p}}$



Pooling Layer

?

Occurs after a convolutional layer. Reduces the dimensionality of the resulting conv layer



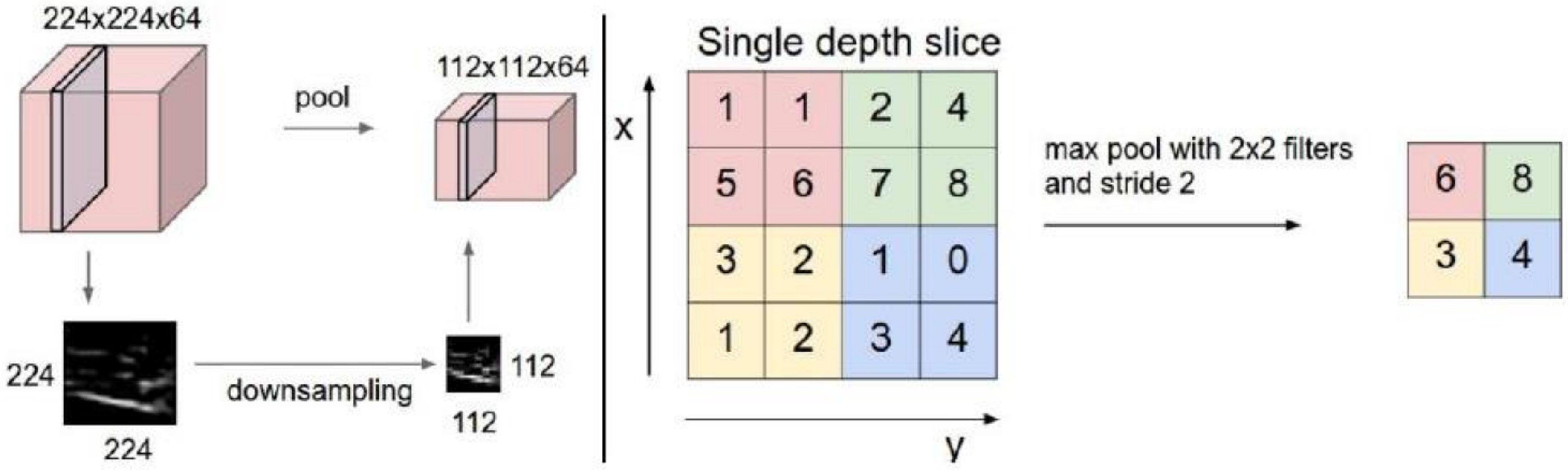
Famously Used

Three Types of Pooling Operations

Extracted from: Alzubaidi, L., Zhang, J., Humaidi, A.J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M.A., Al-Amidie, M. and Farhan, L. (2021). Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. *Journal of big Data*, *8*, pp.1-74

Effect of Pooling

- Reduce the spatial size of the representation and reduce the amount of parameters
- Effectively down-sampling the input to increase the receptive field size
- Max operation with stride of 2 is a popular choice



Activation Function

?

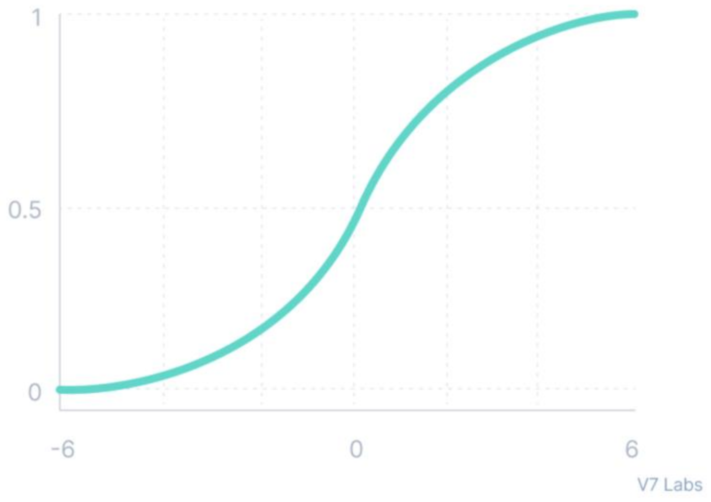
A.k.a. *transfer function* – determines to neural network's output

Choose

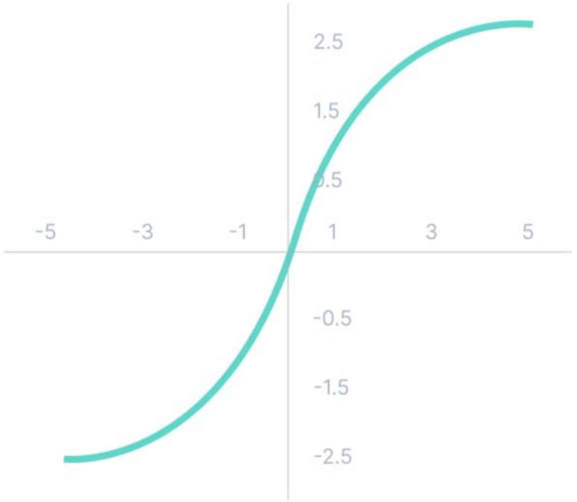
- Sigmoid
- Tanh
- ELU *(Exponential Linear Unit)*
- Leaky ReLU
- ReLU *(Rectified Linear Unit)*

Famously Used

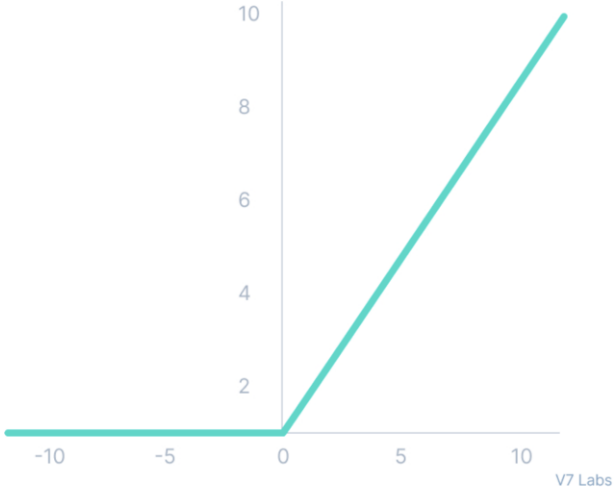
Sigmoid



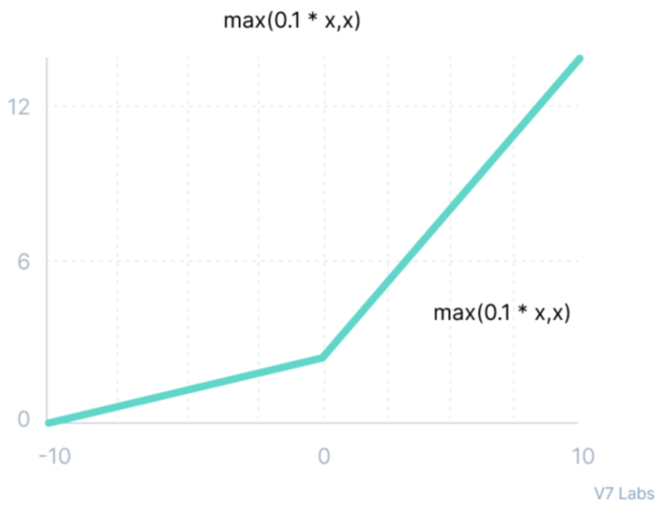
Tanh



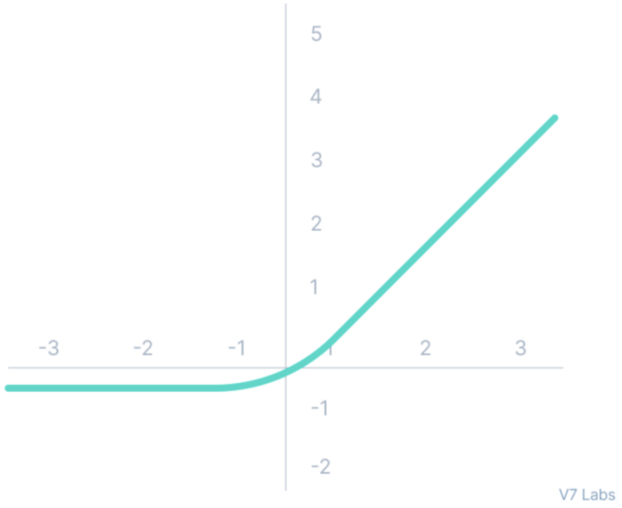
ReLU



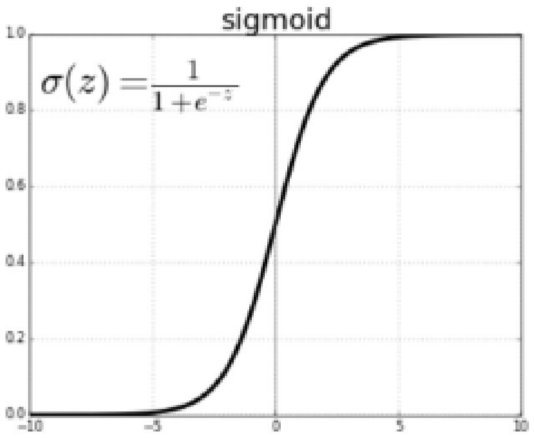
Leaky ReLU

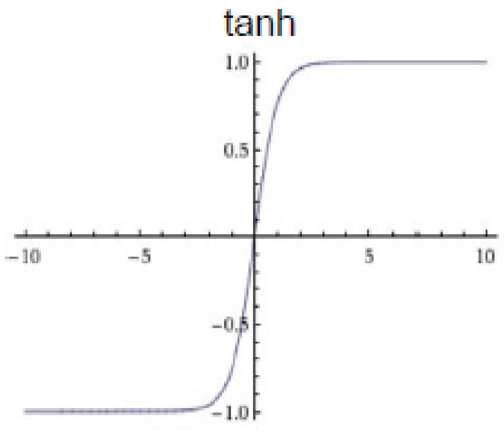


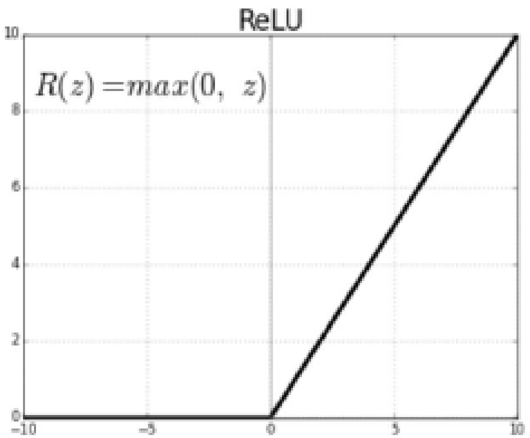
ELU

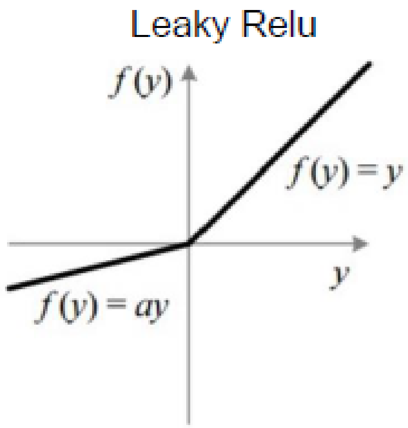


Graphs extracted from: Bagheti, P. (27 May 2021). *Activation Functions in Neural Networks [12 Types & Use Cases]*. V7. https://www.v7labs.com/blog/neural-networks-activation-functions









- ✓ Transfer the range to [0, 1]
- x Saturate and kill gradients: small gradient at region of 0 and 1

- ✓ Zero centred range: [-1, 1]
- x Saturate and kill gradients: small gradient at region of 0 and 1

- ✓ Solves vanishing/exploding gradients
- ✓ Simple to calculate
- x Some neurons can be ‘dead’ with negative input

- ✓ Overcome the ‘dying neuron’ problem
- x Performance not consistent

Fully Connected Layer

- **FINALE** layer
- Utilises features extracted from previous layers, performs task classification

?

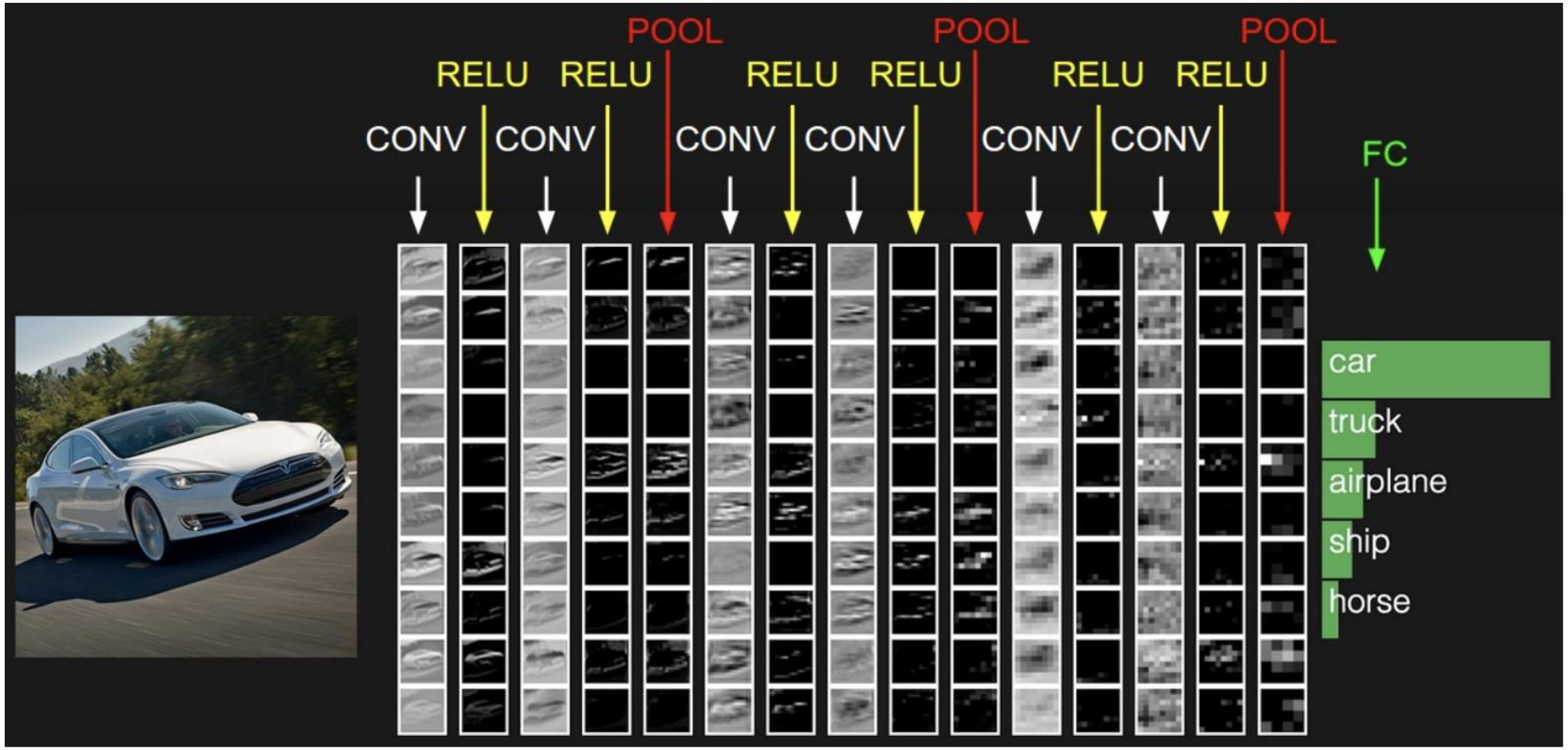
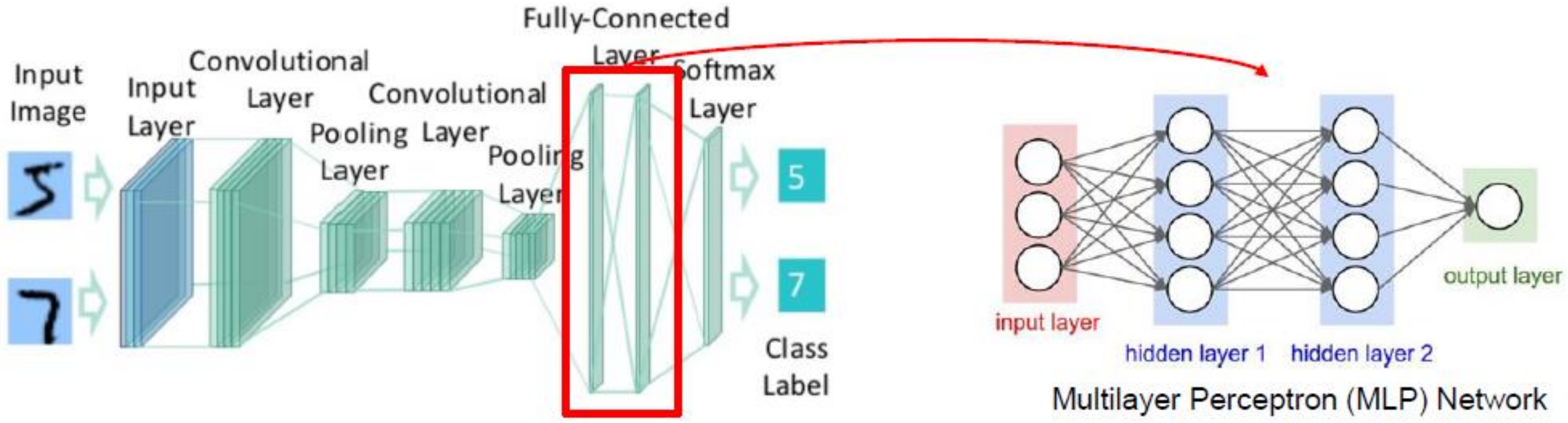


Image extracted from: Standard University. (n.d.). *CS231n Convolutional Neural Networks for Visual Recognition*. https://cs231n.github.io/convolutional-networks/#fc

Fully connected layer and softmax

- Global feature learning: fully connected to all activations in the previous layer, as the same in MLP.
- Softmax - converts the prediction to the range of [0..1] for each class

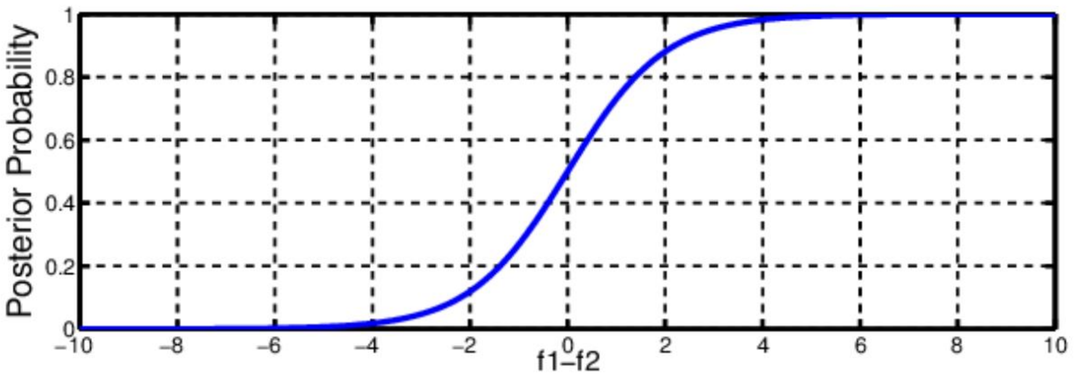


Output Layer

?

- Performs a *logistic function* to classify tasks
- Uses ***Softmax*** activation function where probability ranges from 0 to 1 – scores given to each class
- Sometimes, **embedded** within the FC layer

Softmax Activation Function



Graphs extracted from: Chen, B., Deng, W., & Du, J. (2017). Noisy Softmax: Improving the Generalization Ability of DCNN via Postponing the Early Softmax Saturation In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5372-5381)

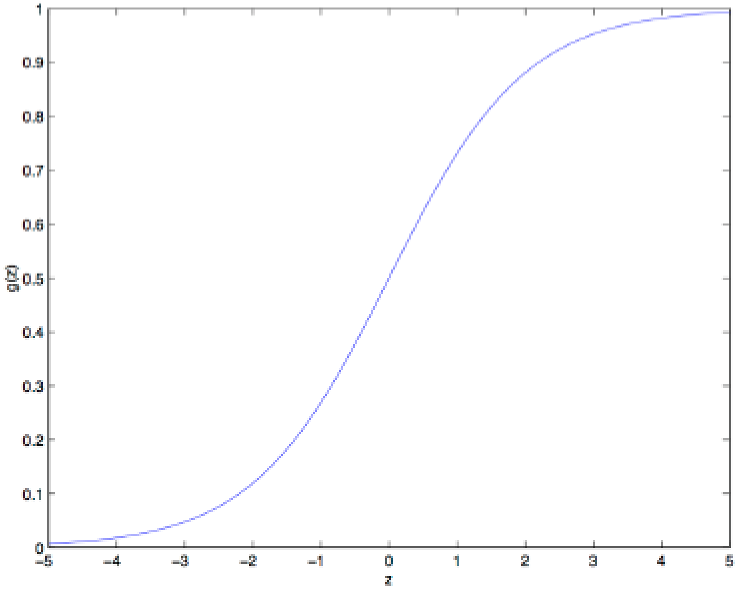
Softmax

- With *Linear regression*, we try to predict a real value *y,* for an input value *x*
    - Perhaps we are predicting someone's age from a picture

- Linear regression is not good if we are predicting if input data should be assigned class A or class B
    - Perhaps we are predicting Happy or not-Happy from a picture

- To do this there are more suitable kinds of regression
    - E.g. We can use *Logistic regression*
        - This can be used to "squash" a number into one of two class

Example: *sigmoid*function →



Softmax

So why do we need a *softmax*function?

- We need a different function if we have *multiple* (ie.>2, non-binary) classes
  Works a bit like logistic regression, but for multiple classes

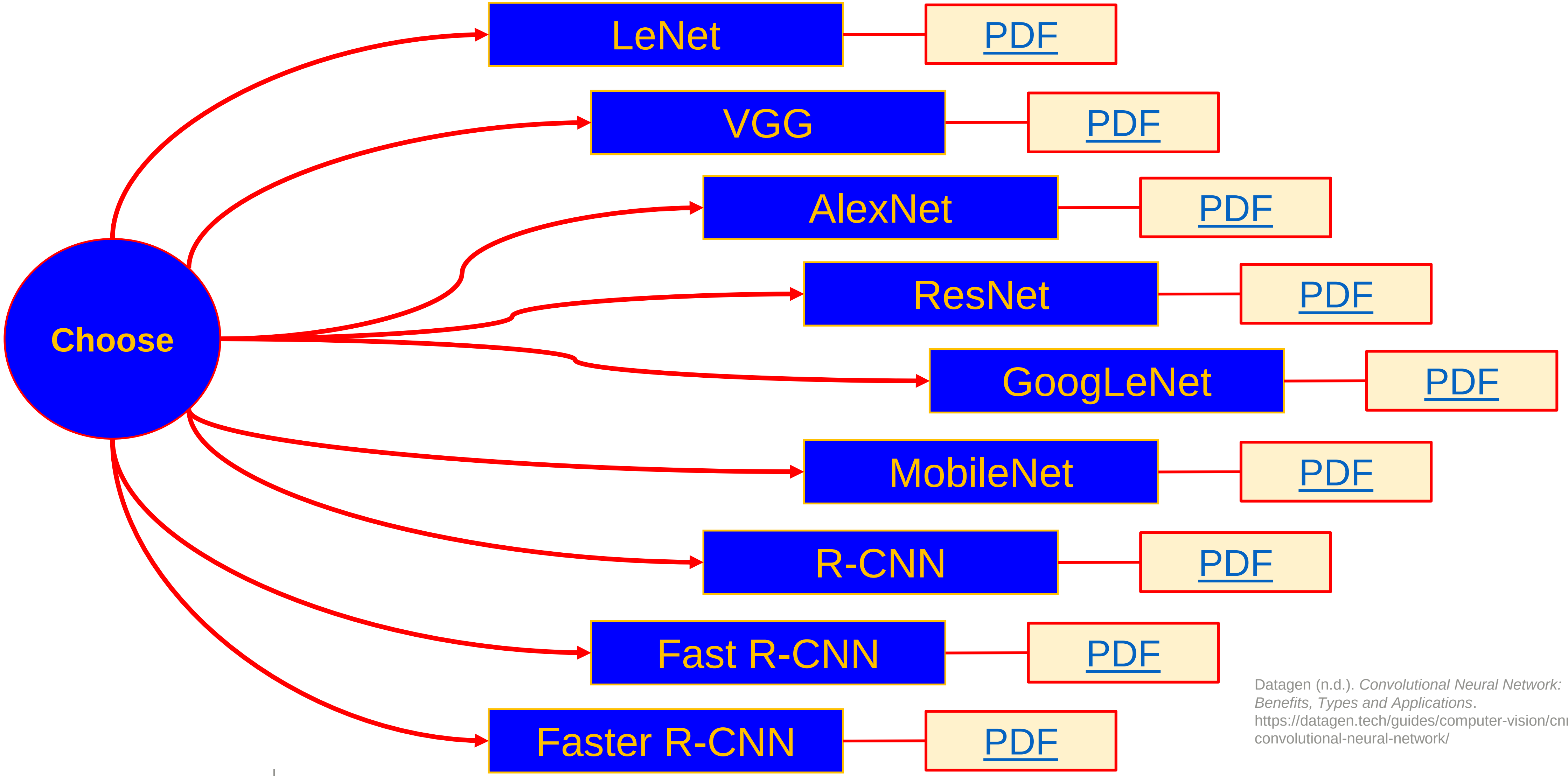
- A softmax function takes a set of numbers as an input, and outputs a probability distribution spread over a set of *k* classes
  - We want to know which class *k* is the most likely given a data point
  - e.g. predicting if a person is happy, sad, grumpy, sleepy, etc. from a picture.

- Softmax allows us to do this
  - Gives us a probability for each class for a given input in the range 0..1
  - Probabilities sum to 1. e.g:

Class	Probability
Happy	0.05
Sad	0.32
Grumpy	0.44
Sleepy	0.19

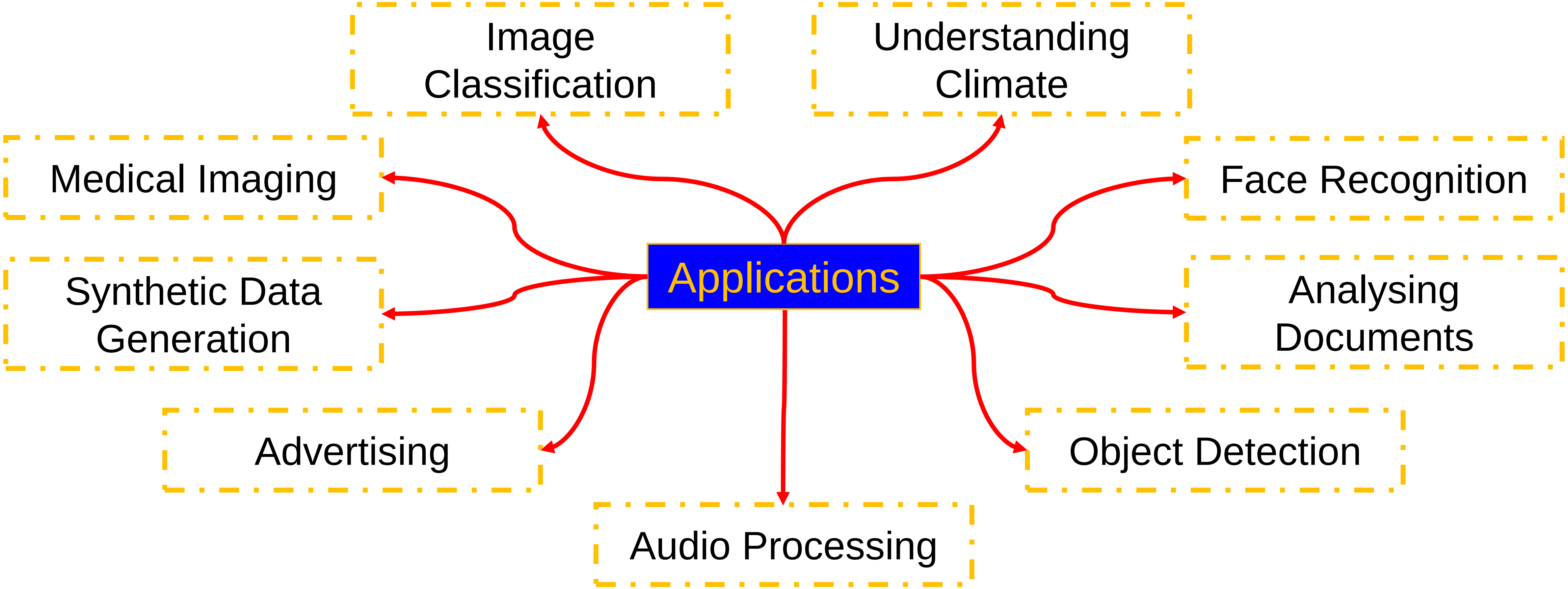
Applications

Famous CNN



Datagen (n.d.). *Convolutional Neural Network: Benefits, Types and Applications*. https://datagen.tech/guides/computer-vision/cnn-convolutional-neural-network/

Some Practical Applications of CNN



Keita, Z. (Nov 2023). *An Introduction to Convolutional Neural Network (CNNs)*. https://www.datacamp.com/tutorial/introduction-to-convolutional-neural-networks-cnns

Ray, P. (14 Jan 2021). *Convolutional Neural Network (CNN) and its application - All you need to know.* https://medium.com/analytics-vidhya/convolutional-neural-network-cnn-and-its-application-all-u-need-to-know-f29c1d51b3e5

MATLAB (n.d.). *What is a Convolutional Neural Network*. https://www.mathworks.com/discovery/convolutional-neural-network.html

University of Nottingham
UK | CHINA | MALAYSIA

Future

Vision Transformer (ViT)

- Designed for Computer Vision with remarkable results
- Uses **neural network** to split images into smaller patches, allowing model(s) to capture both local and global relationships within images.[1]

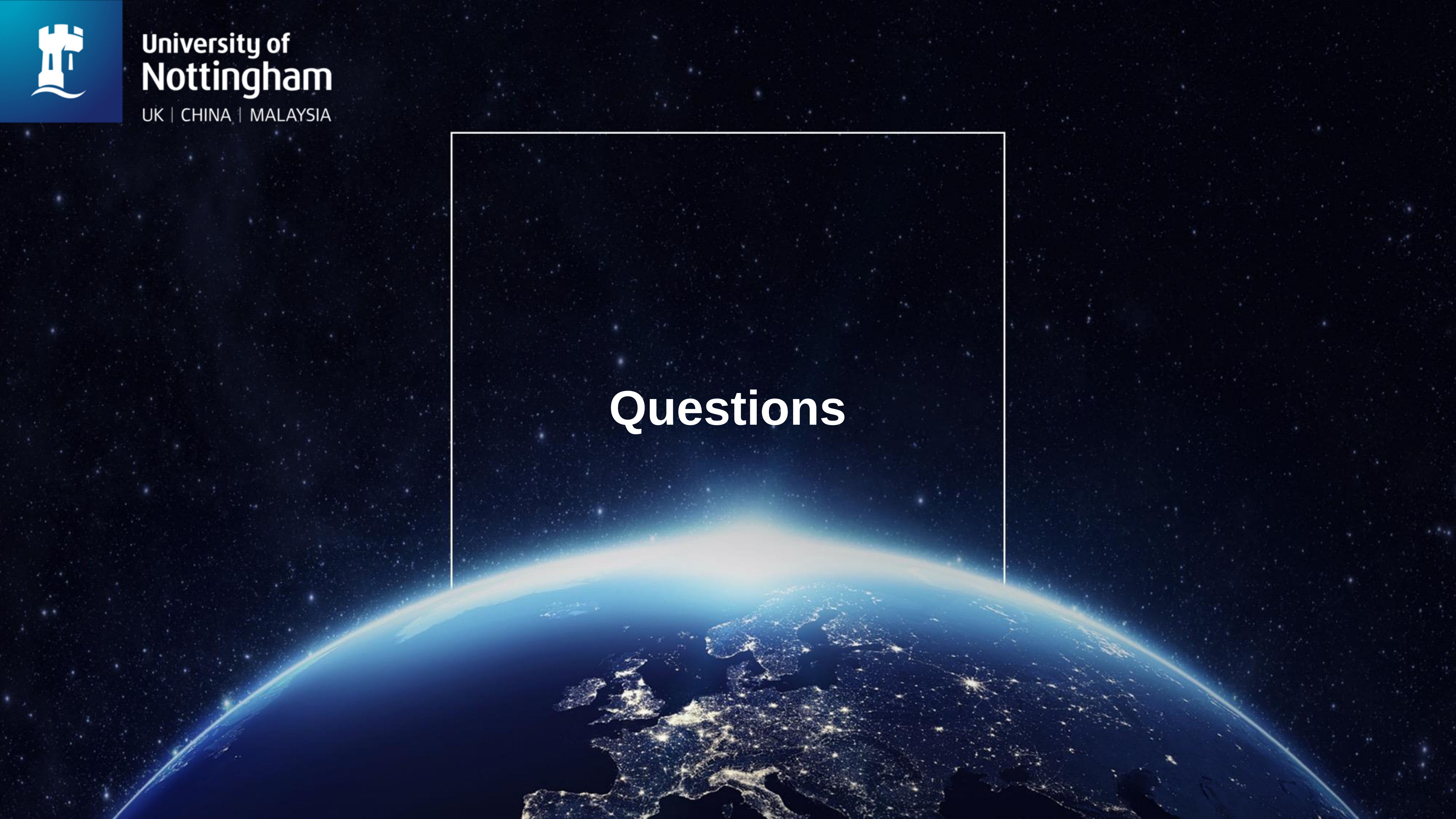
Animation extracted from: Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S. and Uszkoreit, J., (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

[1] **Extracted and modified from**: Hettiarachchi, H. (12 August 2023). *Unveiling Vision Transformers: Revolutionizing Computer Vision Beyond Convolution.* https://medium.com/@hansahettiarachchi/unveiling-vision-transformers-revolutionizing-computer-vision-beyond-convolution-c410110ef061

Summary

1. Background
2. Components
3. Applications
4. Future

University of Nottingham
UK | CHINA | MALAYSIA



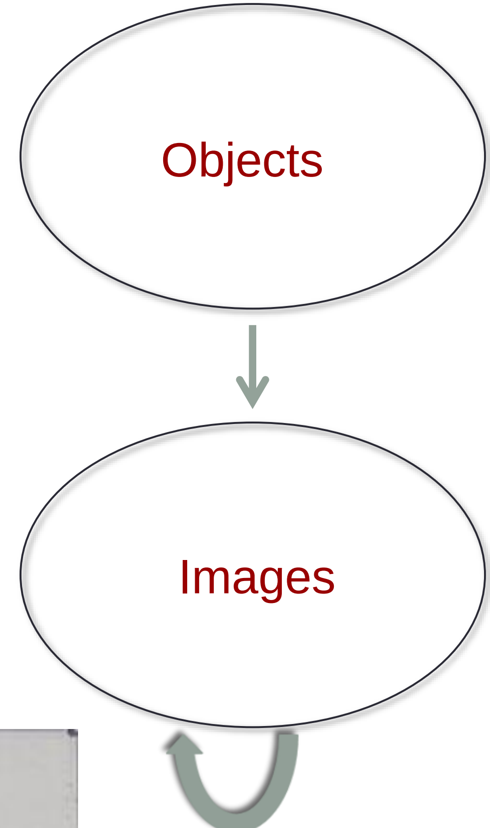
Questions

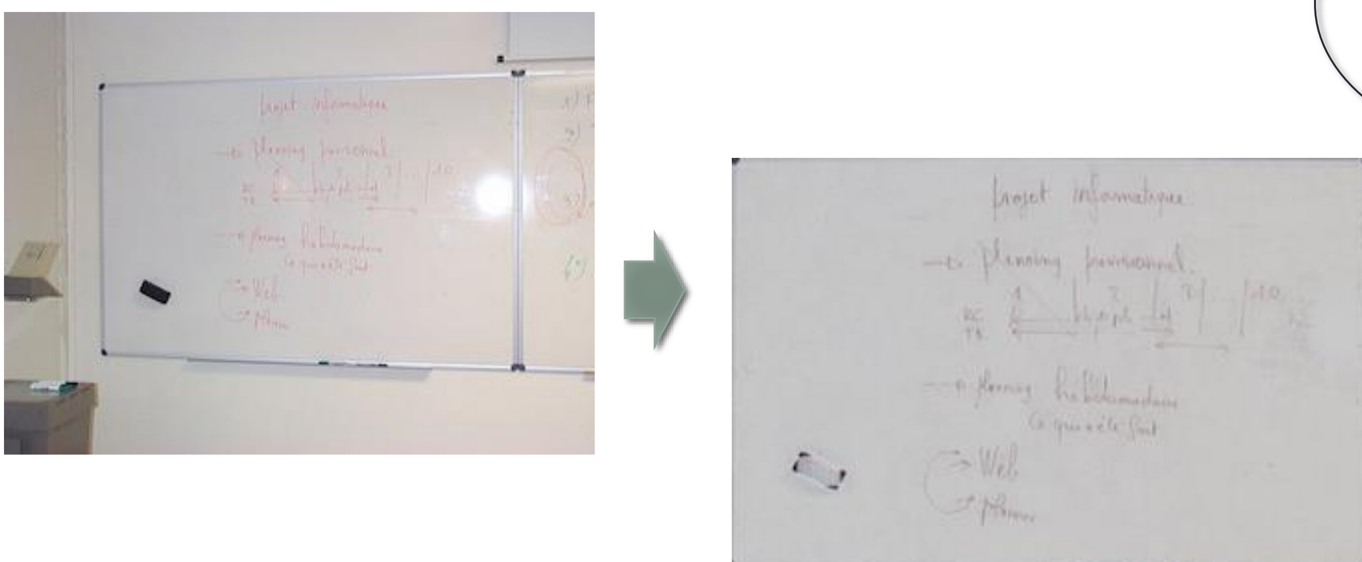
COMP2005

The End – The Whiteboard Problem Revisited

Remember This?

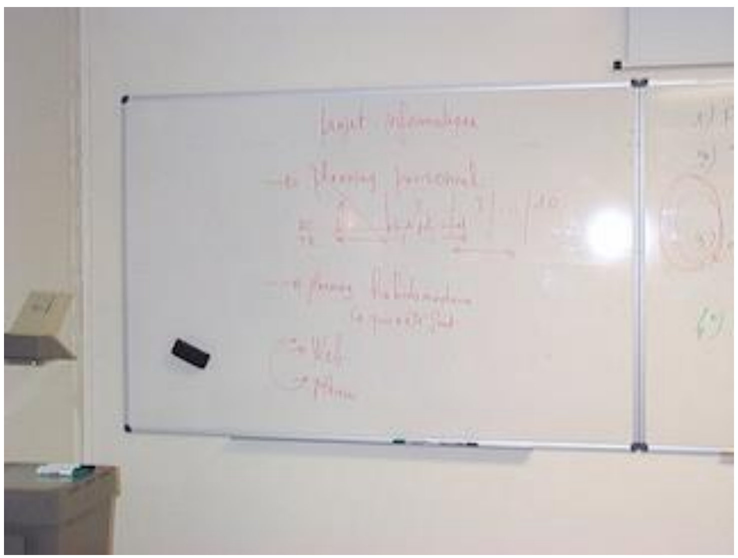
- Image(s) in, image(s) out
- Key information more easily seen/extracted
- More aesthetically pleasing

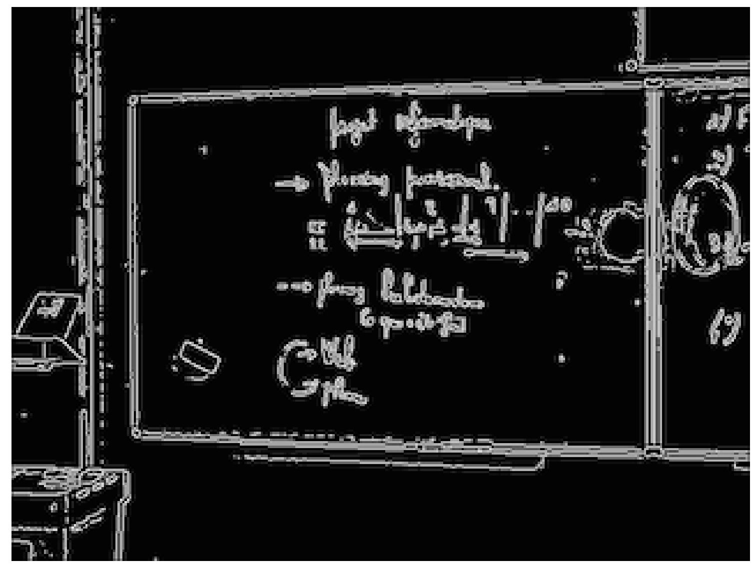




Whiteboard Problem Revisited

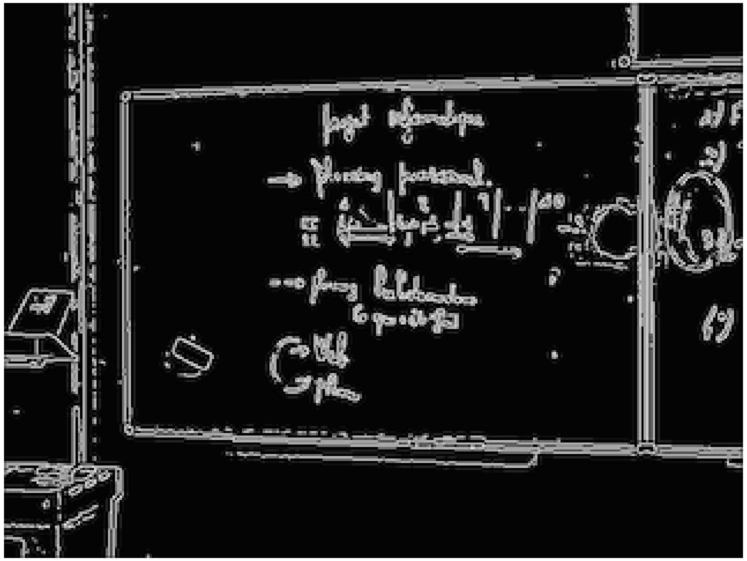
- Step 1: Edge detection
  - Achieved here with a variation on the Canny operator

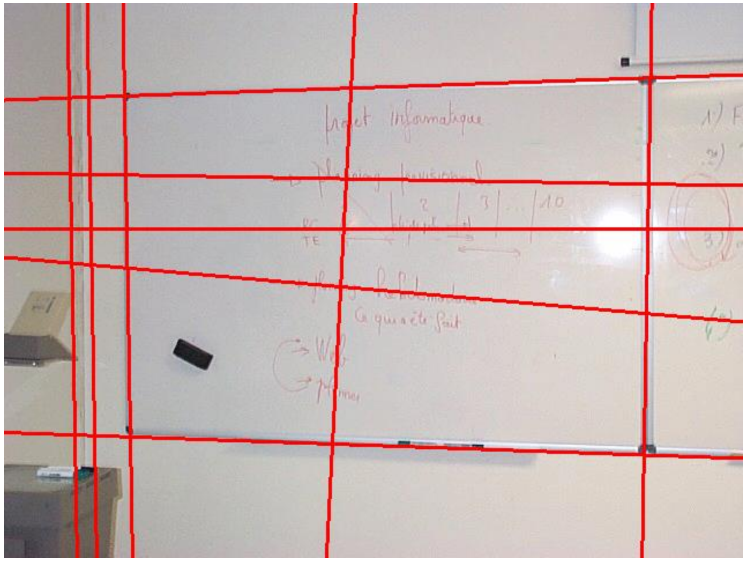




Whiteboard Problem Revisited

- Step 2: Line Finding
  - Two Hough Transforms; one detecting near horizontal (20^{o} to -20^{o}) and one near vertical (70^{o} to 110^{o})

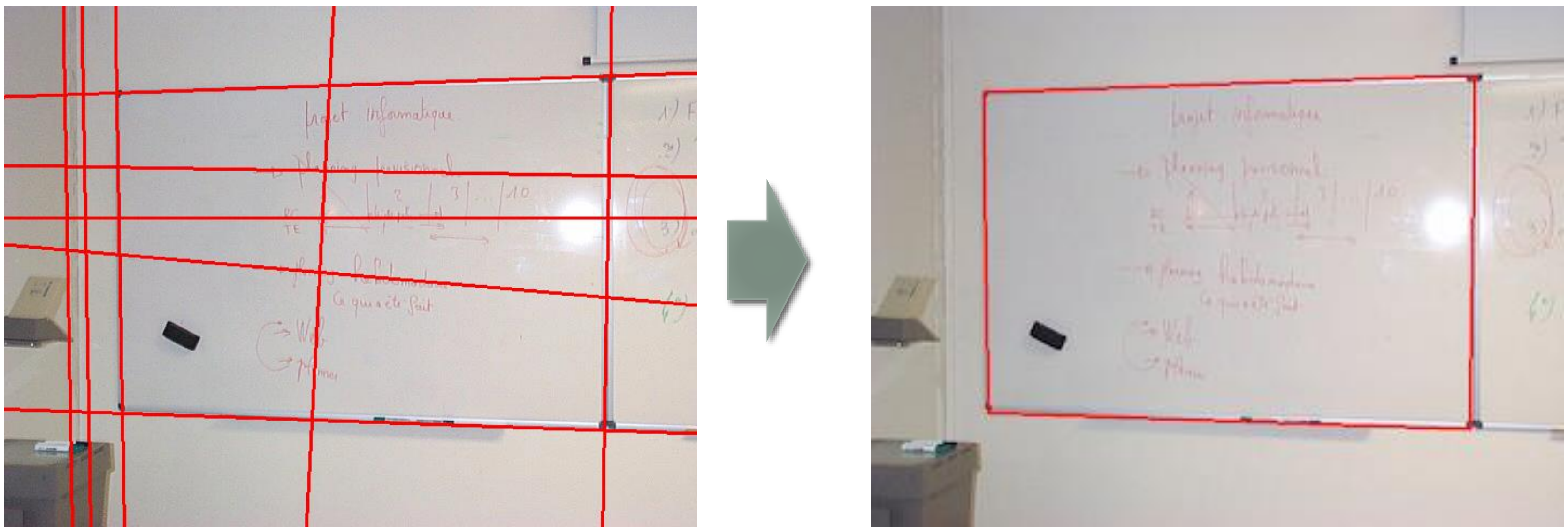




  - Keep only the 5 longest horizontal and vertical lines

Whiteboard Problem Revisited

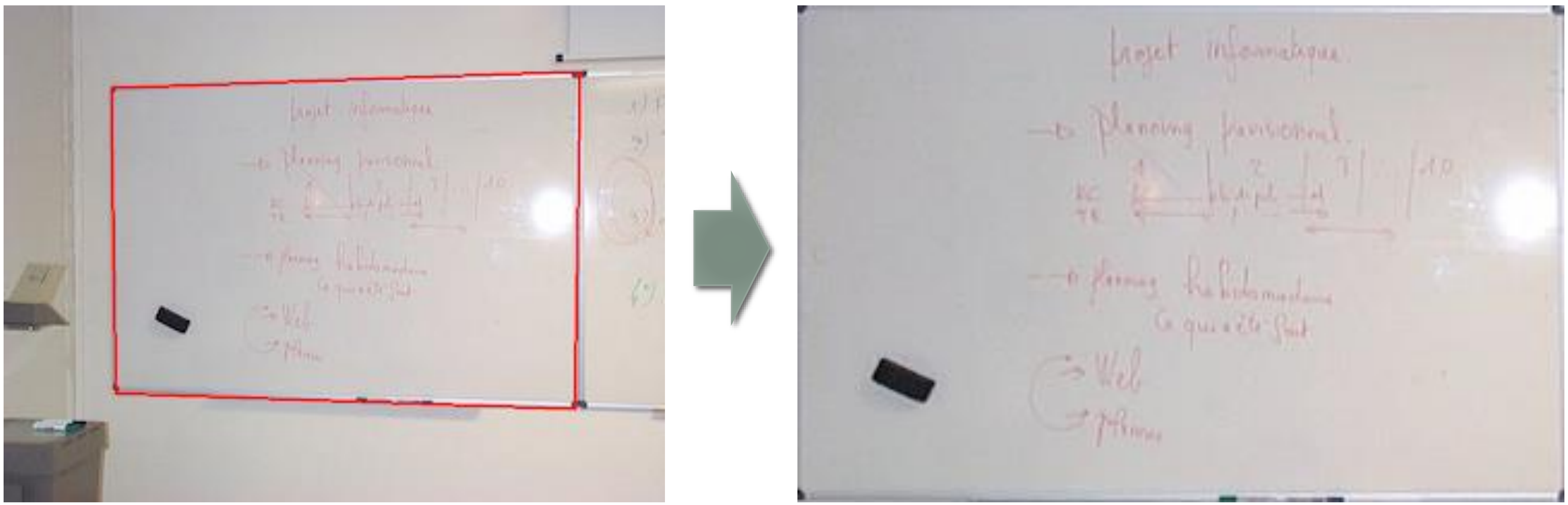
- Step 3: Detect whiteboard border
  - Find quadrilaterals above threshold size with neighbouring edges at 90º and opposites sides oriented the same (+/- 30º)



  - Pick the one which lies over the most edges

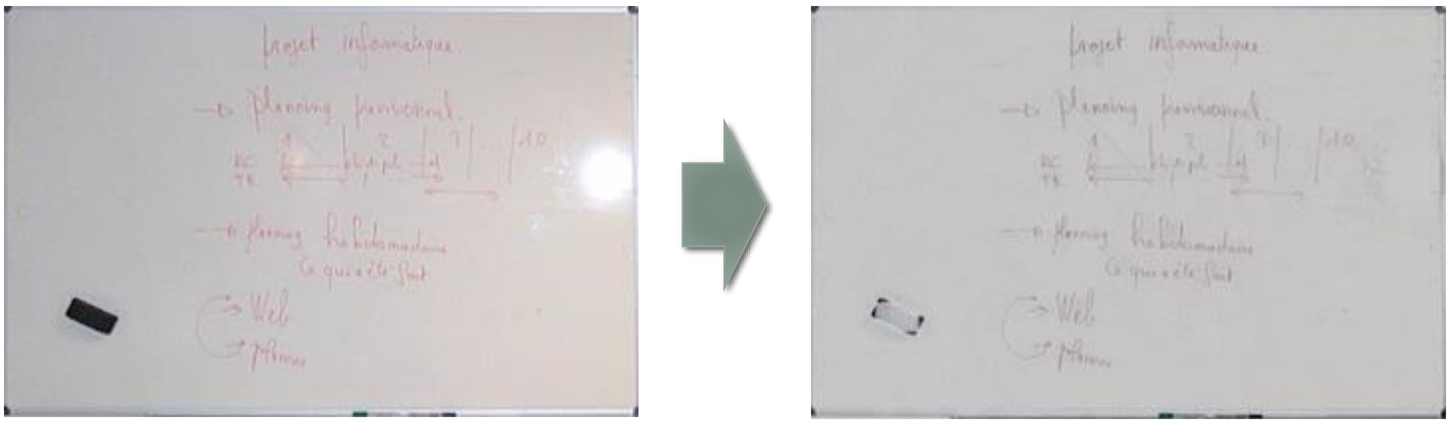
Whiteboard Problem Revisited

- Step 4: Distortion Correction
  - This requires Geometric Transformation of the image



Whiteboard Problem Revisited

- Step 5: Illumination Correction
  - Need to enhance the high frequencies (writing)



  - Frequency domain processing is one way
  - In spatial domain:
    - <u>Approximate</u> low frequency component by smoothing
    - Subtract smoothed image from original to <u>approximate</u> high frequencies
    - Add mean of “low frequency” image to “high frequency” image to make it bright enough

The End

- Image formation and acquisition
  - Background material: cameras, Bayer patterns
  - Basic terminology: sampling and quantisation
  - A little image processing: re-sampling, re-sizing, re-quantisation
- Colour spaces
  - RGB, HSV, LAB etc
  - Image pre-processing: choosing a colour space is a key step in practical applications, but not really IP
- Point transforms
  - Gain, bias, contrast stretching, gamma correction
  - Simplest methods, *but useful*

The End

- Spatial Filtering
  - Convolution is key
  - Noise removal: mean, Gaussian filtering
  - Enhancement: unsharp masking, Laplacian filtering
  - Edge detection: Roberts, Sobel, Marr-Hildreth, Canny
- Non-linear filtering
  - Median, anisotropic diffusion, bilateral filtering

Linear and non-linear filters are at the heart of the image processing toolbox

- Thresholding and Binary Images
  - Otsu, Unimodal thresholding
  - Adaptive Thresholding
  - Connected components, morphology (erosion, dilation)

The End

- Histogram methods
  - Histogram equalisation
  - Comparing images: histogram intersection, histogram ratio
  - An application: image retrieval
- Frequency domain(not covered this year)
  - An overview, broad structure of frequency domain methods
  - Something you need to be aware of

The End

- Compression
  - *Increasingly important*
  - Types of redundancy: coding, spatial, psychovisual
  - Structure of compression systems
  - Components and complete schemes: Huffman coding, GIF, JPEG
- Segmentation and Line finding
  - Region growing, split and merge (quadtrees), watersheds
  - Hough transforms
- Convolutional Neural Network

A set of tools that can be used to create image processing pipelines

Primary Text Book

R.C. Gonzalez and R.E. Woods. (2018). *Digital Image Processing*. (Fourth Edition). Prentice Hall.

[Available in the Library]

For more details on each topic please refer to the priary text book for this module

Module Aims

- To introduce the fundamentals of digital image processing - theory and practice.
  - **Assessed by exam: how do techniques work, and what do they do**
- To gain practical experience in writing programs to manipulate digital images.
  - **Assessed by coursework – no coding in the exam**
- To lay the foundation for studying advanced topics in related fields.
  - **G53VIS next year?**

The Exam

- 1 hour exam
- Answer All question
- Focus on image processing methods
  - What they do
  - How they work
  - When they are appropriate
  - *Knowledge, Comprehension, Application*
  - Questions are (loosely) structured, topic area indicated

Read and answer the question

Take no. of marks available into account