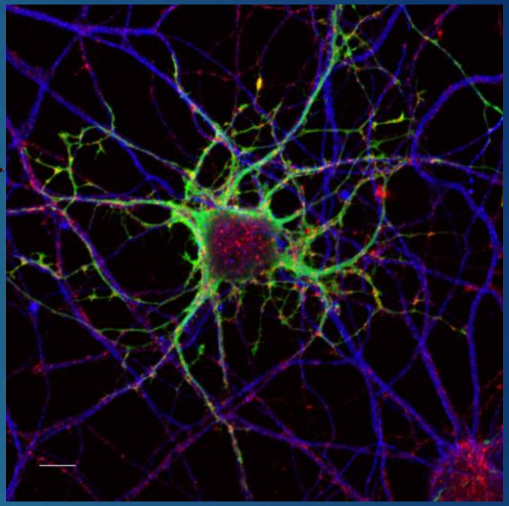
Convolutional Neural Networks



INTRODUCTION TO IMAGE CLASSIFICATION

(MODULE 6 – VISION MODELS) – By Christelle JULIAS

Content

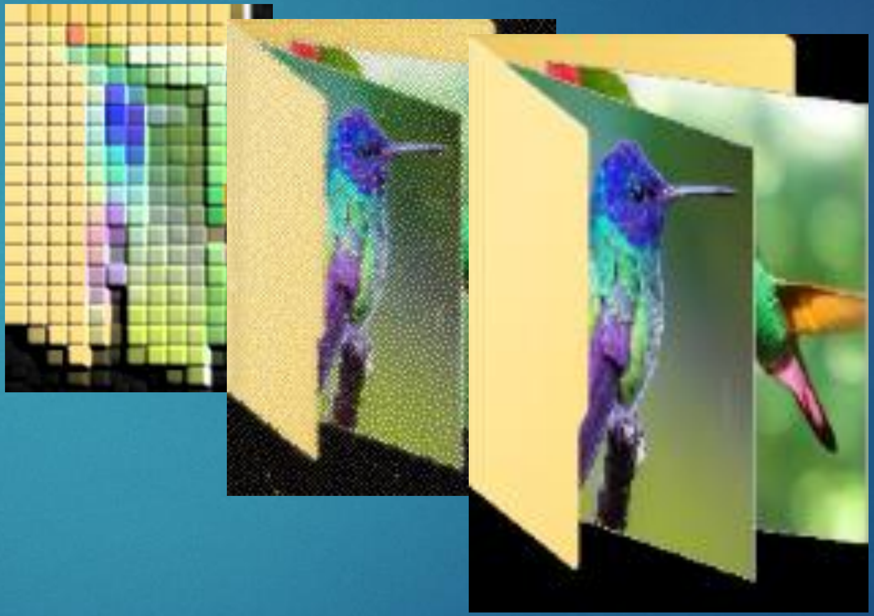
- Discovery and Overview
- [Computer Vision - Images]
- Use Cases of CNNs IRL
  - Image Classification
  - Object Recognition
- What are Convolutional Neural Networks?
- How do they work?
  - Core Components: Convolution, Filters, and Pooling layers
- Let’s explore!
- How do we manipulate Convolutional Neural Networks?
  - Creation
  - Training
  - Fine-tuning
  - A bit of Maths

Images: Hummingbird by Pluspng , neuron, by Flickr

Muffins or Chihuahuas?

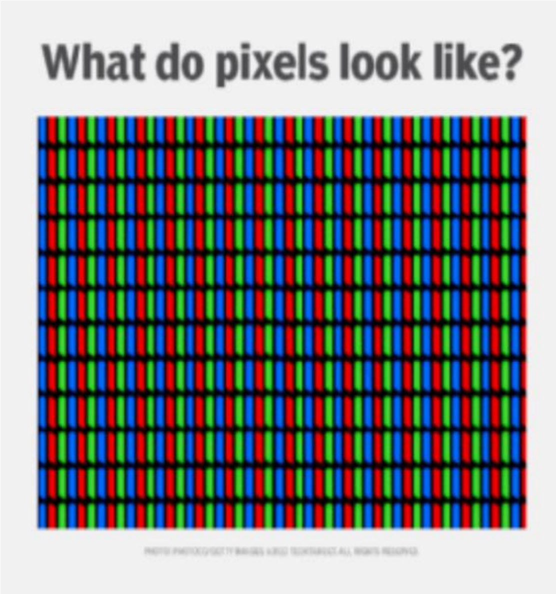
Introduction

WHAT CAN WE DO WITH IMAGES?



Images are « sets of pixels »

1. A **Pixel** (short for "picture element") : smallest unit of a digital image or display.
2. Represents a **single point** in an image. Measure of size for calibration. Larger pixels capture more light, smaller provide finer details.
3. Displays various colors based on its **RGB** (red, green, blue) **components**. In Black &White:binary, in Grayscale: B, W & shades of gray.



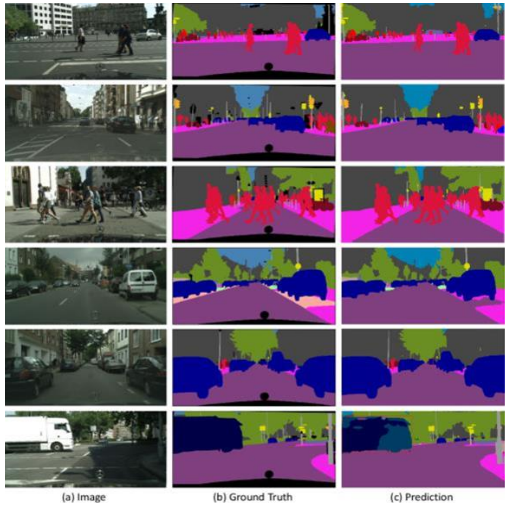
- **Composition:** made up of subpixels, typically three for RGB color representation.
- **Resolution:** total number of pixels in an image. Ex: a 1920 x 1080 display has over 2 million pixels.
- **Function:** More pixels generally lead to clearer and more detailed images=> impact the overall quality of visual displays (ex.FullHD)

Image Processing techniques include

- **Enhancement:** parameters such as brightness, contrast, and sharpness
- **Restoration:** revamp degraded original images
- **Segmentation:** divide into "semantic/meaningful" parts according to the pixel distribution.

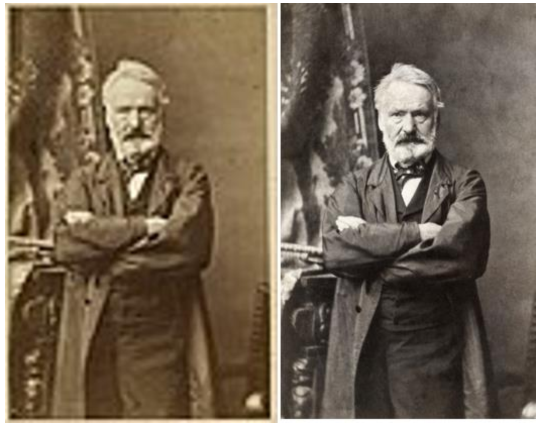
- **Compression:** to reduce the file size through lossy or lossless methods (MPEG, HEIF)
- **Generation:** GANs (Generative Adversarial Networks) to create new images or enhance
- **Morphological Processing:** shape image structures (twist, wrinkle etc.)

Examples



Semantic Segmentation

Ex: spatial data within contexts (streets, hours, references with self-driving cars), pixel level classification



Restorative Filters

Ex: Antique Photos (here Victor Hugo, the French Novelist of the !9th Century, by Bertall !867)

Morphological Processing + Generation

Ex: Beauty filters (Instagram, Adobe, PicsArt …), Background removal and replacement

Types of Object Recognition Algorithms

Image classification

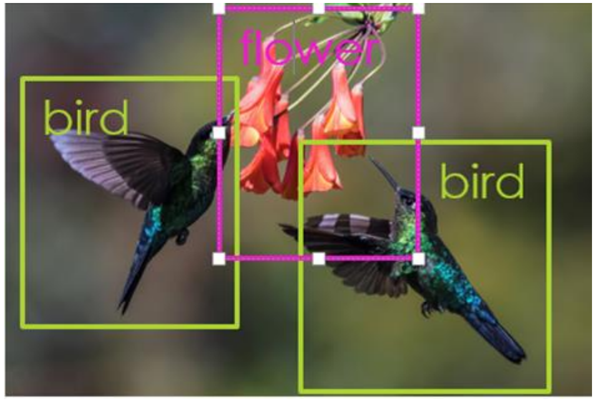
- -**Classifies** / labels the object contained in the picture (what is it?)
- -Provides a **probability** of the guess~(it's 90% likely to be a bird)

Traditional CNNs

Both Classification & Localization

- **Detects** the objects and also localizes it in the picture simultaneously(where is the bird?)

Simplified Yolo, R- CNNs
(You Only Look Once, Region CNN models)



Object Detection

- **Detects** the content of an image, presence of **several objects** (many bounding boxes) (what are those?)

Yolo, R- CNNs

QUIZZ

Identify real world application of Image Classification

Image Classification / Recognition

[Is it a [label, name]?]

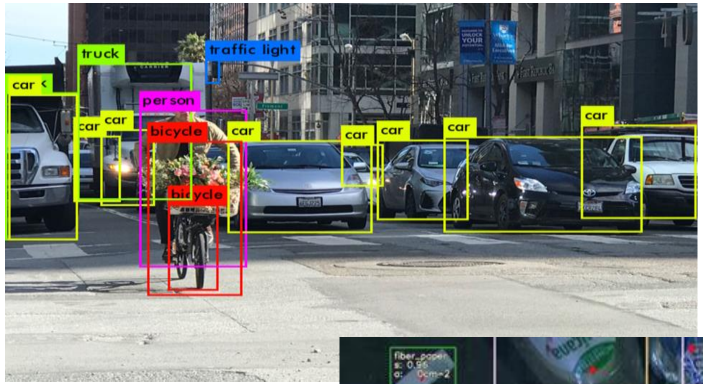
-Classifying images into predefined categories.
- Recognizing **patterns** and **features**.
- Widely used in applications like: **photo tagging** in social media and **medical imaging** for diagnostics identifying pathologies.

Images:petapixel, projectpro.io

Object Detection

(Presence: Is there an [object name]in the image / landscape/ supermarket aisle etc.?

- - **Identifying** and **locating** objects within a picture simultaneously.
- - Applications include **self-driving cars, facial recognition, security cameras, sorting/ recycling wastes, tracking wildlife** etc



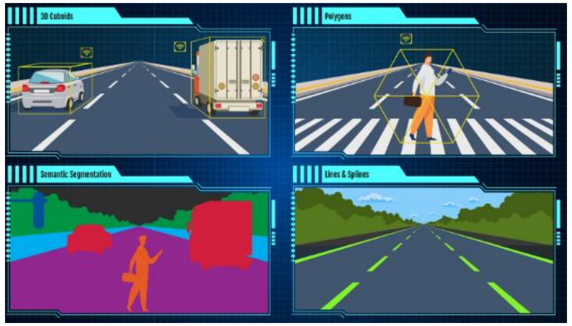
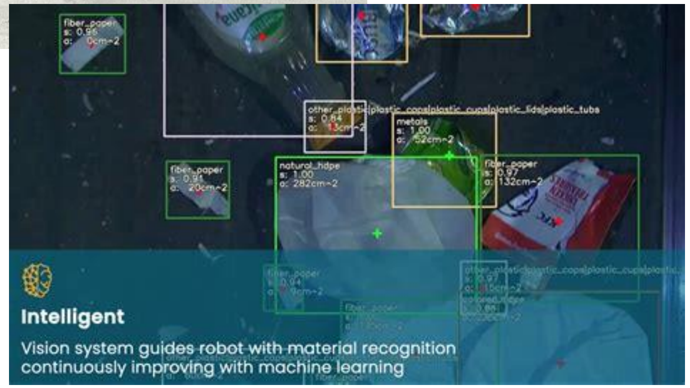


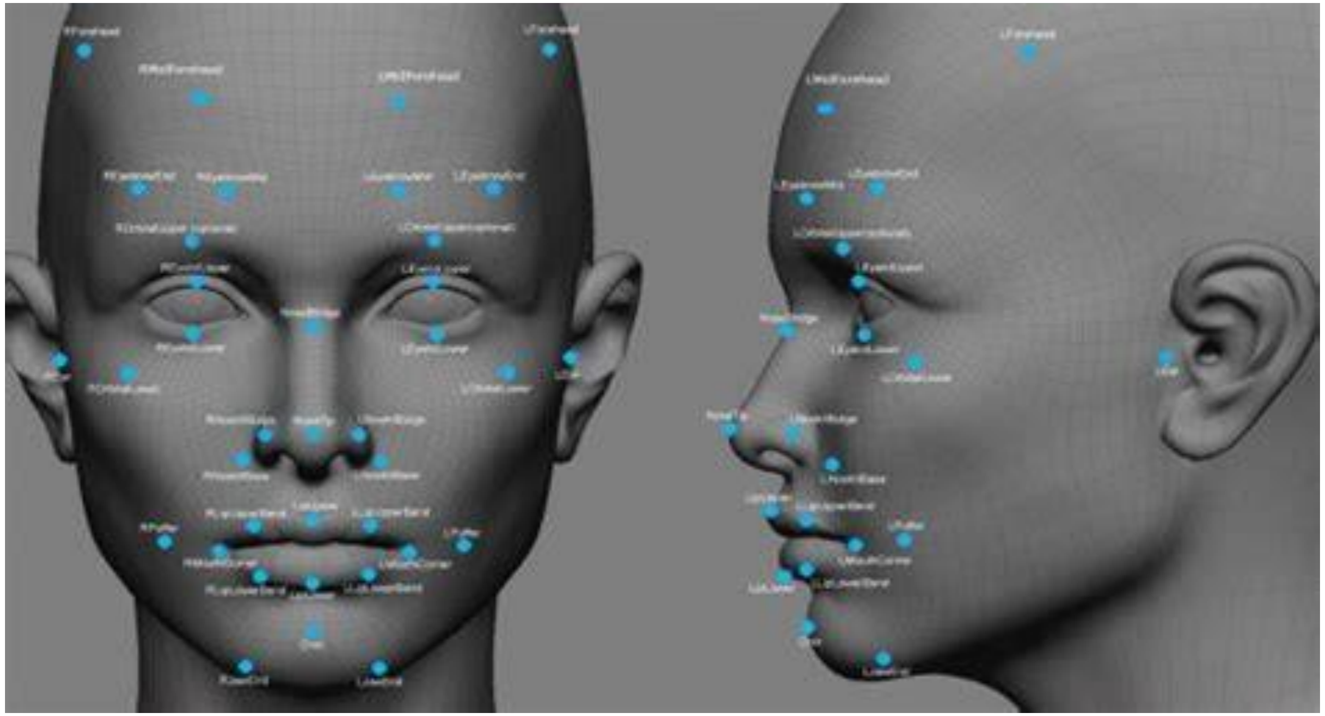
Image:ywseo



Face Recognition

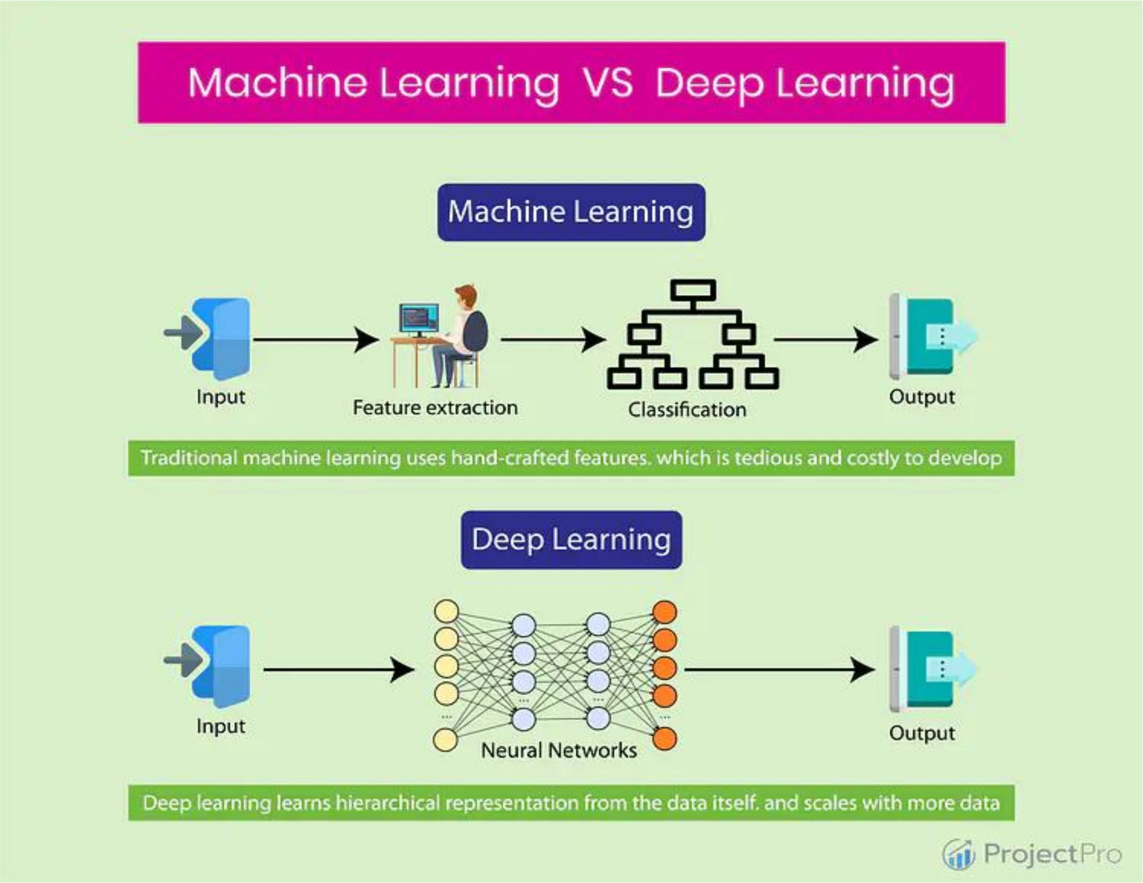
[is it {person/celebrity_name}/?]

- Attaching clues or "landmarks", to distinctive features
- **Biometric screening**
- **ID tracking** across video frames
- **Emotional intelligence** (microexpressions)



Machine Learning vs Deep Learning

- Traditional **machine learning** uses hand-crafted features which is tedious and costly to develop
- **Deep Learning** learns hierarchical representation from the data itself and scales with more data
- Deep Learning is a subset of Machine Learning. They are complementary.



Deep Learning approach #1

Supervised Learning

Requires **labeled datasets** for training models (e.g., convolutional neural network for image classification)

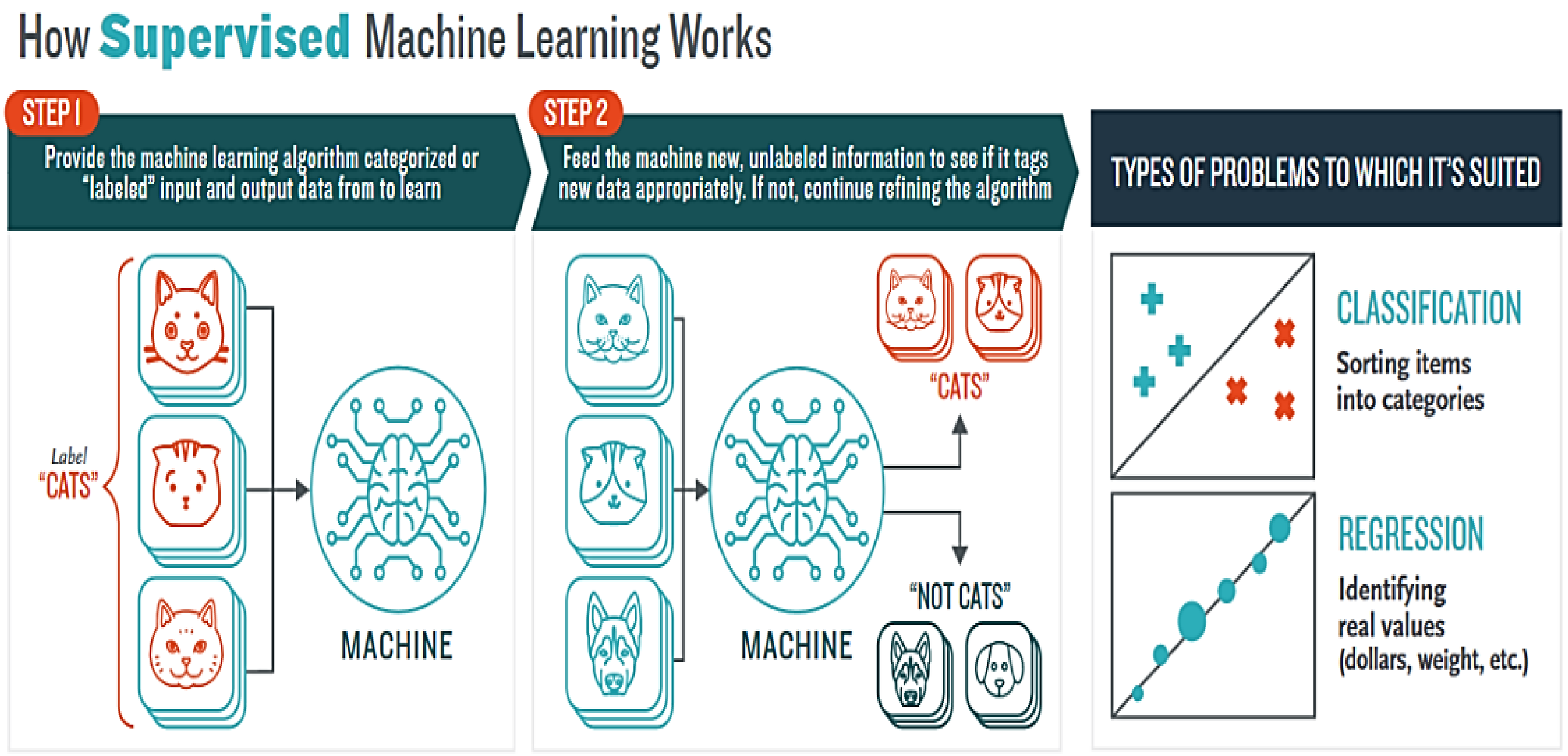


Photo:Lotus

Deep Learning approach #2

Unsupervised Learning

utilizes methods, such as autoencoders and generative adversarial networks (GANs) to learn from unlabeled data.

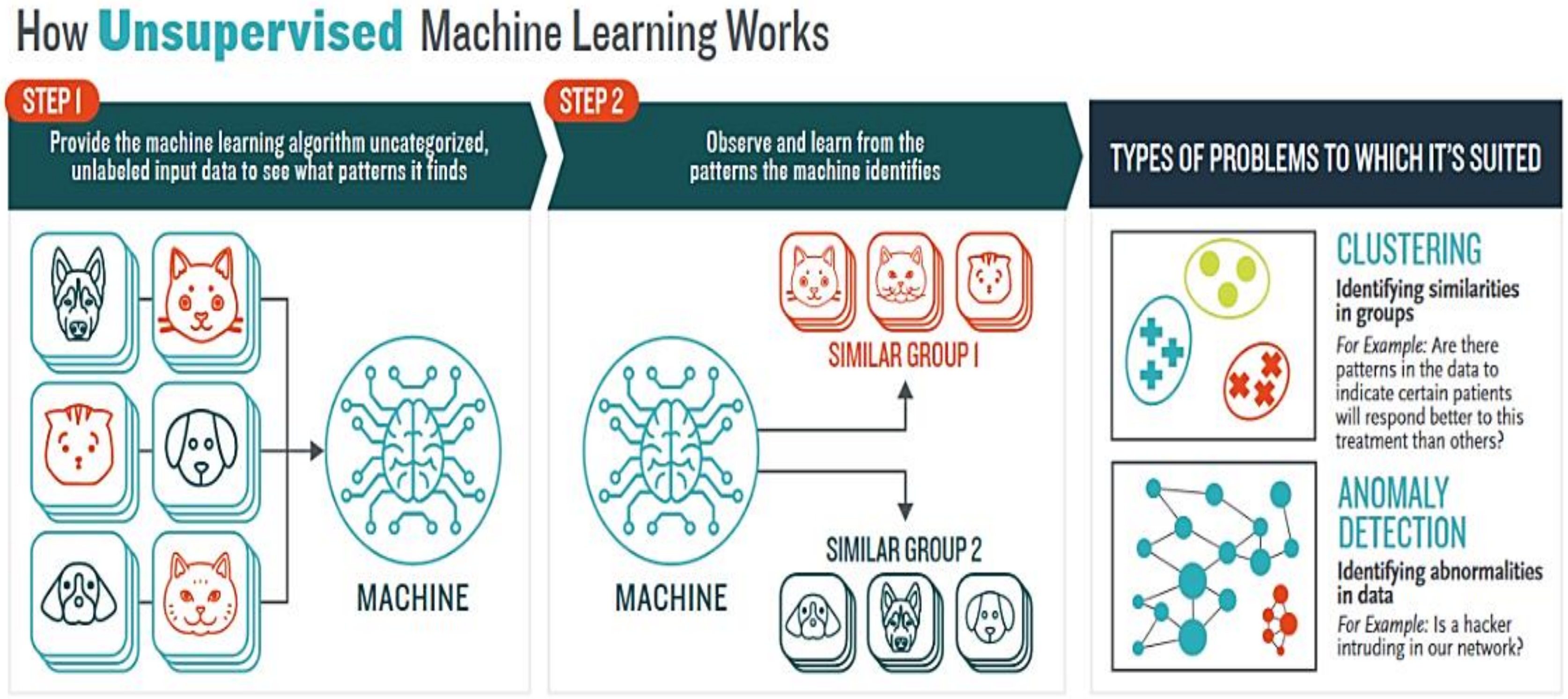
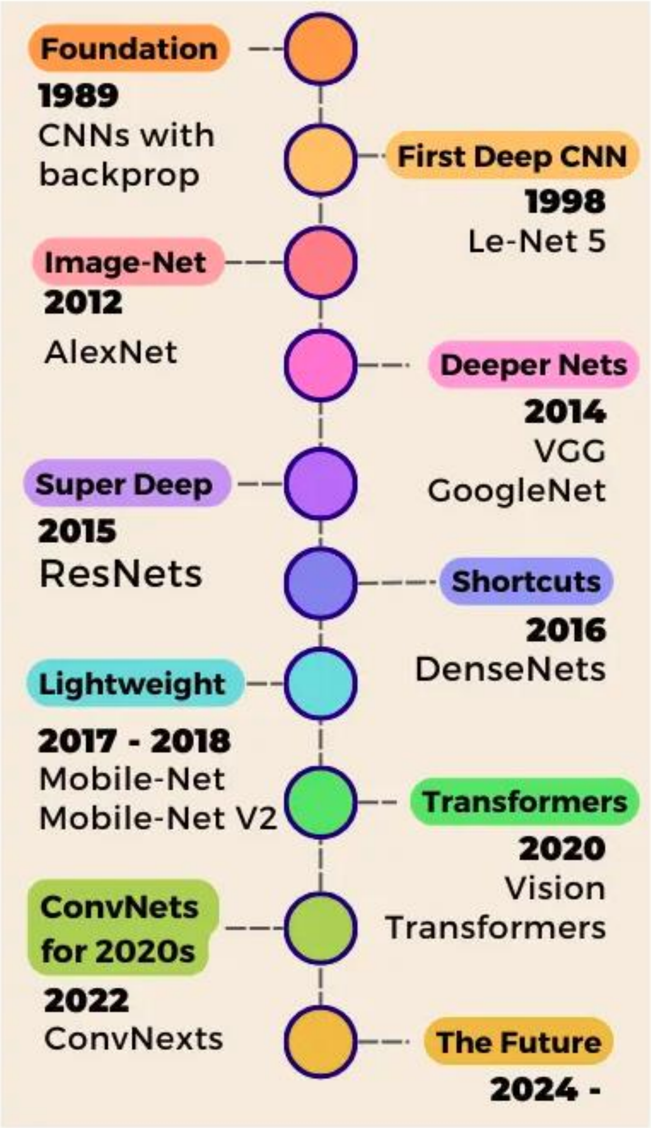


Photo:Lotus

What are Convolutional Neural Networks?

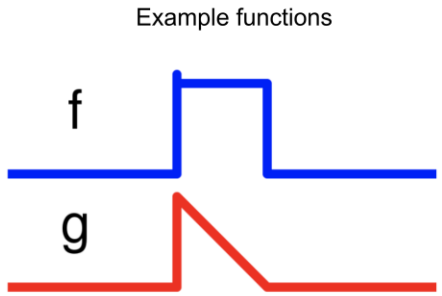
CONVNETS OR CNNS

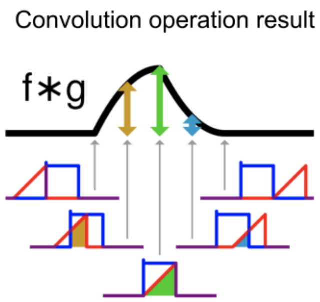


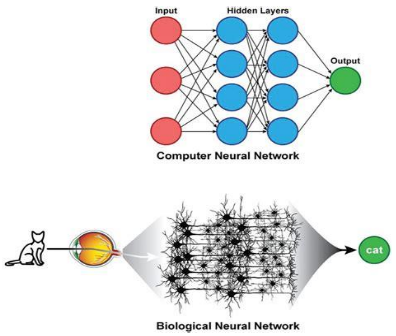
In the context of mathematics

$$(f * g)(t) \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} f(\tau)g(t-\tau)\, d\tau$$

To convolve a kernel with an input signal:
flip the signal, move to the desired time,
and accumulate every interaction with the kernel

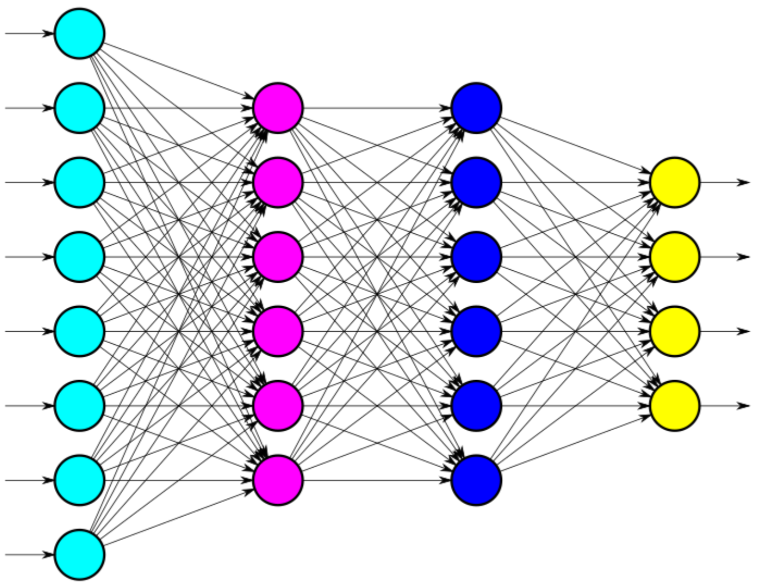






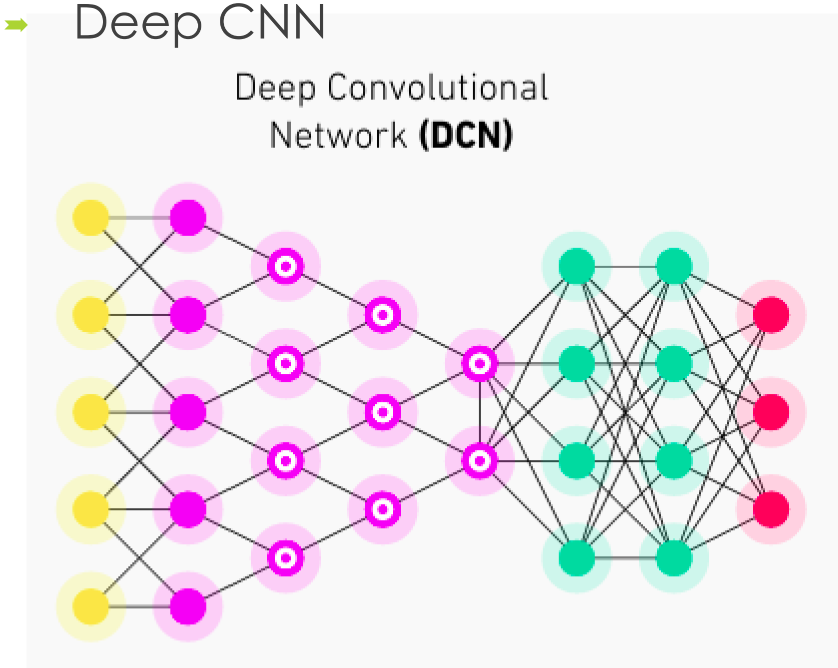
In the context of AI, what is a Convolutional Neural Network?

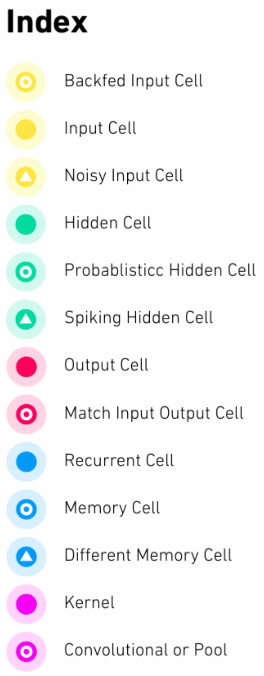
1. A type of **Deep Learning model** designed for image processing, also called ConvNet or CNN
2. **Mimics visual perception** in humans
3. A powerful tool for **visual recognition** and **style manipulation**
4. Consists of multiple **sequential layers** (convolutions, filters, and pooling) to extract features (edges, texture) from images ex: the cat has pointy ears, and a round face
5. Essential for modern applications in **Computer Vision** (multimodal multimedia: video, images)



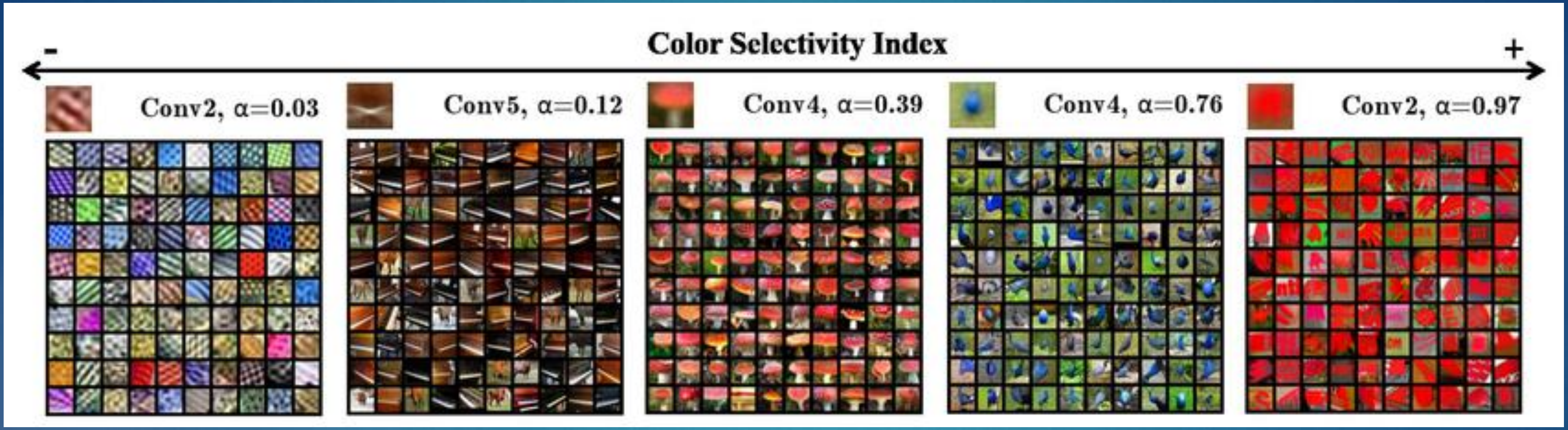
Architecture of a CNN

- Deep CNN



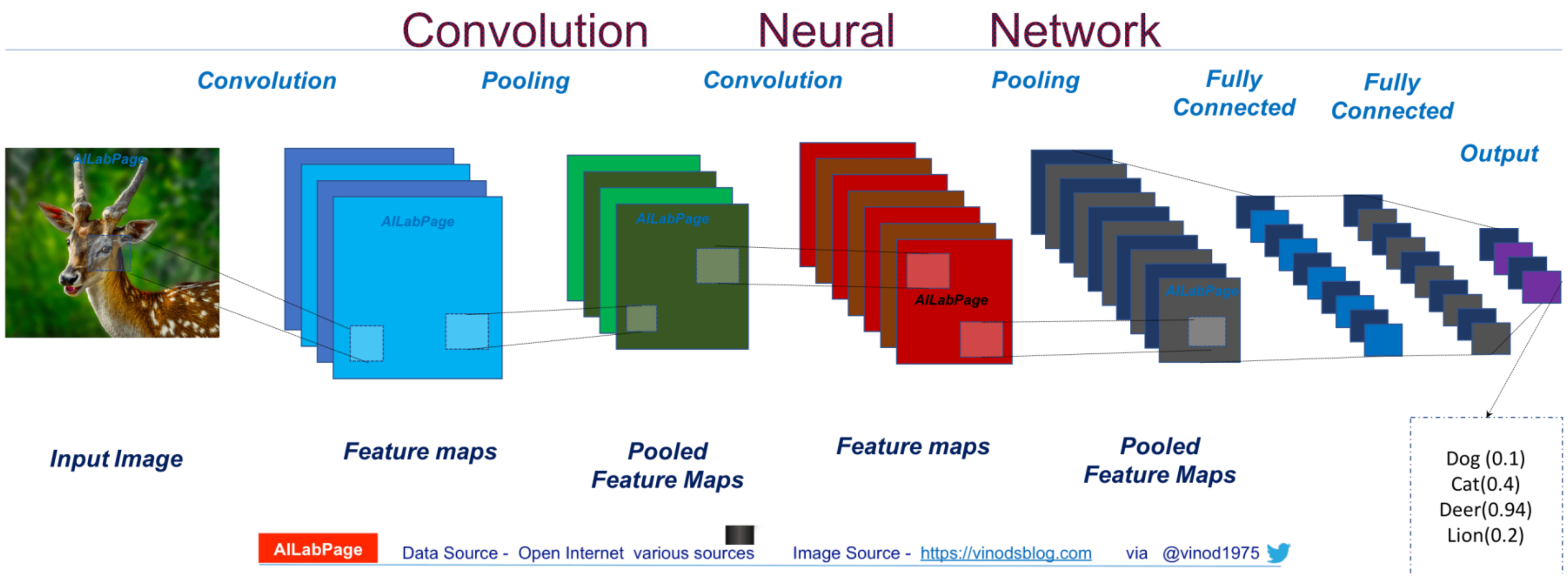


Source: Asimov Institute



COLOR SENSITIVITY INDEX, SOURCE: RESEARCHGATE

CNN for image classification



The Deer class gets 94% of accuracy

Image Classification with CNN

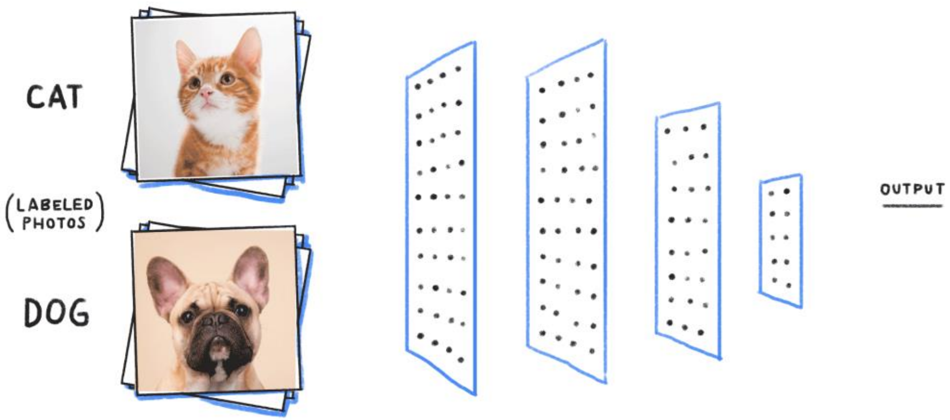
Classification is essentially a guess deduced from the extracted features and object detection may follows it once the object has been recognized.

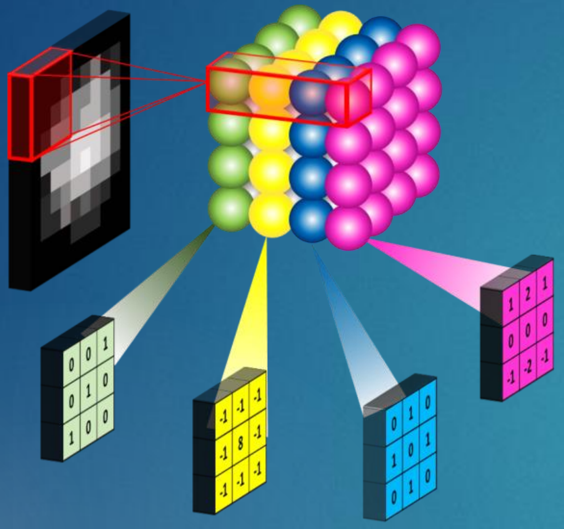
1- **Feature extraction** performed by CNNs

2- **Pooling** as a dimensionality reduction of feature maps

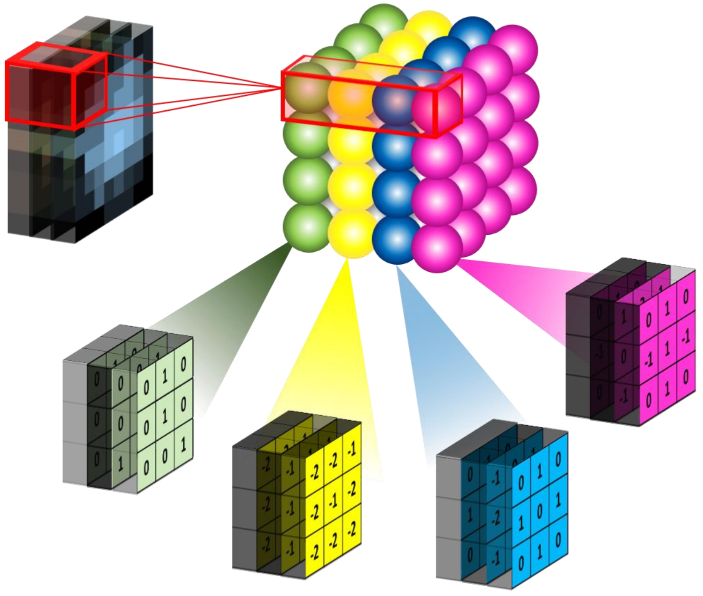
3- **Classification,** once the extracted features pass through fully connected layers (often using Softmax)

4- **Object Detection:** by predicting bounding boxes around detected objects





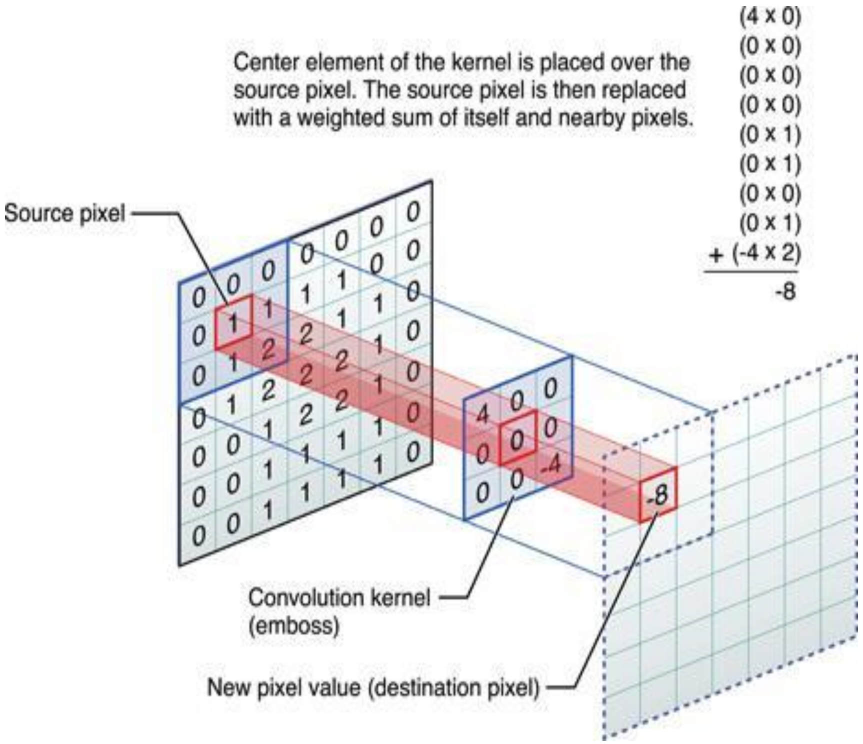
Convolutions



1D AND 3D

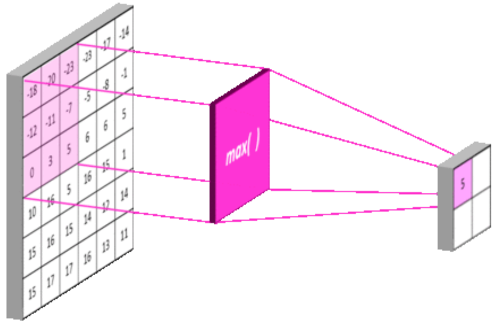
What are Convolutions?

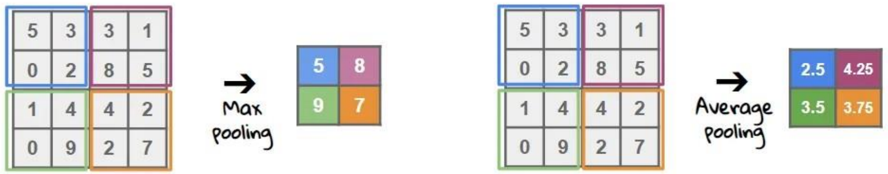
- - Definition: A mathematical operation that combines two functions.
  - Purpose: To extract high-level features from images.
  - Process: Apply a filter (kernel) over the image, computing dot products.
- -



Pooling Layers?

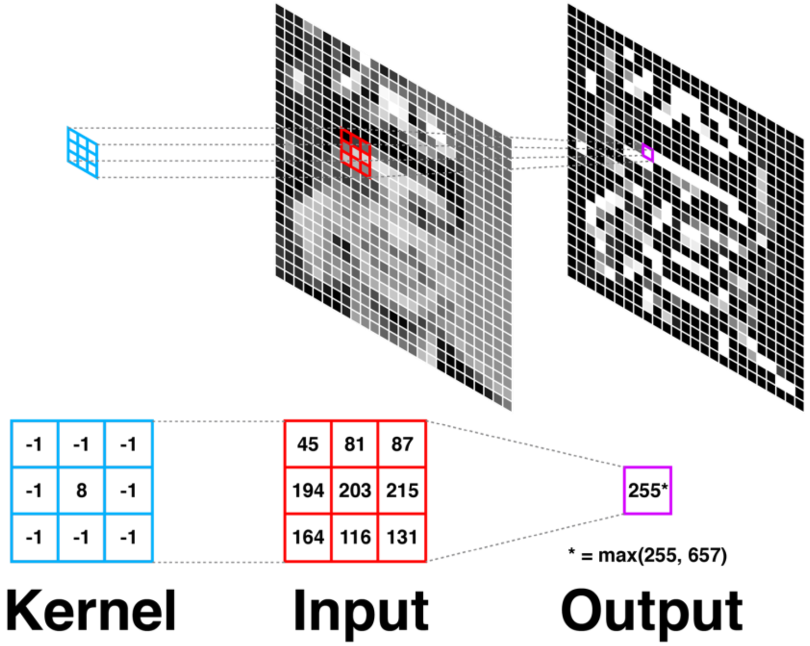
- **Function? Reduces spatial dimensions** of the feature maps.
- **Why?** Images can be voluminous (resolutions, size)
- **Solution:** Focus on the essential features
- **How?** Sliding a window and taking only one value
  - Types:
    - **Max Pooling:** Takes the maximum value from a set of values.
    - **Average Pooling:** Takes the average value.
    - **Benefits:** Decreases computational load and helps prevent overfitting.





Filters / Kernels

Small matrices used in convolutions.
- Detect specific features (Edges, textures).
- Different filters highlight different aspects of the image.

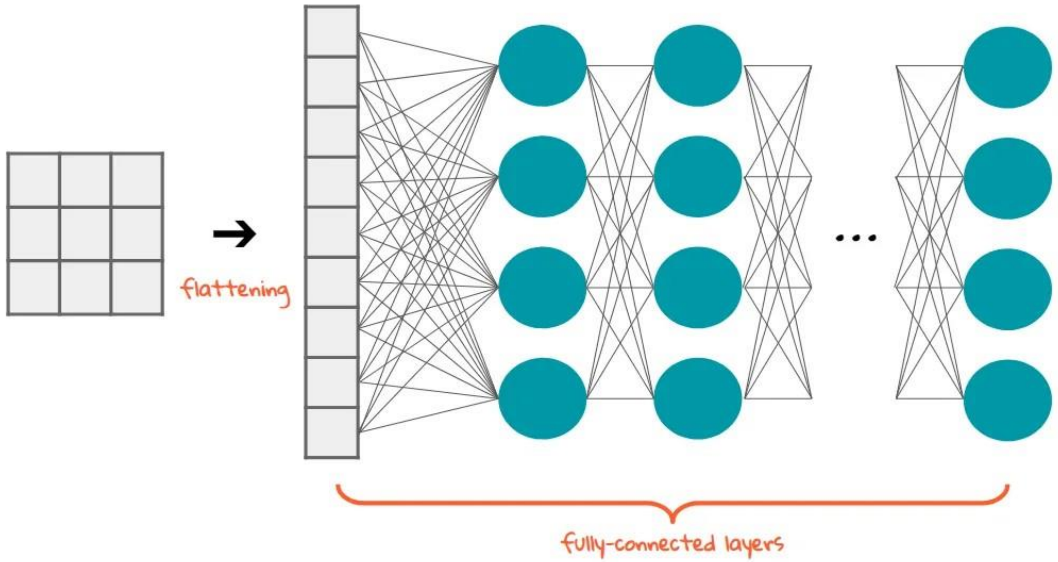


Flattening Layer

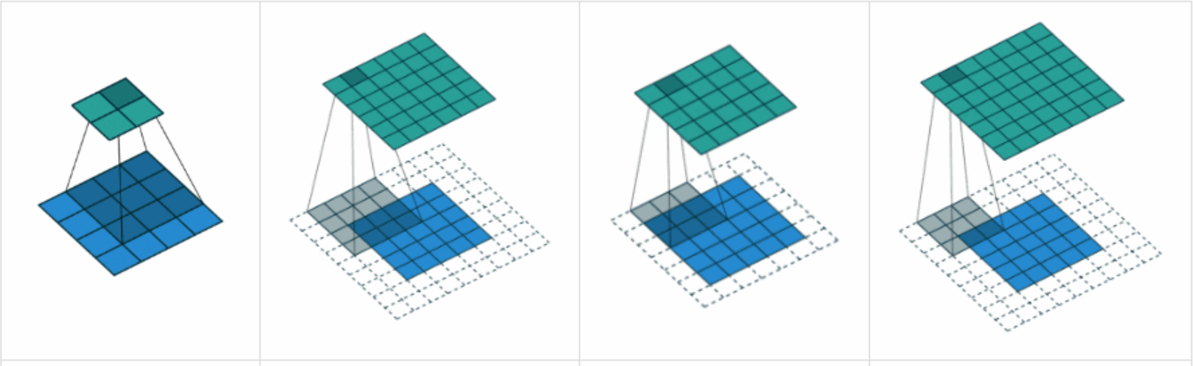
Converting multi-dimensional feature maps from the convolutional and pooling layers into a one-dimensional array.

Why? To connect extracted features to the final classification task. Fully-connected layers require linear input format.

How? Takes the output from the last pooling layer (dimensions: Height, Width, Depth)and reshapes into a single vector with a size Height x Width x Depth



Notion of Padding and Stride



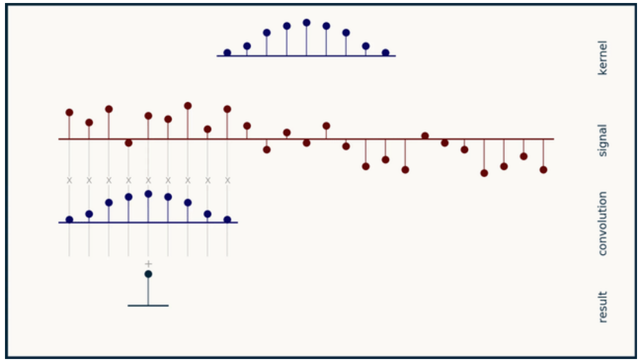
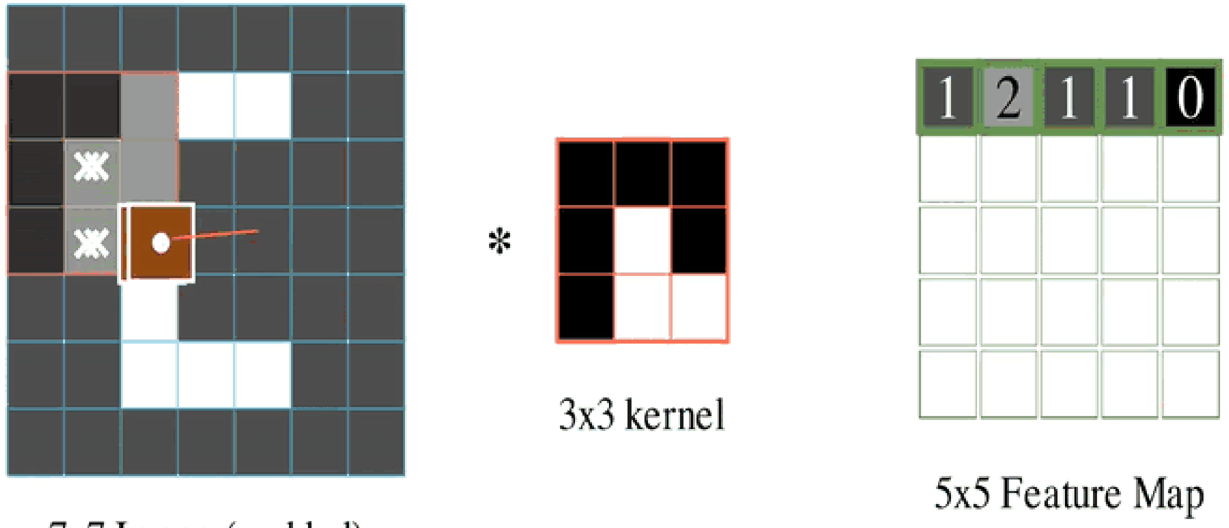


Image: Jiwon Jeong, e2eml.schools

Padding: adding extra pixels around input data before convolution (‘valid’ when none, ‘same’ when output match input)
Stride: number of pixels a convolution filter moves/ steps at a time
Crucial for effective feature extraction (relevant features) and model performance

How to get a feature map?



7x7 Image (padded)

Photo: Abishek Biswas

Training techniques

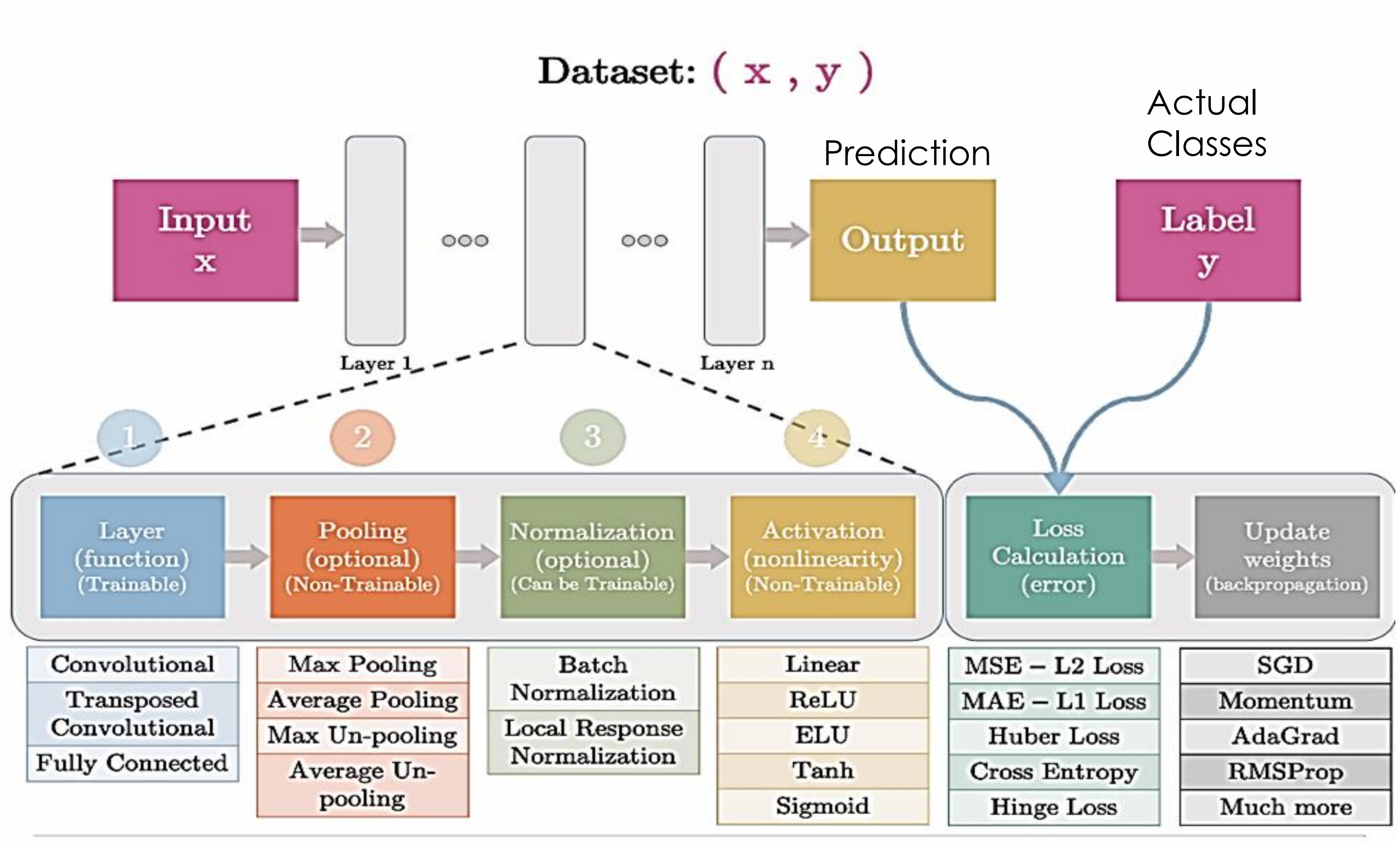
- **ReLu** activation function introduces non-linearity

-**Dropout** regularization prevents overfitting

-**Data Augmentation** provides more data (perspectives, side views, 3D etc)

-**Transfer Learning** reduces time of training and resources ex. deriving other feline families once the concept of a cat is understood by the model

Trainable means can be fine-tuned: Hyperparameters weights and biases



Infographics: Aqeel Anwar

Tools for Computer Vision using CNNs

Frameworks

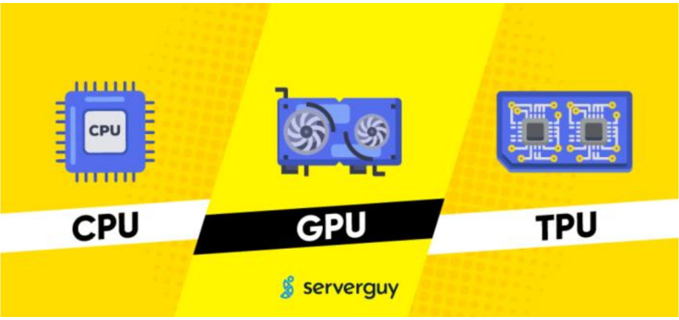
- **TensorFlow:** An open-source library for deep learning developed by Google, ideal for building CNNs.

- **PyTorch:** A flexible deep learning framework from Facebook, popular for research and production.

- **Keras:** A high-level API that simplifies building CNNs, often used with TensorFlow.

- **MATLAB:** Provides tools for designing and deploying CNNs through its Deep Learning Toolbox

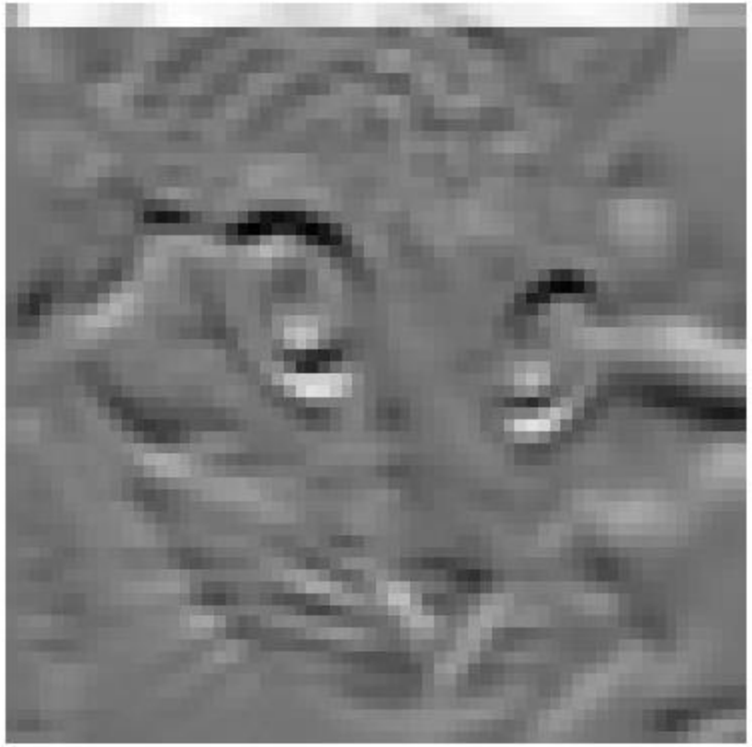
CPU: Central Processing Unit
GPU: Graphic Processing Unit
TPU: Tensor Processing Unit



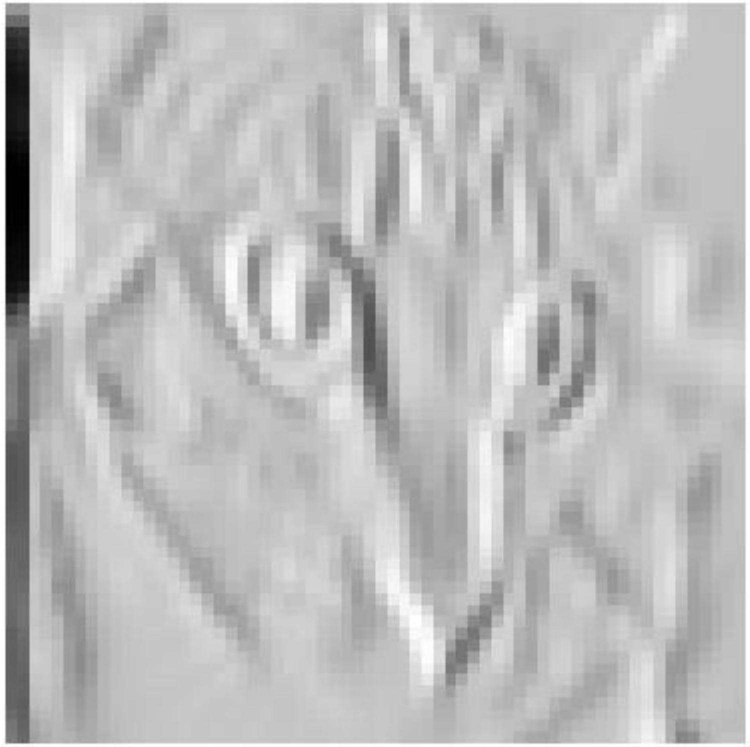
Filters

Original input image

- Horizontal edge filter

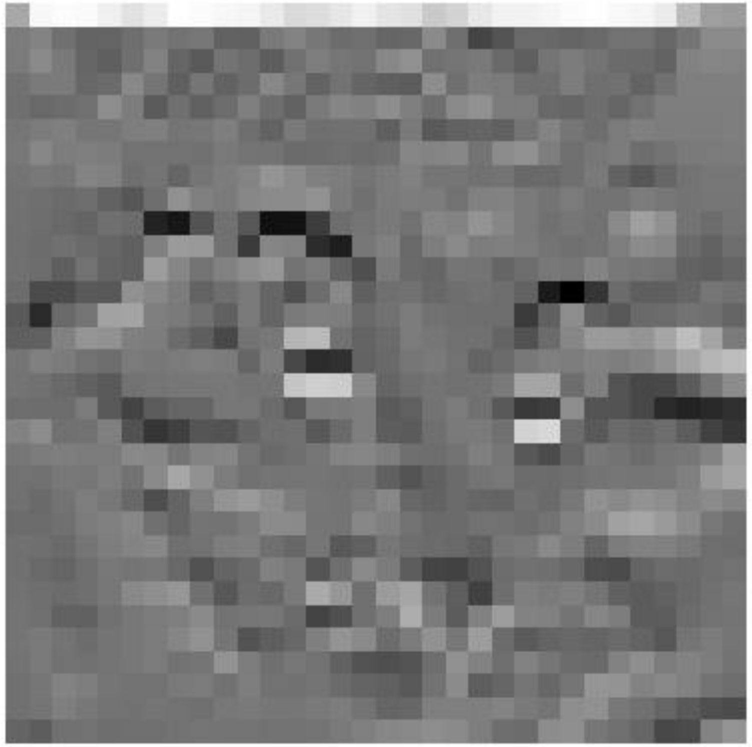


- Vertical edge filter

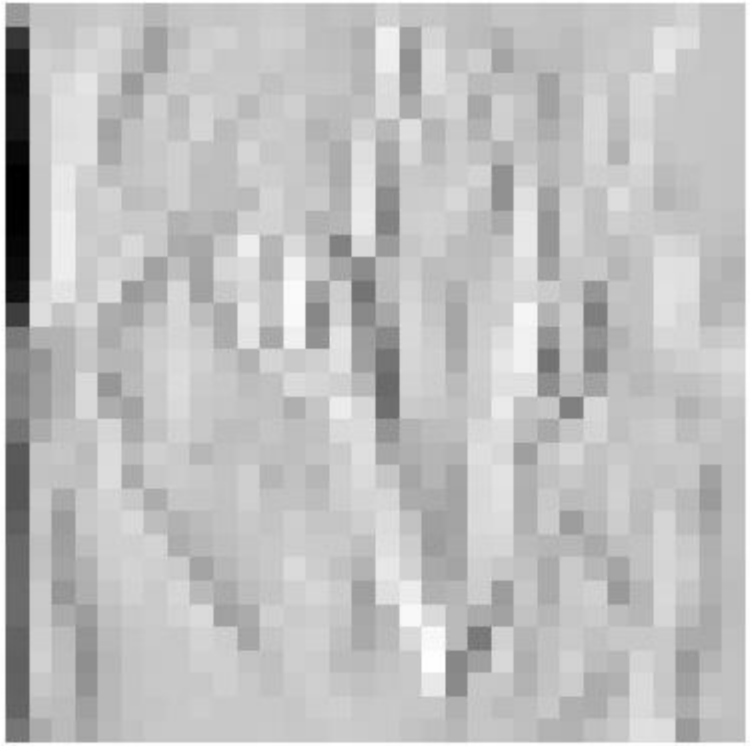


STRIDES

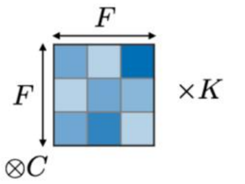
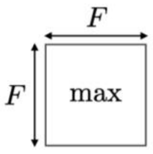
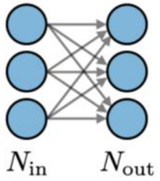
Horizontal edge filter with stride2



Vertical edge filter with stride2



Notion of complexity in training

	CONV	POOL	FC
Illustration			
Input size	$I \times I \times C$	$I \times I \times C$	N_{in}
Output size	$O \times O \times K$	$O \times O \times C$	N_{out}
Number of parameters	$(F \times F \times C + 1) \cdot K$	0	$(N_{\text{in}} + 1) \times N_{\text{out}}$
Remarks	• One bias parameter per filter • In most cases, $S < F$ • A common choice for K is $2C$	• Pooling operation done channel-wise • In most cases, $S = F$	• Input is flattened • One bias parameter per neuron • The number of FC neurons is free of structural constraints