# 1. A BRIEF OVERVIEW OF COMPUTER VISION

Computer Vision is a scientific field that deals with how computers can gain high-level understanding from digital images or videos. In the last decades, impressive milestones have been reached in various fields: in medicine, segmentation of brain tumor has high clinical relevance for the estimation of the volume and spread of a tumor and skeleton segmentation techniques have been able to provide a fast and reliable 3D observation of fractured bones; in the security industry, CV techniques such as real-time face recognition or object detection, combined with biometry are able to provide an easier control over entire buildings.

"However, despite all of these advances, the dream of having a computer interpret an image at the same level as a two-year old remains elusive."[1]. Describing the world and reconstructing its properties, such as light, color and shape, is an effortless task for humans and even animals, yet an "intelligent" machine must resort to physics-based and probabilistic models to disambiguate between different possible solutions; why is this so difficult? The problem is based both on the still limited understanding of biological vision and on the complexity of vision perception in a dynamic and nearly infinite varying physical world.

…

# 2. IMAGE RECOGNITION USING CONVOLUTIONAL NEURAL NETWORKS

Image Recognition is a computer vision technique that allows machines to interpret and categorize what they "see" in images or videos. Recognizing image patterns and extracting features is often the initial step of more complex computer vision techniques, like object detection or image segmentation. There are, however, various standalone applications that make the technique an essential machine learning task and the employment of neural networks has become the state-of-the-art approach for image recognition.

A model is trained to take an image as the input and output a target class, a set of labels describing the image. Along with a predicted class, a model may also output a confidence score related to how certain it is that an image belongs to a class. The technique can be broken into two separate branches: single and multiclass recognition; in single class image recognition, a model, or binary classifier, predicts only one label per image. On the other hand, multiclass models can assign several labels to a single image, outputting a confidence score for each one.

Nearly all image recognition models begin with an encoder, which is made up of blocks of layers that learn statistical patterns in the pixels of images that correspond to the labels they are attempting to predict. The encoder is then typically connected to a fully connected or dense later, that outputs confidence scores for each possible label.

[1] R. Szeliski, Computer Vision: Algorithms and Applications, Springer, 2010

## 2.1 DEEP LEARNING AND CONVOLUTIONAL NEURAL NETWORKS

Generally speaking, an Artificial Neural Network is an algorithm designed to recognize patterns in data and group them together; it is based on a collection of connected units or nodes, called artificial neurons, which can receive signals, process them and then signal the other neurons connected to them through connections called edges. Each neuron and edge can have an adjustable weight, which increases or decreases the strength of the signal received; they may also have a threshold such that a signal is sent only if the aggregate signal crosses the threshold.
The original goal of the ANN approach was to solve problems in the same way that a human brain would, however over time the focus shifted to performing specific tasks in various fields, such as computer vision, speech recognition, machine translation or medical diagnosis.

Formally, “A neural network is a sorted triple *(N, V, w)* with two sets *N*, *V* and a function *w*, where *N* is the set of neurons and *V* a set $\{(i, j) \mid i, j \in N\}$ whose elements are called connections between neuron i and neuron j. The function $w: V \rightarrow R$ defines the weights, where $w((i, j))$, the weight of the connection between neuron i and neuron j, is shortened to $w_{i,j}$. Depending on the point of view it is either undefined or 0 for connections that do not exist in the network.”[2]

Usually neurons are grouped into layers, each one performing a different transformation on their input; signals travel from the first layer (input layer) to the last (output layer), with the possibility of traversing the layers multiple times, depending on the classification of the network. In a feedforward network, the signal can only travel forward and after a transformation is performed, the new values become the input values of the next layer; they are often used in data mining problems. A feedback network, on the other hand, has feedback paths, which allow the signals to travel in both directions using loops; they are often used in optimization problems, where the network looks for the best arrangement of interconnected factors.

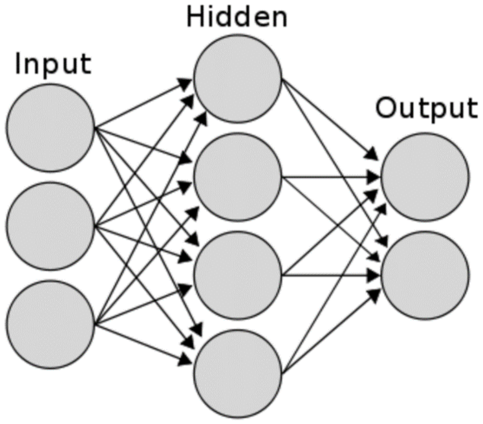


*Figure 1 A typical two-layer neural network. Input layer does not count as the number of layers of a network*

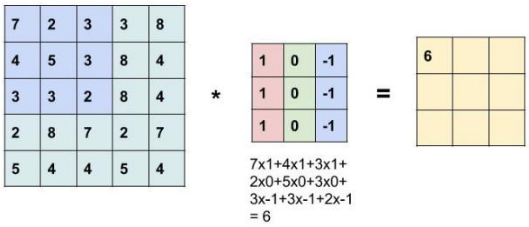
What happens inside a neuron? The input node receives a number representing the information, which is presented as an activation value, where each node is given a number; higher the number, the greater the activation. Based on the connection strength and transfer function, the activation value passes to the next node. Each of the nodes sums the activation values that it receives, by calculating a weighted sum, and modifies that sum based on its transfer function. Then an activation function is applied, used by the neuron to understand if it needs to pass along a signal or not. The activation runs through the network until it reaches the output nodes, which return the information

[2] D. Kriesel, A Brief Introduction to Neural Networks, 2005

in a way a human interpreter can understand. The network will then use a cost function to compare the output with the expected result
(cfr. A. Bonner, The Guide to Deep Learning: Artificial Neural Networks).

Deep Learning architectures suitable for image recognition are based on variations of Convolutional Neural Networks (CNNs). A CNN is a Deep Learning algorithm which can take an input image, assign importance to various aspects/objects in it (learnable weights and biases) and be able to differentiate one from the other. While in primitive methods filters are hand-engineered, with enough training, CNNs have the ability to learn these filters/characteristics.
The architecture of such a network is analogous to the connectivity patterns of human neurons and was inspired by the organization of the visual cortex, where individual neurons respond to stimuli only in a restricted region of the visual field, known as receptive field; a collection of such fields overlap to cover the entire visual area.

The first part of the CNN consists of convolutional and max-pooling features extractor layers, while the second part consists of the fully connected layer which performs non-linear transformations of the extracted features and acts as the classifier. If the neurons in the convolutional layer find the features they are looking for they produce a high activation.
In image processing, to calculate convolution at a particular location (x, y), a k x k sized chunk, the kernel, is extracted from the image, centred at location (x ,y); the values in this chunk are then multiplied element-by-element with the convolution filter, also sized k x k, and then they are added together to obtain a single output.



$$\begin{array}{|c|c|c|c|c|}\hline 7 & 2 & 3 & 3 & 8 \\ \hline 4 & 5 & 3 & 8 & 4 \\ \hline 3 & 3 & 2 & 8 & 4 \\ \hline 2 & 8 & 7 & 2 & 7 \\ \hline 5 & 4 & 4 & 5 & 4 \\ \hline \end{array} * \begin{array}{|c|c|c|}\hline 1 & 0 & -1 \\ \hline 1 & 0 & -1 \\ \hline 1 & 0 & -1 \\ \hline \end{array} = \begin{array}{|c|c|c|}\hline 6 & & \\ \hline & & \\ \hline & & \\ \hline \end{array}$$

7x1+4x1+3x1+
2x0+5x0+3x0+
3x-1+3x-1+2x-1
= 6

*Figure 2 Convolution operation example (V. Gupta, Image Classification using Convolutional Neural Networks in Keras)*

Storing an image means keeping track of the color information associated to each individual pixel in a color matrix; the size of each pixel depends on the color depth (8-16-24 bit). Once images reach a notable dimension, calculations can get very intensive, so the role of the CNN is to reduce the images into a form easier to process, without losing features which are critical for prediction sake.
This is achieved by a max pooling layer, which is responsible of reducing the spatial size of the image (not the depth); this reduces the number of parameters, avoiding overfitting, the condition when a trained model learns too much out of the training data and loses the ability to generalize.
A common form of pooling is max pooling where a filter of size p is taken and the maximum operation over the sized part of the image is applied.

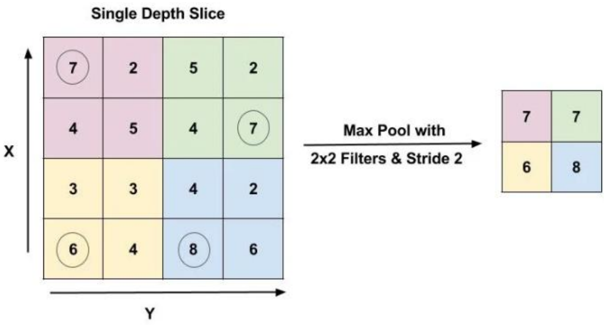


*Figure 3 Max pooling example (V., Image Classification using Convolutional Neural Networks in Keras)*

The fully connected layer is made up of an ANN, which purpose is to combine the detected features into more attributes, in order to predict the classes with greater accuracy.

Many neural network architectures[3] exist for image recognition, including:

- AlexNet: deep neural network winner of the ImageNet classification in 2012; it's widely credited with sparking a resurgence of interest in using deep convolutional neural networks to solve computer vision problems. The network is relatively large, with over 60 million parameters and many internal connections, thanks to dense layers that make the network quite slow to run in practice.
- VGGNet: network developed by researchers from the Visual Geometry Group (VGG) at Oxford University. VGGNet has more convolution blocks than AlexNet, making it "deeper", and it comes in 16- and 19-layer varieties, referred to as VGG16 and VGG19, respectively.

[3] Fritz AI, Image Recognition Guide, https://www.fritz.ai/image-recognition/

# 3. OPENCV

OpenCV is an open source computer vision and machine learning software library, written in C++, originally developed by Intel; it was later supported by Willow Garage then Itseez.
It supports a variety of deep learning frameworks, such as TensorFlow or PyTorch and offers over 2500 computer vision algorithms, ranging from classical statistical algorithms to modern machine learning-based techniques, including neural networks. These algorithms can be used to detect and recognize faces, identify objects, track camera movements or find similar images from an image database.

Since version 3.3, OpenCV is widely used to run CNNs and other neural network-based computer vision architectures; it is important to understand that OpenCV is not used to train the neural networks, but it can load a network, prepare and process images for it, apply it to the images and output the result.

Deep learning execution process:

- Load a model from disk.
- Pre-process images to serve as inputs to the neural network.
- (Run other computer vision algorithms on the input images if necessary)
- Pass the image through the network and obtain output classification.
- (Run other computer vision algorithms on the output if necessary)