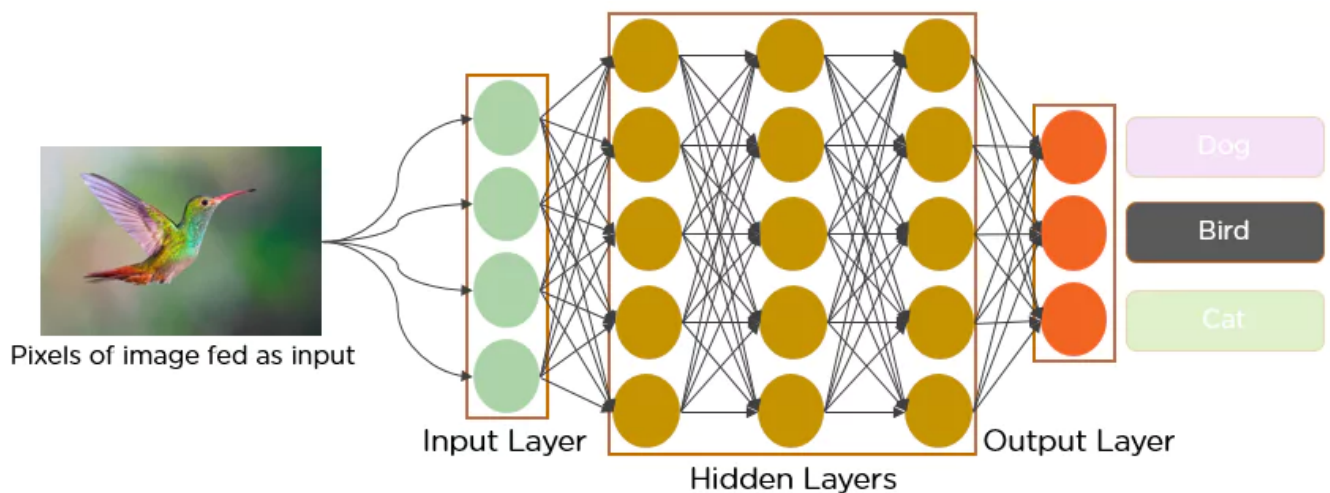


Introduction to Convolutional Neural Networks (CNN)

Introduction

Since the 1950s, AI researchers have sought to develop systems that can understand visual data. This effort gave birth to the field of Computer Vision. In 2012, a significant breakthrough occurred when researchers from the University of Toronto developed AlexNet, an AI model that significantly outperformed previous image recognition algorithms. AlexNet, created by Alex Krizhevsky, won the 2012 ImageNet contest with 85% accuracy, far surpassing the runner-up's 74%. This success was driven by CNNs, a type of neural network that mimics human vision.



Over the years, CNNs have become fundamental in computer vision tasks such as image classification, object detection, and segmentation. Modern CNNs are implemented using programming languages like Python and leverage advanced techniques to extract and learn features from images.

Hyperparameters, optimization techniques, and regularization methods are crucial for training these models effectively.

Since AlexNet, numerous improvements and new architectures like VGG, ResNet, and EfficientNet have been developed, pushing the boundaries of what CNNs can achieve. Today, CNNs are essential in many applications, from autonomous driving to medical image analysis.

In this article, we explore what is CNN, or convolutional neural network, a fundamental concept in deep learning. A convolutional neural network is a type of CNN model that employs the CNN algorithm to analyze data. This technique is integral to CNN ML and CNN machine learning, where it excels in image processing. Understanding CNN in machine learning and CNN in deep learning is crucial for leveraging its capabilities effectively.

Learning Objectives

- Learn what Convolutional Neural Networks (CNNs) are and how they mimic human vision to process visual data.
- Explore the key components of CNNs, including convolutional layers, pooling layers, and fully connected layers.
- Discover how CNNs are used in various computer vision tasks such as image classification, object detection, and segmentation.
- Understand the importance of hyperparameters, optimization techniques, and regularization methods in training CNN models

effectively.

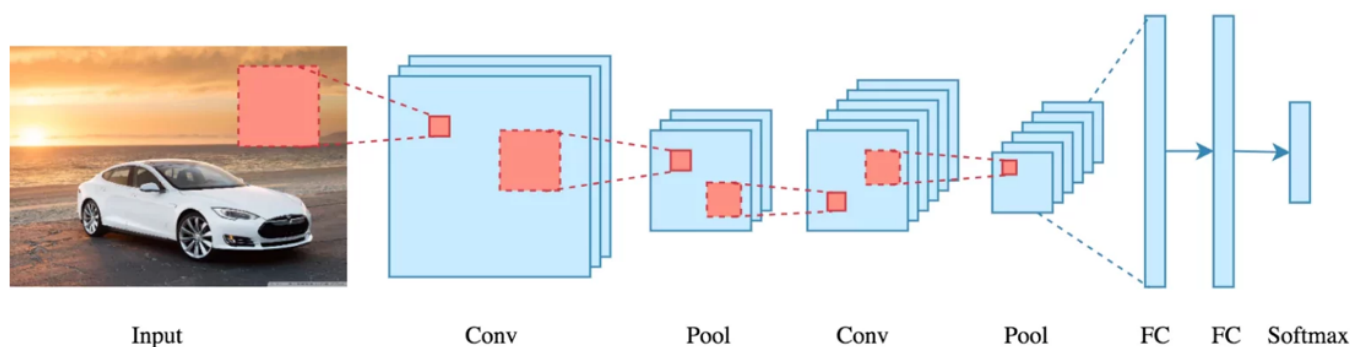
Table of contents



1. What are Convolutional Neural Network (CNN) ?
2. Artificial Neurons Role in CNN
3. Feature Extraction in CNN
4. Background of Convolutional neural networks (CNNs)
5. What is a Pooling Layer?
6. Limitations of Convolutional neural networks (CNNs)

What are Convolutional Neural Network (CNN) ?

In deep learning, a convolutional neural network (CNN/ConvNet) is a class of deep neural networks, most commonly applied to analyze visual imagery. The cnn architecture uses a special technique called Convolution instead of relying solely on matrix multiplications like traditional neural networks. Convolutional networks use a process called convolution, which combines two functions to show how one changes the shape of the other.



But we don't need to go behind the mathematics part to understand what a CNN is or how it works. The bottom line is that the role of the convolutional networks is to reduce the images into a form that is easier to process, without losing features that are critical for getting a good prediction.

The bottom line is that the role of the ConvNet is to reduce the images into a form that is easier to process, without losing features that are critical for getting a good prediction.

Background of Convolutional neural networks (CNNs)

CNNs were first developed and used around the 1980s. The most that a Convolutional Neural Network (CNN) could do at that time was recognize handwritten digits. It was mostly used in the postal sector to read zip codes, pin codes, etc. The important thing to remember about any deep learning model is that it requires a large amount of data to train and also requires a lot of computing resources. This was a major drawback for CNNs at that period, and hence CNNs were only limited to the postal sectors and it failed to enter the world of machine learning. Backpropagation, the algorithm used to train neural networks, was also computationally expensive at the time.

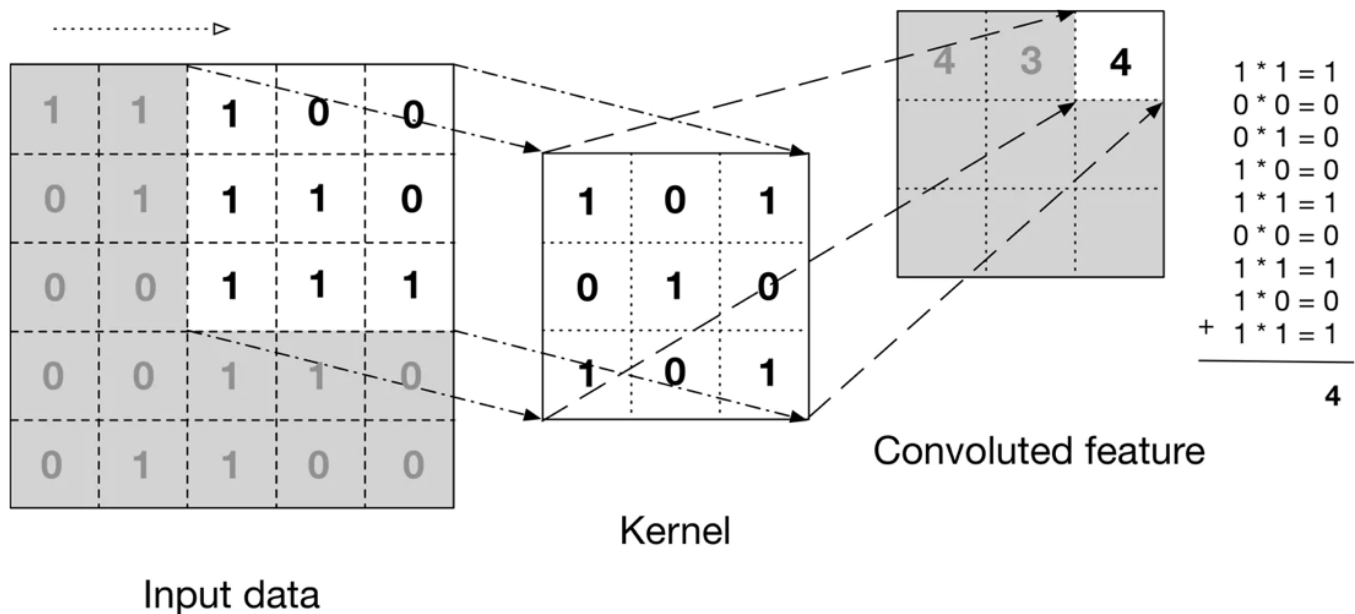
In 2012, Alex Krizhevsky realized that it was time to bring back the branch of deep learning that uses multi-layered neural networks. The availability of large sets of data, more specific ImageNet datasets with millions of labeled images, and an abundance of computing resources enabled researchers to revive CNNs.

How does CNN Work?

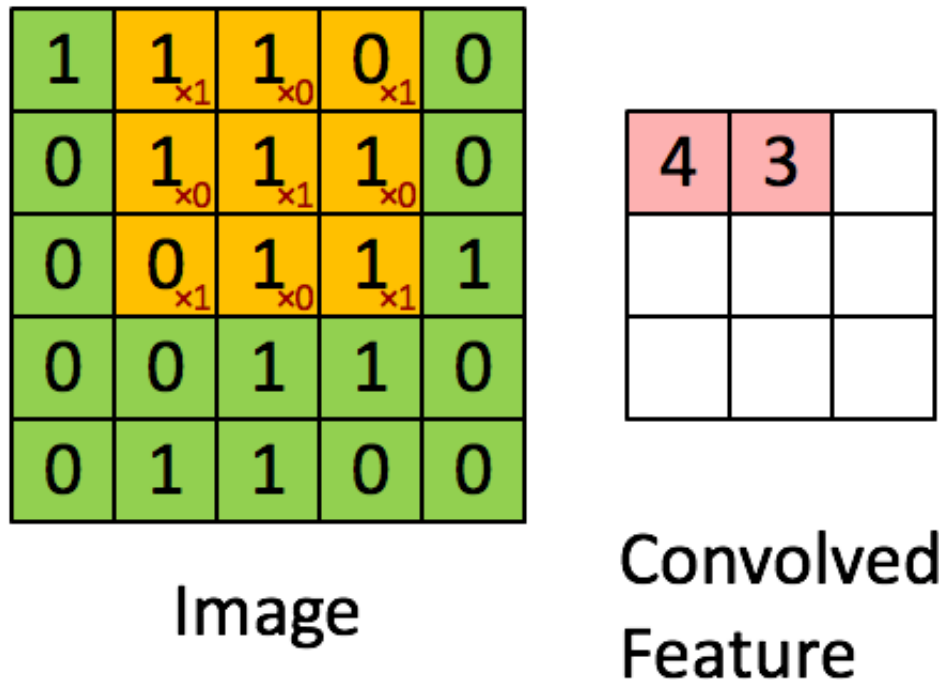
Before we go to the working of [Convolutional neural networks](#) (CNN), let's cover the basics, such as what an image is and how it is represented. An RGB image is nothing but a matrix of pixel values having three planes whereas a grayscale image is the same but it has a single plane. Take a look at this image to understand more.



For simplicity, let's stick with grayscale images as we try to understand how CNNs work.

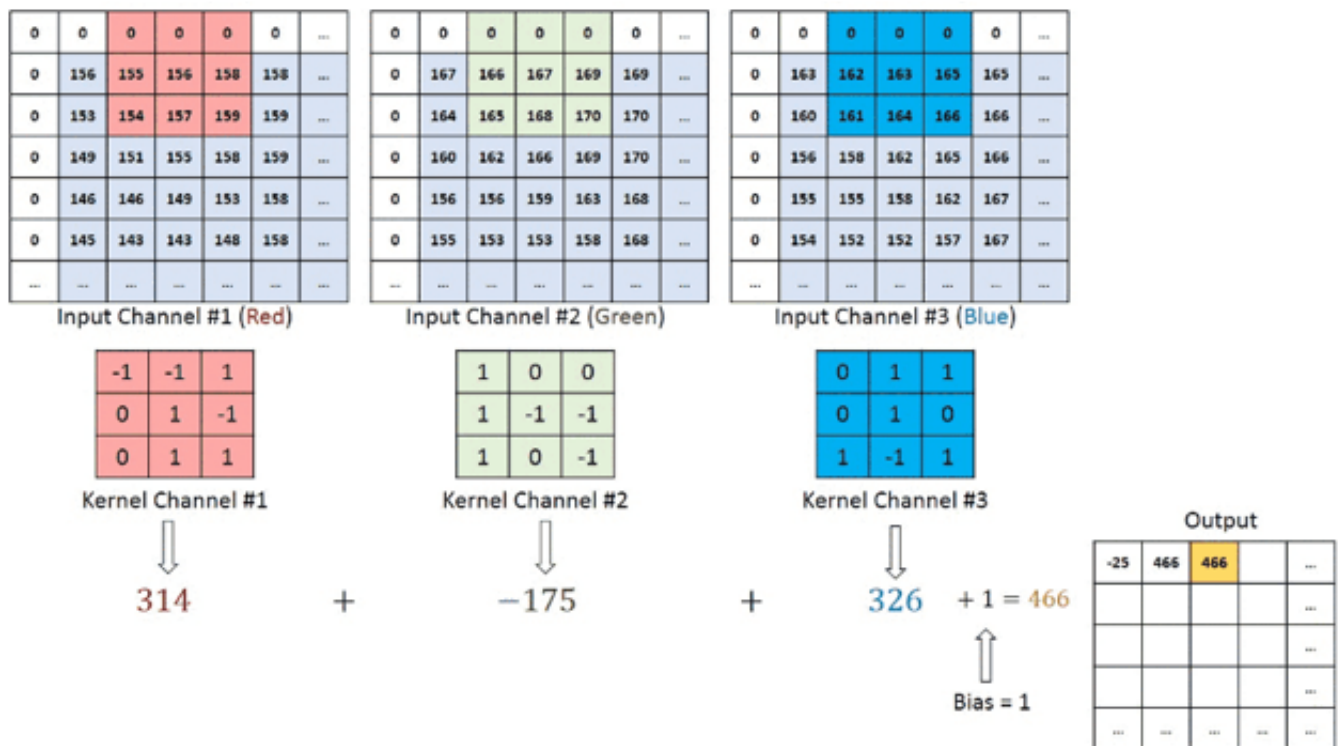


The above image shows what a convolution is. We take a filter/kernel(3×3 matrix) and apply it to the input image to get the convolved feature. This convolved feature is passed on to the next layer.



In the case of RGB color channel, take a look at this animation to

understand its working



The number of parameters in a CNN layer depends on the size of the receptive fields (filter kernels) and the number of filters. Each neuron in a CNN layer receives inputs from a local region of the previous layer, known as its receptive field. The receptive fields move over the input, calculating dot products and creating a convolved feature map as the output. Usually, this map then goes through a rectified linear unit (ReLU) activation function. Classic CNN architectures like LeNet and more modern ones like ResNet employ this fundamental principle.

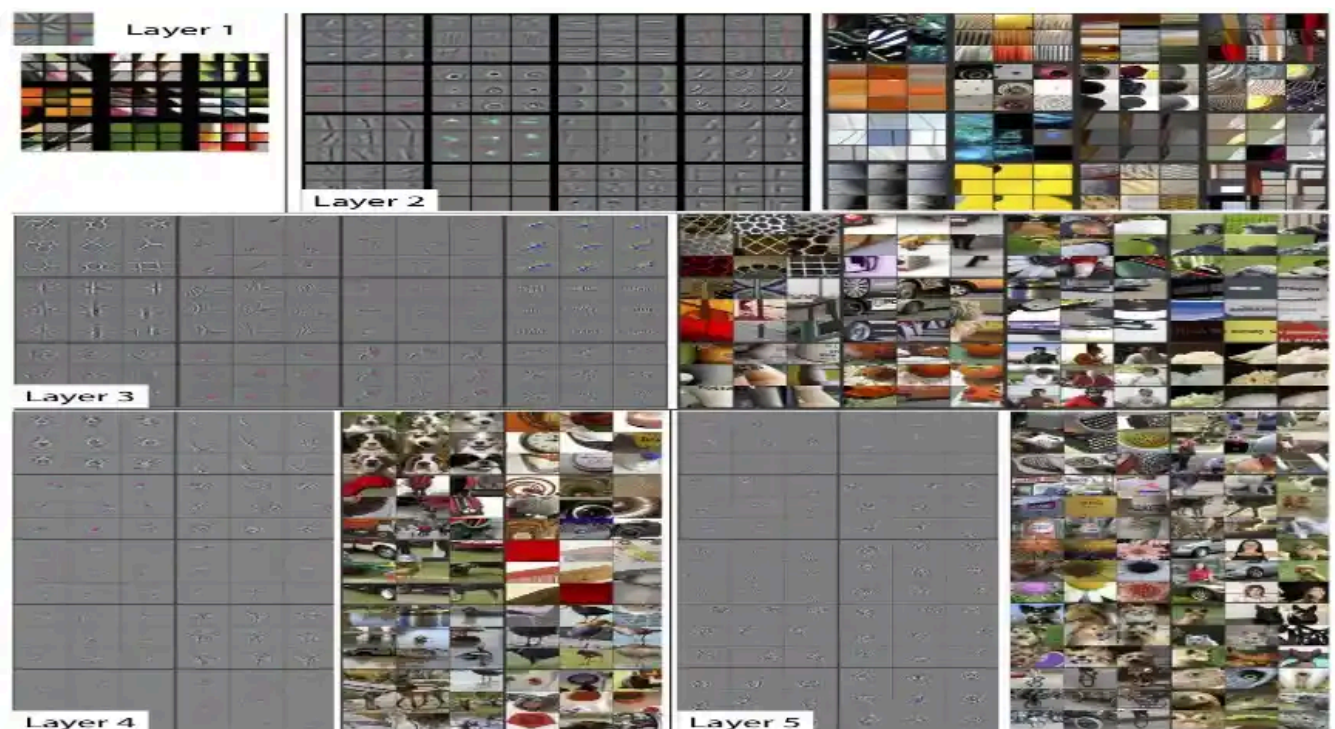
Convolutional neural networks are composed of multiple layers of artificial neurons.

Artificial Neurons Role in CNN

Artificial neurons, a rough imitation of their biological counterparts, are mathematical [functions](#) that calculate the weighted sum of multiple inputs and output an activation value. When you input an image in a ConvNet, each layer generates several activation functions that are passed on to the next layer for feature extraction.

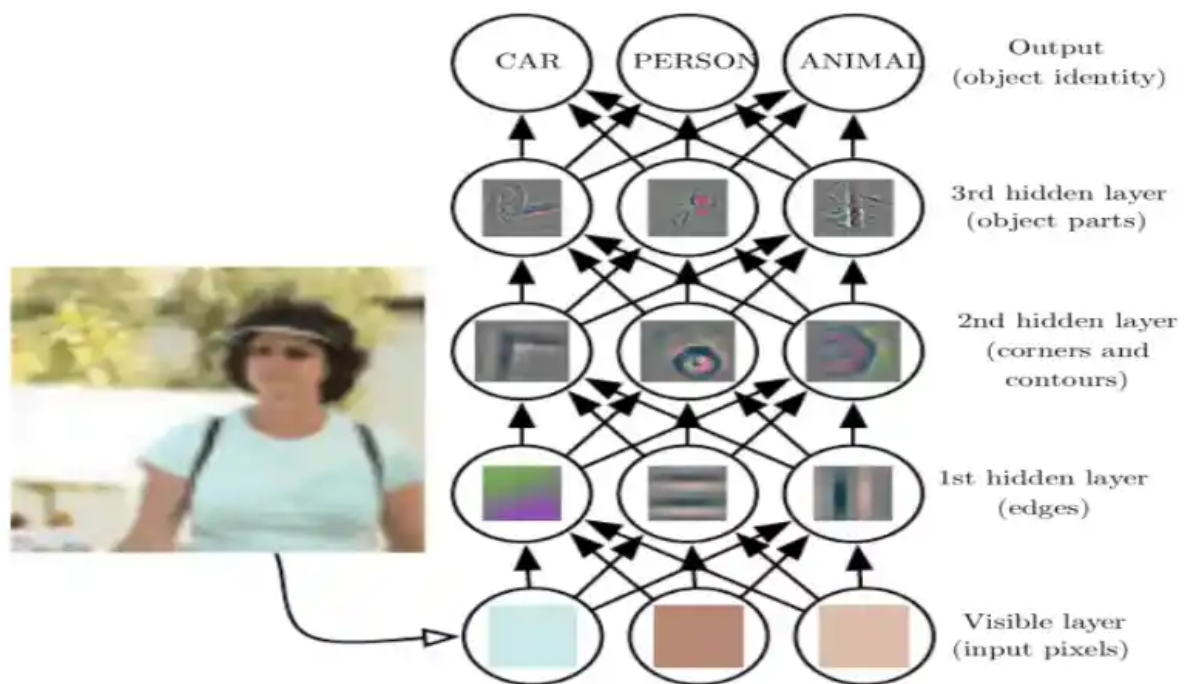
Feature Extraction in CNN

The first layer usually extracts basic features such as horizontal or diagonal edges. This output is passed on to the next layer which detects more complex features such as corners or combinational edges. As we move deeper into the network, it can identify even more complex features such as objects, faces, etc. Unlike recurrent neural networks, ConvNets are feed-forward networks that process the input data in a single pass.



Based on the activation map of the final [convolution](#) layer, the classification

layer outputs a set of confidence scores (values between 0 and 1) that specify how likely the image is to belong to a “class.” For instance, if you have a ConvNet that detects cats, dogs, and horses, the output of the final layer is the possibility that the input data contains any of those animals. Gradient descent is commonly used as the optimization algorithm during training to adjust the weights of the input layer and subsequent layers.



Background of Convolutional neural networks (CNNs)

CNNs were first developed and used around the 1980s. The most that a Convolutional Neural Network (CNN) could do at that time was recognize handwritten digits. It was mostly used in the postal sector to read zip codes, pin codes, etc. The important thing to remember about any deep learning model is that it requires a large amount of data to train and also requires a

lot of computing resources. This was a major drawback for CNNs at that period, and hence CNNs were only limited to the postal sectors and it failed to enter the world of machine learning. Backpropagation, the algorithm used to train neural networks, was also computationally expensive at the time.

In 2012, Alex Krizhevsky realized that it was time to bring back the branch of deep learning that uses multi-layered neural networks. The availability of large sets of data, more specific ImageNet datasets with millions of labeled images, and an abundance of computing resources enabled researchers to revive CNNs.

What is a Pooling Layer?

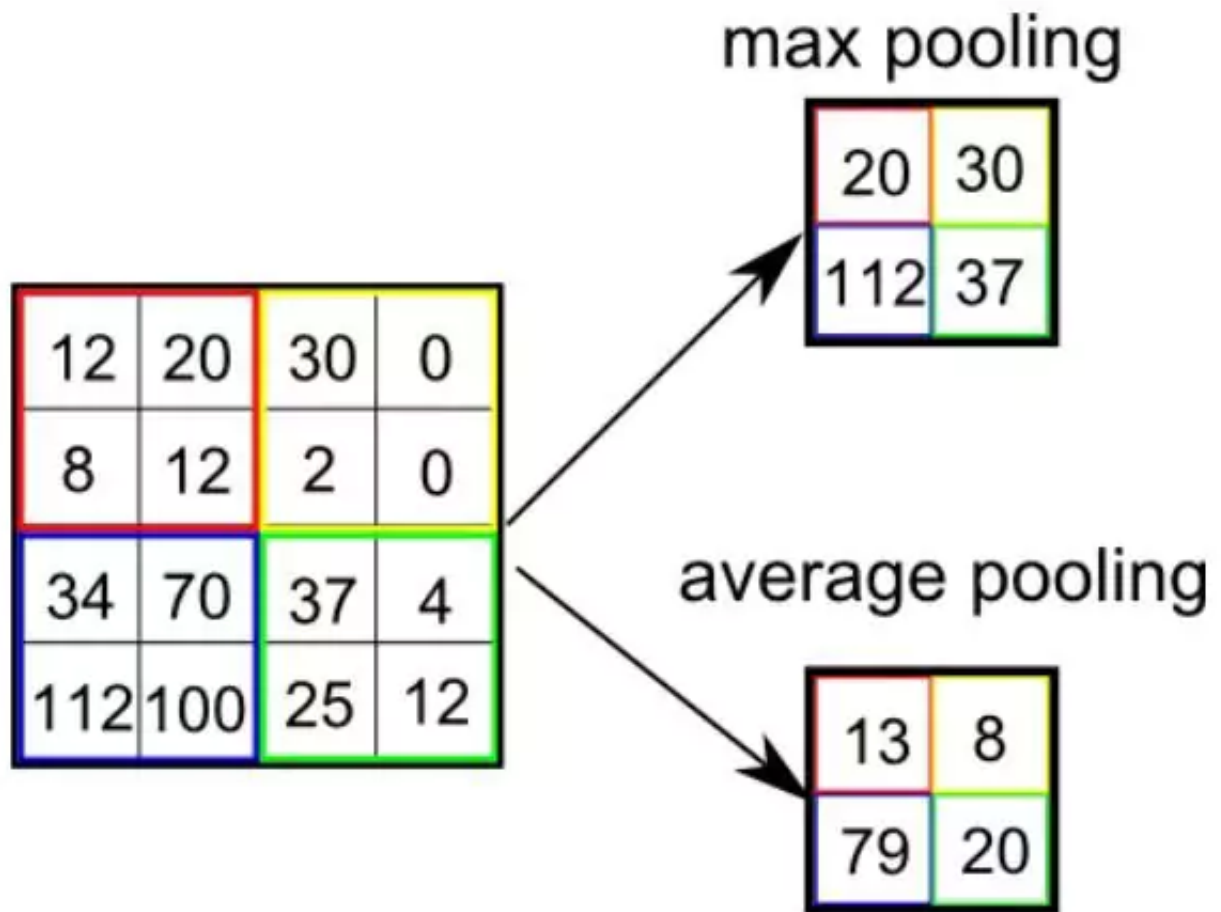
Similar to the [Convolutional](#) Layer, the Pooling layer is responsible for reducing the spatial size of the Convolved Feature. This is to **decrease the computational power required to process the data** by reducing the dimensions. There are two types of pooling average pooling and max pooling. I've only had experience with Max Pooling so far I haven't faced any difficulties.

3.0	3.0	3.0
3.0	3.0	3.0
3.0	2.0	3.0

3	3	2	1	0
0	0	1	3	1
3	1	2	2	3
2	0	0	2	2
2	0	0	0	1

So what we do in Max Pooling is we find the maximum value of a pixel from a portion of the image covered by the kernel. Max Pooling also performs as a **Noise Suppressant**. It discards the noisy activations altogether and also performs de-noising along with dimensionality reduction.

On the other hand, **Average Pooling** returns the **average of all the values** from the portion of the image covered by the Kernel. Average Pooling simply performs dimensionality reduction as a noise-suppressing mechanism. Hence, we can say that **Max Pooling performs a lot better than Average Pooling**.



Limitations of Convolutional neural networks (CNNs)

Despite the power and resource complexity of CNNs, they provide in-depth results. At the root of it all, it is just recognizing patterns and details that are so minute and inconspicuous that it goes unnoticed to the human eye. But when it comes to **understanding** the contents of an image it fails.

Let's take a look at this example. When we pass the below image to a CNN it detects a person in their mid-30s and a child probably around 10 years. But when we look at the same image we start thinking of multiple different scenarios. Maybe it's a father and son day out, a picnic or maybe they are camping. Maybe it is a school ground and the child scored a goal and his

dad is happy so he lifts him.



These limitations are more than evident when it comes to practical applications. For example, CNN's were widely used to moderate content on social media. But despite the vast resources of images and videos that they were trained on it still isn't able to completely block and remove inappropriate content. As it turns out it flagged [a 30,000-year statue with nudity](#) on Facebook.

Several studies have shown that CNNs trained on ImageNet and other popular datasets fail to detect objects when they see them under different lighting conditions and from new angles.

Does this mean that CNNs are useless? Despite the limits of convolutional

neural networks, however, there's no denying that they have caused a revolution in artificial intelligence. Today, CNN's are used in many [computer vision applications](#) such as facial recognition, image search, and editing, augmented reality, and more. As advances in ConvNets show, our achievements are remarkable and useful, but we are still very far from [replicating the key components of human intelligence](#).

Conclusion

In this article, we've explored Convolutional Neural [Networks](#) (CNNs), delving into their functionality, background, and the role of pooling layers. Despite their effectiveness in image recognition, CNNs also come with limitations, including susceptibility to adversarial attacks and high computational requirements. CNNs are trained using a loss function that measures the difference between the predicted output and the ground truth. Fine-tuning pre-trained models on specific image data is a common practice to achieve better performance.

Additionally, CNNs can be used for segmentation tasks, which involve labeling each pixel in an image. Unlike traditional multilayer perceptrons, the network architecture of CNNs is designed to take advantage of the spatial and temporal dependencies in image data. Overall, CNNs have revolutionized the field of computer vision and continue to be an active area of research. Despite their effectiveness in image recognition, CNNs also come with [limitations](#), including susceptibility to adversarial attacks and high computational requirements.