Fundamentals of Convolutional Neural Networks (CNNs)

Parth Agrawal

January 2026

1 Introduction to CNNs

CNNs are a type of neural network used to learn features in images or other similar data using kernels (filters). CNNs are used for image and video recognition, classification, and other applications.

CNNs are initially inspired by the working of the visual cortex. In 1968, Hubel and Wiesel demonstrated that there are simple visual cells that can recognize straight edges and complex cells that respond to the position of these edges. CNNs deploy a similar mechanism of using convolutions to extract features (hopefully) to locate edges, patterns, etc, to feed to neural networks. This is what distinguishes them from standard neural networks, which attempt to operate on all pixels.

A simple way to think about it is to consider how we recognize objects in a blurry image. We look for smaller edges or matching patterns, which we piece together to identify the object. CNNs adopt a similar strategy.

2 The Convolution Operation

The convolution operation occurs when a combination of two functions produces a third function as a result, where filters are applied across the input image to detect features.

The filters or kernels are small rectangular matrices that *slide* over the image from left to right and top to bottom. The convolution involved element-wise multiplication of this kernel with the input image, creating a feature map. We compute the output by summing up the multiplication given by the formula -

$$S(i,j) = (I * K)(i,j) = \sum_m \sum_n I(i+m, j+n) K(m,n)$$

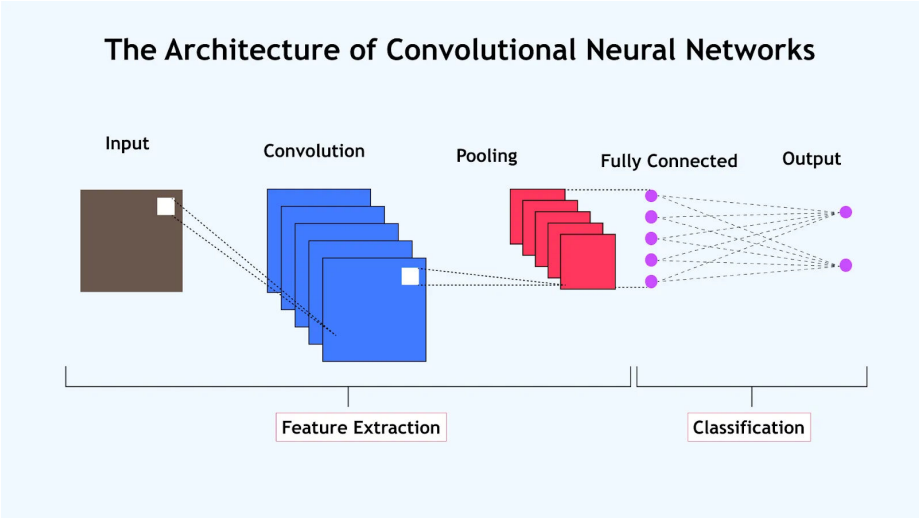

Figure 1: General CNN Architecture

Where m and n represent the no. of rows and columns.

By mapping each pixel in this manner, we slide through the entire image, detecting and refining the edges. It is precisely how to determine these edges, which can be achieved by varying the values of kernels, that our model learns to do.

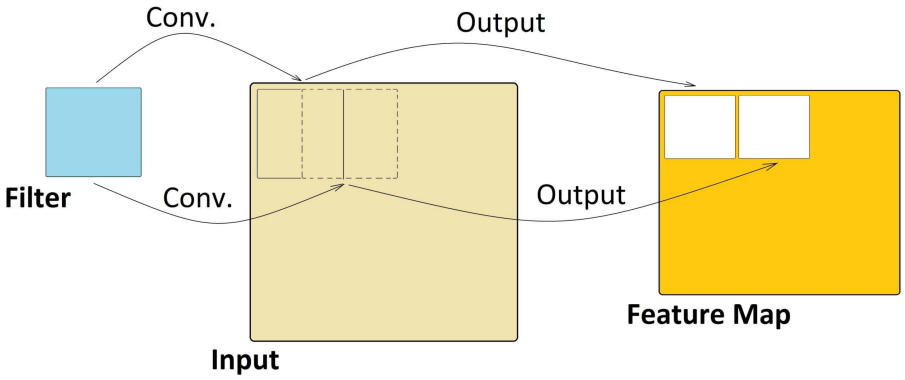

Figure 2: Filter and feature map

3 Filters (Kernels) and Feature Maps

Kernels are small matrices used for feature extraction. Values of kernels are learned through backpropagation—a kernel slides over the image, performing element-wise multiplication and summation of results.

The output of applying a kernel is called a feature map, which shows the specific features detected in the image. Multiple kernels are used at each layer of the CNN to extract complex features.

Some specific types of kernels include Edge detection kernels, such as the Sobel, Prewitt, and Laplacian kernels. We can also use kernels to sharpen or smooth the image features.

Each kernel specializes in detecting a specific feature, such as a particular edge or another characteristic.

4 Padding and Strides

Padding is a technique used to preserve the spatial dimensions of the input image after convolution operations, by adding extra pixels around the border of each pixel map before convolution.

Padding is helpful for preventing information loss at boundaries and corners. It is of two types -

- **Valid Padding:** No padding pixels are added to the input feature map, meaning the outer map is smaller. This is used in cases where we need to reduce the size of an image.

  $(n \times n) \cdot (f \times f) \longrightarrow (n-f+1) \times (n-f+1) (n \times n) \cdot (f \times f) \longrightarrow (n-f+1) \times (n-f+1)$

- **Same Padding:** Padding is added to make the input feature map and the output feature map of the same size. This is useful for protecting spatial features.

$$[(n+2p) \ddot{O} (n+2p) image] \cdot [(f \times f) filter] \longrightarrow [(n \times n) image]$$

Strides are the step size by which the filter moves during a convolution. It is the number of pixels long a step the kernel takes.

5 Pooling Layers (Downsampling)

Pooling Layers are used in CNN to reduce the spatial dimensions while retaining the most important information. It involves sliding a two-dimensional filter over the image and summarising the most important features.

Pooling is of three main types -

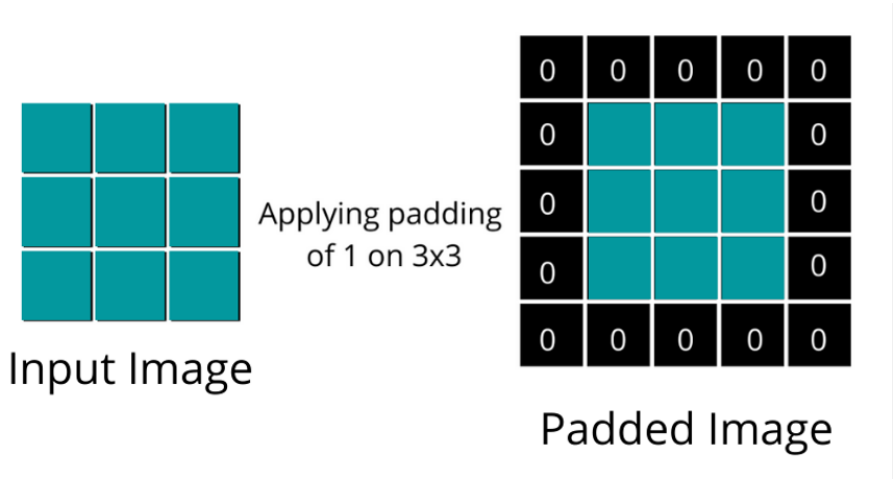

Figure 3: Padding (Valid vs Same)

- **Max Pooling:** It selects the maximum element in the region covered by the filter. It is used to preserve the most important features, such as edges and textures.
- **Average Pooling:** It is computing the average of elements present in the filter region. It is used to provide a generalized context.
- **Global Pooling:** It will reduce each channel in the feature map into a single value, which is maximum in the case of global max pooling and average in the case of global average pooling.

A common choice in pooling layers is a size of 2x2 with a stride of 2.

6 Convolution Over Volumes (3D Input)

So far, we have been referring to the conclusion based on a 2D image. However, in reality, colour images are composed of three colour channels (RGB). It is more like a 3d cube if we consider the value of each pixel. However, we can apply convolution to such a structure with ease, referred to as convolution over volumes.

The process is straightforward: use three (or the number of channels) filters, one for each channel. Then multiply the elements in the filter region by filter values, and add *all* of them up to get one pixel of the feature map. In other words, use three kernels and combine their results.

One thing that comes up is that having a single feature map often limits the potential extracted from the image; thus, what we usually do is use multiple groups of filters (such as two groups of three, one for horizontal edges and the

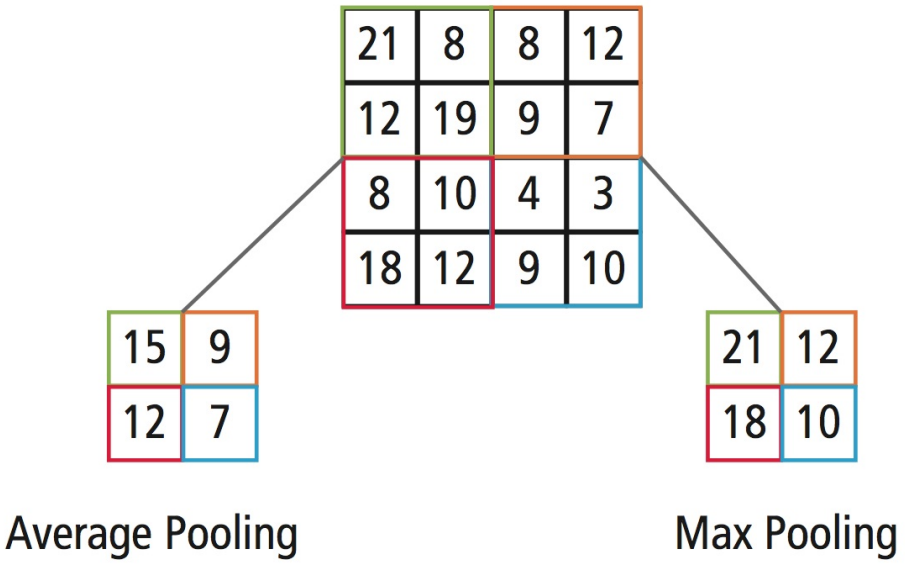

Figure 4: Max Pooling vs. Average Pooling

other for vertical) and obtain a 3D feature map. This is much more useful and practical.

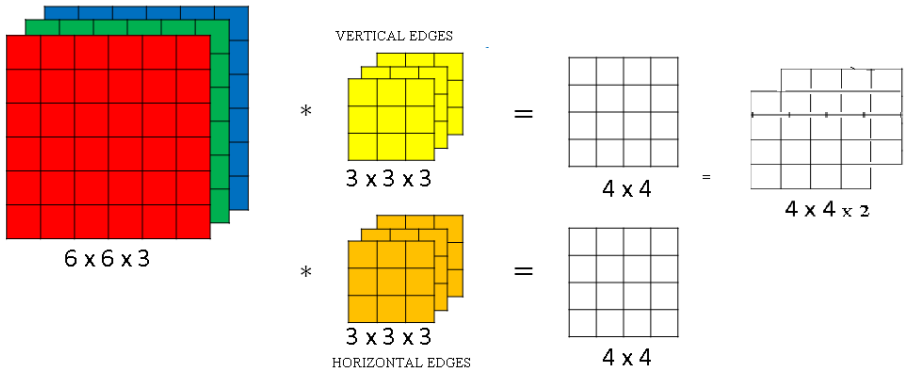

Figure 5: Convolution over volumes

7 References

- GeeksforGeeks. (2025, July 11). Introduction to convolution neural network. GeeksforGeeks. https://www.geeksforgeeks.org/machine-learning/introduction-convolution-neural-network/

- Kothiya, A. (2021, June 23). Understanding “convolution” operations in CNN. Medium. https://medium.com/analytics-vidhya/convolution-operations-in-convolution-neural-network-128906ece7d3/
- GeeksforGeeks. (2025, July 23). Kernels (filters) in convolutional neural network. GeeksforGeeks. https://www.geeksforgeeks.org/deep-learning/kernels-filters-in-convolutional-neural-network/
- GeeksforGeeks. (2023, December 13). CNN | Introduction to padding. GeeksforGeeks. https://www.geeksforgeeks.org/machine-learning/cnn-introduction-to-padding/
- GeeksforGeeks. (2025, December 3). CNN | Introduction to pooling layer. GeeksforGeeks. https://www.geeksforgeeks.org/deep-learning/cnn-introduction-to-pooling-layer/
- UPSC Fever. (n.d.). Convolutions over volume. UPSC Fever. https://upscfever.com/upsc-fever/en/data/deeplearning4/6.html