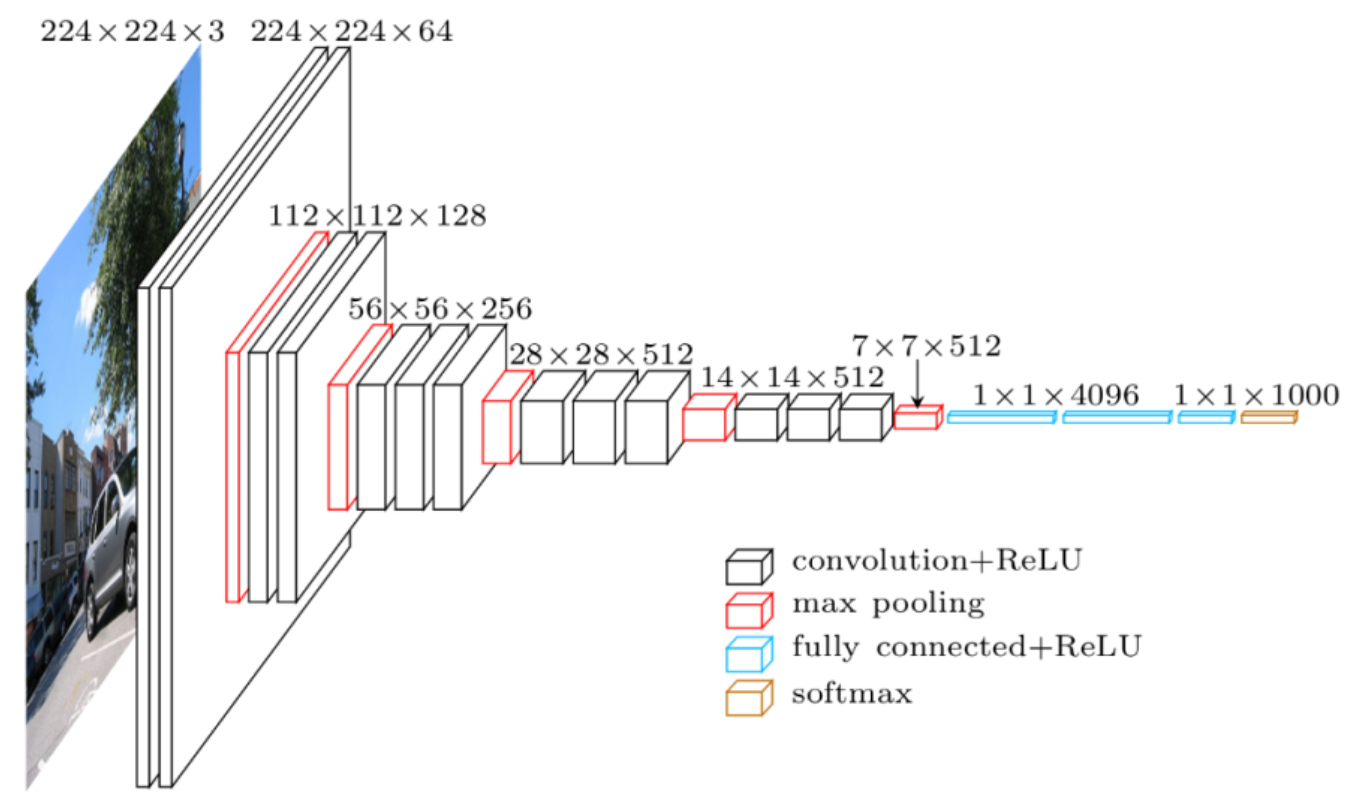
**Day 42**

**What to do?**

Learn about CNN and related terms, and why CNNs. Also learn about convolutional layer.

**CNN:**

When computer vision problems are talked about, convolutional neural networks (CNNs) are known to be heavily used to solve them. Examples of CNNs are facial recognition, image recognition, object detection and self-driving cars (but are not limited). CNNs are an advanced feature of neural networks, with multiple dimensions and filters. CNNs are computationally effective. It is the one algorithm that can handle hundreds of thousands of neurons.

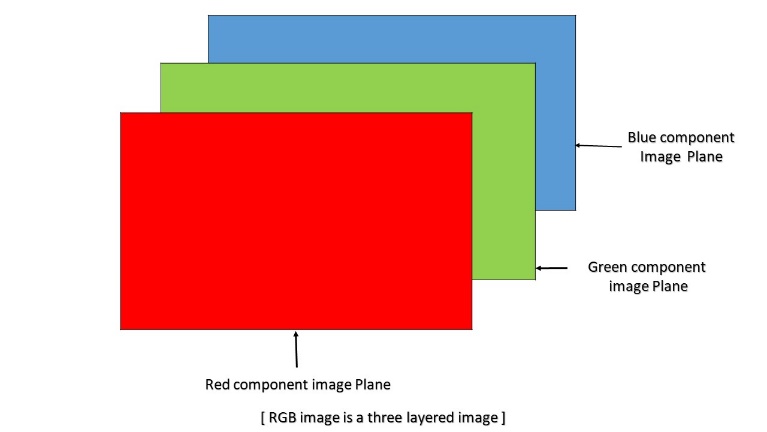


The input image above is of size (224, 224, 3): 3 because it is an RGB image. When you pass this input in a plain neural network, it would take the input size of 150,528 neurons, which is computationally super expensive. If the input is that huge, imagine how many other layers’ neurons would be based on the architecture above.

**Terms:**

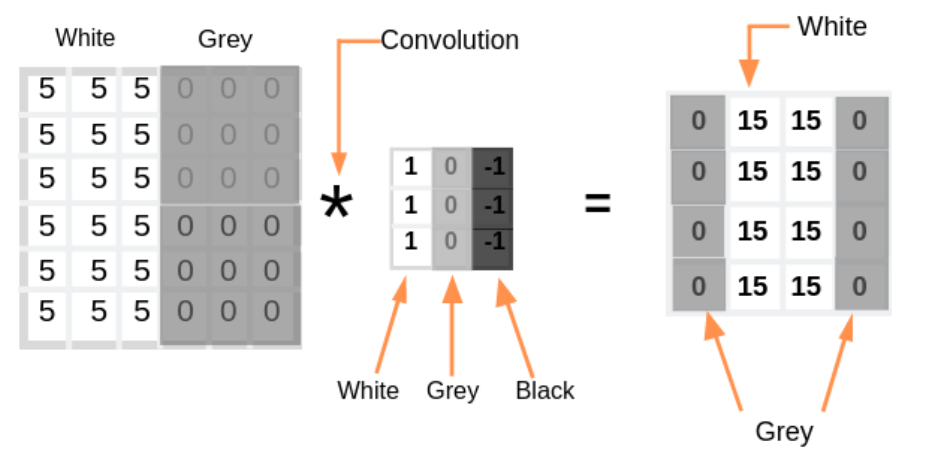
1. **Image representation:**

Usually, images have three dimensions, and this because it is a color image. Grayscale images only have width and height; however, color images have an additional dimension with value 3 (height X width X 3).



1. **Edge detection:**

All images have edges that combine to make an image. To detect edges (horizontal or vertical), input image is “convoluted” with a matrix “filter”. Convolution operator is nothing but summation of element – wise multiplication. Consider an image that’s white and gray, that is separated by a vertical edge. When the input is convoluted with a filter, it gives an output like below.

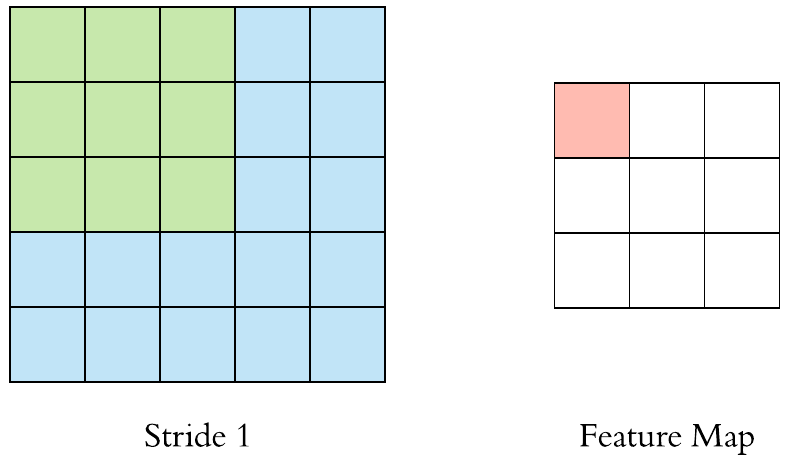


**\* =**

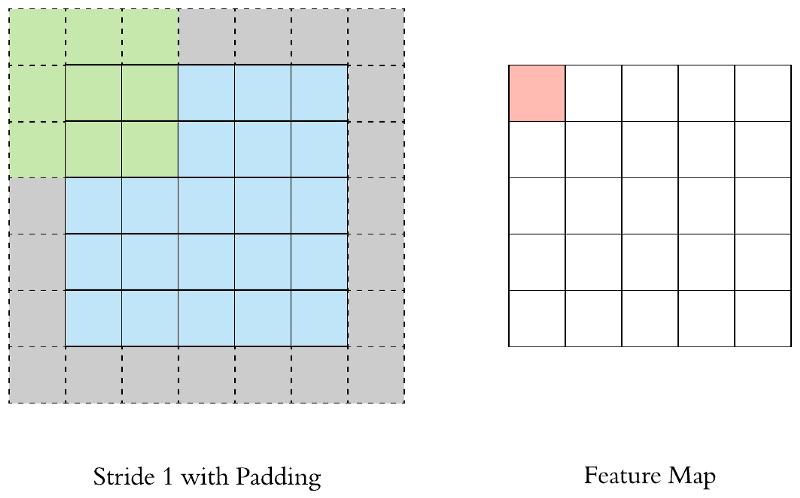
The output, as shown, has a huge white area which indicates the vertical edge that is separating the white and black area in the input. If we have an image of size N x N that is convoluted with a filter of size f x f, then the output image size is N-f+1 x N-f+1.

1. **Stride and padding:**

Stride denotes how many steps the filter should take during convolution. Default value is 1.



As stride increases, the output size decreases. That is the only disadvantage of CNN. As you add convolution computations, the output size decreases greatly. To ensure that the input size is same as output size, padding is used. Padding is the process of adding extra blocks to the image.



When you have an image of size N x N, of filter f x f with stride s and padding p, the output size is x . To make sure that input size is equal to output size,

= N

when s is not equal to 1

when s = 1

**Convolutional Layer:**

It is essential for any neural network to have weights and biases to carry the information. Even CNNs have weights and biases. The weights here and the filters, and the biases are added additionally. Once weights (filters) are multiplied (convoluted) with input and added with bias, they are passed into an activation function which results in an output. The process is repeated over and over to create a convolutional layer.

CNNs here, act as feature extractors, both simple and complex cells.