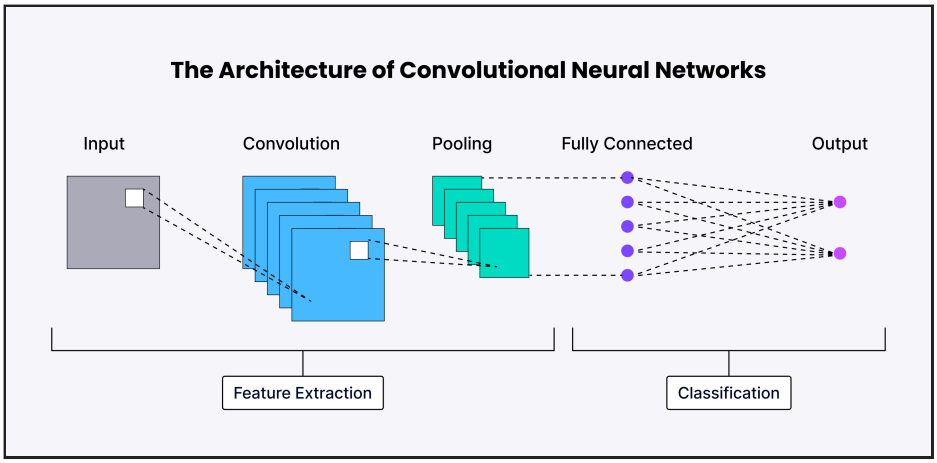
CONVOLUTIONAL NEURAL NETWORKS (CNN)

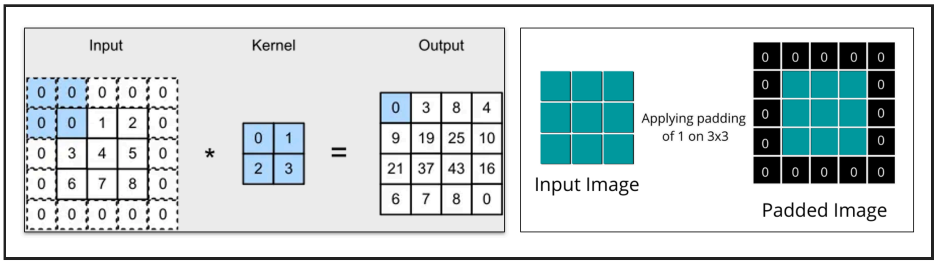
1. A Convolutional Neural Network (CNN) is a type of deep learning model mainly used to analyze visual data like images and videos.
2. It replicates how the human brain processes visual information by automatically learning patterns such as edges, shapes, and objects from raw pixels.
3. It uses convolutional layers to automatically extract features from data (especially images), followed by pooling and fully connected layers to classify or make predictions.
4. Instead of fully connecting all neurons, CNNs use filters (kernels) to scan across the input and extract important features.
5. CNN's require fewer parameters than traditional neural networks, making them faster and more efficient, especially for large images.
6. CNN's are widely used in real-time applications like facial recognition, self-driving cars, medical image analysis and security surveillance.



CORE COMPONENTS OF CONVOLUTIONAL NEURAL NETWORKS (CNN)

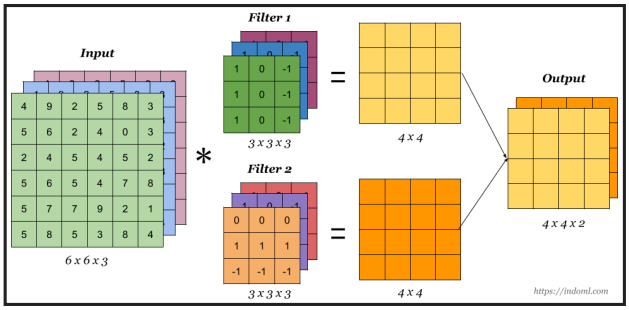
1. Image Kernel's

1. In Convolutional Neural Networks (CNNs), a Filter or Kernel is a small matrix used to detect specific features (like edges, corners, textures) in the input image.
2. In the context of CNN's, filters are referred as **Convolution Kernels**.
3. A filter (kernel) is usually a small grid or matrix (like 3×3, 5×5, etc.).
4. Kernels (or filters) are designed to transform an image into another form such as blurring, sharpening the image etc.
5. Kernel slides (or convolves) over the image and at each location, it performs element wise multiplication and sums the result to produce a feature map. This process is known as **Convolution.**
6. During convolution, we would lose borders. To address this issue we can the image with more values, know as **padding.**
7. For better understanding, visit - https://setosa.io/ev/image-kernels/#



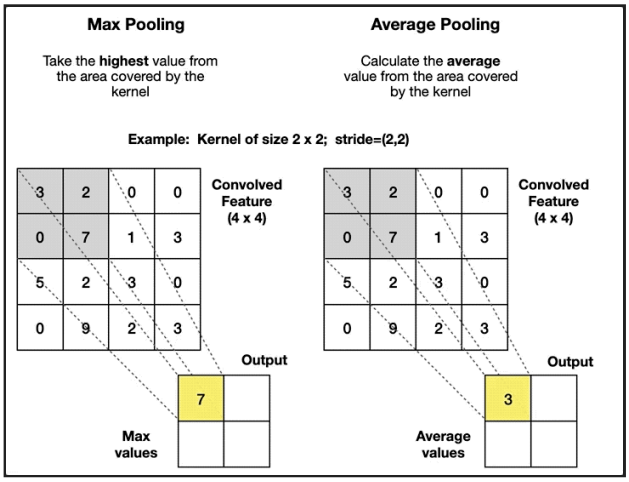
2. Convolutional Layers

1. A Convolutional Layer is a neural network layer that applies filters (kernels) to the input image to detect specific features such as edges, textures, shapes, etc.
2. A Convolutional layer is created when we apply multiple image filters to the input image.
3. Then, the layer will be trained to figure out the best filter weight values.
4. A CNN help in reducing the parameters by focusing in **Local Connectivity**.
5. Local Connectivity - each neuron in a convolutional layer is connected only to a small region of the input, not the whole image.



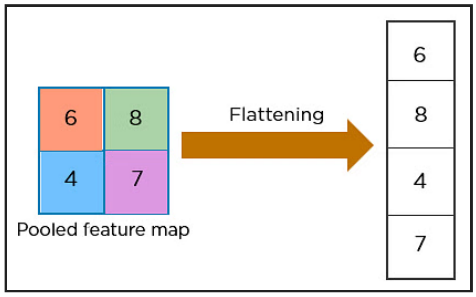
3. Pooling Layers

1. Pooling layers accept convolution layers as input.
2. A Pooling Layer reduces the spatial dimensions (width and height) of the feature maps while retaining the most important information.
3. Pooling layers will remove a lot of information in the input data, but in most classification tasks, it's helpful in filtering noise, improves efficiency, and boosts generalization.



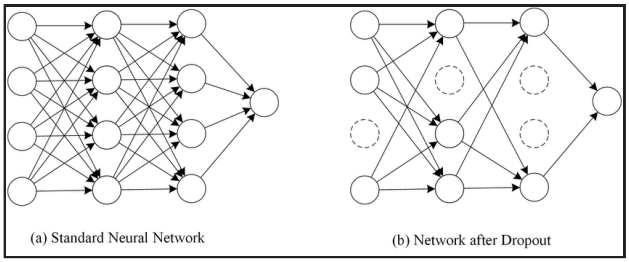
4. Faltten Layers

1. Flattening layers Converts the 2D (or 3D) feature maps from convolutional/pooling layers into a 1D vector.
2. The final Dense (Fully Connected) layers expect a 1D input, Flatten layers solves this issue.
3. These Flattening layers are placed after the last convolution/pooling layer and before the first Dense layer
4. *For Ex: If feature map shape is (8, 8, 32), flattening gives a vector of size 8×8×32 = 2048.*



5. Dropout Layers

1. Dropout is a regularization technique used in neural networks where, during training.
2. During each training step, neurons are randomly ignored (set to zero) with a certain probability 'p'.
3. *For Ex: Dropout(0.5) means 50% of neurons are dropped at random.*
4. It helps to prevent the overfitting



- By applying all these layers the CNN Architecture will be: