__/__/____

○ CONVOLUTIONAL NEURAL NETWORK (CNN)

Convolutional Neural Networks is an advanced version of ANNs, primarily designed to extract features from grid-like matrix datasets. They are the foundation for most modern Computer vision Applications to detect features within visual data.

→ Human Brain Vs CNN

1. Visual Cortex (V_1 - V_5): Region of the brain that receives, integrates and processes visual information relayed from retinas.

Visualise the image

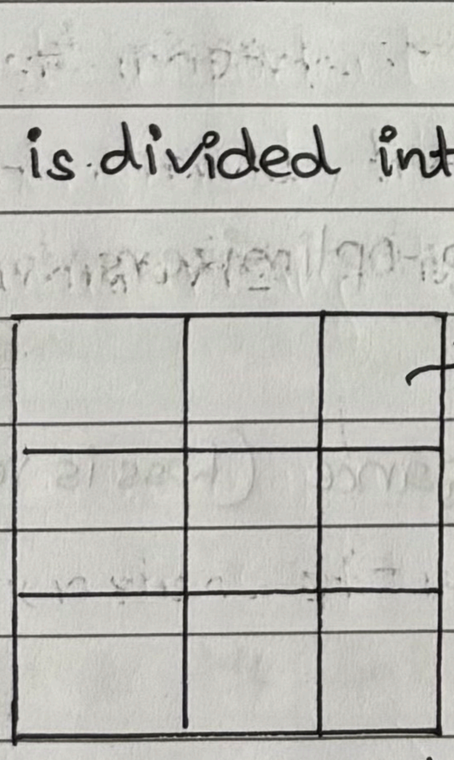
V_1 → Primary Visual Cortex (Orientation of edges and lines)

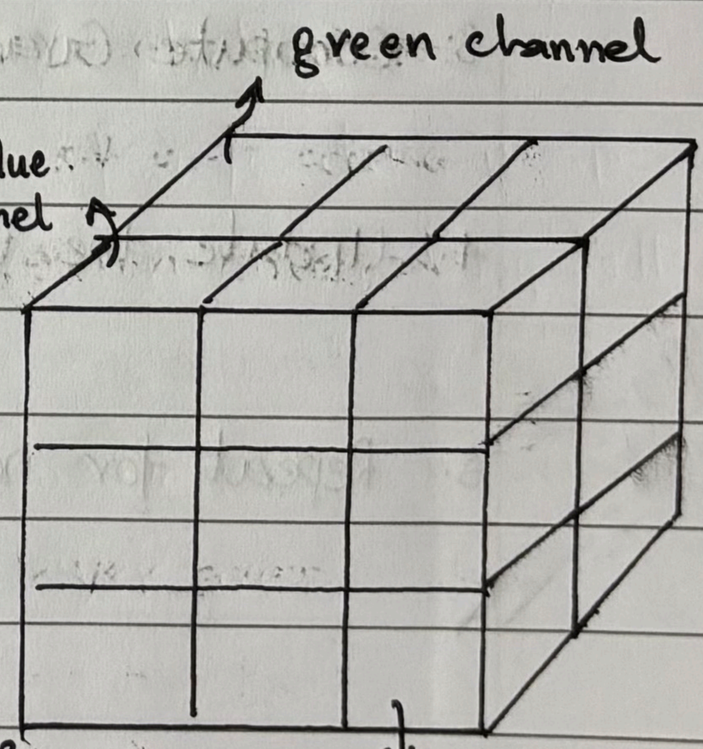
V_2 → Differences in colour, complex patterns, object orientation

V_3 V_4 V_5 → Object Recognition

→ Understanding about Images

Image is divided into pixels



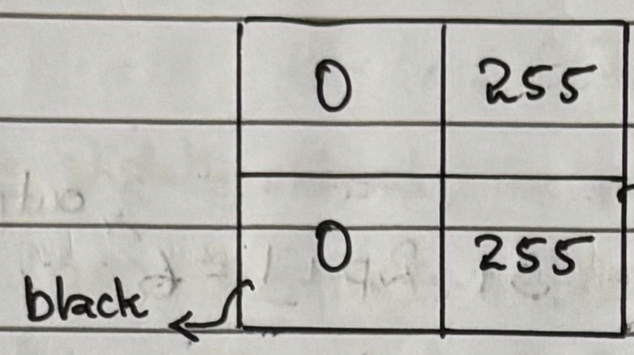


3 × 3 × 1 channel for grey scale image

3 × 3 × 3 channel for RGB image

__/__/____

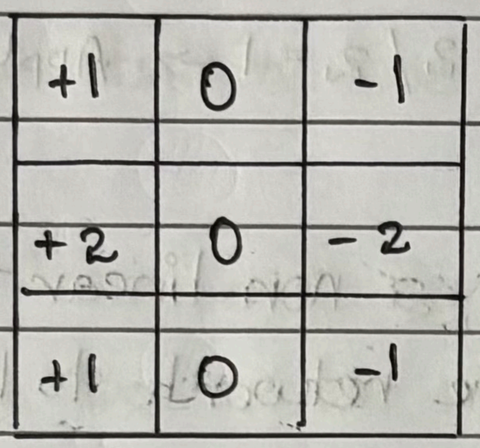
① Convolution Operation in CNN



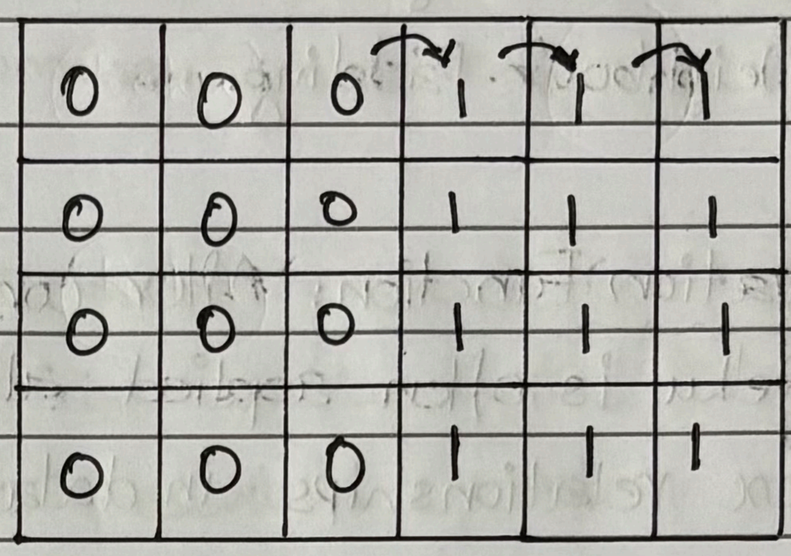
We need to normalize the values to a range of [0, 1], for that we divide each point with 255

filters

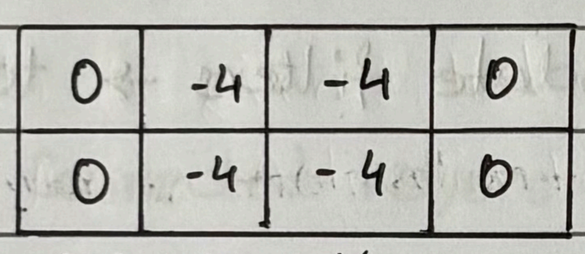
Inputs / normalized values



*



stride = 1



→ featuremap

↳ detects the vertical edges

Information loss is the

{Inputs − Filters + 1 → Outputs} ← Applying ⟶ I = 4, F = 3 → 4 − 3 + 1 = 2

Convolutional Layers apply convolution operation to the input data which involves a filter (or kernel) that slides over the input data, performing element-wise multiplication and summing the results to produce a feature map.

1. Filters (kernels): Small matrices that extract specific features from the input. The values of filters are learned and updated during training.

2. Stride: Refers to the step size with which the filter moves across the input data. Large stride results in smaller output feature maps and faster computation.

__/__/____

3. Padding : To prevent information loss in the output layer we're applying padding. (feature dimensions match input)

If n= 6, n - f + 2p + 1 ⇒

$6 - 3 + 2p + 1 = 6$ ← output

$3 + 2p + 1 = 6$

$2p = 6 - 4$

$P = 2/2 = 1$ → Apply padding of 1

Two types of padding:

1. Zero Padding
2. Neighbour Padding.

4. Activation Function: After convolution, a non-linear function like ReLu is often applied allowing the network to learn complex relationships in data.

Find the derivative of ReLu → Update filters → to get correct O/P.

② Pooling Layer

Pooling Layer is used in CNNs to reduce the spatial dimensions (width and height) of the input feature maps while retaining the most important information. It involves sliding a two dimensional filter over each channel of a feature map and summarizing the features within the region covered by filter.

For a feature map with dimensions $n_h \times n_w \times n_c$, the dimensions of the output after a pooling layer are:

$$\left(\frac{n_h - \text{filter} + 1}{\text{stride}}\right) \times \left(\frac{n_w - \text{filter} + 1}{\text{stride}}\right) \times n_{channel}$$

A typical CNN model architecture consists of multiple convolutional and pooling layers stacked together.

__/__/____

⇒ Importance : 1. Dimensionality Reduction

2. Overfitting Prevention

3. Translation Invariance → even if an object in an image is slightly shifted, the pooled output will remain relatively unchanged.

1. Max Pooling: Max pooling selects the maximum element from the region of the feature map covered by the filter.

The output after max-pooling layer would be a feature map containing the most prominent features of previous feature map.

2. Average Pooling : It computes the average of elements present in the region of feature map covered by the filter. Average pooling gives the average features present in a patch.

③ Fully Connected layers