

Convolution neural networks CNN, convnets

① Assumptions:

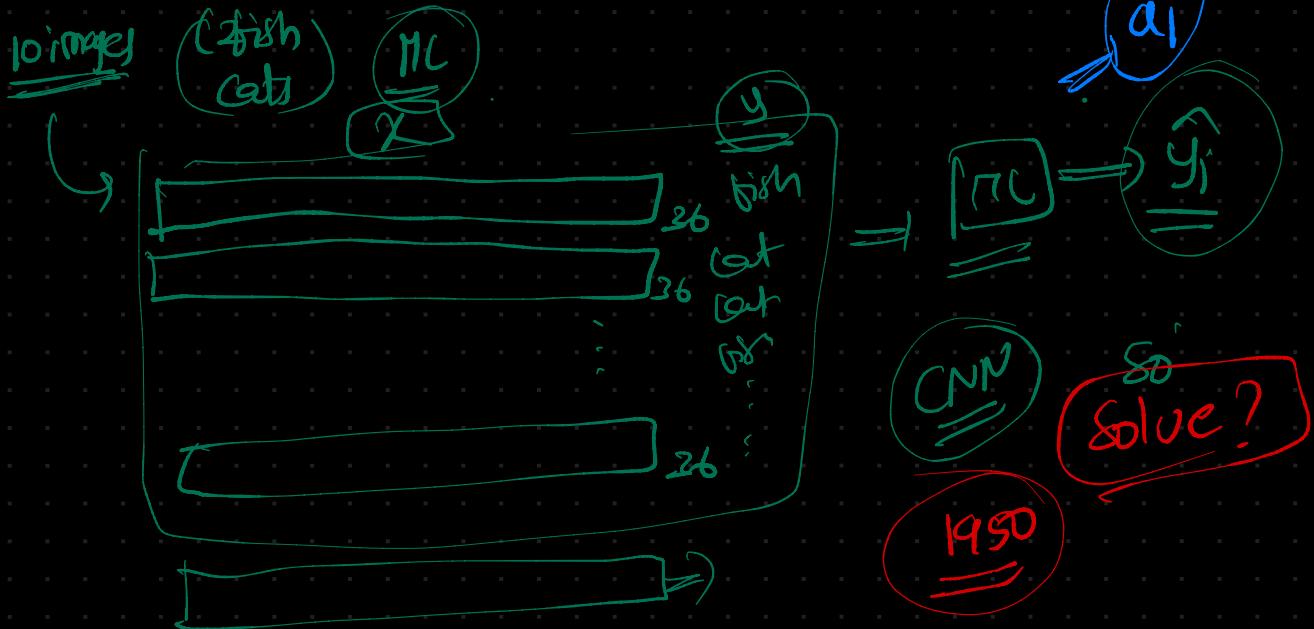
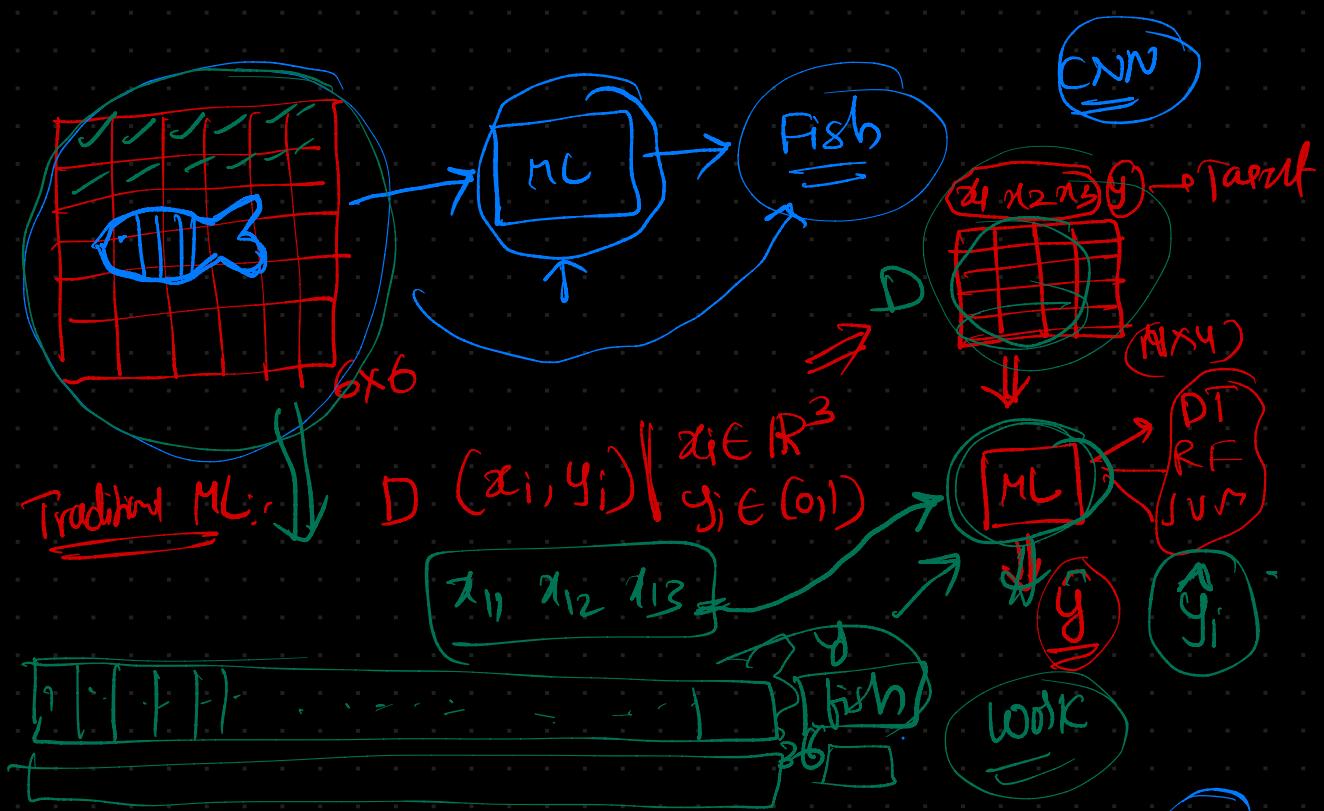
- └ ① Basic understanding of Python →
- └ ② Basics of ML & DL.
- └ ③ Matplotlib (for plotting).

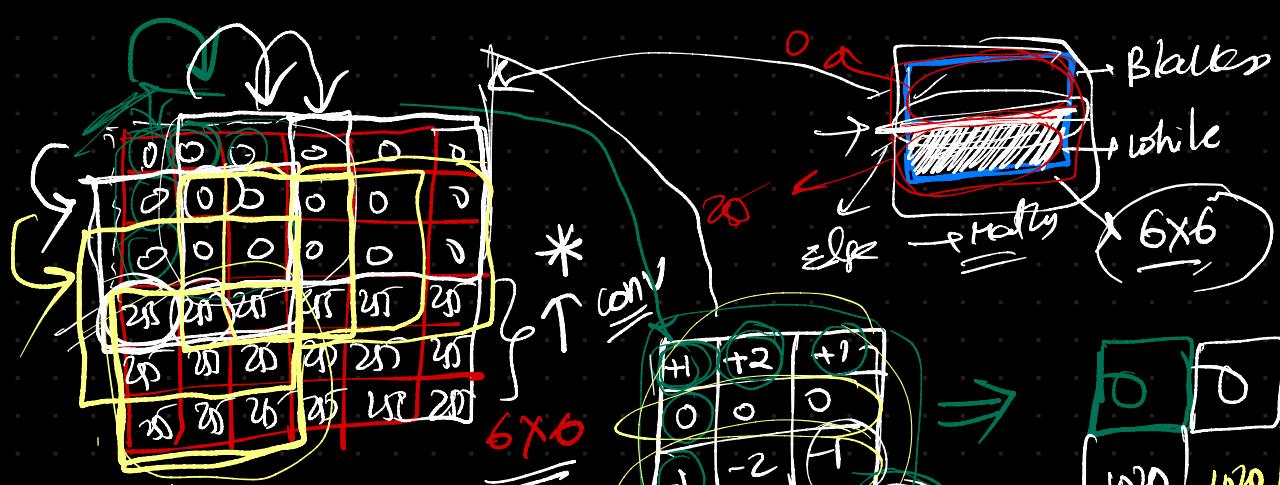
② Session to be covered

- ① complete theory & history of CNN (cats vs dogs)
- ② Coding of CNN → End to End Project
- ③ Transfer learning → Hyper Parameter tuning & Performance metrics.
- ④ Deployment

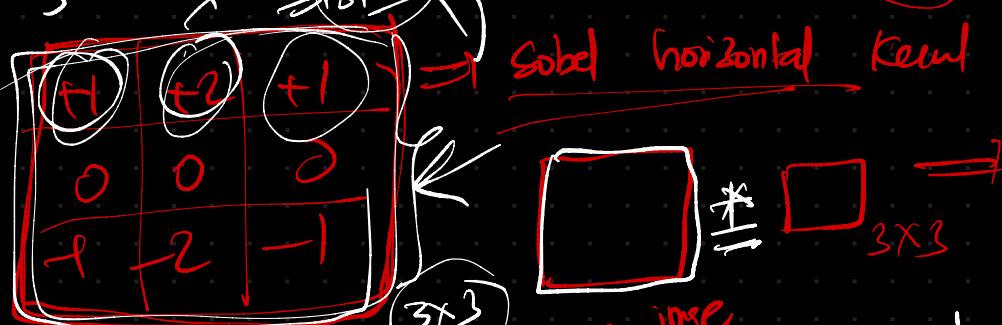
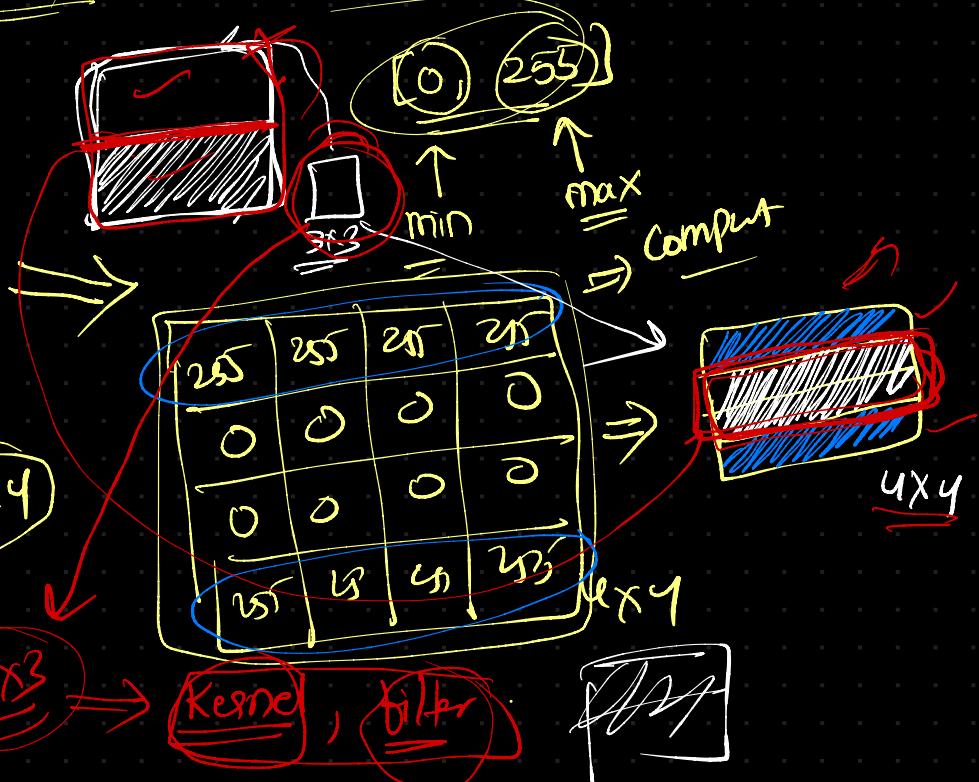
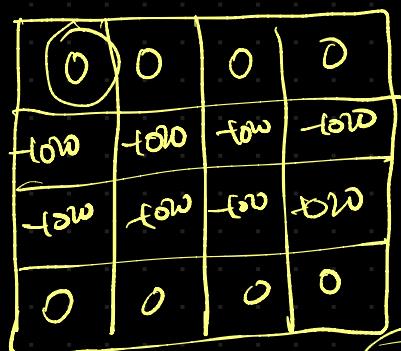
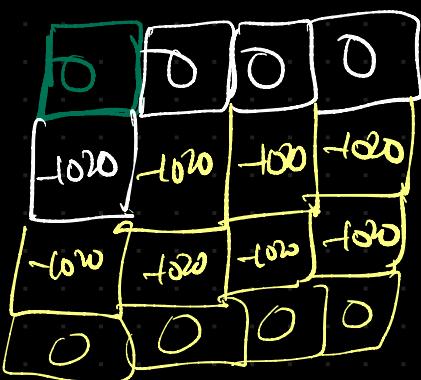
② Agenda:

- ① what are Images & Pixels → Comput
- ② Inspiration for Convnets. [History].
- ③ How edge detection works.
- ④ What is convolution
- ⑤ Padding, strides, Pooling.
- ⑥ End of RGB (Color) Images.
- ⑦ How to build CNN network for Image classification.



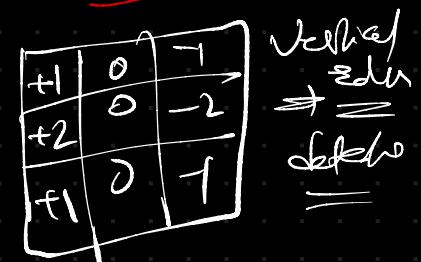


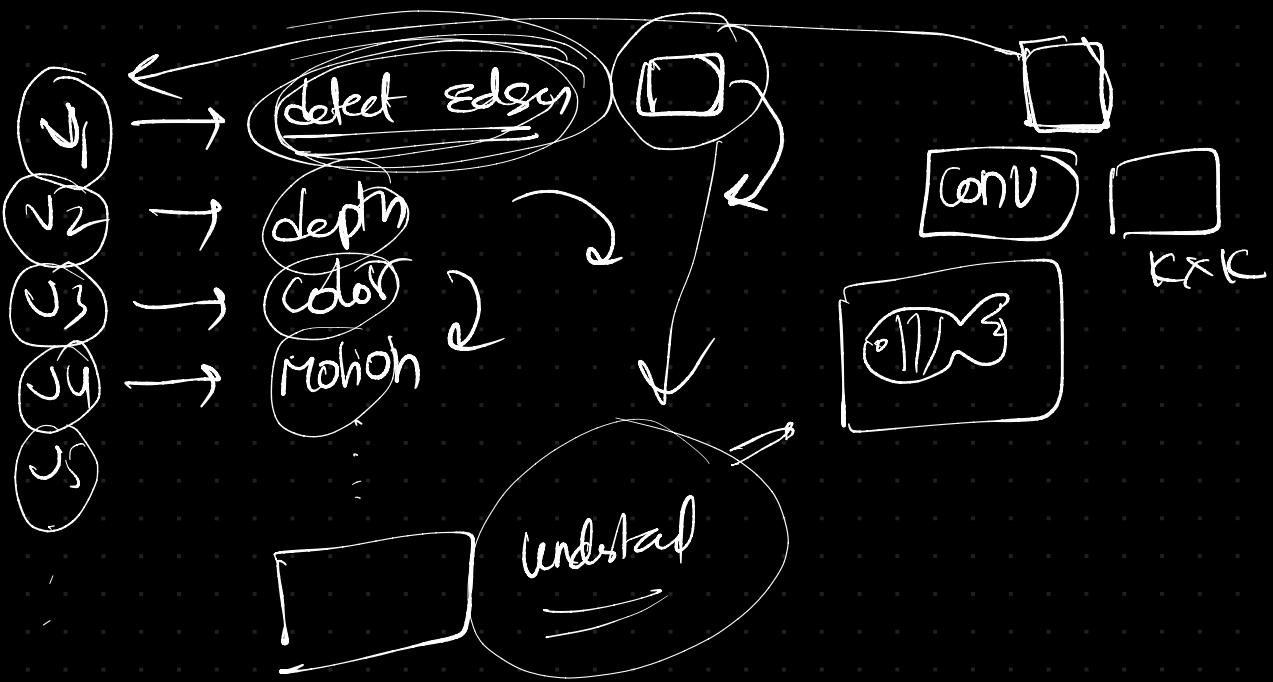
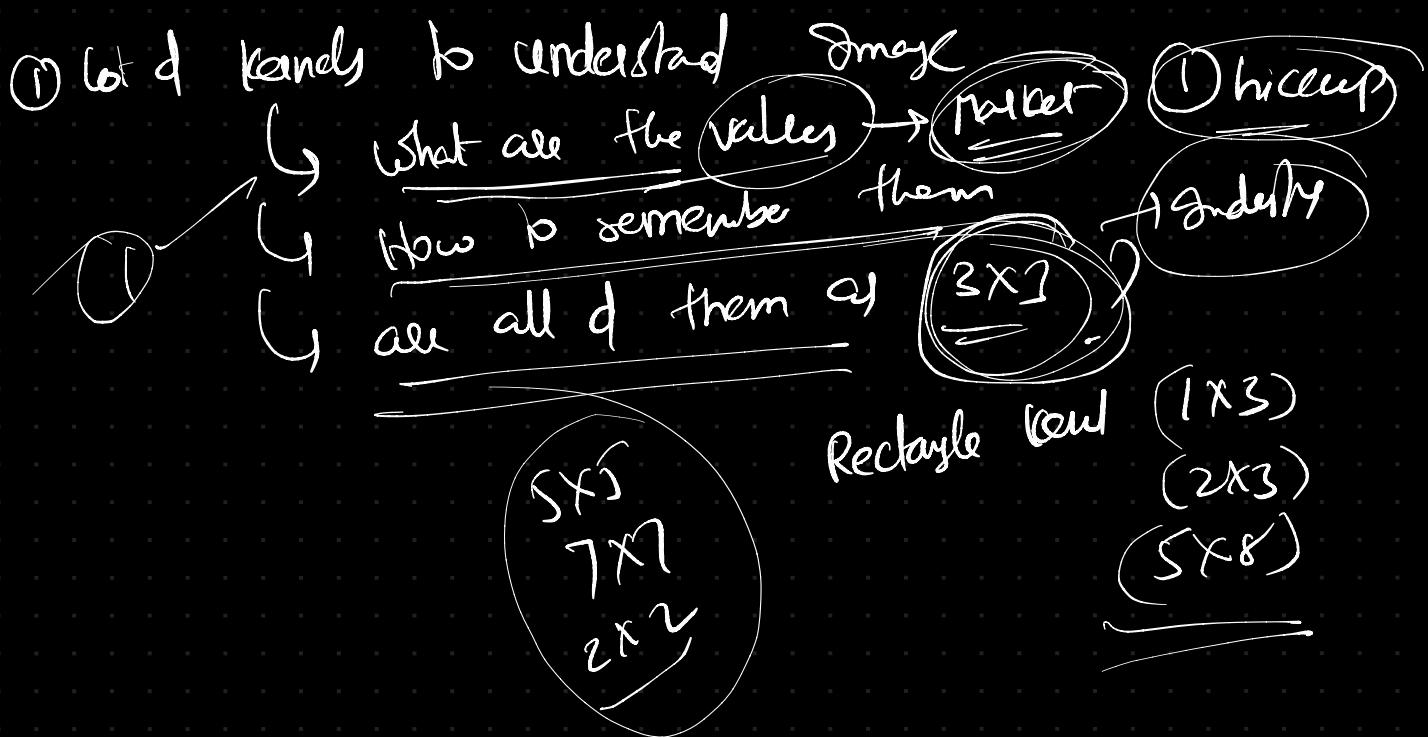
$$\begin{aligned}
 & (+)(255) + (+2)(255) + (+1)(255) \\
 & 0 + 0 + 0 + \\
 & (-1)(255) + (-2)(255) + (-1)(255)
 \end{aligned}$$

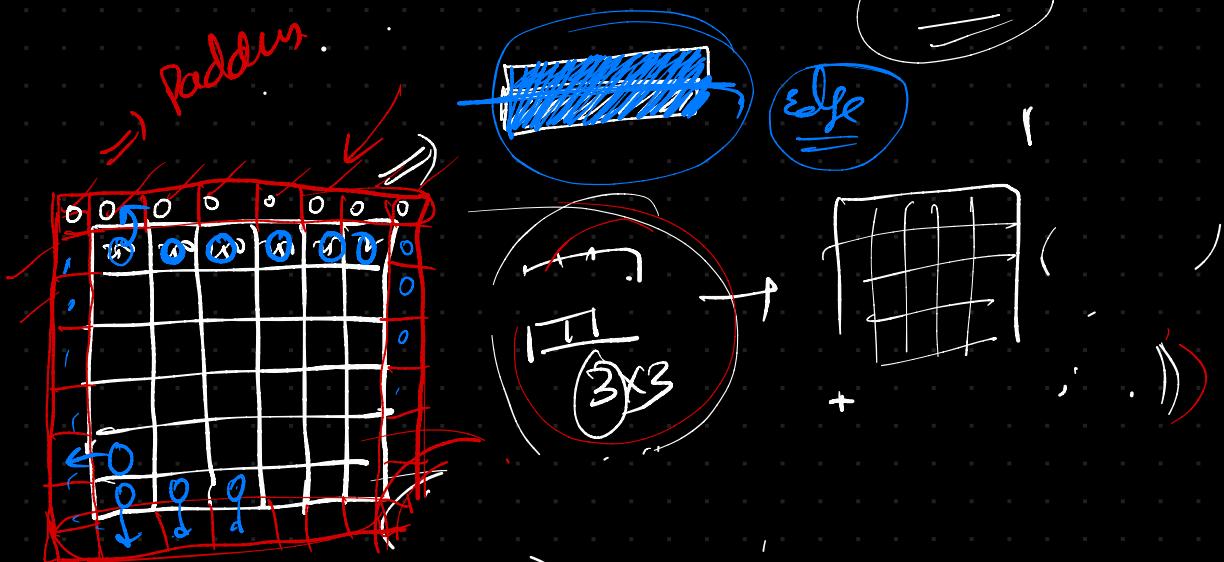
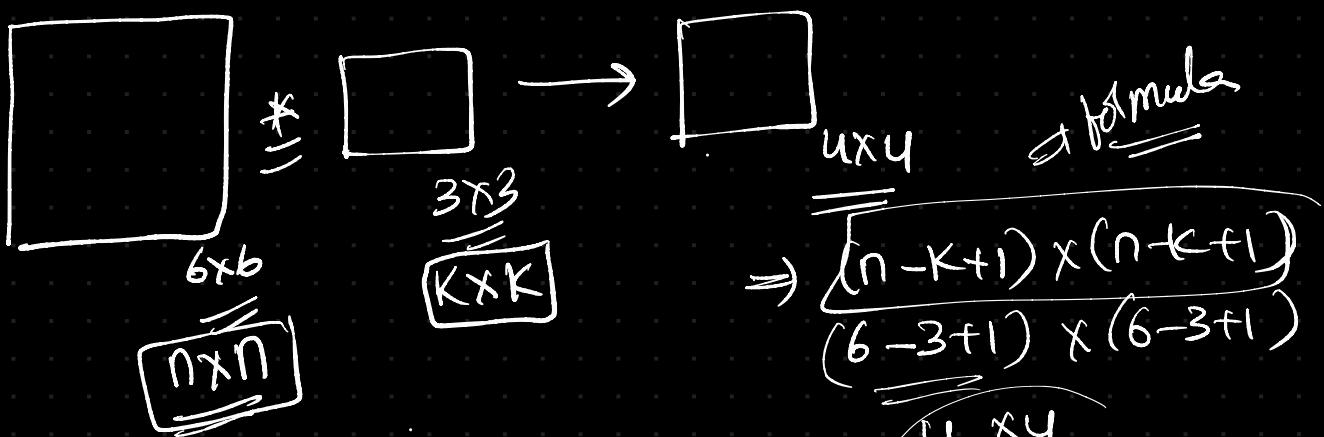
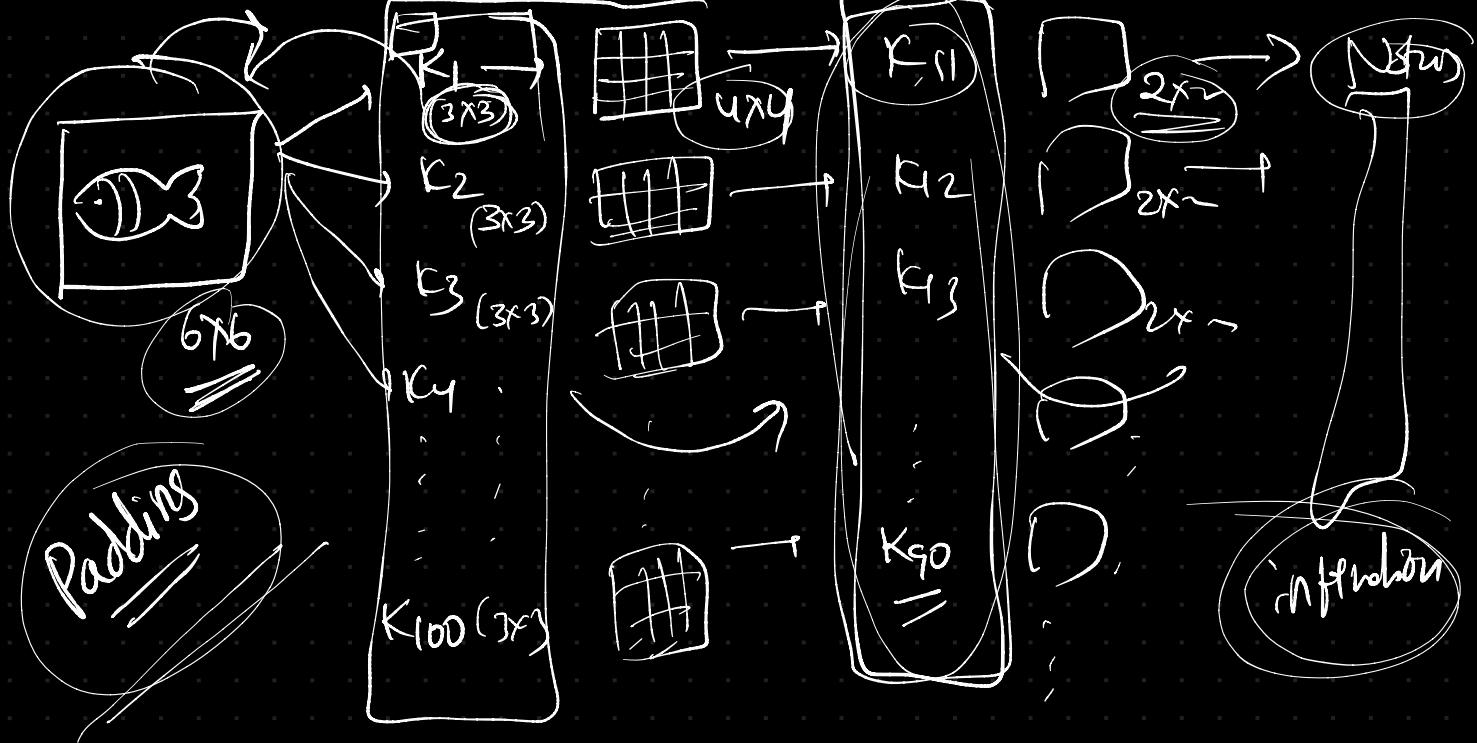


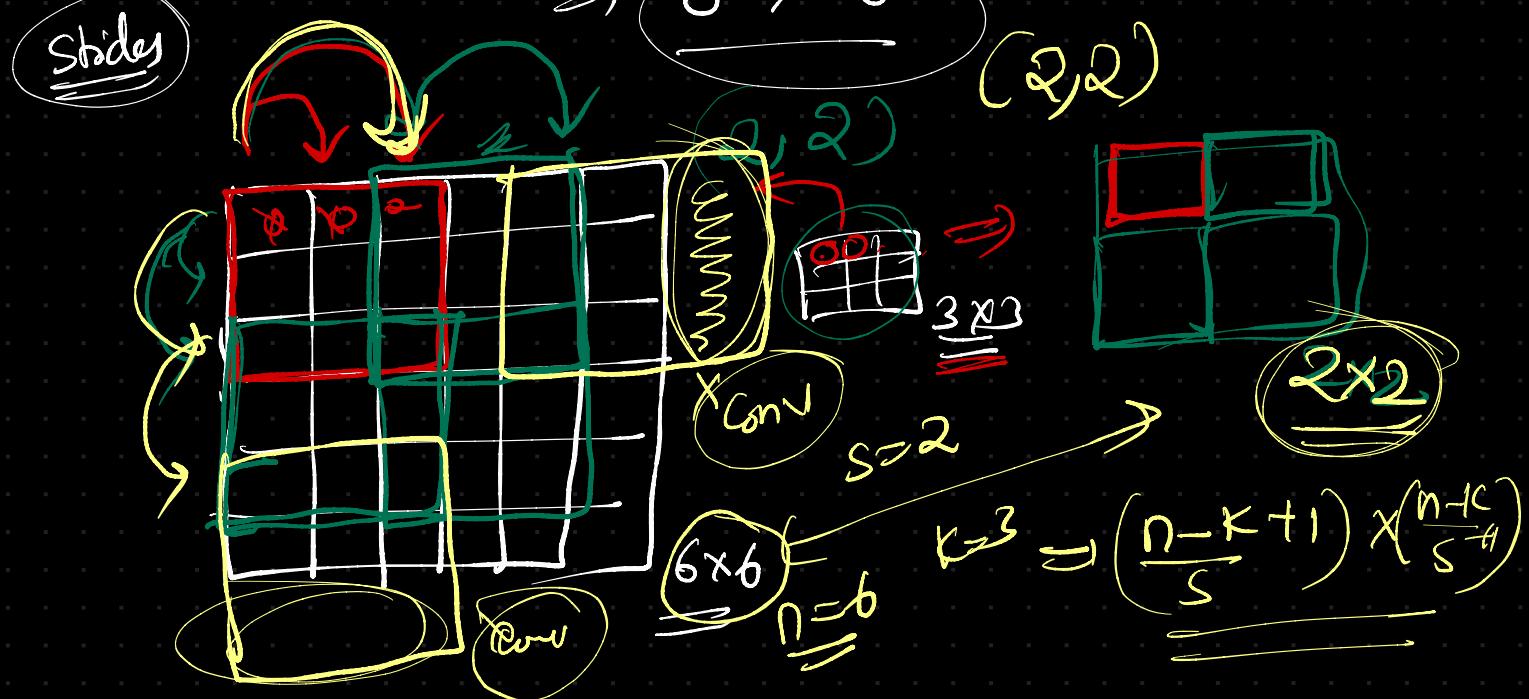
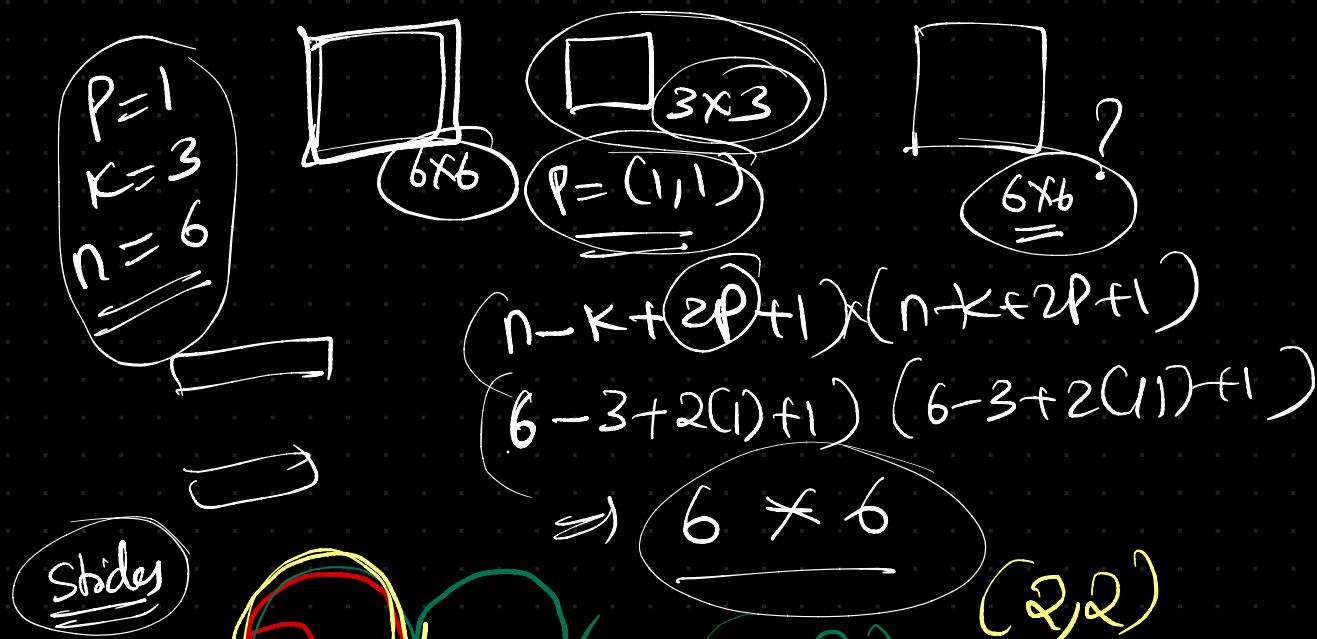
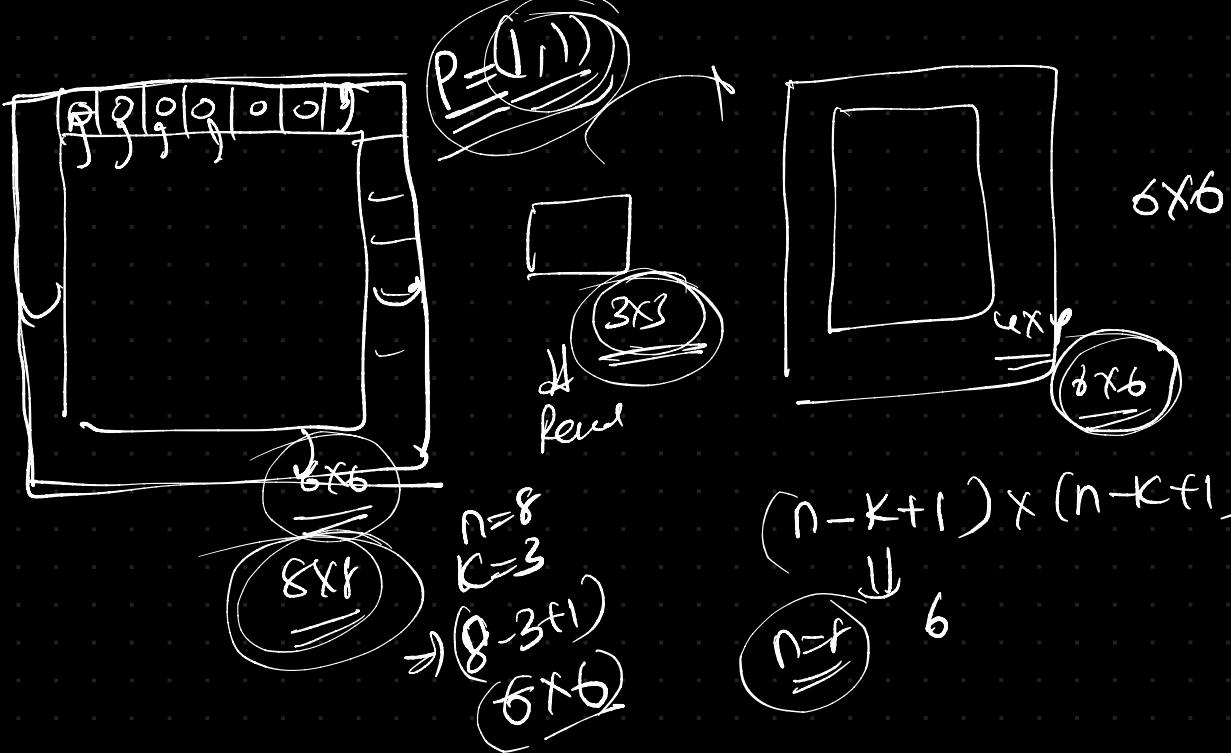
② other way to detect vertical edges
 How are these values defined?

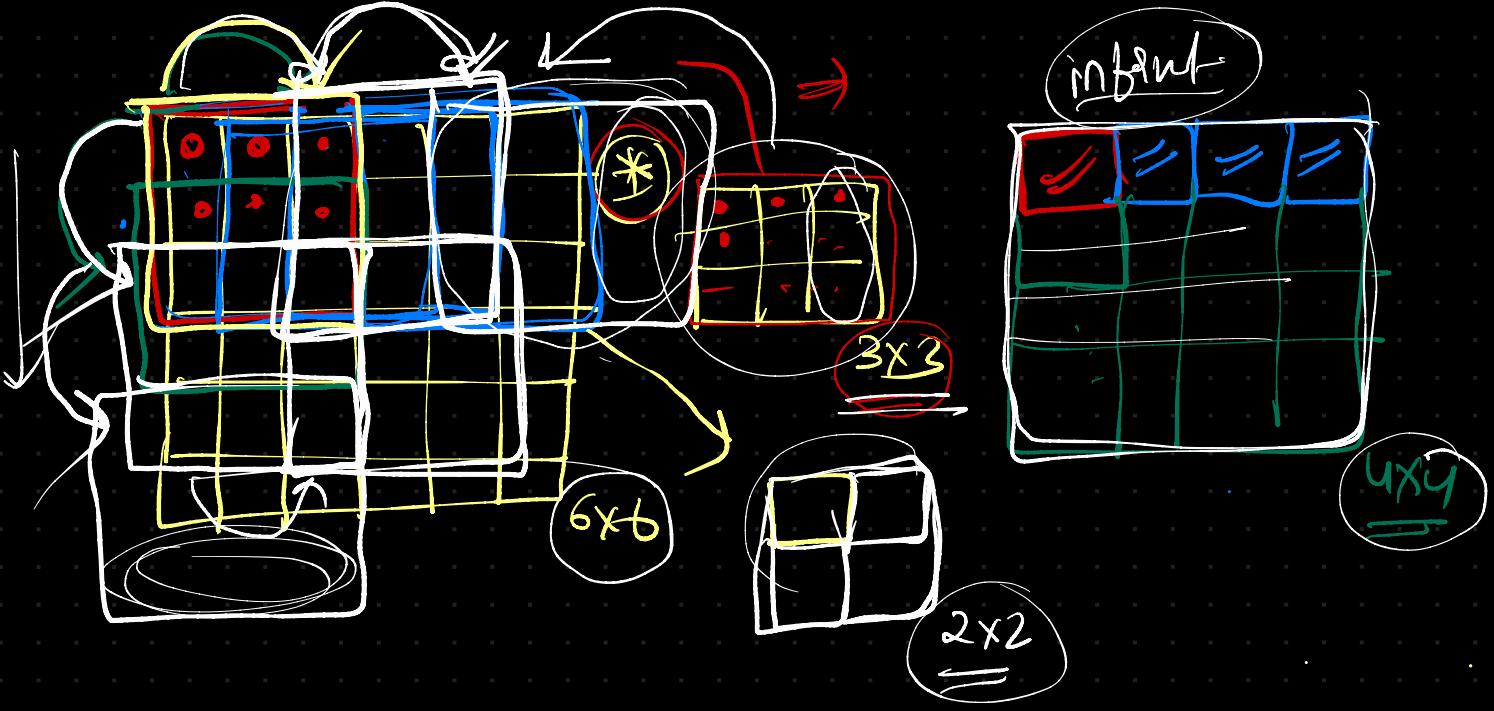
detected horizontal edges



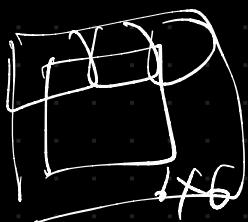




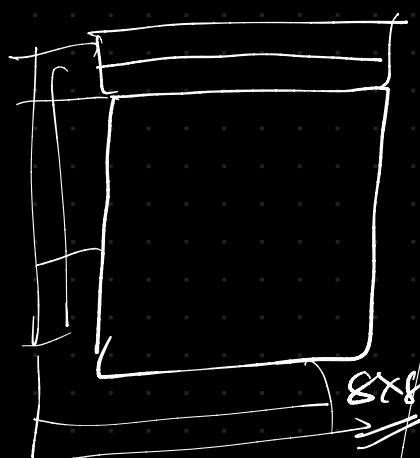




$$\begin{aligned} n &= 6 \\ k &= 3 \\ s &= (2, 2) \\ p &= (1, 1) \\ &= \end{aligned}$$



- o/p shape
- ① only Kernel $\rightarrow (n-k+1) \times (n-k+1)$
 - ② Padding + Kernel $\rightarrow (n-k+2p+1) \times (n-k+2p+1)$
 - ③ strides + Kernel $\rightarrow (\frac{n-k}{s}+1) \times (\frac{n-k}{s}+1)$
 - ④ padding + strides + Kernel $\rightarrow (\frac{n-k+2p}{s}+1) \times (\frac{n-k+2p}{s}+1)$



- o/p shapes
- ① K $\leftarrow \begin{matrix} 3 \times 3 \\ 4 \times 4 \\ 2 \times 2 \end{matrix}$ \Rightarrow o/p shapes
 - ② $K+p$ $\leftarrow \begin{matrix} (2, 2) \\ (1, 1) \end{matrix} \Rightarrow$ o/p shapes
 - ③ $S+K$ \rightarrow o/p shapes
 - ④ $P+S+K$ \rightarrow o/p shapes

