CNN [convolutional neural N/w]

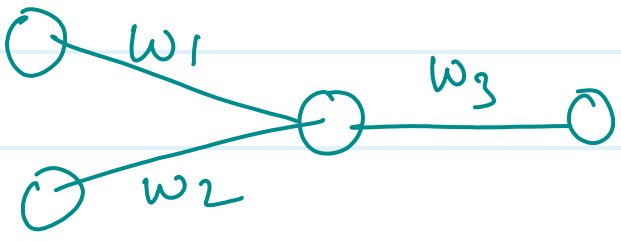
convolution + ANN, Architecture

① weight Initializing Technics

key point for weight initializing

ⓐ Weight should be small
ⓑ weight should not be same
ⓒ weight should have good variance

w_1	w_2	w_2
0.05	0.06	0.07
0.05	0.1	0.15 ✓



no. of i/p = 2
no. of o/p = 1

① Uniform Distribution –

$$w_{ij} \approx \text{uniform Dist.} \left[\frac{-1}{\sqrt{\text{no. of I/P}}}, \frac{1}{\sqrt{\text{no. of I/P}}} \right]$$

lower upper

i = weight

j = Layer

$$\Rightarrow \left[\frac{-1}{\sqrt{2}}, \frac{1}{\sqrt{2}} \right]$$

② Xavior / Glorot Initialization

(a) Xavior Normal Initialization

(b) Xavior uniform initialization

③ X N I

$$w_{ij} \approx N(0, \sigma)$$

$$\sigma = \sqrt{\frac{2}{\text{no. of I/p} + \text{no. of O/p}}}$$

(b) X U I

$$w_{ij} \approx \text{uniform dist.} \left[\frac{-\sqrt{6}}{\sqrt{\text{I/p} + \text{O/p}}}, \frac{\sqrt{6}}{\sqrt{\text{I/p} + \text{O/p}}} \right]$$

(3) kaiming he Initialization

(a) he Normal

$$w_{ij} \approx \mathcal{N}(0, \sigma)$$

$$\sigma = \sqrt{\frac{2}{\text{no. of I/p}}}$$

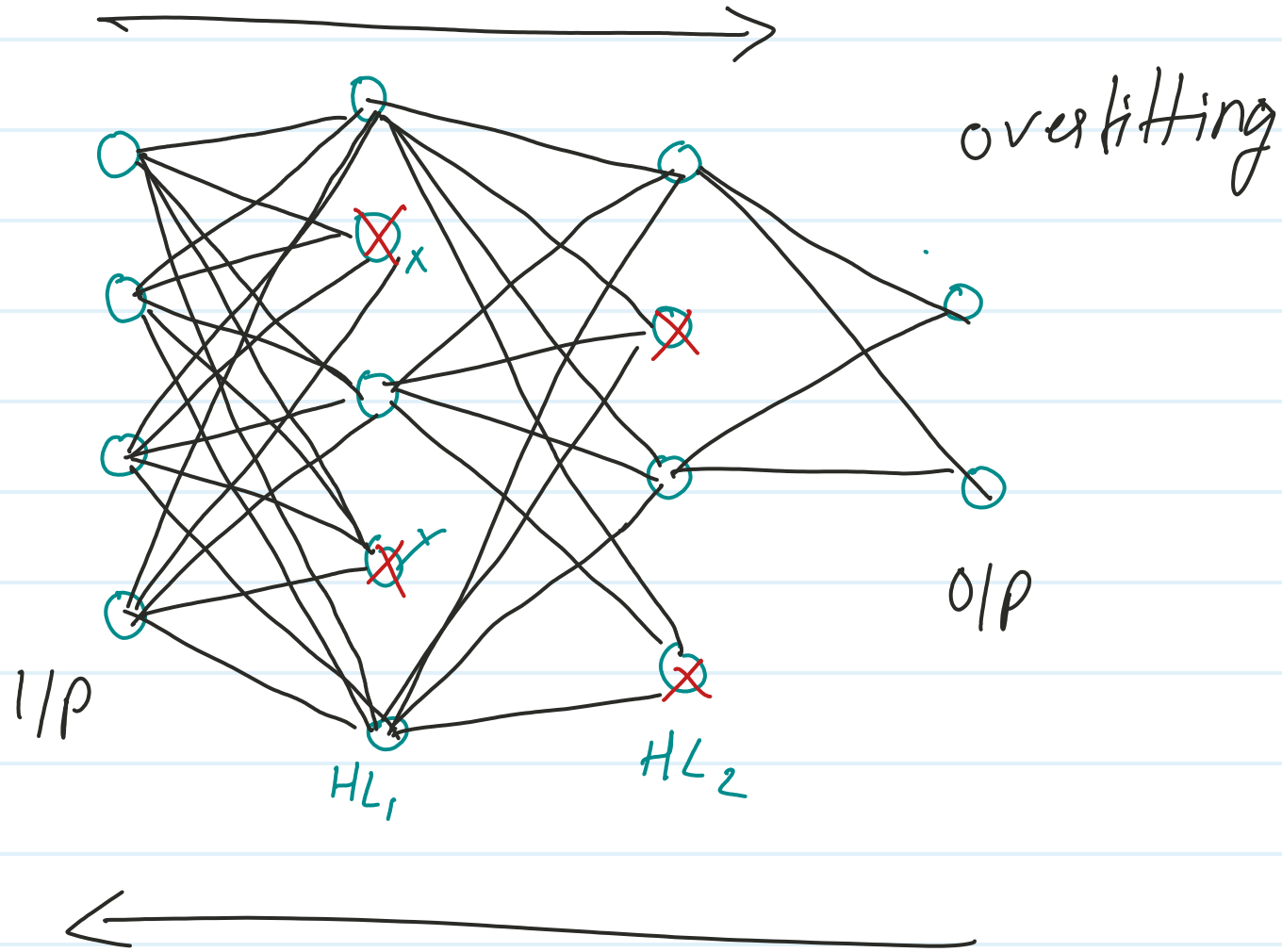
(b) he uniform

$$w_{ij} \approx \text{uni. dist.} \left[-\sqrt{\frac{6}{\text{I/p}}}, \sqrt{\frac{6}{\text{I/p}}} \right]$$

In the CNN we useed most of the time xavior glolot initializatian teeh.

Dropout Layer -

Used to prevent from overfitting.



$P_{HL_1} = 0.5$, $P_{HL_2} = 0.35$, $P_{HL_3} = 0.25$

epoch = In every epoch random neuron will be drop.

CNN

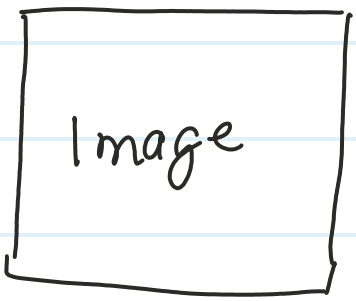
visual cortex -

(V_1 - V_5)

V_1 - primary visualization (orientation, edges, lines)

V_2 = Diff. in color, complex recognition

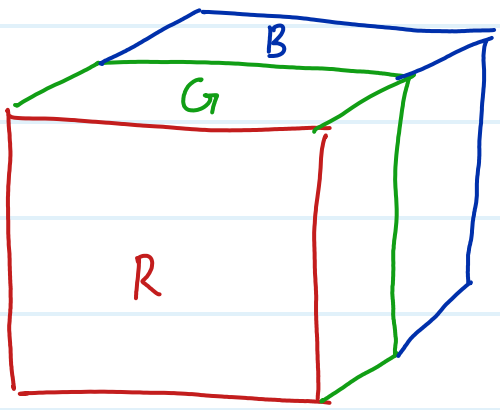
V_3 V_4 V_5



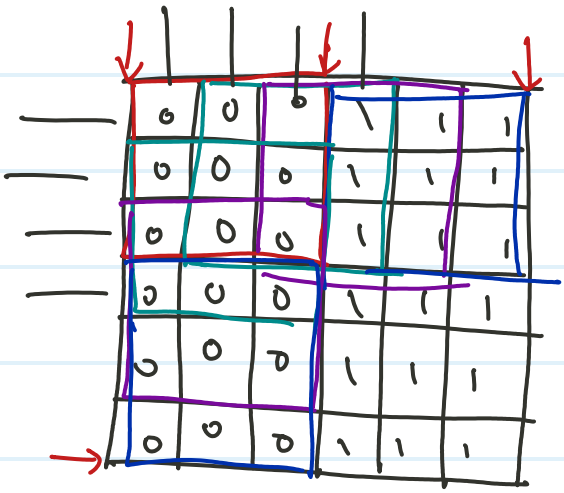
→ RGB

→ Gray scale

→ A - channel for transparency



convolution operation



6 x 6

I/p ↑

+1	0	-1
+2	0	-2
+1	0	-1

3x3 — vertical edge filter

⇐ filter

1	2	1
0	0	0
-1	-2	-1

— Horizontal edge filter.

stride (shift)

o/p →

0	-4	-4	0
0	-4	-4	0
0	-4	-4	0
0	-4	-4	0

4 x 4

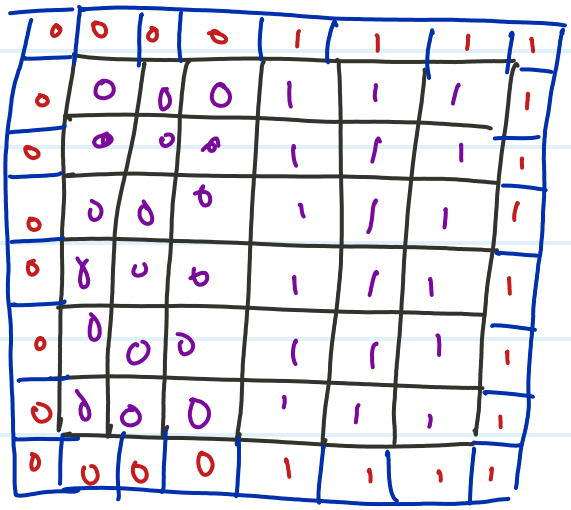
$(n-f)+1 =$

$6-3+1 = 4$

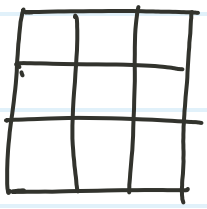
we have 6×6 metrics with filter image by 3×3 metrics but getting final image by 4×4,

So we are loosing some informetion.

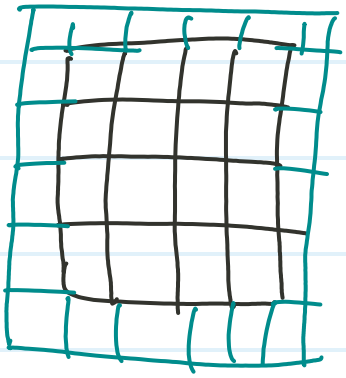
To overcome this we use another teehnig called padding



6×6



3×3



4×4

8×8

6×6

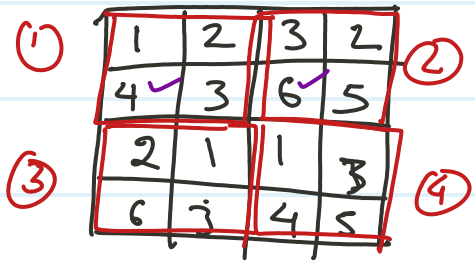
$$n - f + 2p + 1 = 6$$

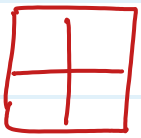
$$6 - 3 + 2p + 1 = 6$$

$$2p = 6 - 4$$

$$p = 2/2 = \boxed{1}$$

max pulling

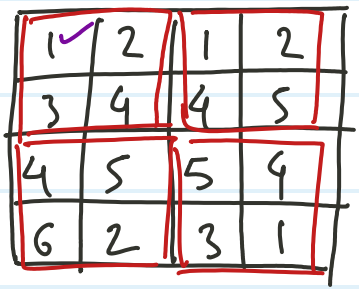


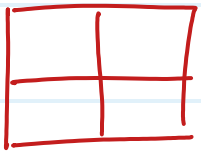


4	6
6	5

strids jump = 2

min pulling

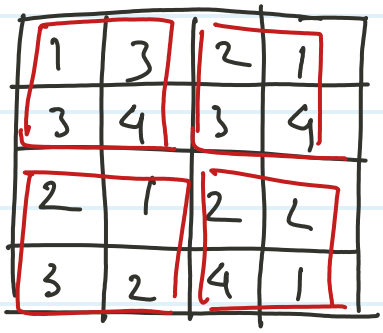


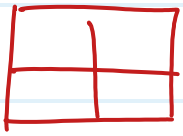


1	1
2	1

Smoothning of image for its use min and max pooling

mean pulling





2.9	2.9
2	2.1