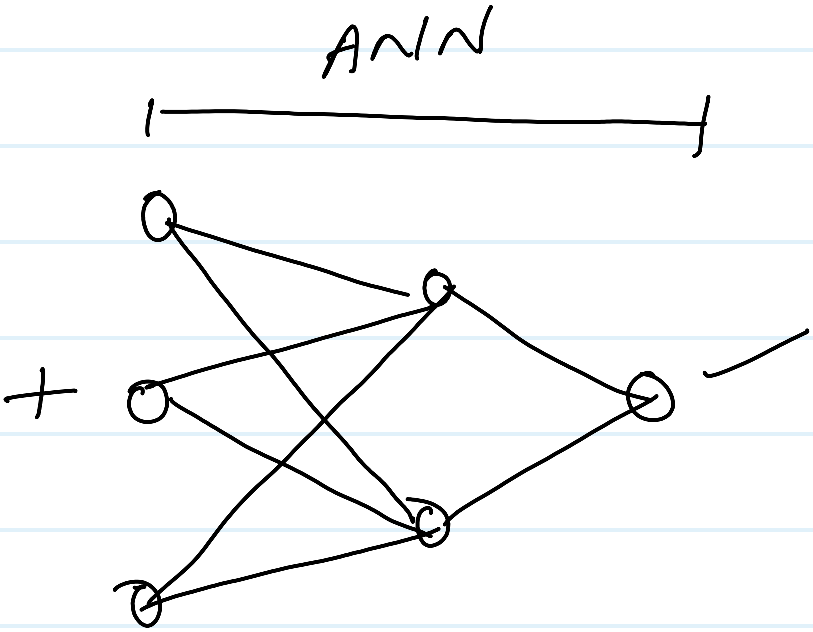
CNN



CNN [convolutional Neural N/w]

<u>convolution + ANN,</u>

Architecture

① weight Initializing Technics

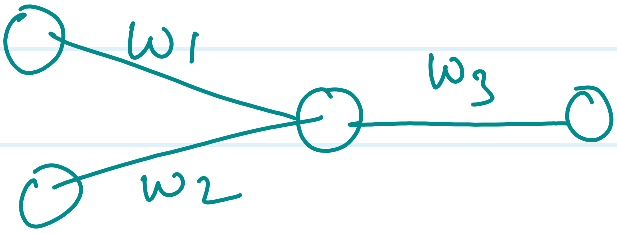
key point for weight initializing

ⓐ 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

(I) Uniform Distribution –

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

i = weight
j = Layer

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

(II) Xavior/Glorot Initialization

(a) xavior Normal Initialization
(b) xavior uniform initialization

(a) X N I

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

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

ⓑ XUI

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

③ kaiming he Initialization

ⓐ he Normal

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

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

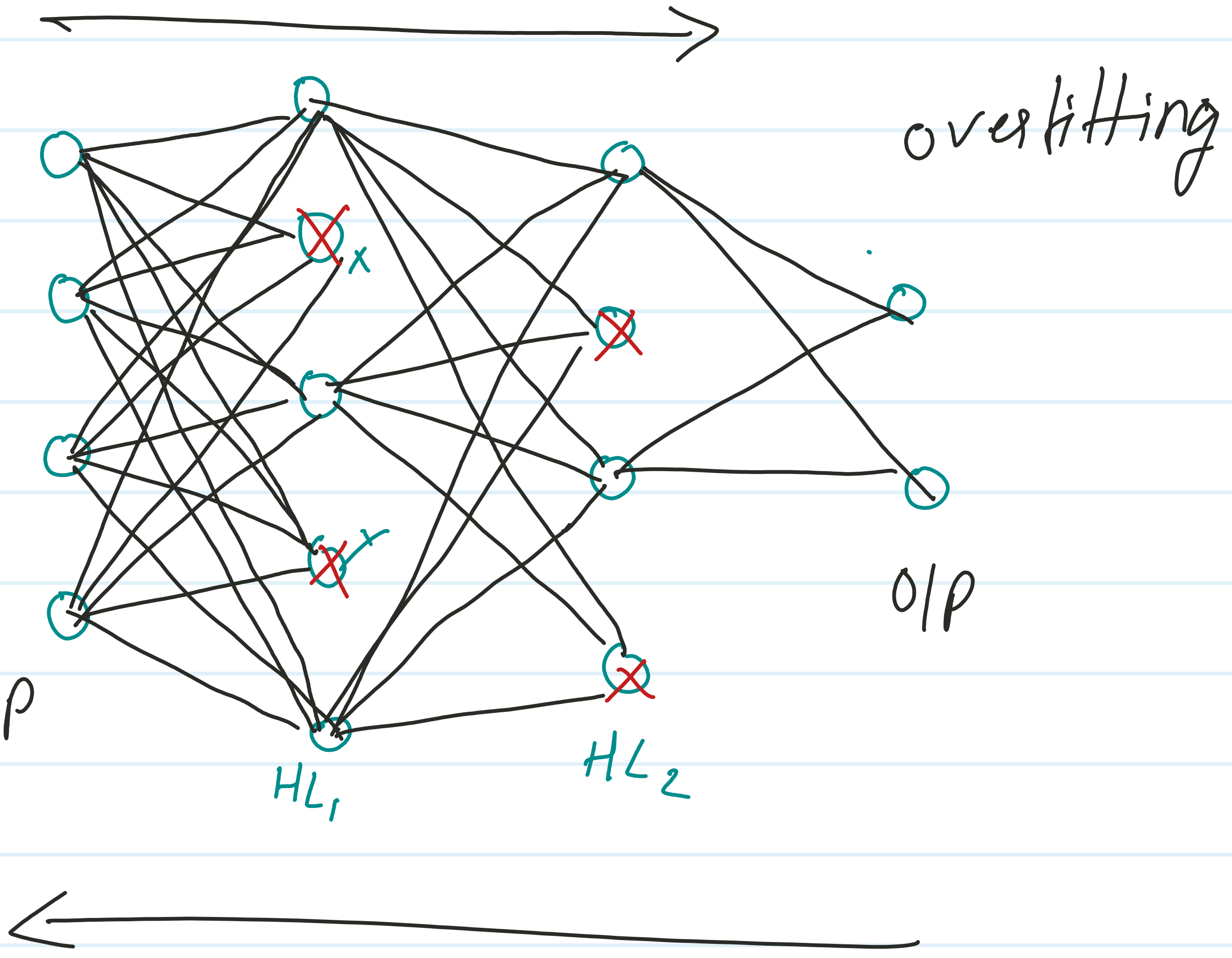
ⓑ he uniform

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

In the CNN we used most of the time xavior glolot initialization tech.

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 (orientantion, edges, Lines)

↓

V_2 = Diff. in color, complex recognition

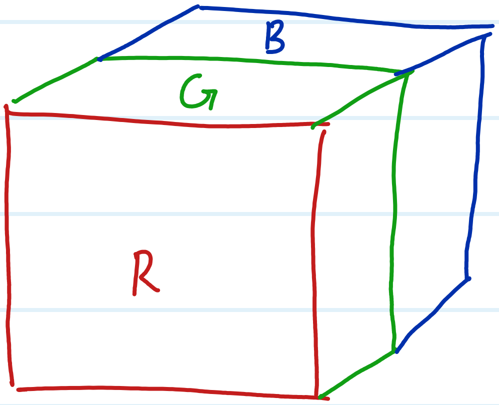
V_3 V_4 V_5

Image

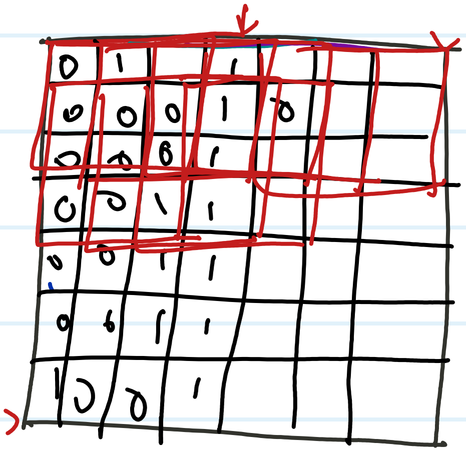
→ RGB

→ Gray sale

→ A- channel for transparency



Convolution operation



6 x 6

I/p ↑

+1	0	−1
+2	0	−2
+1	0	−1

— vertical edge filter

3×3

⇐ filter

1	2	1
0	0	0
−1	−2	−1

— Horizontal edge filter.

stride (shift)

O/p →

0	-4	-4	6
0	-4	-4	0
0	-4	-4	0
0	-4	-4	0

4x4

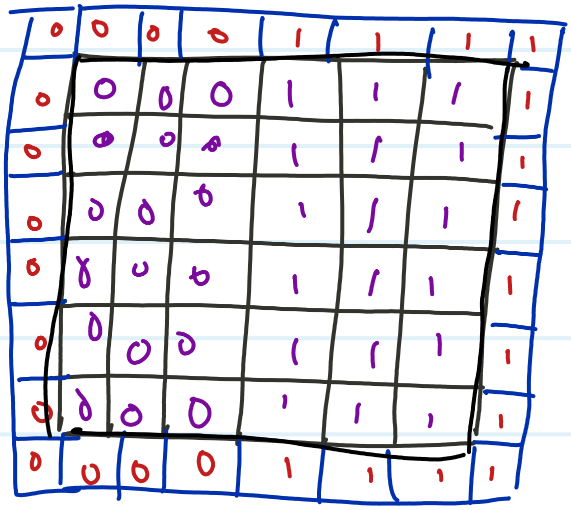
$(n-f)+1 =$

$6-3+1 = 4$

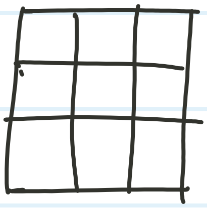
we have 6x6 metrics with filter image by 3x3 metrics but getting final image by 4x4,

So we are loosing some information.

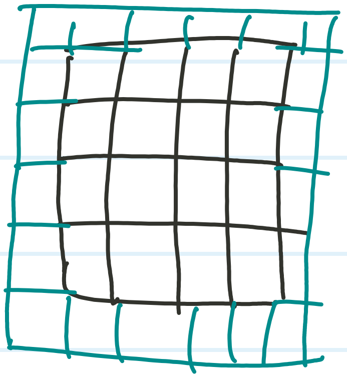
To overcome this we use another techniq called padding



6x6



3x3



4x4

→ 8x8

6x6

Padding

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

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

$$2p = 6 - 4$$

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

max pulling

2	1	4	2
3	0	0	1
1	2	3	1
2	1	4	0

3	4
2	4

strids jump = 2

min pulling

1✓	2	1	2
3	4	4	5
4	5	5	4
6	2	3	1

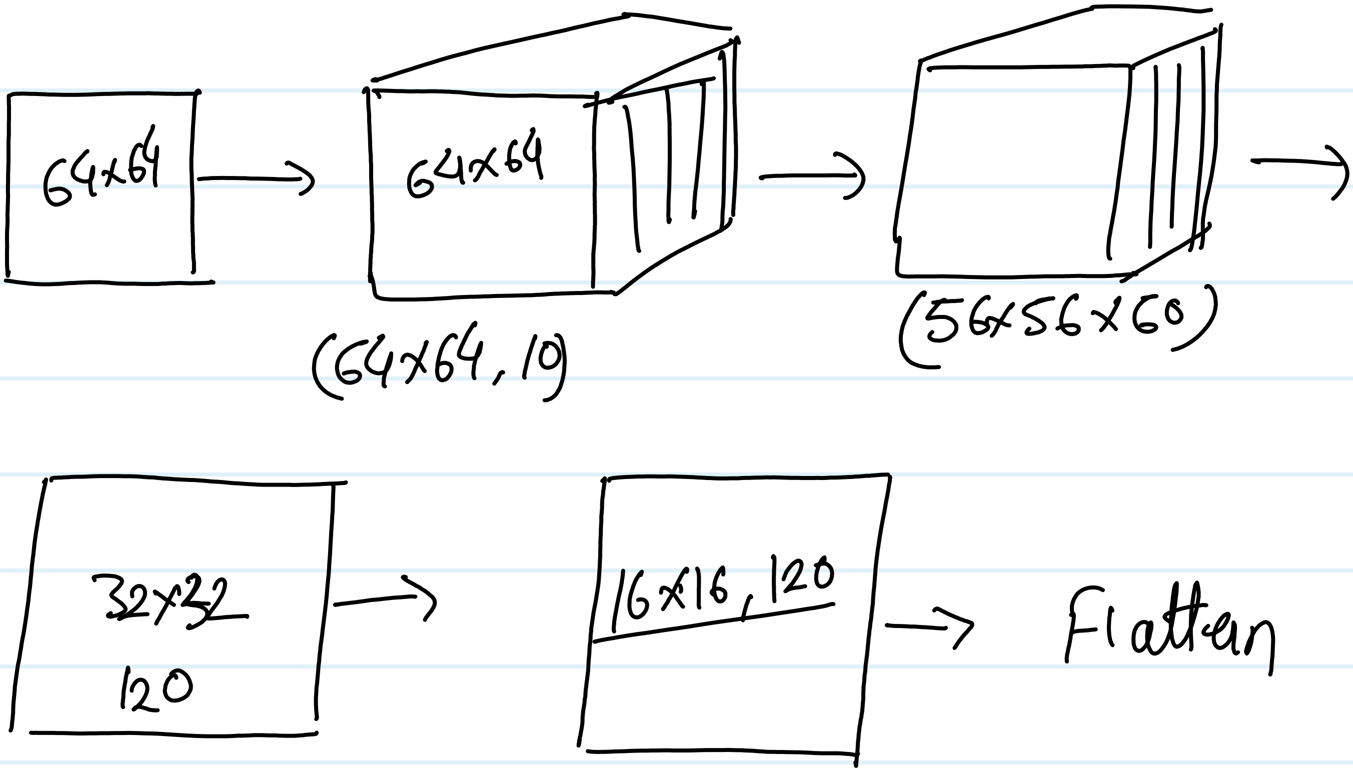
1	1
2	1

Smoothning of image for its use min and max pooling

mean pulling (Avg pulling)

1	3	2	1
3	4	3	4
2	1	2	2
3	2	4	1

2.9	2.9
2	2.1



16x16x120