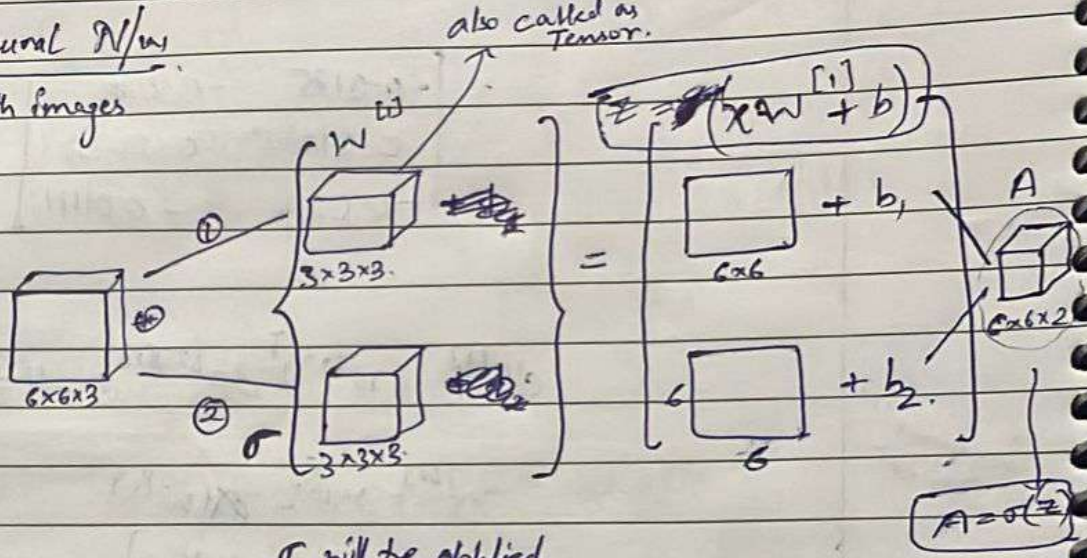
17/3/25.

1) Tensorflow / Keras.
2) Pytorch.

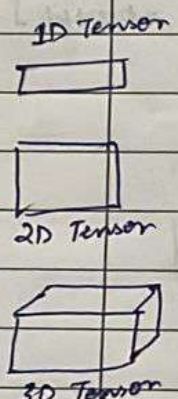
Convolution Neural N/w

- works with images



Padding
$p = \frac{f-1}{2}$

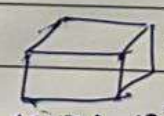
$$Z^{[2]} = W^{[2]} A + B.$$
$$A^{[2]} = \sigma(Z^{[2]})$$



(H, W, C, ~~N~~ F)
↓ Filters
(3, 3, 3, 2)

Example:- If u have 10 filters of 3×3×3, how many learnable parameters are there, which will be learned during back propogation.

Sol:-



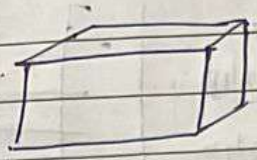
$= 27 + b_1$

Total elements
$27 = (3 \times 3 \times 3)$
+ 10 biases

$27 \times 10 = 270 + 10 = \underline{280}$ (Ans)
↓
10 biases

$n+2p-f$

Q.



$39 \times 39 \times 3$

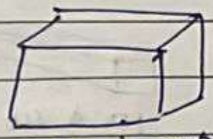
$f^{[1]} = 3$, $f = 3 \times 3$

$s^{[1]} = 1$

$P = 0$

$n_c = 10$.

$\longleftarrow (37 \times 37 \times 3, 10)$ Tensor.



Dimension of this Tensor

$\Rightarrow \left[\frac{n+2p-f}{s}+1\right]$

$n = 39$.

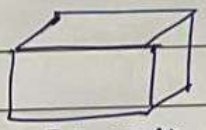
$p = 0$

$\left[\frac{39+2\times0-3}{1}+1\right]$

$= 37 \times 37 \times 10$

no of filters (n_c)

$\left[3 \times 3 \times 3 \times 10 + 10 = 280\right]$



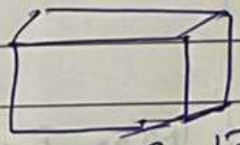
$37 \times 37 \times 10$

$f^{[2]} = 5$ (2nd layer filter of size 5)

$s^{[2]} = 2$

$p^{[2]} = 0$ [None]

$n_c^{[2]} = 20$.



? $17 \times 17 \times 20$

$(5, 5, 10, 20)$

$= 5 \times 5 \times 10 \times 20 + 20$

$= 5k$.

$\left[\frac{37+2\times0-5}{2}+1\right]$

$37 - 5 = 32$

$[16+1] \Rightarrow 17 \times 17 \times 20$

$f^{[3]} = 5$

$s^{[3]} = 2$ (stride)

$n_c^{[3]} = 40$

padding = 'valid'

$(5, 5, 20, 40)$

$(5 \times 5 \times 20 \times 40 + 40)$

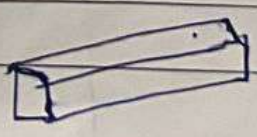
20k.

cal. padding $p = \frac{f-1}{2} = \frac{5-1}{2} = 2$.

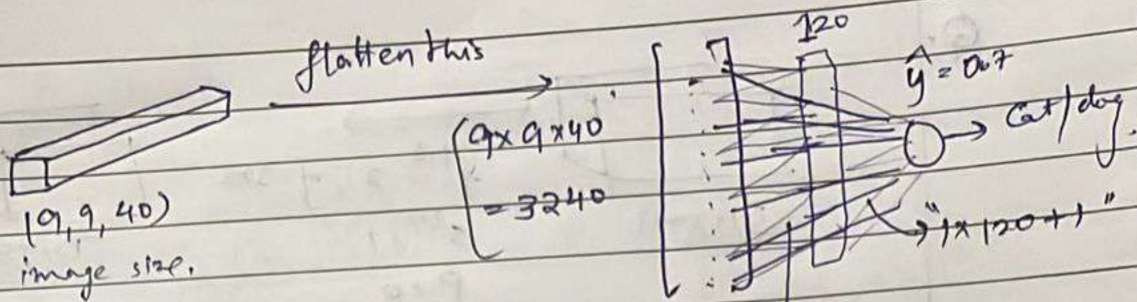
$\left[\frac{n+2p-f}{s}+1\right] = \left[\frac{17+2\times2-5}{2}+1\right]$

$= 9$.

$\Rightarrow (9, 9, 40)$



_/_/_



120x 3240 + 120
bias.

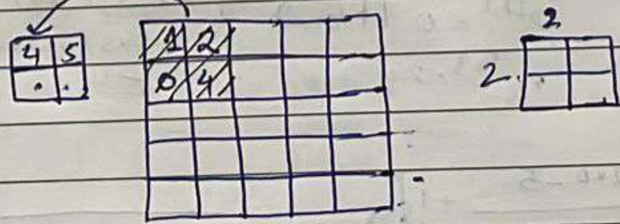
= [4L]

324
x 12
648
324x
3888 0
120
3 9 0 00

(Total hyperparam. 280 + 20k + 5k + 4L + 121)
= 4.25 L.

Max Pooling. (There are no trainable parameter)

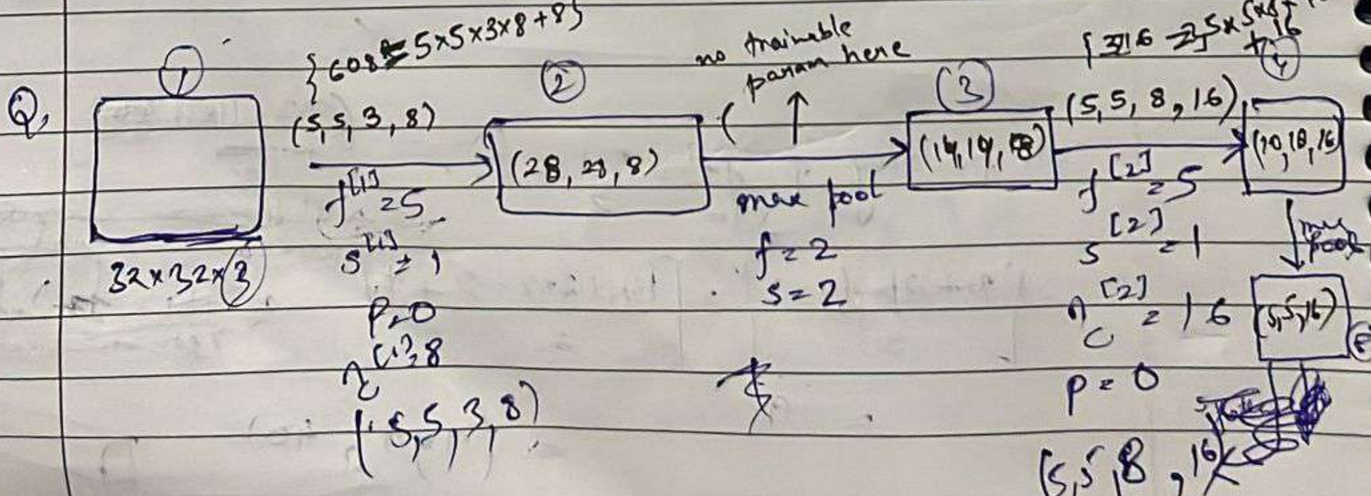
image reduction.



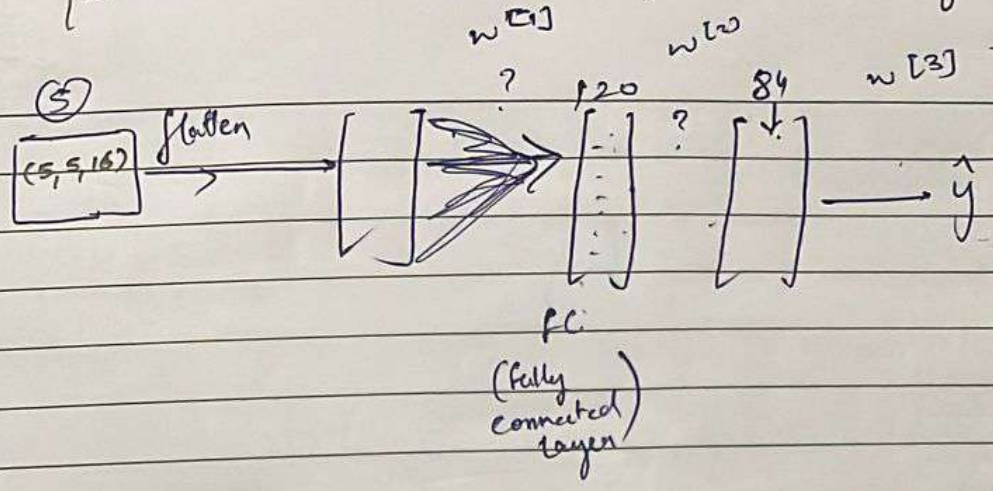
stride = 2.
filter = 2

maximum value of pixel is selected
mostly (2x2) matrix is used,

(32, 32, 3) —Max pooling→ (16, 16, 3)
channel will remain same.



{LeNet-5 architecture.} ⇒ {CNN + max pooling}



① $f^{[1]} = 5$, $s^{[1]} = 1$, $p = 0$, $n_c^{[1]} = 8$

$\left[\frac{n + 2p - f}{s} + 1\right] = \left[\frac{32 + 2\times0 - 5}{1} + 1\right]$

$= 28.$

$(28, 28, 8)$.

② $(14, 14, 8)$.

③ $\left[\frac{14 + 2\times0 - 5}{1} + 1\right] = 10.$ $(10, 10, 16)$.

④ $(5, 5, 16)$

⑤ $5 \times 5 \times 16 = 400$

⑥ $120 \times 400 + 120 = 48120$

⑦ $84 \times 120 + 84 = 10{,}164$

⑧ $1 \times 84 + 1 = 85.$

Total = $48120 + 10164 + 85 + (5\times5\times3\times8 + 8) + (5\times5\times8\times16 + 16)$

$= 62\text{K}. (62193)$