| Layer | Output Volume Shape | Number of Parameters |
|---|---|---|
| Input | $28 \times 28 \times 3$ | 0 |
| CONV5-8 | ? | ? |
| Leaky ReLU | ? | ? |
| POOL-2 | ? | ? |
| FLATTEN | ? | ? |
| FC-10 | ? | ? |

Input $28 \times 28 \times 3$ 0

CONV 5-8 $26 \times 26 \times 8$ $8(5 \times 5 \times 3 + 1)$

$$\frac{28 - 5 + 2(1)}{1} + 1$$

$= 26$

(assuming padding = 1, stride = 1)

→ 8 $(5 \times 5 \times 3 + 1)$

1 → bias associated with each kernal

8 → no. of neurons, as there is kernal associated with each neuron

5 x 5 x 3 → same as I/p img's channel depth

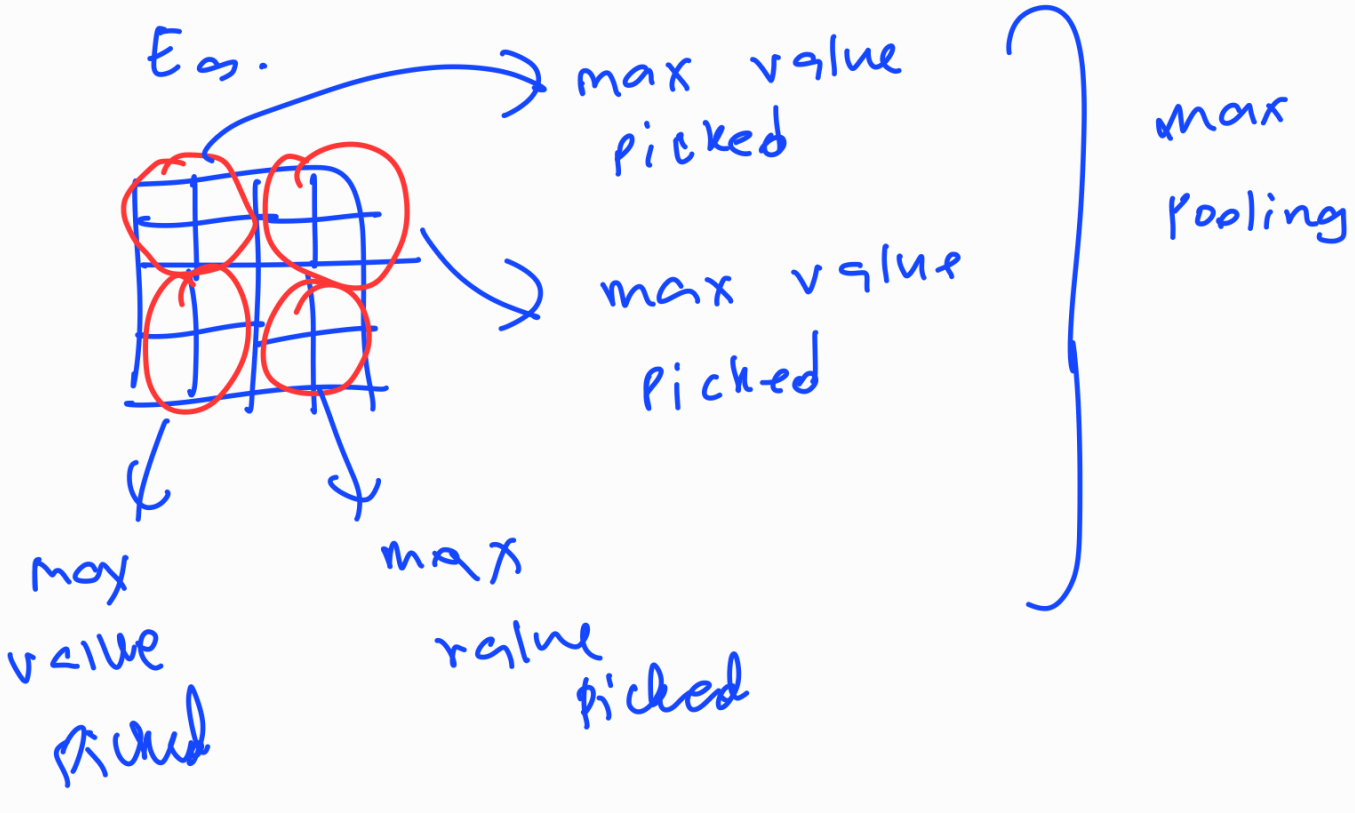
↓ Kernal size

| | | |
|---|---|---|
| LeakyReLU | 26 x 26 x 8 | 0 |

Point wise activation so no change in output size and no parameters

| | | |
|---|---|---|
| Pool-2 | 13 x 13 x 8 | 0 |

This just reduces the size of input image. Here assuming stride is 2, kernal size = 2, size becomes half



Flatten 1352 0

converts 3D tensor to 1D

just multiply previous output shape

Fc - 10 10 x 1 10 (1352 + 1)

↓ ↓ ↓

these are 10 neurons so bias

each neuron will have 1352

parameters.

(In normal NN, there is weight

associated with each pixel unlike

CNN)

↓

where learning through

happens channel + kernals

wise & locally