

Convolutional Neural Networks

23 January 2024 08:23 PM

1) CONVOLUTION

2) PADDING

3) POOLING

4) STRIDES

1) CONVOLUTION OPERATION:

$$\begin{array}{c}
 A \\
 \begin{array}{ccc} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \\ a_{20} & a_{21} & a_{22} \end{array}
 \end{array}
 *
 \begin{array}{c}
 F \\
 \begin{array}{ccc} f_{00} & f_{01} & f_{02} \\ f_{10} & f_{11} & f_{12} \\ f_{20} & f_{21} & f_{22} \end{array}
 \end{array}
 = a_{00}f_{00} + a_{01}f_{01} + a_{02}f_{02} \\
 + a_{10}f_{10} + a_{11}f_{11} + a_{12}f_{12} \\
 + a_{20}f_{20} + a_{21}f_{21} + a_{22}f_{22}$$

Ex1:

$$\begin{array}{c}
 \begin{array}{ccc} 1 & 0 & 2 \\ 2 & 1 & 1 \\ 0 & 1 & 1 \end{array}
 * \begin{array}{ccc} 3 & 0 & 1 \\ 2 & 1 & 2 \\ 1 & 0 & 1 \end{array}
 = 13
 \end{array}$$

$$\begin{array}{c}
 \begin{array}{ccc} 2 & 4 & 3 \\ 7 & 3 & 6 \\ 8 & 1 & 2 \end{array}
 * \begin{array}{ccc} 1 & 0 & 2 \\ 0 & 2 & 1 \\ 3 & 1 & 2 \end{array}
 = 49
 \end{array}$$

$$\begin{array}{c}
 \begin{array}{c}
 \begin{array}{c} \cancel{\cancel{1}} \quad \cancel{\cancel{1}} \\ \cancel{\cancel{1}} \quad \cancel{\cancel{1}} \end{array} \\
 \begin{array}{cccccc} 1 & 0 & 2 & 3 & 1 & 1 \\ 2 & 1 & 2 & 4 & 2 & 3 \\ 3 & 2 & 0 & 1 & 4 & 1 \\ 2 & 1 & 0 & 1 & 2 & 1 \\ 1 & 0 & 3 & 2 & 1 & 3 \\ 3 & 0 & 1 & 2 & 1 & 1 \end{array}
 \end{array}
 * \begin{array}{c}
 \begin{array}{c} \cancel{\cancel{1}} \quad \cancel{\cancel{1}} \\ \cancel{\cancel{1}} \quad \cancel{\cancel{1}} \end{array} \\
 \begin{array}{ccc} 1 & 0 & 2 \\ 1 & 2 & 1 \\ 2 & 1 & 0 \end{array}
 \end{array}
 = \begin{array}{c}
 \begin{array}{c} \cancel{\cancel{1}} \quad \cancel{\cancel{1}} \\ \cancel{\cancel{1}} \quad \cancel{\cancel{1}} \end{array} \\
 \begin{array}{cccccc} 19 & 19 & 17 & 22 \\ 18 & 14 & 13 & 24 \\ 9 & 9 & 20 & 14 \\ 12 & 12 & 16 & 15 \end{array}
 \end{array}$$

INPUT $(6, 6)$ FILTER $(3, 3)$ $(4, 4)$

$$\begin{array}{c}
 \begin{array}{c} \cancel{\cancel{1}} \quad \cancel{\cancel{1}} \\ \cancel{\cancel{1}} \quad \cancel{\cancel{1}} \end{array} \\
 \begin{array}{cccccc} \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \end{array}
 \end{array}
 *
 \begin{array}{c}
 \begin{array}{c} \cancel{\cancel{1}} \quad \cancel{\cancel{1}} \\ \cancel{\cancel{1}} \quad \cancel{\cancel{1}} \end{array} \\
 \begin{array}{c} \cdot \cdot \cdot \\ \cdot \cdot \cdot \\ \cdot \cdot \cdot \end{array}
 \end{array}
 = \begin{array}{c}
 \begin{array}{c} \cancel{\cancel{1}} \quad \cancel{\cancel{1}} \\ \cancel{\cancel{1}} \quad \cancel{\cancel{1}} \end{array} \\
 \begin{array}{c} \cdot \cdot \cdot \\ \cdot \cdot \cdot \\ \cdot \cdot \cdot \end{array}
 \end{array}$$

3×3 8×8 6×6

$$\begin{array}{c}
 * \\
 \boxed{\text{Input: } 8 \times 8} \quad * \quad \boxed{\text{Filter: } 5 \times 5} \quad = \quad \boxed{\text{Output: } 4 \times 4}
 \end{array}$$

$n \times n$: Shape of input

$n_f \times n_f$: Shape of filter

OUTPUT SHAPE : $n - n_f + 1, n - n_f + 1$

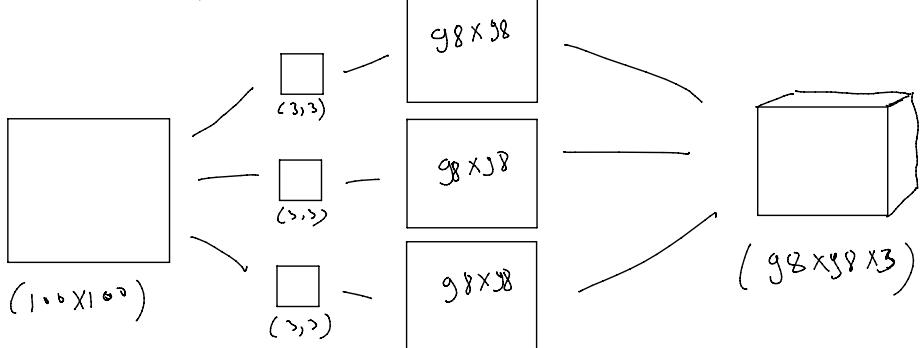
Ex.

$$\begin{array}{c}
 \boxed{\text{Input: } 100 \times 100} \quad * \quad \boxed{\text{Filter: } 3 \times 3} \quad = \quad \boxed{\text{Output: } 98 \times 98}
 \end{array}$$

* RGB IMAGES :-

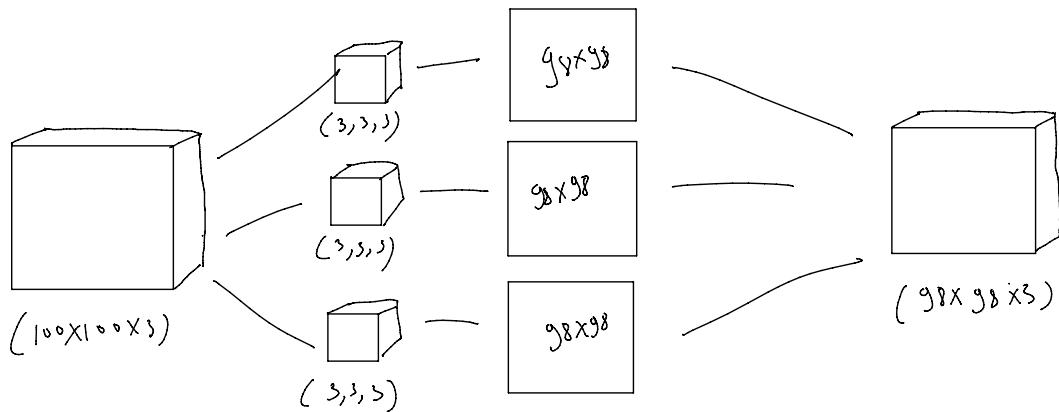
$$\begin{array}{c}
 \boxed{\text{Input: } 100 \times 100 \times 3} \quad * \quad \boxed{\text{Filter: } 3 \times 3 \times 3} \quad = \quad \boxed{\text{Output: } 98 \times 98}
 \end{array}$$

* MULTIPLE FILTERS :-



* MULTIPLE FILTERS (RGB) :-

* MULTIPLE FILTERS (RGB) :-



* SIGNIFICANCE OF CONVOLUTION OPERATION :-

$$\begin{array}{c} \text{Input Image} \\ * \end{array} \quad \begin{array}{c} \text{Filter} \\ = \end{array} \quad \begin{array}{c} \text{Output Image} \\ \text{Vertical edge} \end{array}$$

A diagram showing a convolution operation where a vertical edge filter (a 3x3 matrix with values 1, 0, -1; 1, 0, -1; 1, 0, 1) is applied to an input image to detect vertical edges.

$$\begin{array}{c} \text{Input Image} \\ * \end{array} \quad \begin{array}{c} \text{Filter} \\ = \end{array} \quad \begin{array}{c} \text{Output Image} \\ \text{Horizontal edges} \end{array}$$

A diagram showing a convolution operation where a horizontal edge filter (a 3x3 matrix with values 1, 1, 1; 0, 0, 0; -1, -1, -1) is applied to an input image to detect horizontal edges.



Vertical edges



Horizontal edges

* PADDING :-

$$\begin{matrix} & \times & = & \\ \begin{matrix} 8 \times 8 \end{matrix} & \begin{matrix} (3 \times 3) \end{matrix} & & \begin{matrix} (2 \times 2) \end{matrix} \end{matrix}$$

$$\begin{matrix} & \times & = & \\ \begin{matrix} (12, 12) \end{matrix} & \begin{matrix} 5 \times 5 \end{matrix} & & \begin{matrix} 8 \times 8 \end{matrix} \end{matrix}$$

* VALID = No Padding

* SAME = Maintains Resolution

- > $n \times n$: Shape of input
- > $n_f \times n_f$: Shape of filter
- > p : no. of padding layer

OUTPUT SHAPE : $n - n_f + 1 + 2p, n - n_f + 1 + 2p$



* STRIDES :-

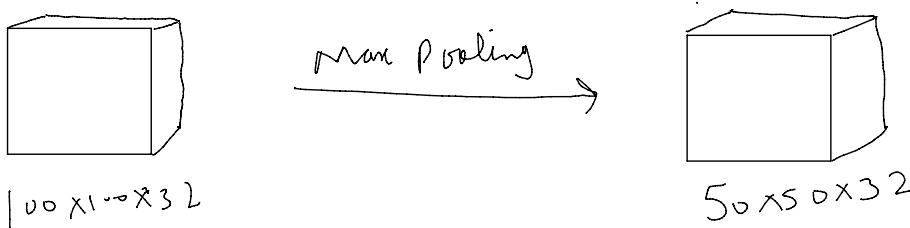
OUTPUT : $\frac{n - n_f + 2p}{s} + 1, \frac{n - n_f + 2p}{s} + 1$

$\text{X} \rightarrow \text{X} \rightarrow \text{X} \rightarrow \text{X}$

* POOLING:-

1) MAX POOLING:-

$$\begin{matrix} 2 & 3 & 2 & 8 \\ 9 & 1 & 3 & 5 \\ 8 & 1 & 4 & 7 \\ 1 & 3 & 3 & 2 \end{matrix} = \begin{matrix} 9 & 8 \\ 8 & 7 \end{matrix}$$



2) AVERAGE POOLING:-

$$\begin{matrix} 2 & 3 & 2 & 8 \\ 9 & 1 & 3 & 5 \\ 8 & 1 & 4 & 7 \\ 1 & 3 & 3 & 2 \end{matrix} = \begin{matrix} 3.75 & 4.5 \\ 3.25 & 4 \end{matrix}$$

$\text{X} \rightarrow \text{X} \rightarrow \text{X} \rightarrow \text{X}$

* CONVOLUTIONAL NEURAL NETWORK:-

INPUT :
 28×28

$$\left\{ \begin{array}{l} \text{CONV1}(16, 3 \times 3, \text{padding: 'Same'}) : 28 \times 28 \times 16 \\ \text{MAXPOOLING}(2, 2) : 14 \times 14 \times 16 \\ \text{CONV2}(32, 3 \times 3, \text{padding: 'Valid')} : 12 \times 12 \times 32 \\ \text{MAXPOOLING}(2, 2) : 6 \times 6 \times 32 \end{array} \right.$$

$$\left\{ \begin{array}{l} \text{FLATTEN}() : [1152, 1] \\ \text{DENSE}(100) : [100, 1] \\ \dots [10, 1] \end{array} \right.$$

PLACEMENT
 DENSE(100)
 DENSE(10)

: [100,]
 : [10,]

Input 100x100x3

Ex.2 Conv(16, 5x5, Padding = 'Valid')
maxpool(2,2)

O/P 96x96x16 # Parameters 5x5x3x16 + 16

Conv2(32, 3x3, Padding = 'Valid')
maxpool(2,2)

46x46x32 3x3x16x32 + 32

Conv3(64, 3x3, Padding = 'Same')
maxpool(2,2)

23x23x64 3x3x32x64 + 64

Conv4(128, 3x3, Padding = 'Same')
maxpool(2,2)

11x11x128 3x3x64x128 + 128

Floating()

[3200]

Dense(500)

[500]

Dense(100)

[100]

Dense(1)

[1]

O/P

200x200

Ex.3 Conv(32, 7x7, 'valid')
maxpool(2,2)

194x194x32 7x7x32 + 32

97x97x32

Conv2(64, 5x5, 'Same')

97x97x64 5x5x32x64 + 64

maxpool(2,2)

48x48x64

Conv3(128, 3x3, 'Valid')
maxpool(2,2)

46x46x128

3x3x64x128 + 128

23x23x128

Conv4(256, 3x3, 'Valid')
maxpool(2,2)

21x21x256

3x3x128x256 + 256

10x10x256