Break

Convolutional neural network (CNN)

1) What is Convolution?

It is the first Component of CNN

CNN is used mostly when you have data like images, audios, videos.

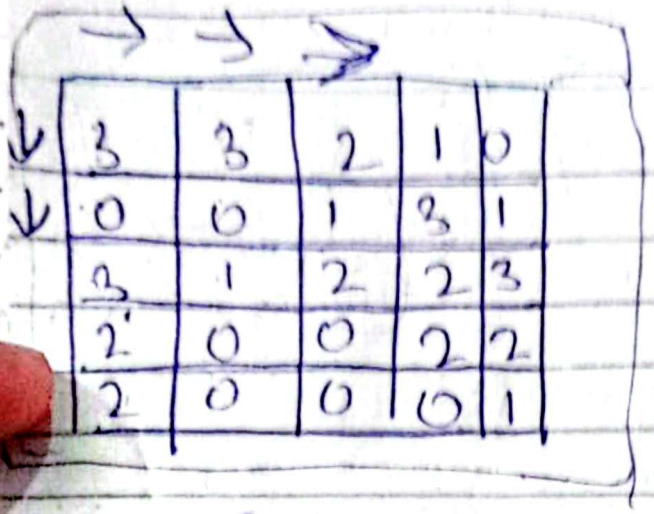
kernal filter filter selector.gd

Kernal is nothing but a filter that is used to extract the features from the images.

This is how CNN provides dimensionality reduction of image while retaining important features.

kernal filter is a matrix that moves over the input data performs dot product with the sub-region of input data and gets the output as the matrix of dotProduct.

kernal moves on input data by a stride value



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

5×5 input image

0	1	2
2	2	0
0	1	2

3×3 kernal filter

1st Convolutional =

$3\times0 + 3\times1 + 2\times2 + 0\times2 + 0\times2 + 1\times0 + 3\times0 + 1\times1 + 2\times2$

$= 0 + 3 + 4 + 0 + 0 + 0 + 0 + 1 + 4$

$3 + 4 + 1 + 4$

$= 12$

Output of 1 Convolutional is 12

12	12	17
10	17	19

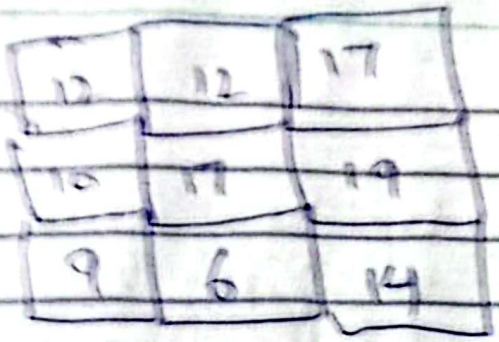
2nd Convolutional

$3\times0 + 2\times1 + 1\times2 + 0\times2 + 1\times2 + 3\times0 + 1\times0 + 2\times1 + 2\times2$

$= 0 + 2 + 2 + 0 + 2 + 0 + 0 + 2 + 4$

$2 + 2 + 2 + 2 + 4$

feature map / Convolative map



It becomes 3x3

formula ⇒ Output = [Size of inpd - Size of kernel] +1

$$O = [i - k] + 1$$

$$= [5 - 3] + 1 \Rightarrow 3$$

5x5 ↑ image

3x3 ↑ kernel

This formula only works if we are using kernal filters.

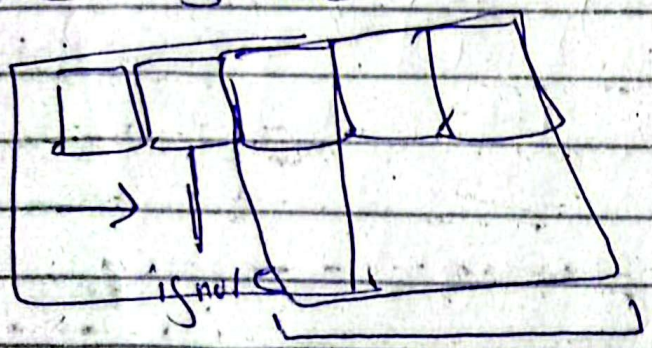
Stride 81

The filter is moved across the image from left to right with one pixel column change on the horizental movements then one pixel row change on the vertical movement

The amount of movement b/w application of filters to the input image is reffered to as stride, and it is almost always symmetrical in height and width dimensions.

The default stride or strides in two dimensions is (1,1) for the height and the width movement

we can we stride =2. means we jump one stride



take a jump then do the

In case of using 2 stride, formula can be derived as

$$\text{Output} = \left[\frac{\text{input image size} - \text{kernal size}}{\text{Stride}}\right] + 1$$

$$O = \left[\frac{5-3}{2}\right] + 1$$

$$O = \left[\frac{2}{2}\right] + 1$$

$$O = 1+1 \Rightarrow \boxed{2} \quad \text{2x2}$$

Padding :-

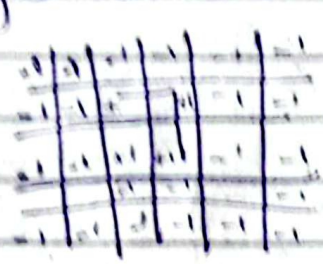
When reducing dimensionality of image by using kernal filter we have to preserve same information as it was present in orignal input image, In this case padding helps us.

Padding is also called as border problem solver.

Convolutional neural networks

An Computer language Computer detect object through edges/shapes

Images have data in pixels which alternatively changes as



① we identify pattern / object / shape / edges as features. and extract them

② we process these features through Relu activation function that converts negative values to positive values

This will decrease the Computational cost and enhance size of object as irrelevent part of image is removed and relevant part is focused.

③ Scaling (Min max scaling) to reduce size of number.

helps dealing with images that have overlapping objects

Image size is decreased so our object with image is focused.

④ Then image is passed to fully connected neural network.

And Softmax activation function is used for classification and prediction that we probabilities. (deals with multiclass classification)

Sigmoid function also do same but for single or 2 class classfiction

Layers

L1 :- Convolutional layer (Kernal filter/feature extraction 1st

L2 :- Activation function (Relu)

L3 :- Pooling layer (helps shrinking the size of image.

L4 :- Information passed to fully connected neural network (Making Decision)

2nd activation function.

Total CNN has 2 activation function. Normally 2nd one can be changed but Relu in most cases is used.

helps

Activation function ~~helps~~ focuses on relevant part

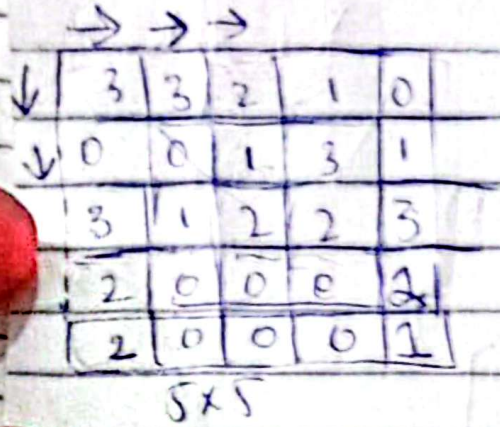
Pooling layer ~~Summarizing~~ is just like summarizing the whole thing into shorter version

Daily life example of CNN layers

Pooling layer creates pooling so important features are not lost.

ex 2

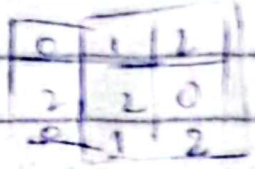
stride = 1



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

5×5

filter



0	1	2
2	2	0
0	1	2

3×3

→ represents black
→ represents white

Dot Product will give result as convolutional map or feature map with same size as filter kernel's size

$3\times0 + 3\times1 + 2\times2$
$+ 0\times2 + 0\times2 + 1\times0 = 12$
$+ 3\times0 + 1\times1 + 2\times2$

12	12	17
10	17	19
9	6	14

$3 \times 0 + 2 \times 1 + 1 \times 2$

$+ 0 \times 2 + 1 \times 2 + 3 \times 0 = 12$

$1 \times 0 + 2 \times 1 + 2 \times 2$

$2 \times 0 + 1 \times 1 + 0 \times 2$

$+ 1 \times 2 + 3 \times 2 + 1 \times 0 = 17$

$+ 2 \times 0 + 2 \times 1 + 3 \times 2$

$0 \times 0 + 0 \times 1 + 1 \times 2$

$+ 3 \times 2 + 1 \times 2 + 2 \times 0 = 10$

$+ 2 \times 0 + 0 \times 1 + 0 \times 2$

$0 \times 0 + 1 \times 1 + 3 \times 2$

$+ 1 \times 2 + 2 \times 2 + 2 \times 0 = 17$

$+ 0 \times 0 + 0 \times 1 + 0 \times 2$

To find Output's no of feature map how many rows, column will be

formula

$$[\text{Size of input} - \text{Size of kernal}] + 1$$

$$[5 - 3] + 1$$

$$= 3$$

$$\Rightarrow 3 \times 3$$

with value of stride $= 2$, remember that you will give jump of 2 after applying first kernal.

Means you always start convolutunal layer with first row and first column then after it you will give jump.

For Stride = 2

formula

$$\left[\frac{\text{Size of input} - \text{Size of kernal}}{\text{stride}}\right] + 1$$

$$\frac{5-3}{2} + 1 \quad = 2$$

12	17
9	14

Output
2×2

The value of stride depends on number of objects to identify in image

For detection of 1 object to clearly capture it use stride = 1

Use stride 2 if you are fetching ~~multiple~~ objects
↓
single

out of multiple objects

Padding

As we have seen the edges are not given much importance in convolution layer
so this is called Border problem
we will apply border problem solver / Padding:

Padding 1D

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

Input 5x5 becomes 7x7

Now we will again apply kernel

0	1	2
2	2	0
0	1	2

0x0 + 0x1 + 0x2
+ 0x2 + 3x2 + 3x0 = 6
+ 0x0 + 0x1 + 0x2

6	14	17	3

$0\times0 + 0\times1 + 0\times2$

$+ 3\times2 + 3\times2 + 2\times0 = 14$

$+ 0\times0 + 0\times1 + 2\times1$

$0\times0 + 0\times1 + 0\times2$

$+ 1\times2 + 0\times2 + 0\times0 = 3$

$+ 3\times0 + 1\times1 + 0\times2$

$0\times0 + 1\times0 + 2\times0$

$3\times2 + 2\times2 + 1\times0 = 17$

$0\times0 + 1\times1 + 2\times3$

6	14	17	11	3
14	12	12	17	11
8	10	17	17	13
11	9	6	14	12
6	4	4	6	4

Now we will apply pooling layer which help in down sampling.

Selecting relevent features

1) Max Pooling
2) Average pooling
3) Min pooling.

If we apply Max Pooling with 2x2

14	17
11	19

=> Out put will be same as 2x2

we select 2x2 matrix from result output and apply max pooling where we select the max number in 2x2

example

6	14	17	11	3
14	12	12	14	11
8	18	17	19	13
11	9	6	14	12
6	4	4	6	4

3) Column vector
4) which kernel should be used

2 x 1

14	17
11	19

Max pooling

Average pooling takes average,

Min pooling selects min value.

(4) After pooling we will apply flatten layer which convert dimension to 1D

we basically convert into column vector

14	17
11	19

will be converted to column vector

$$\Rightarrow \begin{bmatrix} 14 \\ 17 \\ 11 \\ 19 \end{bmatrix}$$