Tutorial-5

Understanding CNN and Implementation

Day-5 : Deep Learning

ANN → Theoretical, Practical

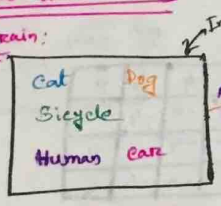
CNN → convolutional neural network es Images, video frames
↓
Image classification, object detection

Agenda:

1. CNN vs Human brain
2. Convolution Operations
   - Convolution
   - padding (prevent to loss img size. If my img size is decreasing we are loosing some info. In order to prevent loss we padding)
   - Strides (moves one cell to other cell is called strides) & stride=1 (jump 1 cell, stride = 2 jump 2 cell.)
   - Filters (kernels) (helps to steack information from image)
3. Max Pooling
4. Flattening (As like as ANN Dense)
5. Practical Implementation (CNN)

1. CNN vs Human Brain

Human Brain:



As soon as passes input through each layer

cereberal cortex
↓
Visual Cortex (Identifies specific image)
Layers
V_1 → understand → human object
V_2 → Animal
V_3 → Map the environment
V_4 V_5 V_6
V_7 → Visual cortex
O/P

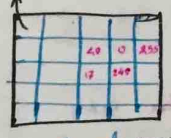
CNN:

We can devide in different stage

1. Convolution

Images → 1 channel ← images are Black & white or RGB

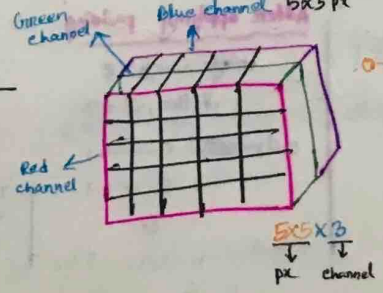
0-255



0 = Black
255 = white color

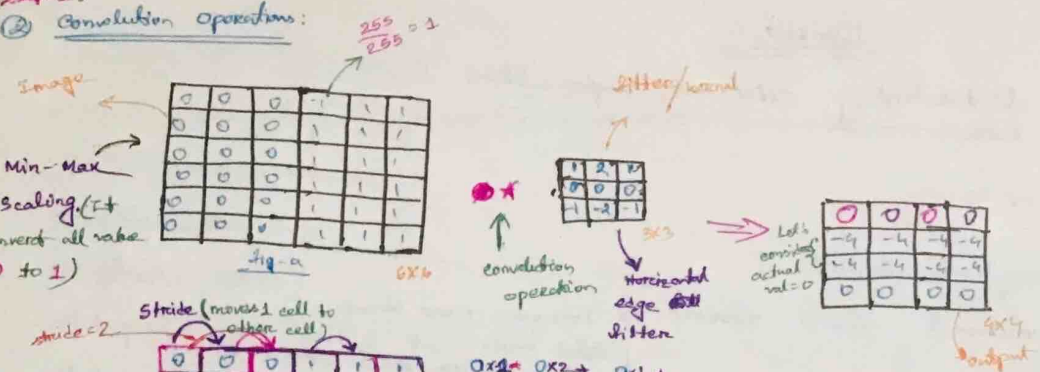
RGB

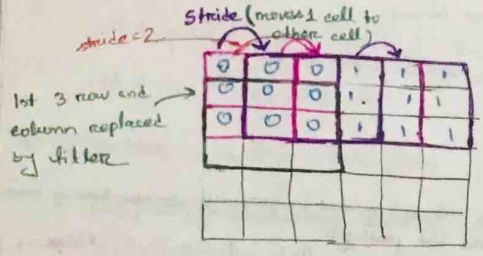
Total channel = 3



step-1:

② Convolution operations:





0x1 + 0x2 + 0x1 +
0x0 + 0x0 + 0x0 +
0x-1 + 0x-2 + 0x-1 =
0+0+0+0+0+0
+0+0+0 = 0

stage-2:

0+0+1+0+0+0+0+0-1 = 0

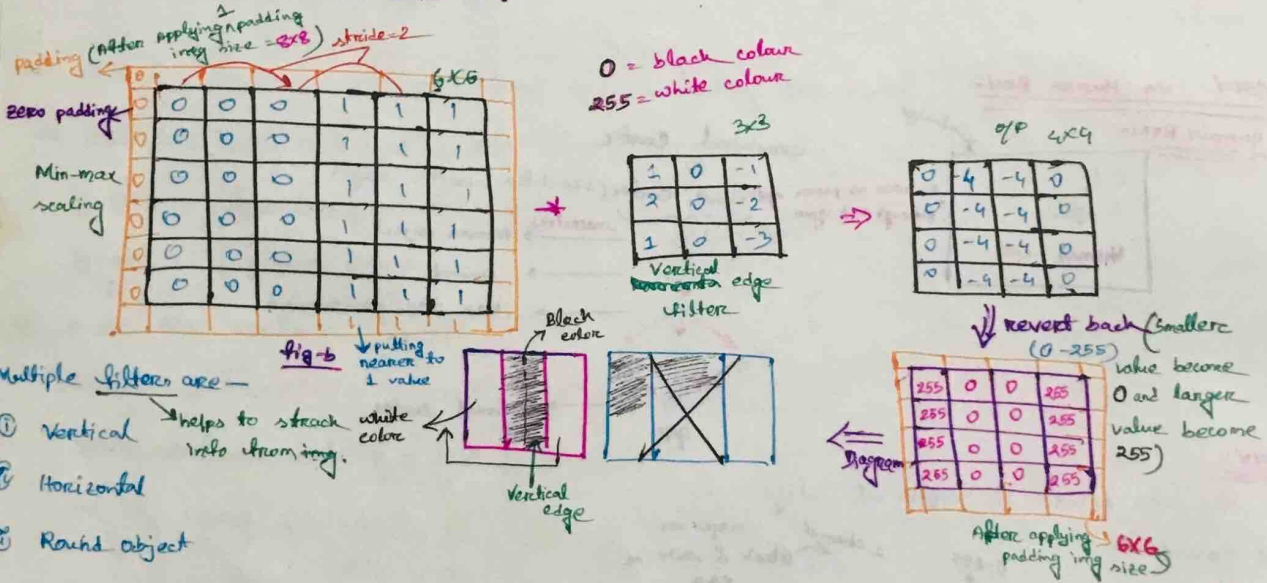
stage-3:

0+2+1+0+0+0+0-2-1 = 0

stage-4:

1+2+1+0+0+0+0-2-2-1 = 0

Never hard coded the filter value. We should update it during b/w propagation. Here we activation function ReLU{max(0, x)} or other activation func.



Multiple filters are —

① Vertical → helps to detect edge in the img.

② Horizontal

③ Round object

For different filter, we will get different types of output.

When passes 6X6 image in a 3X3 filter, output is 4X4

Mathematical formula:

image size, n = 6, p = padding = 1
filter, f = 3, s = stride = 1

output = n-f+1
= 6-3+1
= 3+1
= 4

Output = n-f+1

output = (n+2p-f+1)/s → updating formula

After applying padding

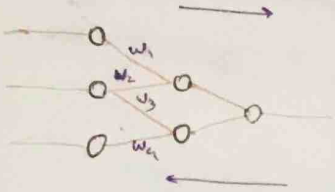
image size, n = 8
filter, f = 3
output = n-f+1
= 8-3+1
= 6

output = 8+2x1-3+1
= 6

n = 8
f = 3
p = 1

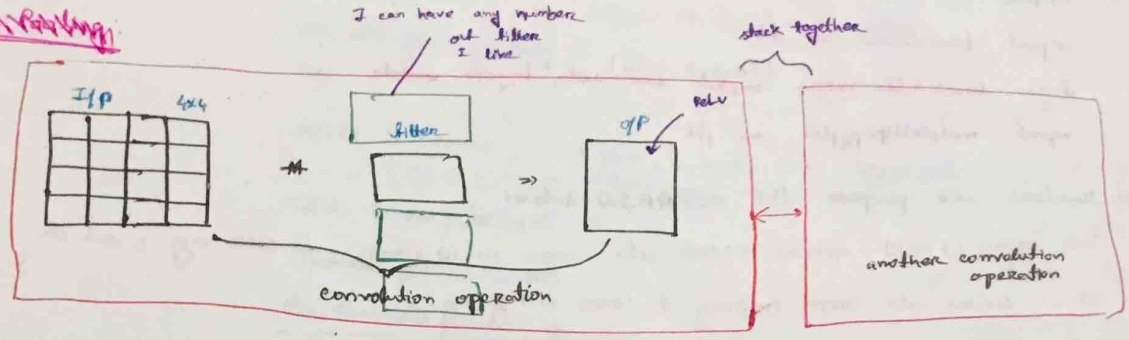
Padding:

If my img size is decreasing, we loose some info to prevent loss we use padding. It means protecting the image adding another layer top of it. What types of value comes inside padding, called? For this diff types of padding. One is 0 padding, nearer the value padding put there.



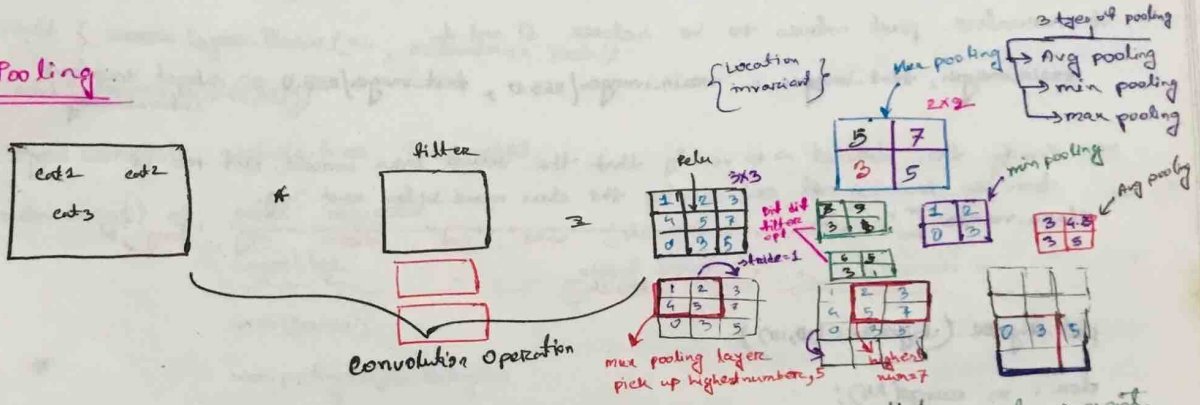
next ds: Transfer learning

② Max Pooling



Convolution operation => the main aim is that we need to learn the filters, we need to learn and update best on the input images

③ Max Pooling



The concept is called location invariant, location invariant says that objects present anywhere but as we pass from many convolution layers I should be able to extract the info in a better and clear way. And for that we specially use something called as Max pooling. It extracts clear info.

3 types of pooling :
- Max (pick highest value from the filter)
- Min (" min " " " " ")
- Avg (Summation all value and provides avg value)

④ Flatening Layer (helps pooling converts pooling into elongate input)

As like as ANN Dense.