## horizontal line

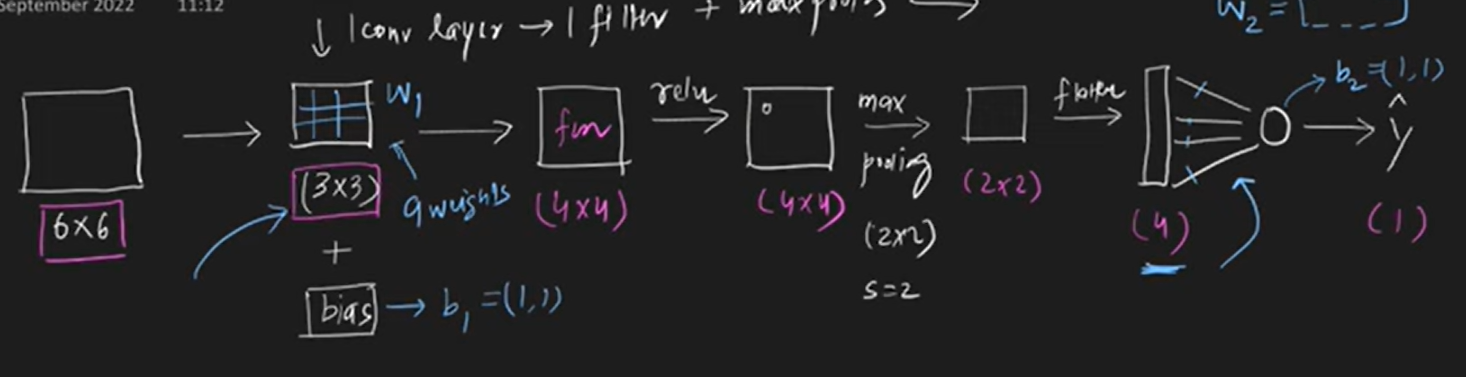
Backpropagation in CNN

15.04.2025

**─**

# Backpropagation in CNN

We are taking a simple setup

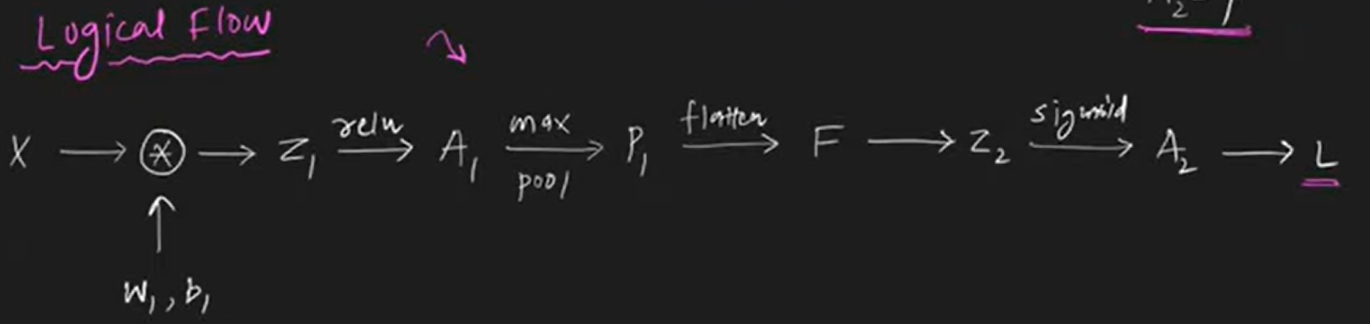


Trainable parameters :

W1 = (3,3) W2= (1,4)

b1 = (1,1) b2 = (1,1) total 15 trainable parameters

The logical flow :



There are the forward propagation equations :

Z1 = Conv(X,W1) + b1 F = flatten(P1)

A1 = relu(Z1) Z2 = W2F + b2

P1 = maxpool(A1) A2 = (Z2)

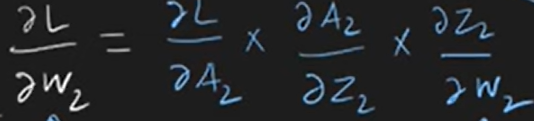
We need to calculate the four gradients here remember all are matrices :

W1 = W1 - () W2 = W2 - ()

b1 = b1 - () b2 = b2 - ()

Dividing it into 2 parts ANN and CNN , First calculating gradients of ANN (W2 , b2) :

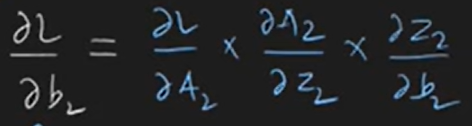
## = (A2 - Y) F’



Shape of the gradient will be same as shape of W2 i.e.

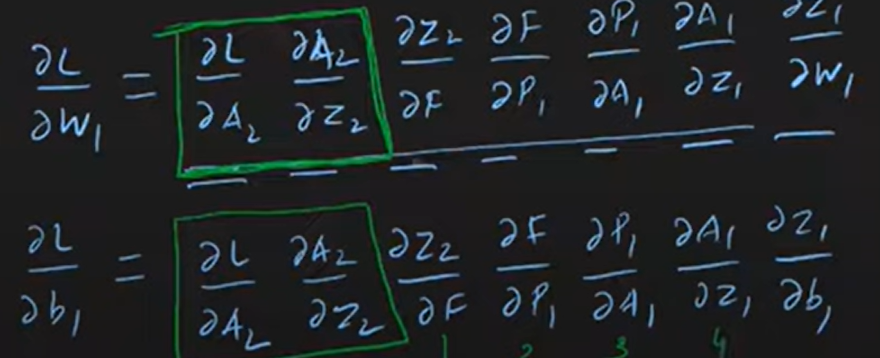
((1,1) - (1,1)) \* (1,4) = (1,1) \* (1,4) = (1,4)

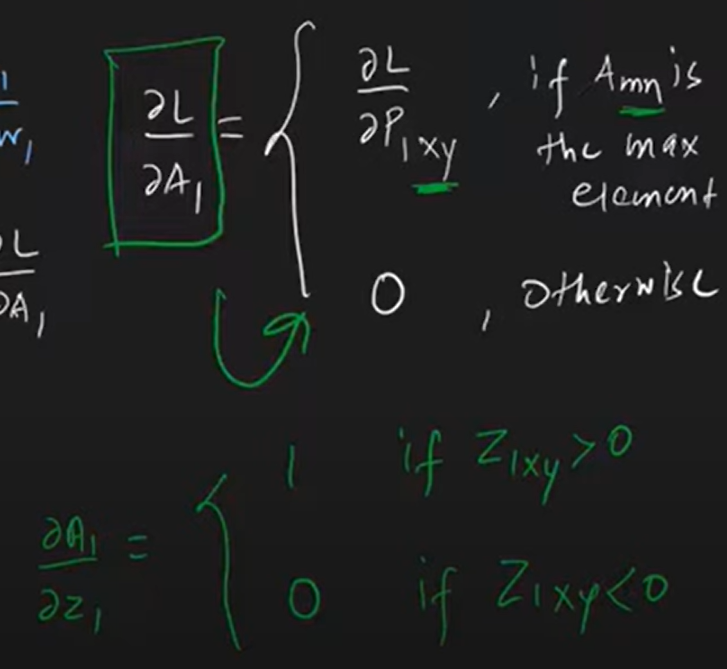
## = (A2 - Y)



Similarly here shape is ((1,1) - (1,1)) = (1,1)

For gradients calculation in CNN :



Calculating the parts ,

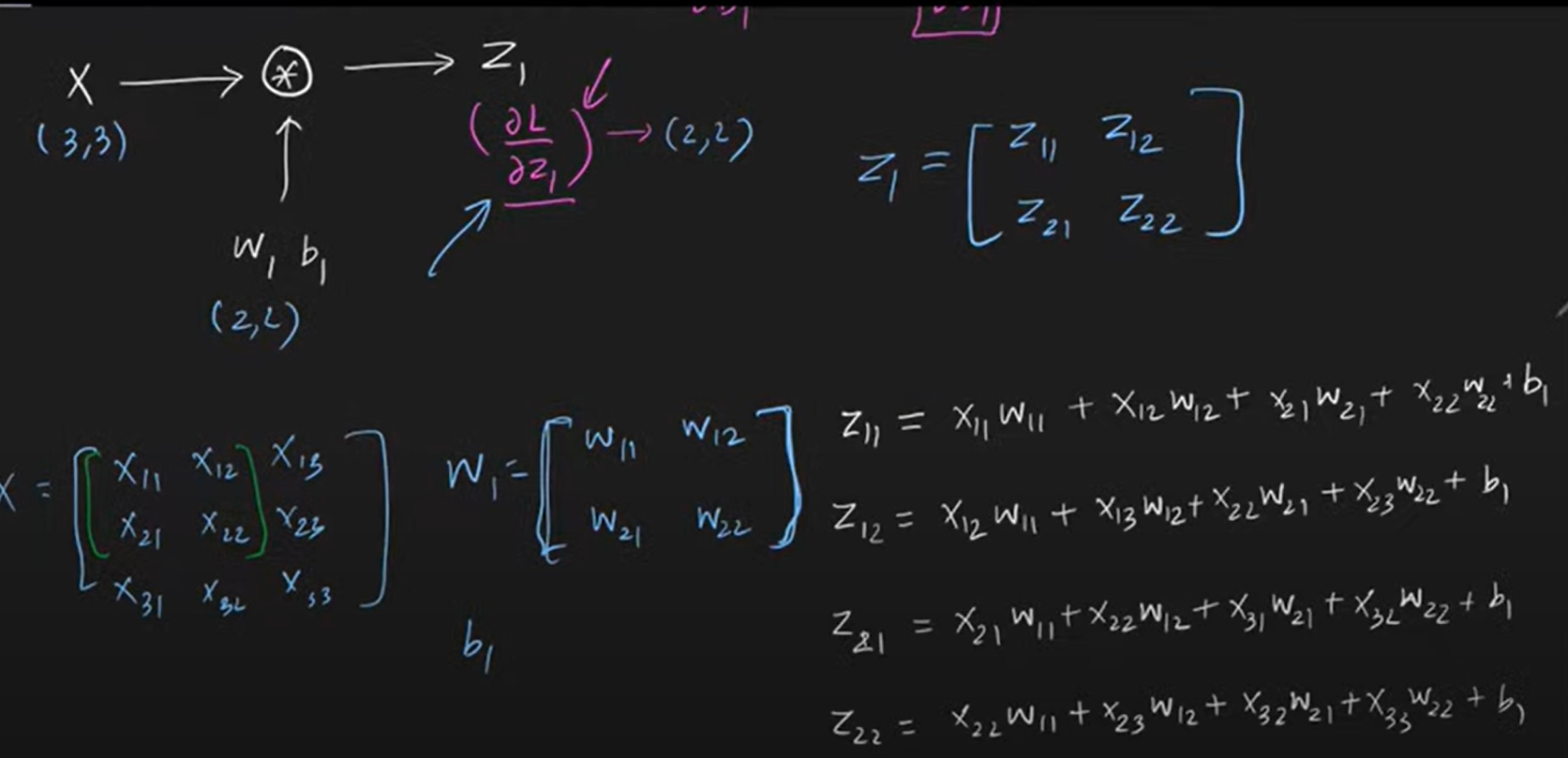
= W2

In pooling and flatten layers there are no trainable patterns, so just reshaping them in reverse

= reshape(p1.shape)

= converting (2X2) to (4X4) , inserting 0s

= differentiation of relu (1 or 0)



## = sum()

## = Conv(X, )

