Centro Brasileiro de Pesquisas Físicas

Redes Neurais profundas e aplicações Deep Learning

Clécio Roque De Bom – debom@cbpf.br

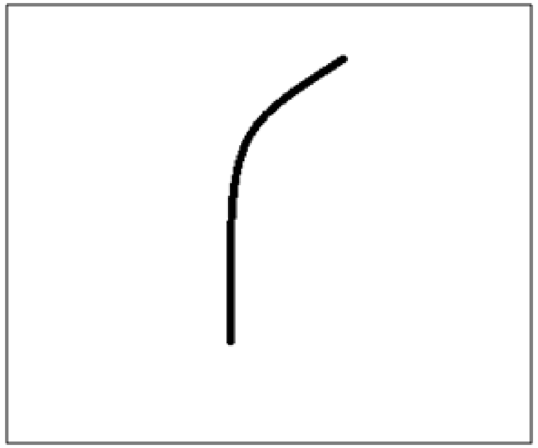
clearnightsrthebest.com

CBPF

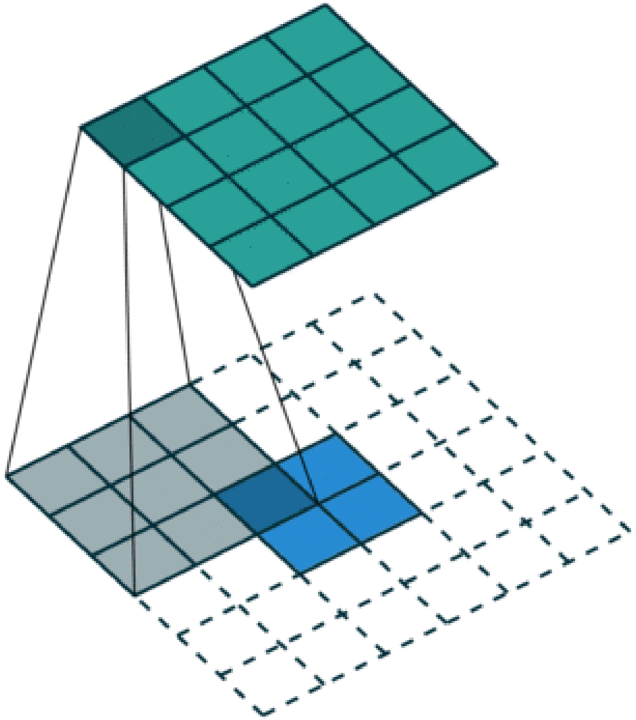
Conv Layers

0	0	0	0	0	30	0
0	0	0	0	30	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	0	0	0	0

Pixel representation of filter

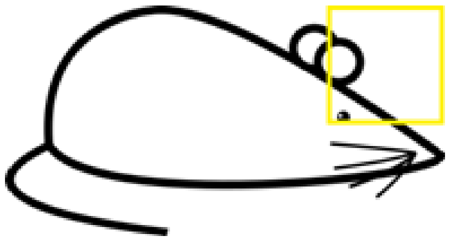


Visualization of a curve detector filter



https://adeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks/

Conv Layers



Visualization of the filter on the image

0	0	0	0	0	0	0
0	40	0	0	0	0	0
40	0	40	0	0	0	0
40	20	0	0	0	0	0
0	50	0	0	0	0	0
0	0	50	0	0	0	0
25	25	0	50	0	0	0

Pixel representation of receptive field

*

0	0	0	0	0	30	0
0	0	0	0	30	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	0	0	0	0

Pixel representation of filter

Multiplication and Summation = 0

Visualization of the receptive field

0	0	0	0	0	0	30
0	0	0	0	50	50	50
0	0	0	20	50	0	0
0	0	0	50	50	0	0
0	0	0	50	50	0	0
0	0	0	50	50	0	0
0	0	0	50	50	0	0

Pixel representation of the receptive field

*

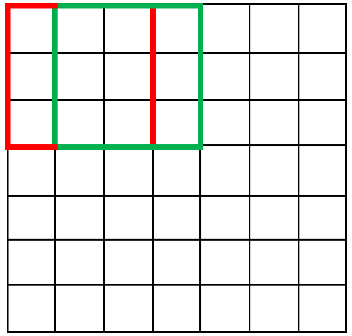
0	0	0	0	0	30	0
0	0	0	0	30	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	0	0	0	0

Pixel representation of filter

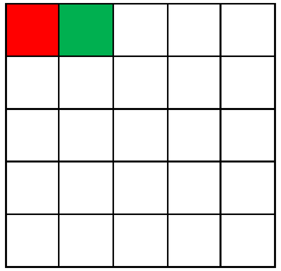
Multiplication and Summation = (50*30)+(50*30)+(50*30)+(20*30)+(50*30) = 6600 (A large number!)

Stride and Padding

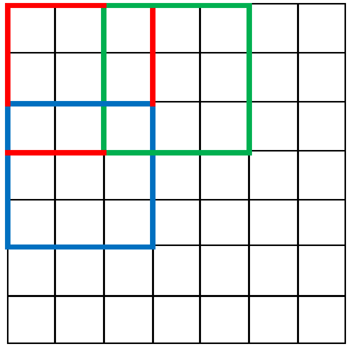
7 x 7 Input Volume



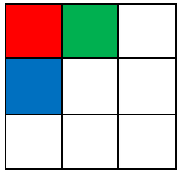
5 x 5 Output Volume

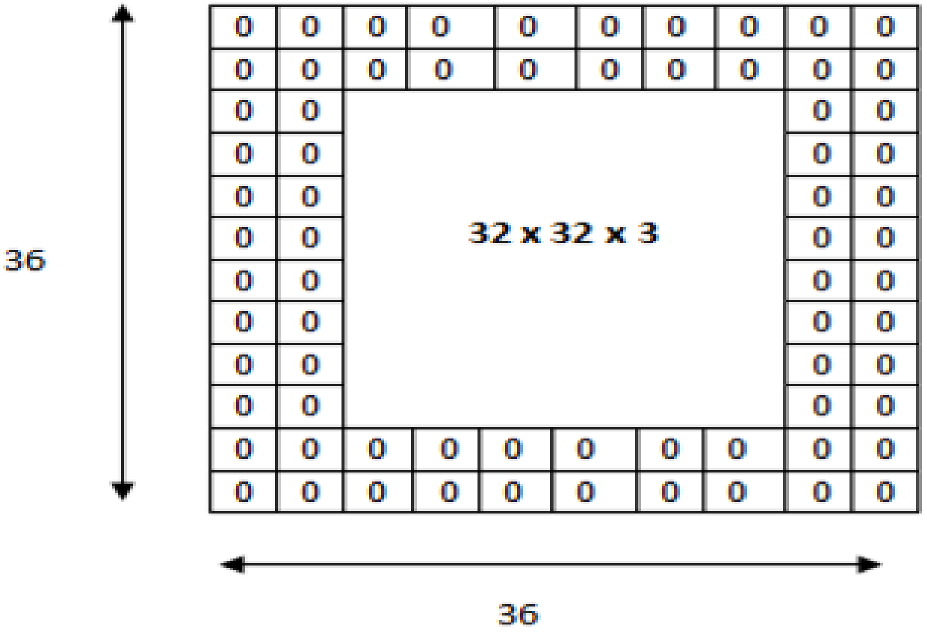


7 x 7 Input Volume

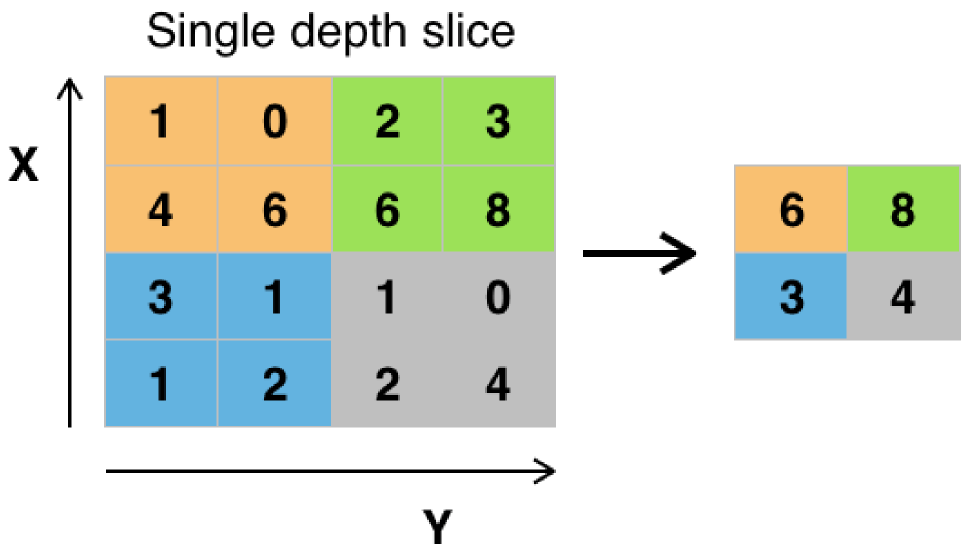


3 x 3 Output Volume

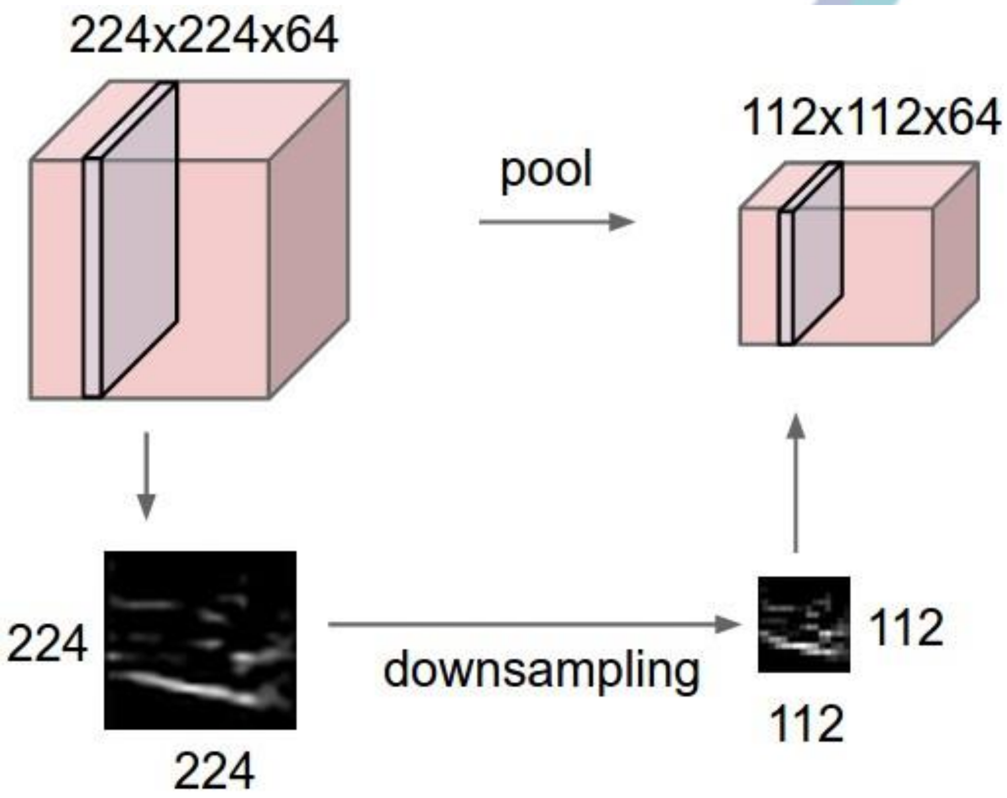




Pooling

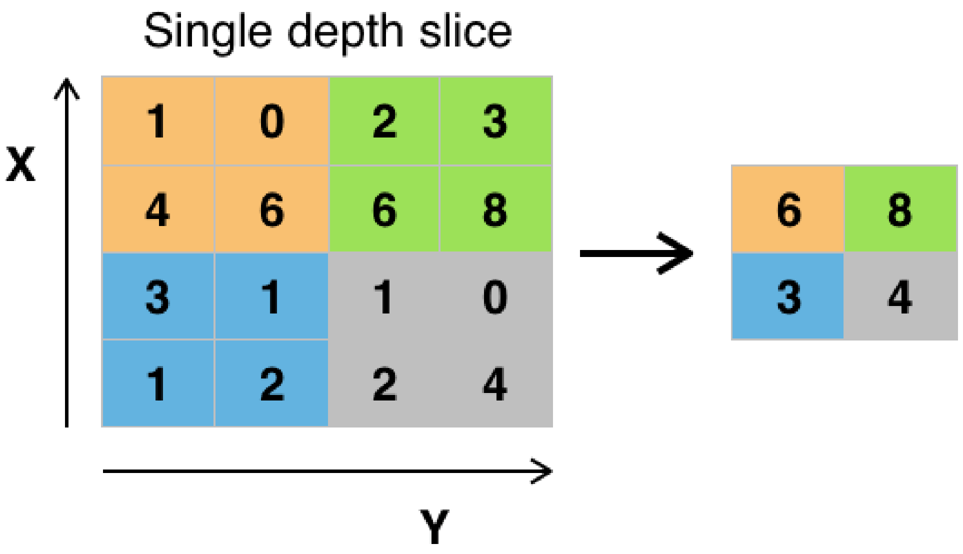


Example of Maxpool with a 2x2 filter and a stride of 2

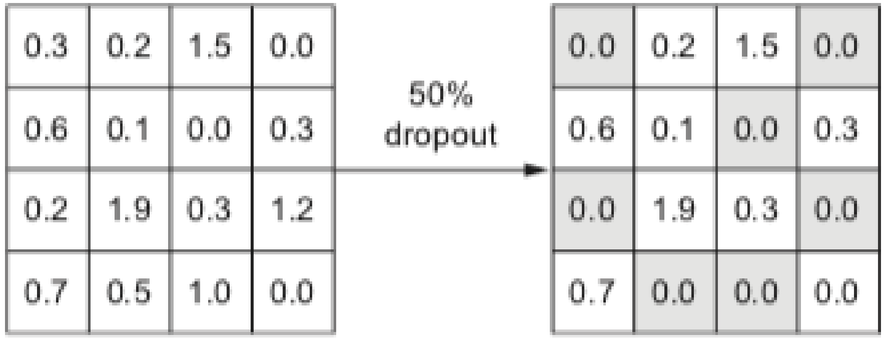


Adapted from : https://ml-cheatsheet.readthedocs.io/en/latest/layers.html

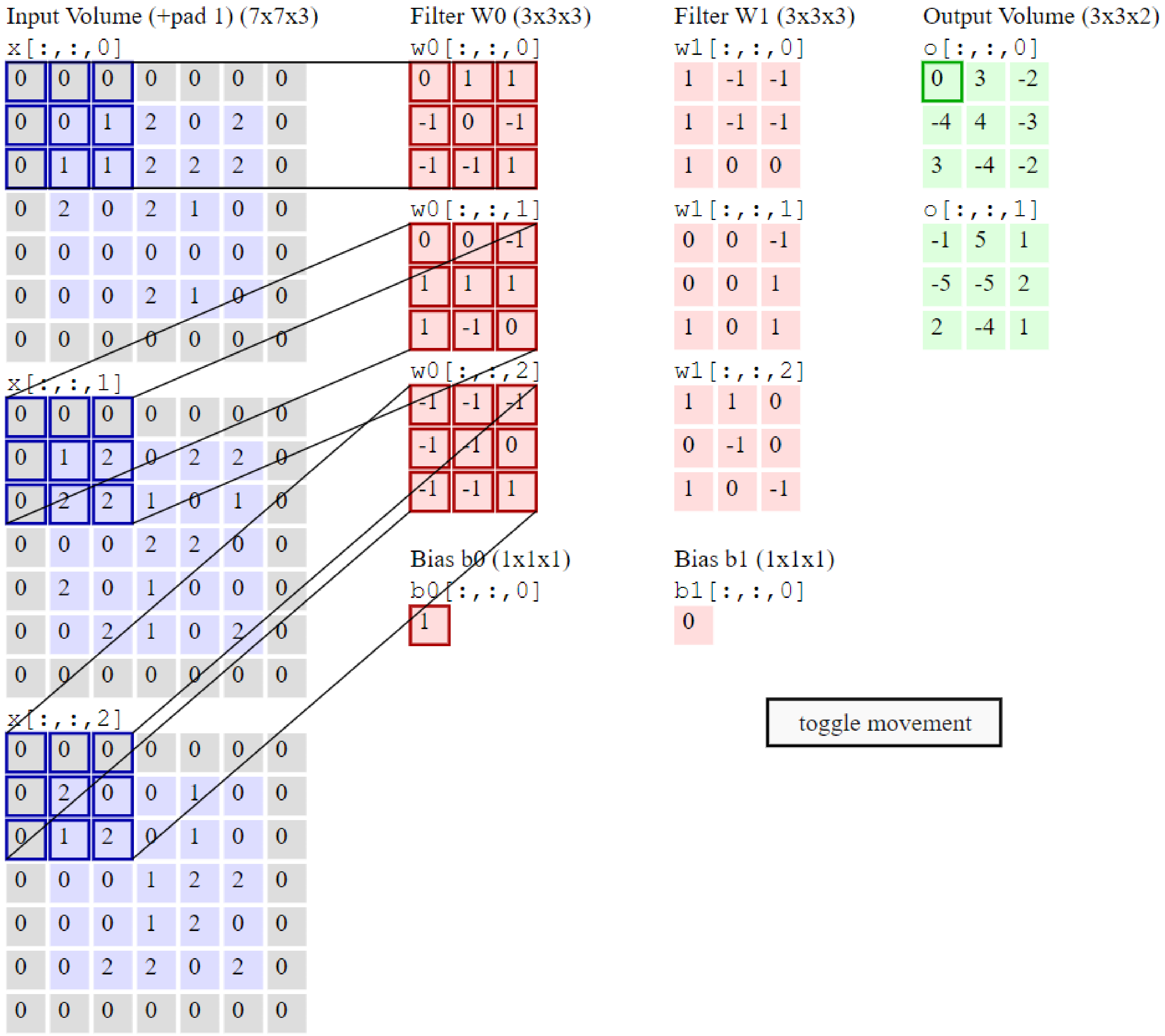
Dropout and Pooling

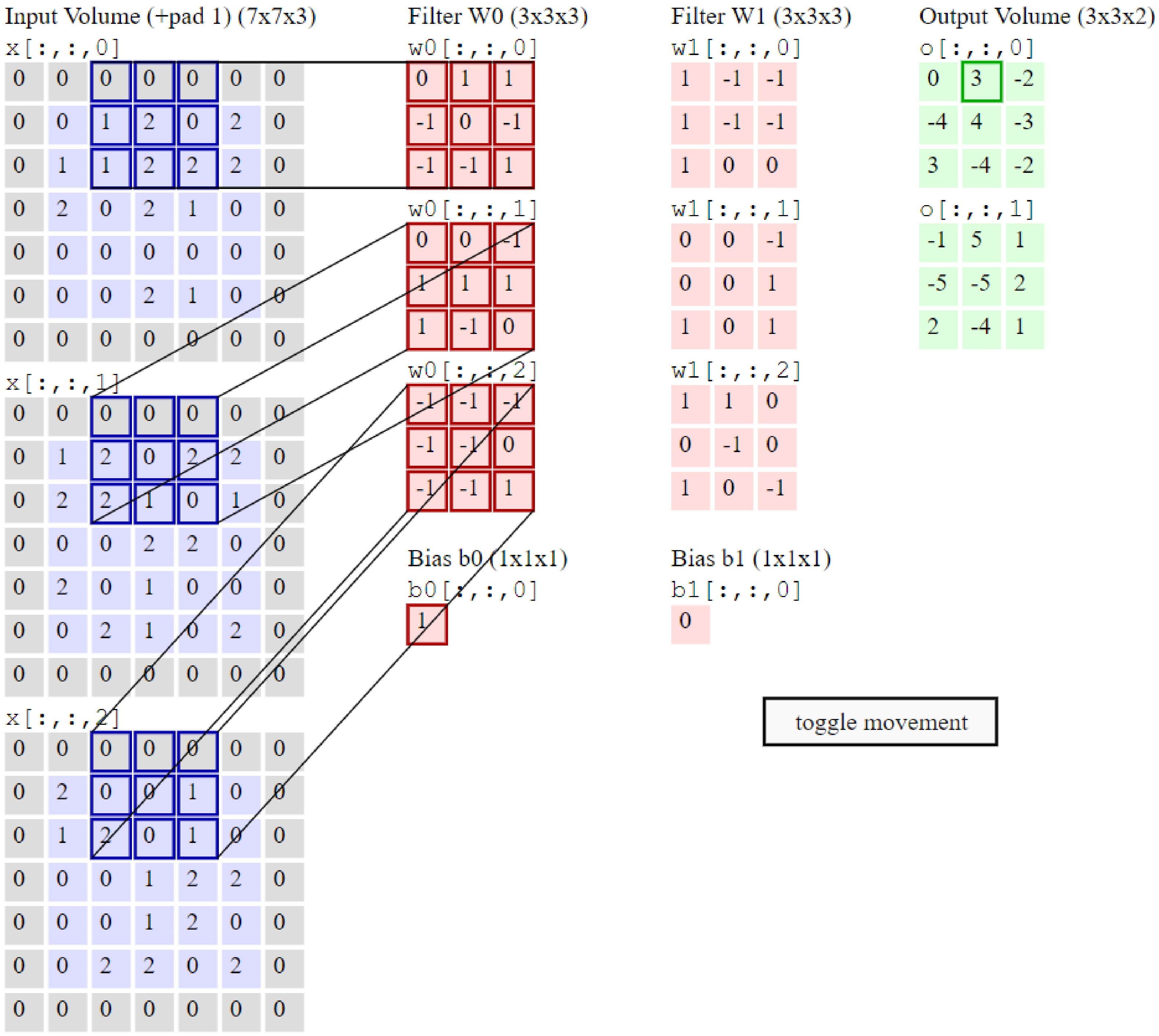


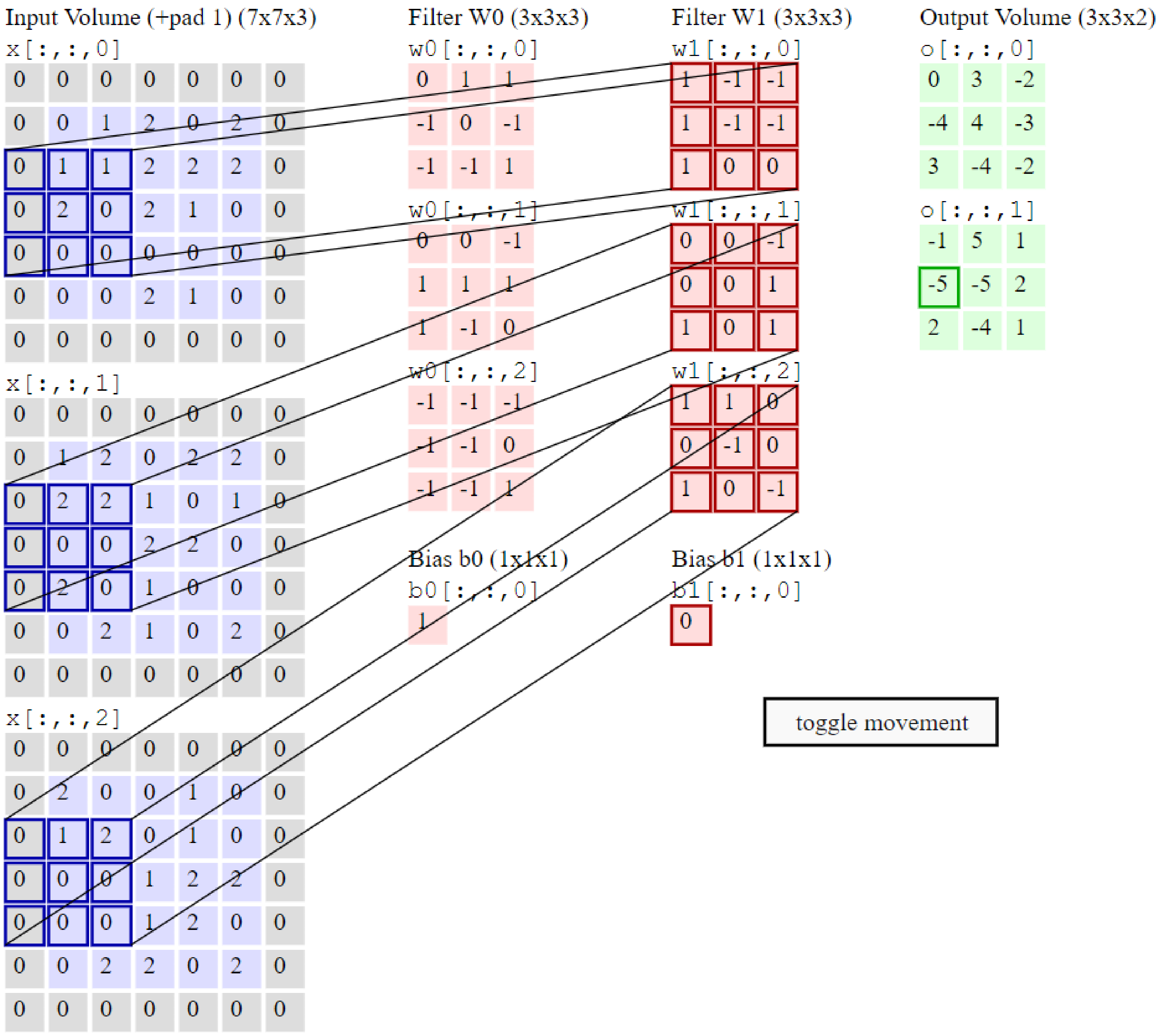
Example of Maxpool with a 2x2 filter and a stride of 2

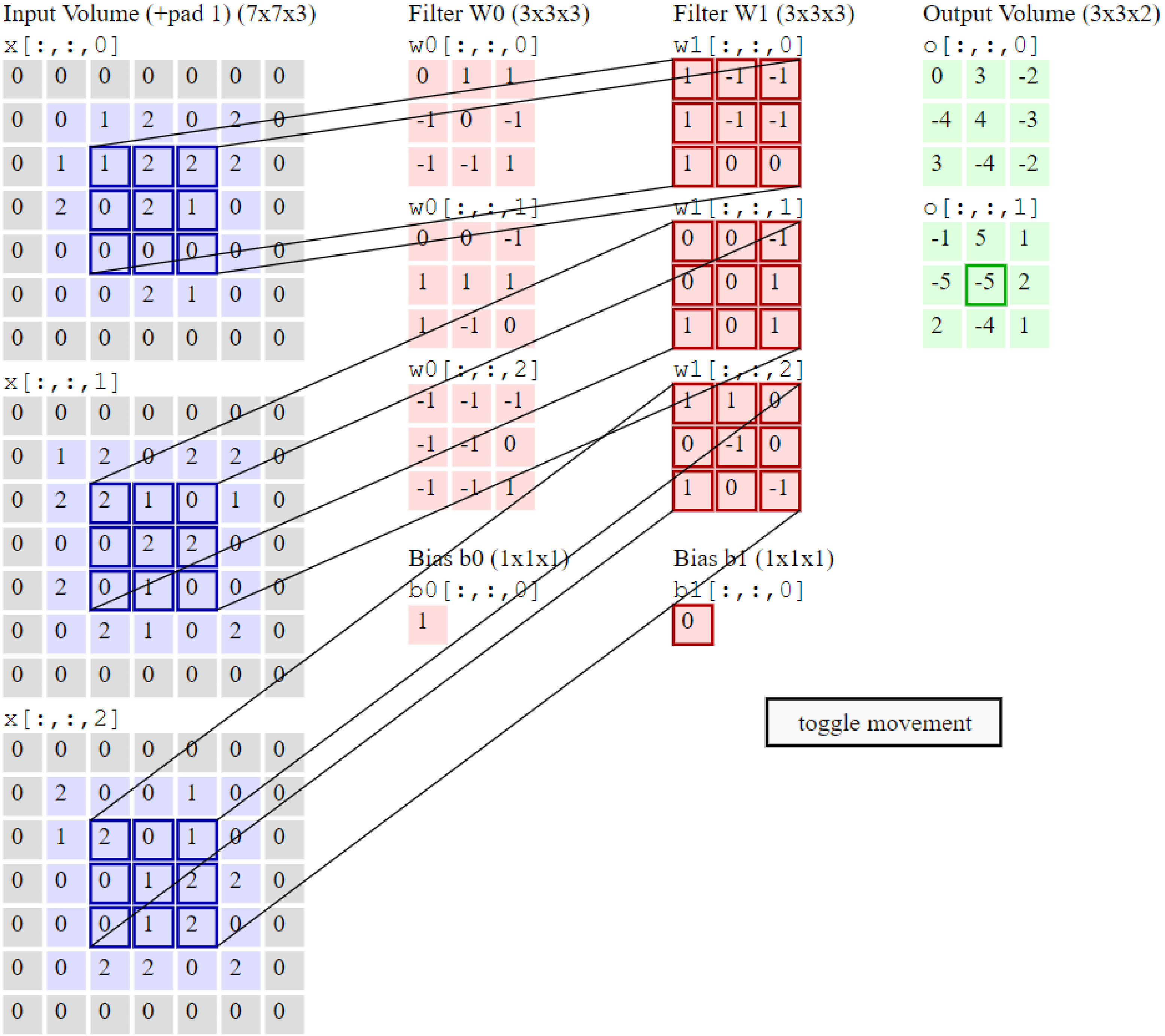


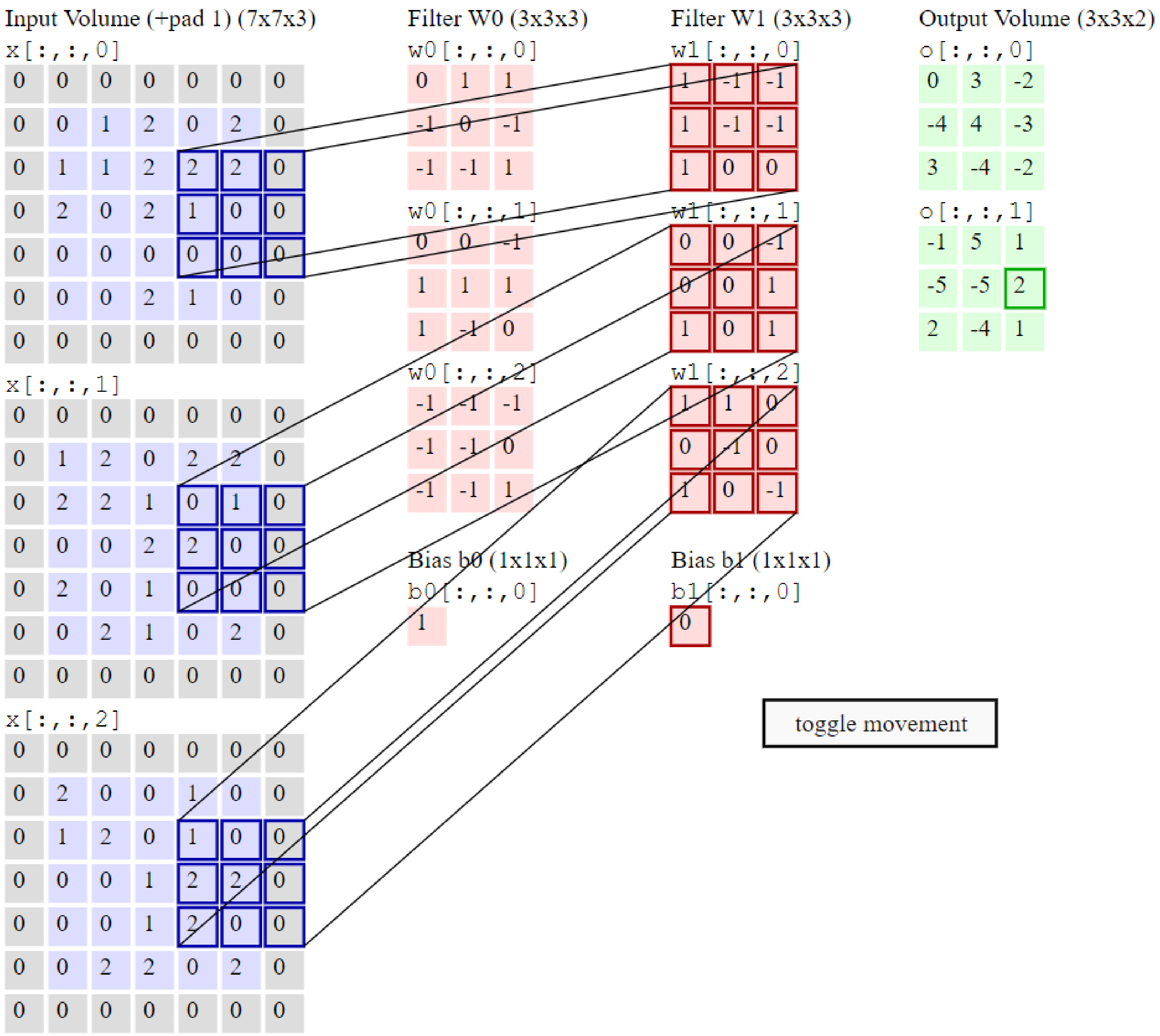
Adapted from : https://ml-cheatsheet.readthedocs.io/en/latest/layers.html


Input Volume (+pad 1) (7x7x3)
x[:,:,0]
0 0 0 0 0 0 0
0 0 1 2 0 2 0
0 1 1 2 2 2 0
0 2 0 2 1 0 0
0 0 0 0 0 0 0
0 0 0 2 1 0 0
0 0 0 0 0 0 0
x[:,:,1]
0 0 0 0 0 0 0
0 1 2 0 2 2 0
0 2 2 1 0 1 0
0 0 0 2 2 0 0
0 2 0 1 0 0 0
0 0 2 1 0 2 0
0 0 0 0 0 0 0
x[:,:,2]
0 0 0 0 0 0 0
0 2 0 0 1 0 0
0 1 2 0 1 0 0
0 0 0 1 2 2 0
0 0 0 1 2 0 0
0 0 2 2 0 2 0
0 0 0 0 0 0 0
Filter W0 (3x3x3)
w0[:,:,0]
0 1 1
-1 0 -1
-1 -1 1
w0[:,:,1]
0 0 -1
1 1 1
1 -1 0
w0[:,:,2]
-1 -1 -1
-1 -1 0
-1 -1 1
Bias b0 (1x1x1)
b0[:,:,0]
1
Filter W1 (3x3x3)
w1[:,:,0]
1 -1 -1
1 -1 -1
1 0 0
w1[:,:,1]
0 0 -1
0 0 1
1 0 1
w1[:,:,2]
1 1 0
0 -1 0
1 0 -1
Bias b1 (1x1x1)
b1[:,:,0]
0
Output Volume (3x3x2)
o[:,:,0]
0 3 -2
-4 4 -3
3 -4 -2
o[:,:,1]
-1 5 1
-5 -5 2
2 -4 1
toggle movement


Input Volume (+pad 1) (7x7x3)
x[:,:,0]
0 0 0 0 0 0 0
0 0 1 2 0 2 0
0 1 1 2 2 2 0
0 2 0 2 1 0 0
0 0 0 0 0 0 0
0 0 0 2 1 0 0
0 0 0 0 0 0 0
x[:,:,1]
0 0 0 0 0 0 0
0 1 2 0 2 2 0
0 2 2 1 0 1 0
0 0 0 2 2 0 0
0 2 0 1 0 0 0
0 0 2 1 0 2 0
0 0 0 0 0 0 0
x[:,:,2]
0 0 0 0 0 0 0
0 2 0 0 1 0 0
0 1 2 0 1 0 0
0 0 0 1 2 2 0
0 0 0 1 2 0 0
0 0 2 2 0 2 0
0 0 0 0 0 0 0
Filter W0 (3x3x3)
w0[:,:,0]
0 1 1
-1 0 -1
-1 -1 1
w0[:,:,1]
0 0 -1
1 1 1
1 -1 0
w0[:,:,2]
-1 -1 -1
-1 -1 0
-1 -1 1
Bias b0 (1x1x1)
b0[:,:,0]
1
Filter W1 (3x3x3)
w1[:,:,0]
1 -1 -1
1 -1 -1
1 0 0
w1[:,:,1]
0 0 -1
0 0 1
1 0 1
w1[:,:,2]
1 1 0
0 -1 0
1 0 -1
Bias b1 (1x1x1)
b1[:,:,0]
0
Output Volume (3x3x2)
o[:,:,0]
0 3 -2
-4 4 -3
3 -4 -2
o[:,:,1]
-1 5 1
-5 -5 2
2 -4 1
toggle movement


Input Volume (+pad 1) (7x7x3)
x[:,:,0]
0 0 0 0 0 0 0
0 0 1 2 0 2 0
0 1 1 2 2 2 0
0 2 0 2 1 0 0
0 0 0 0 0 0 0
0 0 0 2 1 0 0
0 0 0 0 0 0 0
x[:,:,1]
0 0 0 0 0 0 0
0 1 2 0 2 2 0
0 2 2 1 0 1 0
0 0 0 2 2 0 0
0 2 0 1 0 0 0
0 0 2 1 0 2 0
0 0 0 0 0 0 0
x[:,:,2]
0 0 0 0 0 0 0
0 2 0 0 1 0 0
0 1 2 0 1 0 0
0 0 0 1 2 2 0
0 0 0 1 2 0 0
0 0 2 2 0 2 0
0 0 0 0 0 0 0
Filter W0 (3x3x3)
w0[:,:,0]
0 1 1
-1 0 -1
-1 -1 1
w0[:,:,1]
0 0 -1
1 1 1
1 -1 0
w0[:,:,2]
-1 -1 -1
-1 -1 0
-1 -1 1
Bias b0 (1x1x1)
b0[:,:,0]
1
Filter W1 (3x3x3)
w1[:,:,0]
1 -1 -1
1 -1 -1
1 0 0
w1[:,:,1]
0 0 -1
0 0 1
1 0 1
w1[:,:,2]
1 1 0
0 -1 0
1 0 -1
Bias b1 (1x1x1)
b1[:,:,0]
0
Output Volume (3x3x2)
o[:,:,0]
0 3 -2
-4 4 -3
3 -4 -2
o[:,:,1]
-1 5 1
-5 -5 2
2 -4 1
toggle movement


Input Volume (+pad 1) (7x7x3)
x[:,:,0]
0 0 0 0 0 0 0
0 0 1 2 0 2 0
0 1 1 2 2 2 0
0 2 0 2 1 0 0
0 0 0 0 0 0 0
0 0 0 2 1 0 0
0 0 0 0 0 0 0
x[:,:,1]
0 0 0 0 0 0 0
0 1 2 0 2 2 0
0 2 2 1 0 1 0
0 0 0 2 2 0 0
0 2 0 1 0 0 0
0 0 2 1 0 2 0
0 0 0 0 0 0 0
x[:,:,2]
0 0 0 0 0 0 0
0 2 0 0 1 0 0
0 1 2 0 1 0 0
0 0 0 1 2 2 0
0 0 0 1 2 0 0
0 0 2 2 0 2 0
0 0 0 0 0 0 0
Filter W0 (3x3x3)
w0[:,:,0]
0 1 1
-1 0 -1
-1 -1 1
w0[:,:,1]
0 0 -1
1 1 1
1 -1 0
w0[:,:,2]
-1 -1 -1
-1 -1 0
-1 -1 1
Bias b0 (1x1x1)
b0[:,:,0]
1
Filter W1 (3x3x3)
w1[:,:,0]
1 -1 -1
1 -1 -1
1 0 0
w1[:,:,1]
0 0 -1
0 0 1
1 0 1
w1[:,:,2]
1 1 0
0 -1 0
1 0 -1
Bias b1 (1x1x1)
b1[:,:,0]
0
Output Volume (3x3x2)
o[:,:,0]
0 3 -2
-4 4 -3
3 -4 -2
o[:,:,1]
-1 5 1
-5 -5 2
2 -4 1
toggle movement


Input Volume (+pad 1) (7x7x3)
x[:,:,0]
0 0 0 0 0 0 0
0 0 1 2 0 2 0
0 1 1 2 2 2 0
0 2 0 2 1 0 0
0 0 0 0 0 0 0
0 0 0 2 1 0 0
0 0 0 0 0 0 0
x[:,:,1]
0 0 0 0 0 0 0
0 1 2 0 2 2 0
0 2 2 1 0 1 0
0 0 0 2 2 0 0
0 2 0 1 0 0 0
0 0 2 1 0 2 0
0 0 0 0 0 0 0
x[:,:,2]
0 0 0 0 0 0 0
0 2 0 0 1 0 0
0 1 2 0 1 0 0
0 0 0 1 2 2 0
0 0 0 1 2 0 0
0 0 2 2 0 2 0
0 0 0 0 0 0 0
Filter W0 (3x3x3)
w0[:,:,0]
0 1 1
-1 0 -1
-1 -1 1
w0[:,:,1]
0 0 -1
1 1 1
1 -1 0
w0[:,:,2]
-1 -1 -1
-1 -1 0
-1 -1 1
Bias b0 (1x1x1)
b0[:,:,0]
1
Filter W1 (3x3x3)
w1[:,:,0]
1 -1 -1
1 -1 -1
1 0 0
w1[:,:,1]
0 0 -1
0 0 1
1 0 1
w1[:,:,2]
1 1 0
0 -1 0
1 0 -1
Bias b1 (1x1x1)
b1[:,:,0]
0
Output Volume (3x3x2)
o[:,:,0]
0 3 -2
-4 4 -3
3 -4 -2
o[:,:,1]
-1 5 1
-5 -5 2
2 -4 1
toggle movement

Summary. To summarize, the Conv Layer:

- Accepts a volume of size $W_1 \times H_1 \times D_1$
- Requires four hyperparameters:
    - Number of filters K,
    - their spatial extent F,
    - the stride S,
    - the amount of zero padding P.
- Produces a volume of size $W_2 \times H_2 \times D_2$ where:
    - $W_2 = (W_1 - F + 2P)/S + 1$
    - $H_2 = (H_1 - F + 2P)/S + 1$ (i.e. width and height are computed equally by symmetry)
    - $D_2 = K$
- With parameter sharing, it introduces $F \cdot F \cdot D_1$ weights per filter, for a total of $(F \cdot F \cdot D_1) \cdot K$ weights and K biases.
- In the output volume, the d-th depth slice (of size $W_2 \times H_2$) is the result of performing a valid convolution of the d-th filter over the input volume with a stride of S, and then offset by d-th bias.

Ministério da Ciência, Tecnologia e Inovações

Centro Brasileiro de Pesquisas Físicas

Redes Neurais profundas e aplicações Deep Learning

Clécio Roque De Bom – debom@cbpf.br

clearnightsrthebest.com

CBPF