ALIZADEH Charly

Convolutional Neural Networks

Summary

1. Convolution
    a. Kernel convolution
    b. Convolutional layer
2. Pooling
    a. Concept
    b. Exemple

Convolution

Kernel convolution

Intuition

Original image

0	0	0	0	0	0	0	0
0	0	1	0	0	1	0	0
0	0	1	0	0	1	0	0
0	0	1	0	0	1	0	0
0	0	0	0	0	0	0	0
0	1	0	0	0	0	1	0
0	0	1	1	1	1	0	0
0	0	0	0	0	0	0	0

Pixel values

0	0	0
0	0	1
0	0	1

0	0	0
0	1	0
0	1	0

...

0	0	1
0	0	1
0	0	1

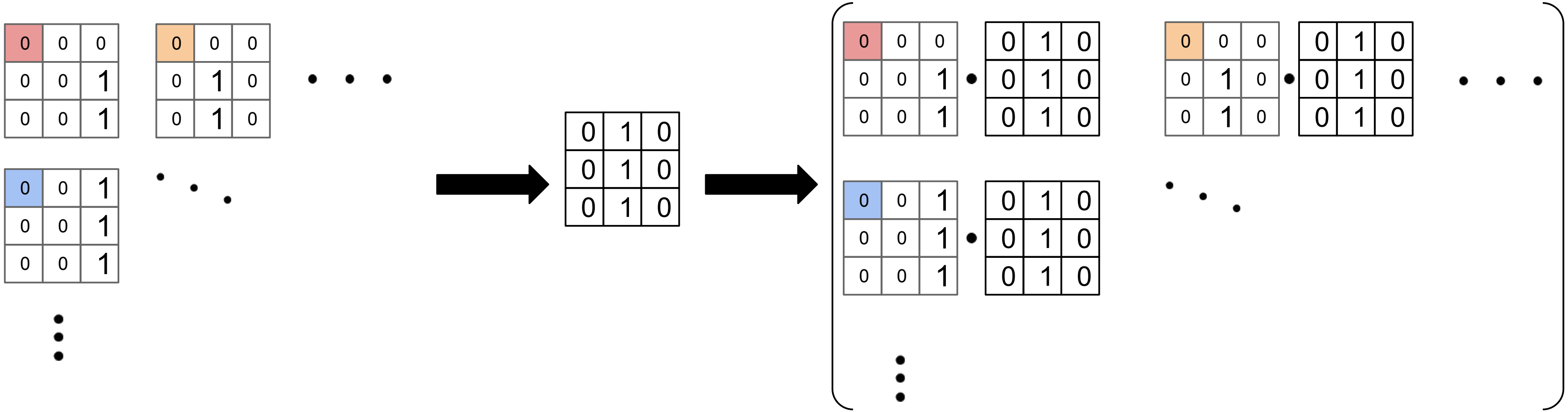
...

Subpart of the image multiplied by the kernel

ALIZADEH Charly

Kernel convolution

Intuition



Subpart of the image

Kernel

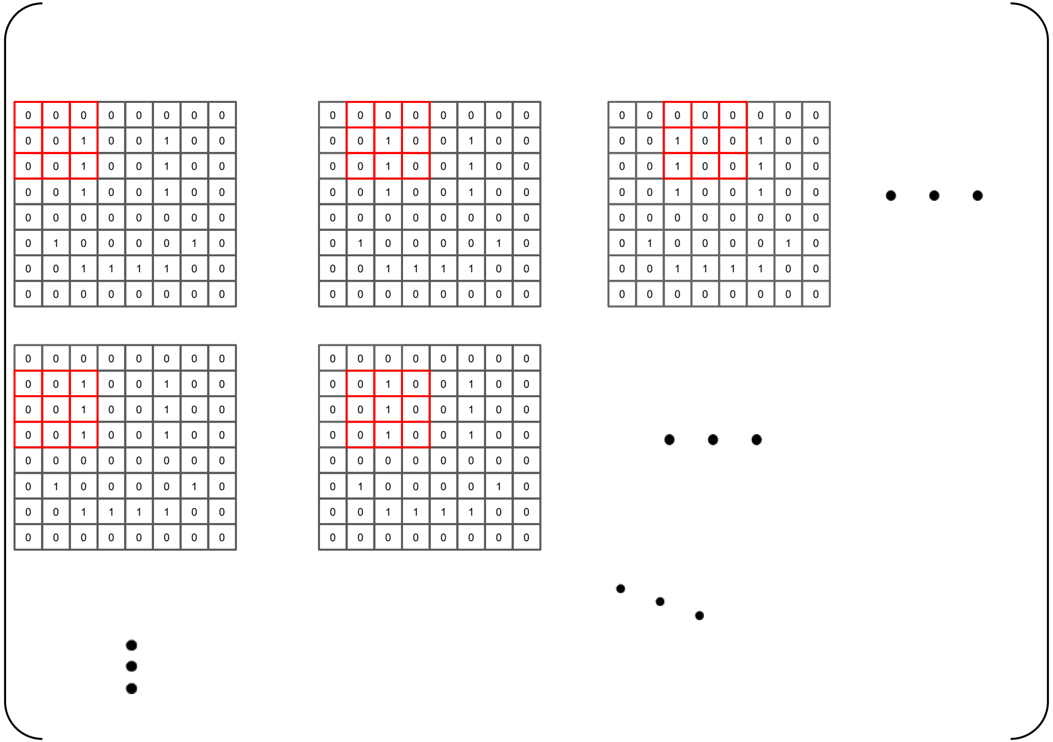
Dot product each subpart of the image with the kernel (also called filter).
Each product gives a scalar, all those scalars give use a new matrix which will be the input of the next layer (after applying an activation function)

ALIZADEH Charly

Kernel convolution

Intuition

$$\begin{bmatrix} 0&0&0&0&0&0&0&0 \\ 0&0&1&0&0&1&0&0 \\ 0&0&1&0&0&1&0&0 \\ 0&0&1&0&0&1&0&0 \\ 0&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&1&0 \\ 0&0&1&1&1&1&0&0 \\ 0&0&0&0&0&0&0&0 \end{bmatrix} \bullet \begin{bmatrix} 0&1&0 \\ 0&1&0 \\ 0&1&0 \end{bmatrix} \Rightarrow$$



Each red highlighted submatrix represents the part that is being multiplied by the kernel, the result being a matrix.

ALIZADEH Charly

Kernel convolution

Mathematical expression

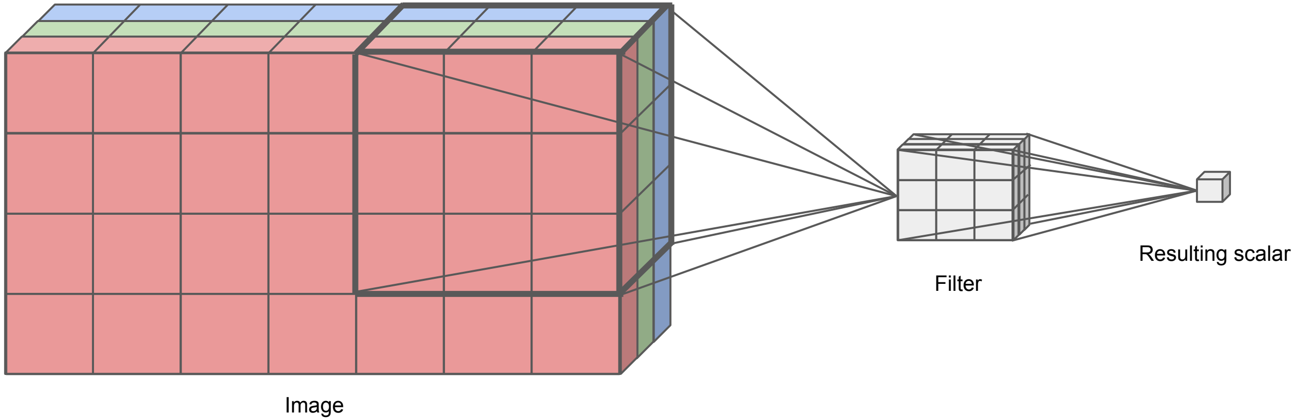
Let P be the matrix containing the image pixels and K the kernel matrix.
We can express the value (i,j) in the matrix resulting of the convolution of P by K thanks to the following formula:

$$Conv_{i,j}(P, K) = \langle P[i, i+k; j, j+k], K \rangle_F$$

Where $\langle \rangle_F$ represents the *Frobenius inner product* and P[a,b;c,d] represents the submatrix of P composed of the intersections of the rows in [a,b] and the columns in [c,d].

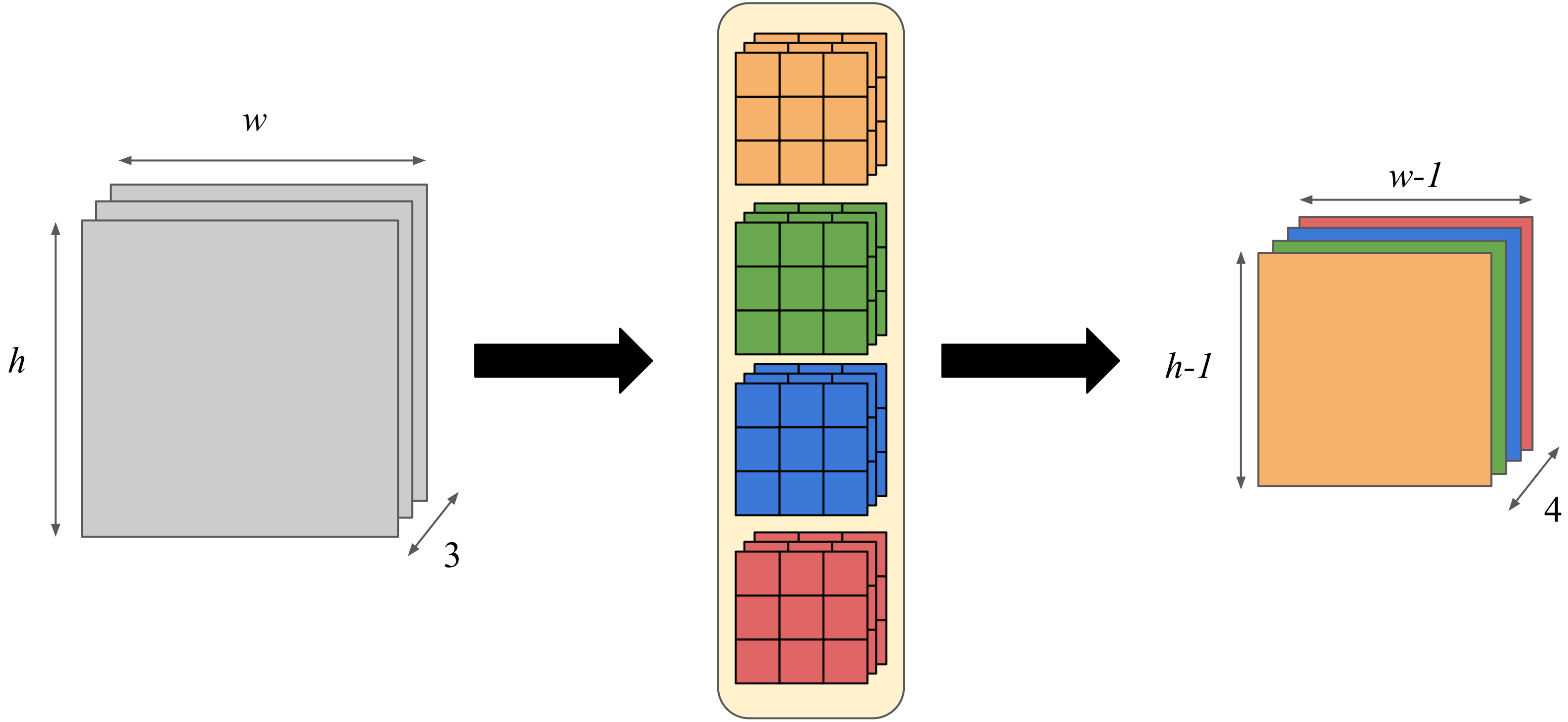
Convolutional layer

A convolutional layer is a set of kernels. Such a layer takes as input a 3-dimensional array (width, height, depth) and output another 3-dimensional array resulting of the convolution of the input with all the kernel.



Result of applying a kernel of dimension 3x3 (extended to 3x3x3 for the dot product) on a subpart of the image. This operation is done for all the image resulting in a 2-dimensional array, and for all the filters resulting in a 3-dimensional array

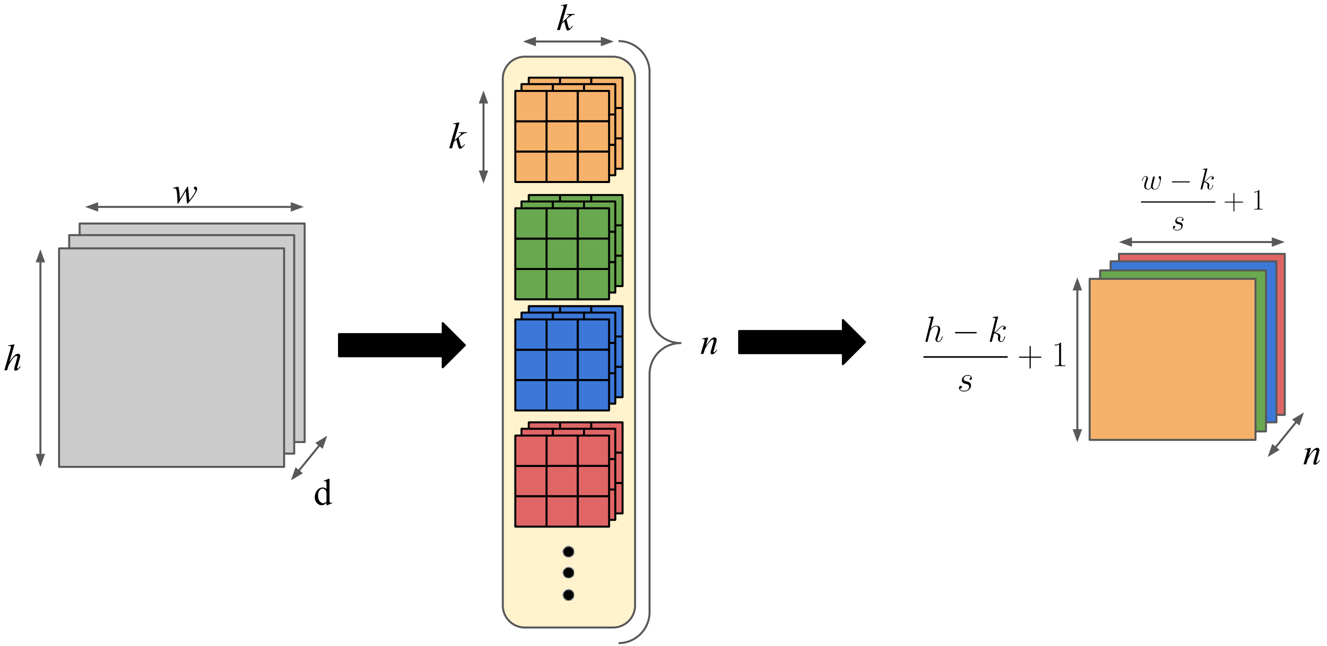
Convolutional layer



Example of convolutional layer with a stride of 1

Convolutional layer

Let P be a matrix of dimension w x h x d, C be a set of n filter each with a dimension of k x k and s be the stride of those filters.
Then the dimension of the output is $$\left(\frac{w-k}{s}+1, \frac{h-k}{s}+1, n\right)$$



Pooling

ALIZADEH Charly

Concepts

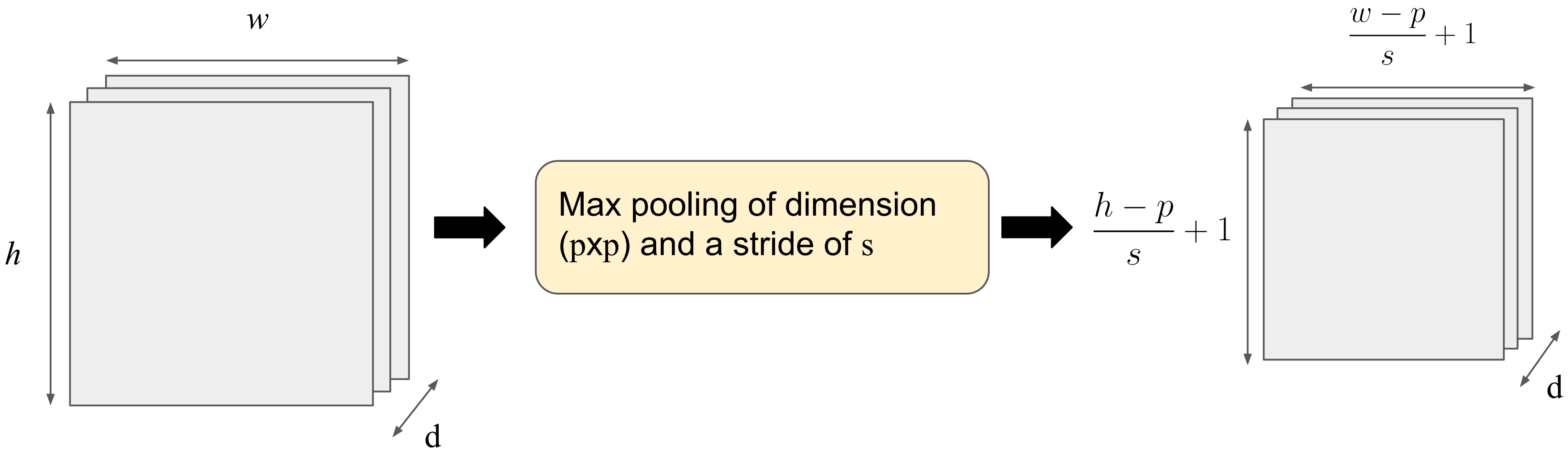
A pooling layer that extract information about a submatrix. For a exemple a max pooling layer will take the maximum value of all the designated submatrix and build a new matrix from it. This pooling is applied on each channel of the 3-dimensional array. So a pooling layer takes as input a 3-dimensional array and output another 3-dimensional with a (possible) different width and height but same depth.

The dimension of the output is: $(\frac{w-p}{s}+1, \frac{h-p}{s}+1, d)$

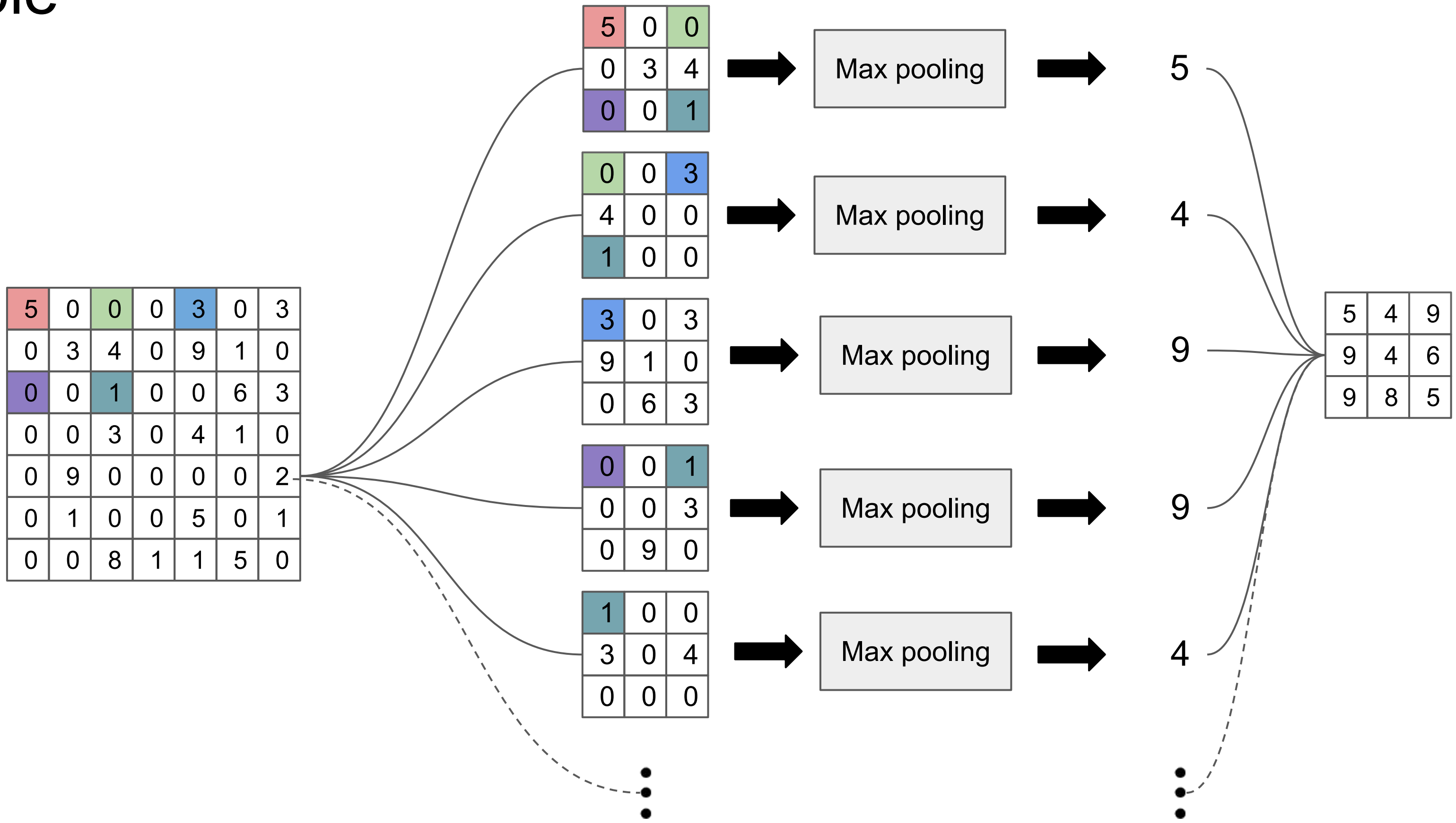
with:

$$w \rightarrow width\ of\ the\ input$$
$$h \rightarrow height\ of\ the\ input$$
$$(p \times p) \rightarrow dimensions\ of\ the\ pooling$$
$$s \rightarrow stride\ of\ the\ pooling$$

Concepts


w
h
d
Max pooling of dimension (pxp) and a stride of s
$\frac{h-p}{s}+1$
$\frac{w-p}{s}+1$
d

Example


5 0 0 0 3 0 3
0 3 4 0 9 1 0
0 0 1 0 0 6 3
0 0 3 0 4 1 0
0 9 0 0 0 0 2
0 1 0 0 5 0 1
0 0 8 1 1 5 0
5 0 0
0 3 4
0 0 1
0 0 3
4 0 0
1 0 0
3 0 3
9 1 0
0 6 3
0 0 1
0 0 3
0 9 0
1 0 0
3 0 4
0 0 0
Max pooling
Max pooling
Max pooling
Max pooling
Max pooling
5
4
9
9
4
5 4 9
9 4 6
9 8 5