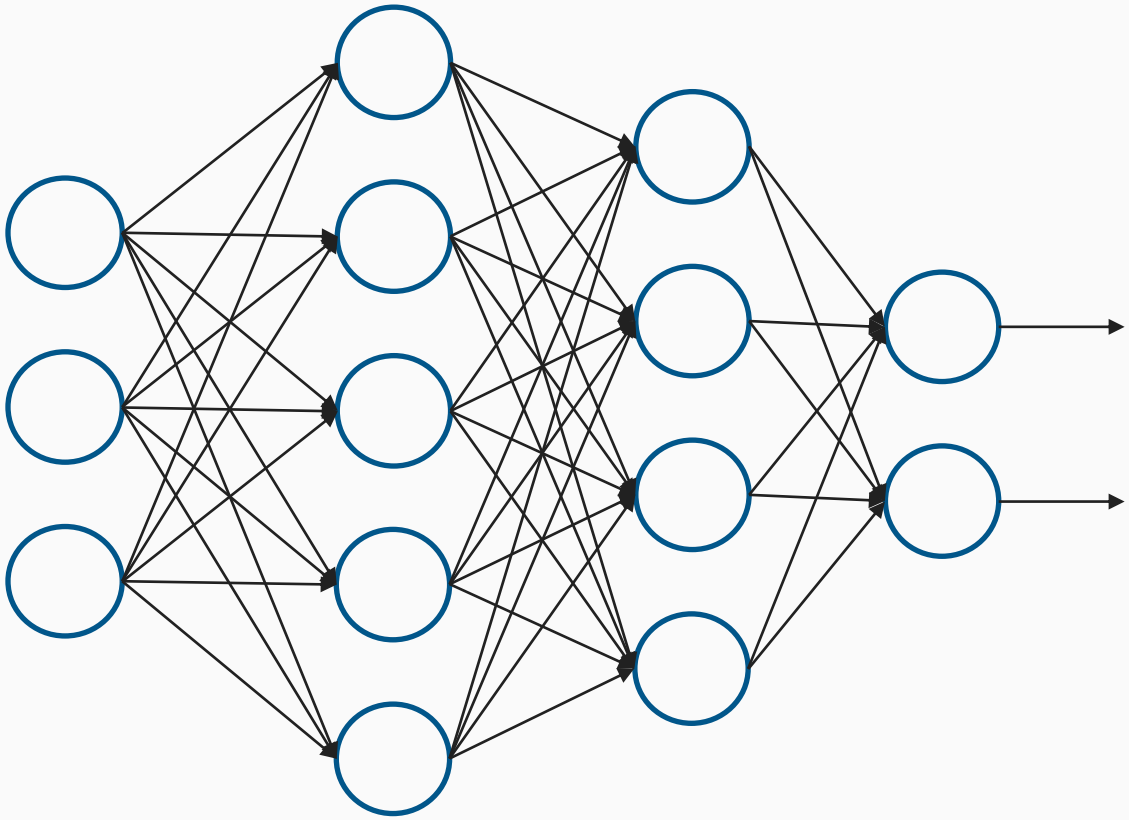
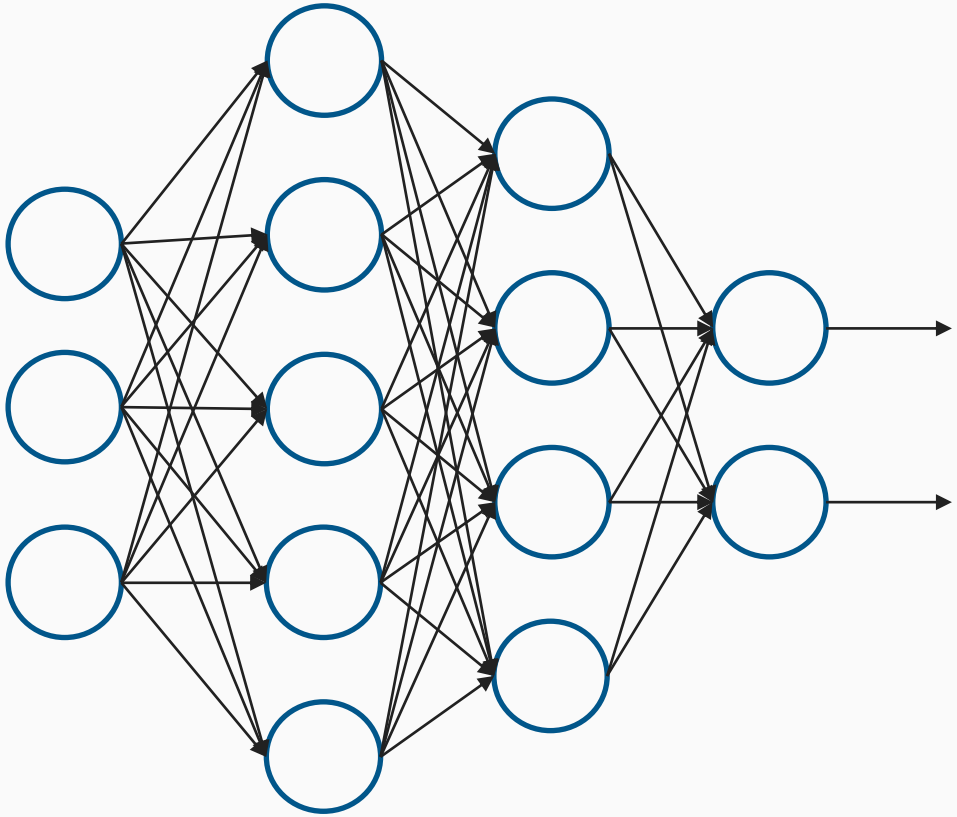
Lecture 5 : CNN

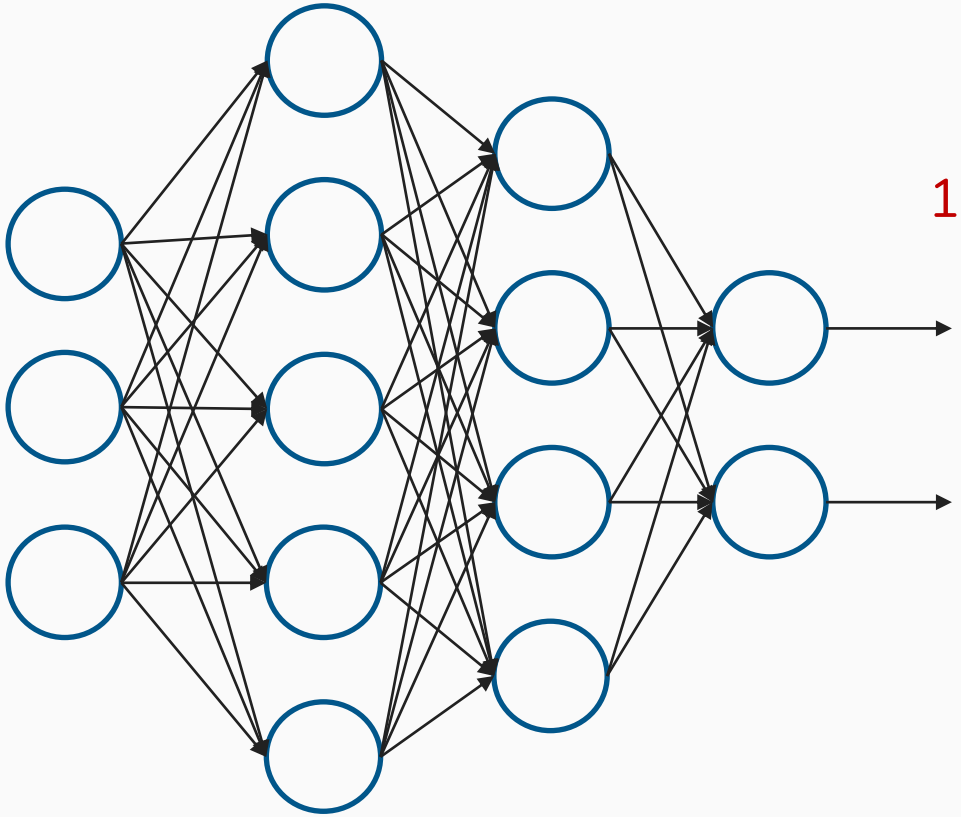
By : Khalil idrissi



Problems with FCN

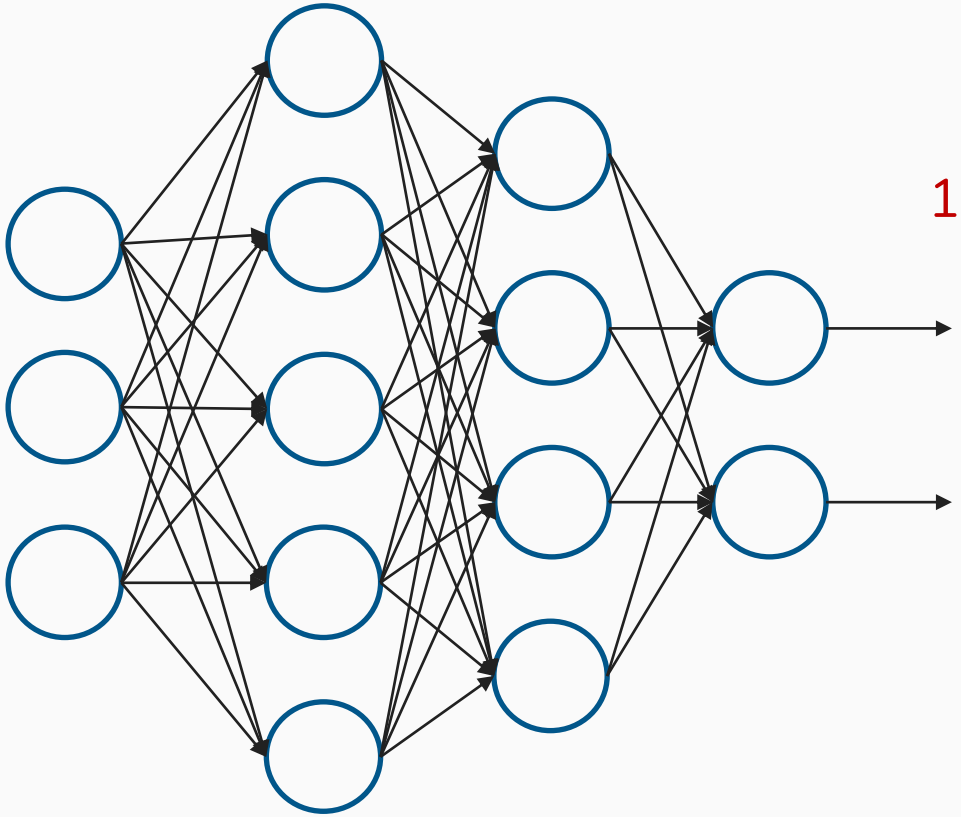


Problems with FCN



1-Explosive number of parameters

Problems with FCN



1-Explosive number of parameters

2-Lack of spatial reasoning

CNN to the rescue

CNN to the rescue

CNN : Convolutional Neural Network

CNN to the rescue

CNN : Convolutional Neural Network

It is a type of NN that uses convolution instead of Matrix multiplication

CNN to the rescue

CNN : Convolutional Neural Network

It is a type of NN that uses ~~convolution~~ instead of Matrix multiplication

CNN to the rescue

CNN : Convolutional Neural Network

Cross-correlation

It is a type of NN that uses ~~convolution~~ instead of Matrix multiplication

Ferq bin convolution w cross correlation site

Cross correlation

= convolution without flipping the kernel

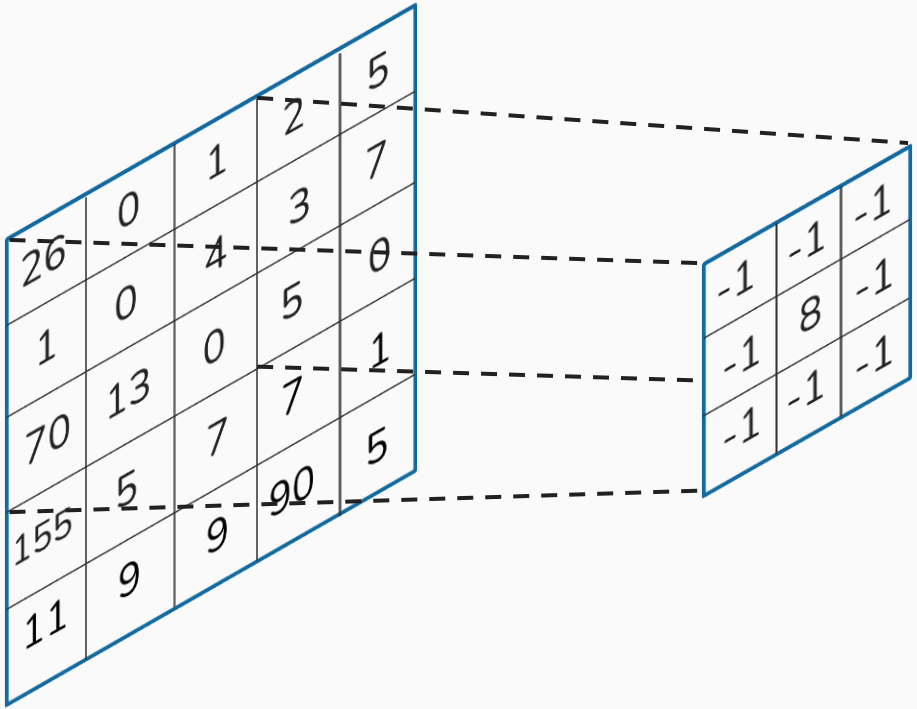
5	0	5	0	5	7
15	7	81	5	2	5
3	9	0	1	3	7
2	13	0	4	5	0
26	70	11	0	7	1
1	155	5	9	90	5

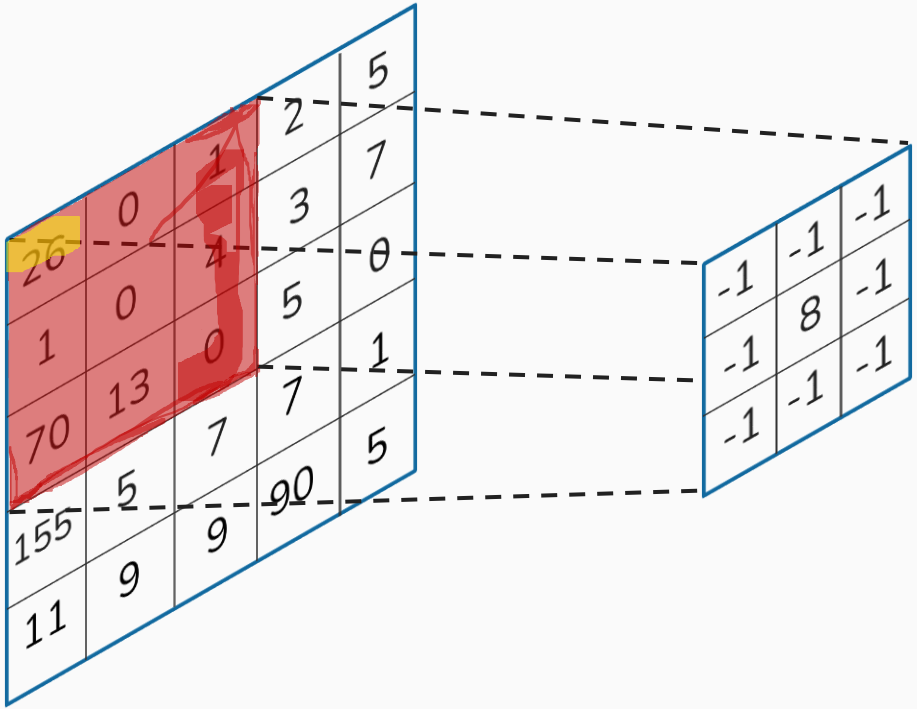
*

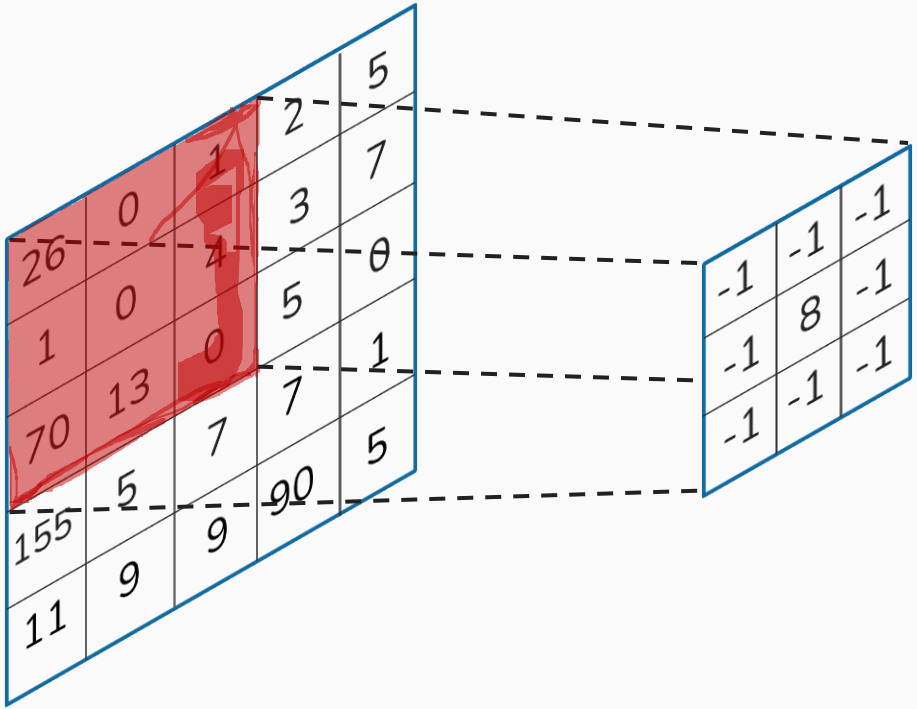
-1	-1	-1
-1	8	-1
-1	-1	-1

26	0	1	2	5
1	0	4	3	7
70	13	0	5	0
155	5	7	7	1
11	9	9	90	5

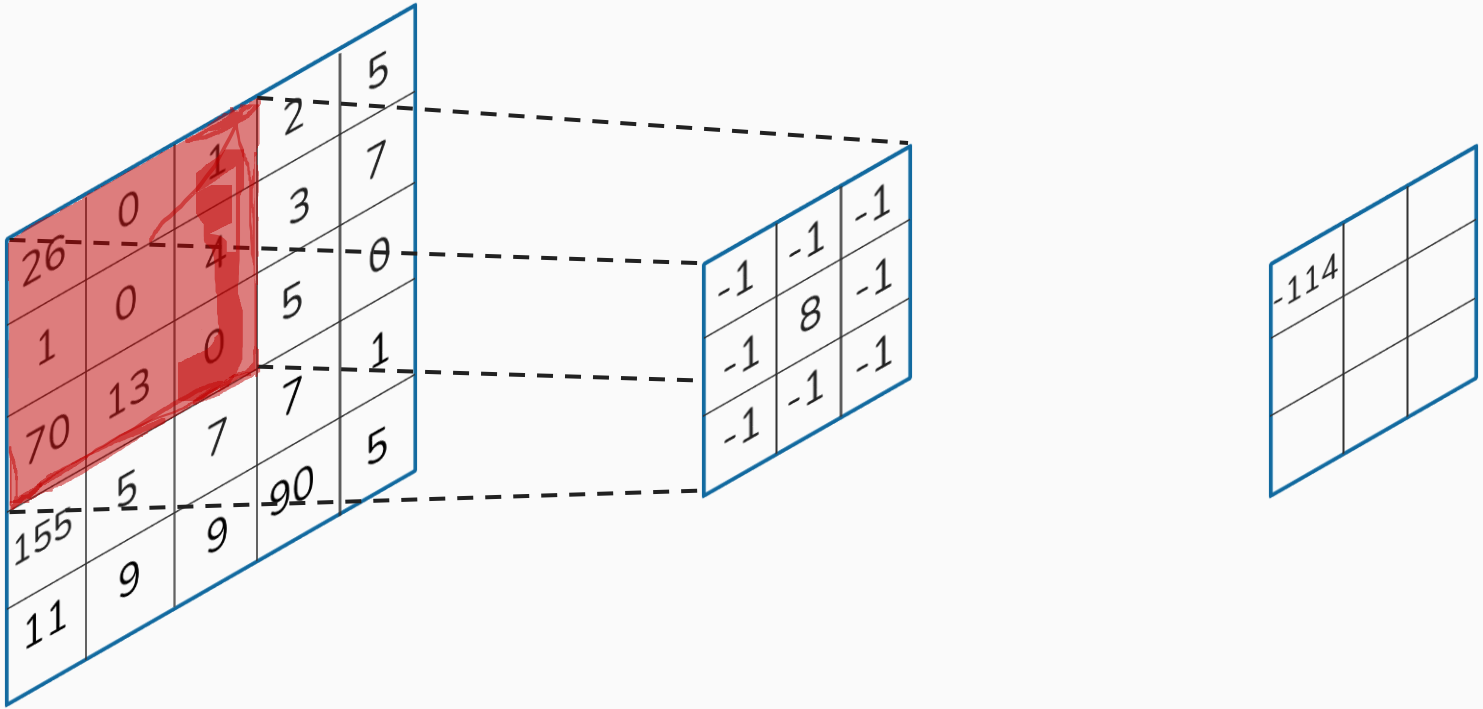
-1	-1	-1
-1	8	-1
-1	-1	-1


26 0 1 2 5
1 0 4 3 7
70 13 0 5 0
155 5 7 7 1
11 9 9 90 5
-1 -1 -1
-1 8 -1
-1 -1 -1

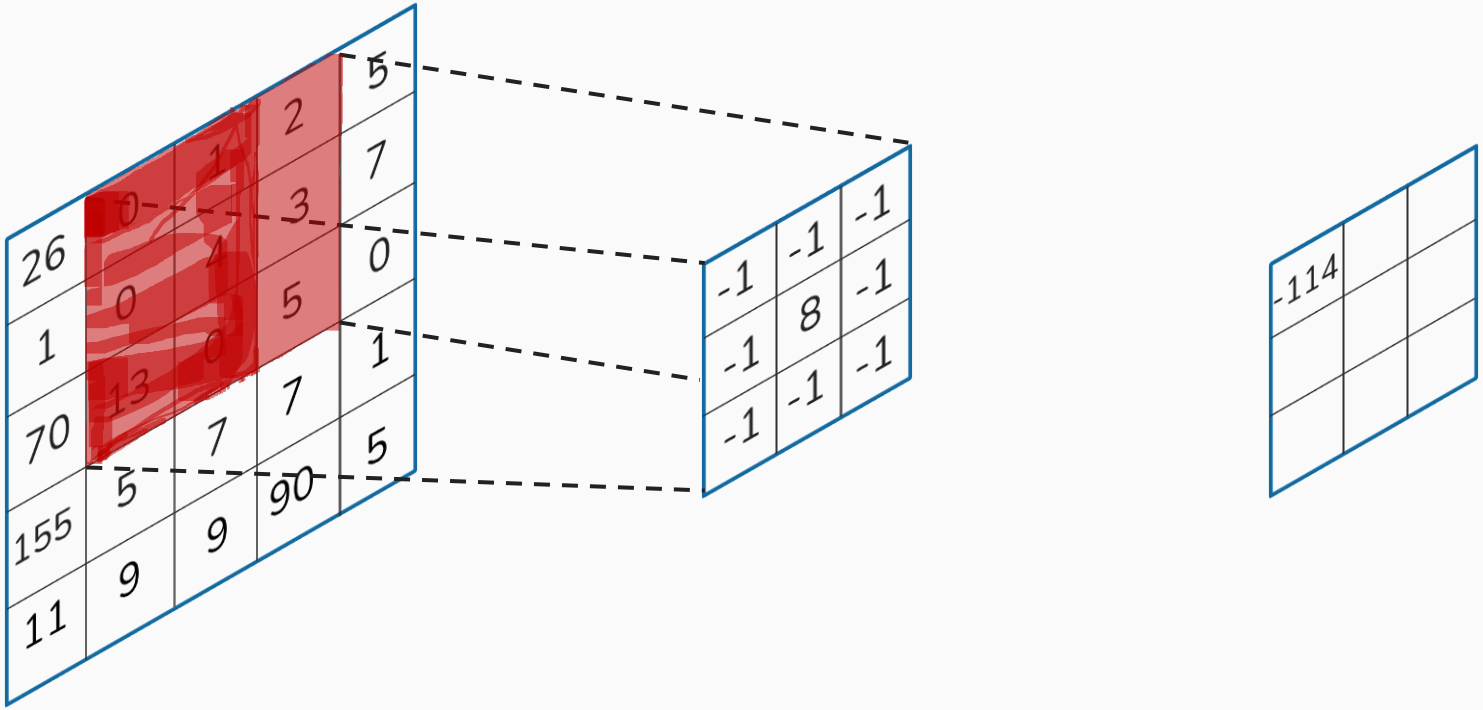

26
0
1
2
5
1
0
4
3
7
1
0
0
5
0
70
13
7
7
1
155
5
9
90
5
11
9
-1
-1
-1
-1
8
-1
-1
-1
-1

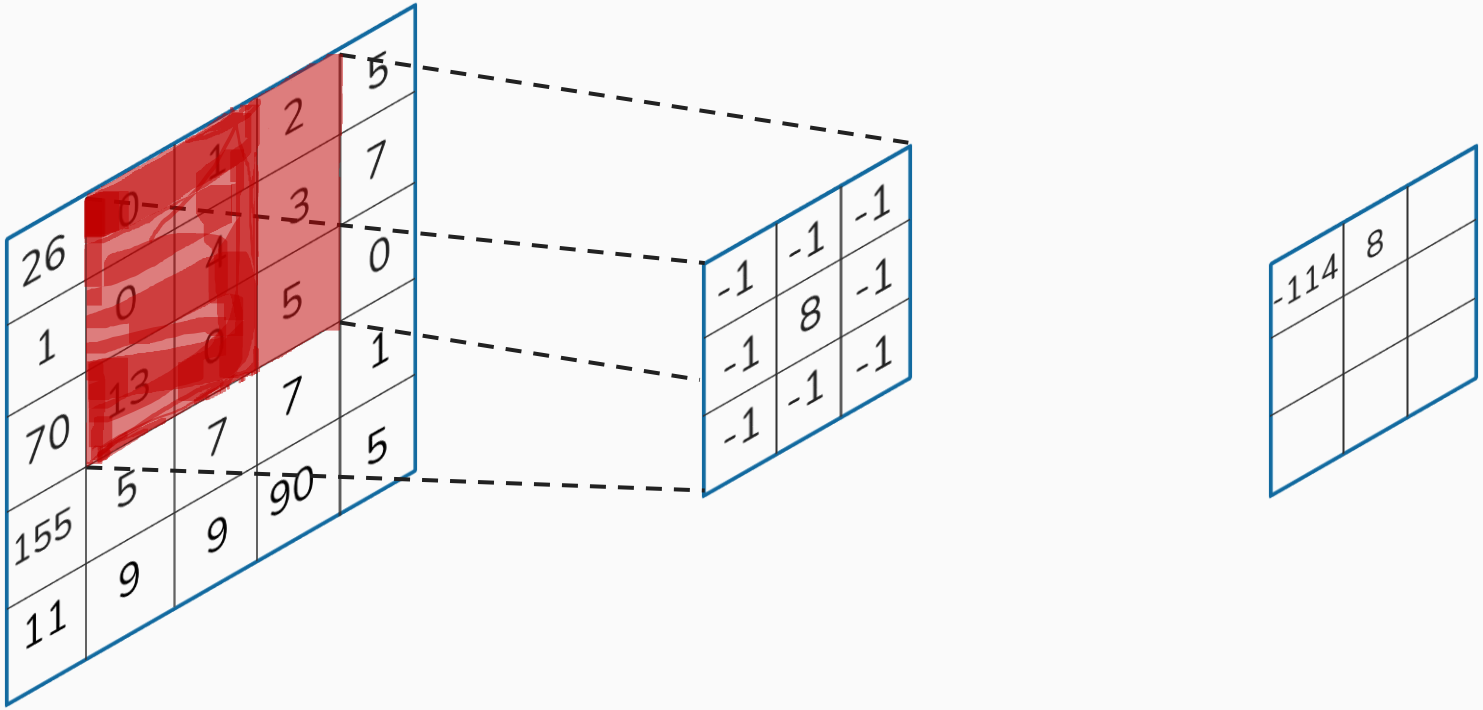


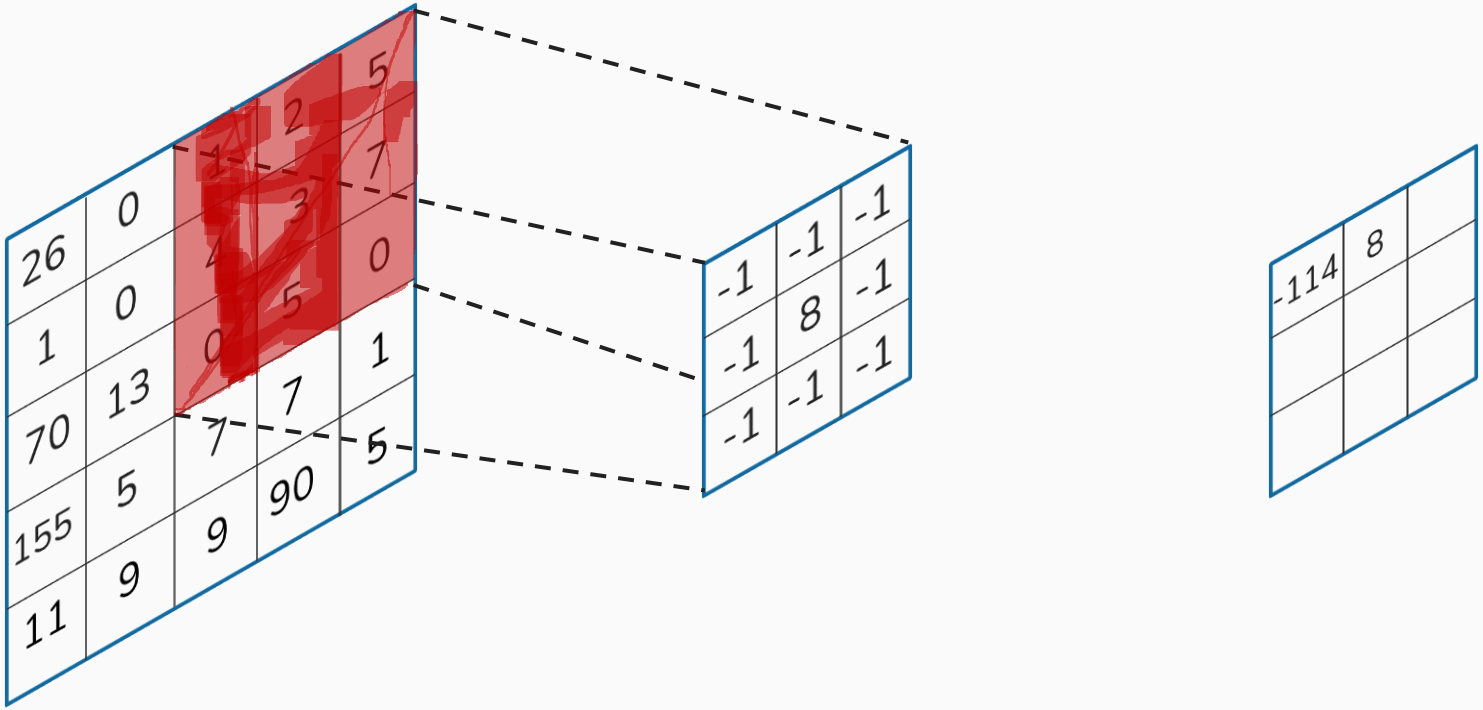
$$26 * (-1) + 0 * (-1) + 1 * (-1) + 1 * 0 + 4 * (-1) + 70 * -1 + 13 * (-1) + 0 * (-1) = -114$$

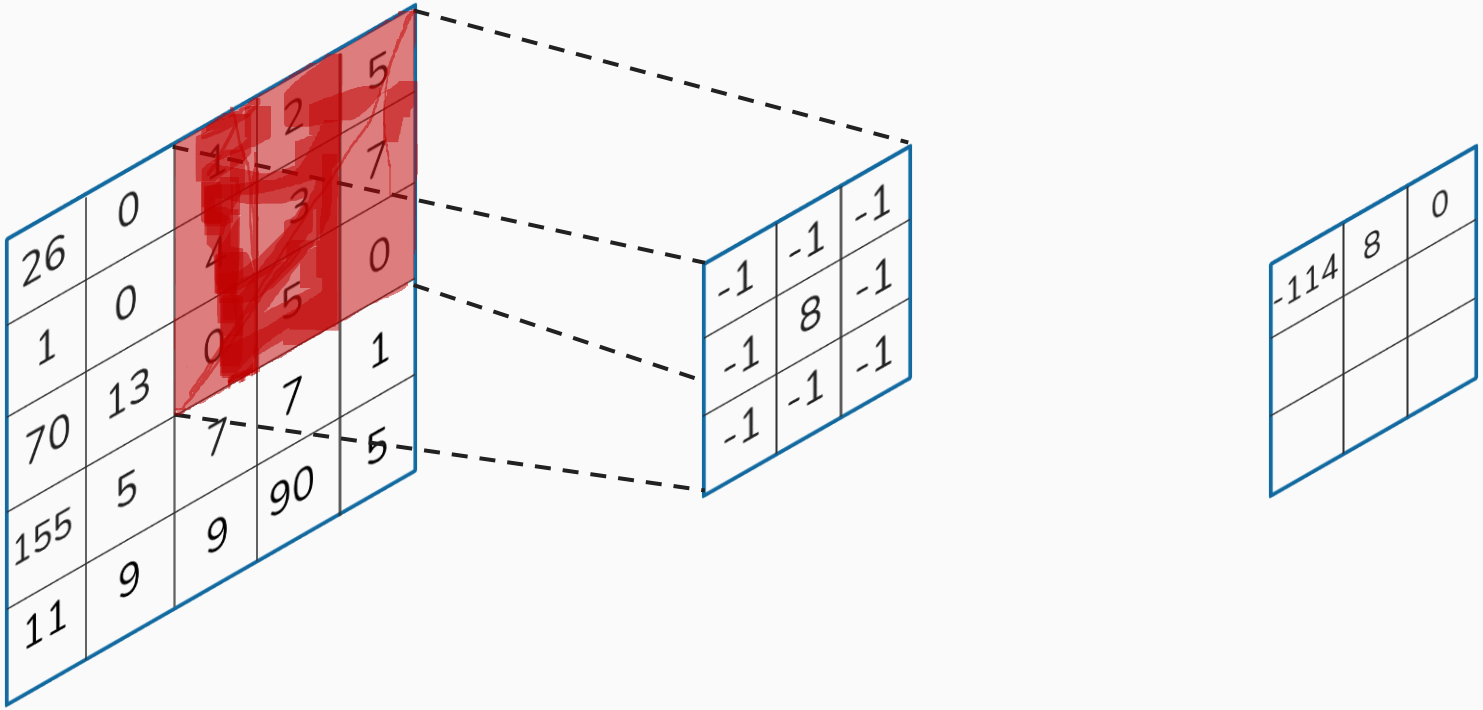


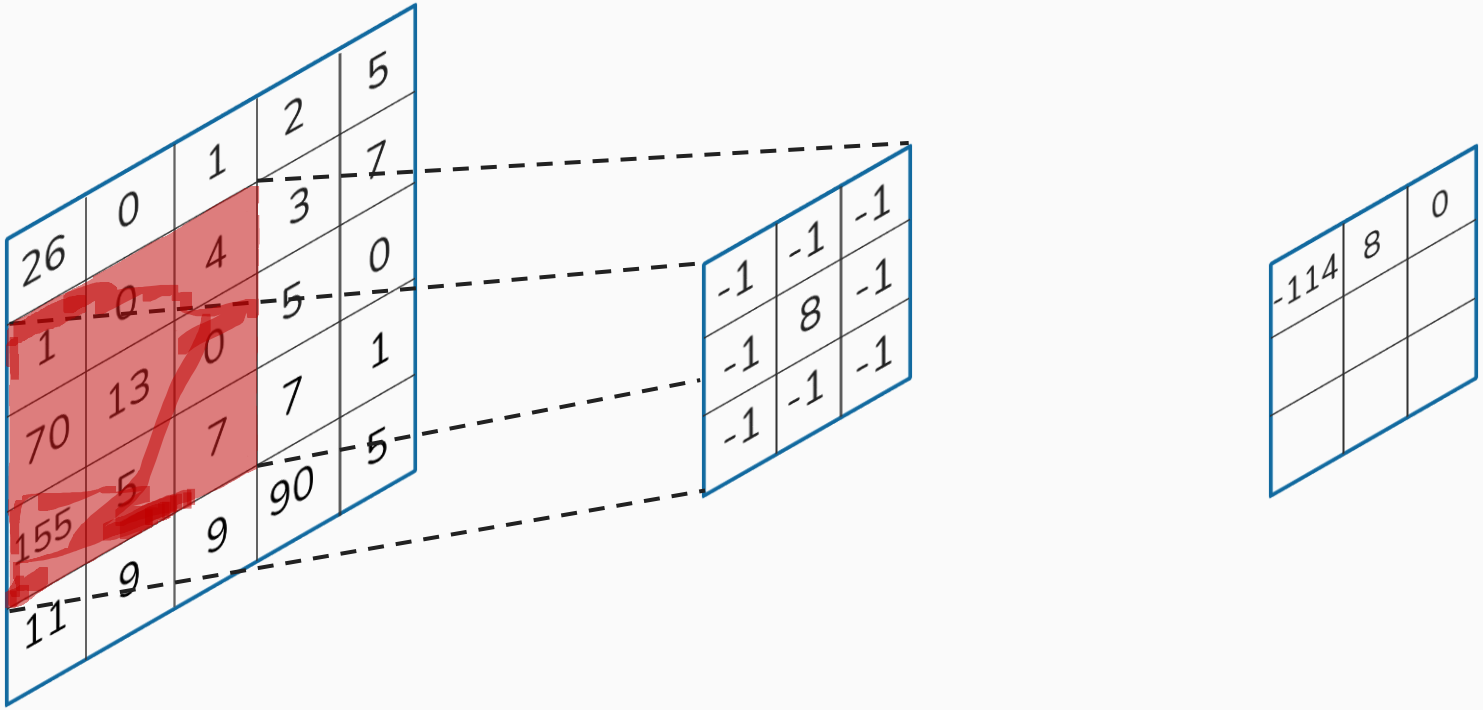
$$26 * (-1) + 0 * (-1) + 1 * (-1) + 1 * 0 + 4 * (-1) + 70 * -1 + 13 * (-1) + 0 * (-1) = -114$$

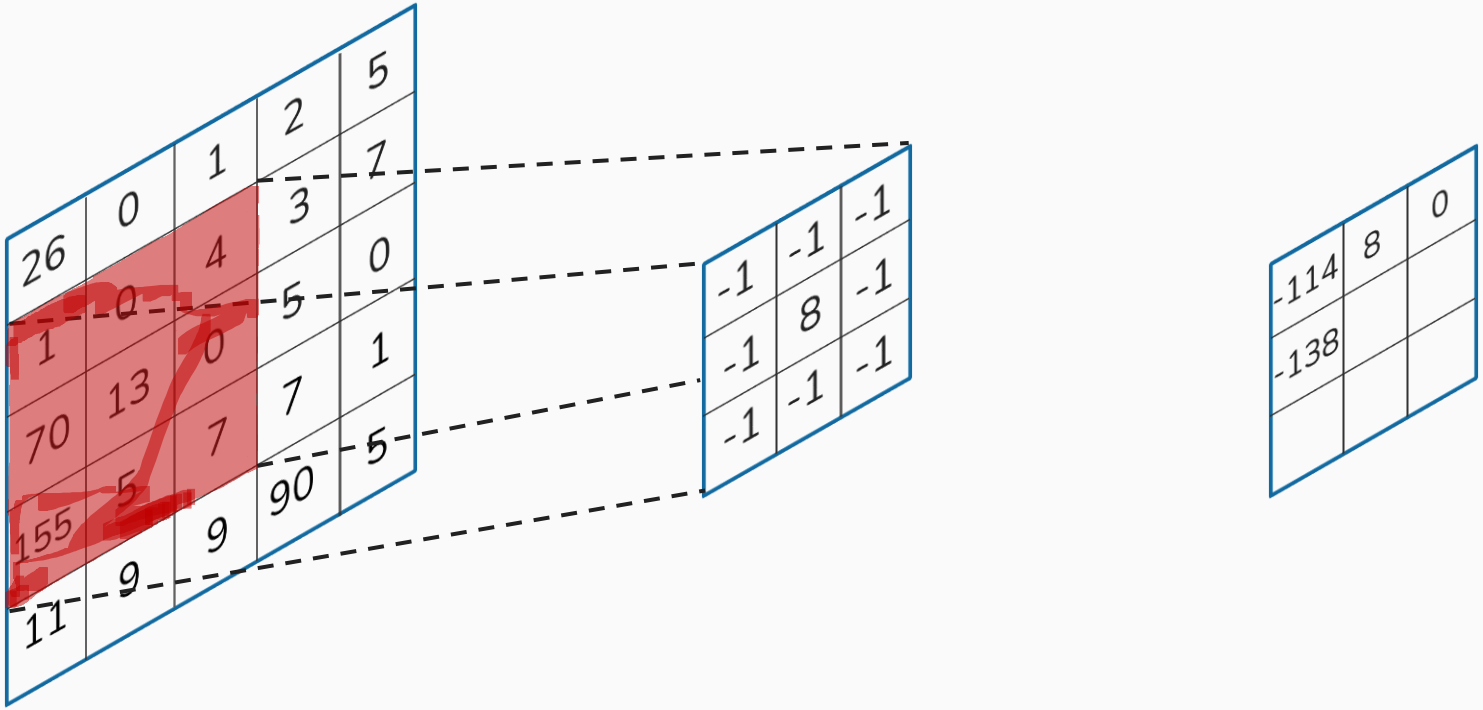

26 0 1 2 5
1 0 4 3 7
70 13 0 5 0
155 5 7 7 1
11 9 9 90 5
-1 -1 -1
-1 8 -1
-1 -1 -1
-114

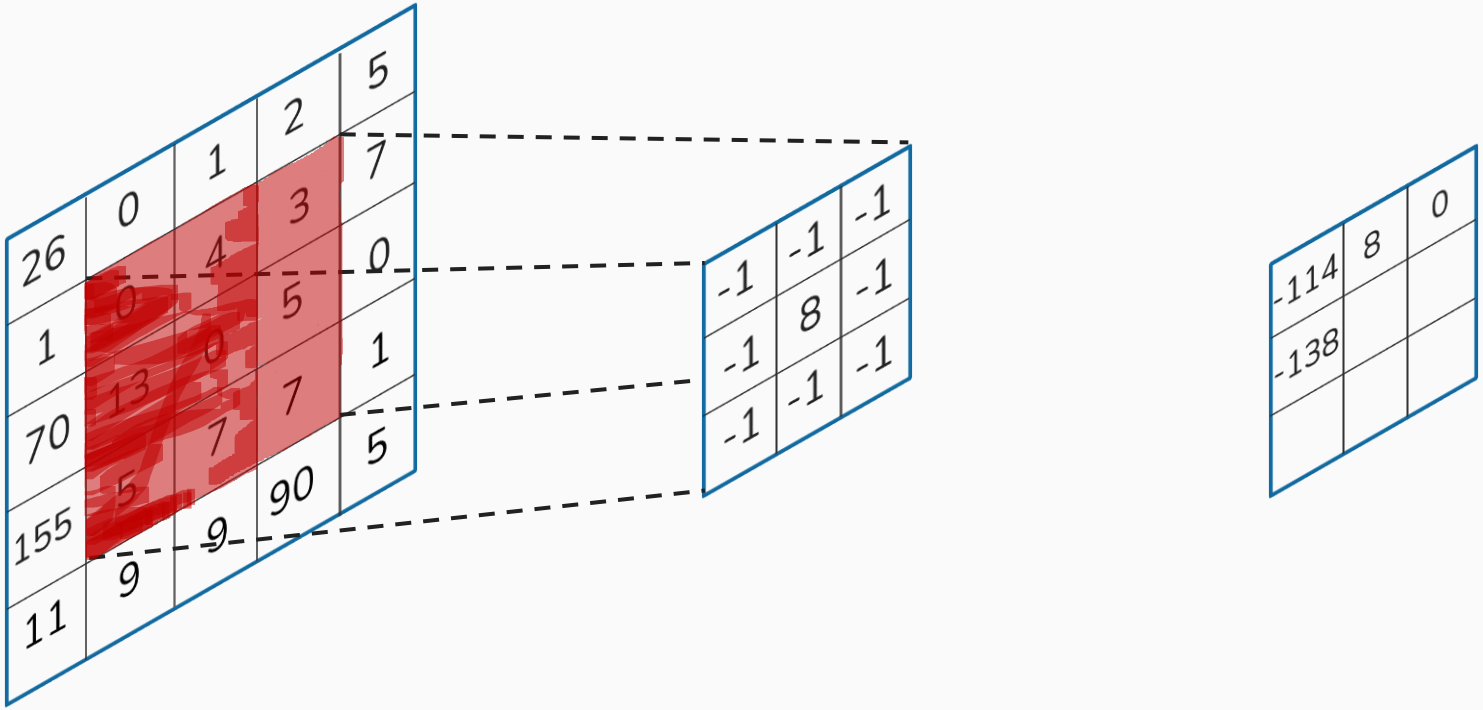

26 0 1 2 5
1 0 4 3 7
70 13 0 5 0
155 5 7 7 1
11 9 9 90 5
-1 -1 -1
-1 8 -1
-1 -1 -1
-114 8

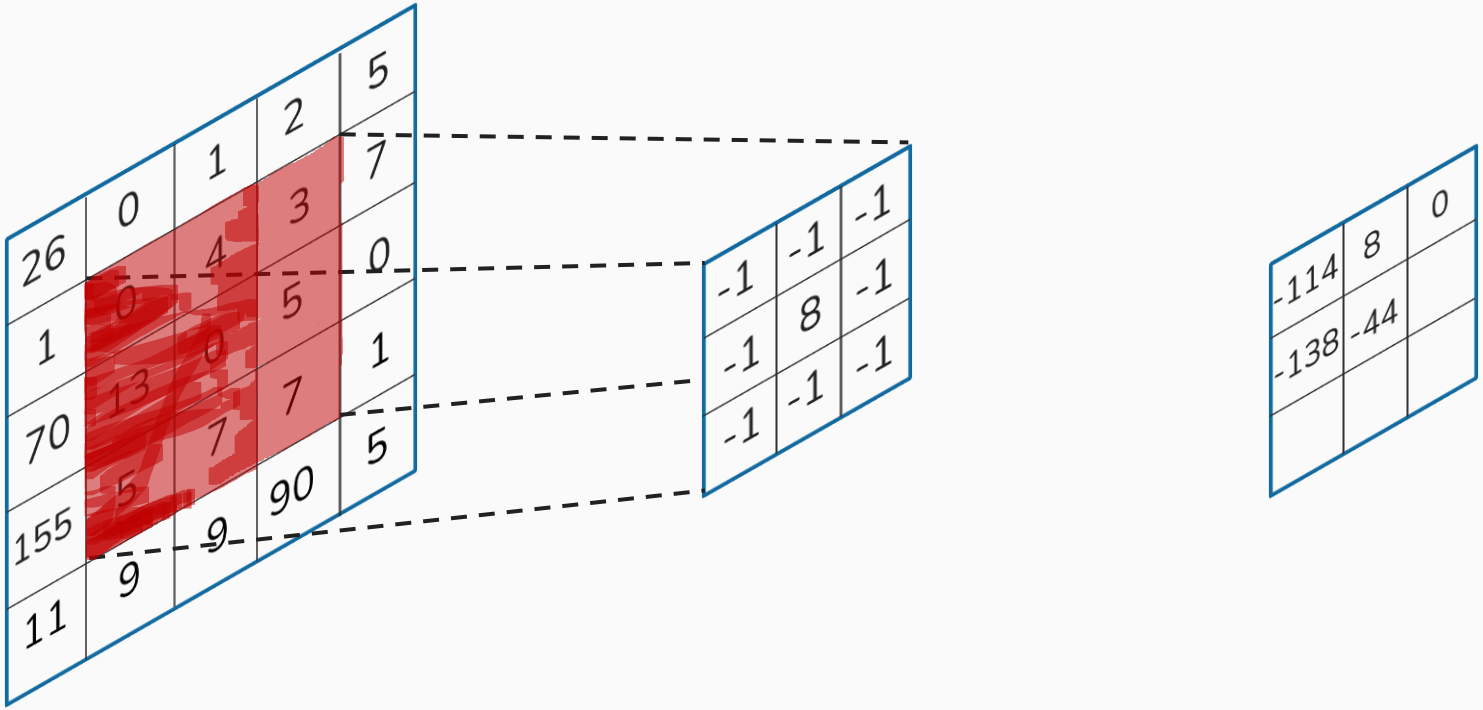

26 0 1 2 5
1 0 4 3 7
70 13 0 5 0
155 5 7 7 1
11 9 9 90 5
-1 -1 -1
-1 8 -1
-1 -1 -1
-114 8

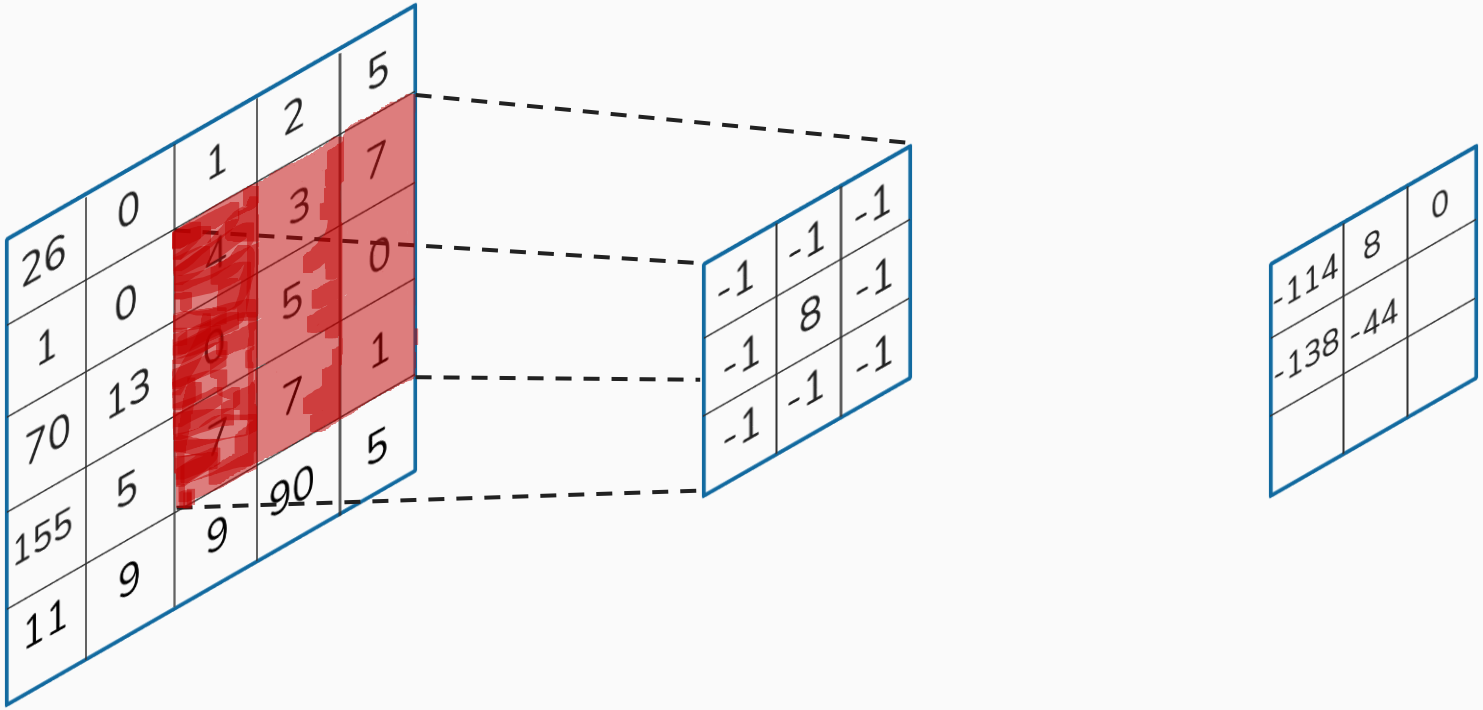

26 0 1 2 5
1 0 4 3 7
70 13 0 5 0
155 5 7 7 1
11 9 9 90 5
-1 -1 -1
-1 8 -1
-1 -1 -1
-114 8 0

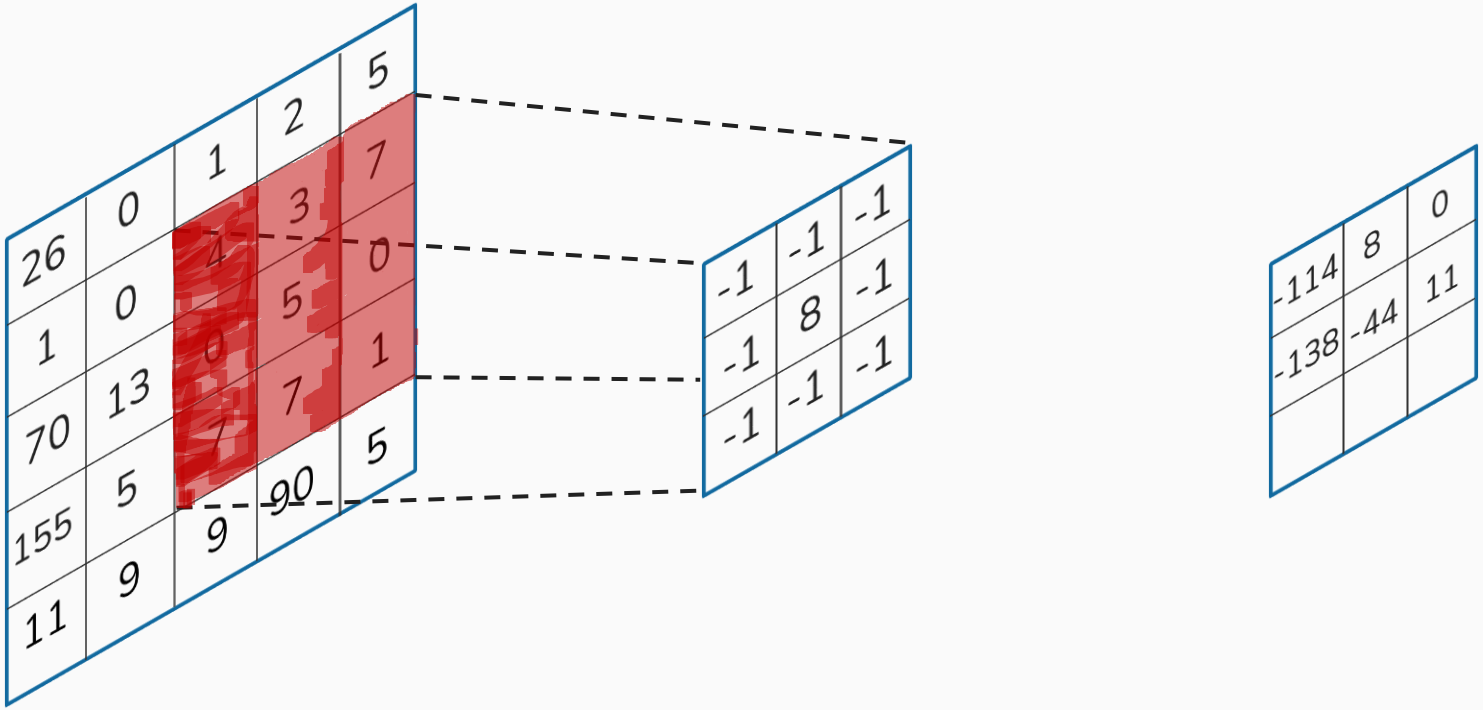

26 0 1 2 5
1 0 4 3 7
70 13 0 5 0
155 5 7 7 1
11 9 9 90 5
-1 -1 -1
-1 8 -1
-1 -1 -1
-114 8 0

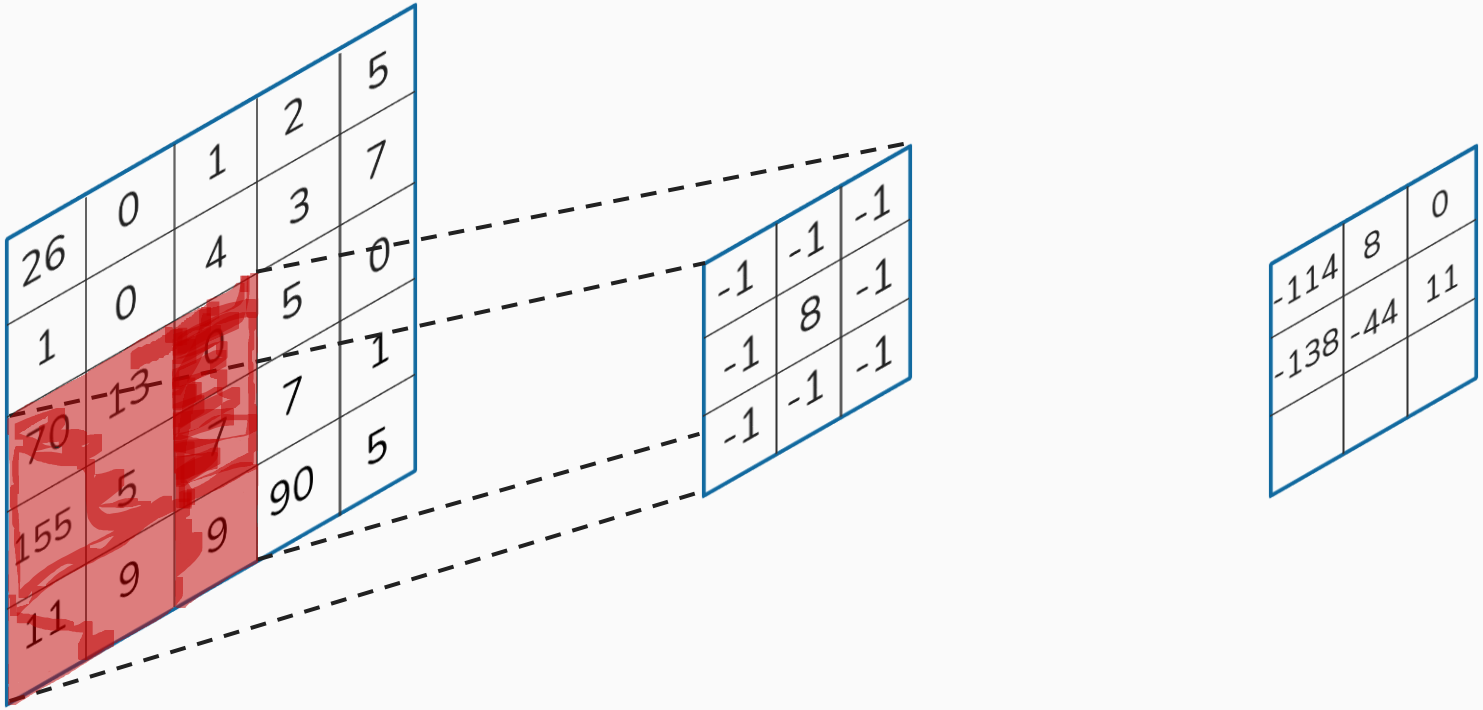

26 0 1 2 5
1 0 4 3 7
70 13 0 5 0
155 5 7 7 1
11 9 9 90 5
-1 -1 -1
-1 8 -1
-1 -1 -1
-114 8 0
-138

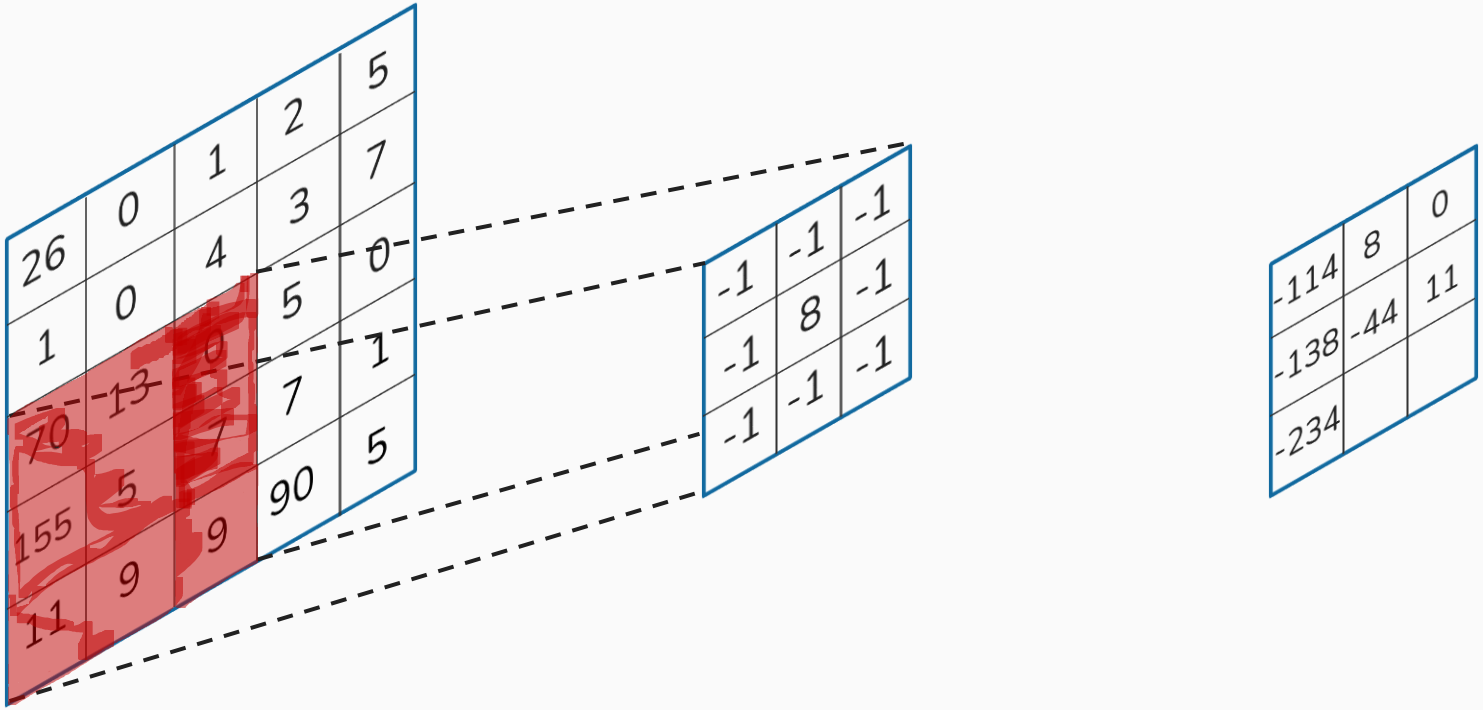

26 0 1 2 5
1 0 4 3 7
70 13 0 5 0
155 5 7 7 1
11 9 9 90 5
-1 -1 -1
-1 8 -1
-1 -1 -1
-114 8 0
-138

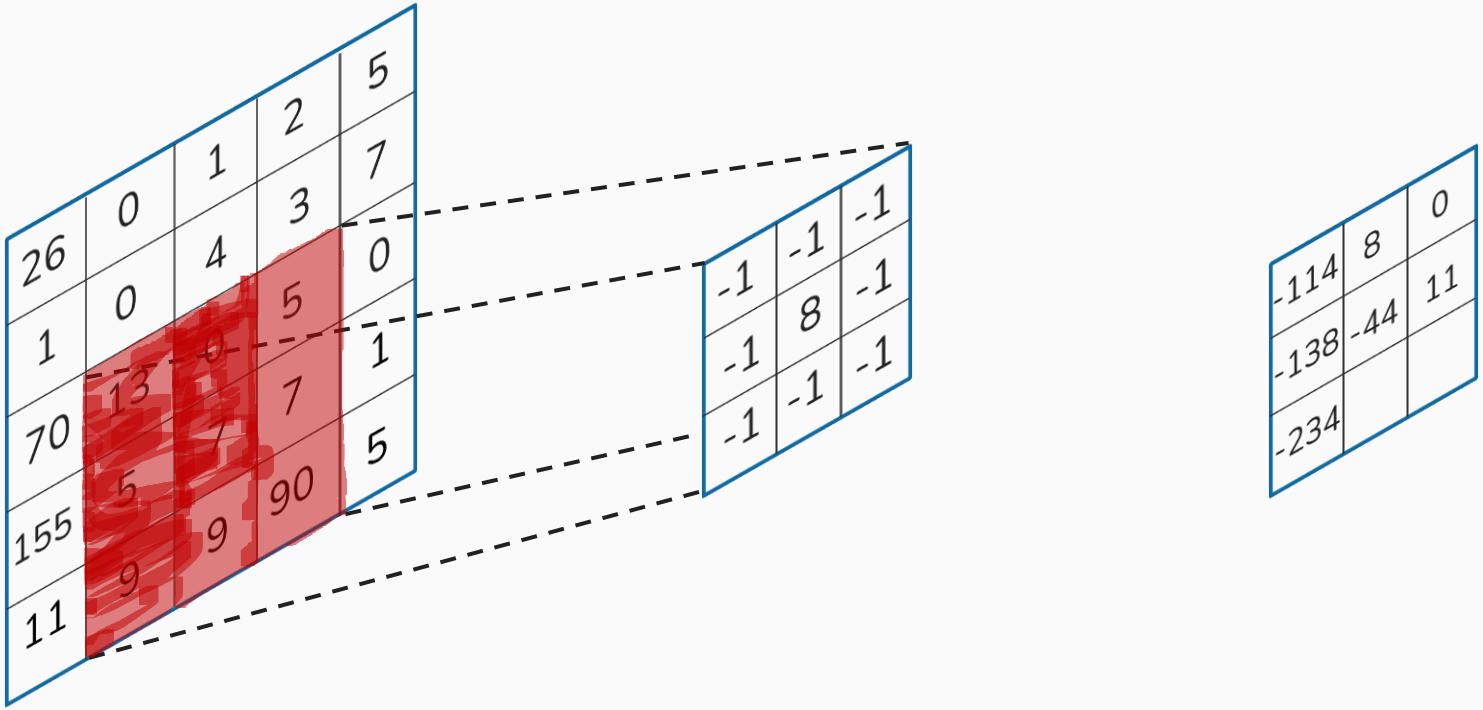

26 0 1 2 5
1 0 4 3 7
70 13 0 5 0
155 5 7 7 1
11 9 9 90 5
-1 -1 -1
-1 8 -1
-1 -1 -1
-114 8 0
-138 -44

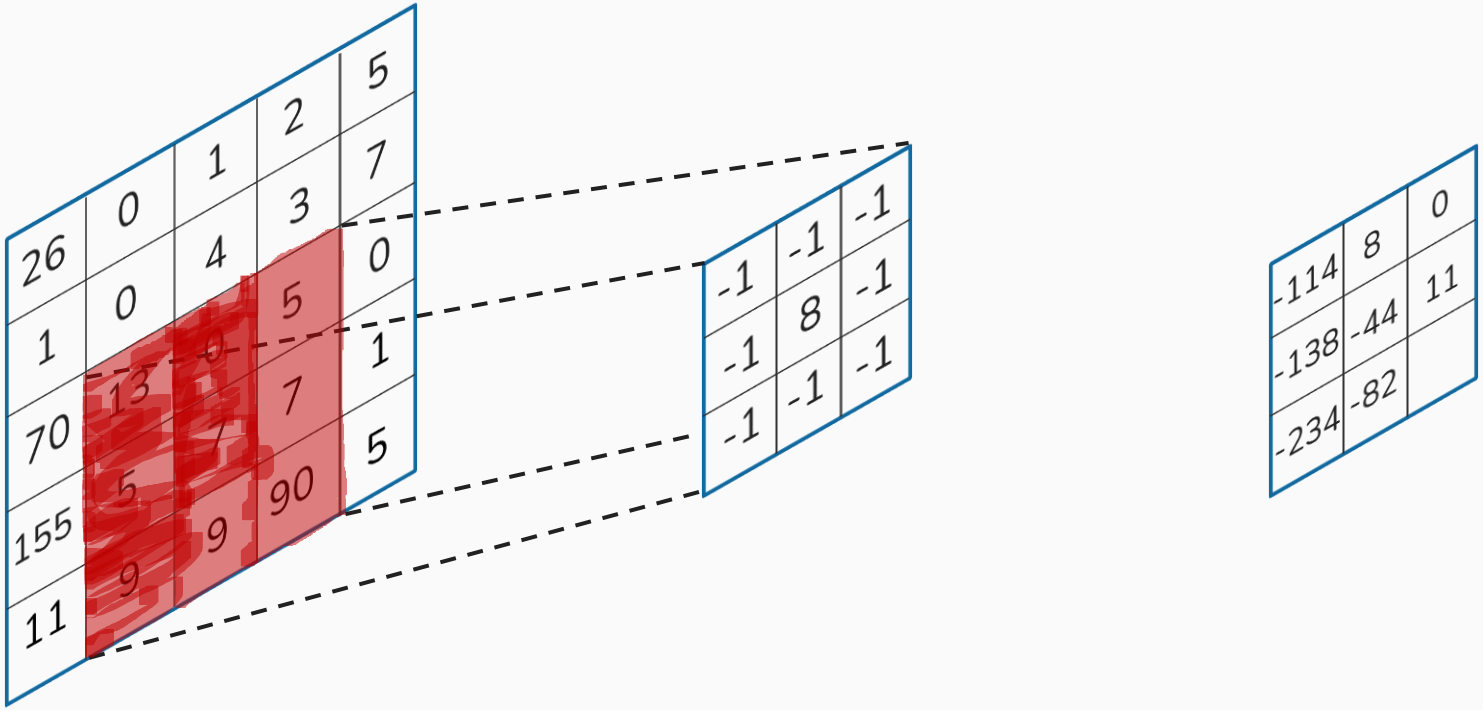

26 0 1 2 5
1 0 4 3 7
70 13 0 5 0
155 5 7 7 1
11 9 9 90 5
-1 -1 -1
-1 8 -1
-1 -1 -1
-114 8 0
-138 -44

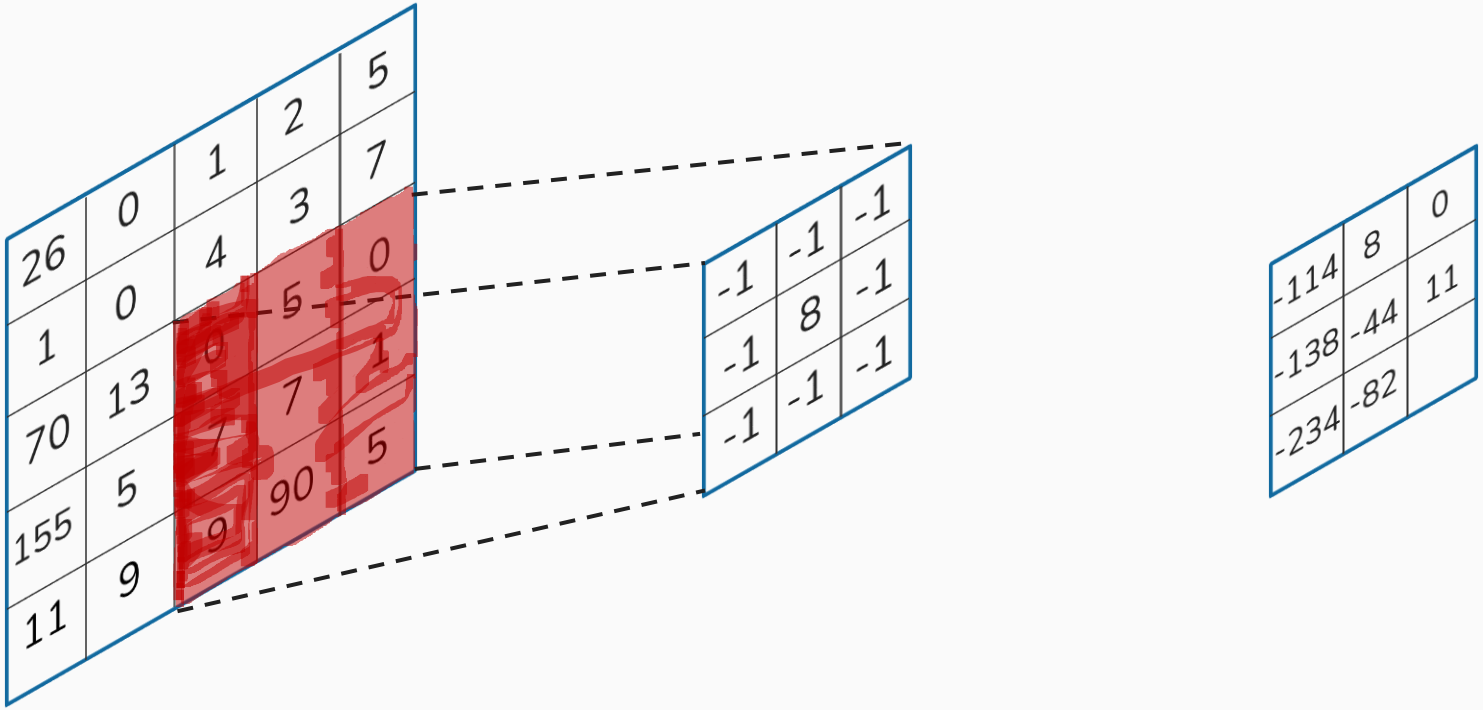

26 0 1 2 5
1 0 4 3 7
70 13 0 5 0
155 5 7 7 1
11 9 9 90 5
-1 -1 -1
-1 8 -1
-1 -1 -1
-114 8 0
-138 -44 11

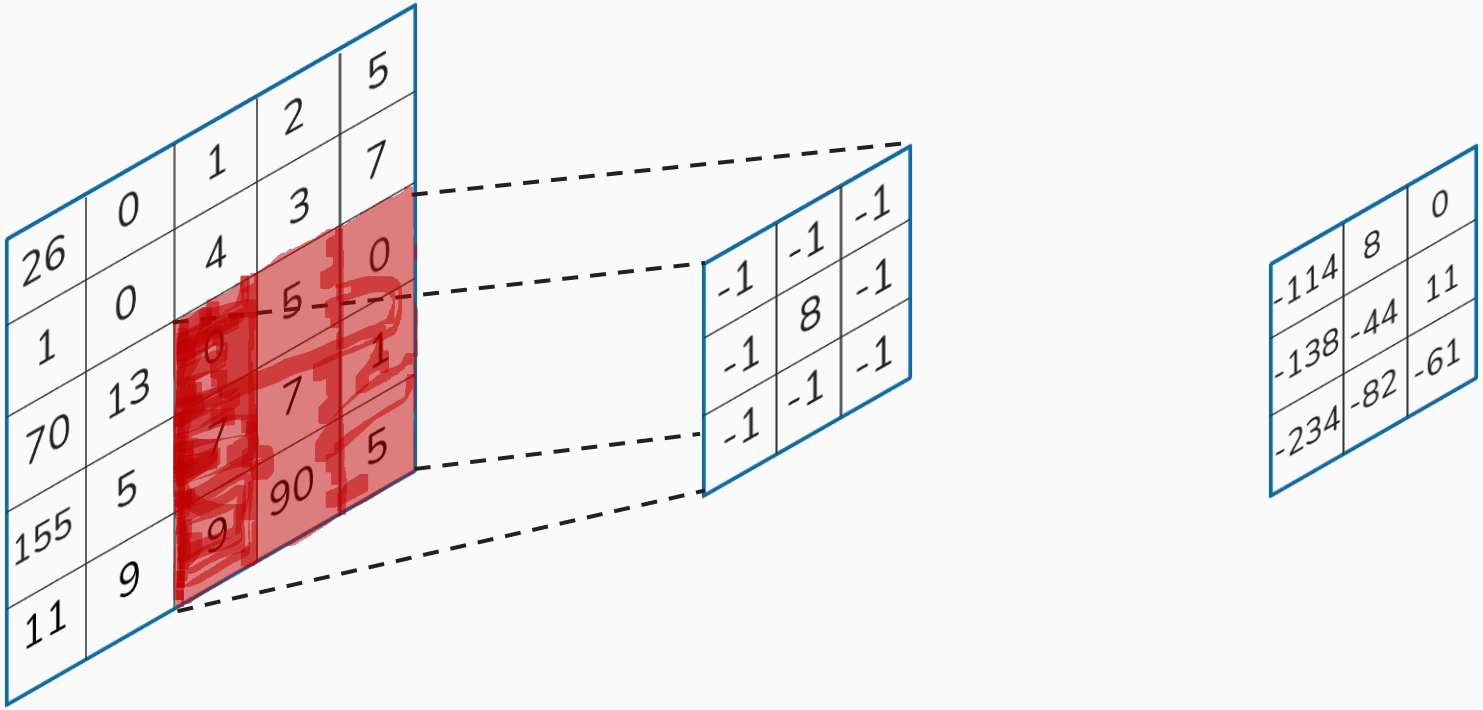

26 0 1 2 5
1 0 4 3 7
70 13 0 5 0
155 5 7 7 1
11 9 9 90 5
-1 -1 -1
-1 8 -1
-1 -1 -1
-114 8 0
-138 -44 11


26 0 1 2 5
1 0 4 3 7
70 13 0 5 0
155 5 7 7 1
11 9 9 90 5
-1 -1 -1
-1 8 -1
-1 -1 -1
-114 8 0
-138 -44 11
-234


26 0 1 2 5
1 0 4 3 7
70 13 0 5 0
155 5 7 7 1
11 9 9 90 5
-1 -1 -1
-1 8 -1
-1 -1 -1
-114 8 0
-138 -44 11
-234


26 0 1 2 5
1 0 4 3 7
70 13 0 5 0
155 5 7 7 1
11 9 9 90 5
-1 -1 -1
-1 8 -1
-1 -1 -1
-114 8 0
-138 -44 11
-234 -82


26 0 1 2 5
1 0 4 3 7
70 13 0 5 0
155 5 7 7 1
11 9 9 90 5
-1 -1 -1
-1 8 -1
-1 -1 -1
-114 8 0
-138 -44 11
-234 -82


26 0 1 2 5
1 0 4 3 7
70 13 0 5 0
155 5 7 7 1
11 9 9 90 5
-1 -1 -1
-1 8 -1
-1 -1 -1
-114 8 0
-138 -44 11
-234 -82 -61

26	0	1	2	5
1	0	4	3	7
70	13	0	5	0
155	5	7	7	1
11	9	9	90	5

*

-1	-1	-1
-1	8	-1
-1	-1	-1

-114	8	0
-138	-44	11
-234	-82	-61

26	0	1	2	5
1	0	4	3	7
70	13	0	5	0
155	5	7	7	1
11	9	9	90	5

*

-1	-1	-1
-1	8	-1
-1	-1	-1

=

-114	8	0
-138	-44	11
-234	-82	-61

26	0	1	2	5
1	0	4	3	7
70	13	0	5	0
155	5	7	7	1
11	9	9	90	5

input

*

-1	-1	-1
-1	8	-1
-1	-1	-1

=

-114	8	0
-138	-44	11
-234	-82	-61

26	0	1	2	5
1	0	4	3	7
70	13	0	5	0
155	5	7	7	1
11	9	9	90	5

input

*

-1	-1	-1
-1	8	-1
-1	-1	-1

Kernel

=

-114	8	0
-138	-44	11
-234	-82	-61

26	0	1	2	5
1	0	4	3	7
70	13	0	5	0
155	5	7	7	1
11	9	9	90	5

input

*

-1	-1	-1
-1	8	-1
-1	-1	-1

Kernel

=

-114	8	0
-138	-44	11
-234	-82	-61

26	0	1	2	5
1	0	4	3	7
70	13	0	5	0
155	5	7	7	1
11	9	9	90	5

input

*

-1	-1	-1
-1	8	-1
-1	-1	-1

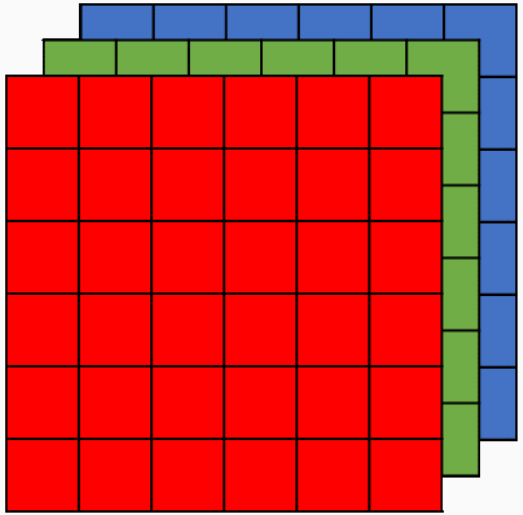
Kernel

=

-114	8	0
-138	-44	11
-234	-82	-61

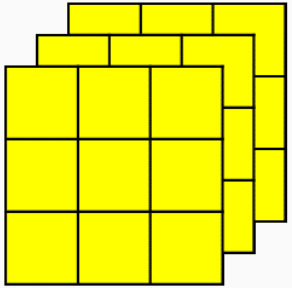
Feature map

For RGB images



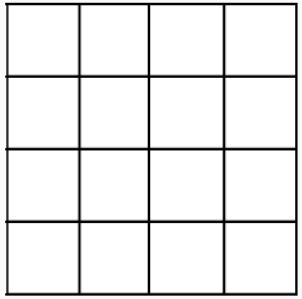
$6 \times 6 \times \boxed{3}$

*



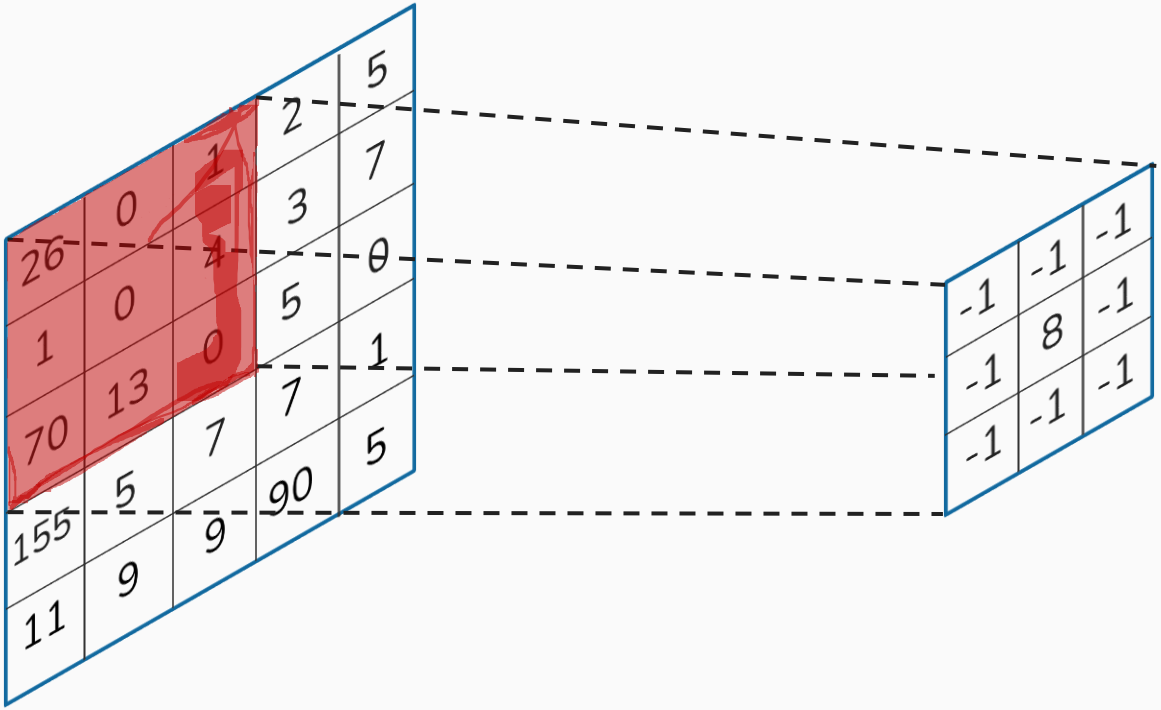
$3 \times 3 \times \boxed{3}$

=

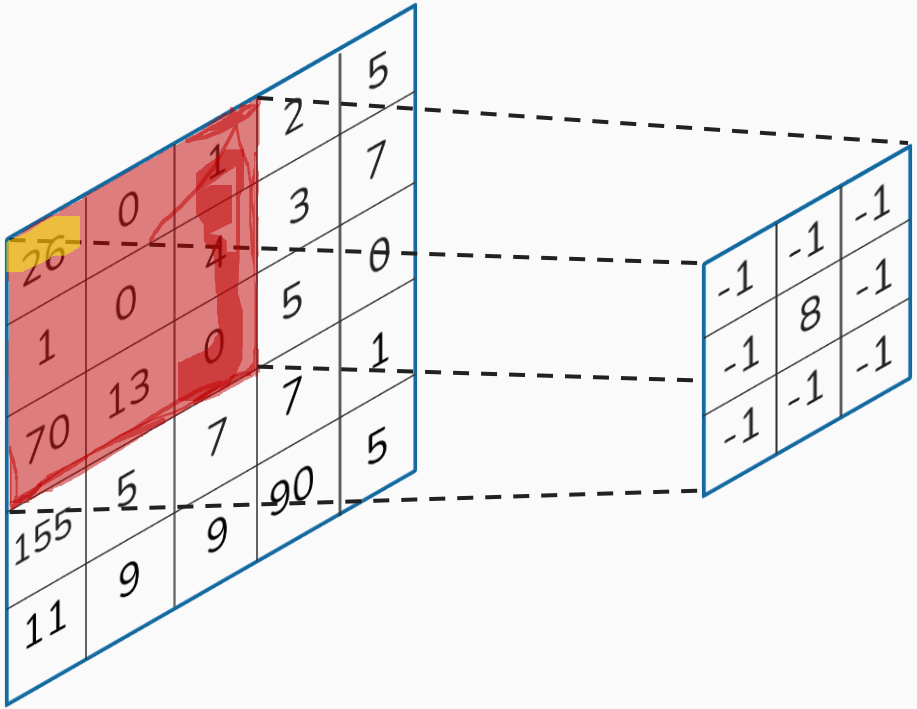


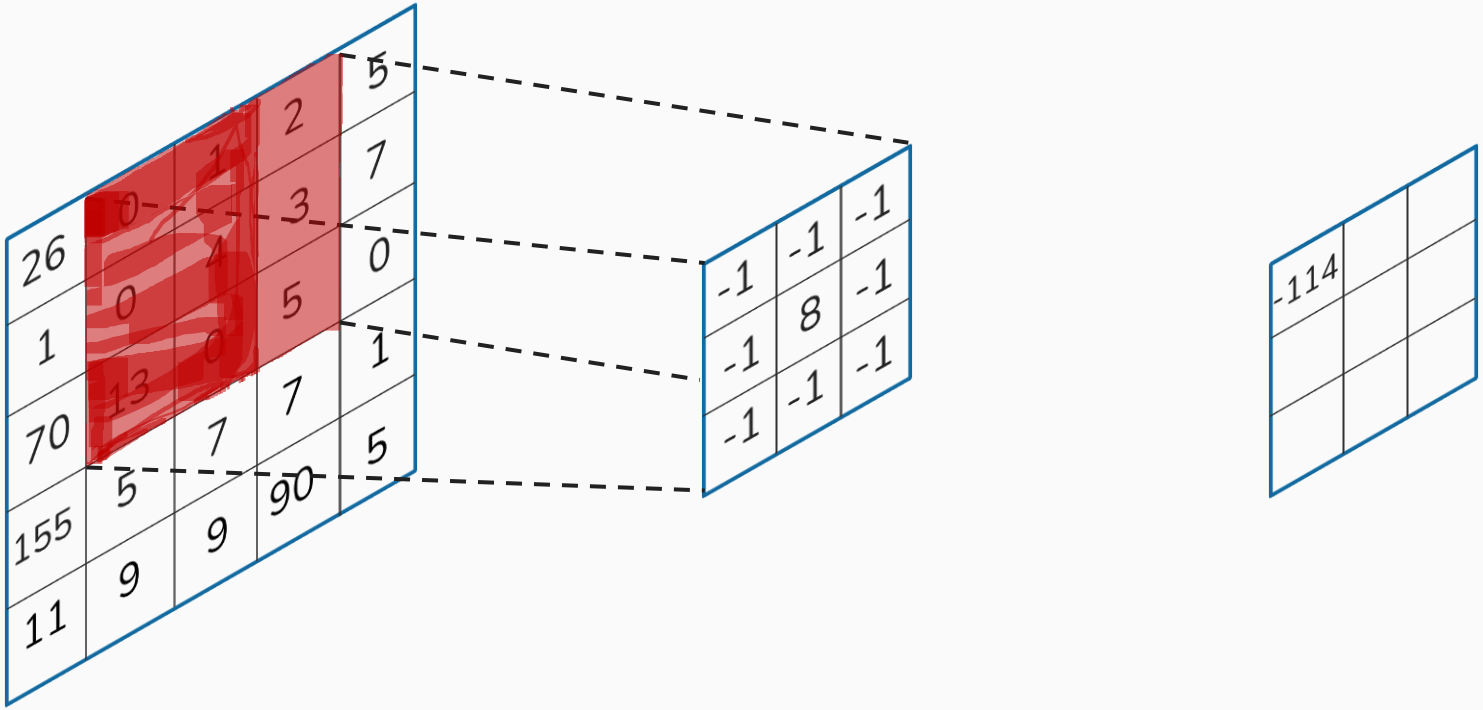
4×4

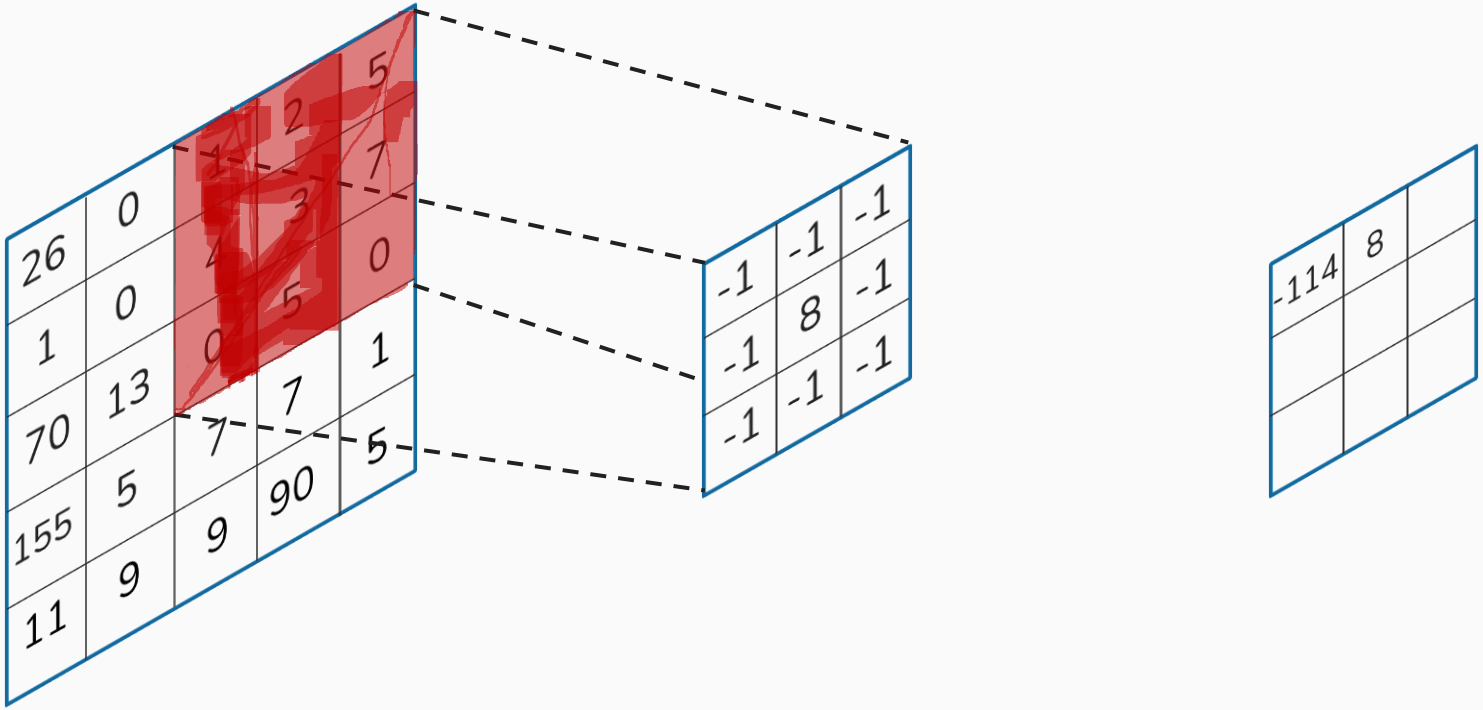
Convnets are not connected to every pixel in the input image but only the pixels in their receptive field

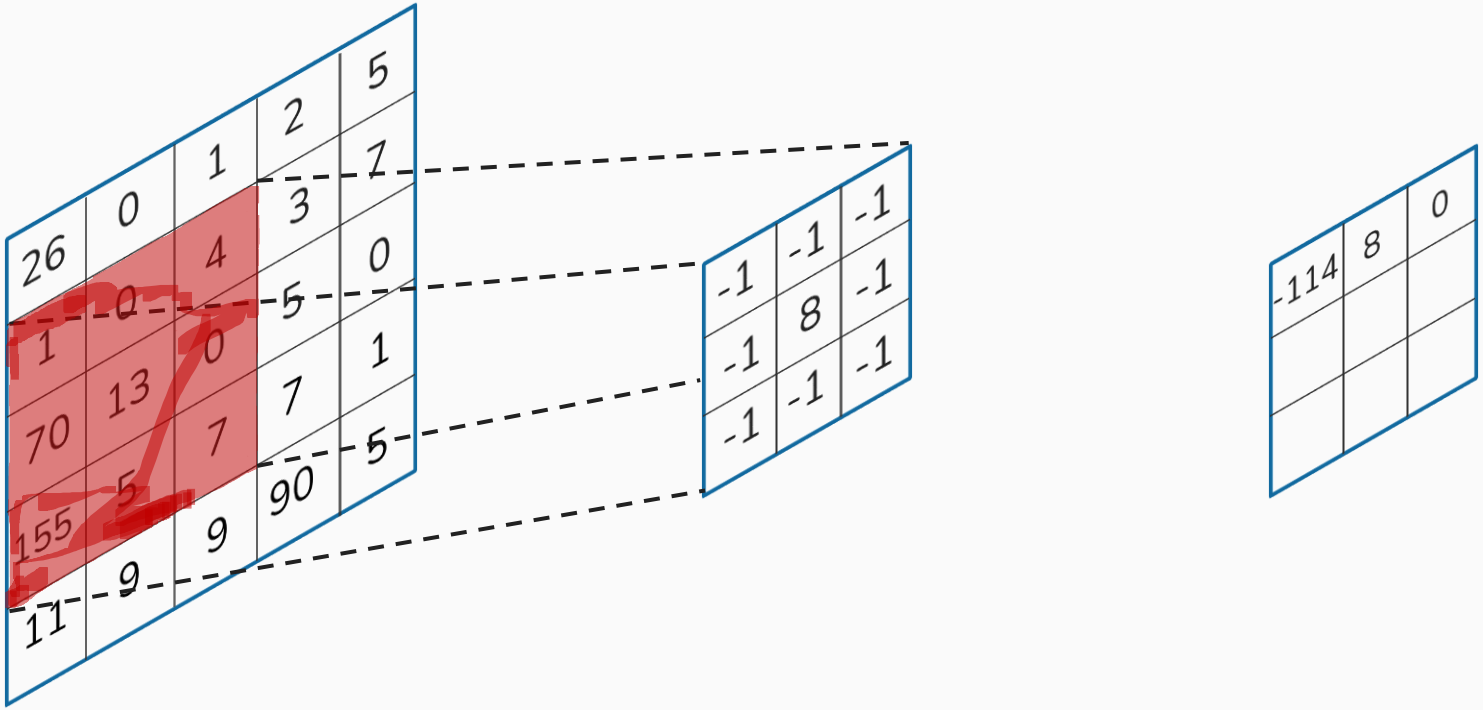

26
0
1
2
5
1
0
4
3
7
1
13
0
5
0
70
7
7
1
155
5
9
90
5
11
9
-1
-1
-1
-1
8
-1
-1
-1
-1

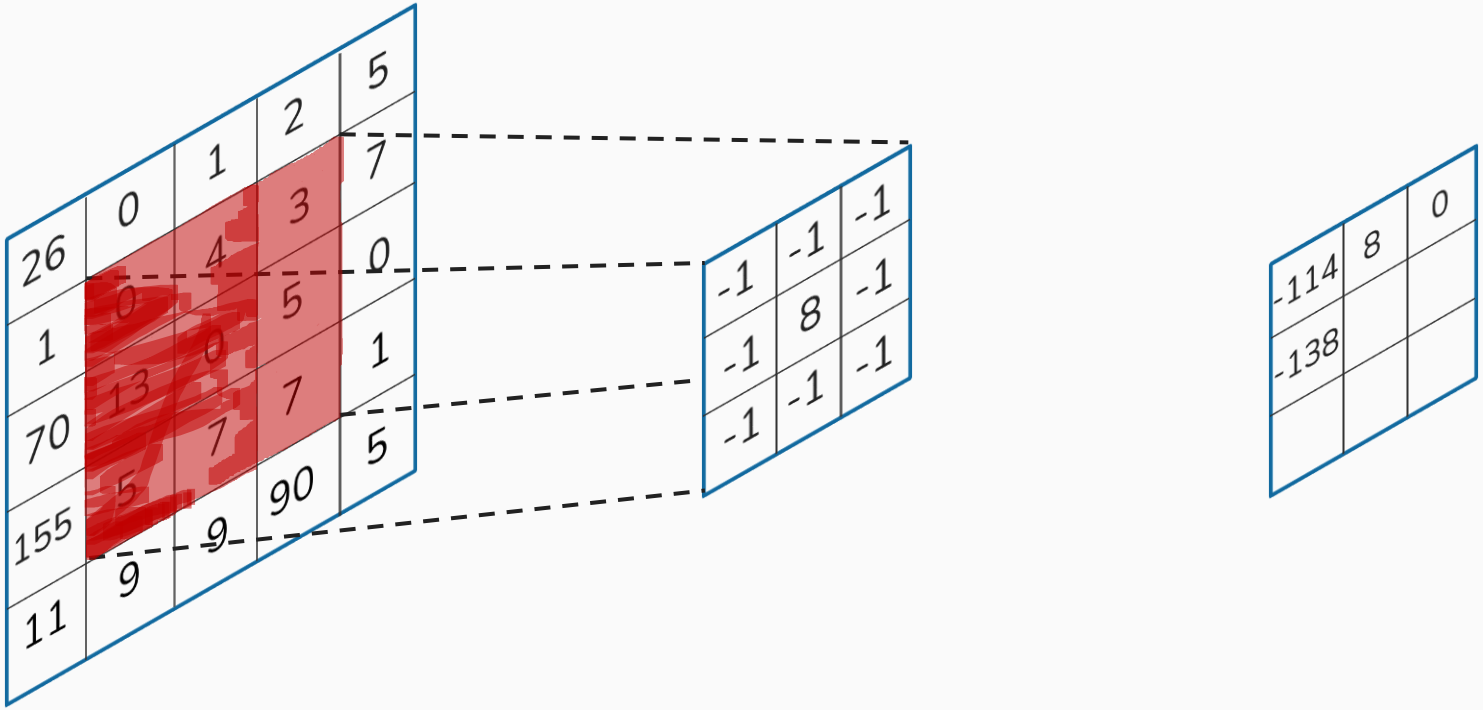
Convnets have shared weights an biases

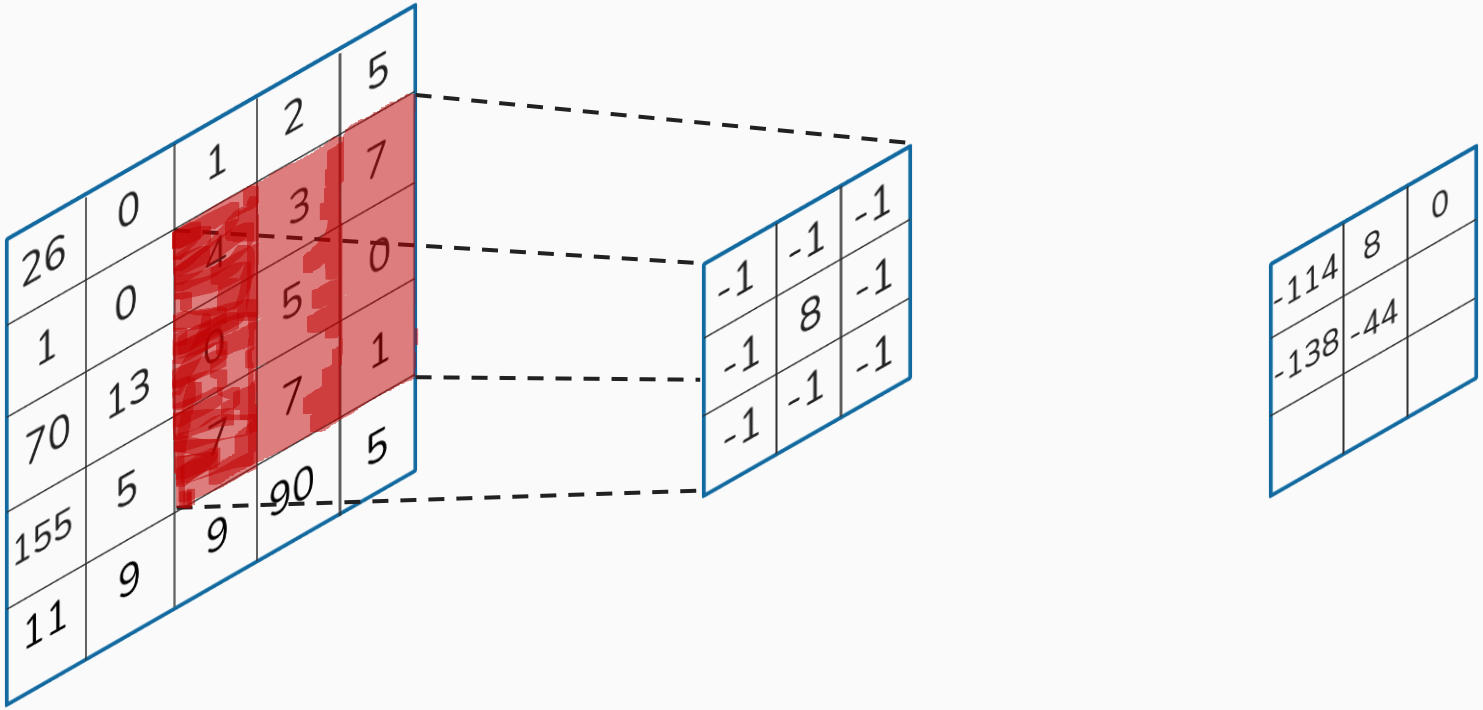

26
0
1
2
5
1
0
4
3
7
1
0
0
5
0
70
13
7
7
1
155
5
9
90
5
11
9
-1
-1
-1
-1
8
-1
-1
-1
-1

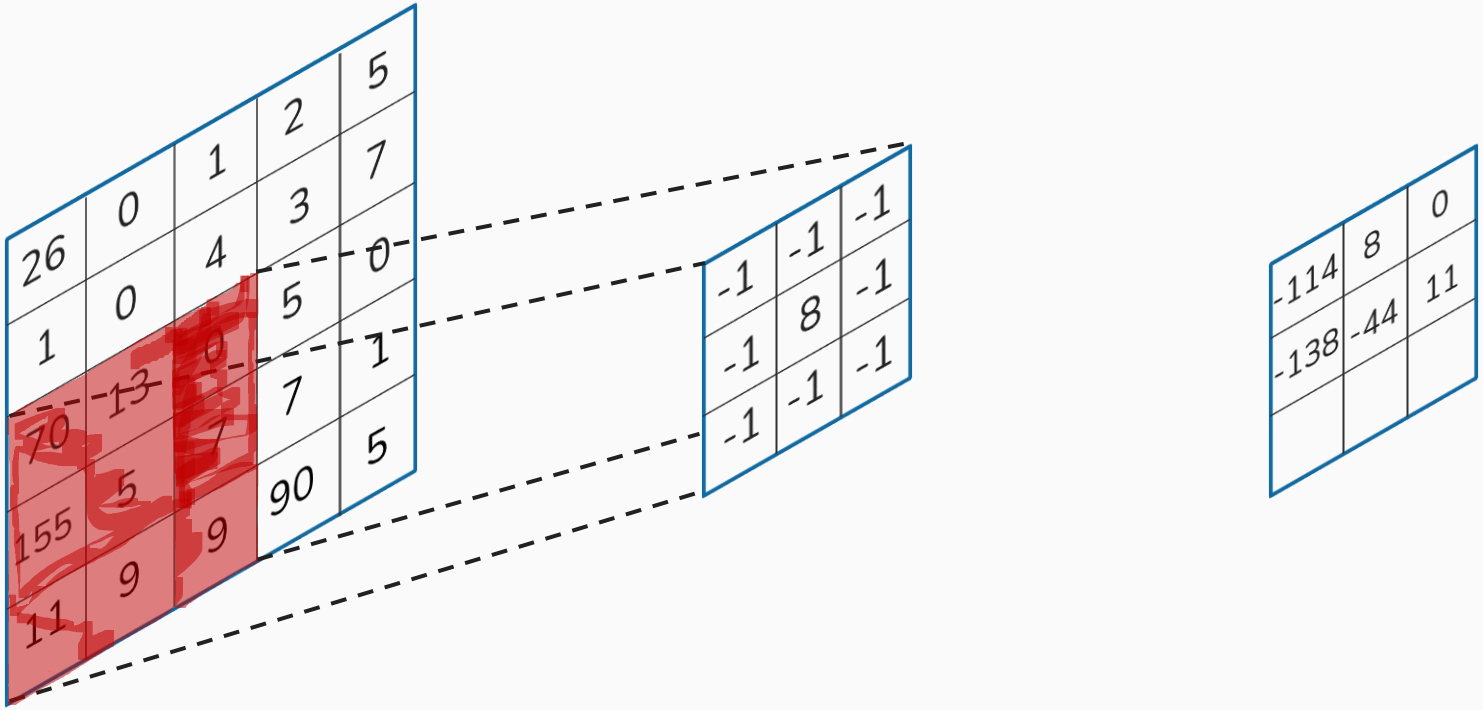

26 0 1 2 5
1 0 4 3 7
70 13 0 5 0
155 5 7 7 1
11 9 9 90 5
-1 -1 -1
-1 8 -1
-1 -1 -1
-114

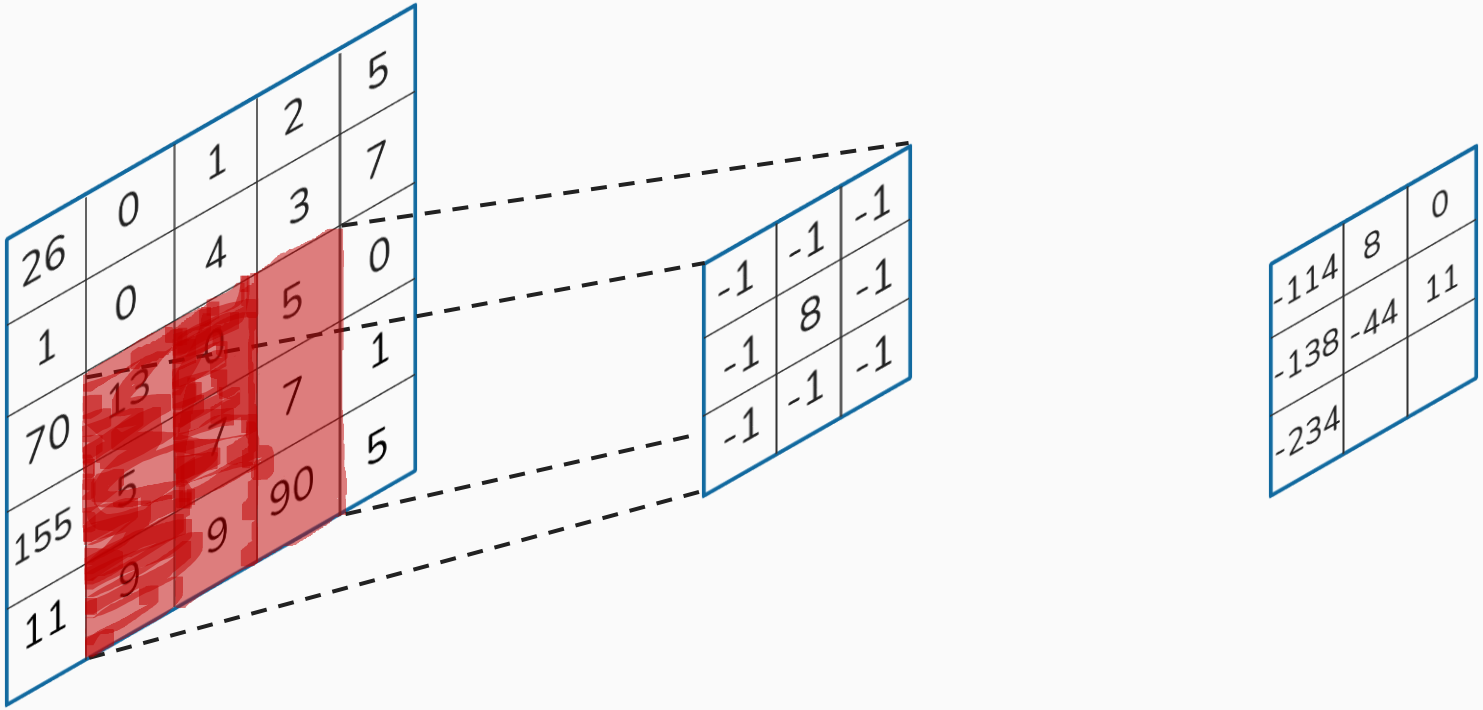

26 0 1 2 5
1 0 4 3 7
70 13 0 5 0
155 5 7 7 1
11 9 9 90 5
-1 -1 -1
-1 8 -1
-1 -1 -1
-114 8

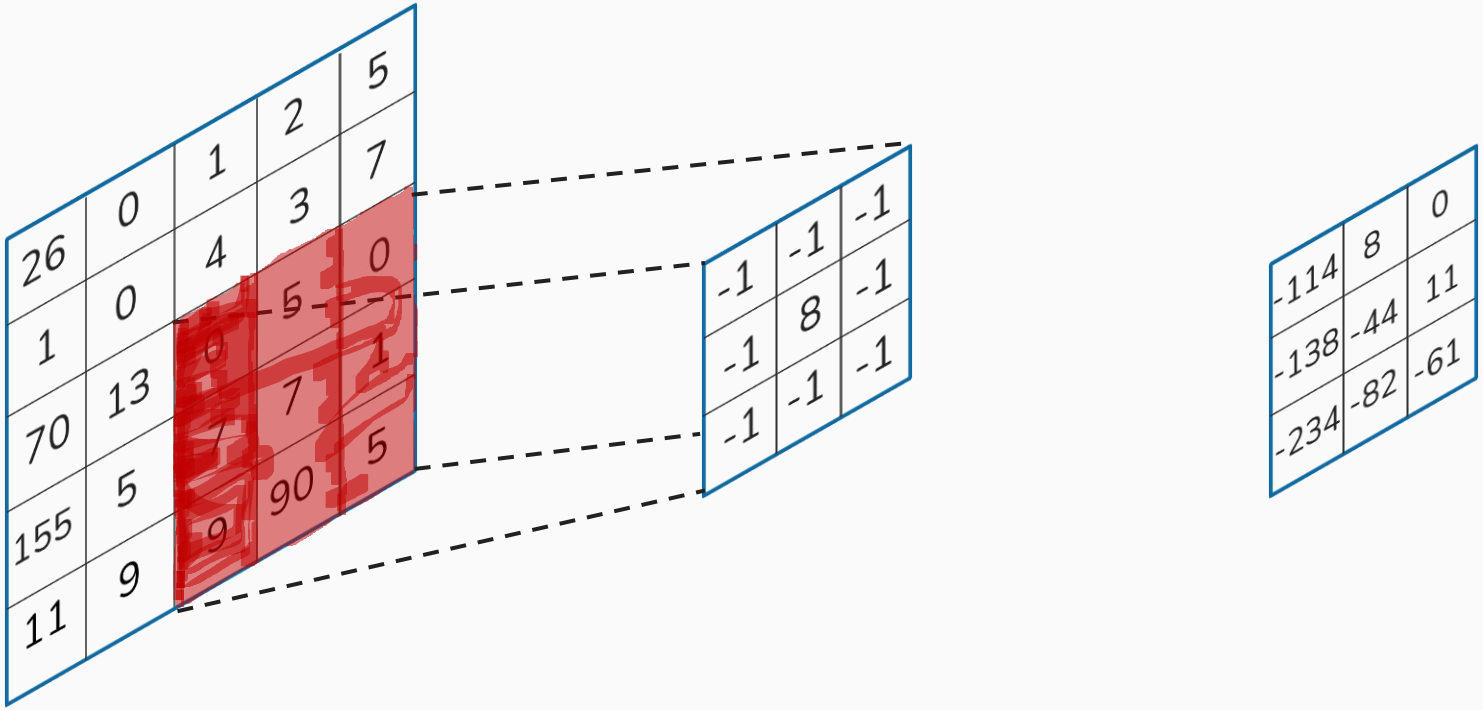

26 0 1 2 5
1 0 4 3 7
70 13 0 5 0
155 5 7 7 1
11 9 9 90 5
-1 -1 -1
-1 8 -1
-1 -1 -1
-114 8 0


26 0 1 2 5
1 0 4 3 7
70 13 0 5 0
155 5 7 7 1
11 9 9 90 5
-1 -1 -1
-1 8 -1
-1 -1 -1
-114 8 0
-138


26 0 1 2 5
1 0 4 3 7
70 13 0 5 0
155 5 7 7 1
11 9 9 90 5
-1 -1 -1
-1 8 -1
-1 -1 -1
-114 8 0
-138 -44

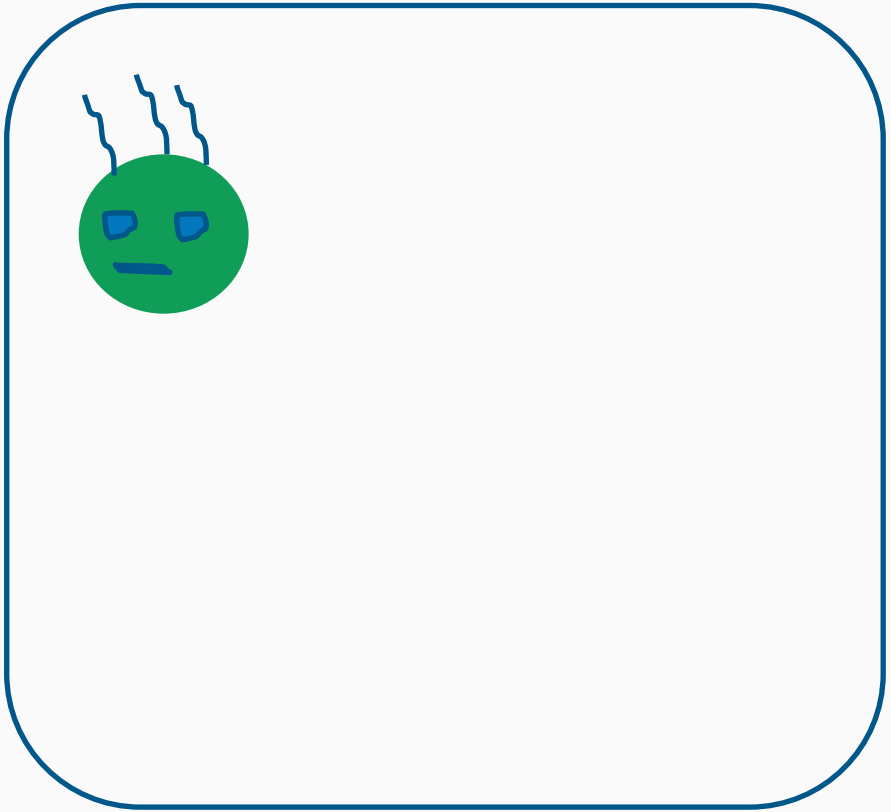

26 0 1 2 5
1 0 4 3 7
70 13 0 5 0
155 5 7 7 1
11 9 9 90 5
-1 -1 -1
-1 8 -1
-1 -1 -1
-114 8 0
-138 -44 11


26 0 1 2 5
1 0 4 3 7
70 13 0 5 0
155 5 7 7 1
11 9 9 90 5
-1 -1 -1
-1 8 -1
-1 -1 -1
-114 8 0
-138 -44 11
-234

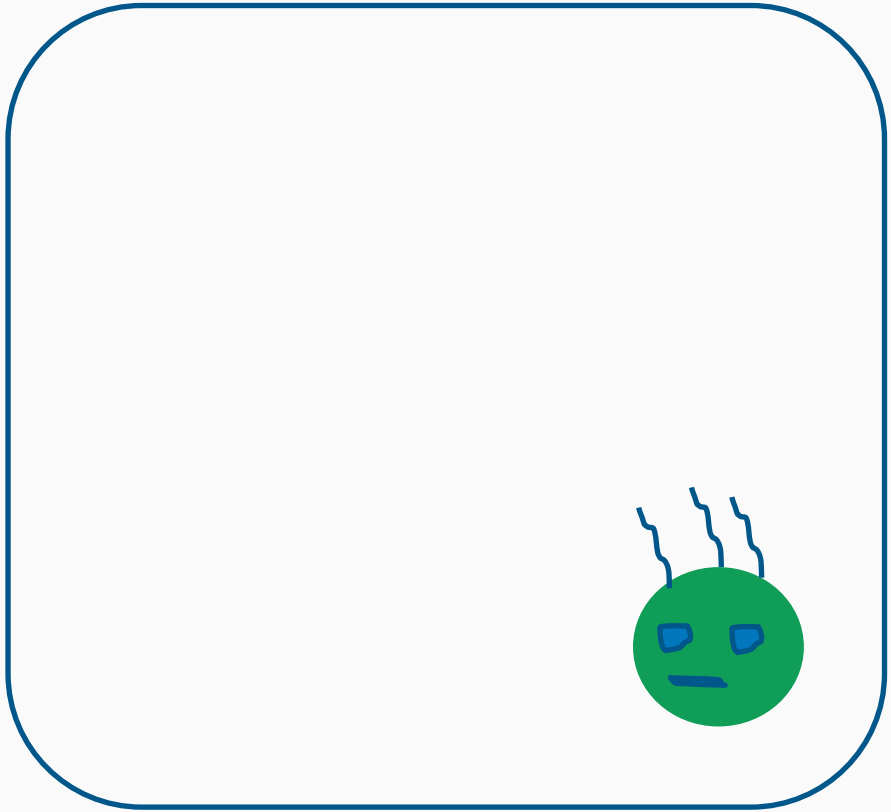

26 0 1 2 5
1 0 4 3 7
70 13 0 5 0
155 5 7 7 1
11 9 9 90 5
-1 -1 -1
-1 8 -1
-1 -1 -1
-114 8 0
-138 -44 11
-234 -82 -61

Convnets are translation invariant

image

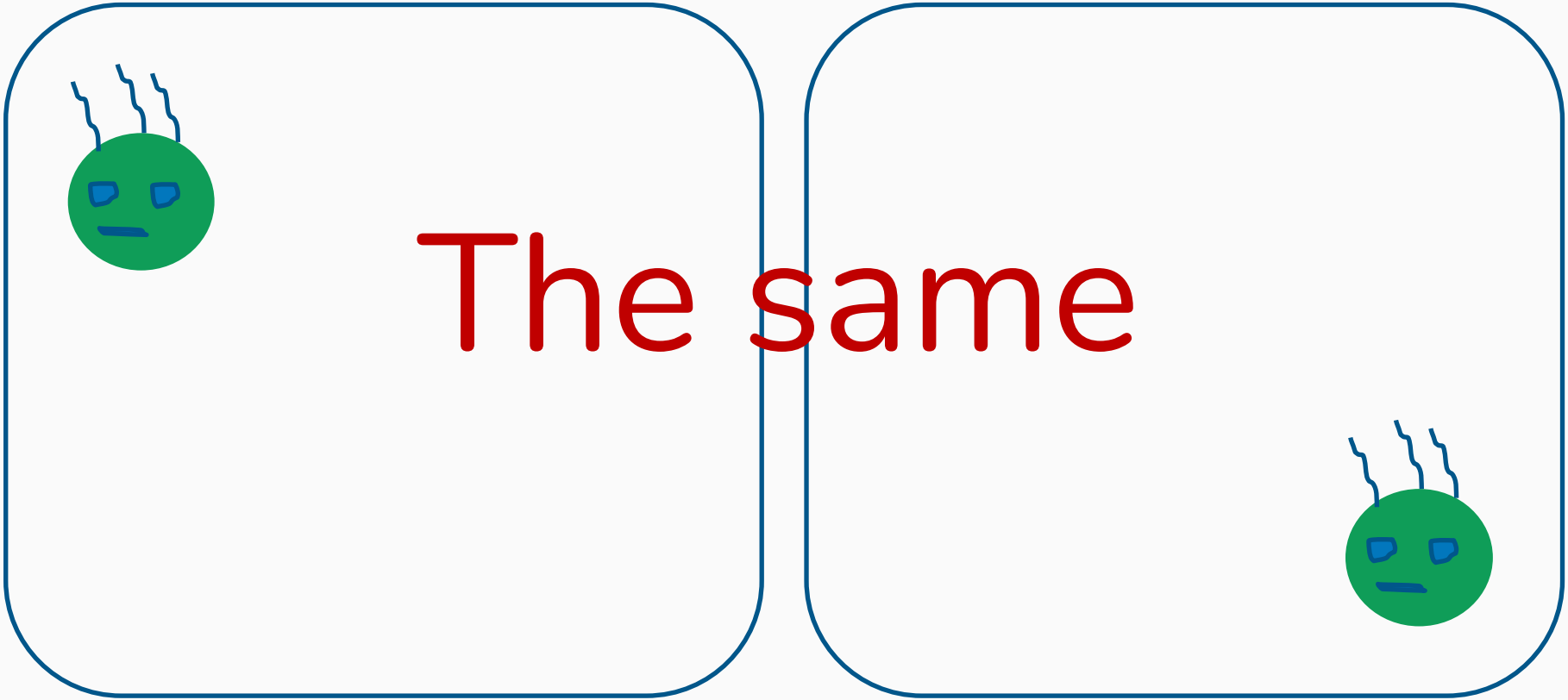


image



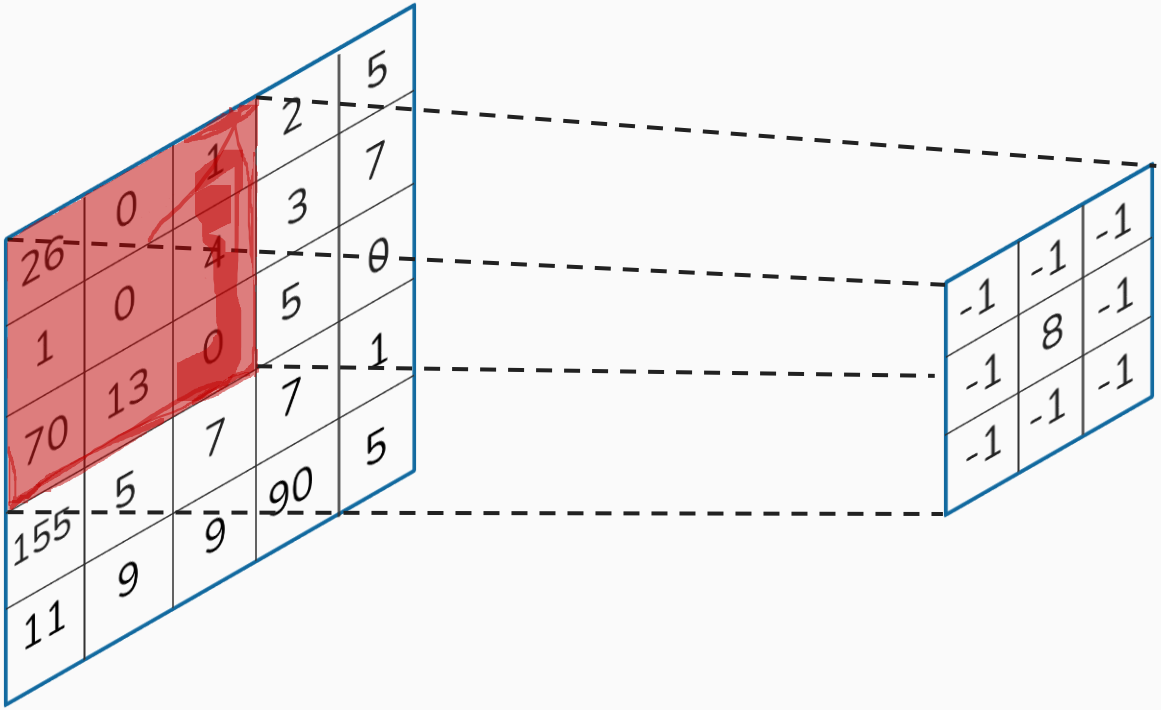
image

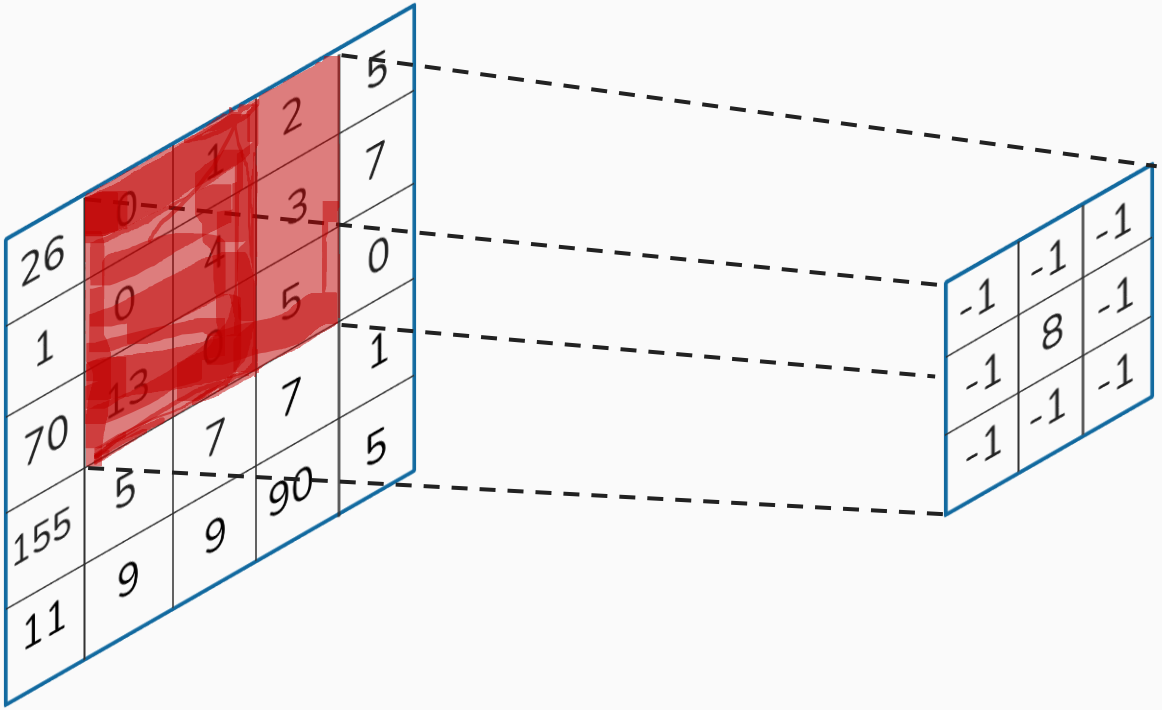
image



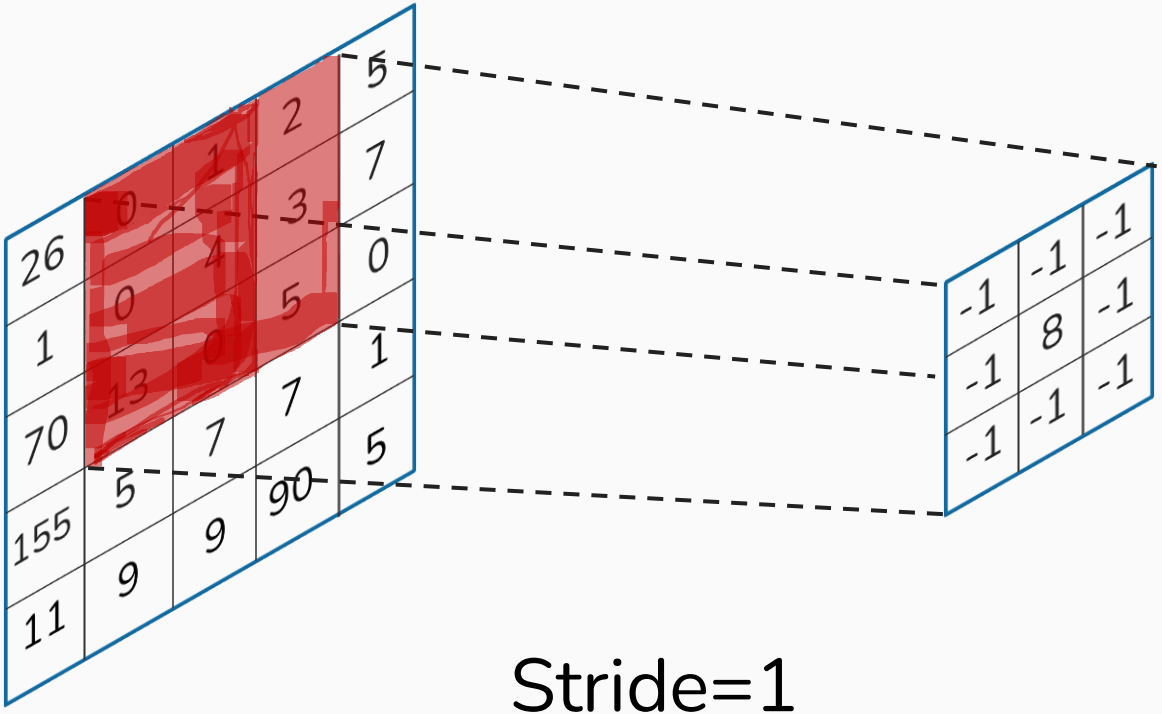
Weights ?

Filters

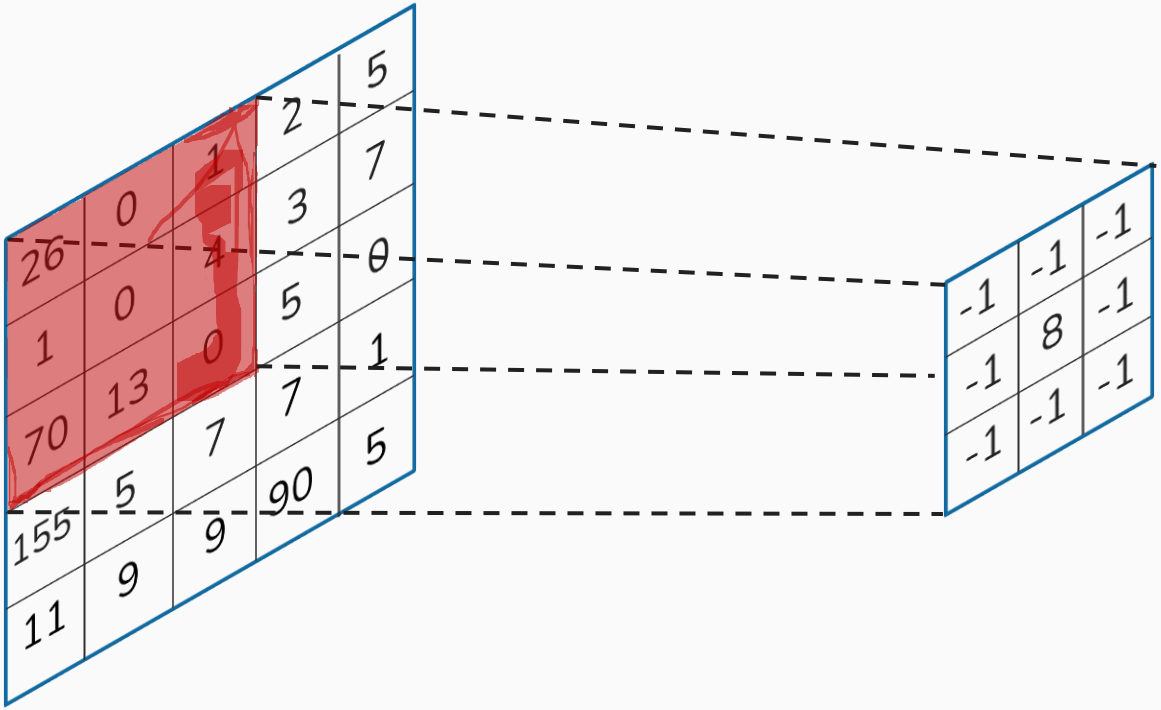

26
0
1
2
5
1
0
4
3
7
1
13
0
5
0
70
7
7
1
155
5
9
90
5
11
9
-1
-1
-1
-1
8
-1
-1
-1
-1

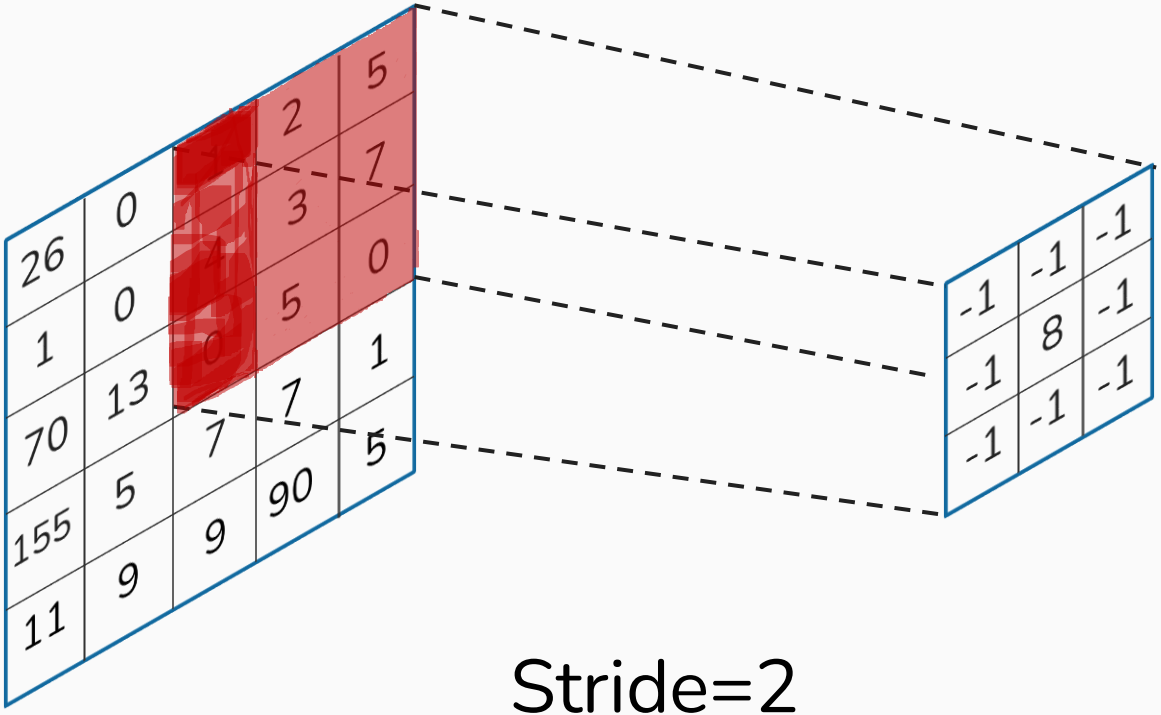

26
1
70
155
11
0
0
13
5
9
9
1
4
0
7
90
2
3
5
7
5
5
7
0
1
5
-1
-1
-1
-1
8
-1
-1
-1
-1

Stride= how much we move the filter at each step



Stride=1


26
0
1
2
5
1
0
4
3
7
1
13
0
5
0
70
7
7
1
155
5
9
90
5
11
9
-1
-1
-1
-1
8
-1
-1
-1
-1



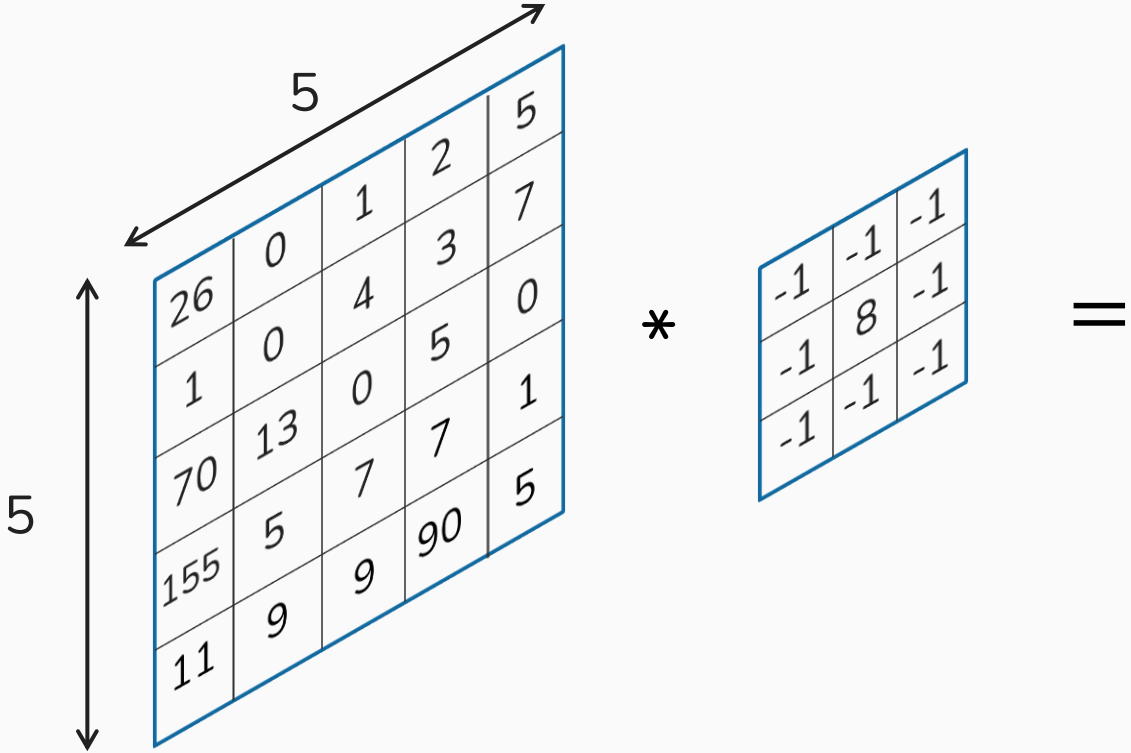
Stride=2

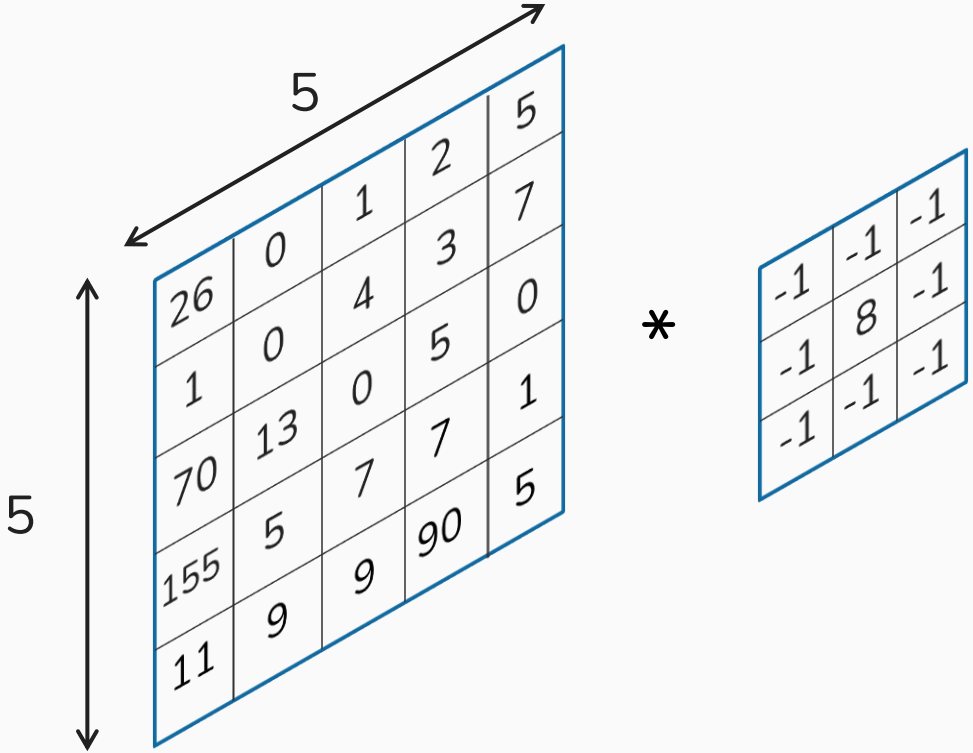
26	0	1	2	5
1	0	4	3	7
70	13	0	5	0
155	5	7	7	1
11	9	9	90	5

*

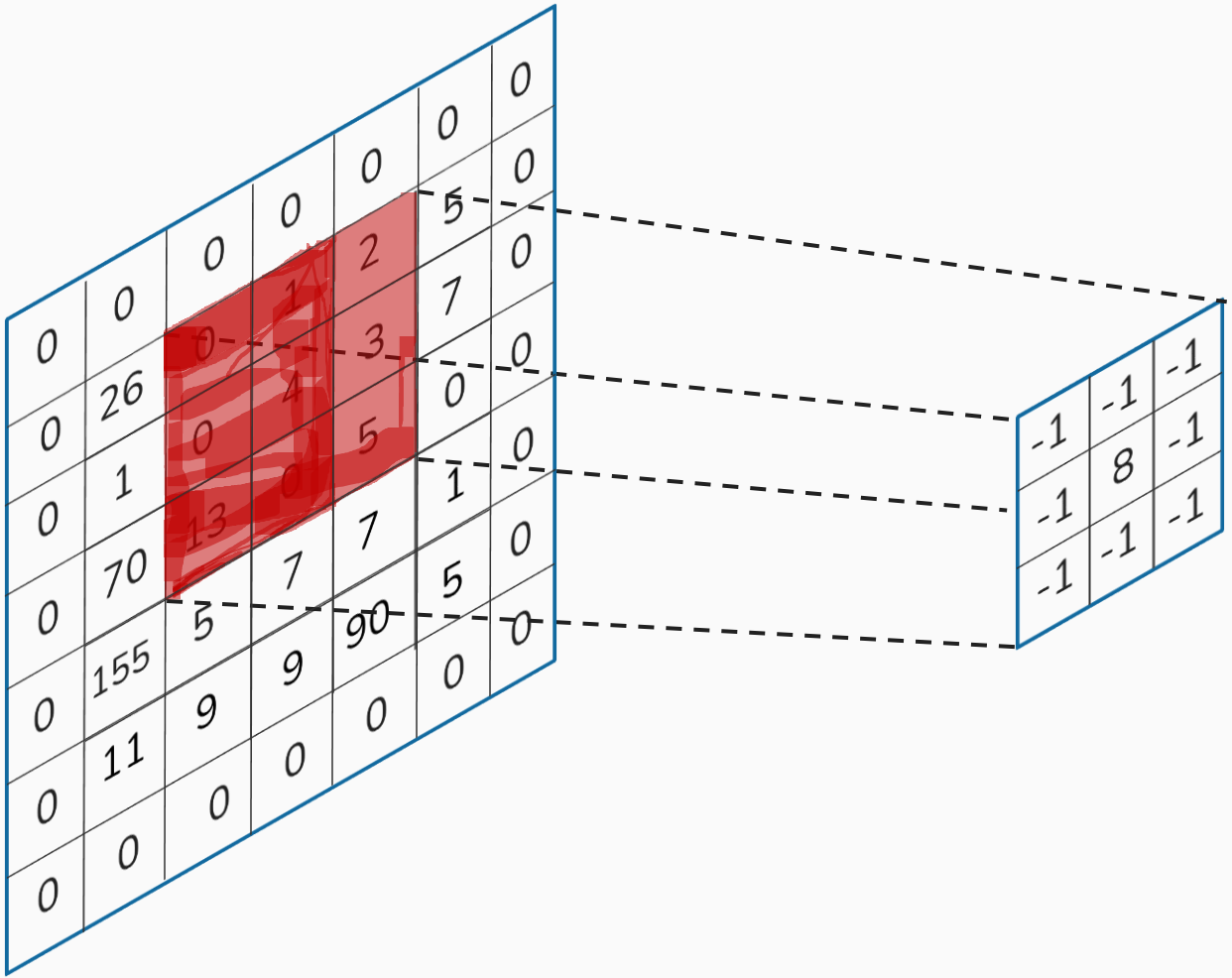
-1	-1	-1
-1	8	-1
-1	-1	-1

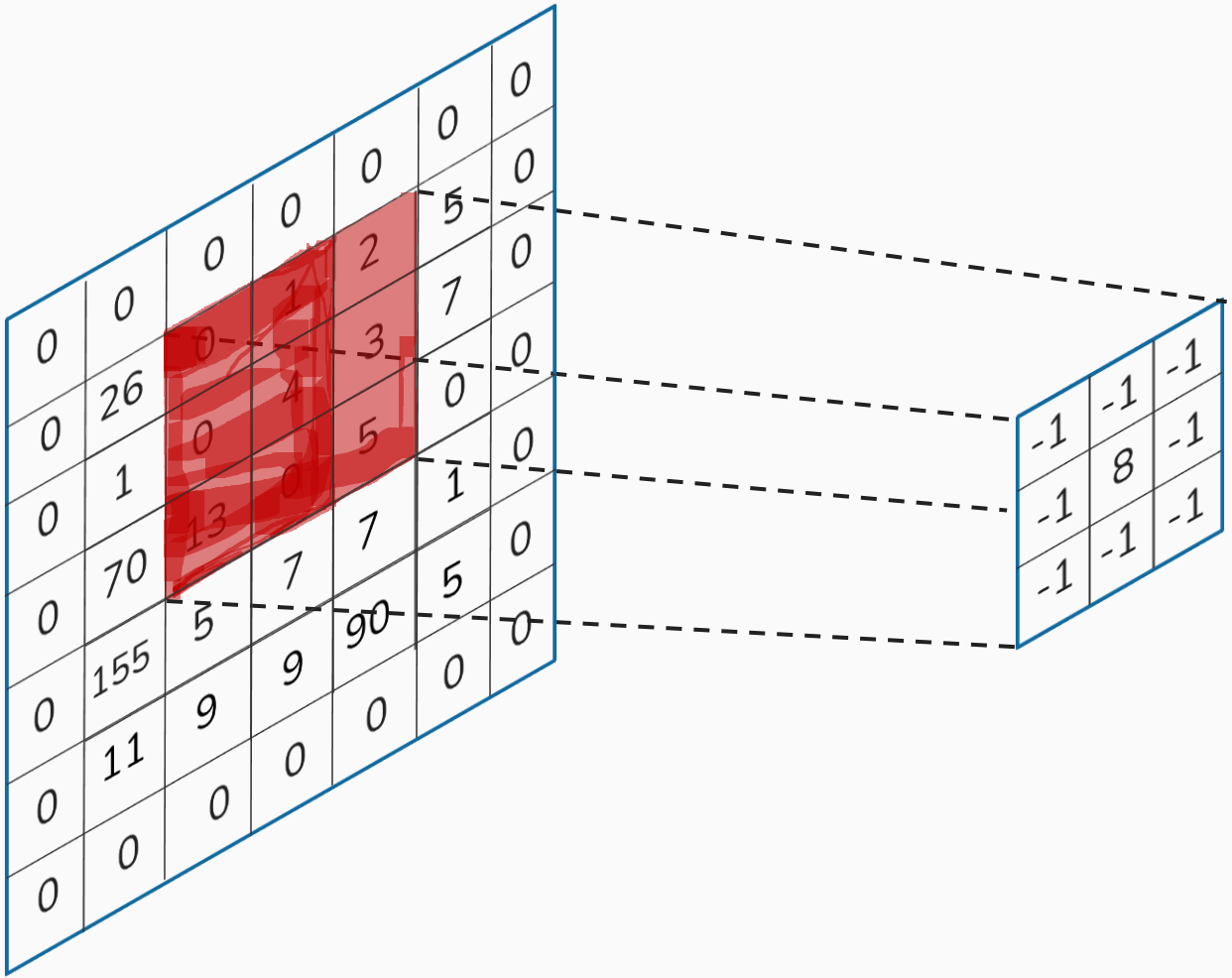
=


5
5
26 0 1 2 5
1 0 4 3 7
70 13 0 5 0
155 5 7 7 1
11 9 9 90 5
*
-1 -1 -1
-1 8 -1
-1 -1 -1
=

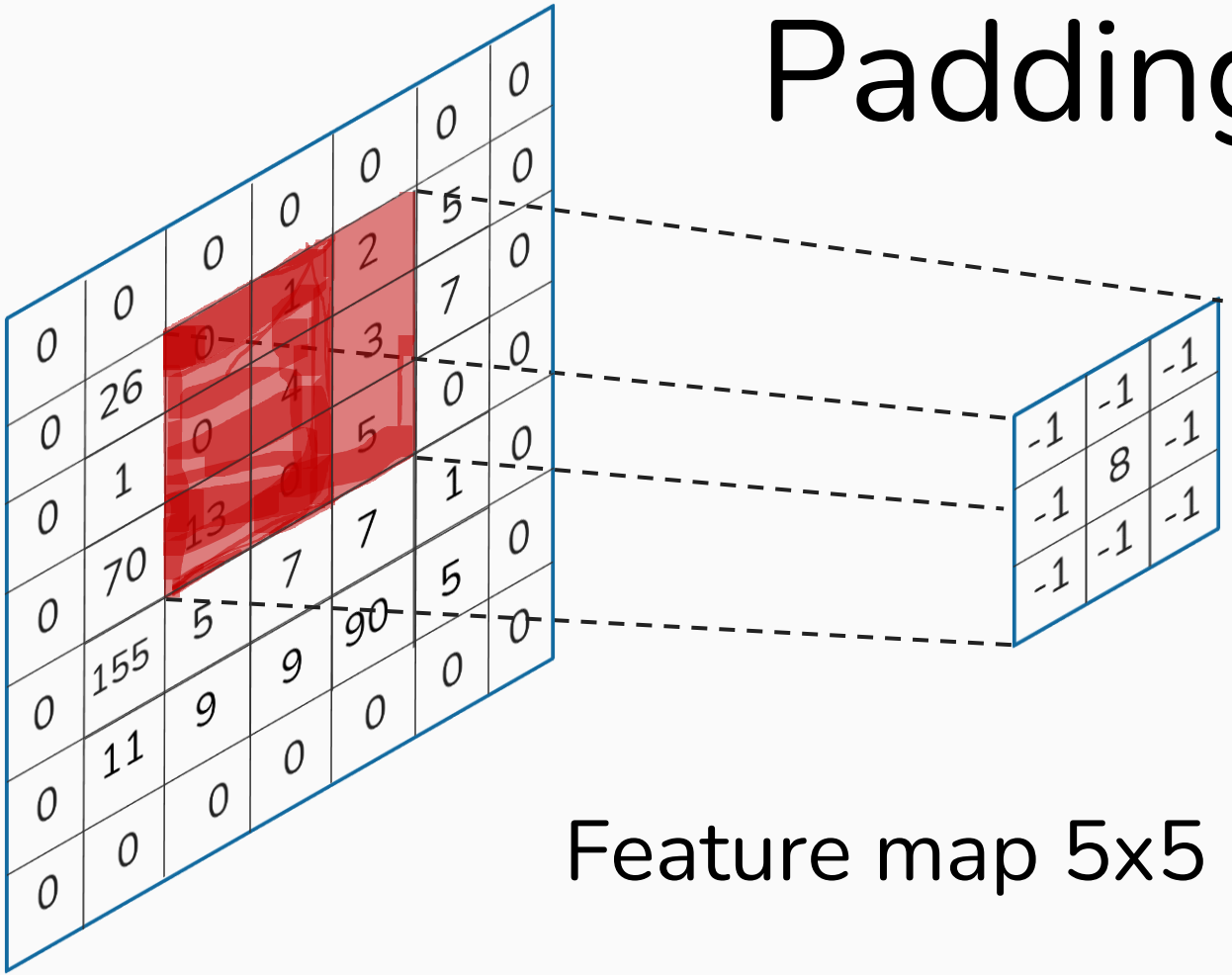


= Feature map 3x3


0 0 0 0 0 0
0 26 0 1 4 0
0 1 0 0 0 0
0 70 13 7 7 1
0 155 5 9 90 5
0 11 9 0 0 0
0 0 0 0 0 0
-1 -1 -1
-1 8 -1
-1 -1 -1


0 0 0 0 0 0
0 26 0 1 4 0
0 1 0 0 0 0
0 70 13 7 7 1
0 155 5 9 90 5
0 11 9 0 0 0
0 0 0 0 0 0
-1 -1 -1
-1 8 -1
-1 -1 -1

Padding



Feature map 5x5

5	0	5	0	5	7
15	7	81	5	2	5
3	9	0	1	3	7
2	13	0	4	5	0
26	70	11	0	7	1
1	155	5	9	90	5

*

-1	-1	-1
-1	8	-1
-1	-1	-1

5	0	5	0	5	7
15	7	81	5	2	5
3	9	0	1	3	7
2	13	0	4	5	0
26	70	11	0	7	1
1	155	5	9	90	5

*

-1	-1	-1
-1	8	-1
-1	-1	-1

=

-114	8	0
-138	-44	11
-234	-82	-61

5	0	5	0	5	7
15	7	81	5	2	5
3	9	0	1	3	7
2	13	0	4	5	0
26	70	11	0	7	1
1	155	5	9	90	5

*

-1	-1	-1
-1	8	-1
-1	-1	-1

=

-114	8	0
-138	-44	11
-234	-82	-61

→

5	0	5	0	5	7
15	7	81	5	2	5
3	9	0	1	3	7
2	13	0	4	5	0
26	70	11	0	7	1
1	155	5	9	90	5

$*$

-1	-1	-1
-1	8	-1
-1	-1	-1

$=$

-114	8	0
-138	-44	11
-234	-82	-61

$\longrightarrow$

$ReLU($

-114	8	0
-138	-44	11
-234	-82	-61

$) =$

$$
\begin{bmatrix} 5 & 0 & 5 & 0 & 5 & 7 \\ 15 & 7 & 81 & 5 & 2 & 5 \\ 3 & 9 & 0 & 1 & 3 & 7 \\ 2 & 13 & 0 & 4 & 5 & 0 \\ 26 & 70 & 11 & 0 & 7 & 1 \\ 1 & 155 & 5 & 9 & 90 & 5 \end{bmatrix}
*
\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}
=
\begin{bmatrix} -114 & 8 & 0 \\ -138 & -44 & 11 \\ -234 & -82 & -61 \end{bmatrix}
\longrightarrow
$$

$$
ReLU\left(\begin{bmatrix} -114 & 8 & 0 \\ -138 & -44 & 11 \\ -234 & -82 & -61 \end{bmatrix}\right) = \begin{bmatrix} 0 & 8 & 0 \\ 0 & 0 & 11 \\ 0 & 0 & 0 \end{bmatrix}
$$

5	0	5	0	5	7
15	7	81	5	2	5
3	9	0	1	3	7
2	13	0	4	5	0
26	70	11	0	7	1
1	155	5	9	90	5

$*$

-1	-1	-1
-1	8	-1
-1	-1	-1

$=$

-114	8	0
-138	-44	11
-234	-82	-61

$\longrightarrow$

$ReLU($

-114	8	0
-138	-44	11
-234	-82	-61

$) =$

0	8	0
0	0	11
0	0	0

5	0	5	0	5	7
15	7	81	5	2	5
3	9	0	1	3	7
2	13	0	4	5	0
26	70	11	0	7	1
1	155	5	9	90	5

$*$

-1	-1	-1
-1	8	-1
-1	-1	-1

$=$

-114	8	0
-138	-44	11
-234	-82	-61

$ReLU($

-114	8	0
-138	-44	11
-234	-82	-61

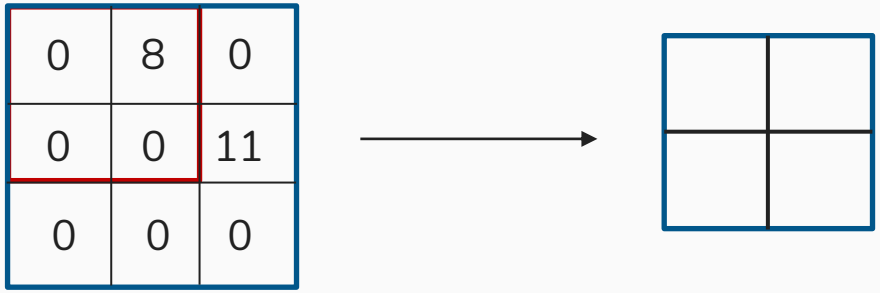
$)$ $=$

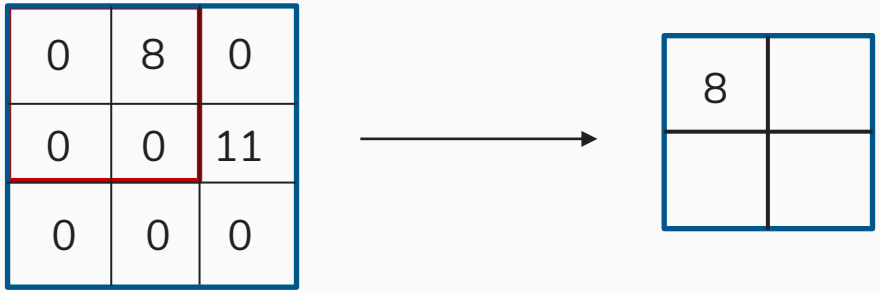
0	8	0
0	0	11
0	0	0

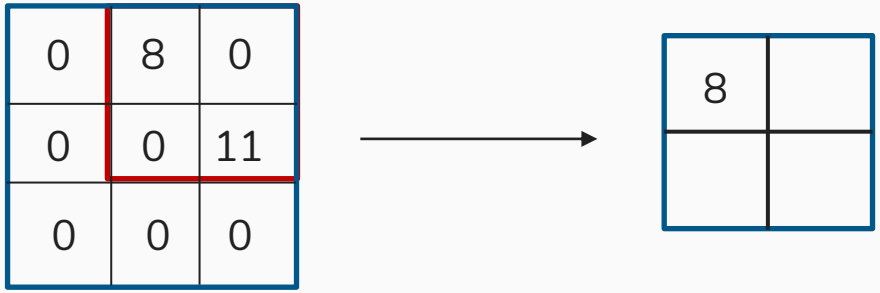
Feature map

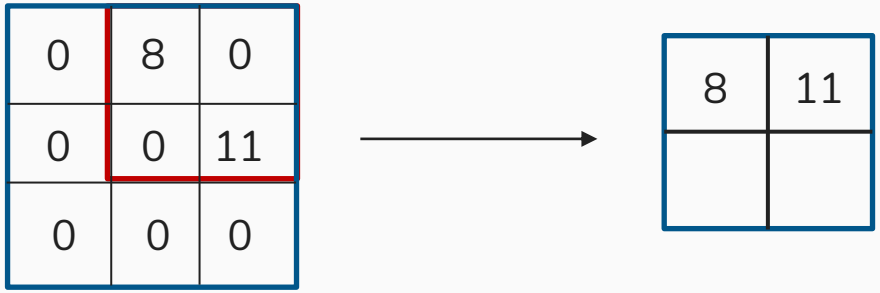
POOLING

0	8	0
0	0	11
0	0	0


0
8
0
0
0
11
0
0
0


0
8
0
0
0
11
0
0
0
8


0
8
0
0
0
11
0
0
0
8


0
8
0
0
0
11
0
0
0
8
11

0	8	0
0	0	11
0	0	0

8	11

0	8	0
0	0	11
0	0	0

8	11
0	

0	8	0
0	0	11
0	0	0

8	11
0	

0	8	0
0	0	11
0	0	0

8	11
0	11

Max Pooling

0	8	0
0	0	11
0	0	0

→

8	11
0	11

Max Pooling

0	8	0
0	0	11
0	0	0

→

8	11
0	11

Max Pooling

0	8	0
0	0	11
0	0	0

8	11
0	11

Max Pooling

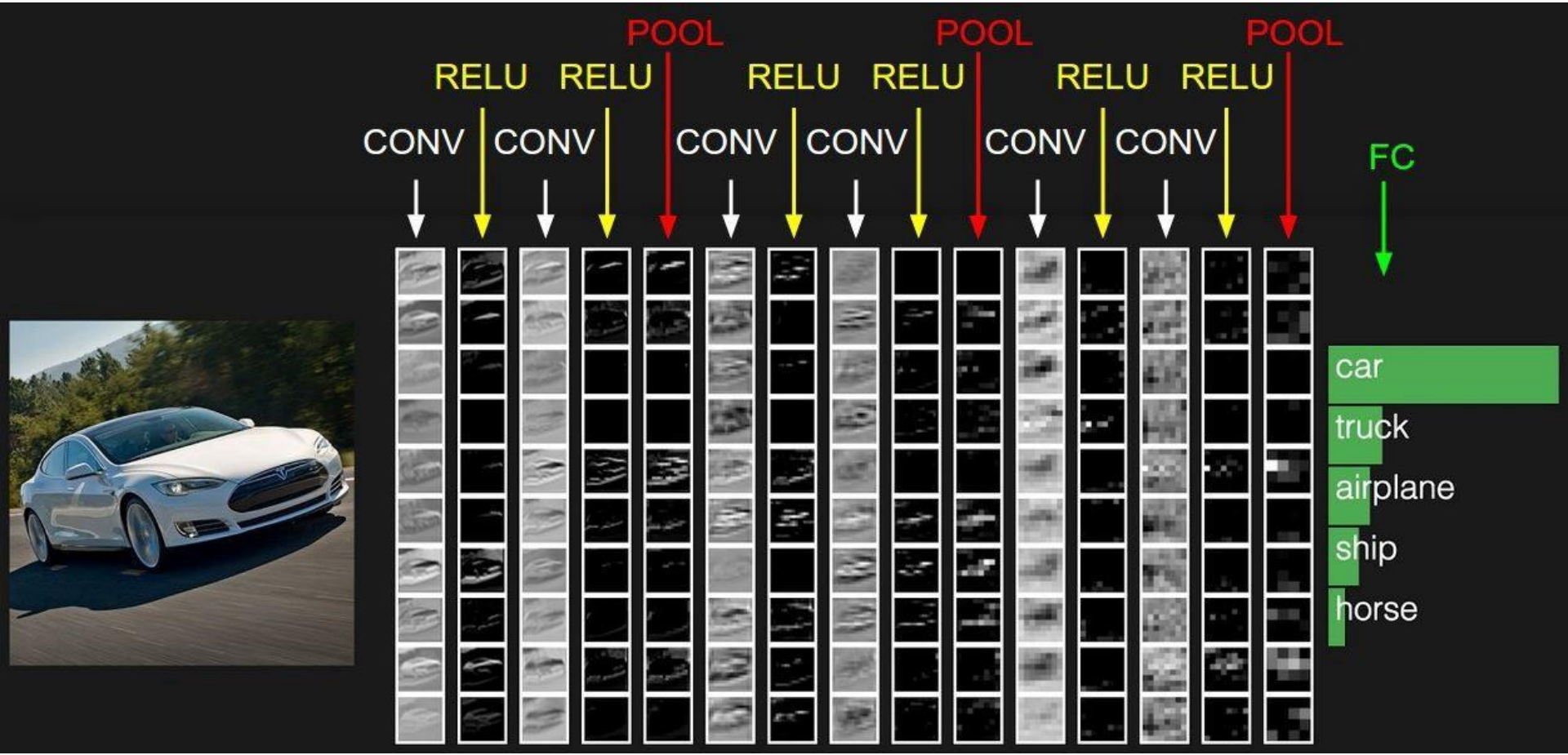
0	8	0
0	0	11
0	0	0

8	11
0	11

Why pooling ?

1-reduce # of paramters

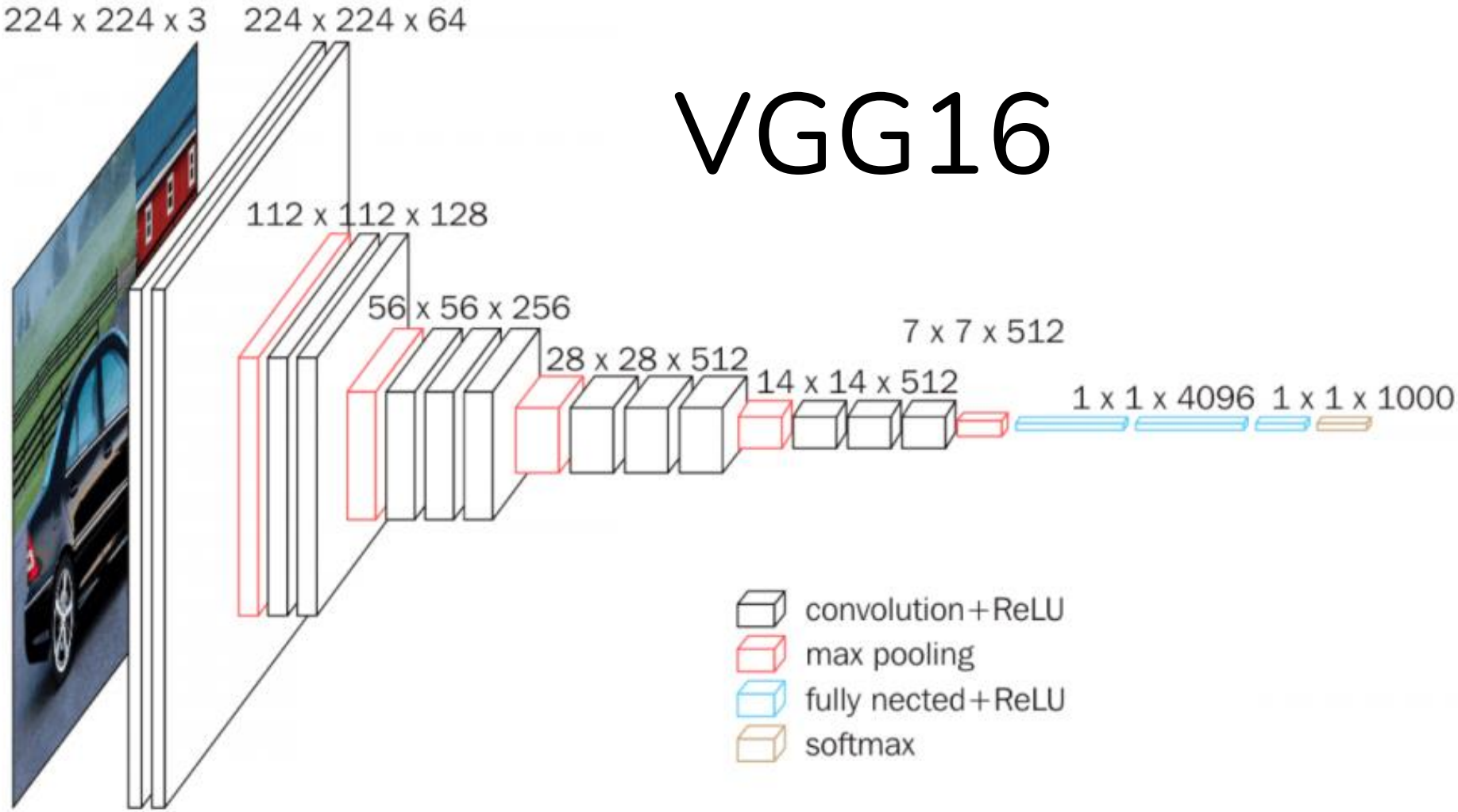
1-reduce # of paramters
2-invariance to local translation


POOL
POOL
POOL
RELU
RELU
RELU
RELU
RELU
RELU
CONV
CONV
CONV
CONV
CONV
CONV
FC
car
truck
airplane
ship
horse

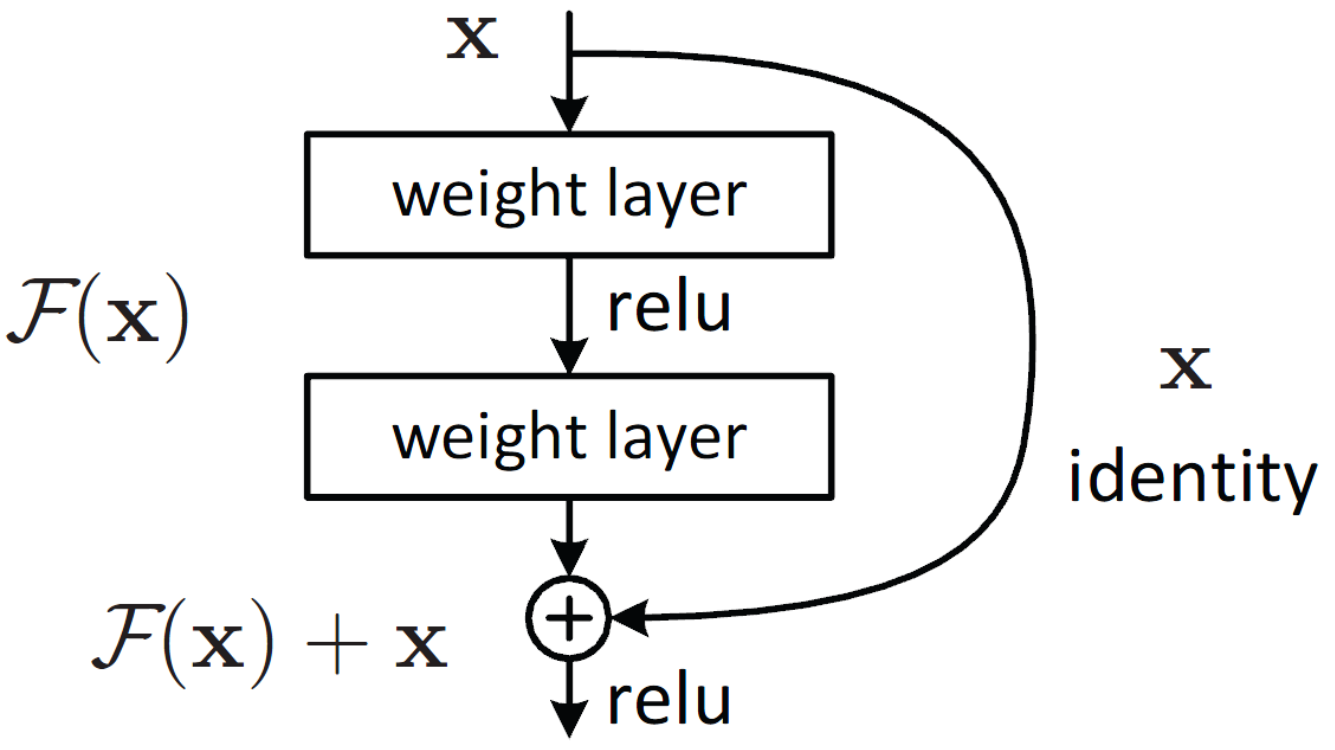
Transfer learning

What has been learned in a distribution P_1 is exploited to improve generalization in another distribution P_2

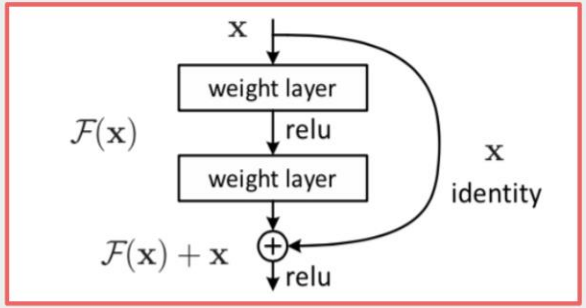
VGG16


224 x 224 x 3
224 x 224 x 64
112 x 112 x 128
56 x 56 x 256
28 x 28 x 512
14 x 14 x 512
7 x 7 x 512
1 x 1 x 4096
1 x 1 x 1000
convolution+ReLU
max pooling
fully nected+ReLU
softmax

Residual learning

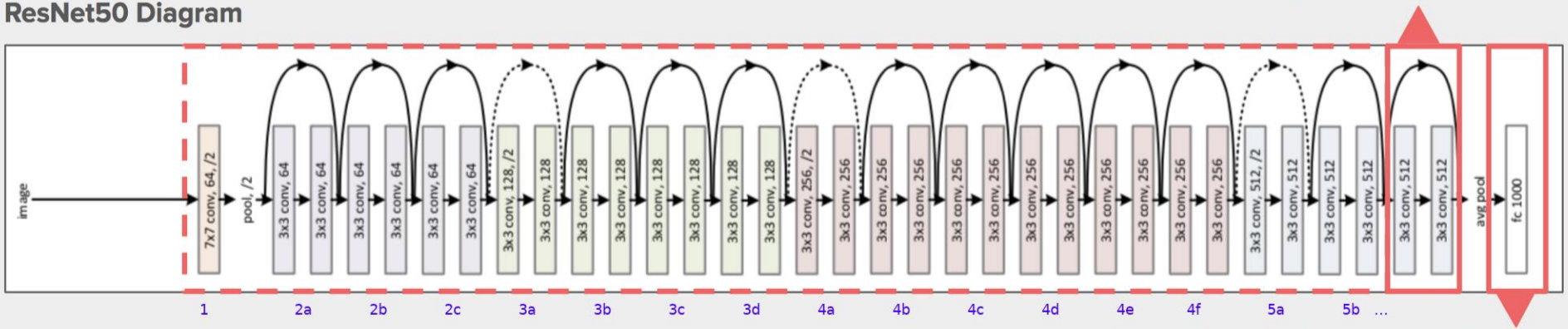


Retrain ResNet50

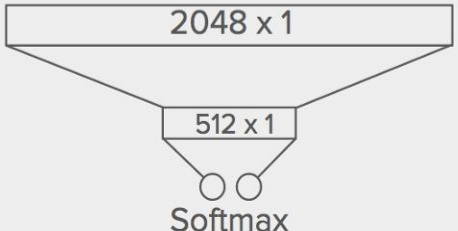


Residual Learning Block

ResNet50 Diagram



Re-architect fully-connected layers



end