

Convolutional Neural Networks

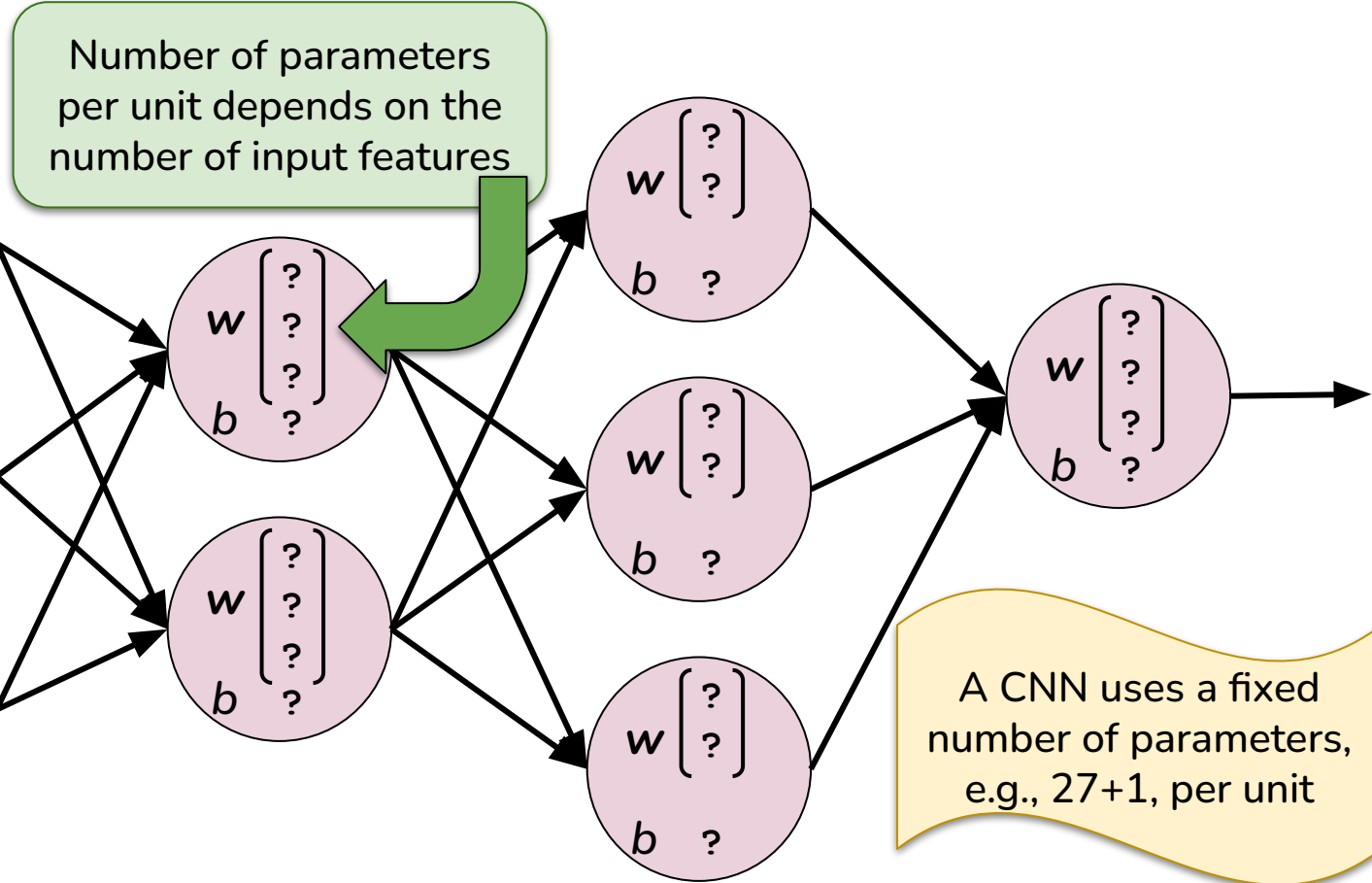
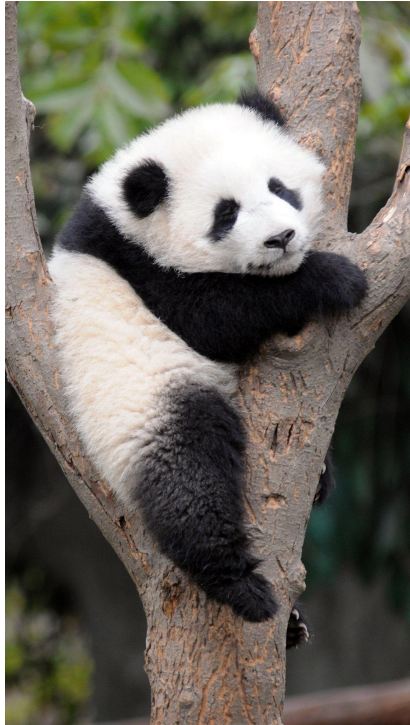


CS344
Deep Learning



Why Convolutional Neural Networks (CNNs)?

(2484, 4416, 3)



Convolution

3	7	9	1	0	4
8	1	3	3	3	0
7	4	6	4	4	6
4	4	1	9	3	6
9	8	4	9	4	3
7	5	2	3	3	1

*

1	0	-1
1	0	-1
1	0	-1

=

Convolution

¹ 3	⁰ 7	⁻¹ 9	1	0	4
¹ 8	⁰ 1	⁻¹ 3	3	3	0
¹ 7	⁰ 4	⁻¹ 6	4	4	6
4	4	1	9	3	6
9	8	4	9	4	3
7	5	2	3	3	1

*

1	0	-1
1	0	-1
1	0	-1

=

0			

$$1*3 + 0*7 + -1*9 + 1*8 + 0*1 + -1*3 + 1*7 + 0*4 + -1*6 = 0$$

Convolution

3	¹ 7	⁰ 9	⁻¹ 1	0	4
8	¹ 1	⁰ 3	⁻¹ 3	3	0
7	¹ 4	⁰ 6	⁻¹ 4	4	6
4	4	1	9	3	6
9	8	4	9	4	3
7	5	2	3	3	1

*

1	0	-1
1	0	-1
1	0	-1

=

0	4		

$$1*7 + 0*9 + -1*1 + 1*1 + 0*3 + -1*3 + 1*4 + 0*6 + -1*4 = 4$$

Convolution

3	7	¹ 9	⁰ 1	⁻¹ 0	4
8	1	¹ 3	⁰ 3	⁻¹ 3	0
7	4	¹ 6	⁰ 4	⁻¹ 4	6
4	4	1	9	3	6
9	8	4	9	4	3
7	5	2	3	3	1

*

1	0	-1
1	0	-1
1	0	-1

=

0	4	11	

$$1*9 + 0*1 + -1*0 + 1*3 + 0*3 + -1*3 + 1*6 + 0*4 + -1*4 = 11$$

Convolution

3	7	9	¹ 1	⁰ 0	⁻¹ 4
8	1	3	¹ 3	⁰ 3	⁻¹ 0
7	4	6	¹ 4	⁰ 4	⁻¹ 6
4	4	1	9	3	6
9	8	4	9	4	3
7	5	2	3	3	1

*

1	0	-1
1	0	-1
1	0	-1

=

0	4	11	-2

$$1*1 + 0*0 + -1*4 + 1*3 + 0*3 + -1*0 + 1*4 + 0*4 + -1*6 = -2$$

Convolution

3	7	9	1	0	4
¹ 8	⁰ 1	⁻¹ 3	3	3	0
¹ 7	⁰ 4	⁻¹ 6	4	4	6
¹ 4	⁰ 4	⁻¹ 1	9	3	6
9	8	4	9	4	3
7	5	2	3	3	1

*

1	0	-1
1	0	-1
1	0	-1

=

0	4	11	-2
9			

$$1*8 + 0*1 + -1*3 + 1*7 + 0*4 + -1*6 + 1*4 + 0*4 + -1*1 = 9$$

Convolution

3	7	9	1	0	4
8	¹ 1	⁰ 3	⁻¹ 3	3	0
7	¹ 4	⁰ 6	⁻¹ 4	4	6
4	¹ 4	⁰ 1	⁻¹ 9	3	6
9	8	4	9	4	3
7	5	2	3	3	1

*

1	0	-1
1	0	-1
1	0	-1

=

0	4	11	-2
9	-7		

$$1*1 + 0*3 + -1*3 + 1*4 + 0*6 + -1*4 + 1*4 + 0*1 + -1*9 = -7$$

Convolution

3	7	9	1	0	4
8	1	¹ 3	⁰ 3	⁻¹ 3	0
7	4	¹ 6	⁰ 4	⁻¹ 4	6
4	4	¹ 1	⁰ 9	⁻¹ 3	6
9	8	4	9	4	3
7	5	2	3	3	1

*

1	0	-1
1	0	-1
1	0	-1

=

0	4	11	-2
9	-7	0	

$$1*3 + 0*3 + -1*3 + 1*6 + 0*4 + -1*4 + 1*1 + 0*9 + -1*3 = 0$$

Convolution

3	7	9	1	0	4
8	1	3	¹ 3	⁰ 3	⁻¹ 0
7	4	6	¹ 4	⁰ 4	⁻¹ 6
4	4	1	¹ 9	⁰ 3	⁻¹ 6
9	8	4	9	4	3
7	5	2	3	3	1

*

1	0	-1
1	0	-1
1	0	-1

=

0	4	11	-2
9	-7	0	4

$$1*3 + 0*3 + -1*0 + 1*4 + 0*4 + -1*6 + 1*9 + 0*3 + -1*6 = 4$$

Convolution

3	7	9	1	0	4
8	1	3	3	3	0
¹ 7	⁰ 4	⁻¹ 6	4	4	6
¹ 4	⁰ 4	⁻¹ 1	9	3	6
¹ 9	⁰ 8	⁻¹ 4	9	4	3
7	5	2	3	3	1

*

1	0	-1
1	0	-1
1	0	-1

=

0	4	11	-2
9	-7	0	4
9			

$$1*7 + 0*4 + -1*6 + 1*4 + 0*4 + -1*1 + 1*9 + 0*8 + -1*4 = 9$$

Convolution

3	7	9	1	0	4
8	1	3	3	3	0
7	¹ 4	⁰ 6	⁻¹ 4	4	6
4	¹ 4	⁰ 1	⁻¹ 9	3	6
9	¹ 8	⁰ 4	⁻¹ 9	4	3
7	5	2	3	3	1

*

1	0	-1
1	0	-1
1	0	-1

=

0	4	11	-2
9	-7	0	4
9	-6		

$$1*4 + 0*6 + -1*4 + 1*4 + 0*1 + -1*9 + 1*8 + 0*4 + -1*9 = -6$$

Convolution

3	7	9	1	0	4
8	1	3	3	3	0
7	4	¹ 6	⁰ 4	⁻¹ 4	6
4	4	¹ 1	⁰ 9	⁻¹ 3	6
9	8	¹ 4	⁰ 9	⁻¹ 4	3
7	5	2	3	3	1

*

1	0	-1
1	0	-1
1	0	-1

=

0	4	11	-2
9	-7	0	4
9	-6	0	

$$1*6 + 0*4 + -1*4 + 1*1 + 0*9 + -1*3 + 1*4 + 0*9 + -1*4 = 0$$

Convolution

3	7	9	1	0	4
8	1	3	3	3	0
7	4	6	¹ 4	⁰ 4	⁻¹ 6
4	4	1	¹ 9	⁰ 3	⁻¹ 6
9	8	4	¹ 9	⁰ 4	⁻¹ 3
7	5	2	3	3	1

*

1	0	-1
1	0	-1
1	0	-1

=

0	4	11	-2
9	-7	0	4
9	-6	0	7

$$1*4 + 0*4 + -1*6 + 1*9 + 0*3 + -1*6 + 1*9 + 0*4 + -1*3 = 7$$

Convolution

3	7	9	1	0	4
8	1	3	3	3	0
7	4	6	4	4	6
¹ 4	⁰ 4	⁻¹ 1	9	3	6
¹ 9	⁰ 8	⁻¹ 4	9	4	3
¹ 7	⁰ 5	⁻¹ 2	3	3	1

*

1	0	-1
1	0	-1
1	0	-1

=

0	4	11	-2
9	-7	0	4
9	-6	0	7
13			

$$1*4 + 0*4 + -1*1 + 1*9 + 0*8 + -1*4 + 1*7 + 0*5 + -1*2 = 13$$

Convolution

3	7	9	1	0	4
8	1	3	3	3	0
7	4	6	4	4	6
4	¹ 4	⁰ 1	⁻¹ 9	3	6
9	¹ 8	⁰ 4	⁻¹ 9	4	3
7	¹ 5	⁰ 2	⁻¹ 3	3	1

*

1	0	-1
1	0	-1
1	0	-1

=

0	4	11	-2
9	-7	0	4
9	-6	0	7
13	-4		

$$1*4 + 0*1 + -1*9 + 1*8 + 0*4 + -1*9 + 1*5 + 0*2 + -1*3 = -4$$

Convolution

3	7	9	1	0	4
8	1	3	3	3	0
7	4	6	4	4	6
4	4	¹ 1	⁰ 9	⁻¹ 3	6
9	8	¹ 4	⁰ 9	⁻¹ 4	3
7	5	¹ 2	⁰ 3	⁻¹ 3	1

*

1	0	-1
1	0	-1
1	0	-1

=

0	4	11	-2
9	-7	0	4
9	-6	0	7
13	-4	-3	

$$1*1 + 0*9 + -1*3 + 1*4 + 0*9 + -1*4 + 1*2 + 0*3 + -1*3 = -3$$

Convolution

3	7	9	1	0	4
8	1	3	3	3	0
7	4	6	4	4	6
4	4	1	¹ 9	⁰ 3	⁻¹ 6
9	8	4	¹ 9	⁰ 4	⁻¹ 3
7	5	2	¹ 3	⁰ 3	⁻¹ 1

*

1	0	-1
1	0	-1
1	0	-1

=

0	4	11	-2
9	-7	0	4
9	-6	0	7
13	-4	-3	11

$$1*9 + 0*3 + -1*6 + 1*9 + 0*4 + -1*3 + 1*3 + 0*3 + -1*1 = 11$$

Filter

3	7	9	1	0	4
8	1	3	3	3	0
7	4	6	4	4	6
4	4	1	9	3	6
9	8	4	9	4	3
7	5	2	3	3	1

Filter (kernel)



1	0	-1
1	0	-1
1	0	-1

*

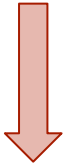
=

0	4	11	-2
9	-7	0	4
9	-6	0	7
13	-4	-3	11

Unit of
convolutional layer -
parameters W (or f)

Stride

Stride of 1



3	7	9	1	0	4
8	1	3	3	3	0
7	4	6	4	4	6
4	4	1	9	3	6
9	8	4	9	4	3
7	5	2	3	3	1

*

1	0	-1
1	0	-1
1	0	-1

=

0	4	11	-2
9	-7	0	4
9	-6	0	7
13	-4	-3	11

Stride

Stride of 2



¹ 3	⁰ 7	⁻¹ 9	1	0	4
¹ 8	⁰ 1	⁻¹ 3	3	3	0
¹ 7	⁰ 4	⁻¹ 6	4	4	6
4	4	1	9	3	6
9	8	4	9	4	3
7	5	2	3	3	1

*

1	0	-1
1	0	-1
1	0	-1

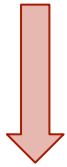
=

0	

$$1*3 + 0*7 + -1*9 + 1*8 + 0*1 + -1*3 + 1*7 + 0*4 + -1*6 = 0$$

Stride

Stride of 2



3	7	¹ 9	⁰ 1	⁻¹ 0	4
8	1	¹ 3	⁰ 3	⁻¹ 3	0
7	4	¹ 6	⁰ 4	⁻¹ 4	6
4	4	1	9	3	6
9	8	4	9	4	3
7	5	2	3	3	1

*

1	0	-1
1	0	-1
1	0	-1

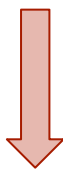
=

0	11

$$1*9 + 0*1 + -1*0 + 1*3 + 0*3 + -1*3 + 1*6 + 0*4 + -1*4 = 11$$

Stride

Stride of 2



3	7	9	1	0	4
8	1	3	3	3	0
¹ 7	⁰ 4	⁻¹ 6	4	4	6
¹ 4	⁰ 4	⁻¹ 1	9	3	6
¹ 9	⁰ 8	⁻¹ 4	9	4	3
7	5	2	3	3	1

*

1	0	-1
1	0	-1
1	0	-1

=

0	11
9	

$$1*7 + 0*4 + -1*6 + 1*4 + 0*4 + -1*1 + 1*9 + 0*8 + -1*4 = 9$$

Stride

Stride of 2



3	7	9	1	0	4
8	1	3	3	3	0
7	4	¹ 6	⁰ 4	⁻¹ 4	6
4	4	¹ 1	⁰ 9	⁻¹ 3	6
9	8	¹ 4	⁰ 9	⁻¹ 4	3
7	5	2	3	3	1

*

1	0	-1
1	0	-1
1	0	-1

=

0	11
9	0

$$1*6 + 0*4 + -1*4 + 1*1 + 0*9 + -1*3 + 1*4 + 0*9 + -1*4 = 0$$

Padding

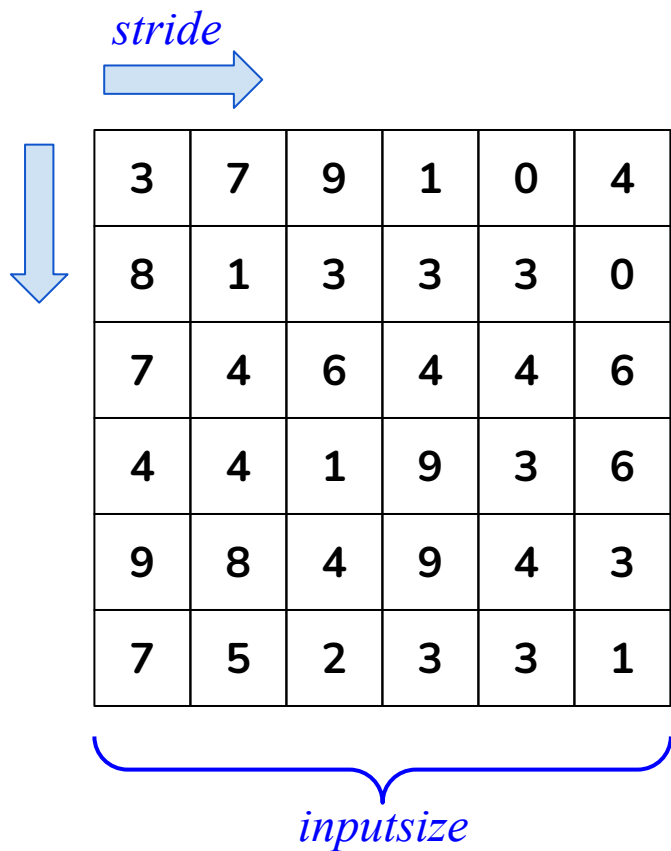
0	0	0	0	0	0	0	0
0	3	7	9	1	0	4	0
0	8	1	3	3	3	0	0
0	7	4	6	4	4	6	0
0	4	4	1	9	3	6	0
0	9	8	4	9	4	3	0
0	7	5	2	3	3	1	0
0	0	0	0	0	0	0	0

Same convolution means padding is used and output is same shape as input

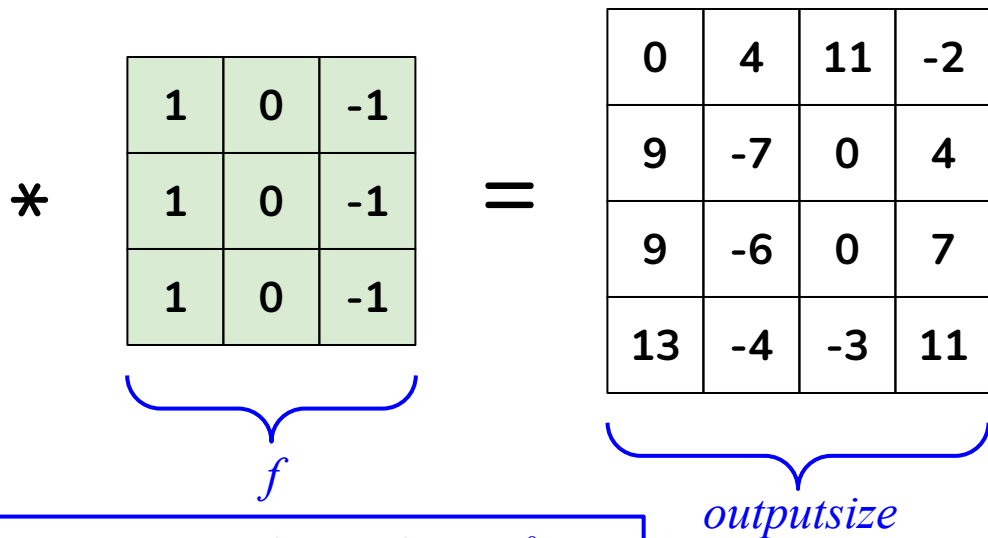
$$\begin{matrix} * & \begin{matrix} \begin{matrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{matrix} \end{matrix} & = \end{matrix}$$

-8	-1	4	9	0	3
-12	0	4	11	-2	7
-9	9	-7	0	4	10
-16	9	-6	0	7	11
-17	13	-4	-3	11	10
-13	10	1	-1	8	7

Convolution



Valid convolution means no padding is used and output is smaller shape than input



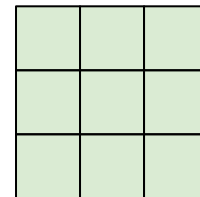
$$\text{outputsize} = \frac{\text{inputsize} - f}{\text{stride}} + 1$$

Hyperparameters

Filter (kernel) size

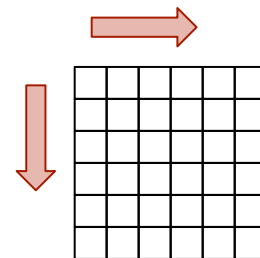
Commonly used value

3 x 3



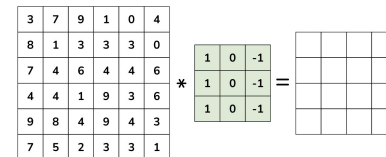
Stride

1



Valid or
same (padding?)

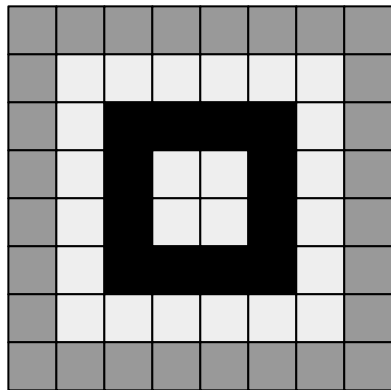
Valid
(padding=0)



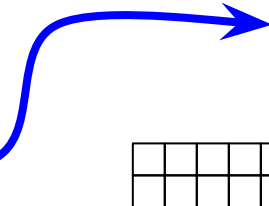
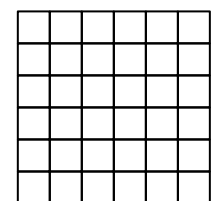
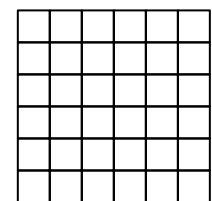
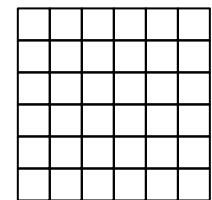
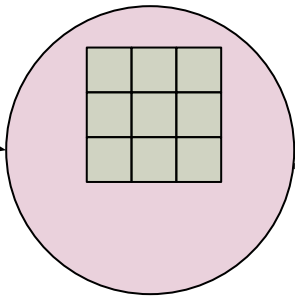
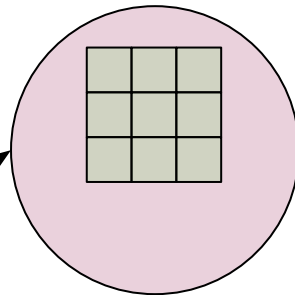
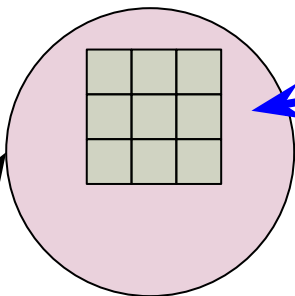
CNN

Convolutional Layer

9 parameters
for W (or f)
per unit

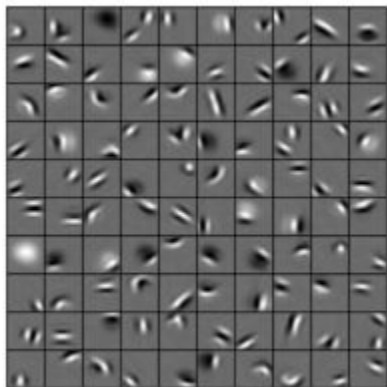


0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
0.5	1	1	1	1	1	1	1	0.5	0.5
0.5	1	0	0	0	0	0	1	0.5	0.5
0.5	1	0	1	1	0	1	0.5	0.5	0.5
0.5	1	0	1	1	0	1	0.5	0.5	0.5
0.5	1	0	0	0	0	1	0.5	0.5	0.5
0.5	1	1	1	1	1	1	0.5	0.5	0.5
0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5

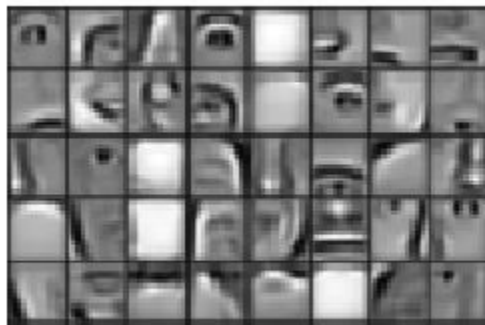


Features at Different Layers

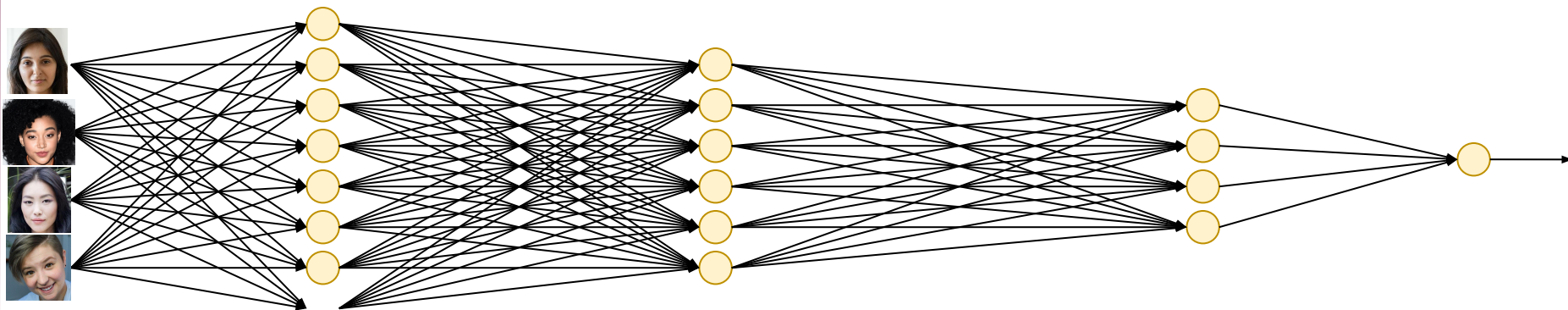
Low-Level



Mid-Level

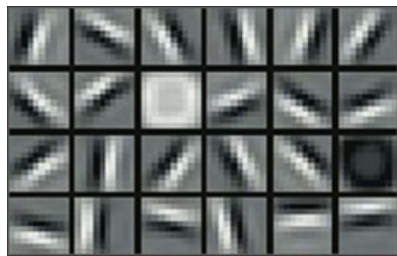


High-Level



Features at Different Layers

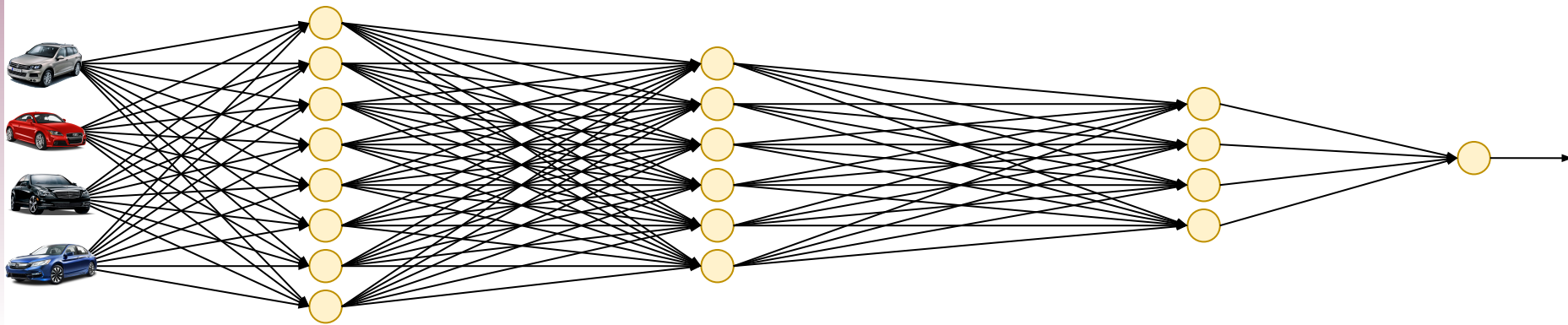
Low-Level



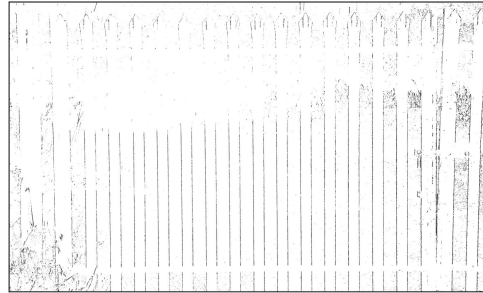
Mid-Level



High-Level



Edge Detection



Vertical Edge Detection

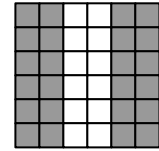
20	20	20	20	0	0	0	0
20	20	20	20	0	0	0	0
20	20	20	20	0	0	0	0
20	20	20	20	0	0	0	0
20	20	20	20	0	0	0	0
20	20	20	20	0	0	0	0
20	20	20	20	0	0	0	0
20	20	20	20	0	0	0	0

*

1	0	-1
1	0	-1
1	0	-1

=

0	0	60	60	0	0
0	0	60	60	0	0
0	0	60	60	0	0
0	0	60	60	0	0
0	0	60	60	0	0
0	0	60	60	0	0
0	0	60	60	0	0
0	0	60	60	0	0



Horizontal Edge Detection

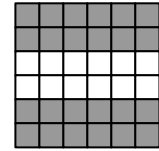
20	20	20	20	20	20	20	20
20	20	20	20	20	20	20	20
20	20	20	20	20	20	20	20
20	20	20	20	20	20	20	20
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

*

1	1	1
0	0	0
-1	-1	-1

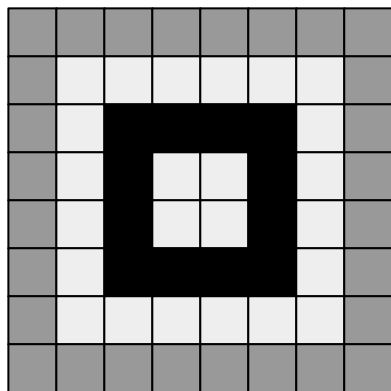
=

0	0	0	0	0	0
0	0	0	0	0	0
60	60	60	60	60	60
60	60	60	60	60	60
0	0	0	0	0	0
0	0	0	0	0	0

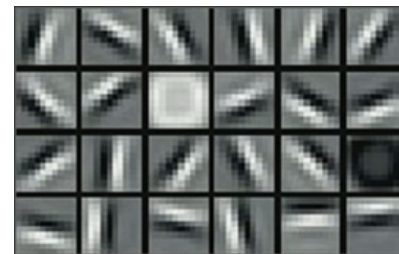
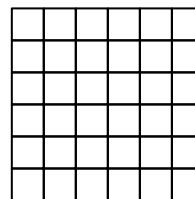
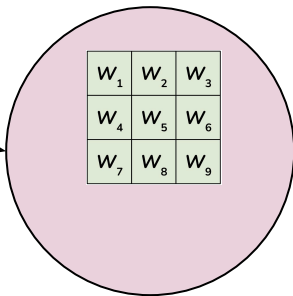
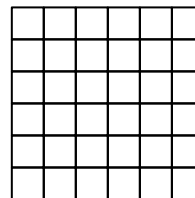
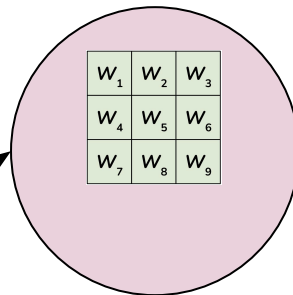
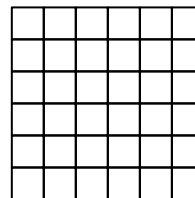
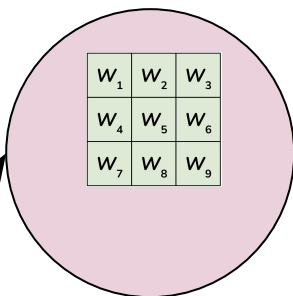


CNN

Convolutional Layer

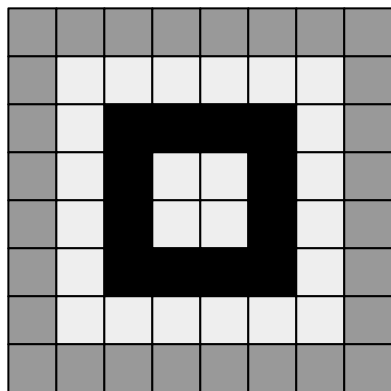


0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
0.5	1	1	1	1	1	1	1	0.5	0.5
0.5	1	0	0	0	0	0	1	0.5	0.5
0.5	1	0	1	1	0	1	0.5	0.5	0.5
0.5	1	0	1	1	0	1	0.5	0.5	0.5
0.5	1	0	0	0	0	1	0.5	0.5	0.5
0.5	1	1	1	1	1	1	0.5	0.5	0.5
0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5

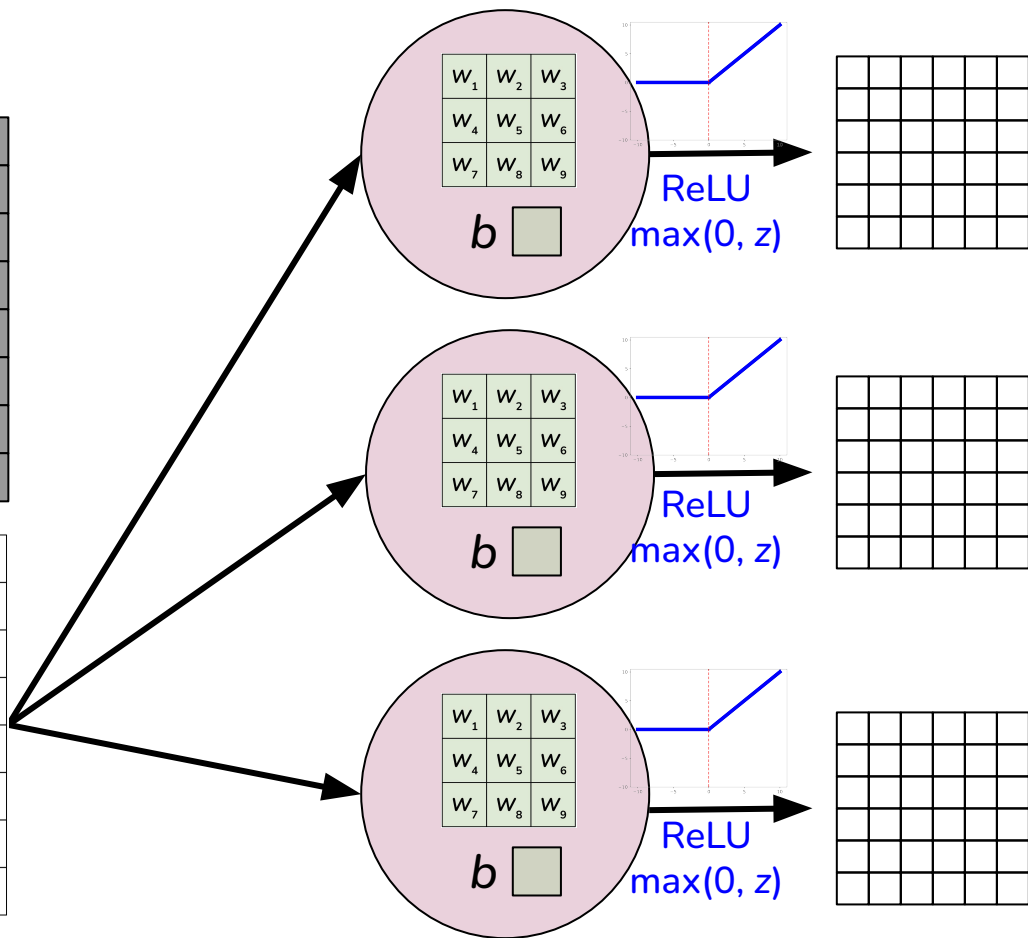


CNN

Convolutional Layer

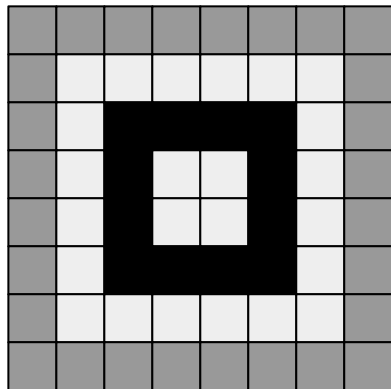


0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
0.5	1	1	1	1	1	1	1	0.5	0.5
0.5	1	0	0	0	0	0	1	0.5	0.5
0.5	1	0	1	1	0	1	0.5	0.5	0.5
0.5	1	0	1	1	0	1	0.5	0.5	0.5
0.5	1	0	0	0	0	1	0.5	0.5	0.5
0.5	1	1	1	1	1	1	0.5	0.5	0.5
0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5

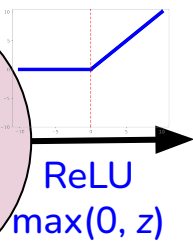
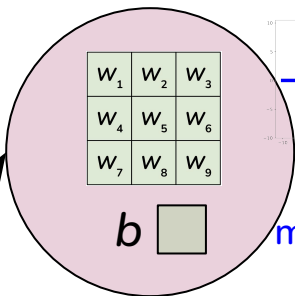


CNN

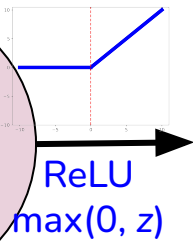
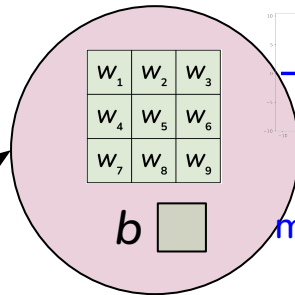
Convolutional Layer

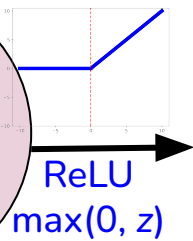
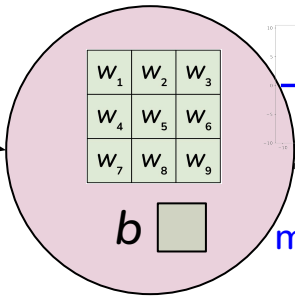


0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
0.5	1	1	1	1	1	1	1	0.5
0.5	1	0	0	0	0	1	0.5	
0.5	1	0	1	1	0	1	0.5	
0.5	1	0	1	1	0	1	0.5	
0.5	1	0	0	0	0	1	0.5	
0.5	1	1	1	1	1	1	0.5	
0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5



0.5								



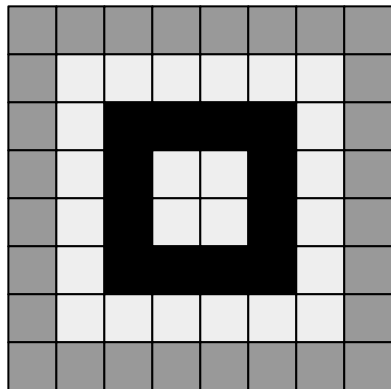


$$z = 0.5 * w_1 + 0.5 * w_2 + 0.5 * w_3 + 0.5 * w_4 + 1 * w_5 + 1 * w_6 + 0.5 * w_7 + 1 * w_8 + 0 * w_9 + b$$

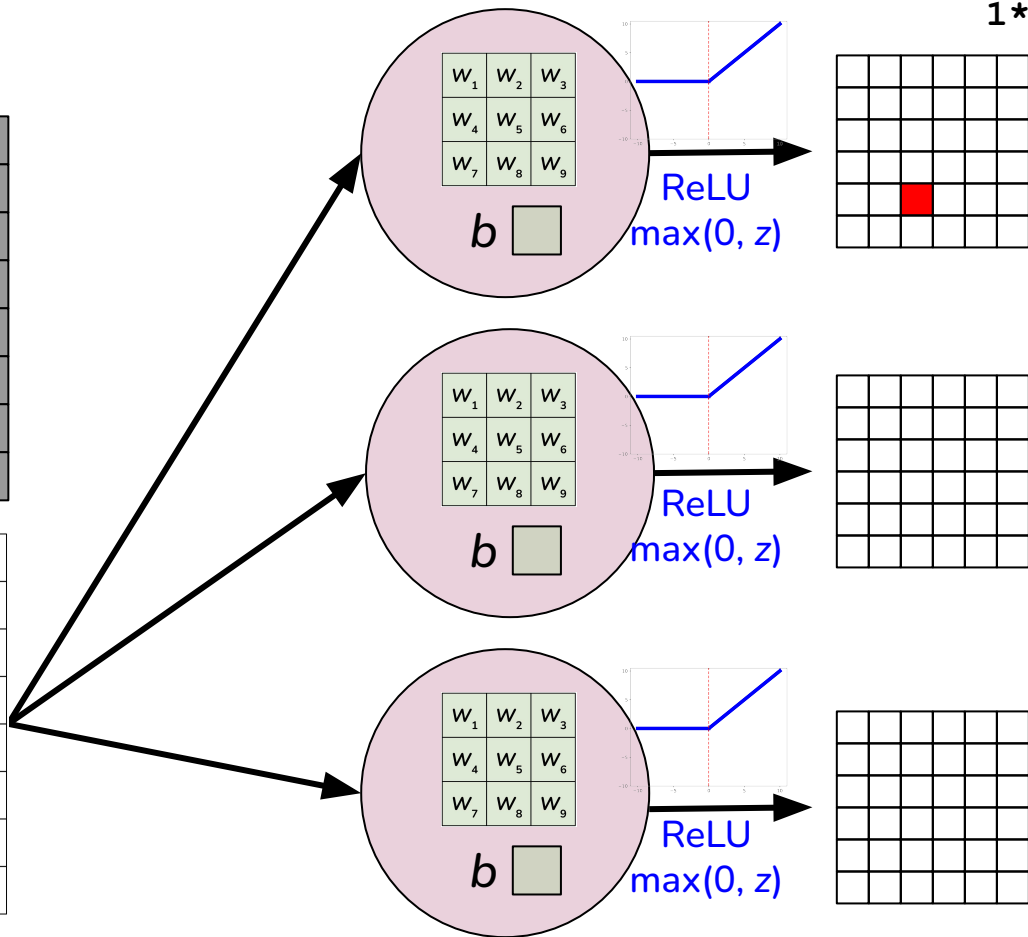
$$a = \max(0, z)$$

CNN

Convolutional Layer



0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
0.5	1	1	1	1	1	1	1	0.5	0.5
0.5	1	0	0	0	0	0	1	0.5	0.5
0.5	1	0	1	1	0	1	0.5	0.5	0.5
0.5	1	0	1	1	0	1	0.5	0.5	0.5
0.5	1	0	0	0	0	1	0.5	0.5	0.5
0.5	1	1	1	1	1	1	0.5	0.5	0.5
0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5

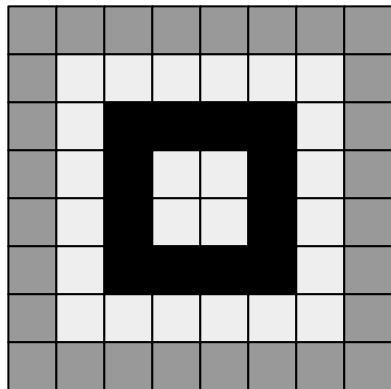


$$z = \begin{matrix} 0*w_1 & + & 1*w_2 & + & 1*w_3 \\ 0*w_4 & + & 0*w_5 & + & 0*w_6 \\ 1*w_7 & + & 1*w_8 & + & 1*w_9 \end{matrix} + b$$

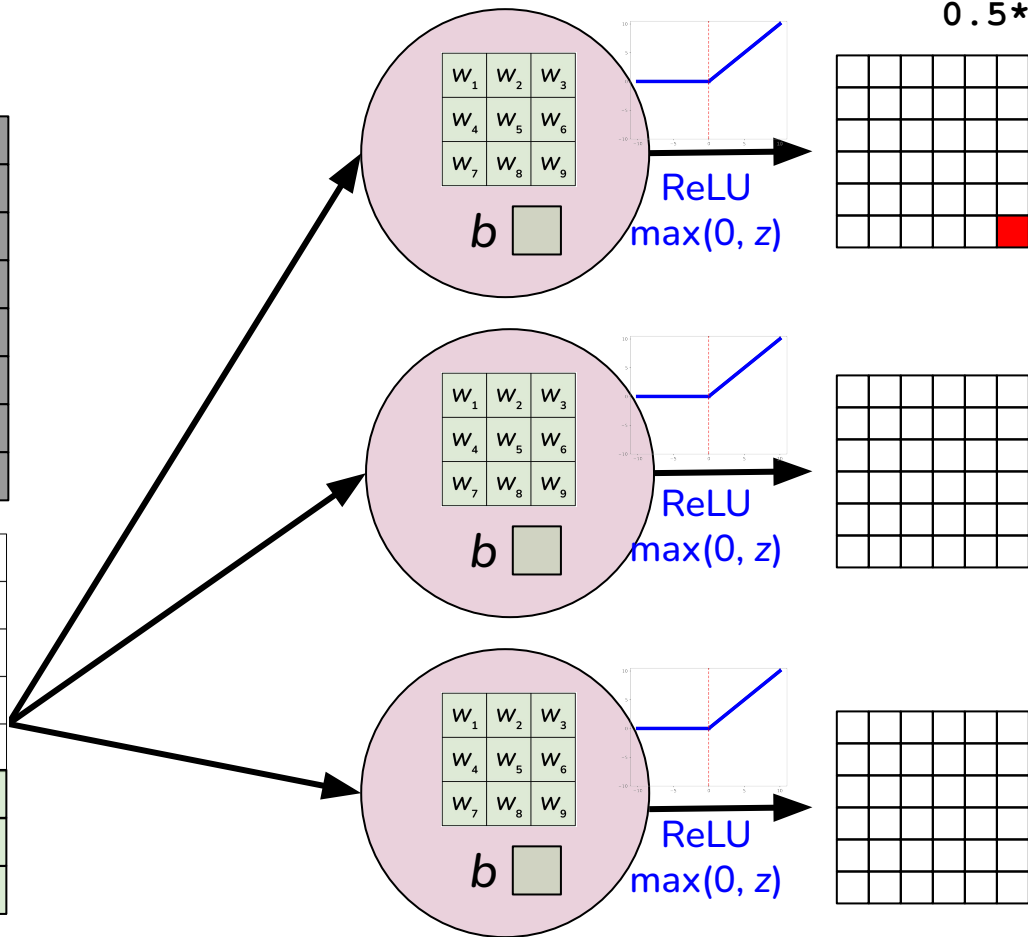
$$a = \max(0, z)$$

CNN

Convolutional Layer



0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
0.5	1	1	1	1	1	1	1	0.5	0.5
0.5	1	0	0	0	0	0	1	0.5	0.5
0.5	1	0	1	1	0	1	0.5	0.5	0.5
0.5	1	0	1	1	0	1	0.5	0.5	0.5
0.5	1	0	0	0	0	1	1	0.5	0.5
0.5	1	1	1	1	1	1	1	0.5	0.5
0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5

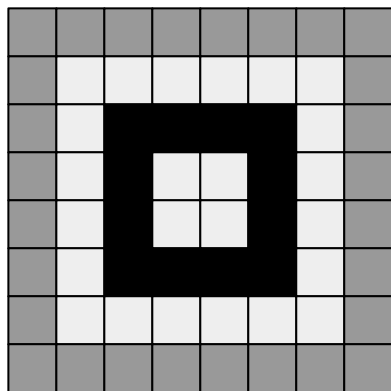


$$z = \begin{matrix} 0*w_1 & + & 1*w_2 & + & 0.5*w_3 \\ 1*w_4 & + & 1*w_5 & + & 0.5*w_6 \\ 0.5*w_7 & + & 0.5*w_8 & + & 0.5*w_9 \end{matrix} + b$$

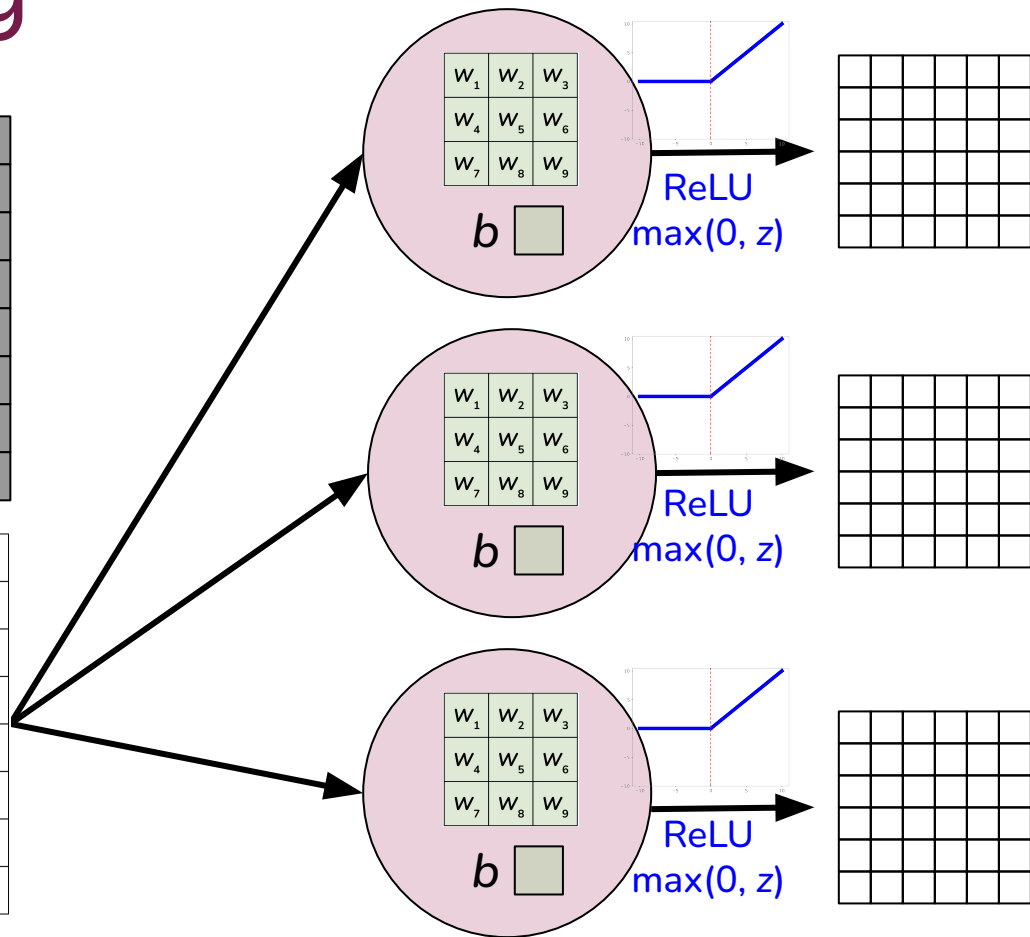
$$a = \max(0, z)$$

Training

Convolutional Layer



0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
0.5	1	1	1	1	1	1	1	0.5	0.5
0.5	1	0	0	0	0	0	1	0.5	0.5
0.5	1	0	1	1	0	1	0.5	0.5	0.5
0.5	1	0	1	1	0	1	0.5	0.5	0.5
0.5	1	0	0	0	0	1	0.5	0.5	0.5
0.5	1	1	1	1	1	1	0.5	0.5	0.5
0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5



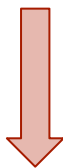
- Choose number of units (filters) in layer
- Initialize 10 parameters in each unit
- Perform gradient descent to update parameters

Neural Network Layers

- ❖ Fully connected (dense)
- ❖ Convolutional
- ❖ Max pooling

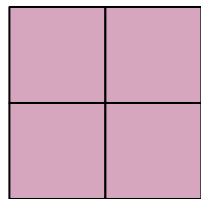
Max Pooling

Stride of 2



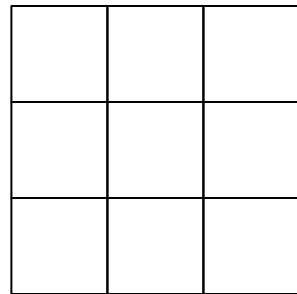
3	7	9	1	0	4
8	1	3	3	3	0
7	4	6	4	4	6
4	4	1	9	3	6
9	8	4	9	4	3
7	5	2	3	3	1

max



2 x 2

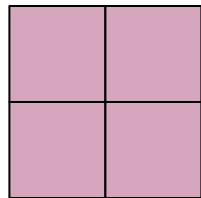
=



Max Pooling

3	7	9	1	0	4
8	1	3	3	3	0
7	4	6	4	4	6
4	4	1	9	3	6
9	8	4	9	4	3
7	5	2	3	3	1

max



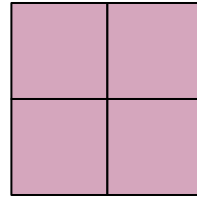
=

8		

Max Pooling

3	7	9	1	0	4
8	1	3	3	3	0
7	4	6	4	4	6
4	4	1	9	3	6
9	8	4	9	4	3
7	5	2	3	3	1

max



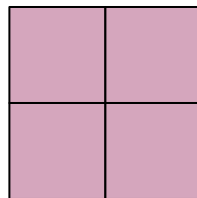
=

8	9	

Max Pooling

3	7	9	1	0	4
8	1	3	3	3	0
7	4	6	4	4	6
4	4	1	9	3	6
9	8	4	9	4	3
7	5	2	3	3	1

max



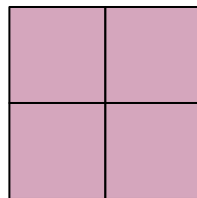
=

8	9	4

Max Pooling

3	7	9	1	0	4
8	1	3	3	3	0
7	4	6	4	4	6
4	4	1	9	3	6
9	8	4	9	4	3
7	5	2	3	3	1

max



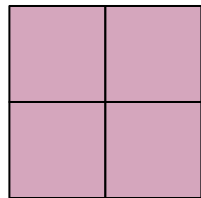
=

8	9	4
7		

Max Pooling

3	7	9	1	0	4
8	1	3	3	3	0
7	4	6	4	4	6
4	4	1	9	3	6
9	8	4	9	4	3
7	5	2	3	3	1

max



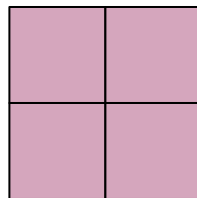
=

8	9	4
7	9	

Max Pooling

3	7	9	1	0	4
8	1	3	3	3	0
7	4	6	4	4	6
4	4	1	9	3	6
9	8	4	9	4	3
7	5	2	3	3	1

max



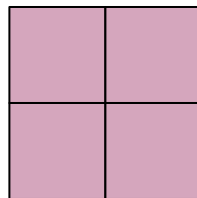
=

8	9	4
7	9	6

Max Pooling

3	7	9	1	0	4
8	1	3	3	3	0
7	4	6	4	4	6
4	4	1	9	3	6
9	8	4	9	4	3
7	5	2	3	3	1

max



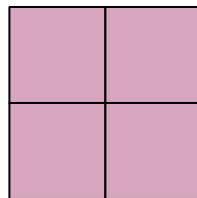
=

8	9	4
7	9	6
9		

Max Pooling

3	7	9	1	0	4
8	1	3	3	3	0
7	4	6	4	4	6
4	4	1	9	3	6
9	8	4	9	4	3
7	5	2	3	3	1

max



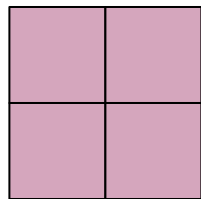
=

8	9	4
7	9	6
9	9	

Max Pooling

3	7	9	1	0	4
8	1	3	3	3	0
7	4	6	4	4	6
4	4	1	9	3	6
9	8	4	9	4	3
7	5	2	3	3	1

max

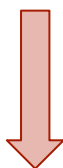


=

8	9	4
7	9	6
9	9	4

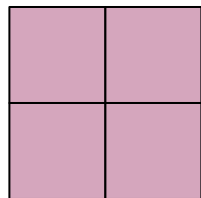
Max Pooling

Stride of 2



3	7	9	1	0	4
8	1	3	3	3	0
7	4	6	4	4	6
4	4	1	9	3	6
9	8	4	9	4	3
7	5	2	3	3	1

max



2 x 2

=

8	9	4
7	9	6
9	9	4

No parameters
to learn!

Output has half the height
and half the width of input

Layers

Conv

(1198, 1918)

MP

(599, 959)

(1200, 1920)

