

# Convolutional Neural Networks

---

# Computer Vision Problems

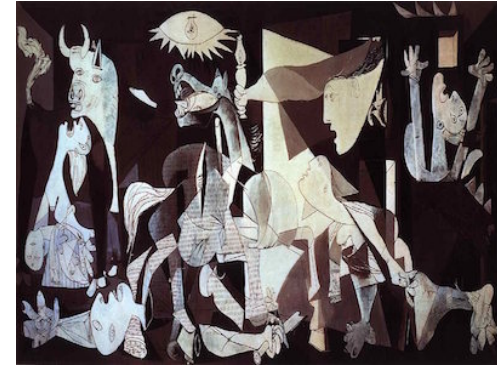
## Image Classification



64x64

→ Cat? (0/1)

## Neural Style Transfer



## Object detection



# Deep Learning on large images



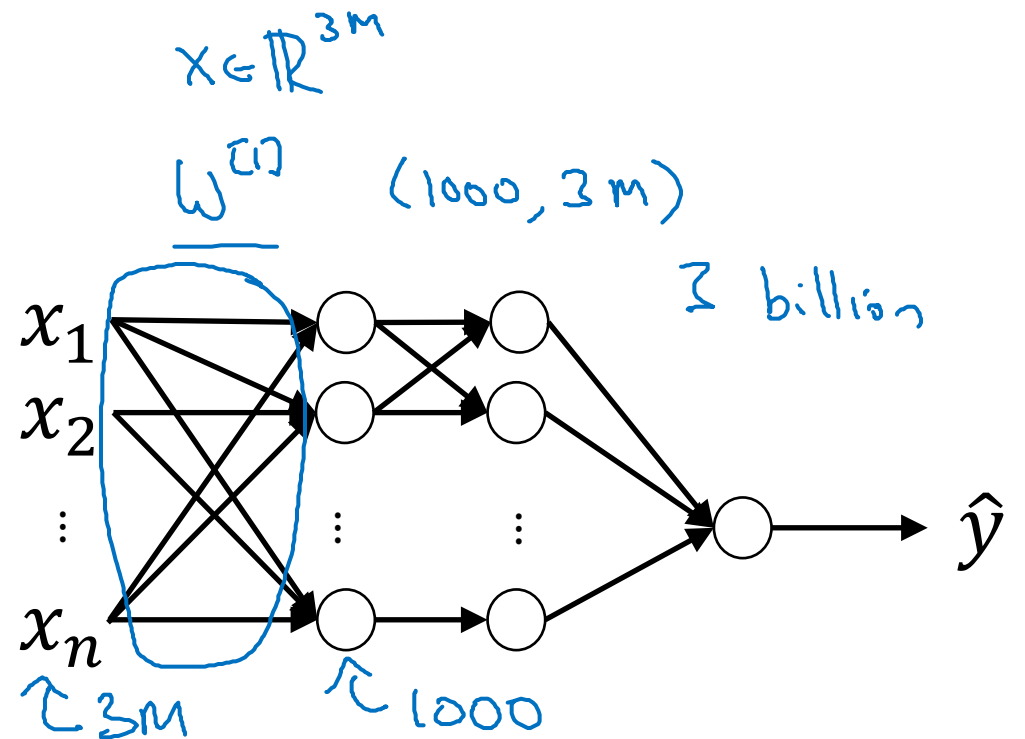
64x64 x 3

→ Cat? (0/1)

12288

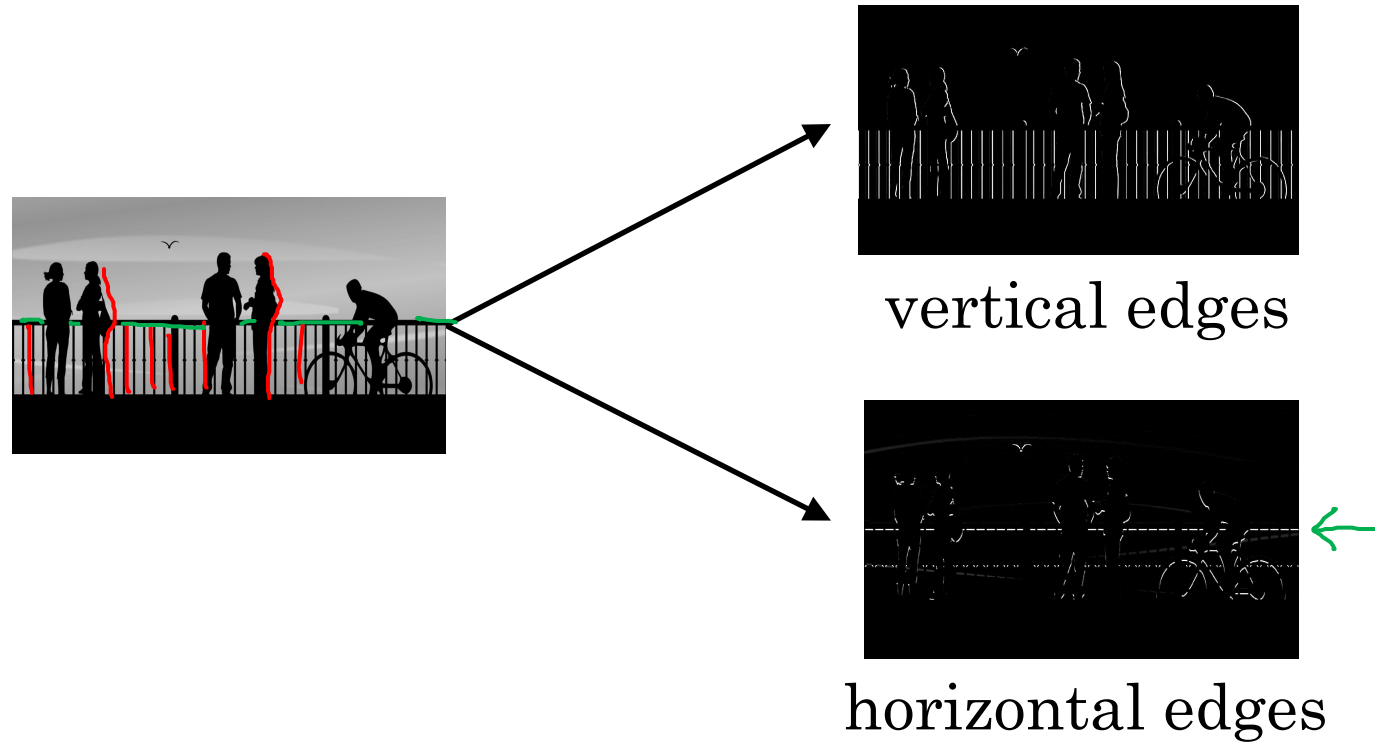


1000x1000 x 3  
= 3 million



# Edge Detection example

# Computer Vision Problem



# Vertical edge detection

$$\rightarrow 3 \times 1 + 1 \times 1 + 2 \times 1 + 0 \times 0 + 5 \times 0 + 7 \times 0 + 1 \times -1 + 8 \times -1 + 2 \times -1 = -5$$

3 <sup>1</sup>	0 <sup>0</sup>	1 <sup>-1</sup>	2	7	4
1 <sup>1</sup>	5 <sup>0</sup>	8 <sup>-1</sup>	9	3	1
2 <sup>1</sup>	7 <sup>0</sup>	2 <sup>-1</sup>	5	1	3
0	1	3	1	7	8
4	2	1	6	2	8
2	4	5	2	3	9

6x6

"convolution"

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

3x3

→ filter kernel

\*

=

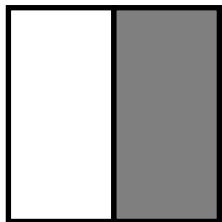
-5	-4	0	8
-10	-2	2	3
0	-2	-4	-7
-3	-2	-3	-16

4x4

# Vertical edge detection

10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0

6x6



\*

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

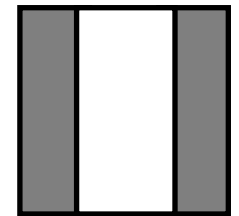
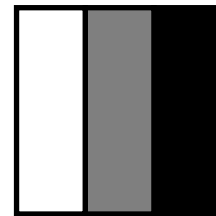
3x3

=

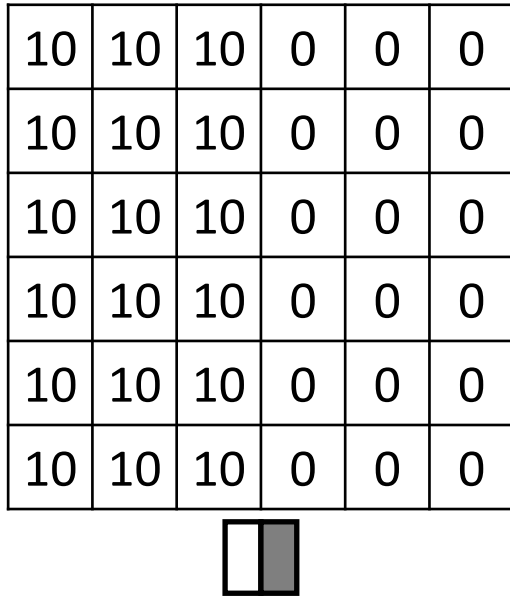
0	30	30	0
0	30	30	0
0	30	30	0
0	30	30	0

4x4

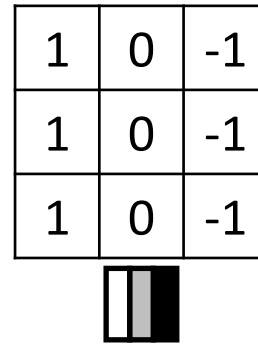
\*



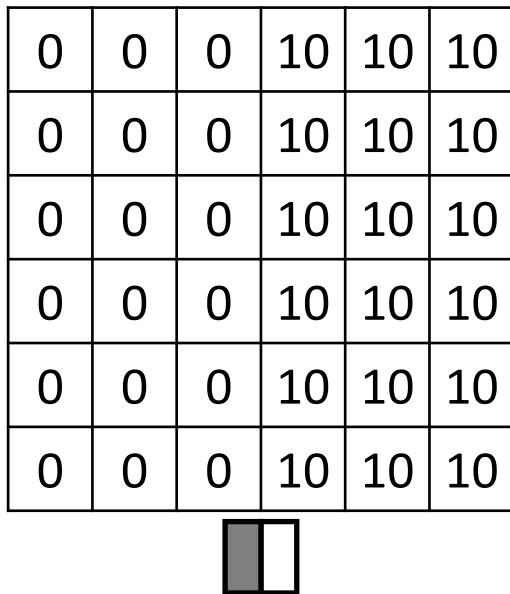
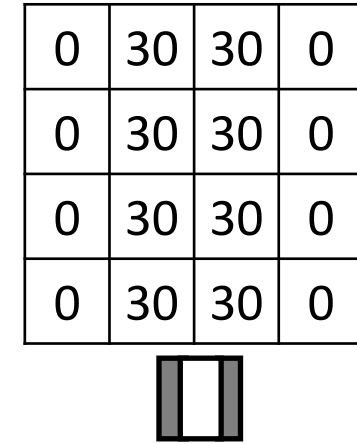
# Vertical edge detection examples



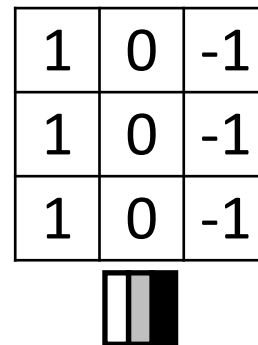
\*



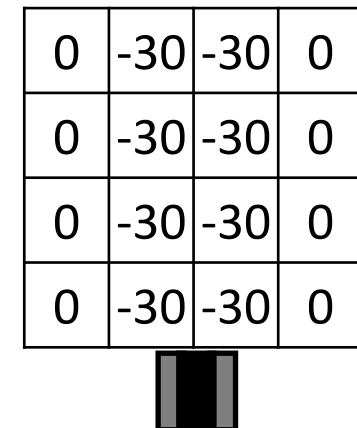
=



\*



=





# Vertical and Horizontal Edge Detection

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

Vertical

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

Horizontal

10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
0	0	0	10	10	10
0	0	0	10	10	10
0	0	0	10	10	10

6x6

\*

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

=

0	0	0	0
30	10	-10	-30
30	10	-10	-30
0	0	0	0



# Learning to detect edges

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



1	0	-1
2	0	-2
1	0	-1

Sobel filter

3	0	-3
10	0	-10
3	0	-3

Scharr filter

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

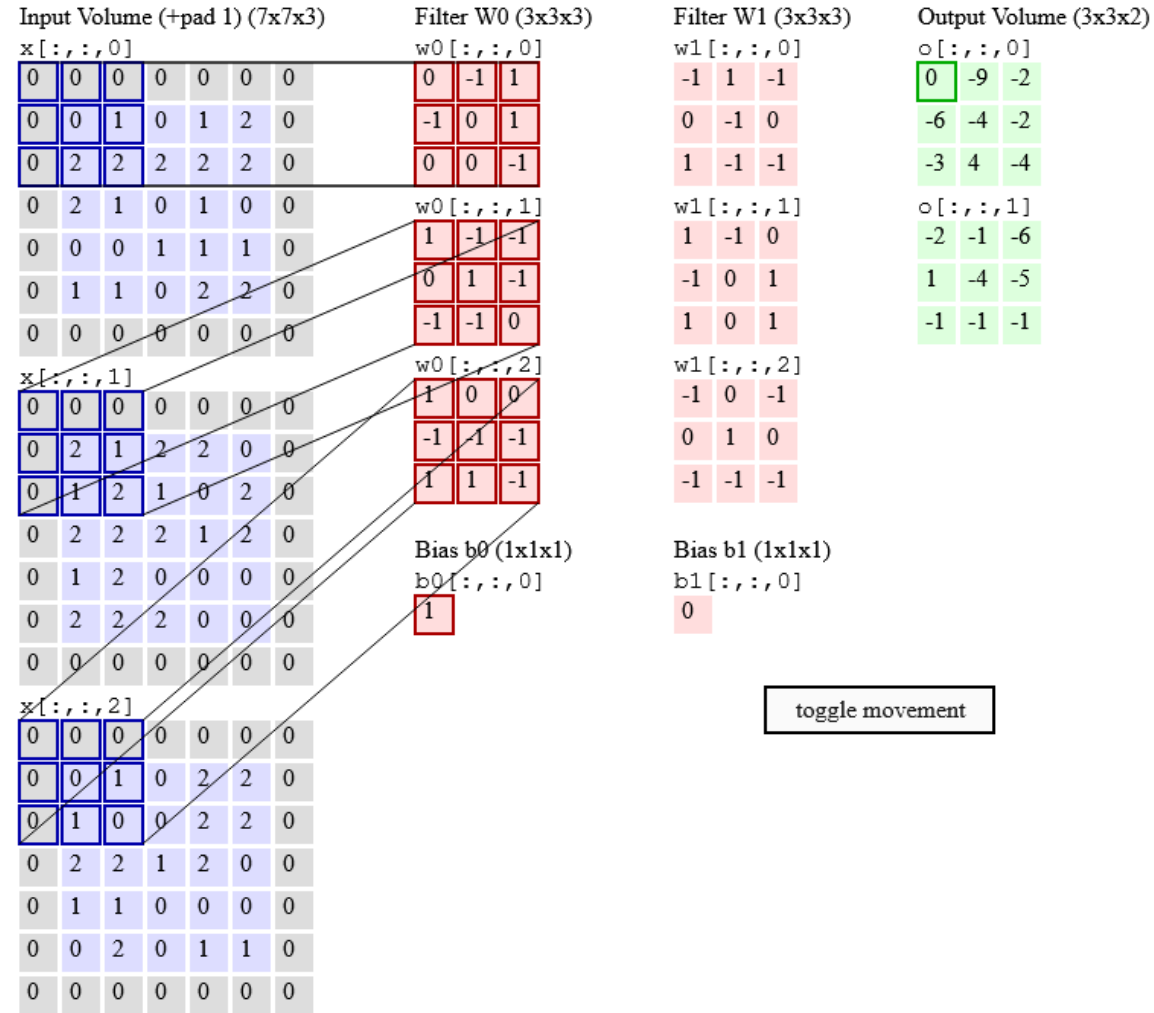
convolution  
\*

$W_1$	$W_2$	$W_3$
$W_4$	$W_5$	$W_6$
$W_7$	$W_8$	$W_9$

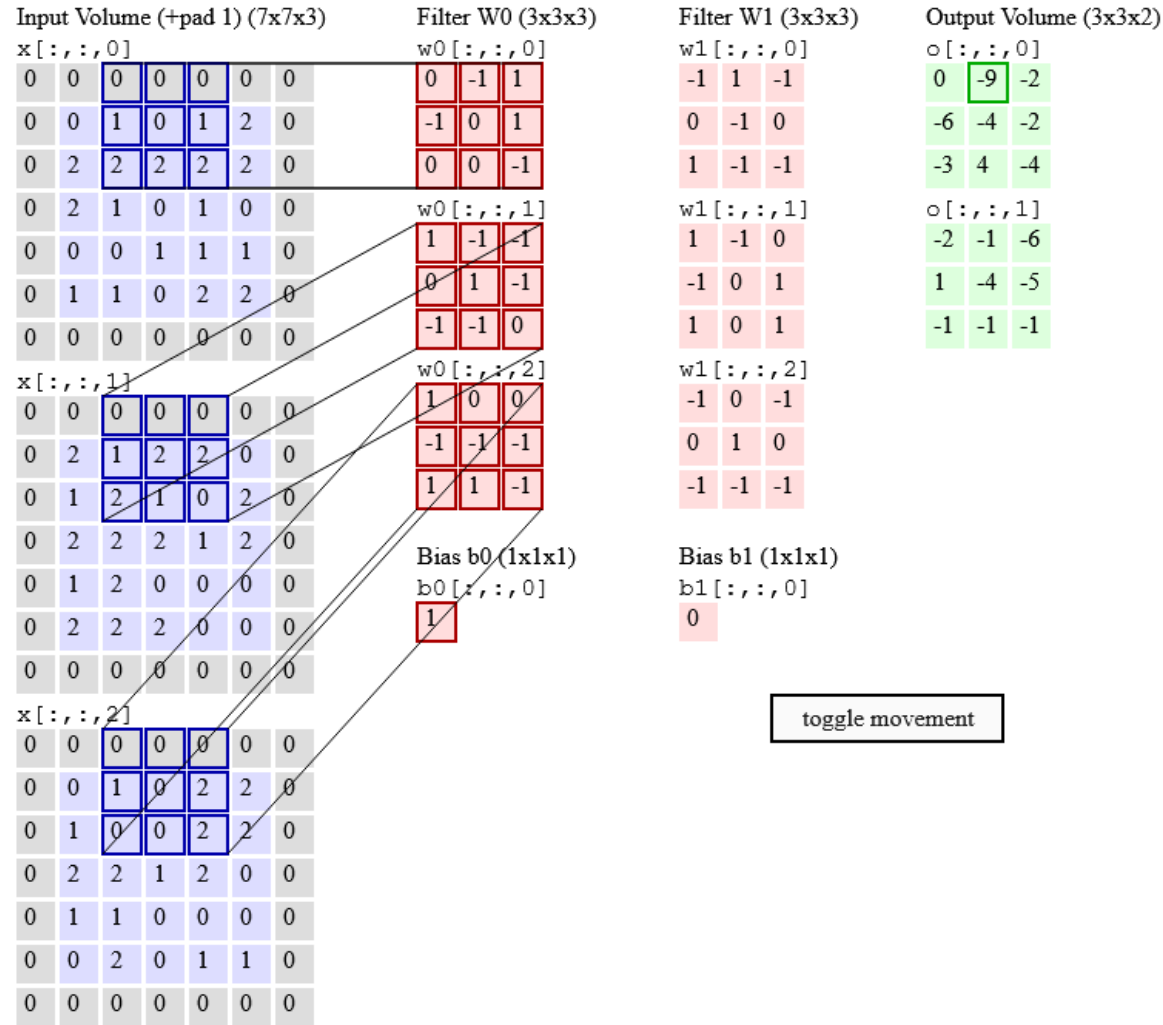
3x3

=  
45°  
70°  
73°


# Visualization



# Visualization



# Visualization

