

Computational Intelligence

Samaneh Hosseini

Isfahan University of Technology

Outline

- Convolutional Neural Networks
 - Computer Vision
 - What Computers “See”
 - Edge Detection Example
 - More Edge Detection

Convolutional Neural Networks: Computer Vision



**“To know what is
where by looking.”**

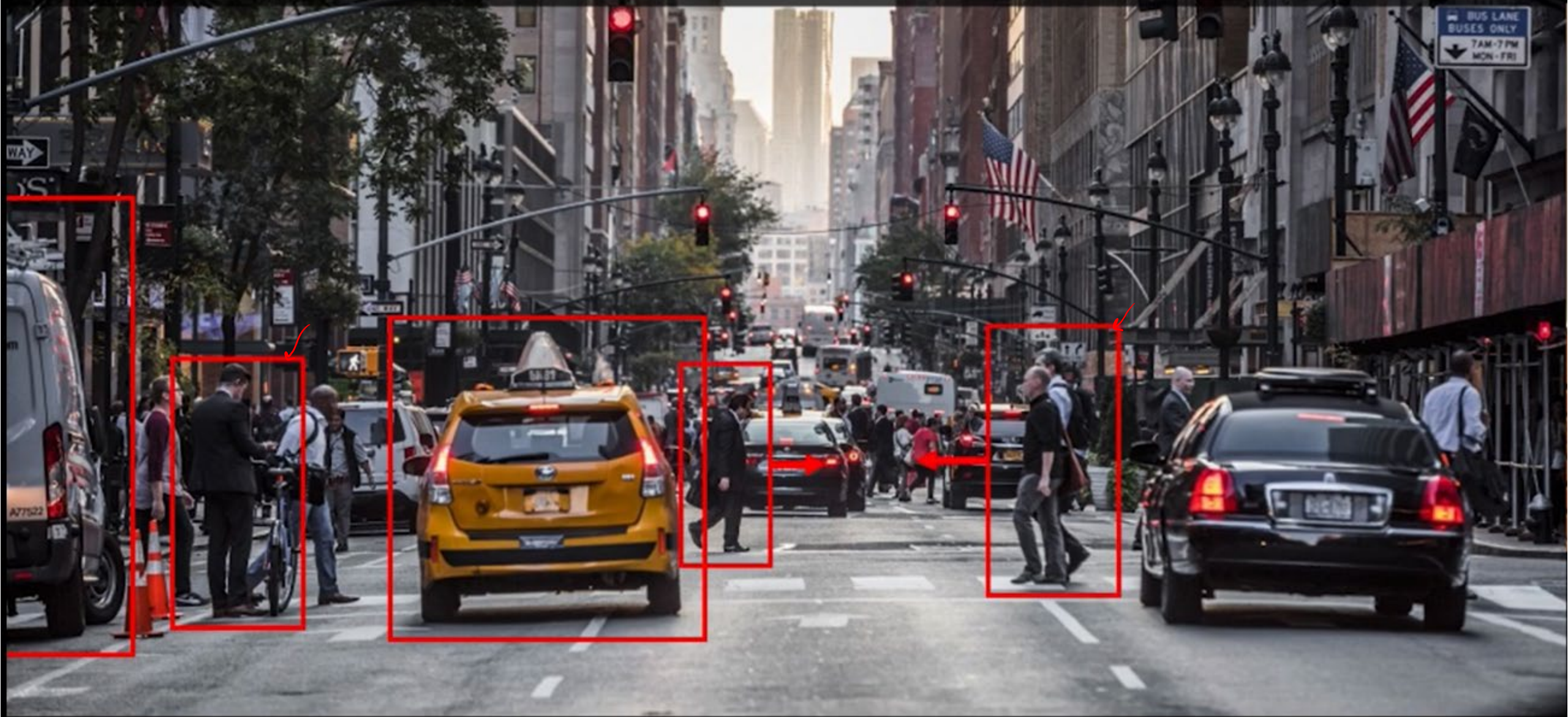
To discover from images what is present in the world, where things are, what actions are taking place, to predict and anticipate events in the world



To discover from images what is present in the world, where things are, what actions are taking place, to predict and anticipate events in the world

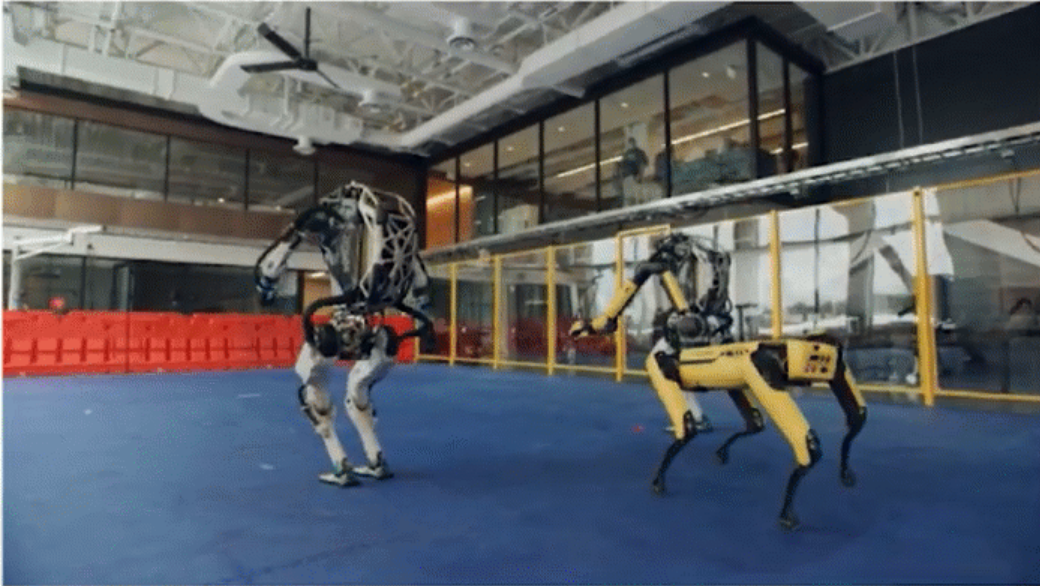


To discover from images what is present in the world, where things are, what actions are taking place, to predict and anticipate events in the world

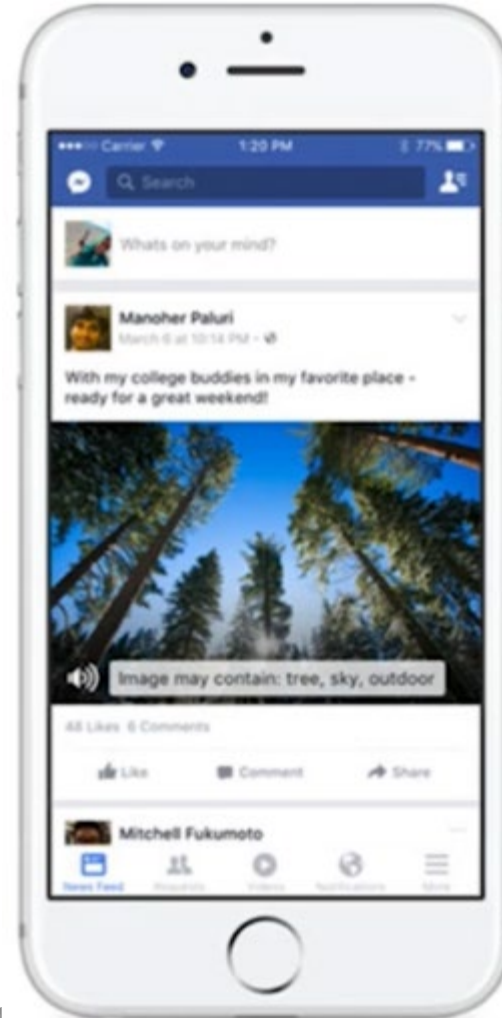


The rise and impact of computer vision

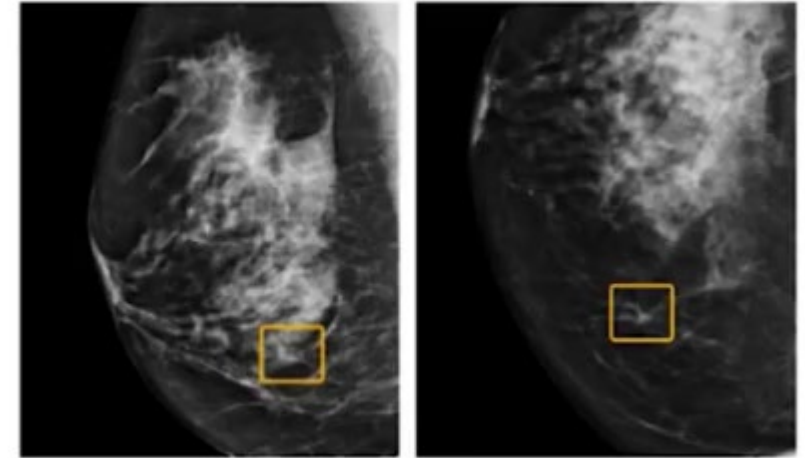
Robotics



Accessibility



Biology & Medicine



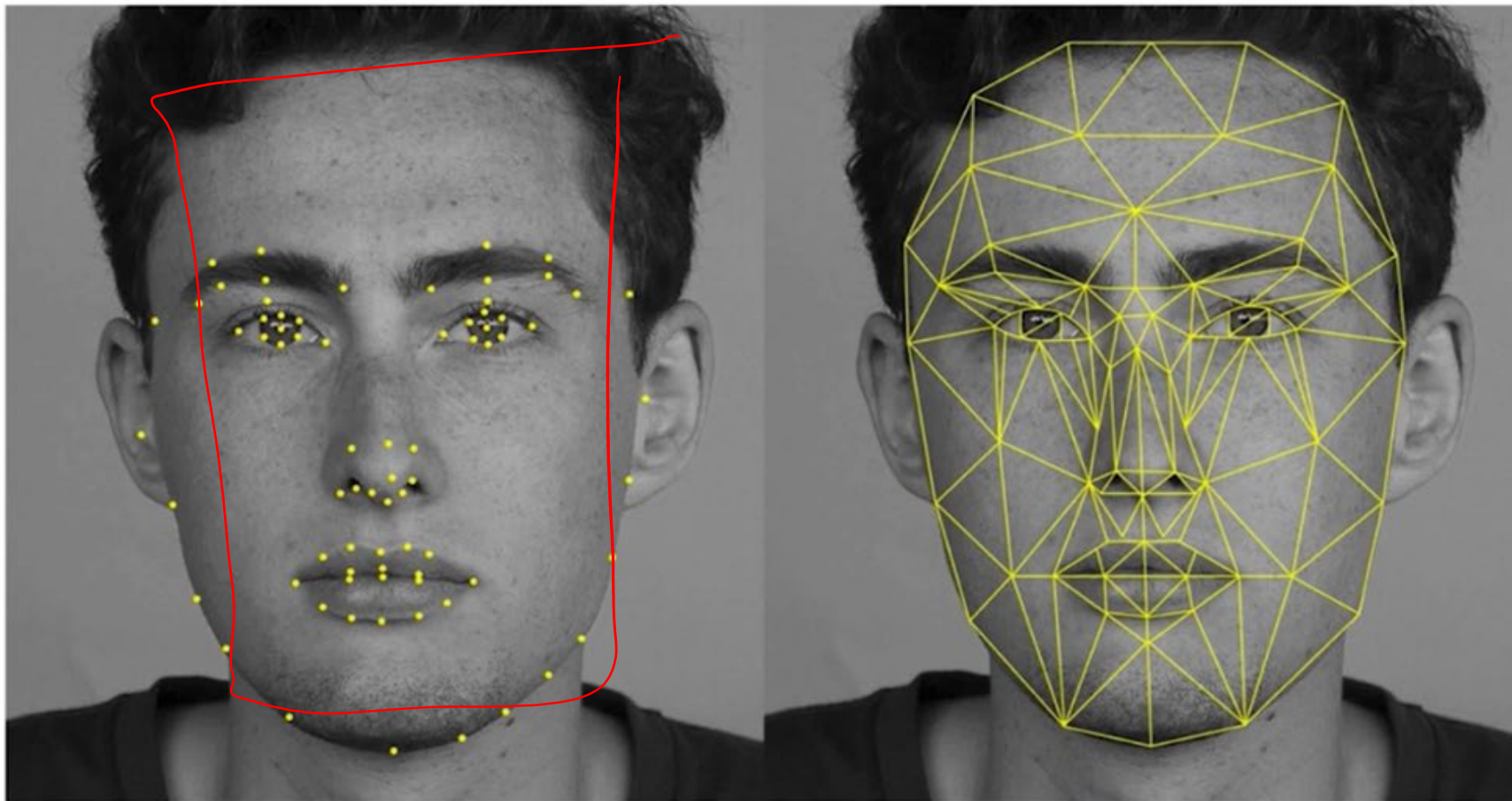
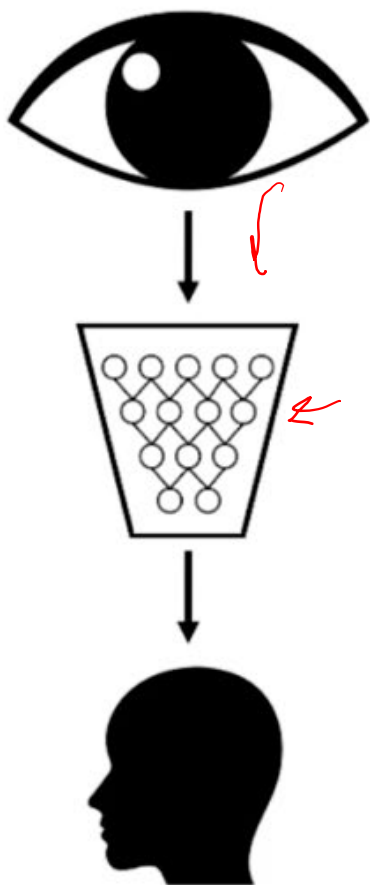
Autonomous driving



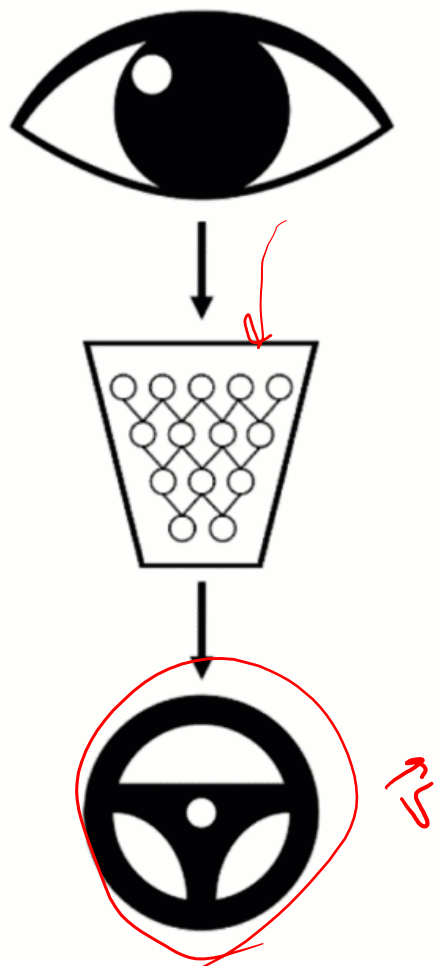
Mobile computing



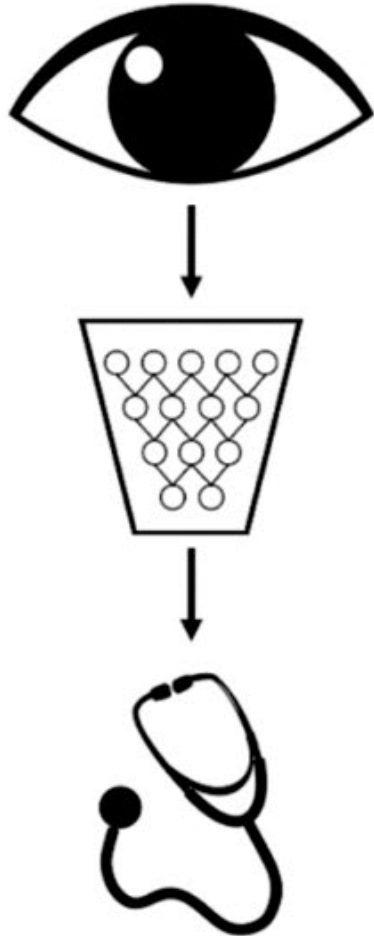
Impact: Facial Detection and Recognition



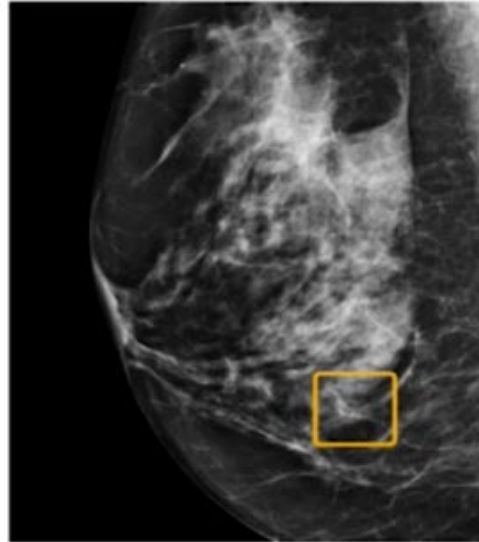
Impact: Self-Driving Cars



Impact: Medicine, Biology, Healthcare



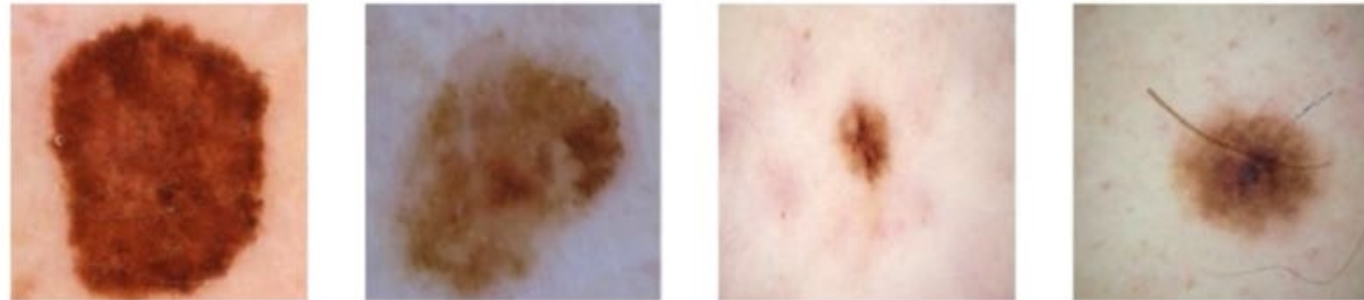
Breast cancer



COVID-19



Skin cancer



Computer vision problems

Image Classification



64x64

→ Cat? (0/1)

Neural Style Transfer



Object detection



Convolutional Neural Networks: What Computers “See”

Images are numbers



157	153	174	168	150	152	129	151	172	161	155	156
155	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	34	6	10	33	48	106	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	106	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	106	36	190
205	174	155	252	236	231	149	178	228	43	95	234
190	216	116	149	236	187	86	150	79	38	218	241
190	224	147	108	227	210	127	102	36	101	255	224
190	214	173	66	103	143	96	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
195	206	123	207	177	121	123	200	175	13	96	218

What the computer sees

157	153	174	168	150	152	129	151	172	161	155	156
155	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	34	6	10	33	48	106	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	106	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	106	36	190
205	174	155	252	236	231	149	178	228	43	95	234
190	216	116	149	236	187	86	150	79	38	218	241
190	224	147	108	227	210	127	102	36	101	255	224
190	214	173	66	103	143	96	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
195	206	123	207	177	121	123	200	175	13	96	218

Take in computer vision



Input Image



187	163	174	168	190	182	129	181	172	161	186	186
185	182	163	74	75	62	33	17	110	210	180	184
180	180	50	14	34	6	10	33	48	106	199	181
206	109	5	124	131	111	120	204	166	15	66	180
194	68	137	251	237	239	239	227	87	71	201	
172	106	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	106	36	190
206	174	165	252	236	231	149	178	228	43	95	234
190	216	116	149	236	187	86	150	79	38	218	241
190	224	147	108	227	210	127	102	36	101	255	224
190	214	173	66	103	143	96	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
195	206	123	207	177	121	123	200	175	13	96	218

Pixel Representation



classification

Lincoln

Washington

Jefferson

Obama

0.8
0.1
0.05
0.05

- **Regression:** output variable takes continuous value
- **Classification:** output variable takes class label. Can produce probability of belonging to a particular class

Deep Learning on Large Images



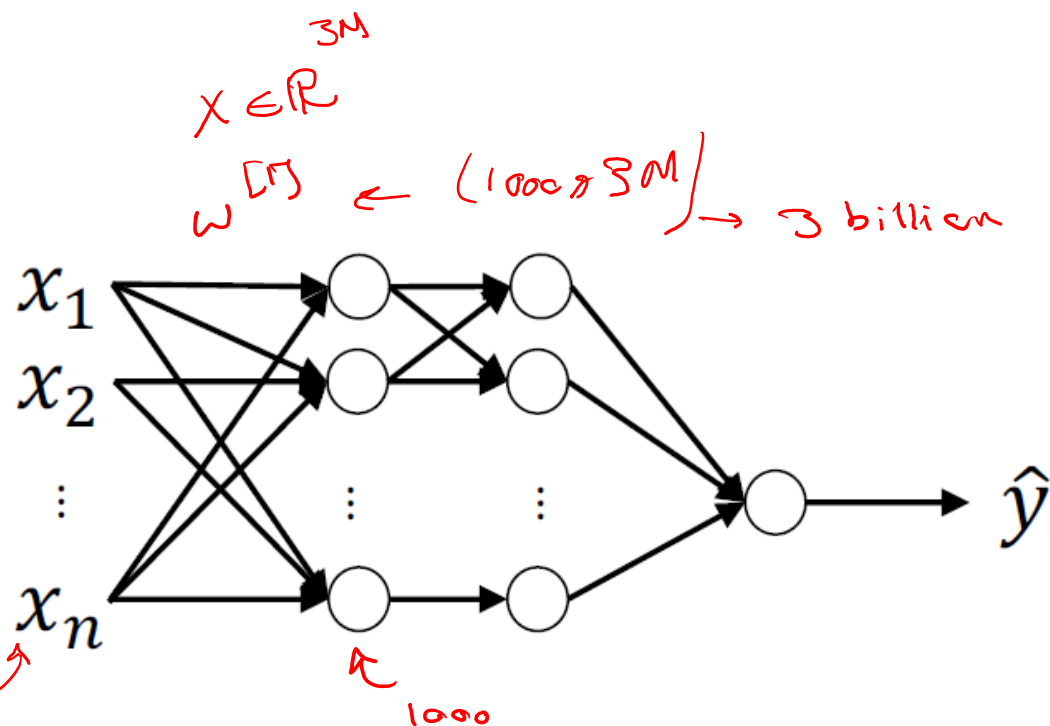
64x64x3

→ Cat? (0/1)

12288

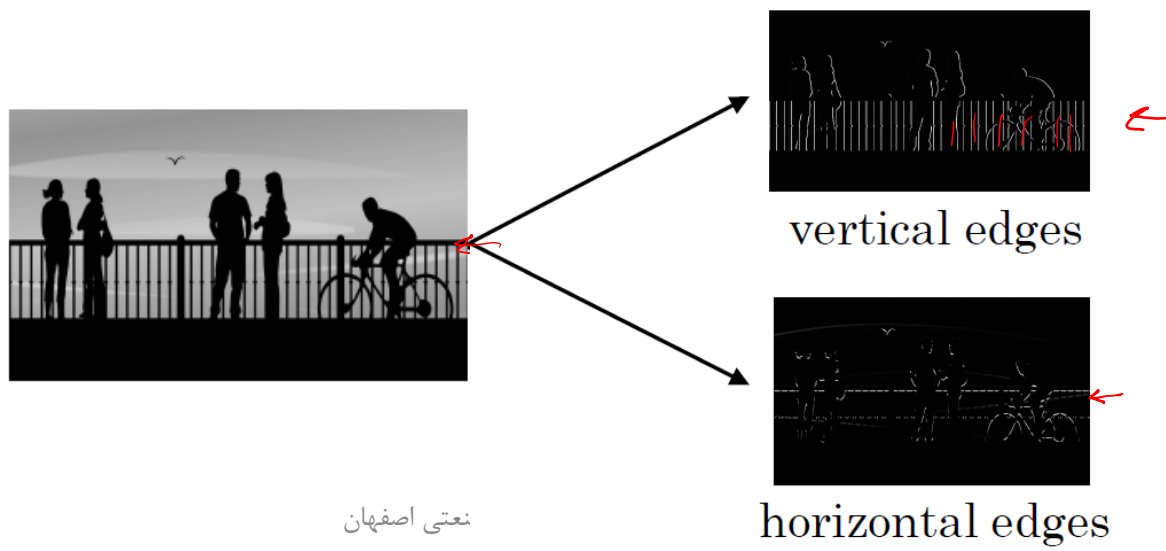


$1000 \times 1000 \times 3$
= 3 million



Convolutional Neural Networks: Edge detection example

Edge detection example



Vertical edge detection

$$3 \times 1 + 1 \times 1 + 2 \times 1 + \underbrace{0 \times 0 + 5 \times 0 + 7 \times 0}_0 + 1 \times (-1) + 8 \times (-1) + 2 \times (-1) = -5$$

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

6x6

*

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

3x3
filter
kernel

=

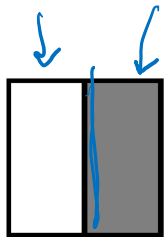
-5	-4	0	8
-10	-2	2	3
0	-2	-4	-7
-3	-2	-3	-16

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

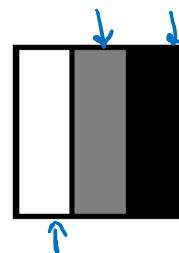
3x3

=

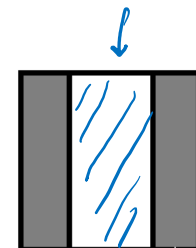
0	30	30	0
0	30	30	0
0	30	30	0
0	30	30	0

4x4

*



سمانه حسینی




نعتی اصفهان

Convolutional Neural Networks: More edge detection


Vertical edge detection examples

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

اگرش

تیر

*


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


↗

=


تیر → اگرش

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




6 و 6

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	10	10	10

تیر

اگرش

*


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



=

اگرش → تیر

0	-30	-30	0
0	-30	-30	0
0	-30	-30	0
0	-30	-30	0
0	-30	-30	0
0	-30	-30	0



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

Diagram illustrating the convolution process for edge detection. A 6x6 input image is convolved with a 3x3 kernel to produce a 4x4 output image.

Input Image (6x6):

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

Handwritten notes on the input image: "30" with a downward arrow on the left, "ردش" (red) and "نیز" (blue) with arrows pointing to the 3x3 region, and "ردش*" (red) with an arrow pointing to the 3x3 region.

Kernel (3x3):

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

Output Image (4x4):

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

Handwritten notes on the output image: "30" with a downward arrow on the left, "ردش" (red) and "نیز" (blue) with arrows pointing to the 3x3 region, and "ردش*" (red) with an arrow pointing to the 3x3 region.

A small diagram at the bottom shows a 3x3 grid with a red box and a blue box, indicating the convolution process.

High level feature detection

Let's identify key features in each image category



Nose,
Eyes,
Mouth

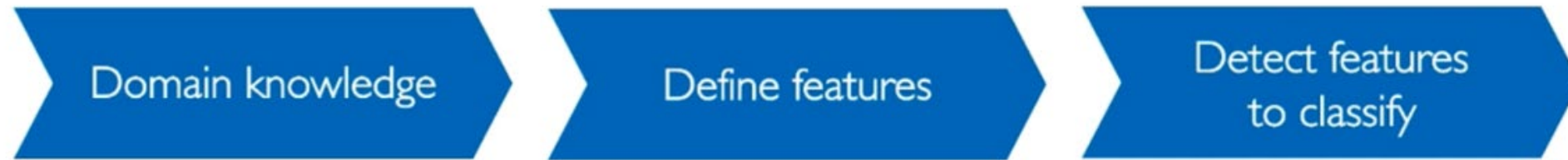


Wheels,
License Plate,
Headlights



Door,
Windows,
Steps

Manual feature extraction



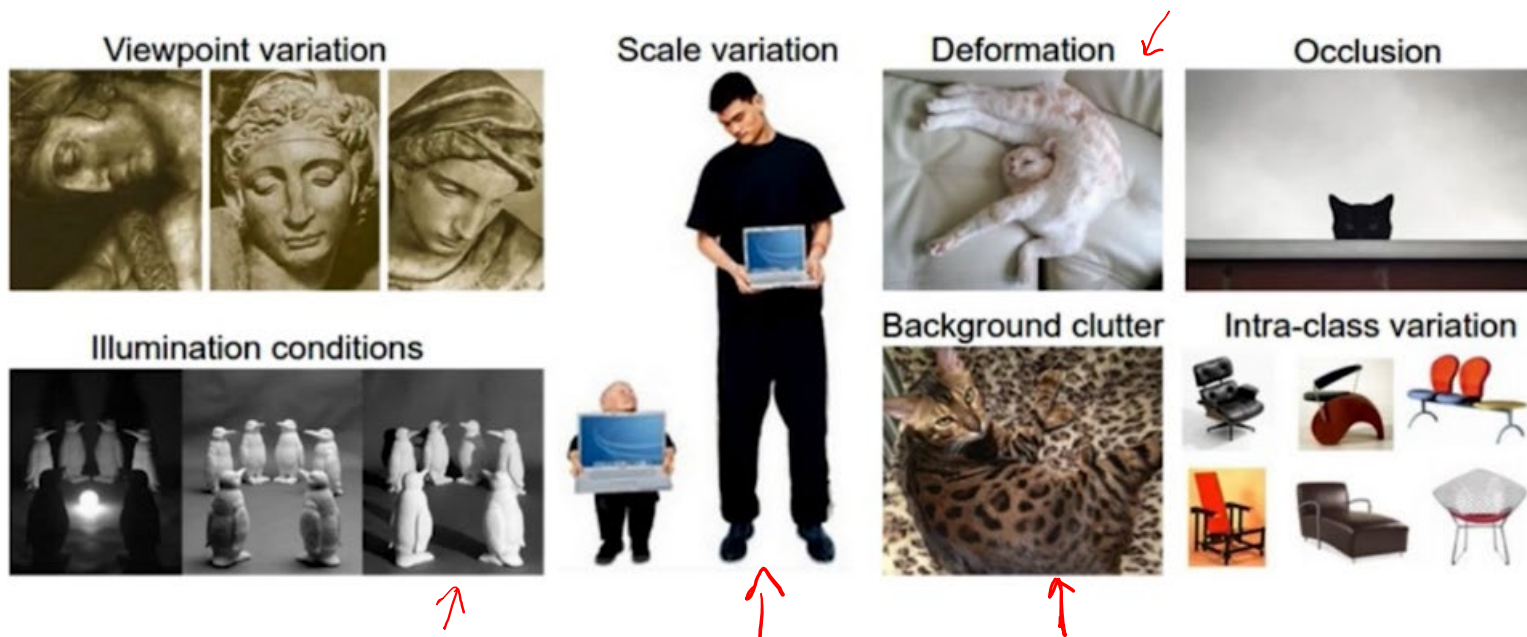
Problems?

Manual feature extraction

Domain knowledge

Define features

Detect features
to classify



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

=

4 5 0
10 0
8 0

Core Foundation Review

- Convolutional Neural Networks
 - Computer Vision
 - What Computers “See”
 - Edge Detection Example
 - More Edge Detection