CS 4104
APPLIED MACHINE LEARNING

Dr. Hashim Yasin

National University of Computer and Emerging Sciences,

Faisalabad, Pakistan.

CONVOLUTIONAL NEURAL NETWORK

CNN

- Neural Networks that use convolution in place of general matrix multiplication in at least one layer

- There are three types of layers in the convolutional network,

  - **Convolution layer (Conv)**
  - **Pooling layer (Pool)**
  - **Fully connected layer (FC)**

Cross-correlation

- Let f be the image,
- w be the kernel of size $m \times n$
  - where $m = 2a + 1$ and $n = 2b + 1$), a and b are the positive integers.
- g be the output image

$$g(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x+s, y+t)$$

This is called a **cross-correlation** operation:

$$g = w \otimes f$$

Cross-correlation

$$g(x,y)=\sum_{s=-a}^{a}\sum_{t=-b}^{b} w(s,t)\,f(x+s,y+t)$$

At any point (x, y), the response $g(x, y)$ of the filter is the sum of product of filter coefficient and the image pixels

$$\begin{aligned} g(x,y) = {} & w(-1,-1)\,f(x-1,y-1)+ \\ & w(-1,0)\,f(x-1,y)+\ldots \\ & w(0,0)\,f(x,y)+\ldots \\ & w(1,1)\,f(x+1,y+1) \end{aligned}$$

Convolution

- Same as cross-correlation, except that the kernel is "*flipped*" (horizontally and vertically)

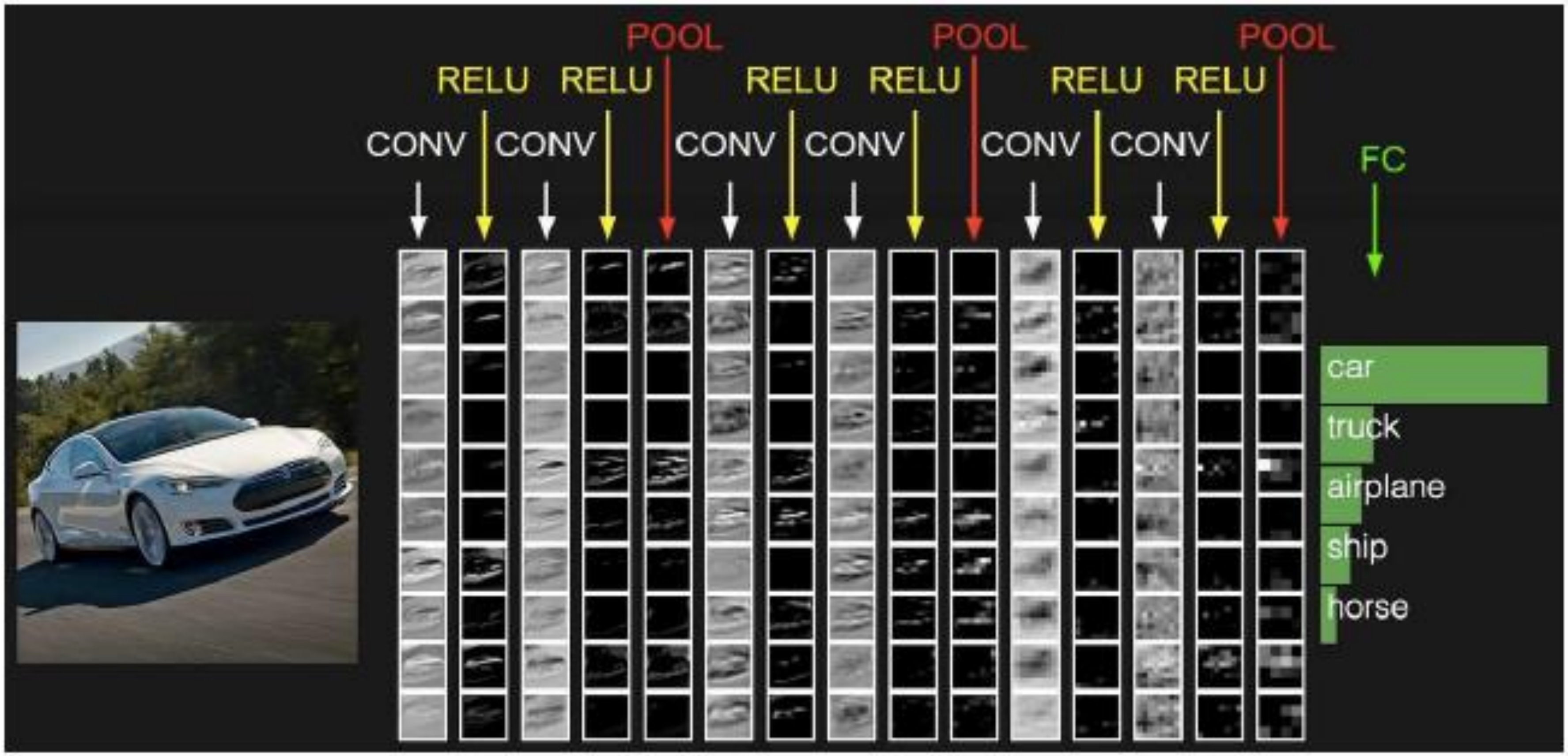
$$g(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x-s, y-t)$$

This is called a **convolution** operation:

$$g = w * f$$

- Convolution is **commutative** and **associative**

CNN … Example


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

CNN … EDGE DETECTION

CNN … Edge Detection

White to Dark pixels

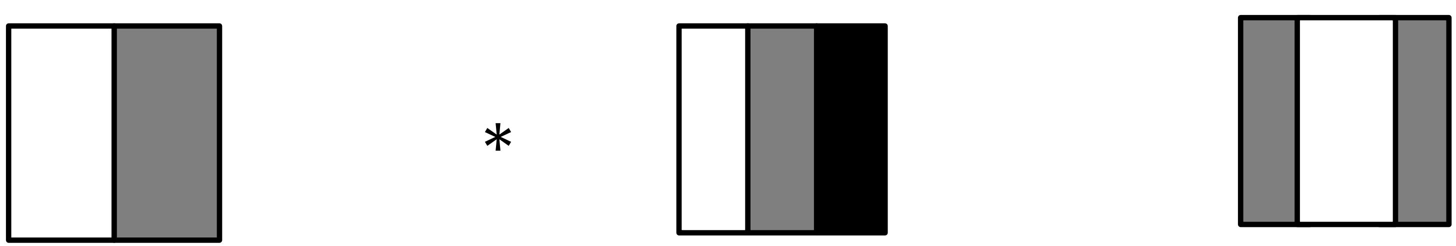
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0

*

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

=

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



CNN … Edge Detection

Dark to White pixels

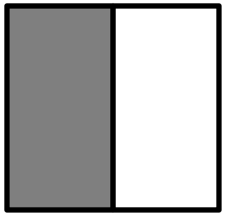
0	0	0	10	10	10
0	0	0	10	10	10
0	0	0	10	10	10
0	0	0	10	10	10
0	0	0	10	10	10
0	0	0	10	10	10

*

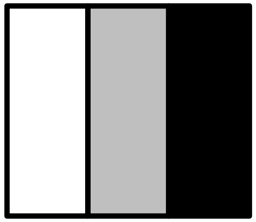
1	0	-1
1	0	-1
1	0	-1

=

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



*



CNN … Edge Detection

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

*

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

Prewitt Filter

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

Sobel Filter

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

Scharr Filter

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

CNN … Edge Detection

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

$*$

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

$=$

Horizontal Edges
Vertical Edges
45° Degree Edges
70° Degree Edges
…

CNN … Edge Detection

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

*

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

=

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

$$n - f + 1 \times n - f + 1$$

CNN … PADDING

CNN … Padding

Convolution without Padding

- Shrinking output
  - If we have a hundred layers of deep net and it'll shrink a bit on every layer, then after a hundred layers we end up with a very small image.
- Throwing away a lot of information
  - The information on the edges are thrown away every time

CNN … Padding

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

$\text{Padding} = p = 1$

$*$

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

$=$

$$n + 2p - f + 1 \times n + 2p - f + 1$$

CNN … Padding

Valid Convolution

$n \times n$ $\quad$ $f \times f$ $\quad$ $n - f + 1 \times n - f + 1$

Same Convolution

Pad so that output size is the same size as the input size.

$$n + 2p - f + 1 \times n + 2p - f + 1$$

$$n + 2p - f + 1 = n$$
$$p = \frac{f - 1}{2}$$

CNN … STRIDE

CNN … Stride

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

$*$

Padding: $p = 0$
Stride: $s = 2$

3	4	4
1	0	2
-1	0	3

$=$

91	100	83
69	91	127
44	72	74

$$\frac{n + 2p - f}{s} + 1$$

CNN … Stride

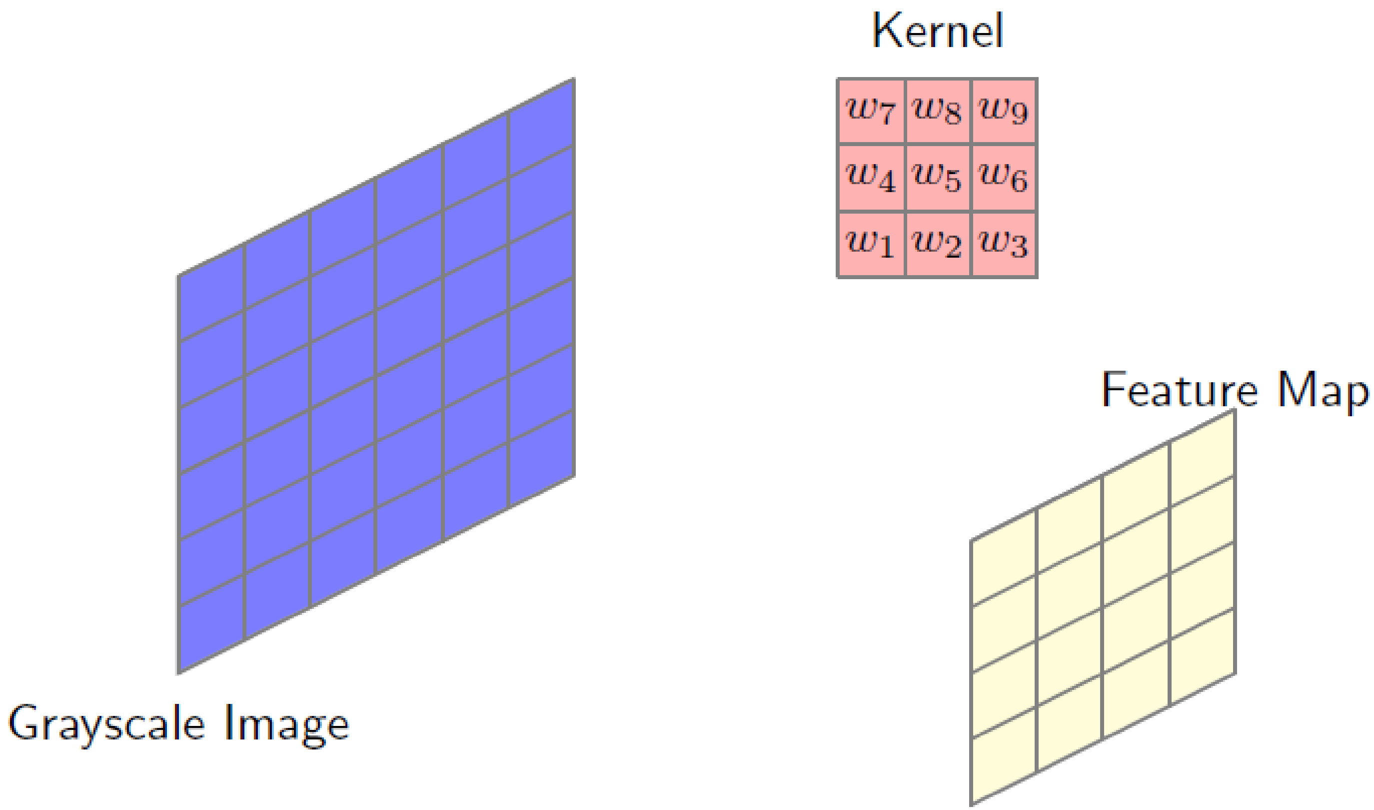
$n \times n,$ $f \times f$

padding $p,$ stride s

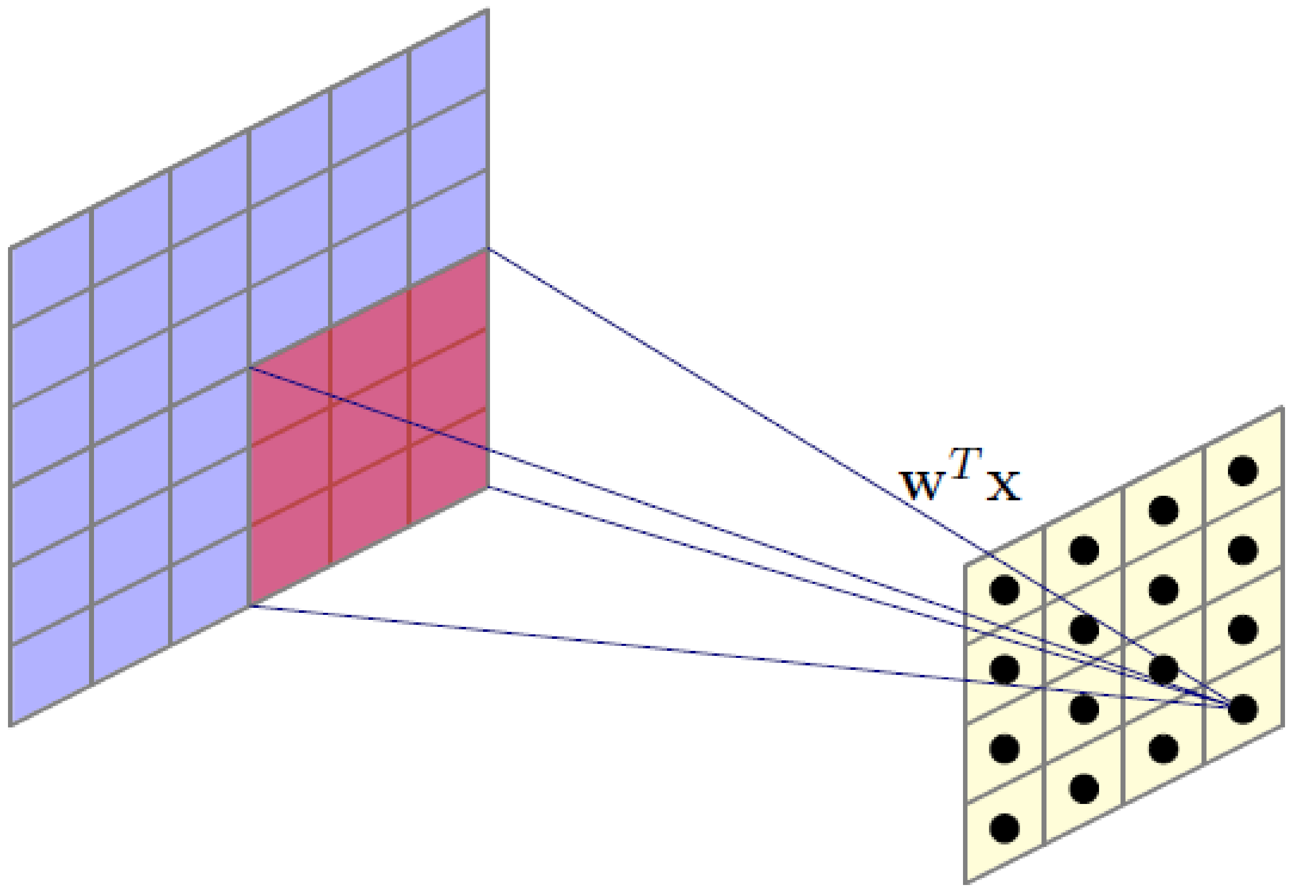
$$\left\lfloor \frac{n+2p-f}{s} + 1 \right\rfloor \times \left\lfloor \frac{n+2p-f}{s} + 1 \right\rfloor$$

CNN … Example

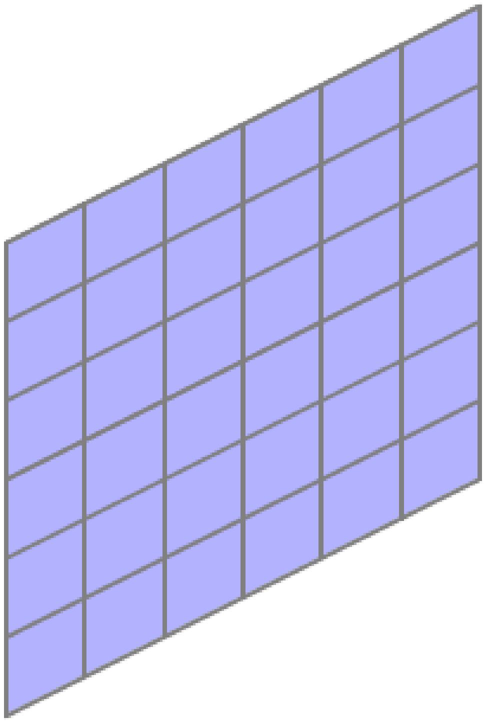
- Convolve image with kernel having weights w

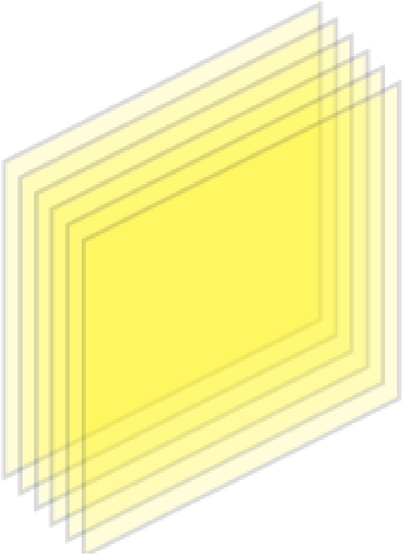


CNN … Example



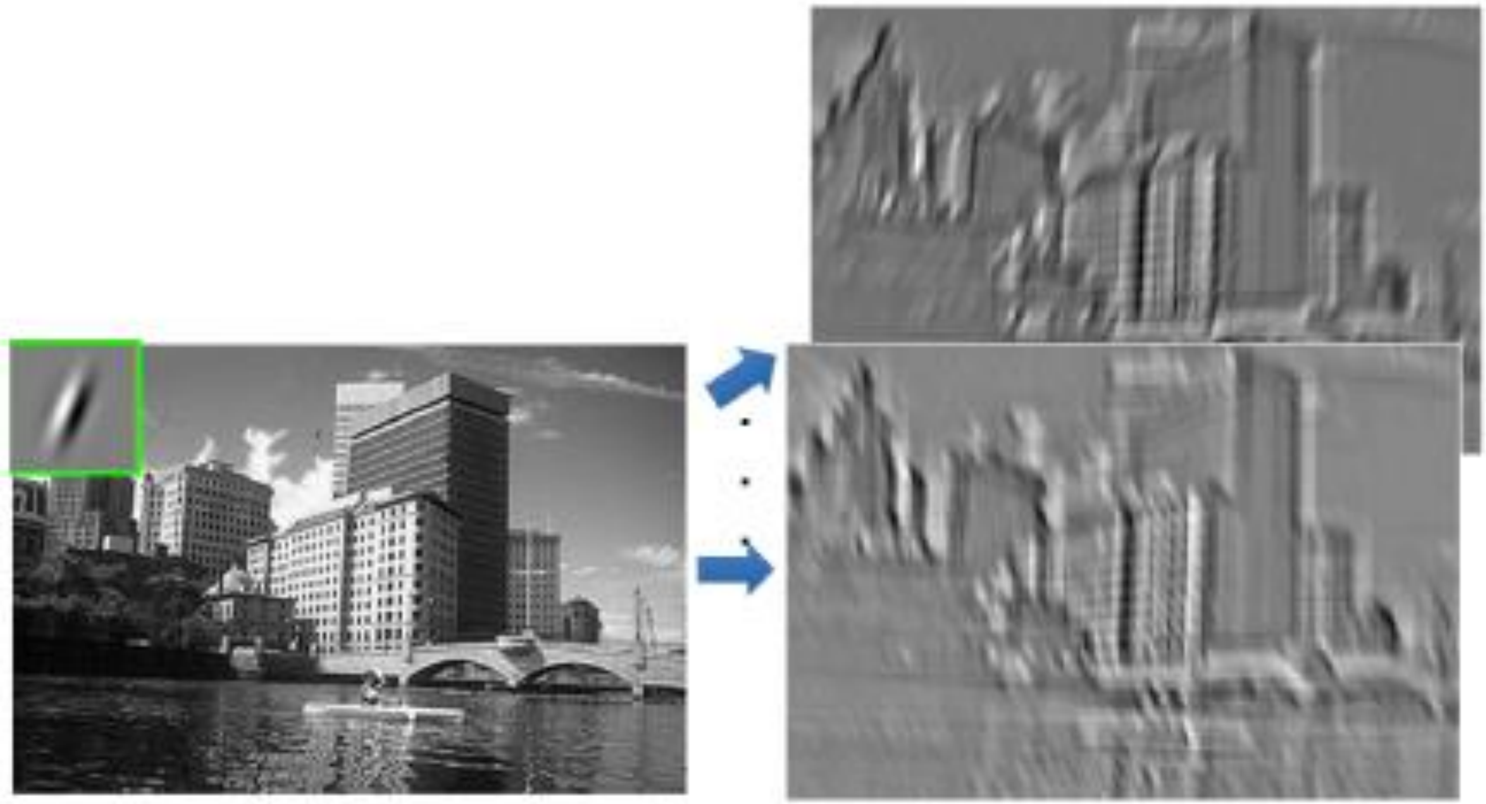
CNN … Multiple Filters





CNN … Multiple Filters (Example)

- If we use 100 filters, we get 100 feature maps



Acknowledgements

Stuart J. Russell and Peter Norvig, Tom M. Mitchell, Jiwon Jeong, Floydhub, Andrej Karpathy