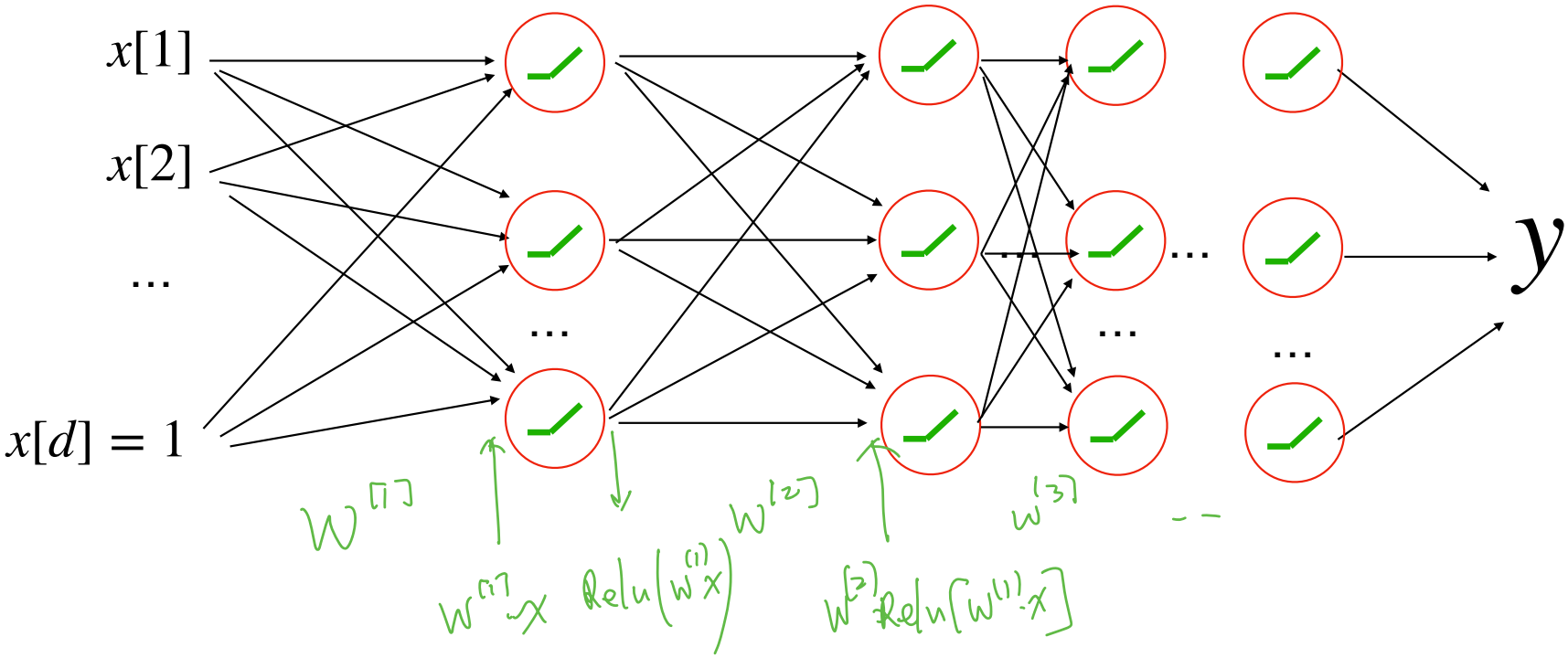
Convolutional Neural Network

Announcements

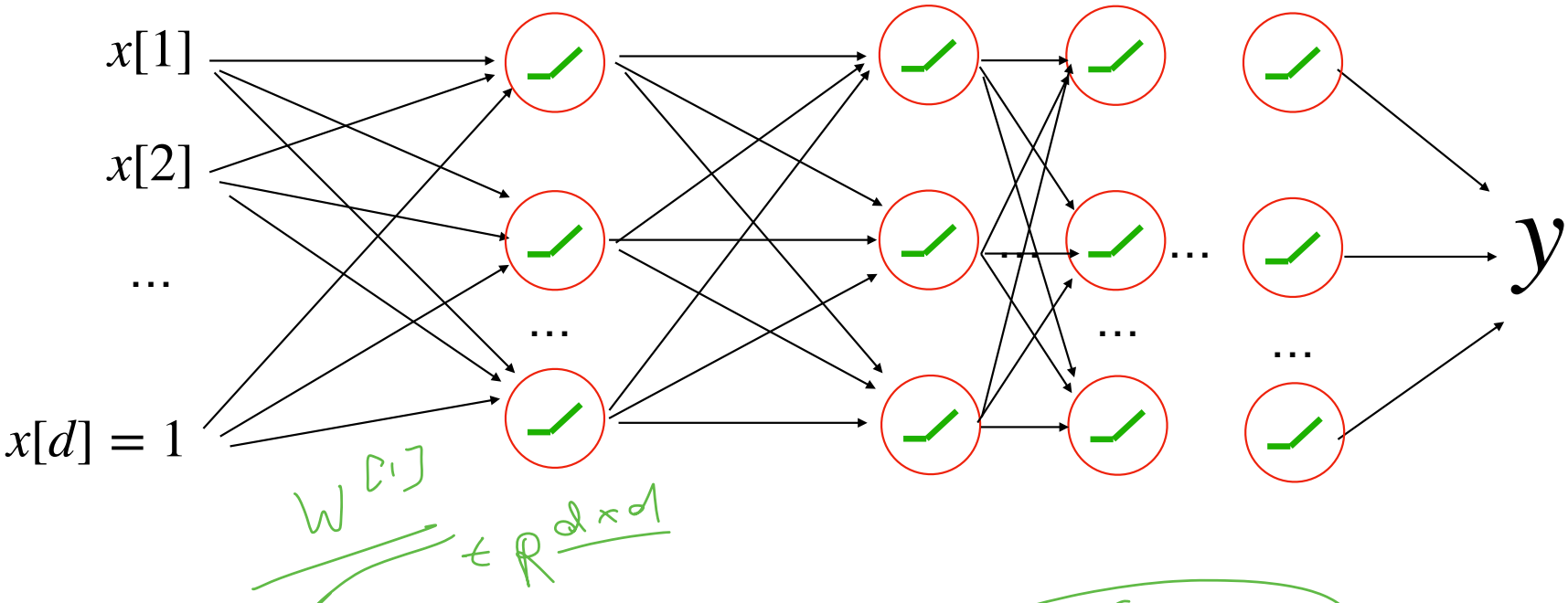
1. Releasing past semesters' final exams

2. Email for makeup final should be out today

Recap on fully connected NN



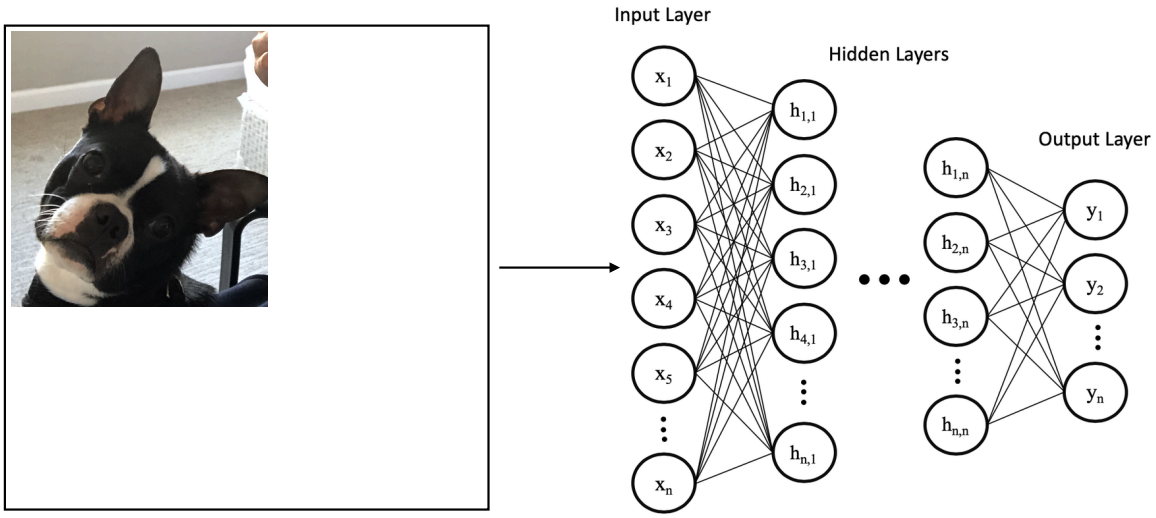
Recap on fully connected NN



Not very scalable when d is large (e.g., when d is a million)

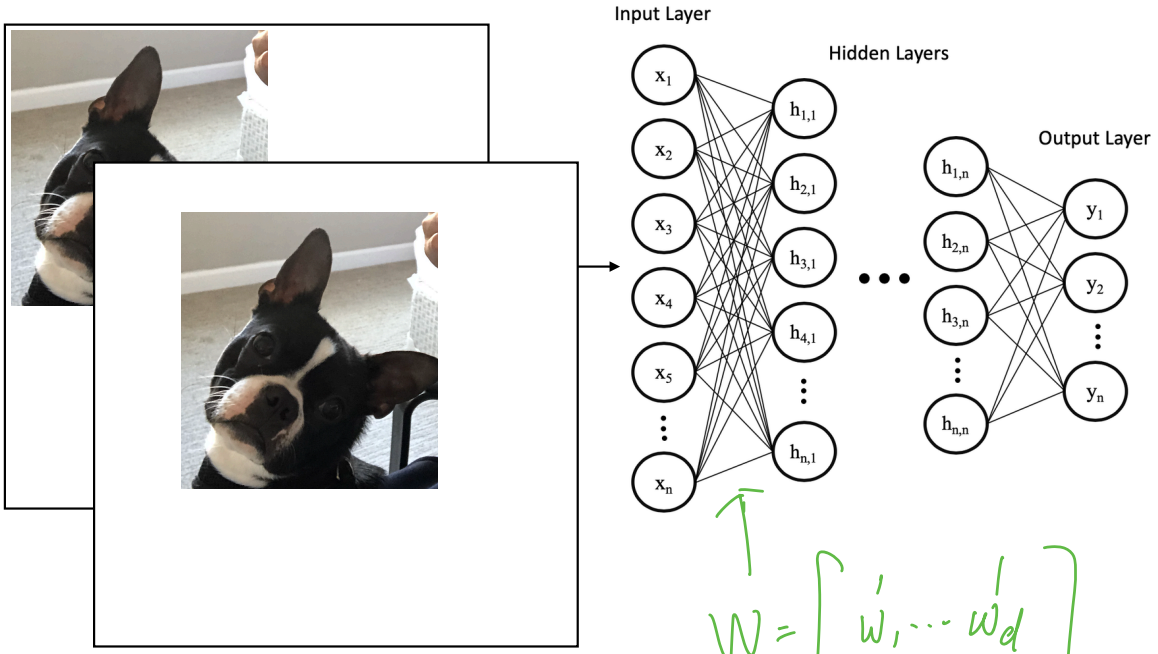
Recap on fully connected NN

Also not translation invariant



Recap on fully connected NN

Also not translation invariant



$$x = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

$$Wx = \begin{bmatrix} w_1 + w_2 \end{bmatrix}$$

$$x' = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

$$W = \begin{bmatrix} w_1 & \cdots & w_d \\ | & & | \end{bmatrix}$$

$$Wx' = \begin{bmatrix} w_3 + w_4 \end{bmatrix}$$

Objective today

Understand the convolution operator and the Convolution network
(designed for dealing w/ image inputs)

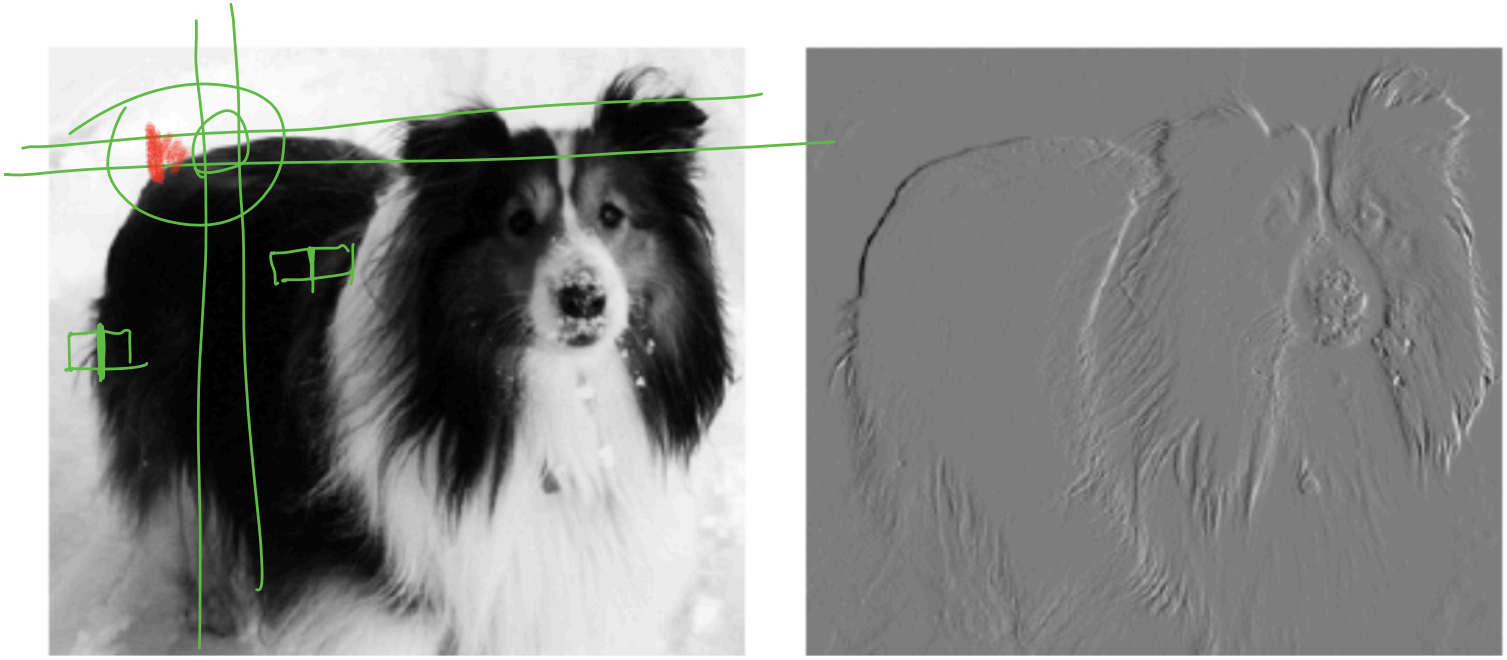
Outline today

1. Edge detector and convolution

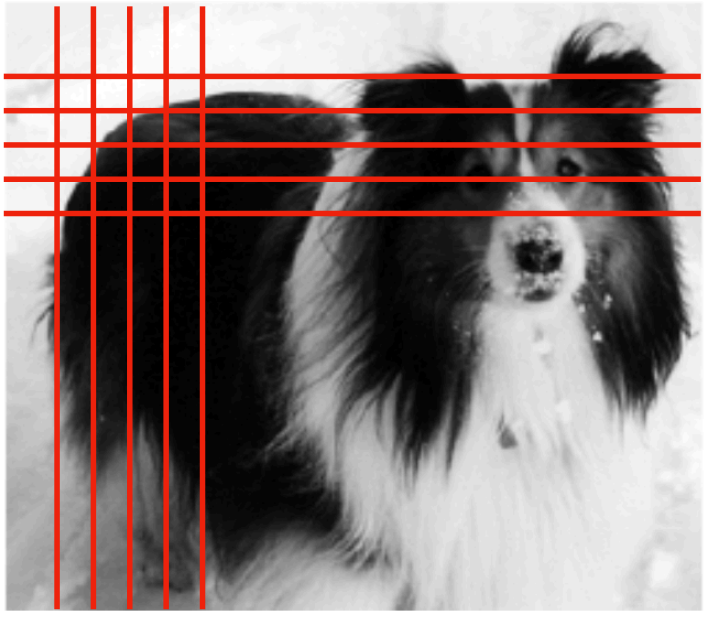
2. Convolution layer and a pooling layer

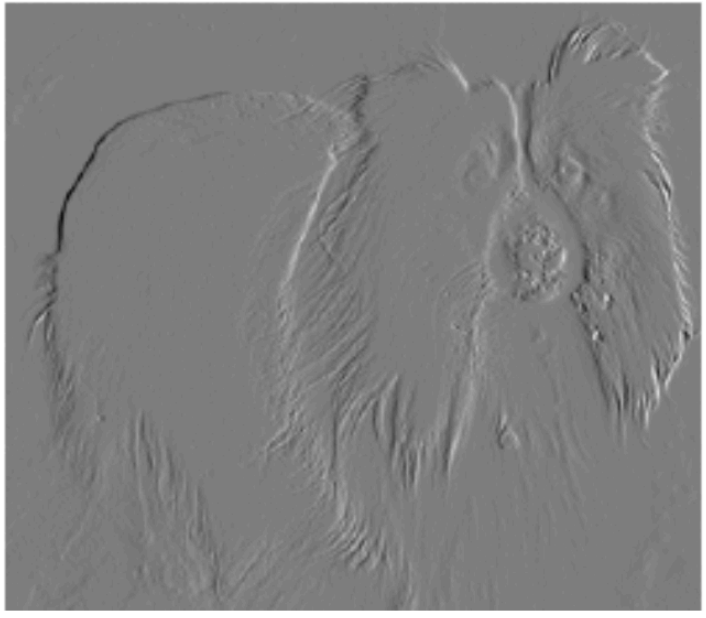
3. Case study on LeNet (ResNet)

Edge detector

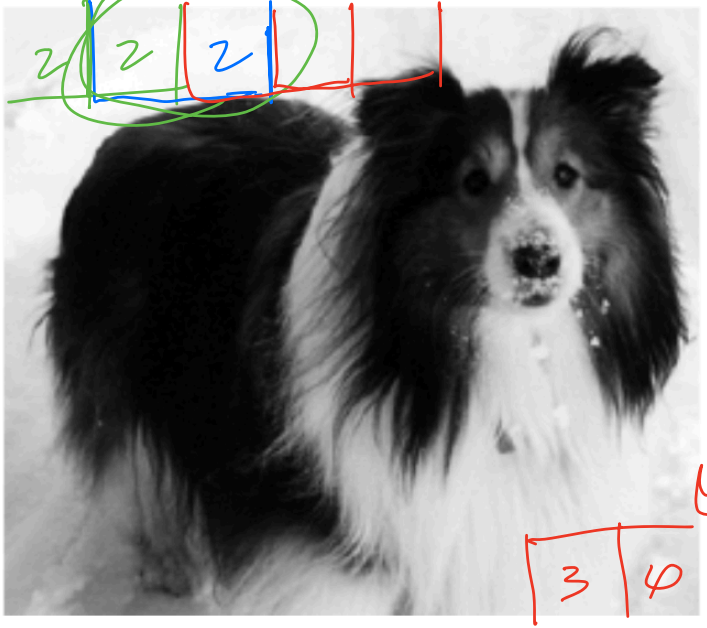


Edge detector





Implementing vertical Edge detector w/ convolution



Kernel 1x2 matrix

-1	1

3 | 0

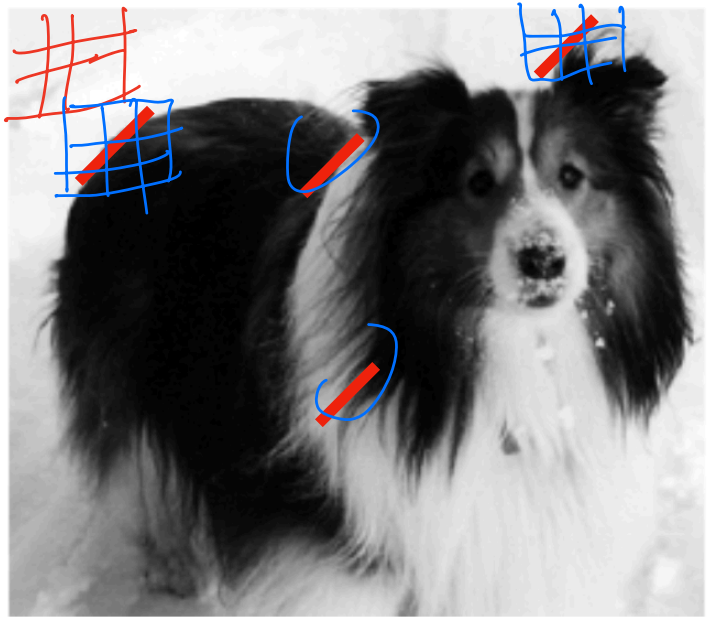
3×(-1) + 0×1 = -3

Other Edge detector

Kernel 3x3 matrix

0	0	1
0	1	0
1	0	0

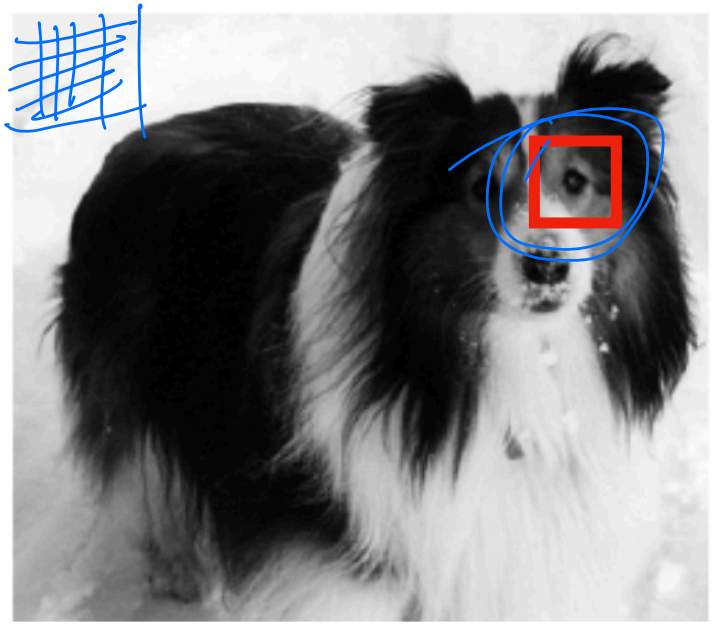
Other Edge detector

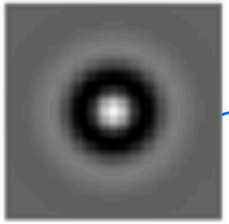


Kernel 3x3 matrix

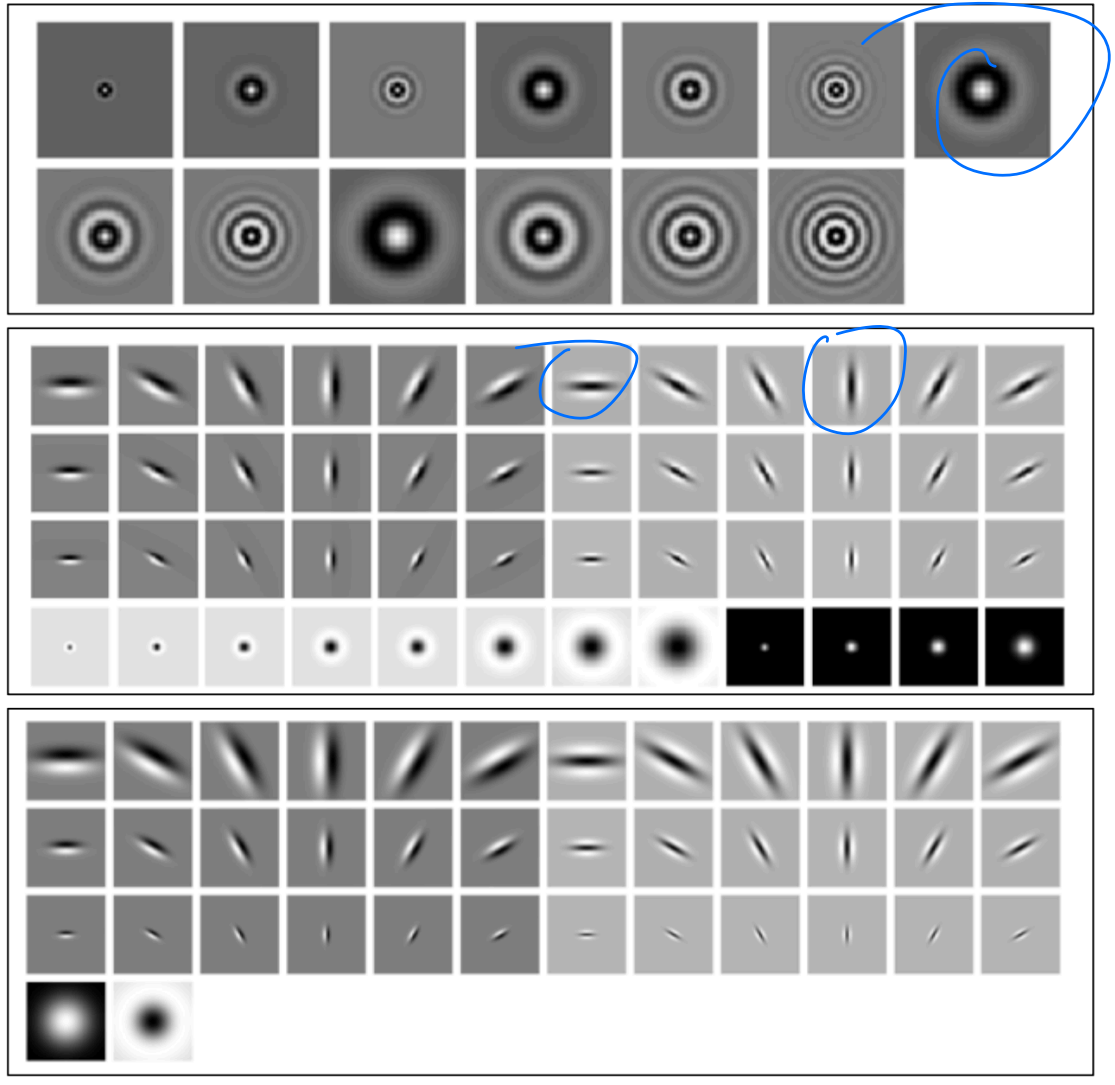
0	0	1
0	1	0
1	0	0

More examples





The Filter bank



Summary of Convolution operator

Image (2d matrix)

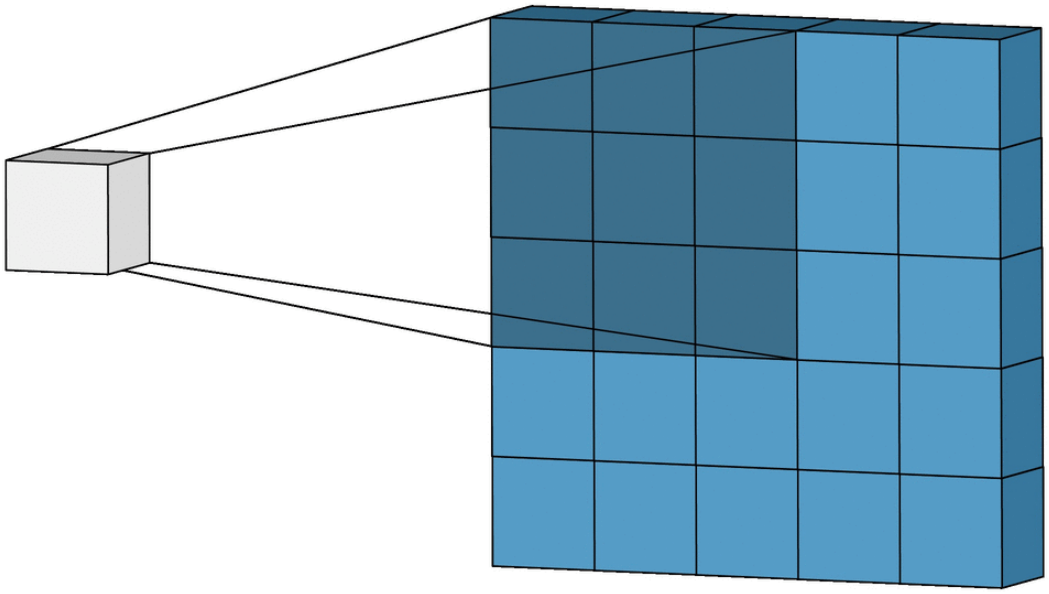
a	b	c	d
e	f	g	h
i	j	k	l

Kernel 2x2 matrix

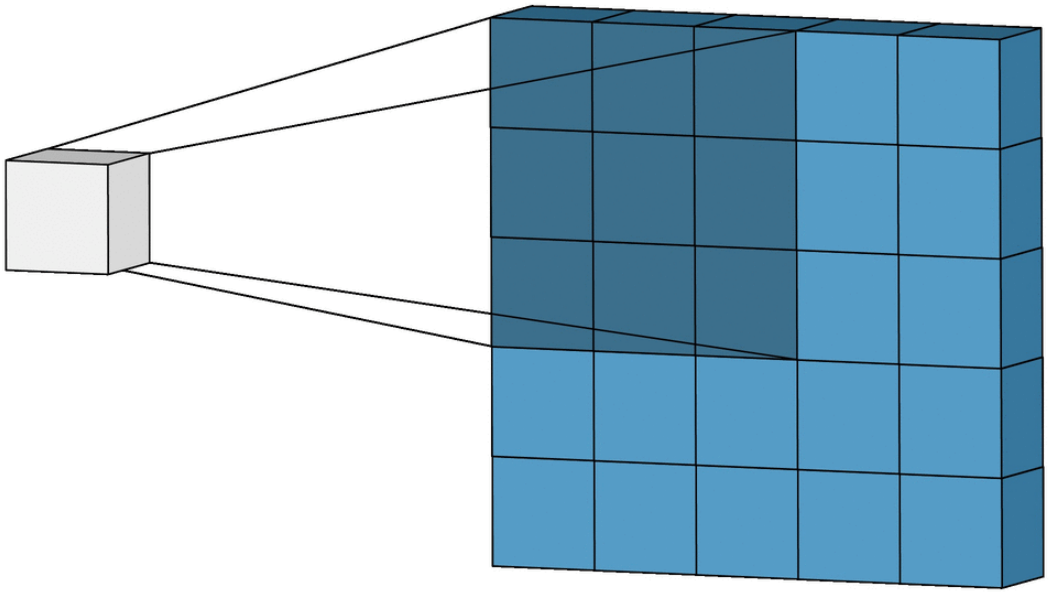
W	X
Y	Z

aw+bx +ey+fz	bw+cx +fy+gz	cw+dx +gy+hz
ew+xf +iy+jz		

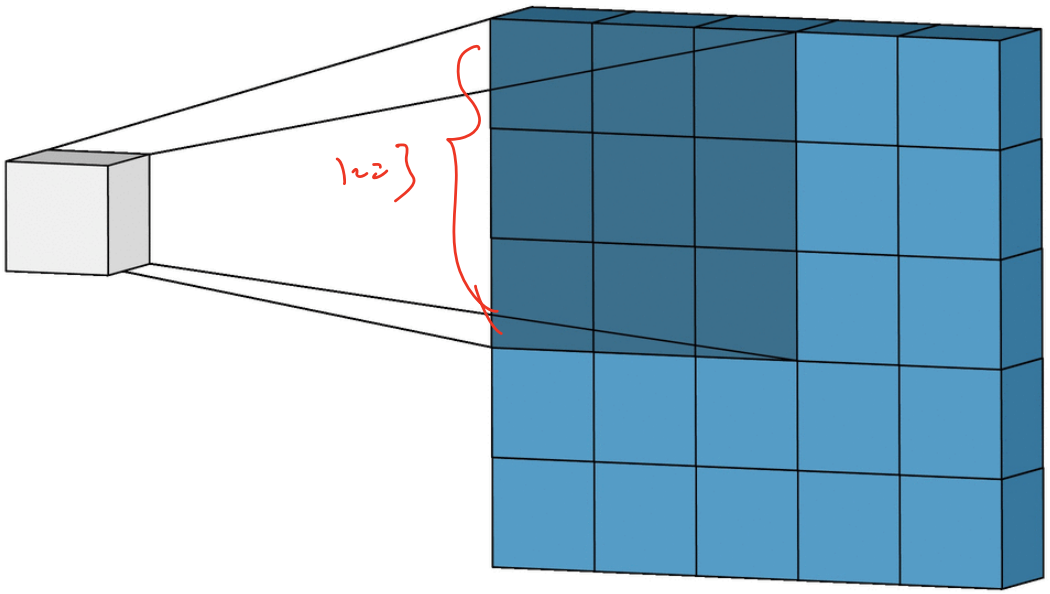
Visualization of convolution



Visualization of convolution



Visualization of convolution



Q: if the image is $d \times d$, and kernel is $k \times k$, what is the dim of the output matrix?

Convolution over volumes (3d tensor)

RGB Image

$d \times d \times 3$ → # of channels

Convolution over volumes (3d tensor)

RGB Image

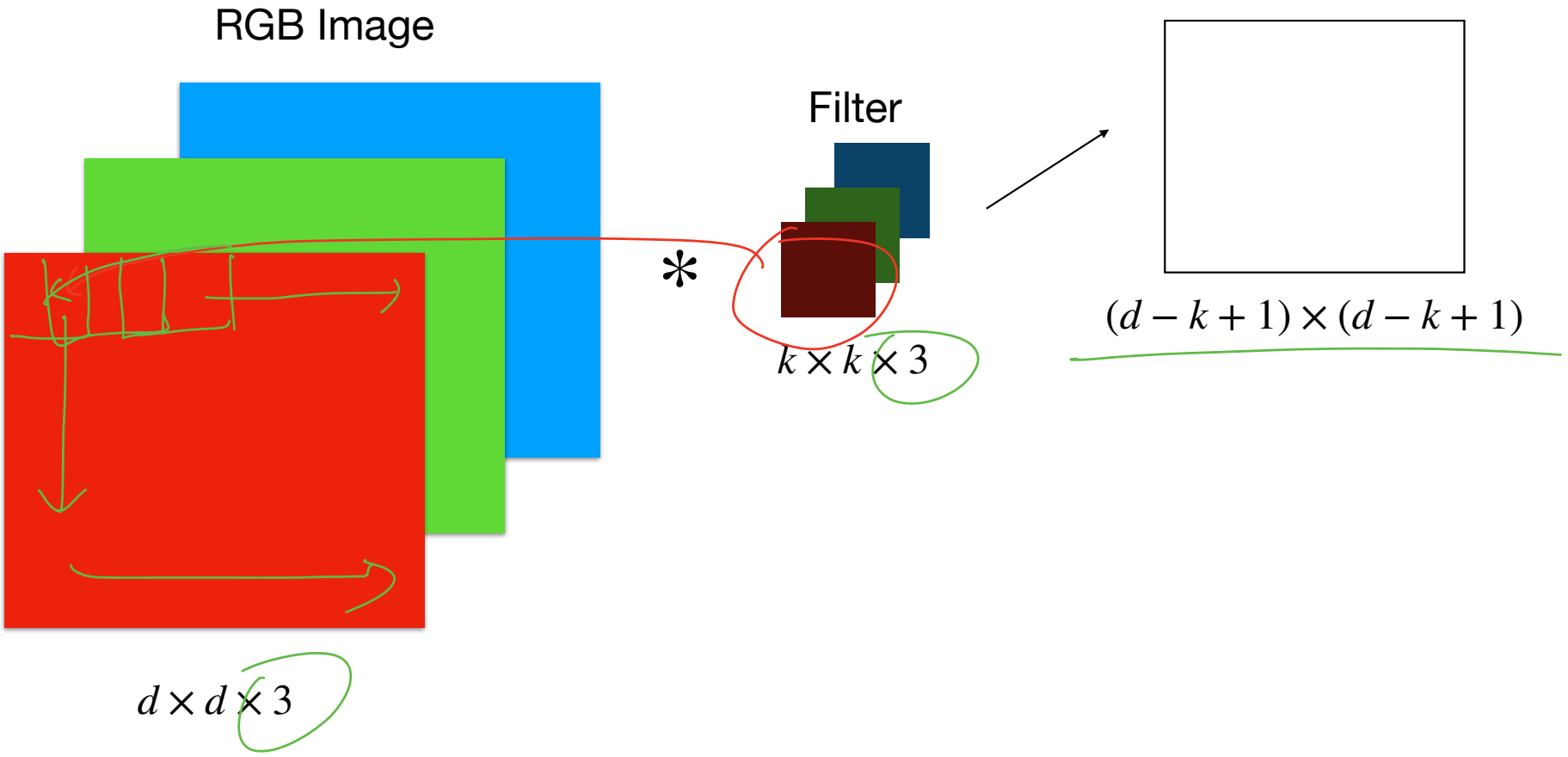
$d \times d \times 3$

$*$

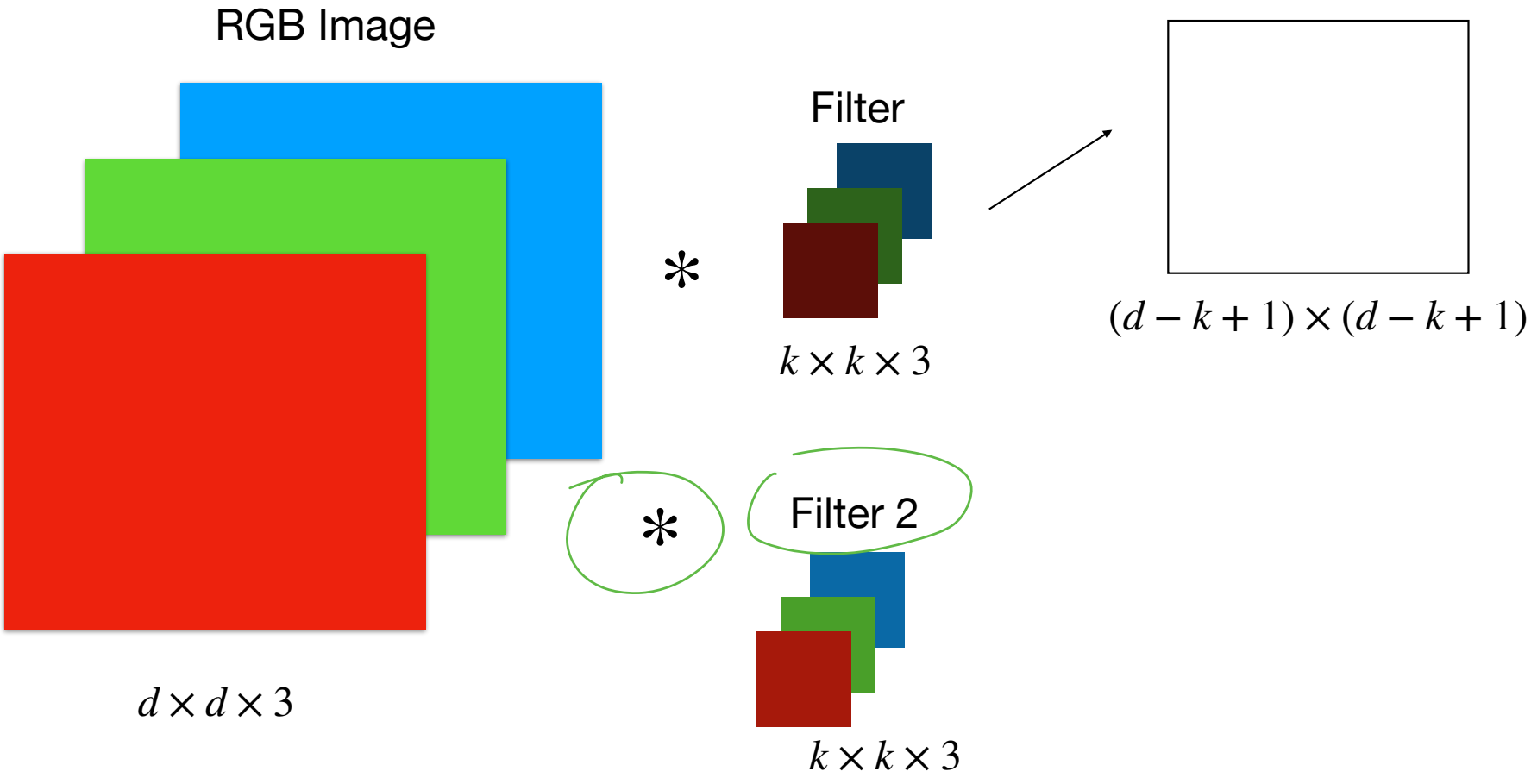
Filter

$k \times k \times 3$

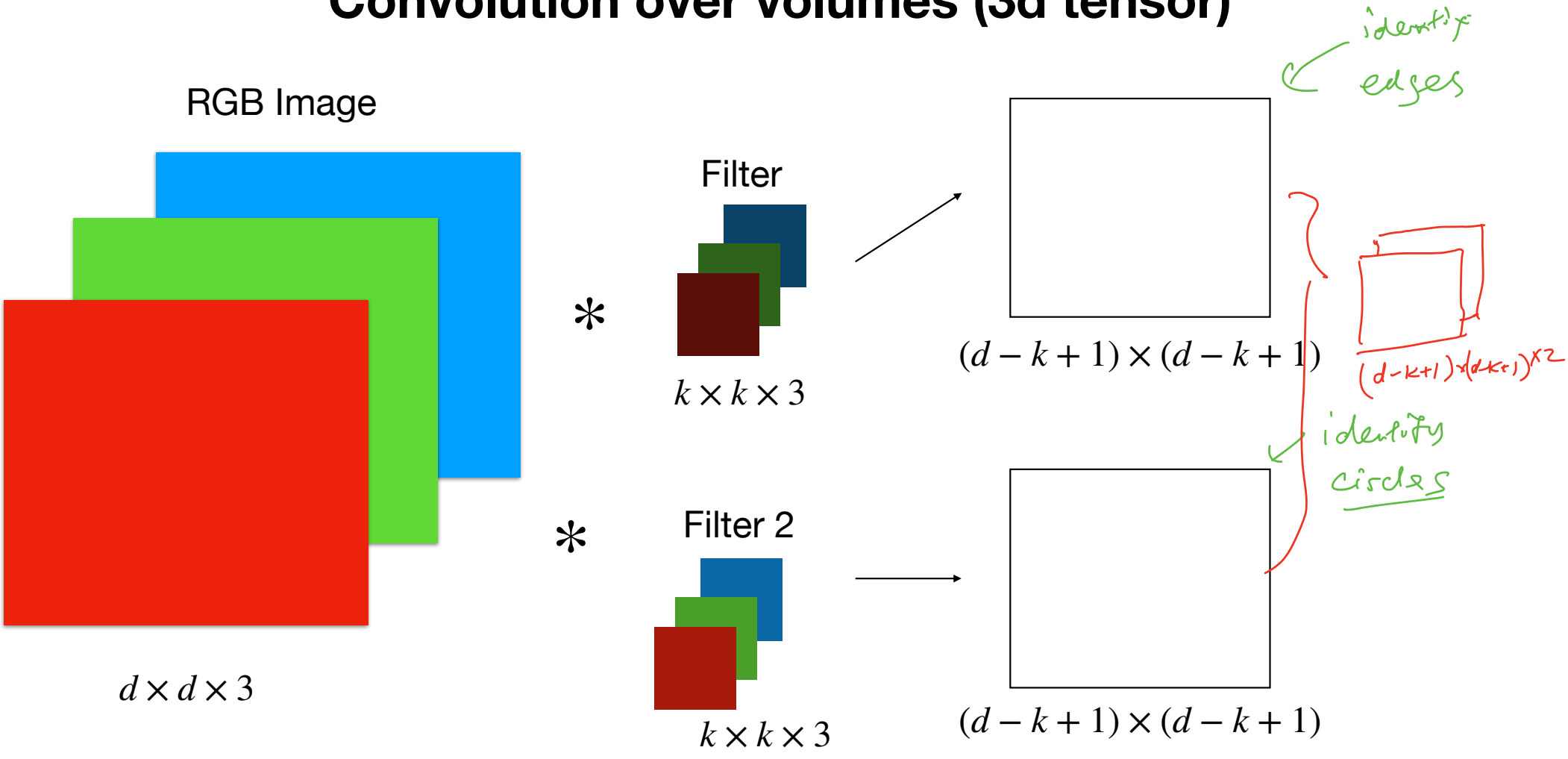
Convolution over volumes (3d tensor)



Convolution over volumes (3d tensor)



Convolution over volumes (3d tensor)



Key question

Can we learn these detectors / filters in an end-to-end fashion?

Outline today

1. Edge detector and convolution

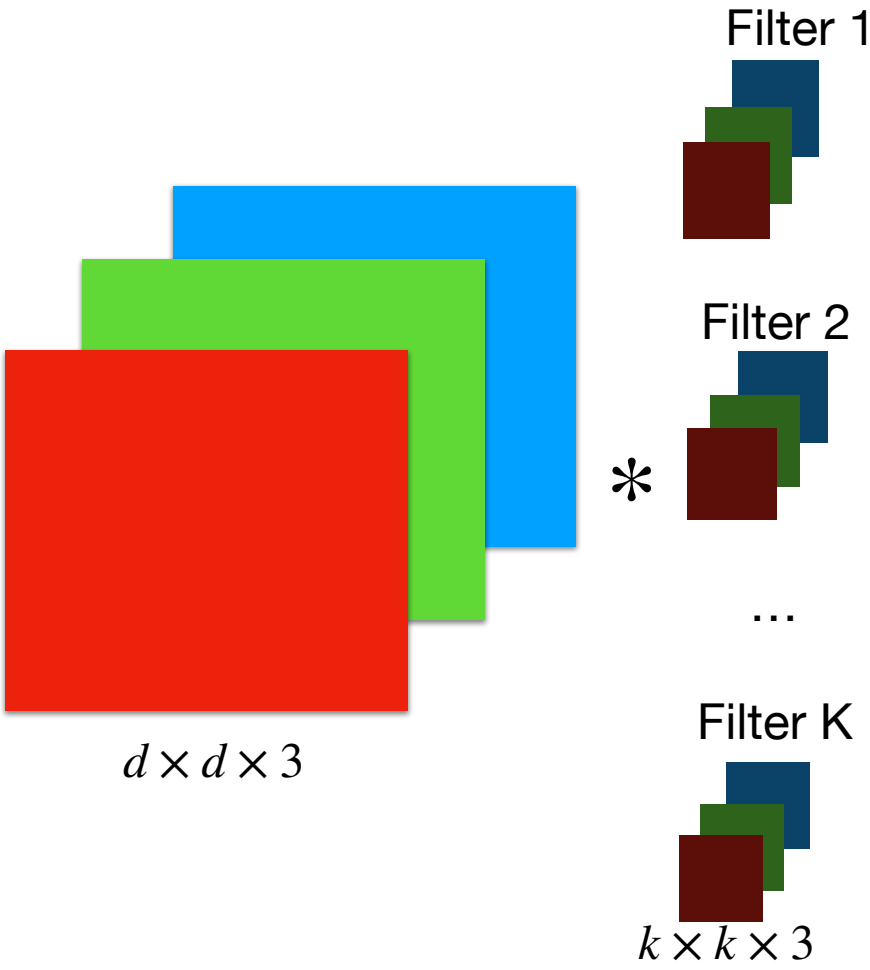
2. Convolution layer and a pooling layer

3. Case study on LeNet (ResNet)

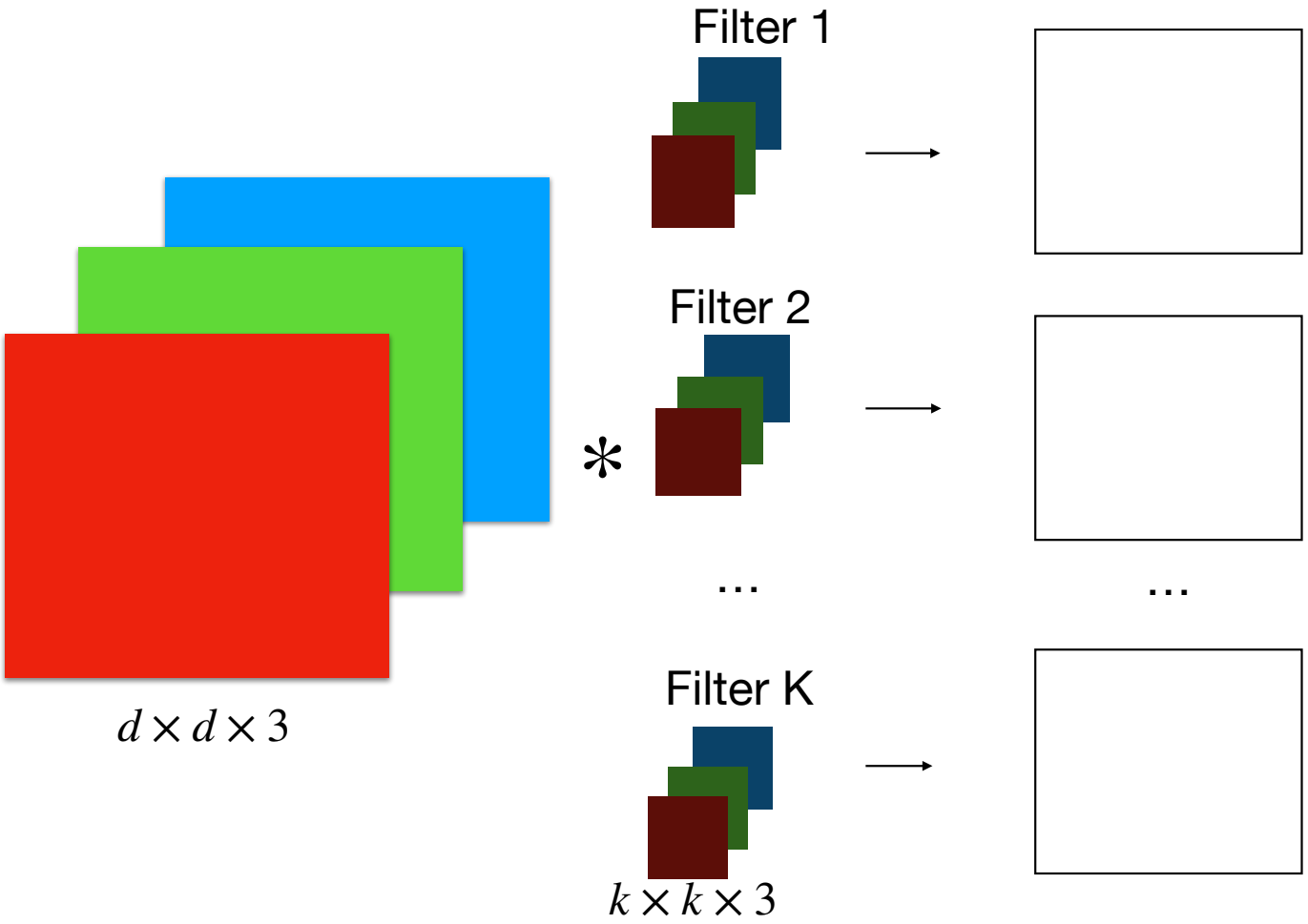
Building a convolution layer

$d \times d \times 3$

Building a convolution layer



Building a convolution layer



Building a convolution layer

$d \times d \times 3$

Filter 1

Filter 2

…

Filter K

$k \times k \times 3$

$*$

ReLu $(\quad + b_1)$

ReLu $(\quad + b_2)$

…

ReLu $(\quad + b_K)$

$b_i \in \mathbb{R}$

$(d-k+1) \times (d-k+1) \times K$

$((k^2) \times 3) \times K + K$

Pooling layer

We use a pooling layer to downsize the inputs

Pooling layer

We use a pooling layer to downsize the inputs

e.g., Max pooling (2x2 filter and stride 2)

Pooling layer

We use a pooling layer to downsize the inputs

e.g., Max pooling (2x2 filter and stride 2)

e.g.,some Output of a convolution layer

1	3	0	6
5	4	12	2
7	1	9	0
4	3	1	8

5	12
7	9

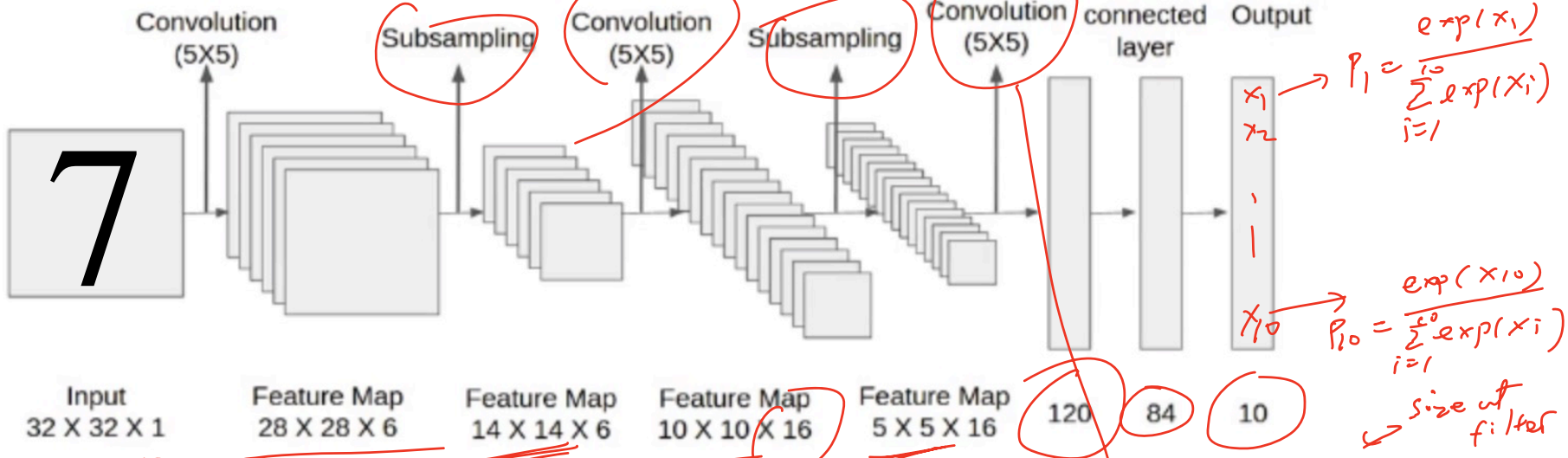
Outline today

1. Edge detector and convolution

2. Convolution layer and a pooling layer

3. Case study on LeNet

LeNet 5



5×5 kernel
6 filters

2×2 filter, stride 2

Conv: 5×5×6 size of filter
16 filters

$P_1 = \frac{\exp(x_1)}{\sum_{i=1}^{10} \exp(x_i)}$

$P_{10} = \frac{\exp(x_{10})}{\sum_{i=1}^{10} \exp(x_i)}$

size of filter

Conv: 5×5×16
120 filters

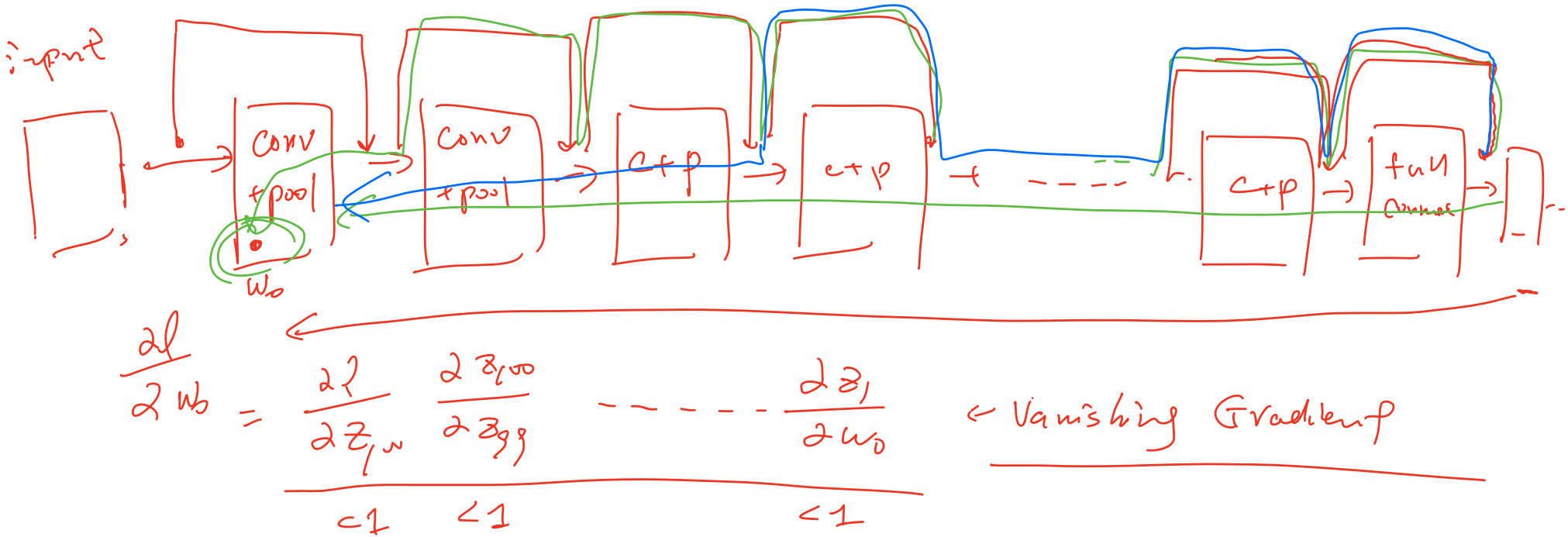
⇒ 1×1×120

$10 = (14 - 5 + 1)$

$28 = 32 - 5 + 1$

# of channels 1

ResNet



(The last reading quiz in on the classic ResNet paper!)

Summary for today

Convolutional neural network works well for images where pixels have strong local spatial correlations

Summary for today

Convolutional neural network works well for images where pixels have strong local spatial correlations

Limitations:
convolution cannot capture global information (correlation among very distant pixels);
Fine-grained details may lost during pooling.