كلية الهندسة
قسم هندسة الروبوت و الأنظمة الذكية

الشبكات العصبية

محاضرة 8

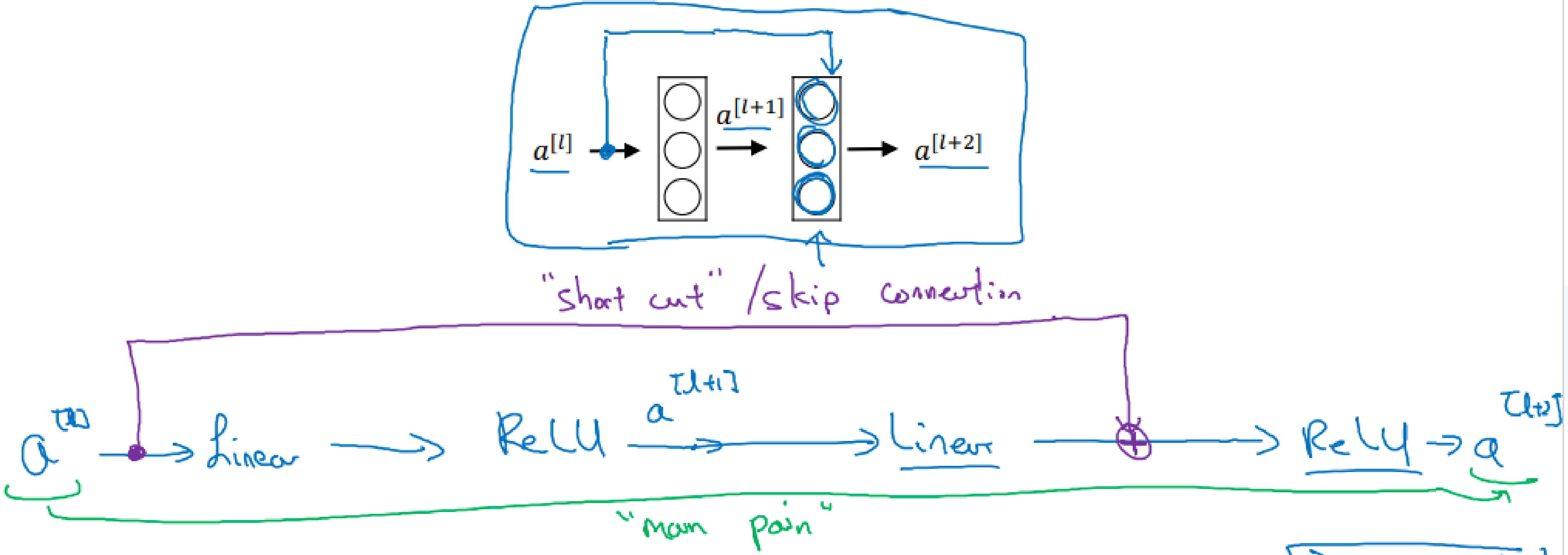
Convolutional Neural Networks (Case Studies)

د. فادي متوج

Case Studies

Residual Networks (ResNets)

Residual block

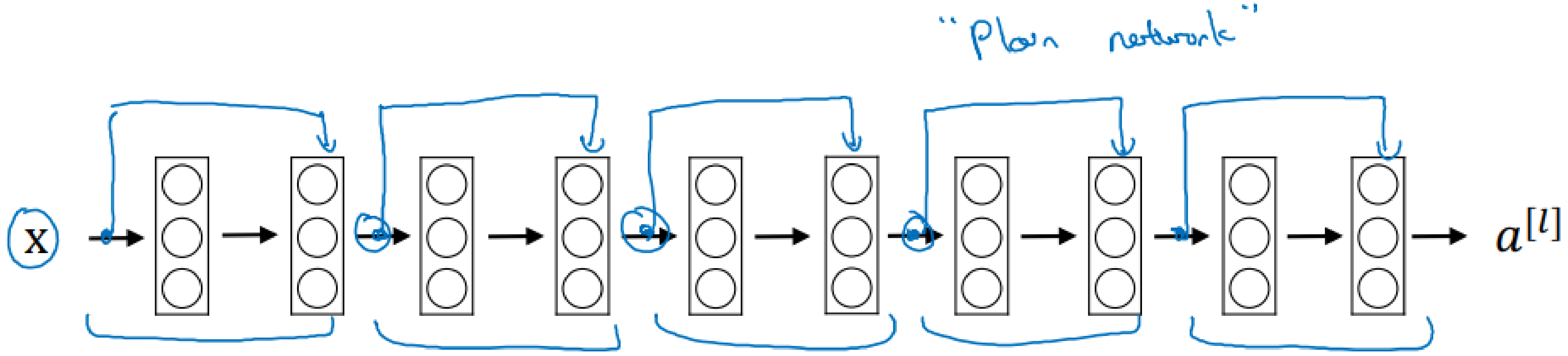


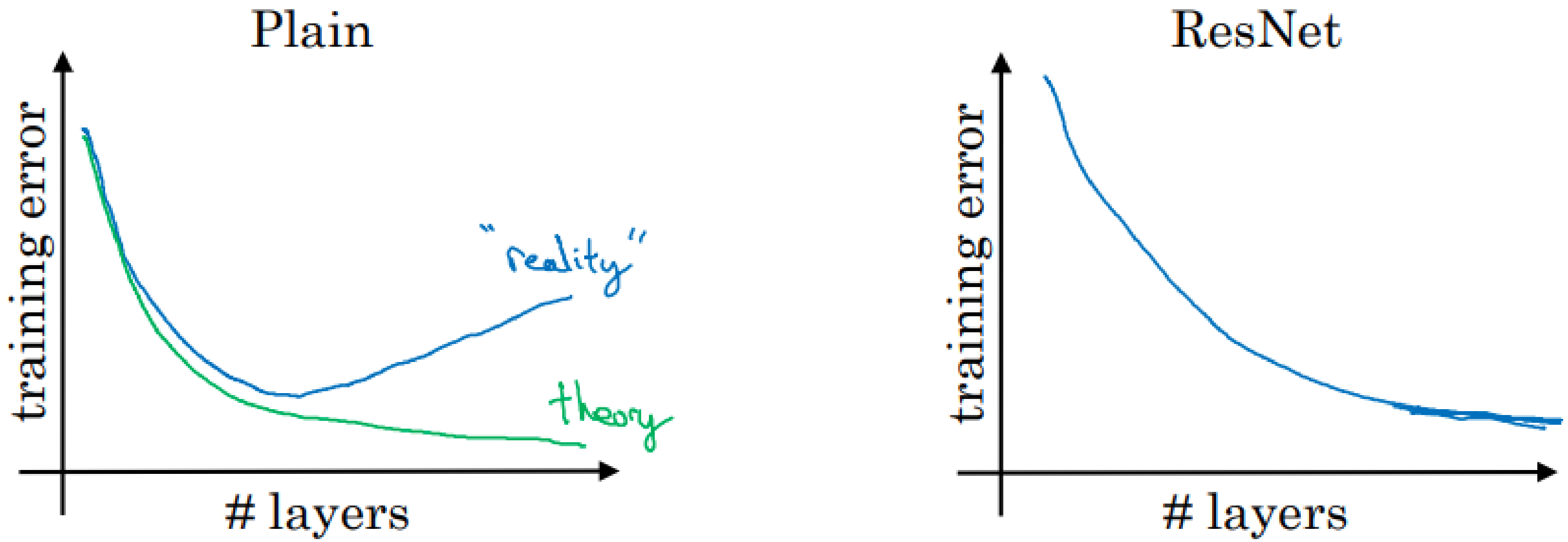
$z^{[l+1]} = W^{[l+1]} a^{[l]} + b^{[l+1]}$ $\quad a^{[l+1]} = g(z^{[l+1]})$ $\quad z^{[l+2]} = W^{[l+2]} a^{[l+1]} + b^{[l+2]}$ $\quad$ ~~$a^{[l+2]} = g(z^{[l+2]})$~~

$$a^{[l+2]} = g(z^{[l+2]} + a^{[l]})$$

[He et al., 2015. Deep residual networks for image recognition]

Residual Network



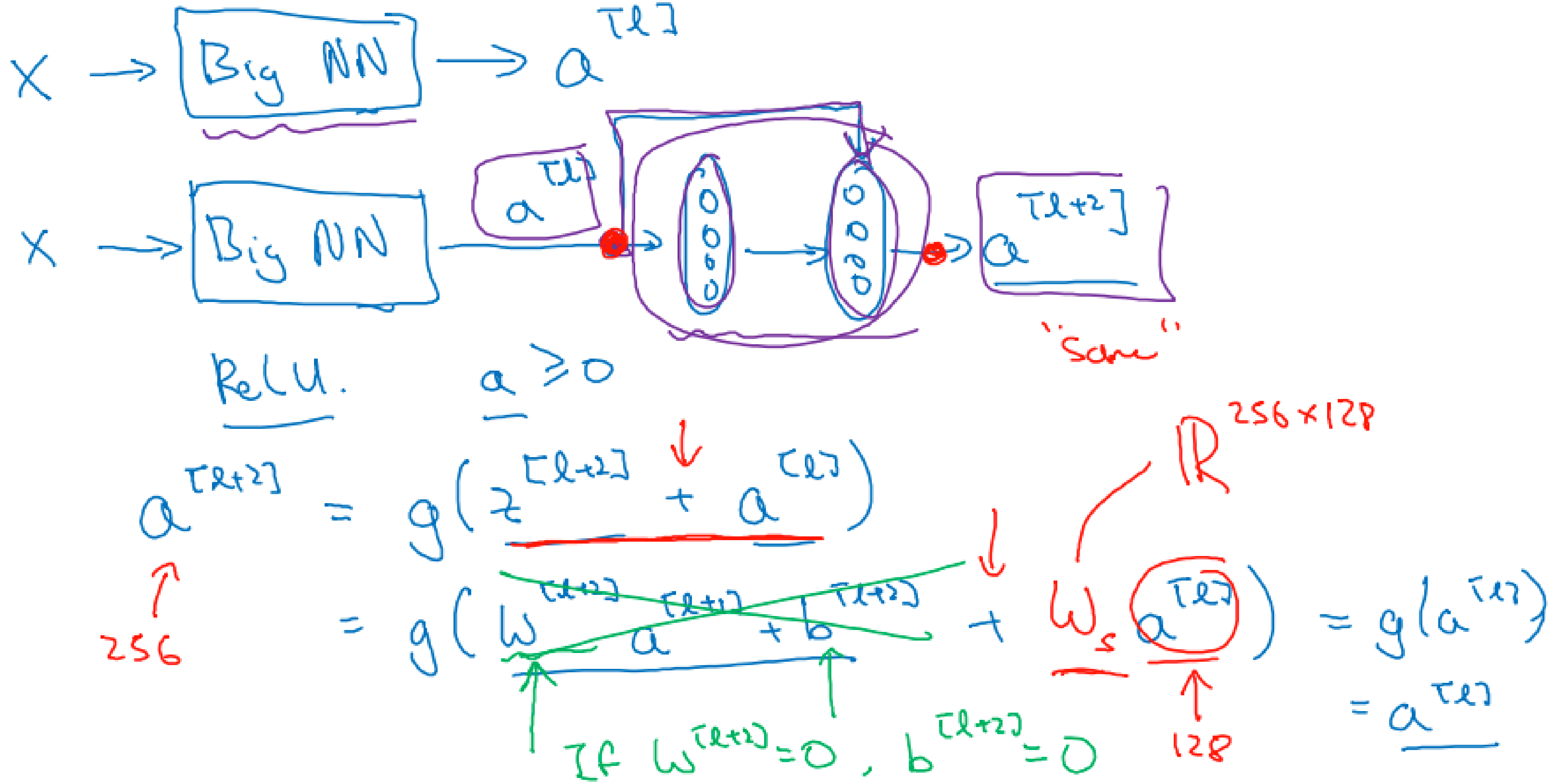


[He et al., 2015. Deep residual networks for image recognition]

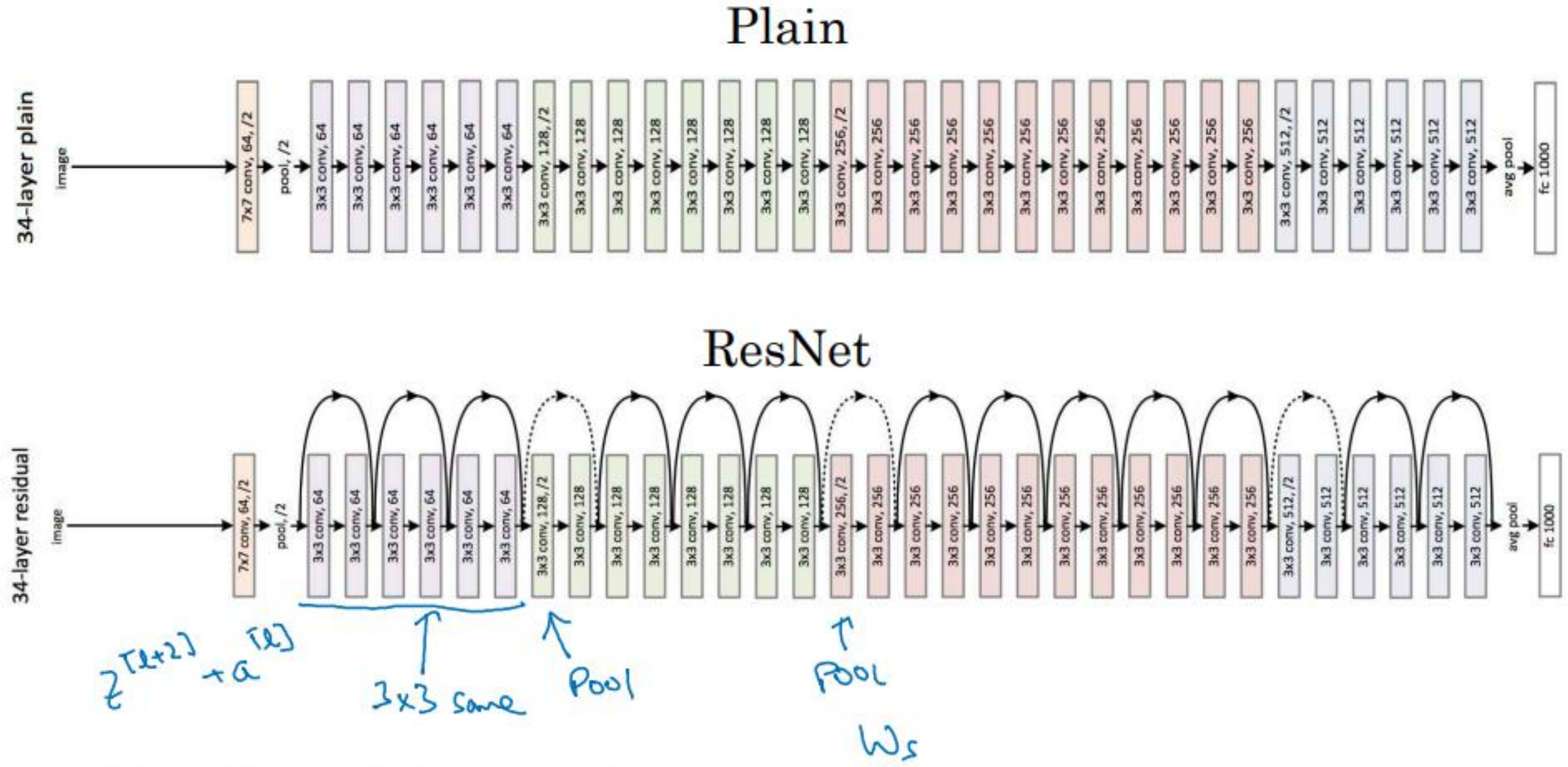
Case Studies

Why ResNets work

Why do residual networks work?



ResNet



[He et al., 2015. Deep residual networks for image recognition]

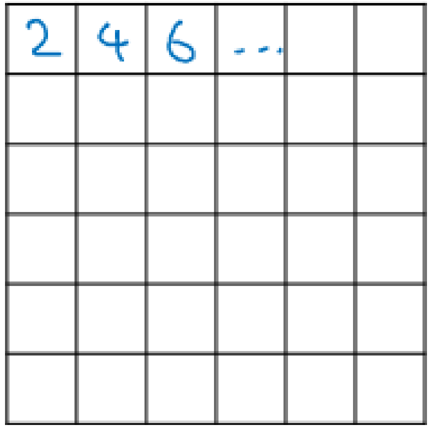
Case Studies

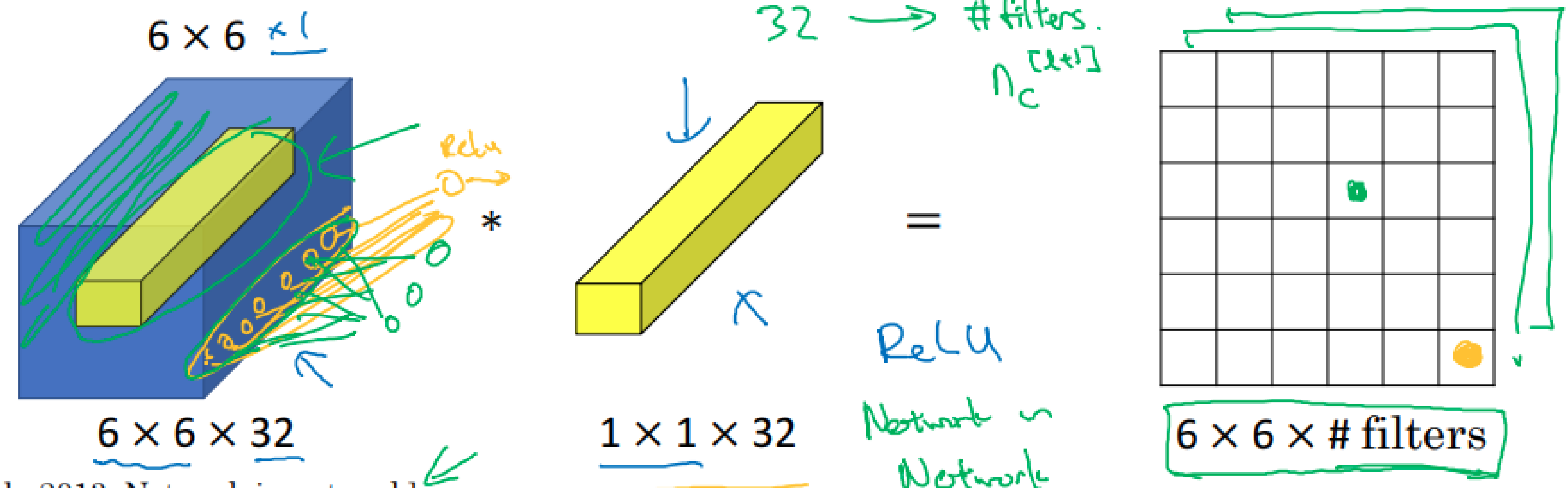
Network in Network and 1×1 convolutions

Why does a 1 × 1 convolution do?

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

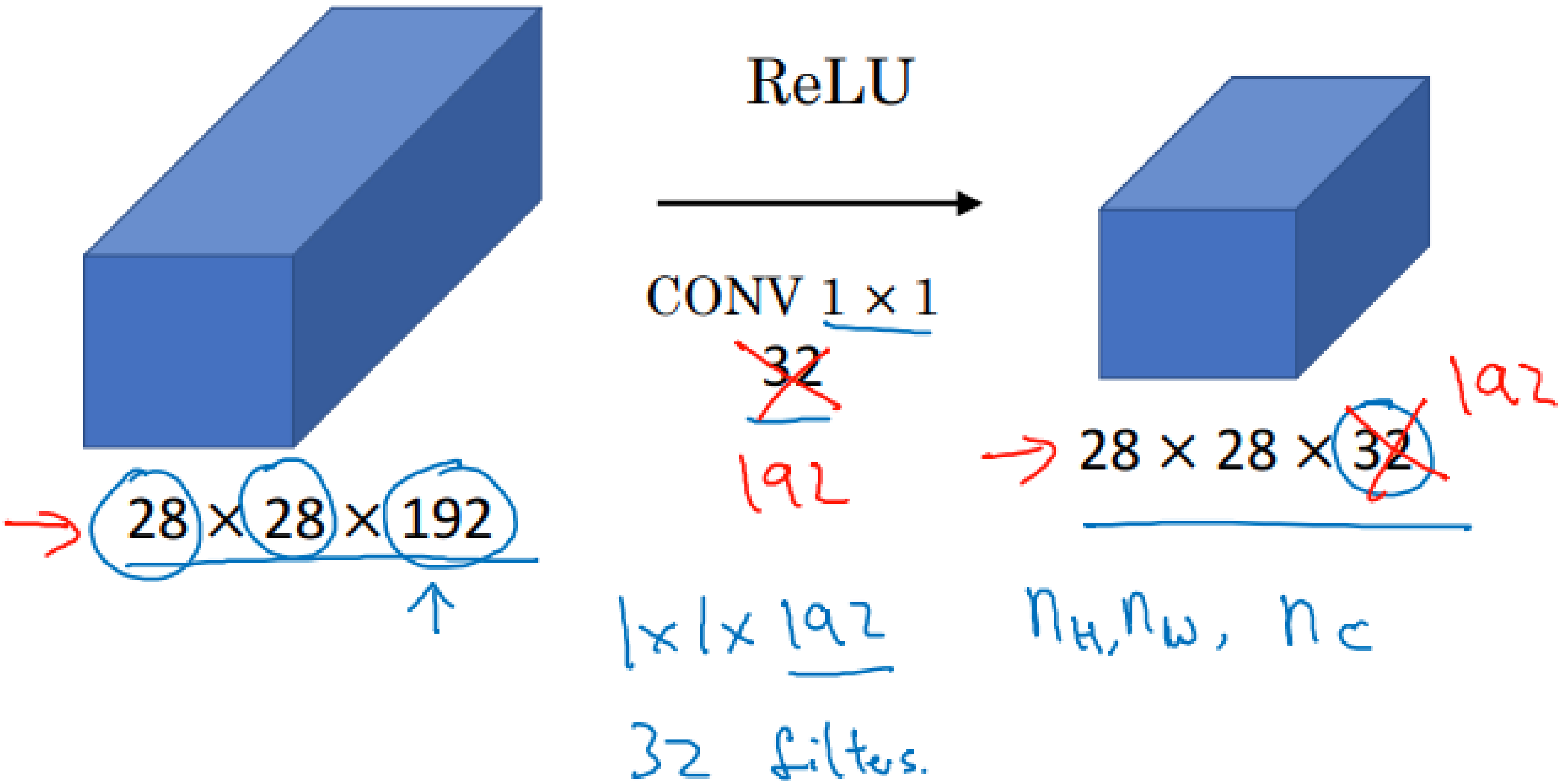
6 × 6 × 1 * 2 =



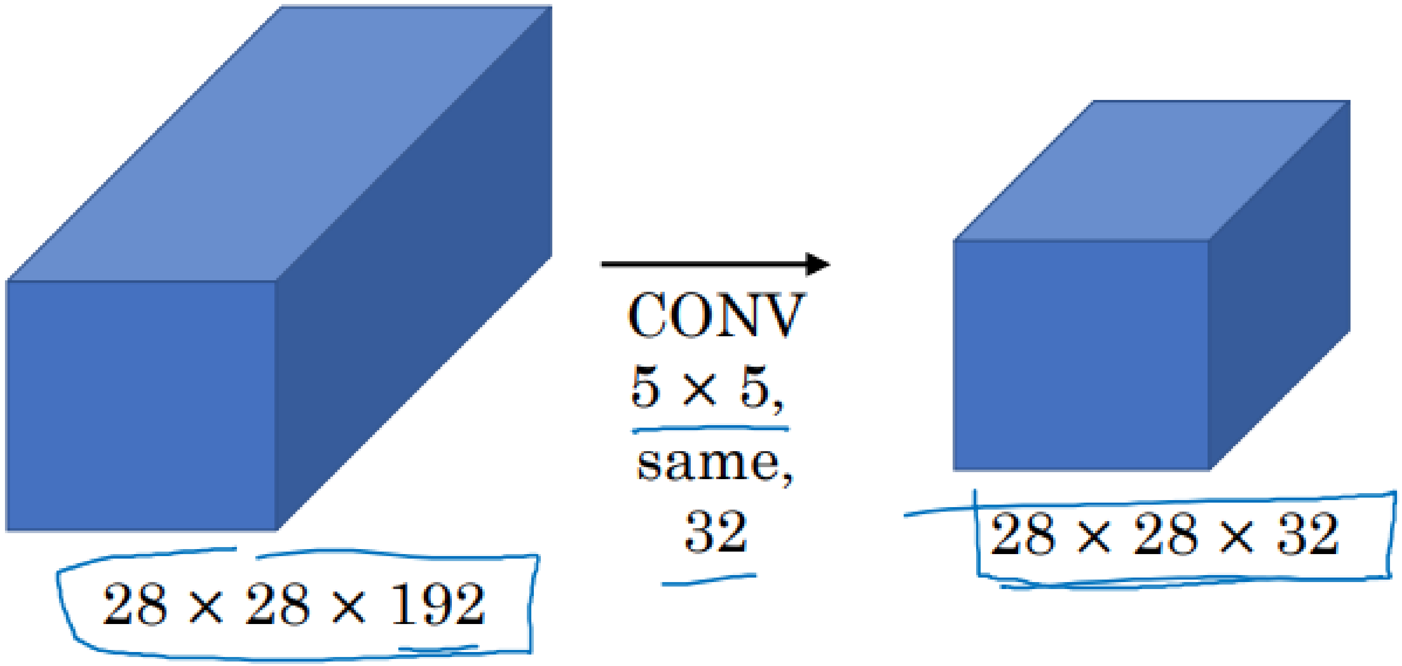


[Lin et al., 2013. Network in network]

Using 1×1 convolutions



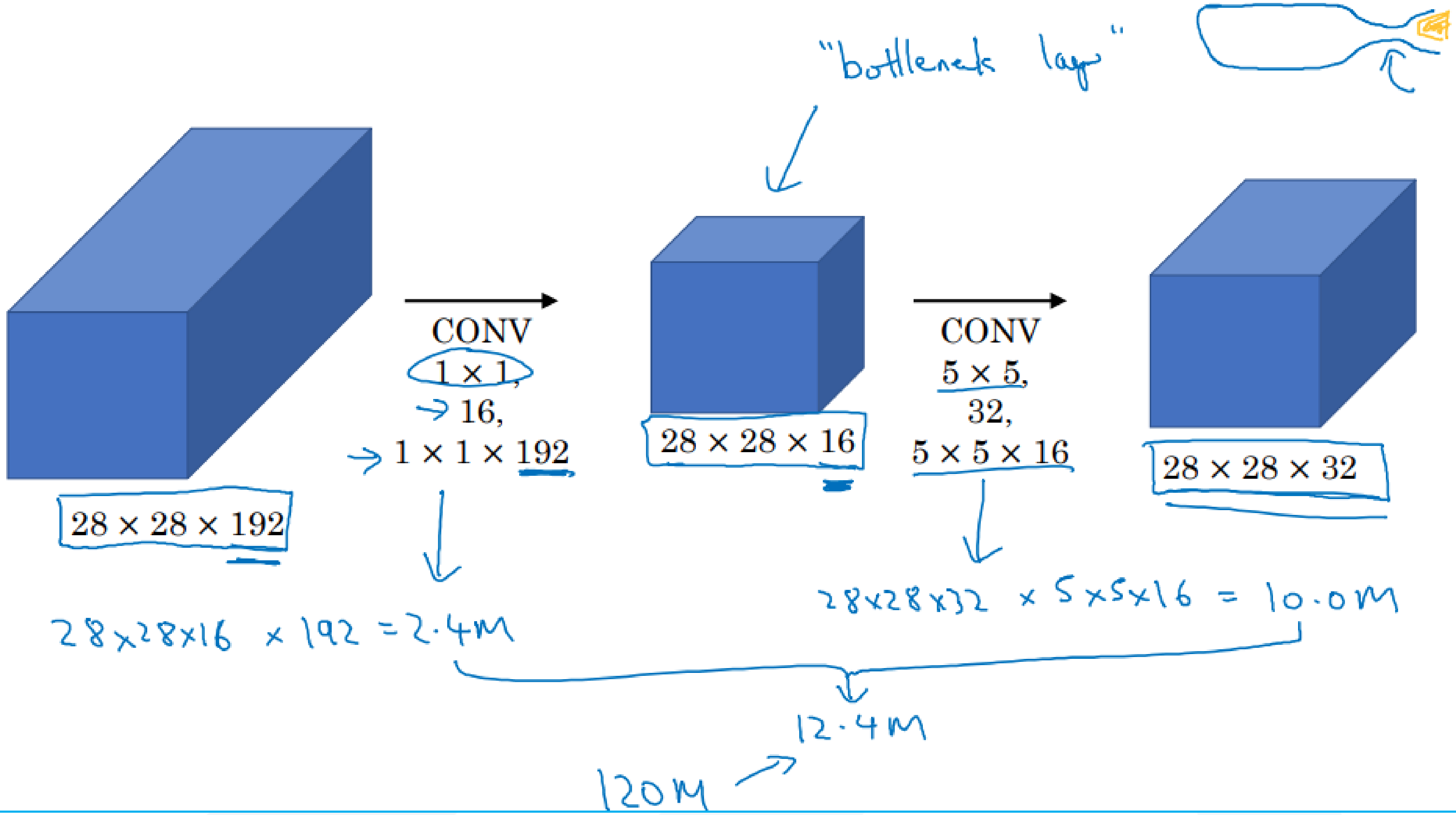
The problem of computational cost



32 filters. filters are 5×5×192.

28×28×32 × 5×5×192 = 120M.

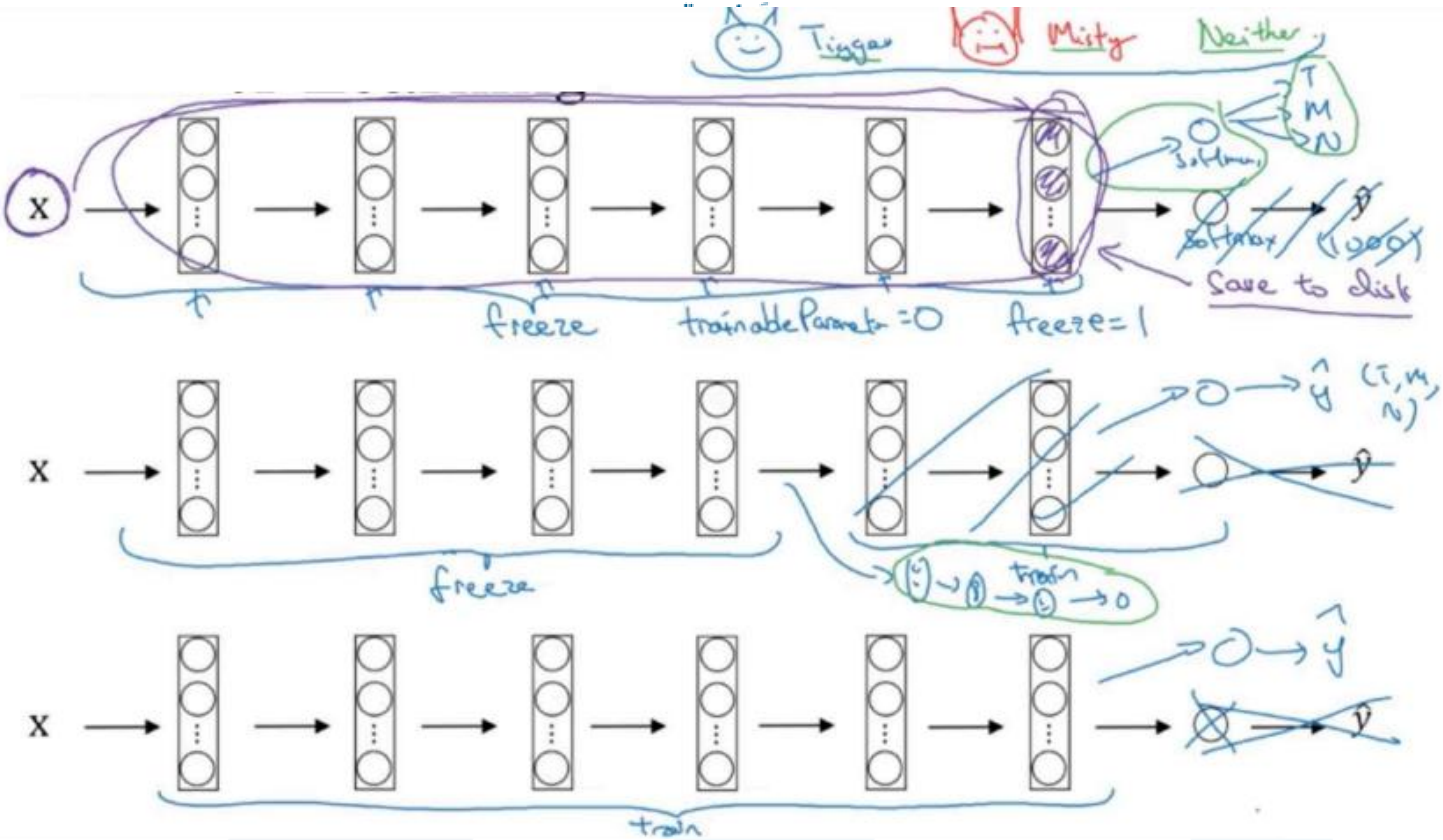
Using 1×1 convolution



Practical advice for using ConvNet

Transfer Learning

Transfer Learning

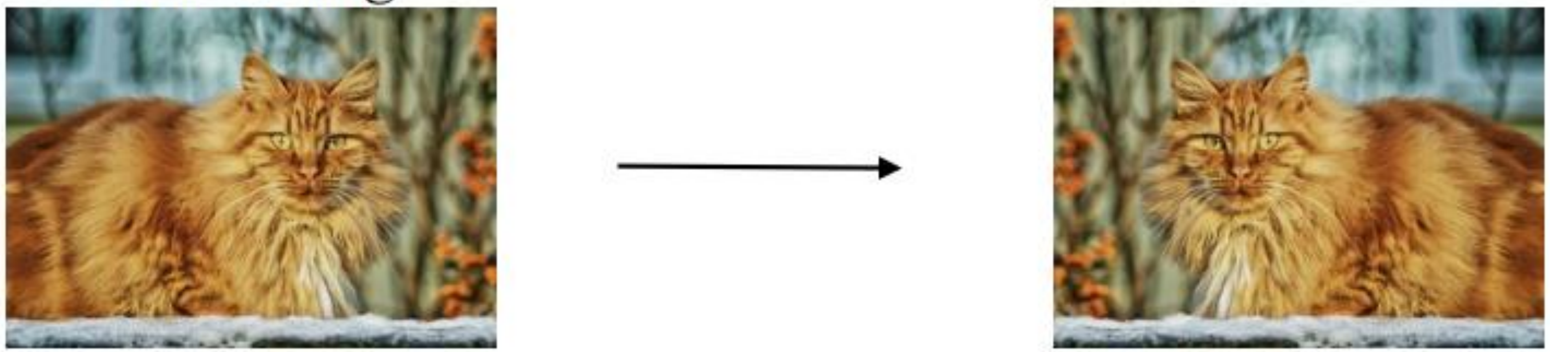


Practical advice for using ConvNet

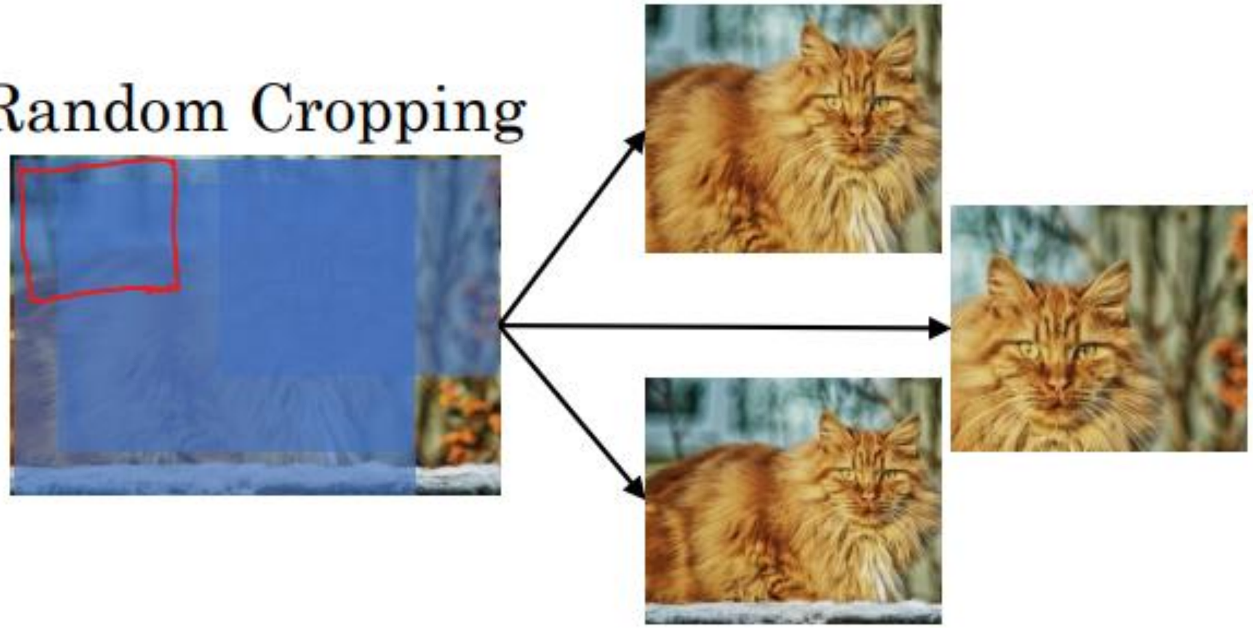
Data augmentation

Common augmentation method

Mirroring

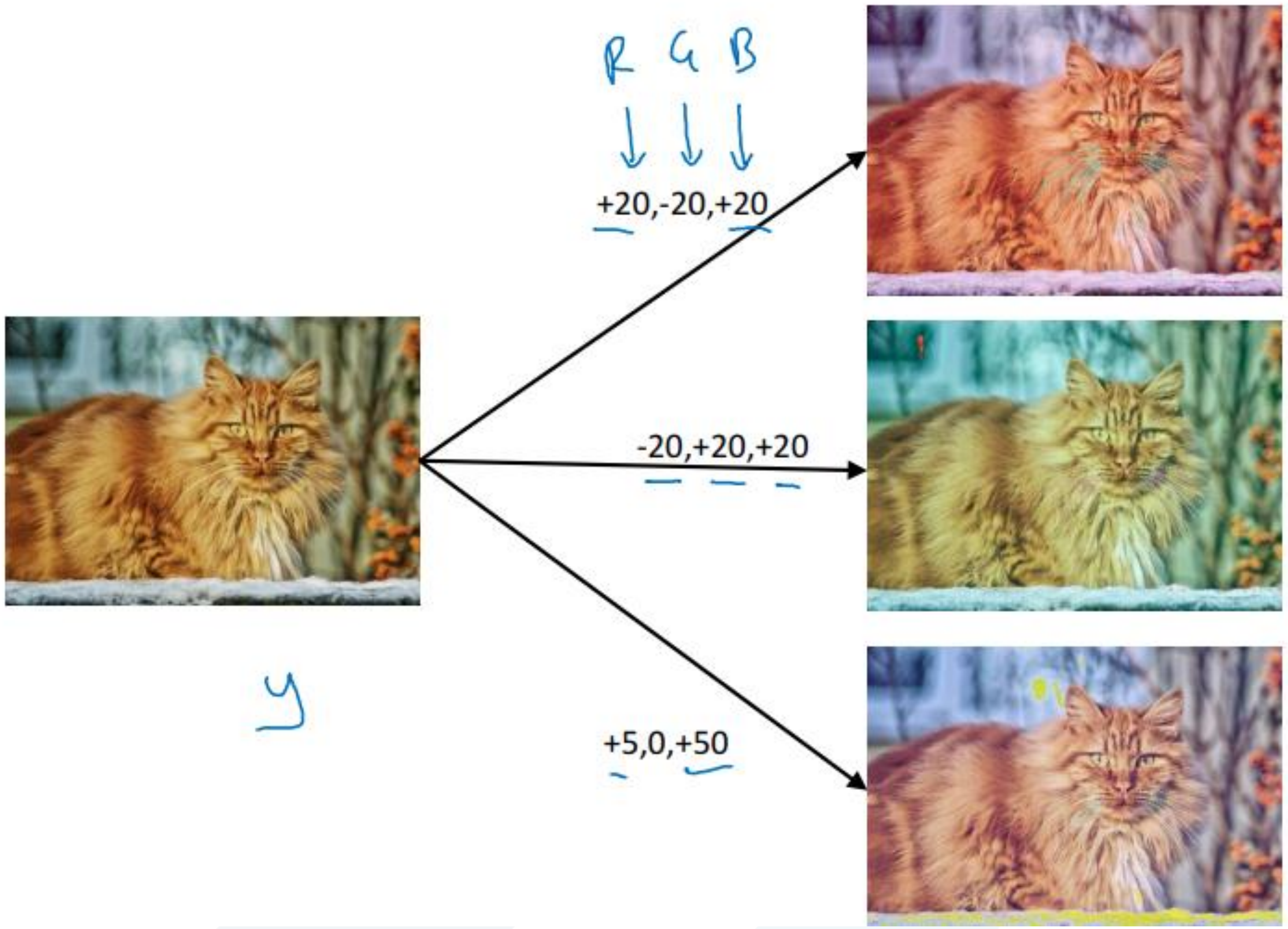


Random Cropping



- Rotation
- Shearing
- Local warping
- ...

Color shifting



Advanced:

PCA

ml-class.org

[AlexNet paper

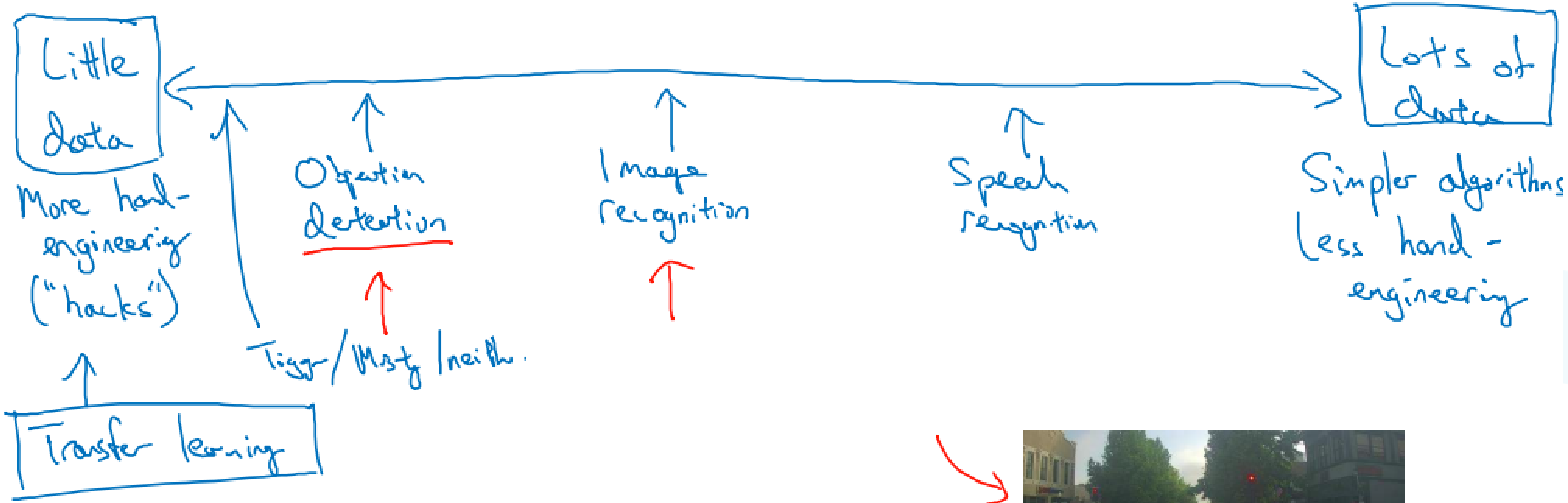
"PCA color augmentation."

RB G

Practical advice for using ConvNet

The state of computer vision

Data vs. hand-engineering



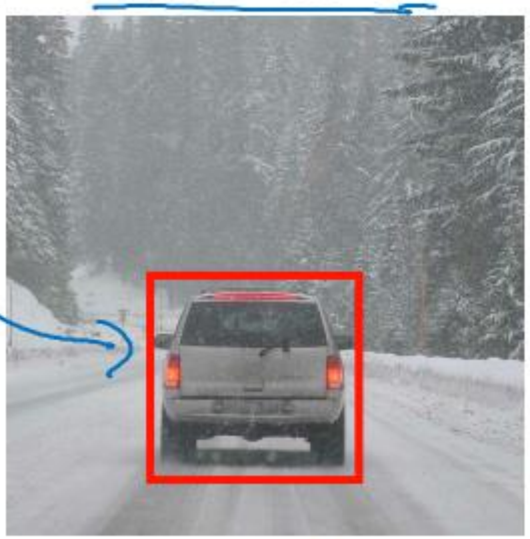
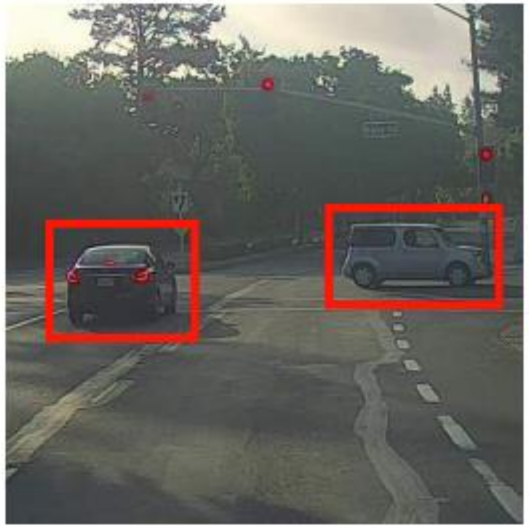
Two sources of knowledge

- Labeled data (x,y)
- Hand engineered features/network architecture/other components

Object Detection

Object localization

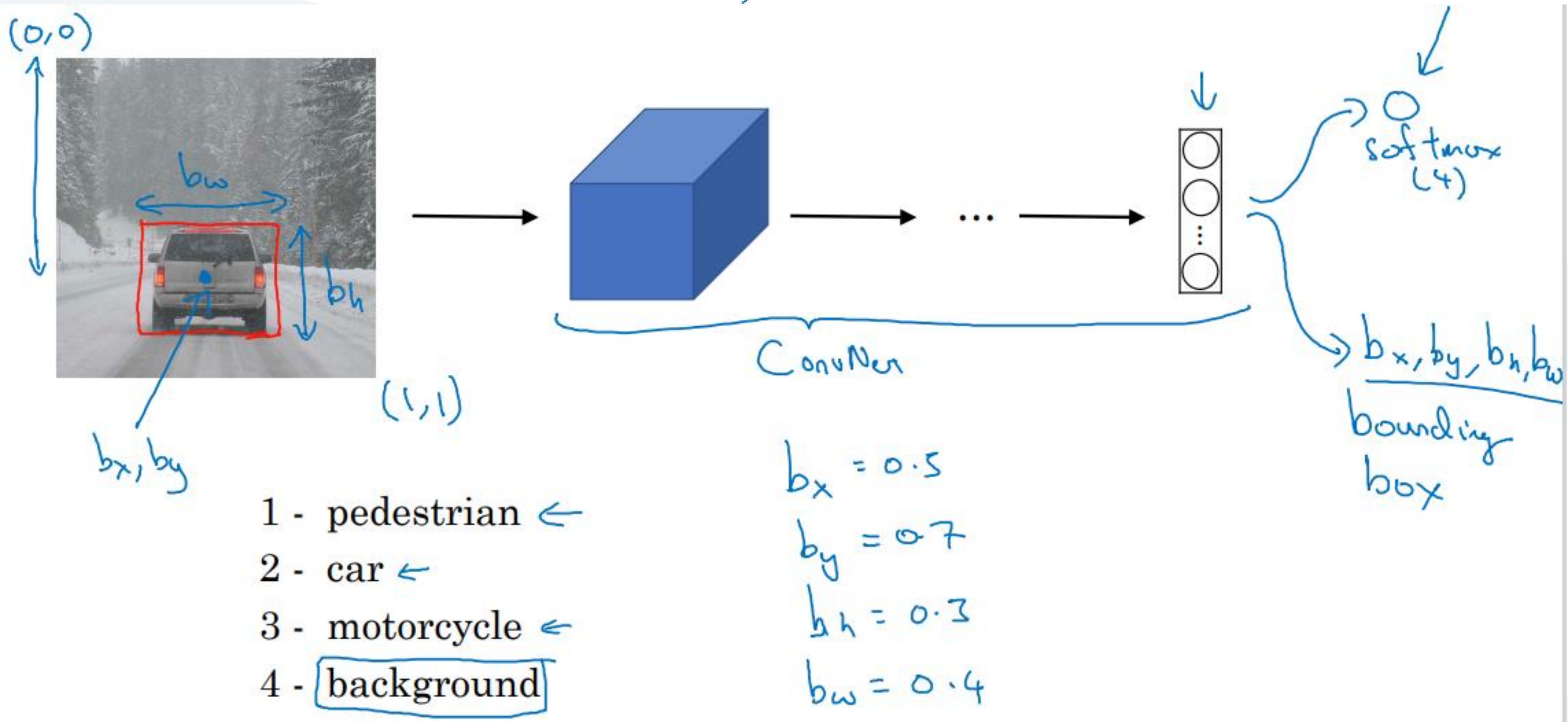
What are localization and detection

Image classification	Classification with localization	Detection
		
"Car"	"Car"	

1 object

multiple objects

Classification with localization



Defining the target label y

1 - pedestrian
2 - car
3 - motorcycle
4 - background

Need to output b_x, b_y, b_h, b_w, class label (1-4)

$$\mathcal{L}(\hat{y},y) = \begin{cases} (\hat{y}_1-y_1)^2 + (\hat{y}_2-y_2)^2 + \dots + (\hat{y}_8-y_8)^2 & \text{if } y_1 = 1 \\ (\hat{y}_1-y_1)^2 & \text{if } y_1 = 0 \end{cases}$$

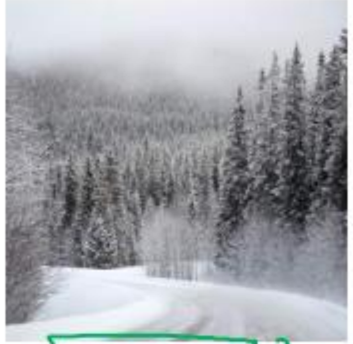
$$y = \begin{bmatrix} P_c \\ b_x \\ b_y \\ b_h \\ b_w \\ c_1 \\ c_2 \\ c_3 \end{bmatrix}$$

P_c: is there any object?

x =

$$\begin{bmatrix} 1 \\ b_x \\ b_y \\ b_h \\ b_w \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

(x, y)



$$\begin{bmatrix} 0 \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \end{bmatrix}$$

P_c

"don't care"