Copyright Notice

PROJECTS : BUILDING Residual Networks very deep networks. The problem with deep networks is vanishing gradients , but in RESnets we skip

deeplearning.ai

Case Studies

Why look at case studies?

Outline

Classic networks:

- LeNet-5 ←
- AlexNet ←
- VGG ←

ResNet (152)

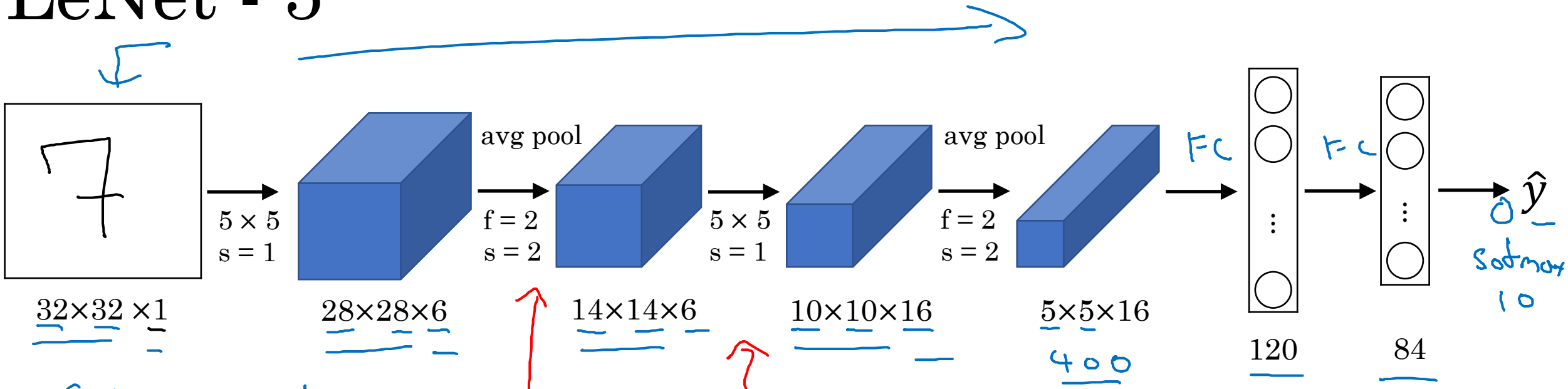
Inception

deeplearning.ai

Case Studies

Classic networks

LeNet - 5



60k parameters.

$n_H, n_w \downarrow$ $n_c \uparrow$

Non-linearity after pooling

$n_{H} \times n_{w} \times n_c$ $f \times f \times n_c$

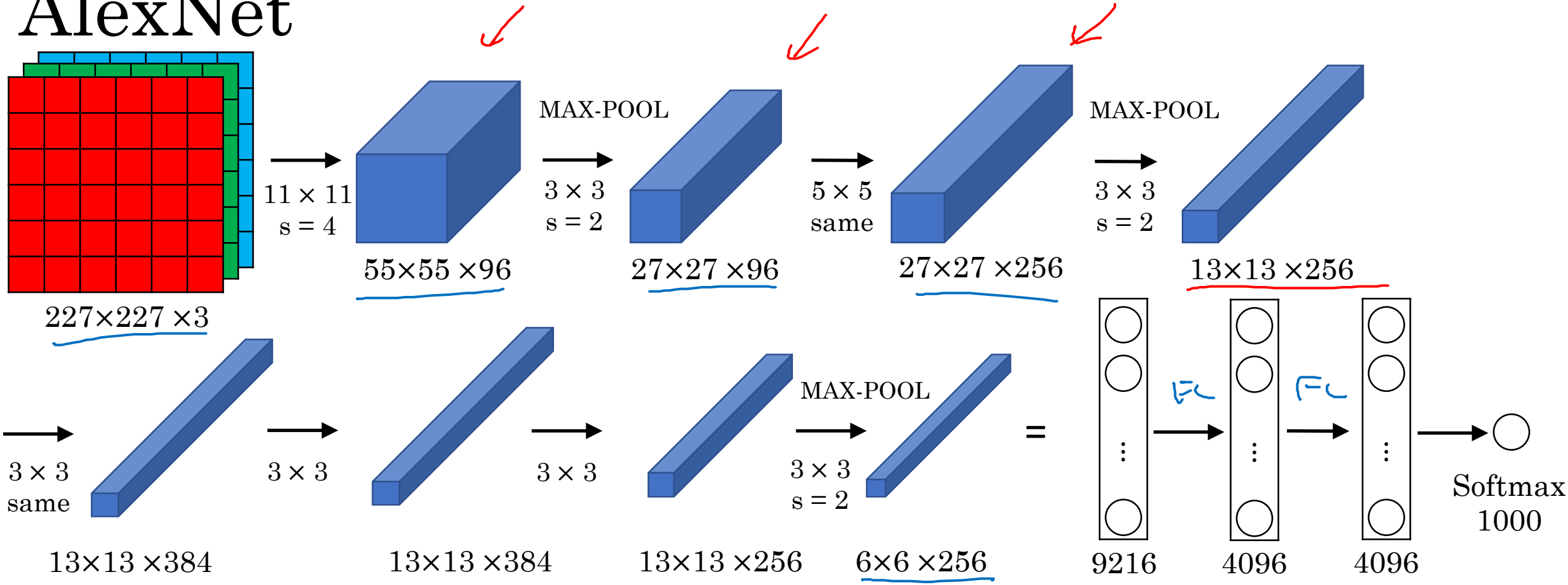
conv pool conv pool fc fc output

Activation: sigmoid/tanh ReLU

II III

[LeCun et al., 1998. Gradient-based learning applied to document recognition]

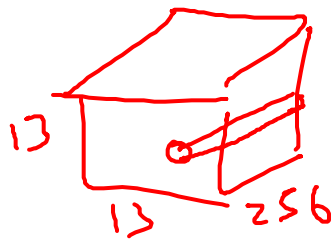
AlexNet



- Similar to LeNet, but much bigger.
- ReLU
- Multiple GPUs.
- Local Response Normalization (LRN)

9216

60M parameters

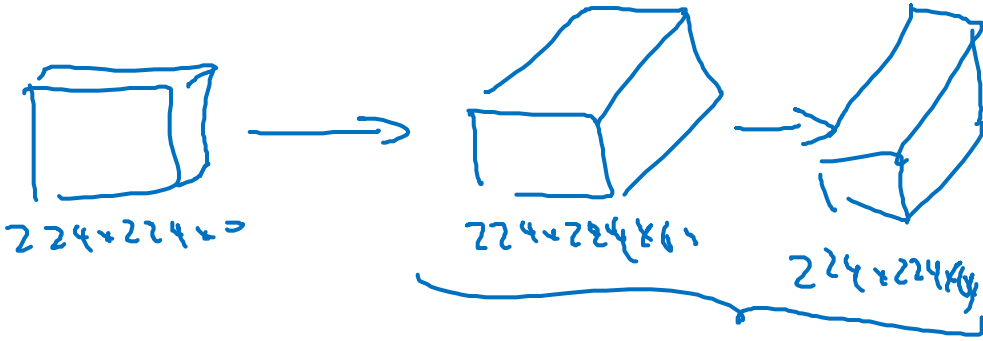


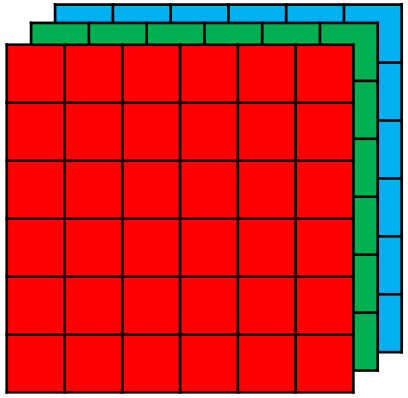
[Krizhevsky et al., 2012. ImageNet classification with deep convolutional neural networks]

VGG - 16

VGG-19

CONV = 3×3 filter, s = 1, same MAX-POOL = 2×2 , s = 2





224×224 ×3

→ [CONV 64] ×2 → 224×224×64 → POOL → 112×112 ×64 → [CONV 128] ×2 → 112×112 ×128 → POOL → 56×56 ×128

→ [CONV 256] ×3 → 56×56 ×256 → POOL → 28×28 ×256 → [CONV 512] ×3 → 28×28 ×512 → POOL → 14×14×512

→ [CONV 512] ×3 → 14×14 ×512 → POOL → 7×7×512 → FC 4096 → FC 4096 → Softmax 1000

$n_H, n_W \downarrow$ $n_c \uparrow$ ~138M

[Simonyan & Zisserman 2015. Very deep convolutional networks for large-scale image recognition]

Andrew Ng

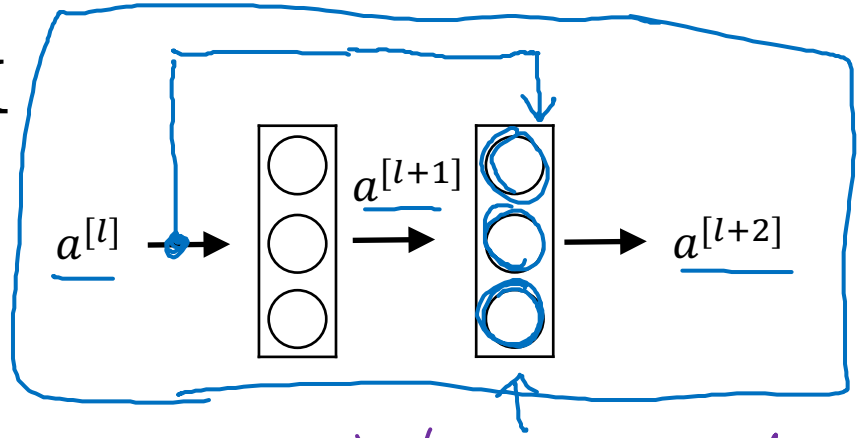
RESNETS allow us to train on much deeper network of over 100 layers. In classic network each information goes through all the layers , but in residua

deeplearning.ai

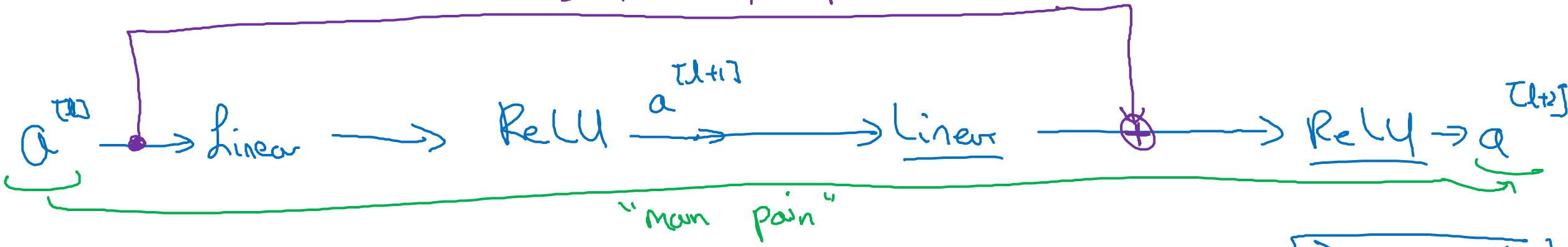
Case Studies

Residual Networks (ResNets)

Residual block



"short cut" / skip connection



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

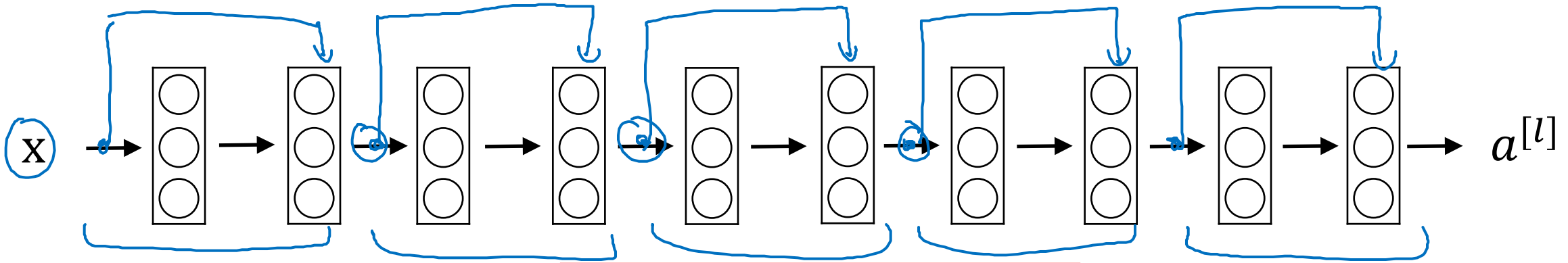
$$a^{[l+2]} = g(z^{[l+2]} + a^{[l]})$$

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

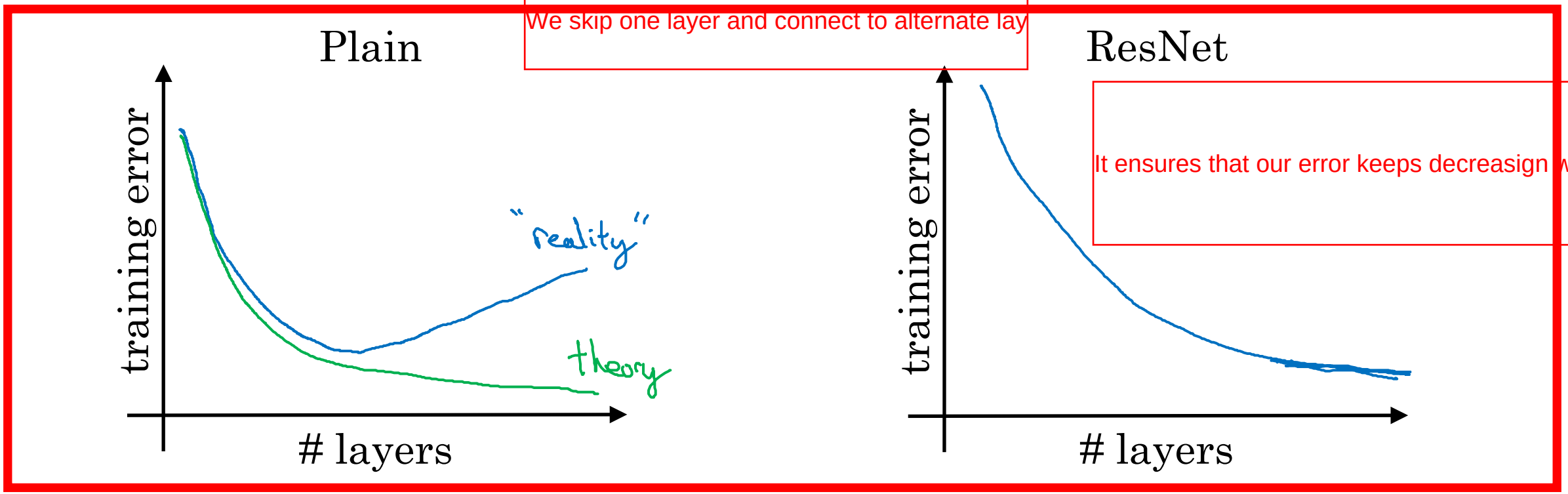
Andrew Ng

Residual Network

"Plain network"



We skip one layer and connect to alternate lay



It ensures that our error keeps decreasign with

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

Andrew Ng

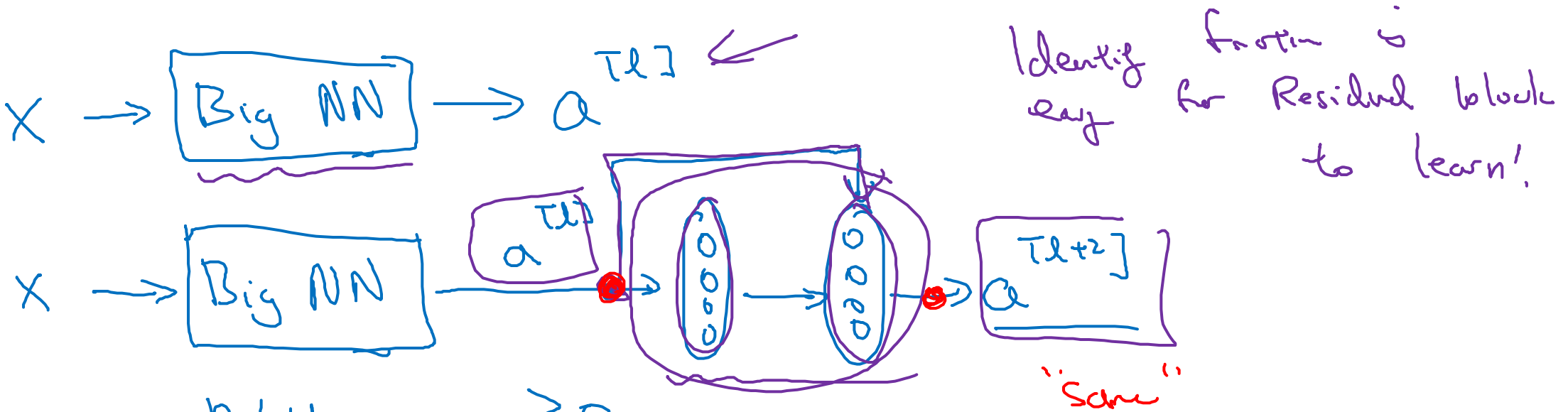
In theory increasing the depth iof NN should continuosly do more good on trianing set, but it's not the case in general. But thsi is true for RESNEtT .RE

deeplearning.ai

Case Studies

Why ResNets work

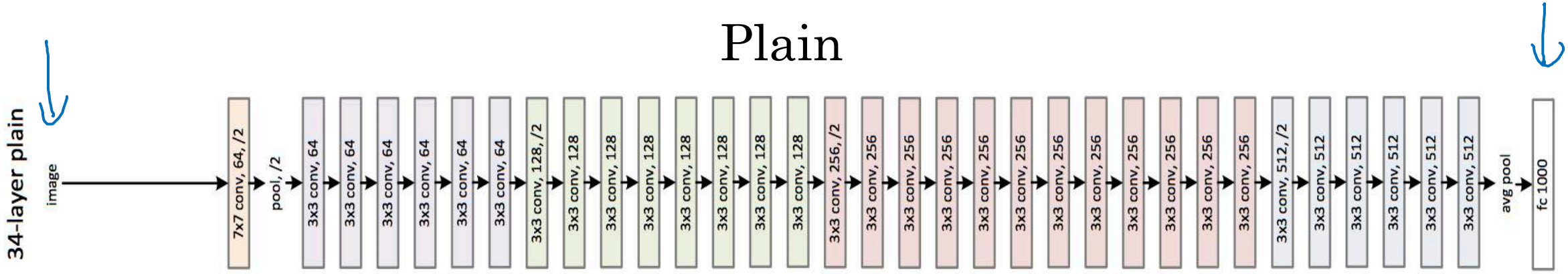
Why do residual networks work?



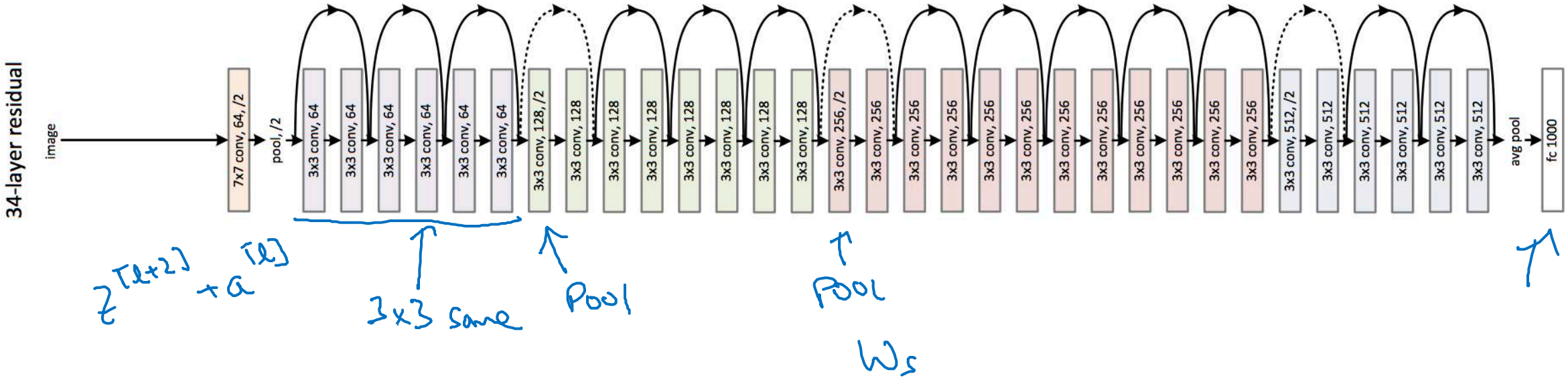
ResNet

dimension of a[l+2] should be same as that of a[l] so same convultion is generally performed. and If not happen we add a

Plain



ResNet



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

Andrew Ng

deeplearning.ai

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

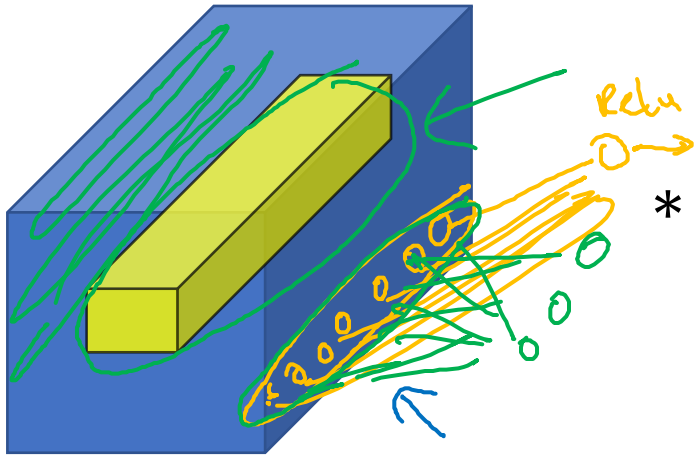
6 × 6 × 1

*

2

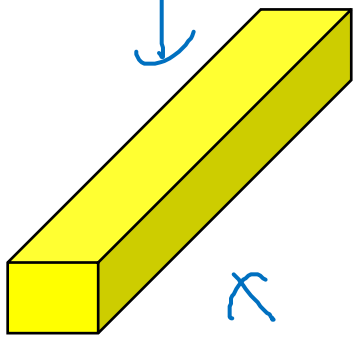
=

2	4	6	...		



6 × 6 × 32

*

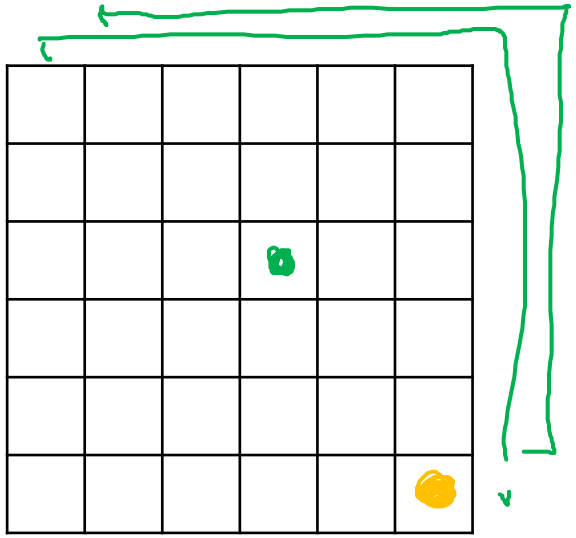


1 × 1 × 32

=

32 → # filters.

$n_c^{[l+1]}$



6 × 6 × # filters

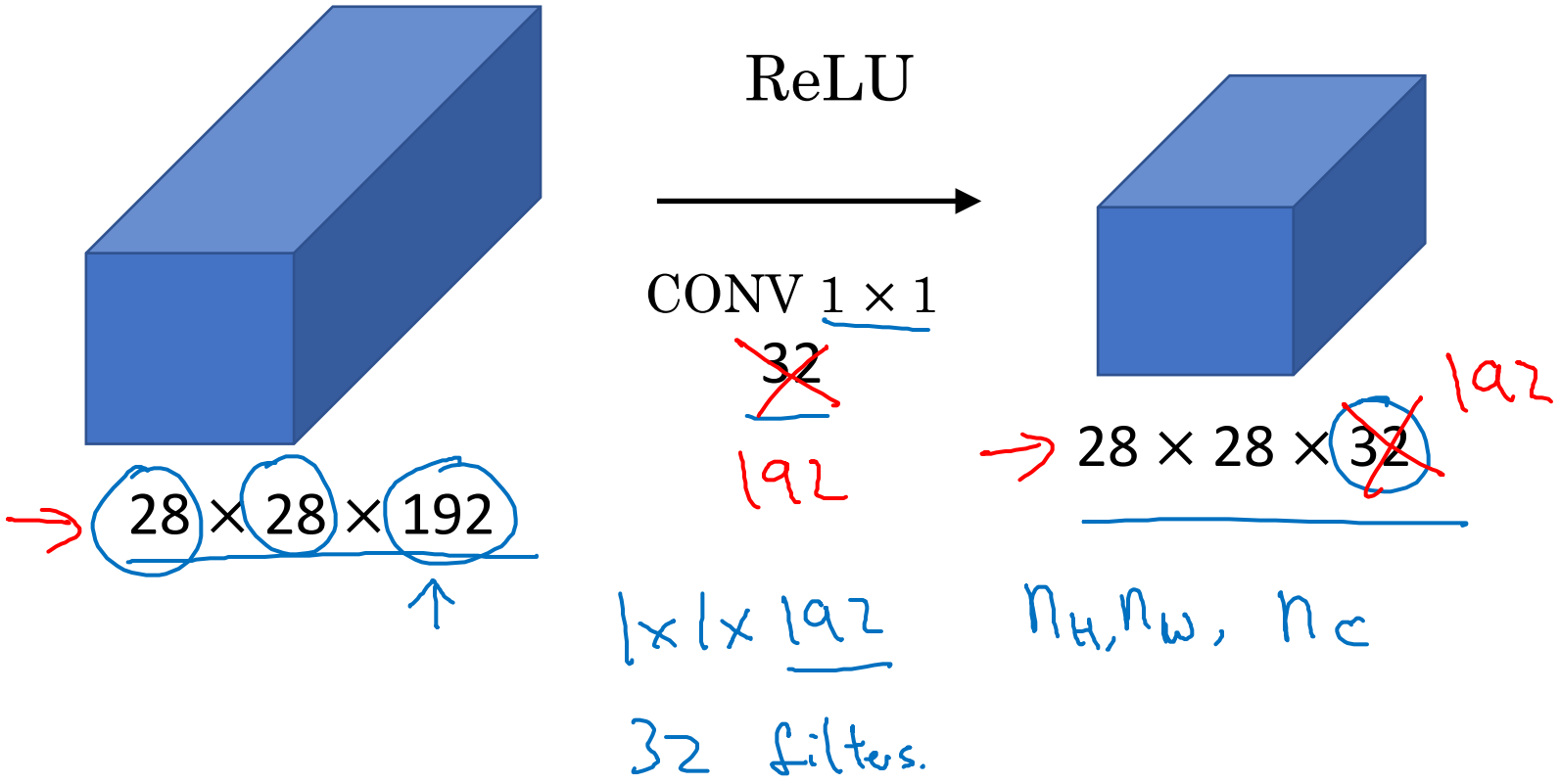
ReLU

Network in Network

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

Andrew Ng

Using 1×1 convolutions



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

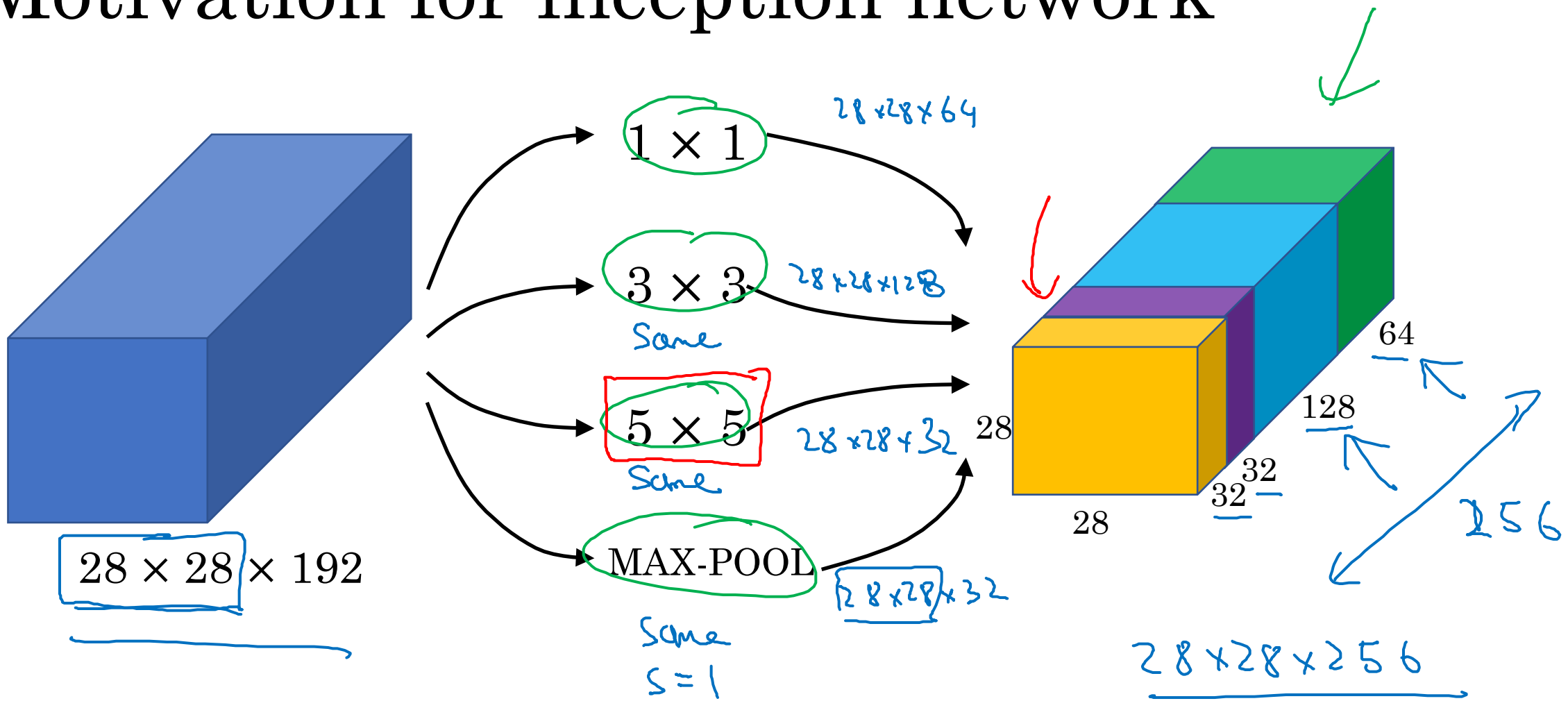
Bahot tarah ke filters use kar lena ek hi time pe like ki see neech diagram. Let the network learn what filters it need. But the problem here is computation

deeplearning.ai

Case Studies

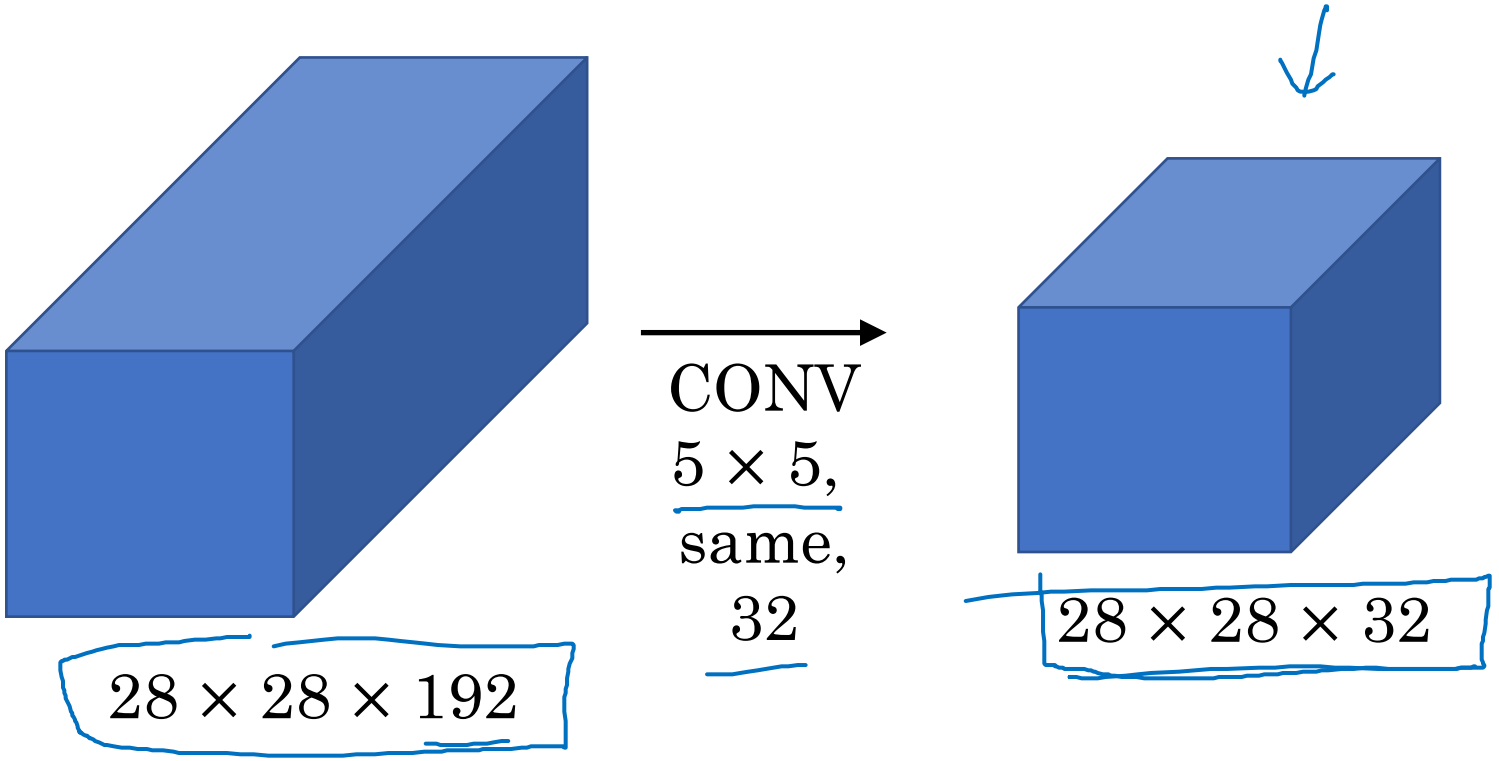
Inception network motivation

Motivation for inception network



[Szegedy et al. 2014. Going deeper with convolutions]

The problem of computational cost



32 filters. filter one 5×5×192.

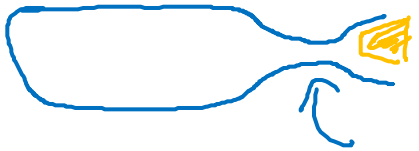
28×28×32 × 5×5×192 = 120M.

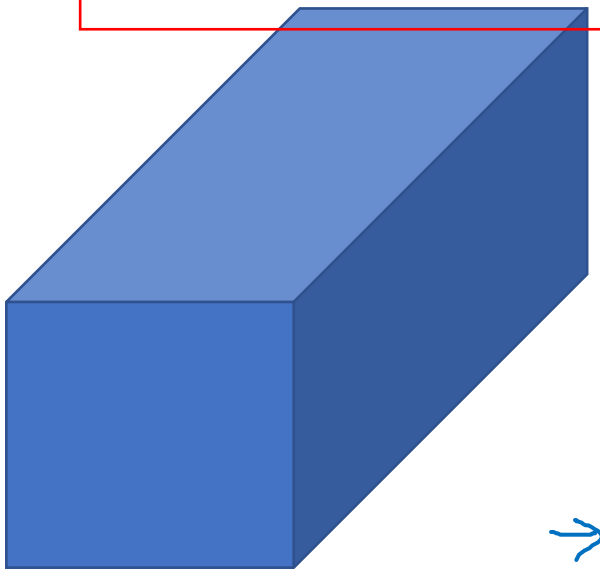
No of computation.

Using 1×1 convolution

The same above artitecture can be achieved through 1x1 filter and with much

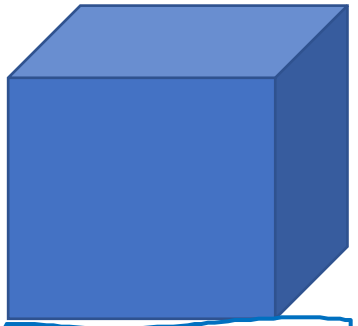
"bottleneck layer"





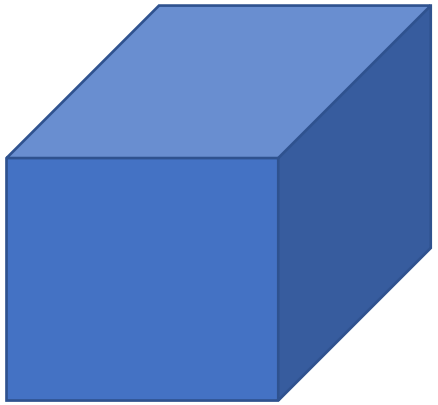
28 × 28 × 192

CONV
1 × 1,
16,
1 × 1 × 192



28 × 28 × 16

CONV
5 × 5,
32,
5 × 5 × 16



28 × 28 × 32

28×28×16 × 192 = 2.4M

28×28×32 × 5×5×16 = 10.0M

12.4M

120M

Andrew Ng

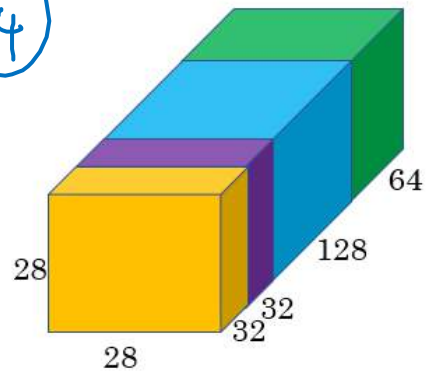
deeplearning.ai

Case Studies

Inception network

Velow deagram representing how we can sue inception in the daigram examle taeken above sli

Inception module



Previous Activation

28 × 28 × 192

1 × 1 CONV

28 × 28 × 64

1 × 1 CONV

96

3 × 3 CONV

28 × 28 × 128

1 × 1 CONV

16

5 × 5 CONV

28 × 28 × 32

MAXPOOL 3 × 3, s = 1 same

28 × 28 × 192

1 × 1 CONV

28 × 28 × 32

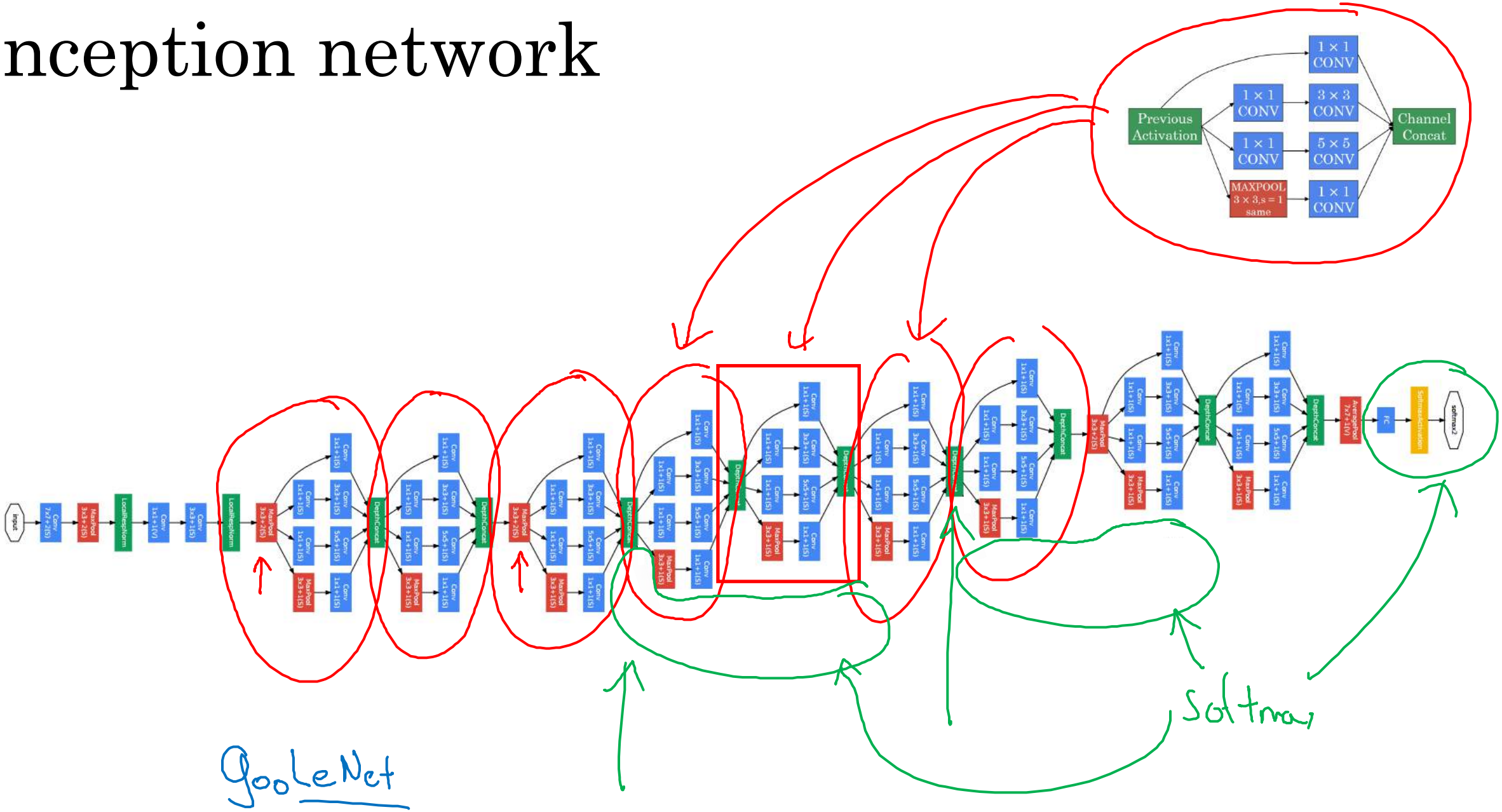
32 filters, 1 × 1 × 192.

Channel Concat

28 × 28 × 256

Andrew Ng

Inception network



[Szegedy et al., 2014, Going Deeper with Convolutions]

http://knowyourmeme.com/memes/we-need-to-go-deeper

Convolutional Neural Networks

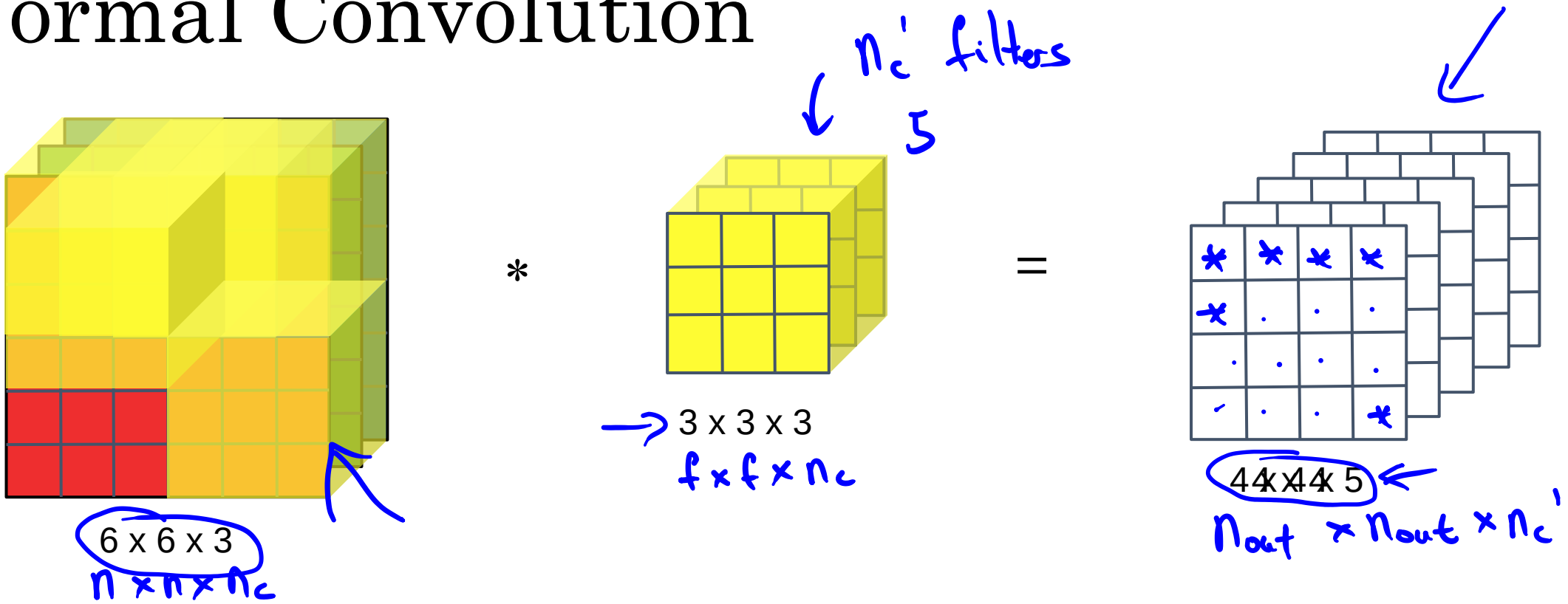
MobileNet

Motivation for MobileNets

- Low computational cost at deployment
- Useful for mobile and embedded vision applications
- Key idea: Normal vs. depthwise-separable convolutions

[Howard et al. 2017, MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications]

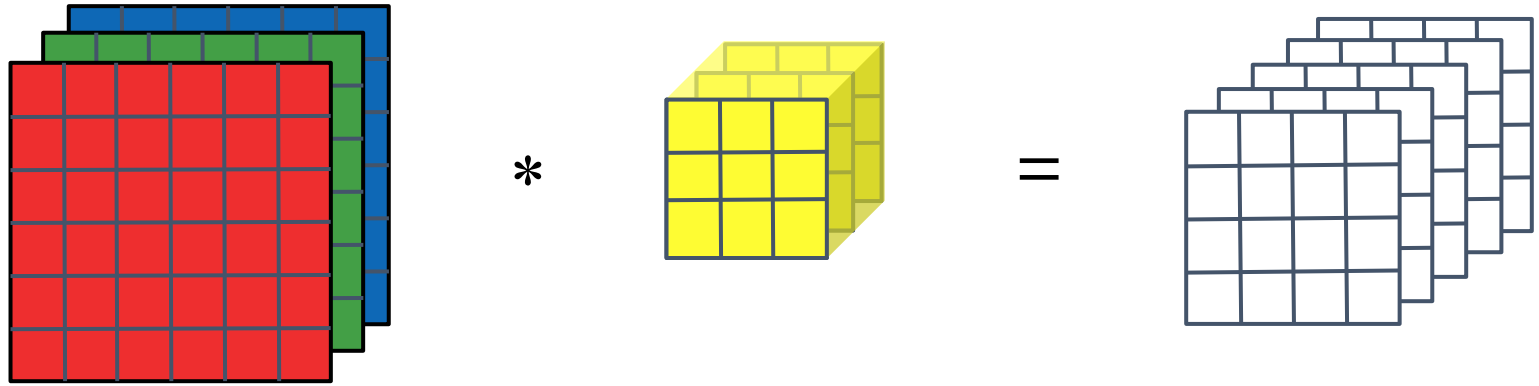
Normal Convolution



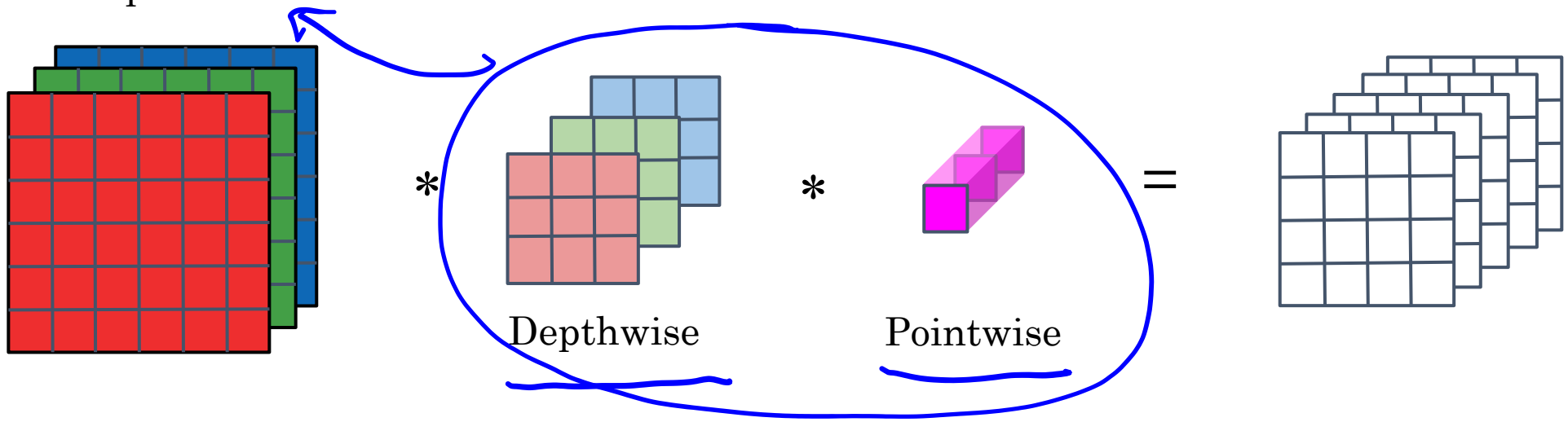
Computational cost	=	#filter params	x	# filter positions	x	# of filters
2160	=	3x3x3	x	4x4	x	5

Depthwise Separable Convolution

Normal Convolution



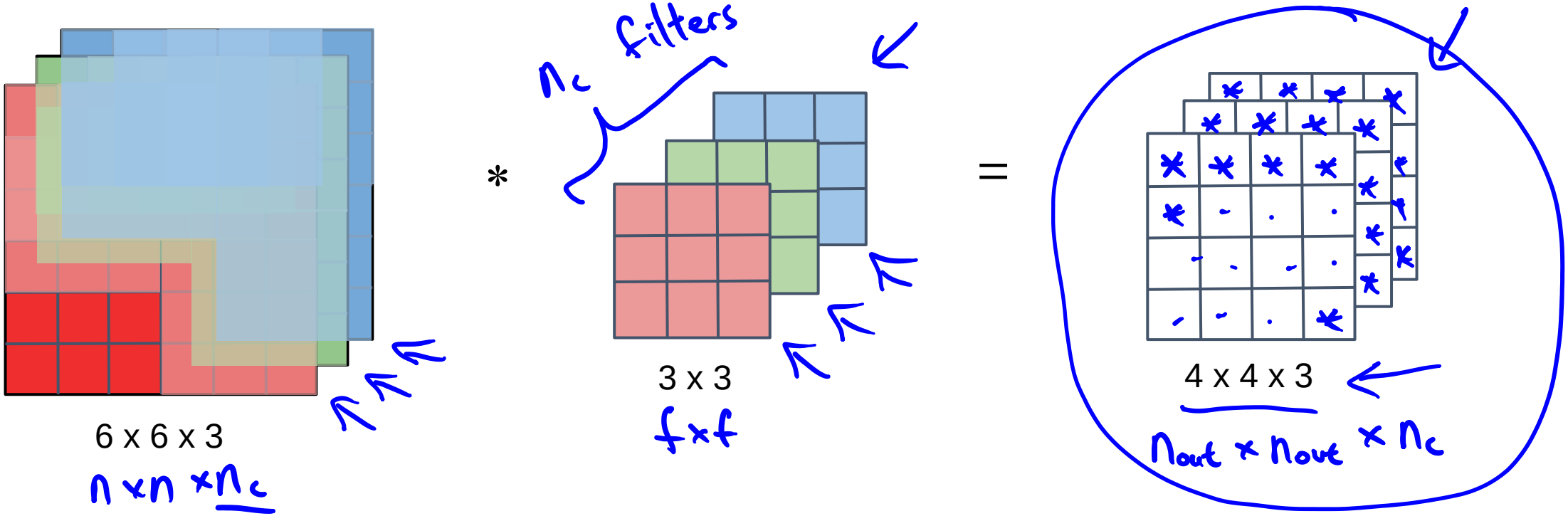
Depthwise Separable Convolution



Andrew Ng

Here we do channel wise convultion like in 2D, red convoves with 1st layer, green with second adnd blue with thiresd and so an and they give

Depthwise Convolution



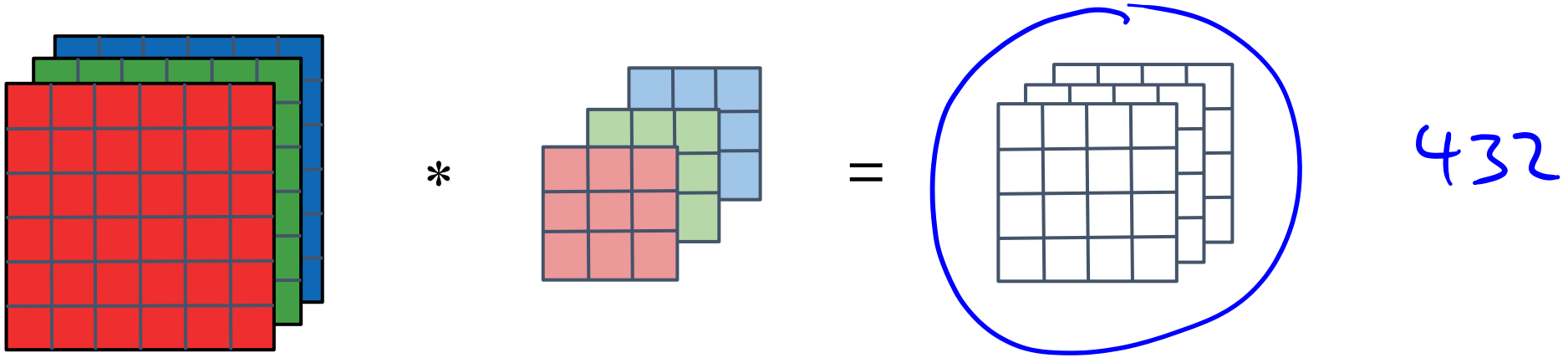
Computational cost = #filter params x # filter positions x # of filters

$432 = 3 \times 3 \times 4 \times 4 \times 3$

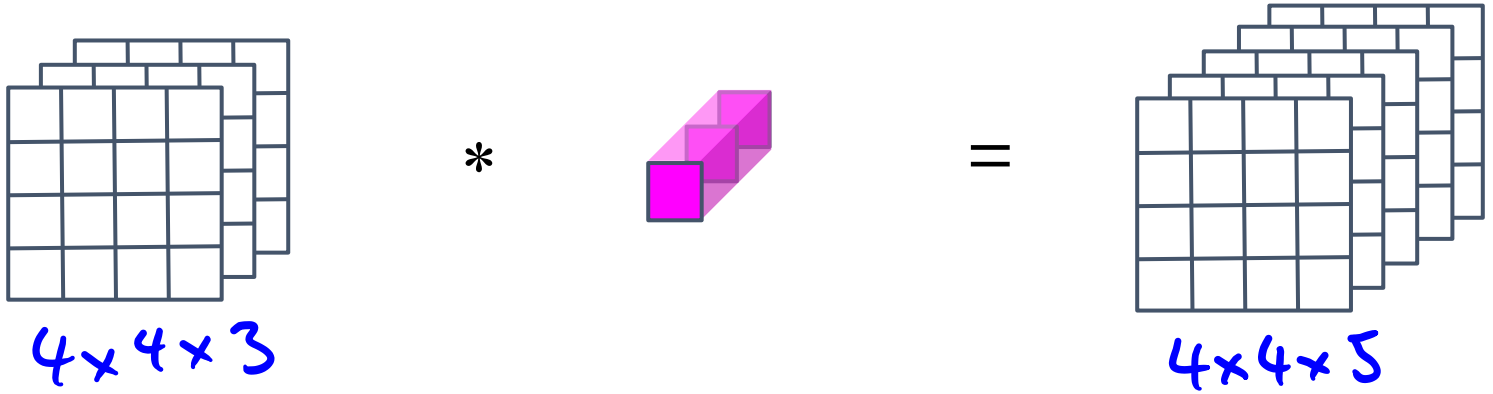
Andrew Ng

Depthwise Separable Convolution

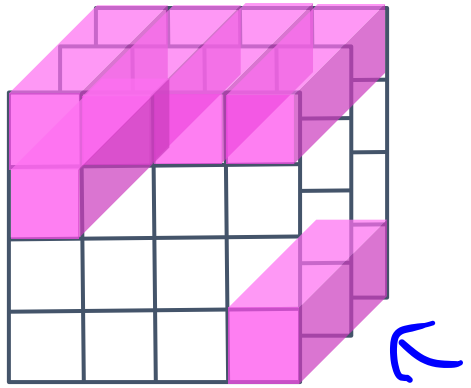
Depthwise Convolution



Pointwise Convolution



Pointwise Convolution



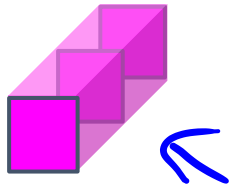
4 x 4 x 3

$n_{out} \times n_{out} \times n_c$

*

n_c' filters

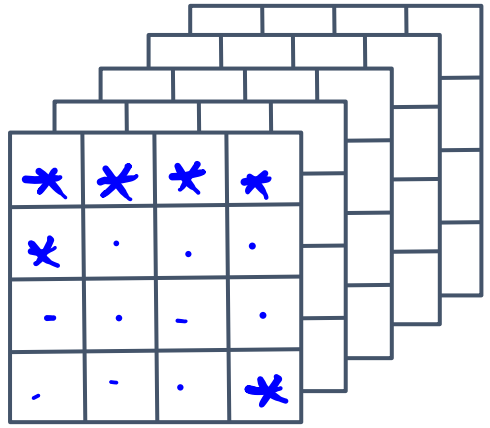
5



1 x 1 x 3

$1 \times 1 \times n_c$

=



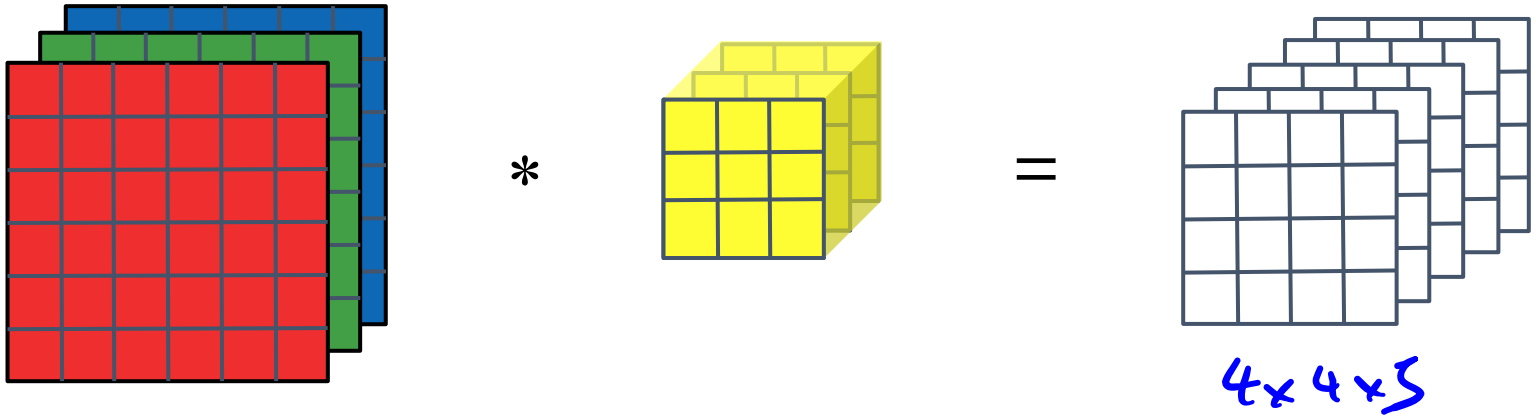
4 x 4 x 5

$n_{out} \times n_{out} \times n_c'$

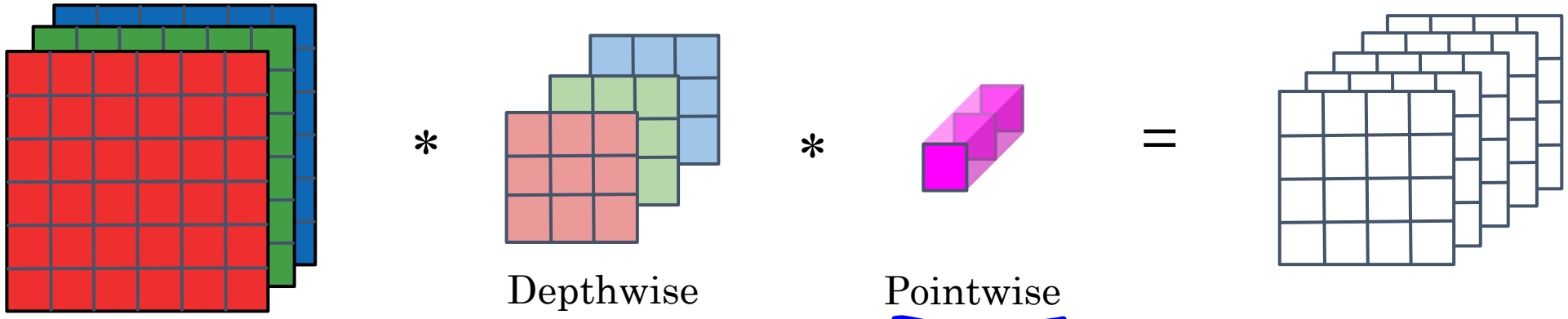
Computational cost	=	#filter params	x	# filter positions	x	# of filters
240	=	$1 \times 1 \times 3$	x	4×4	x	5

Depthwise Separable Convolution

Normal Convolution



Depthwise Separable Convolution



Andrew Ng

Cost Summary

Cost of normal convolution 2160

Cost of depthwise separable convolution

depthwise + pointwise

432 + 240 = 672

$$\frac{672}{2160} = 0.31$$

$$= \frac{1}{n_c} + \frac{1}{f^2}$$

$$\frac{1}{5} + \frac{1}{9}$$

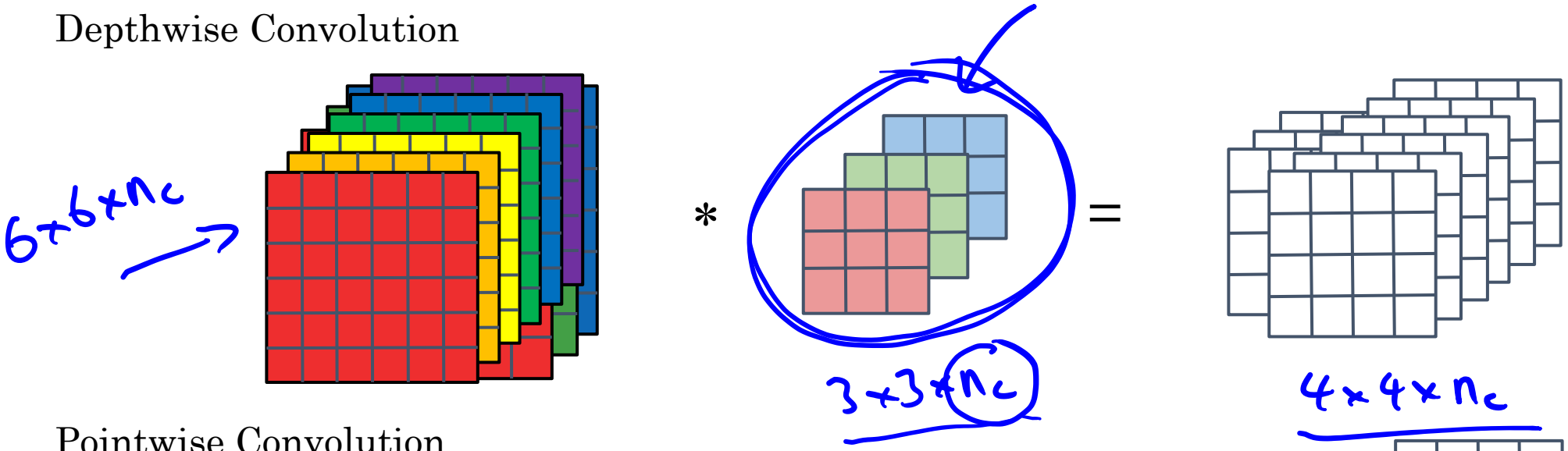
$$= \frac{1}{512} + \frac{1}{3^2}$$

$\sim$10 times cheaper

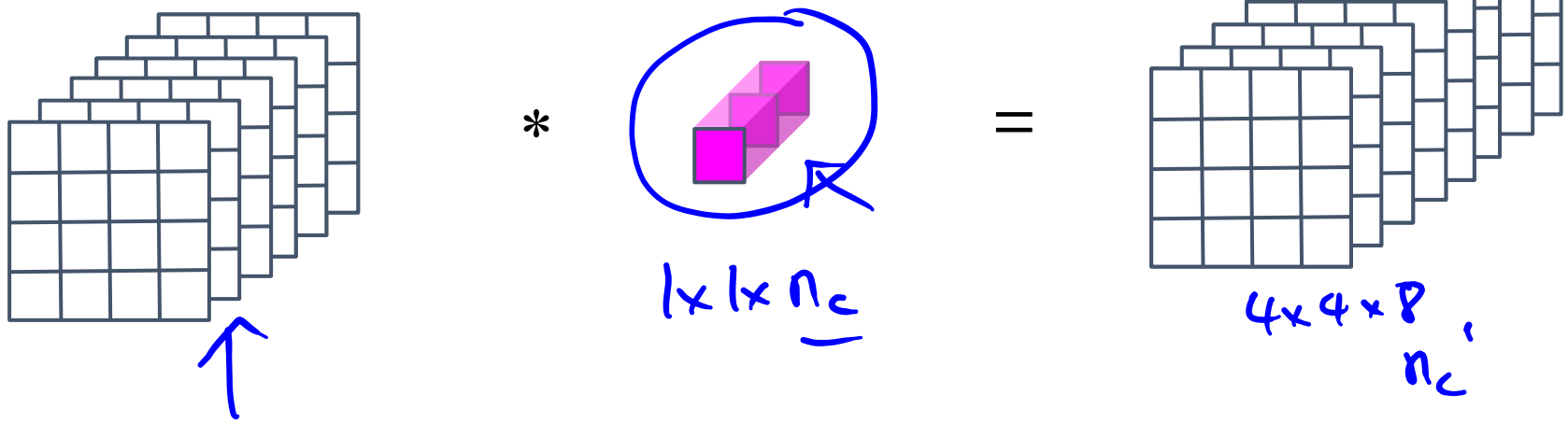
[Howard et al. 2017, MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications]

Depthwise Separable Convolution

Depthwise Convolution



Pointwise Convolution



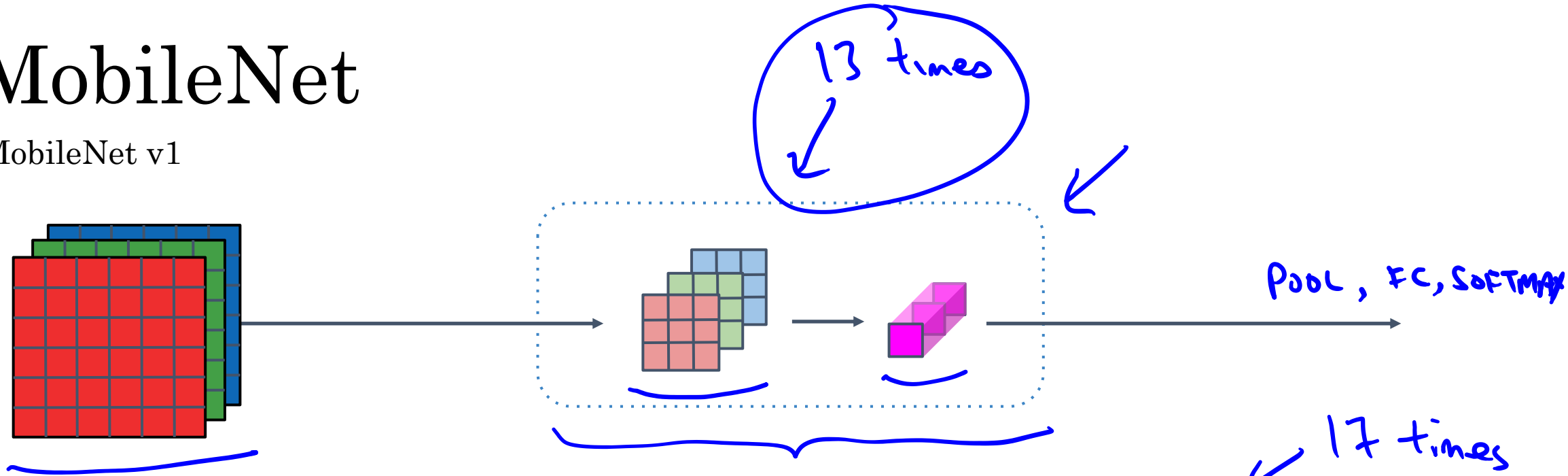
Andrew Ng

Convolutional Neural Networks

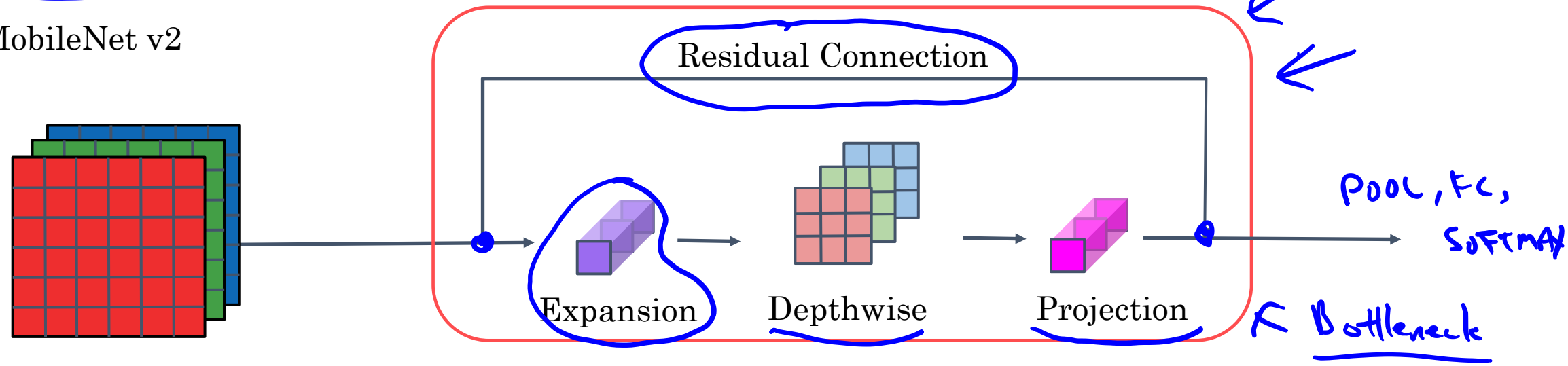
MobileNet Architecture

MobileNet

MobileNet v1

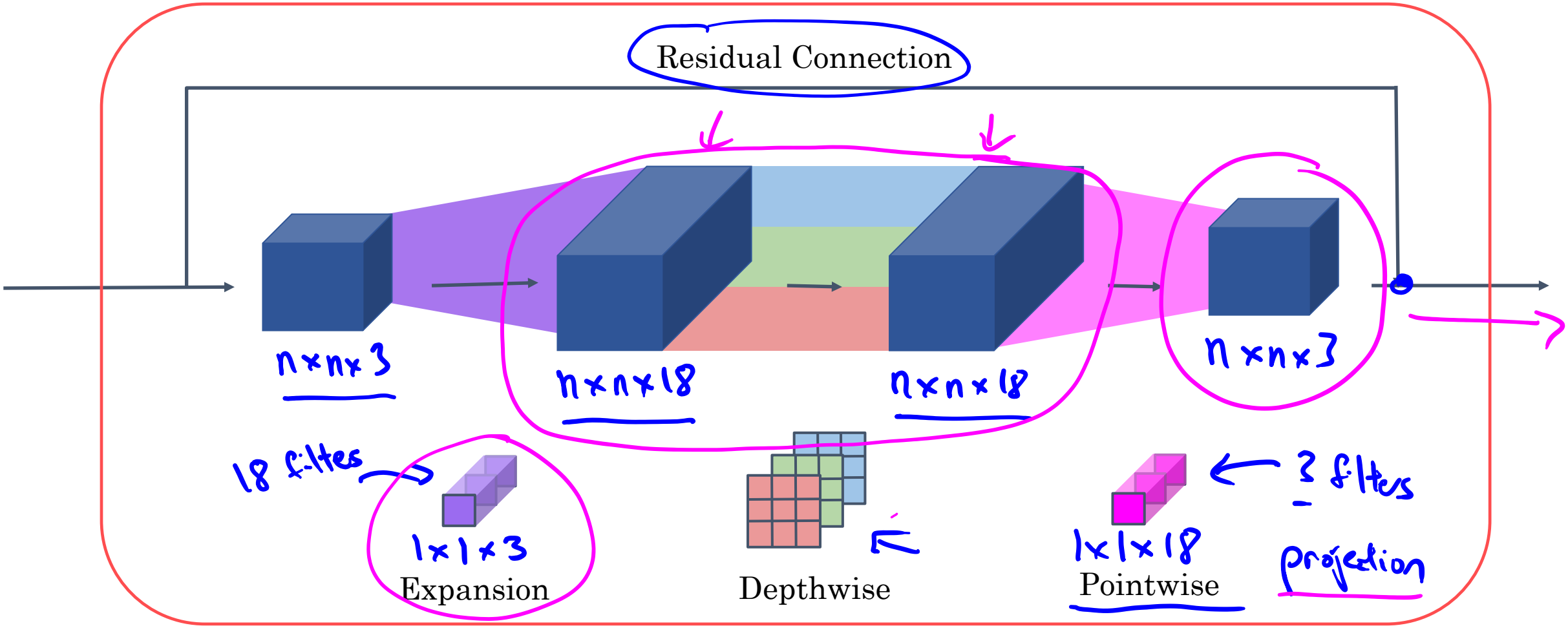


MobileNet v2



[Sandler et al. 2019, MobileNetV2: Inverted Residuals and Linear Bottlenecks]

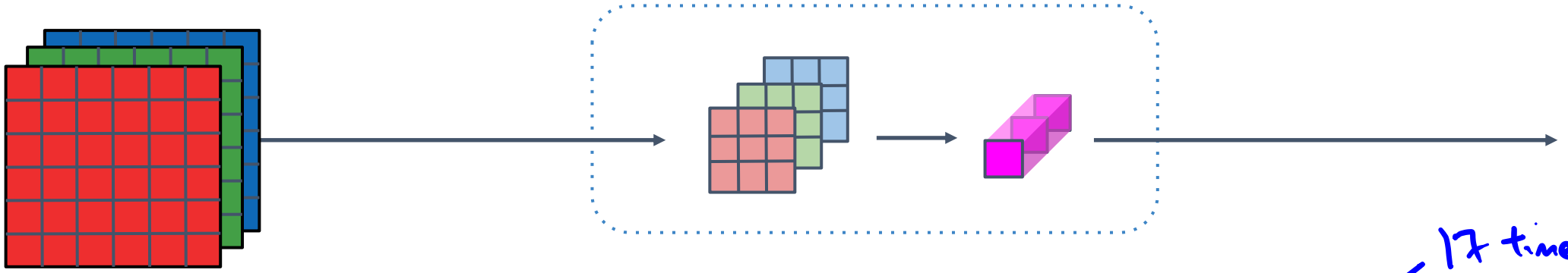
MobileNet v2 Bottleneck



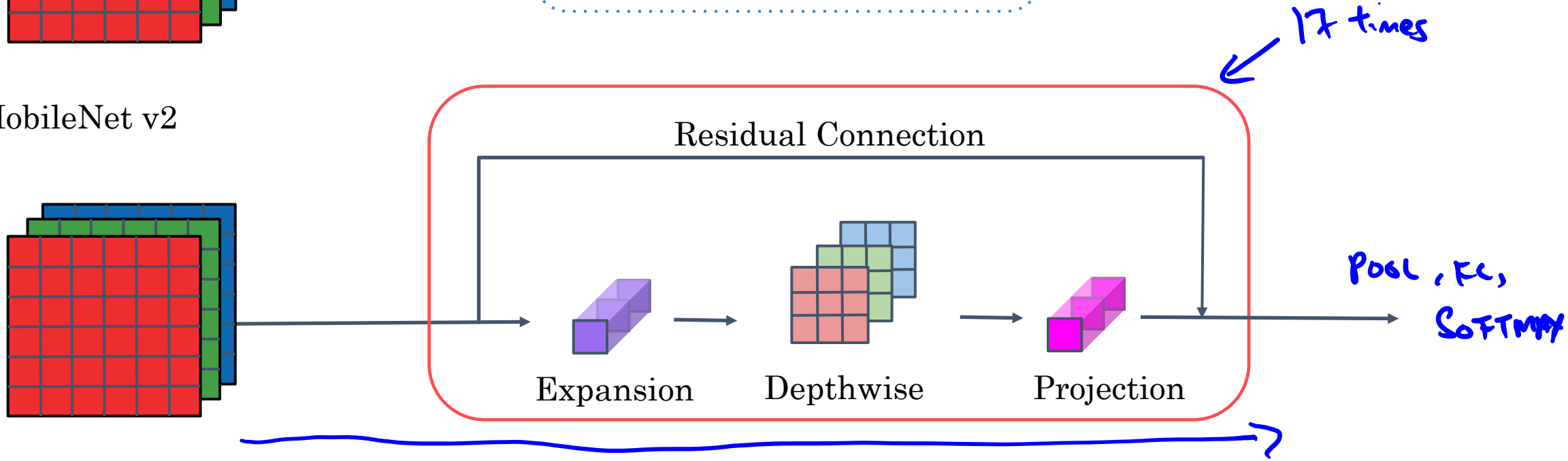
[Sandler et al. 2019, MobileNetV2: Inverted Residuals and Linear Bottlenecks]

MobileNet

MobileNet v1

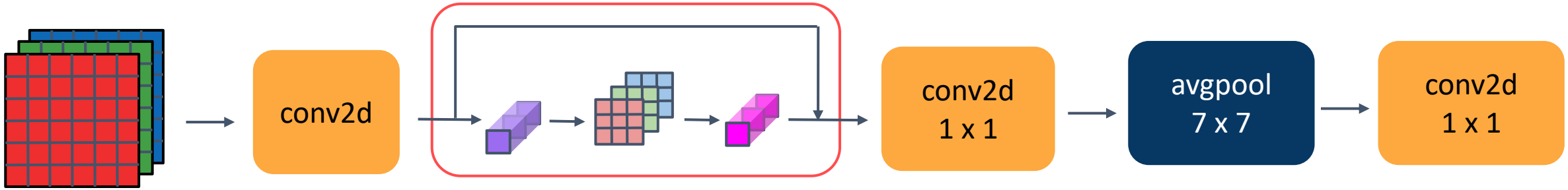


MobileNet v2



[Sandler et al. 2019, MobileNetV2: Inverted Residuals and Linear Bottlenecks]

MobileNet v2 Full Architecture



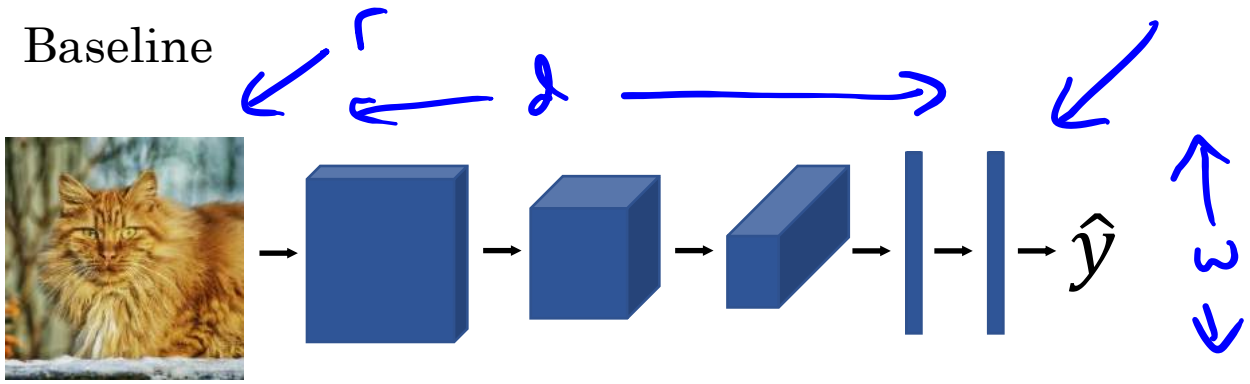
[Sandler et al. 2019, MobileNetV2: Inverted Residuals and Linear Bottlenecks]

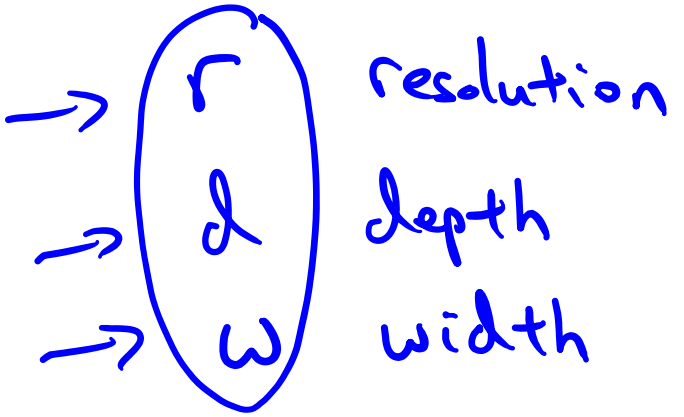
Convolutional Neural Networks

EfficientNet

EfficientNet

Baseline



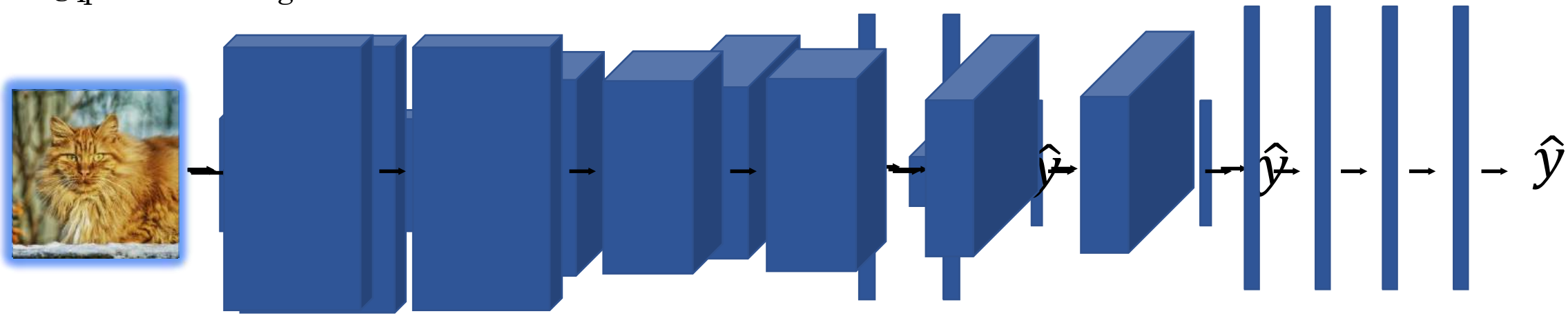


Bigger Resolution

Deeper

Wider

Compound scaling



[Tan and Le, 2019, EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks]

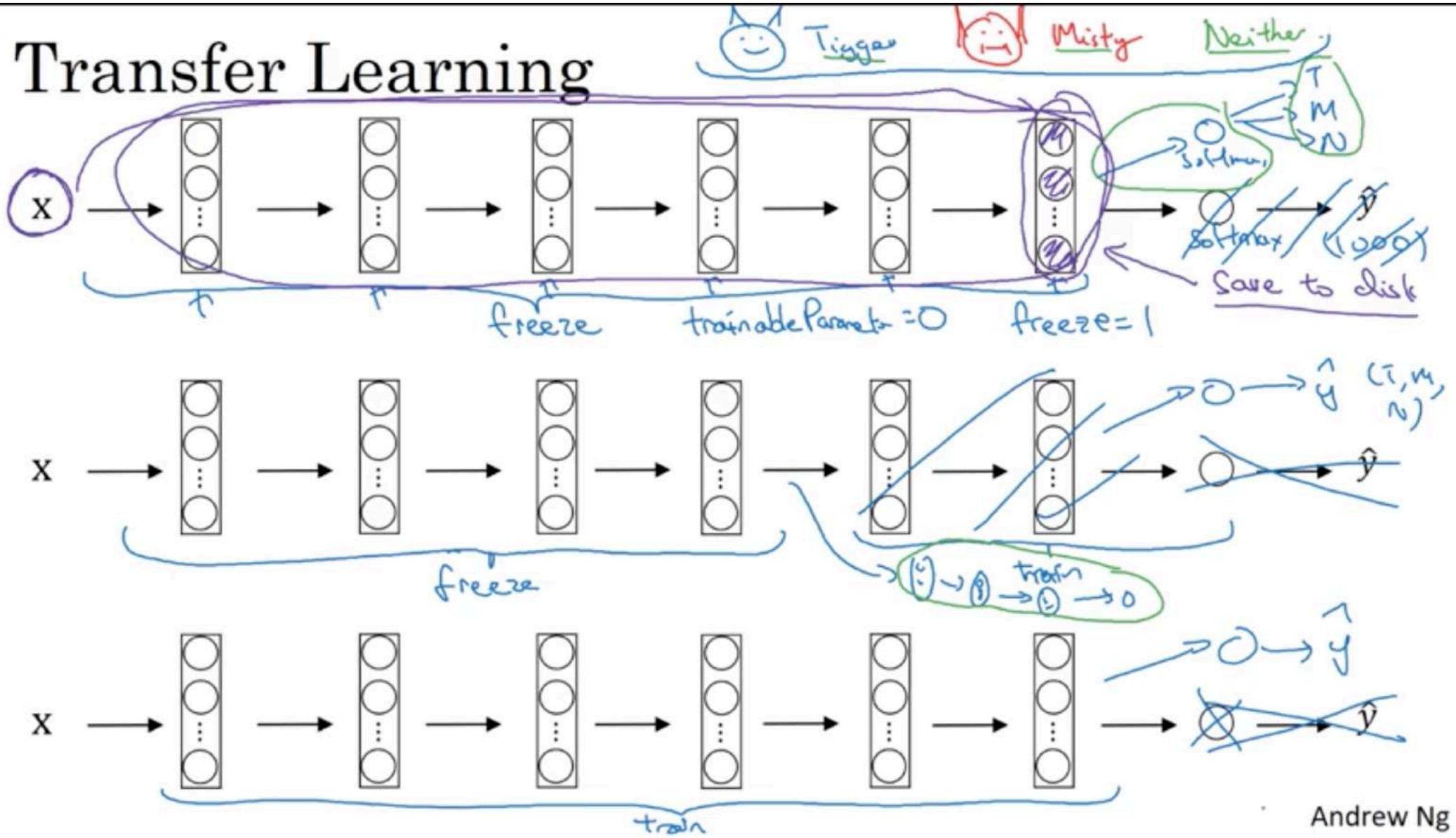
deeplearning.ai

Practical advice for using ConvNets

Transfer Learning

Instead of randomly intialising wts and building things from scratch , we can use wts that have already

Transfer Learning



Andrew Ng

deeplearning.ai

Practical advice for using ConvNets

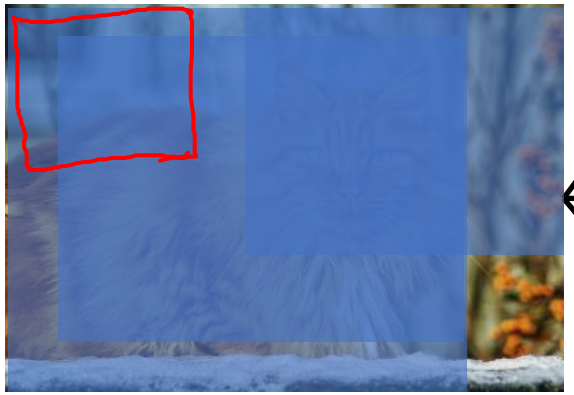
Data augmentation

Common augmentation method

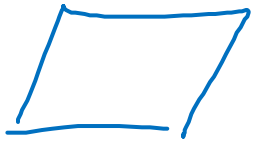
Mirroring

y

Random Cropping



- Rotation
- Shearing
- Local warping
- ...



Color shifting

R G B

+20,-20,+20

-20,+20,+20

+5,0,+50

Advanced:

PCA

ml-class.org

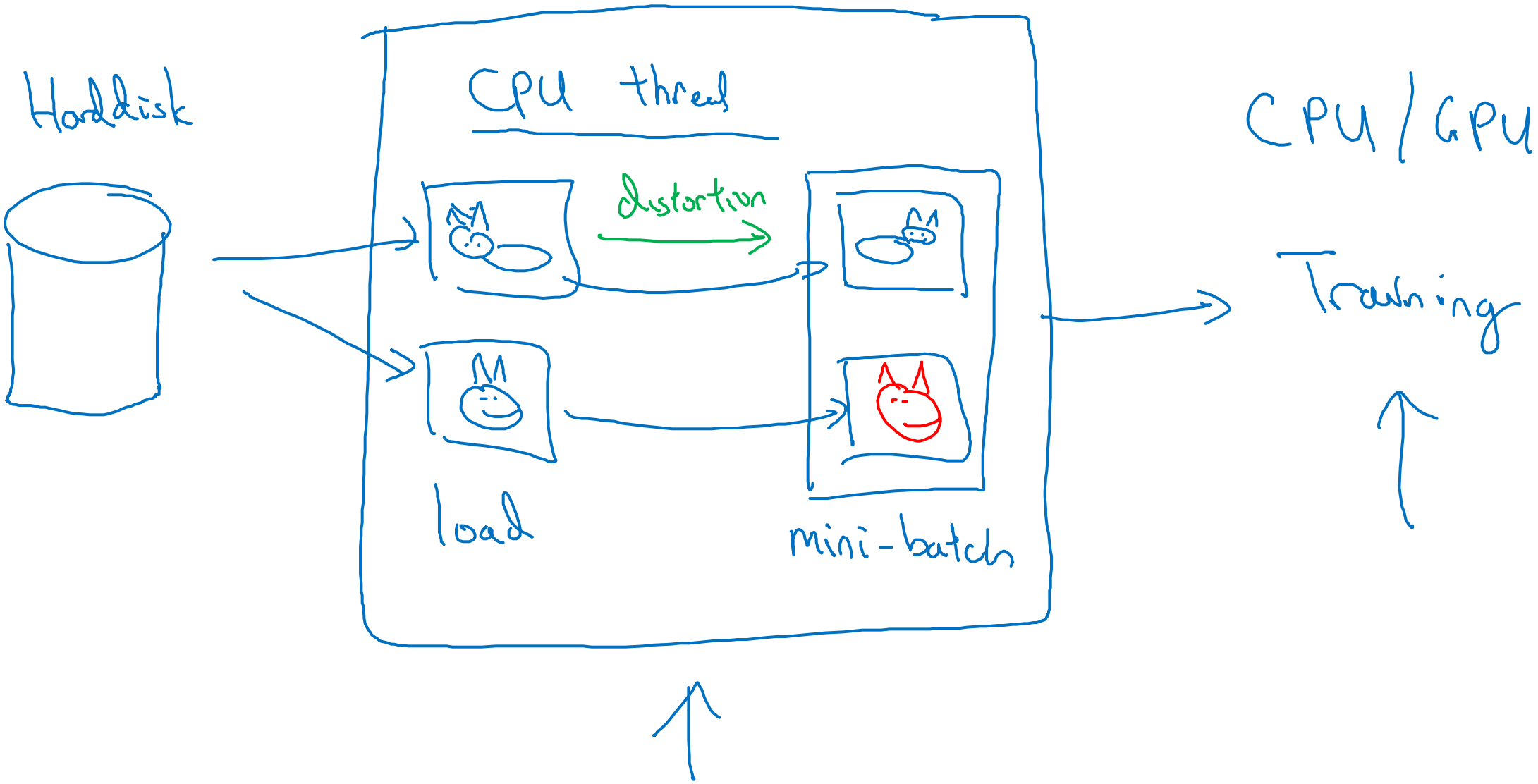
[AlexNet paper

["PCA color augmentation."

R B G

Andrew Ng

Implementing distortions during training

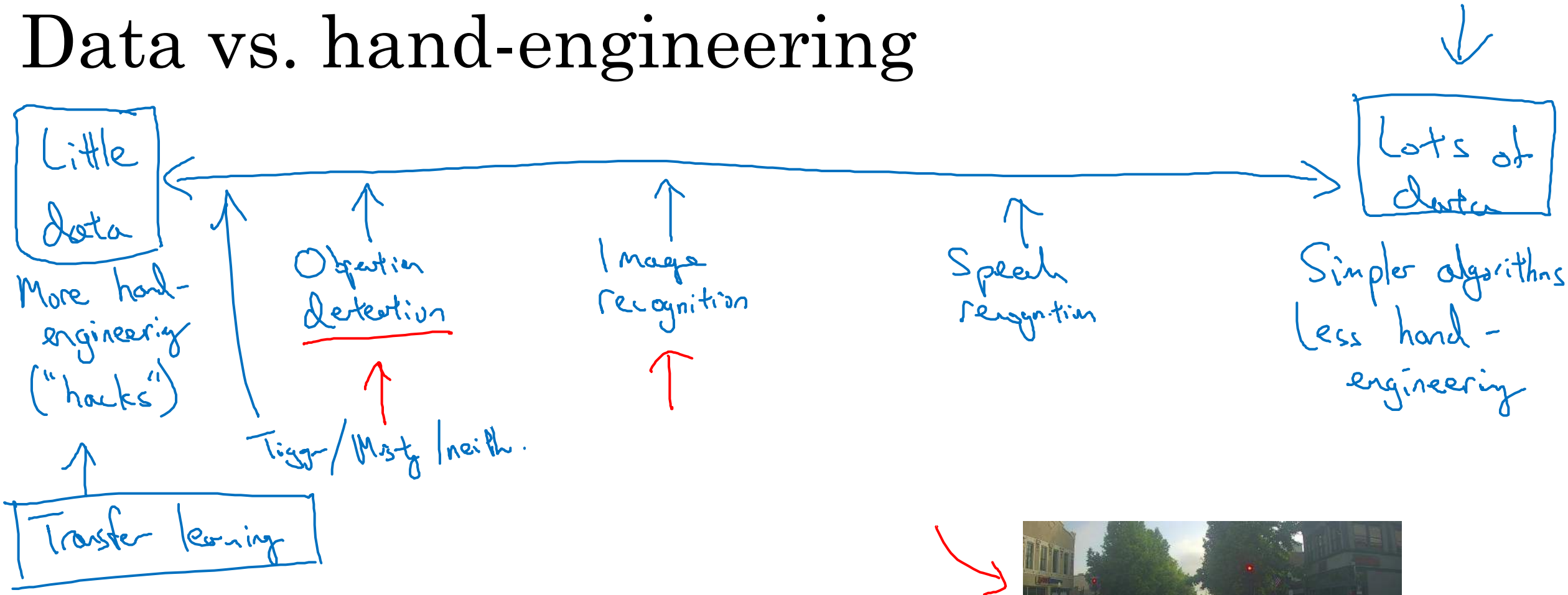


deeplearning.ai

Practical advice for using ConvNets

The state of computer vision

Data vs. hand-engineering



Two sources of knowledge

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

Andrew Ng

Tips for doing well on benchmarks/winning competitions

Ensembling

3-15 networks

- Train several networks independently and average their outputs

Multi-crop at test time

- Run classifier on multiple versions of test images and average results

10-crop

Use open source code

- Use architectures of networks published in the literature
- Use open source implementations if possible
- Use pretrained models and fine-tune on your dataset

Andrew Ng