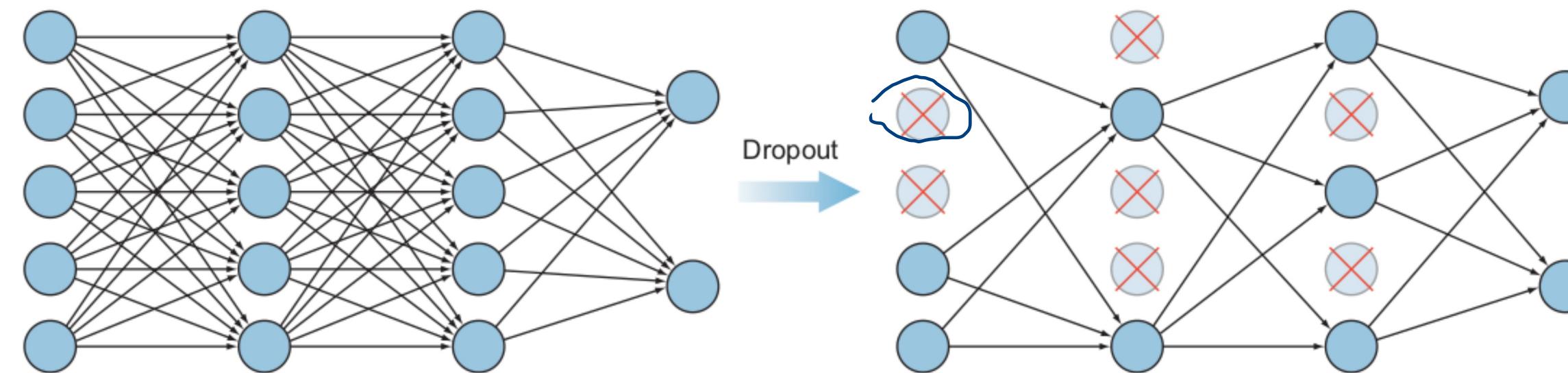


$$\underline{D_o} \rightarrow (\cancel{D_o}) \quad 20\% : \frac{1}{5}$$

لایه مهم دیگر (Drop Out) :

خاموش کردن درصدی از نورون ها



Ref[4]

$\underline{D_o}$ \swarrow تمرین / BP , FF \searrow کاهش ۳٪ صفحه =

Kernel

Input volume (+pad 1) ($7 \times 7 \times 3$)

Red

0	0	0	0	0	0	0	0
0	0	1	1	1	0	0	0
0	2	1	1	0	2	0	0
0	0	0	2	0	2	0	0
0	0	1	1	2	1	0	0
0	0	2	1	2	2	0	0
0	0	0	0	0	0	0	0

Filter ($3 \times 3 \times 3$)

w0 [:, :, 0]

0	0	0
1	0	0
1	0	0

$$0 \times 0 + 0 \times 0 + 0 \times 0 +$$

$$0 \times 1 + 0 \times 0 + 1 \times 0 +$$

$$0 \times 1 + 2 \times 0 + 1 \times 0 = 0$$

$$0 \times 0 + 0 \times 0 + 0 \times 0 +$$

$$0 \times 1 + 0 \times 0 + 1 \times 0 +$$

$$0 \times 1 + 2 \times 0 + 1 \times 0 = 0$$

0	0	0
1	0	0
1	0	0

Feature map

+ 1 for bias

1		
1		
-1		

=

Green

0	0	0	0	0	0	0	0
0	0	1	1	1	0	0	0
0	0	0	1	0	2	0	0
0	1	1	1	0	1	0	0
0	0	2	1	0	2	0	0
0	2	2	0	1	0	0	0
0	0	0	0	0	0	0	0

w0 [:, :, 1]

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

$$0 \times 1 + 0 \times 0 + 0 \times -1 +$$

$$0 \times -1 + 0 \times 1 + 1 \times 1 +$$

$$0 \times -1 + 0 \times 0 + 0 \times 1 = 1$$

$$0 \times 1 + 0 \times 0 + 0 \times -1 +$$

$$0 \times -1 + 0 \times 1 + 1 \times 1 +$$

$$0 \times -1 + 0 \times 0 + 0 \times 1 = 1$$

$$0 \times 1 + 0 \times 0 + 0 \times -1 +$$

$$0 \times -1 + 0 \times 1 + 1 \times 1 +$$

$$0 \times -1 + 0 \times 0 + 0 \times 1 = 1$$

1	7	5
4	4	9
5	5	2

=

Blue

0	0	0	0	0	0	0	0
0	0	2	1	1	0	0	0
0	1	2	1	0	2	0	0
0	1	1	2	2	0	0	0
0	0	1	1	1	1	0	0
0	2	0	1	0	0	0	0
0	0	0	0	0	0	0	0

w0 [:, :, 2]

1	-1	0
1	1	0
0	1	-1

$$0 \times 1 + 0 \times -1 + 0 \times 0 +$$

$$0 \times 1 + 0 \times 1 + 2 \times 0 +$$

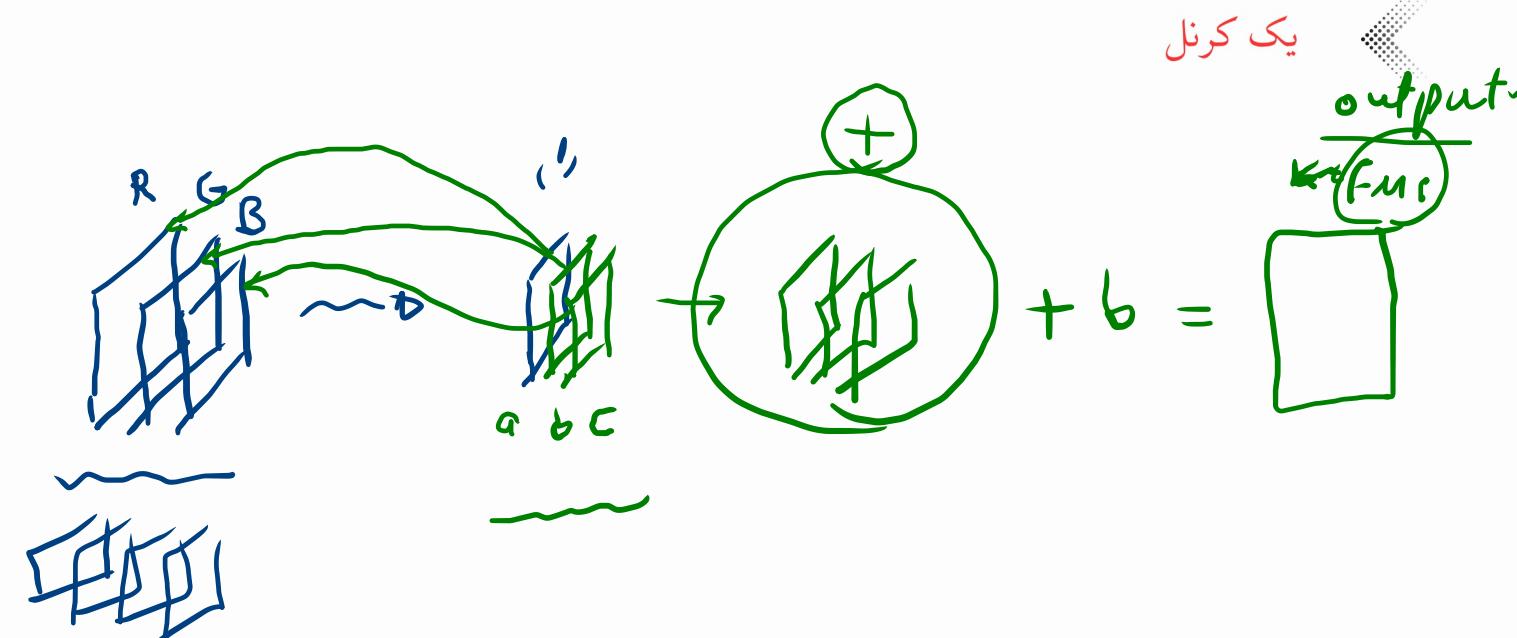
$$0 \times 0 + 1 \times 1 + 2 \times -1 = -1$$

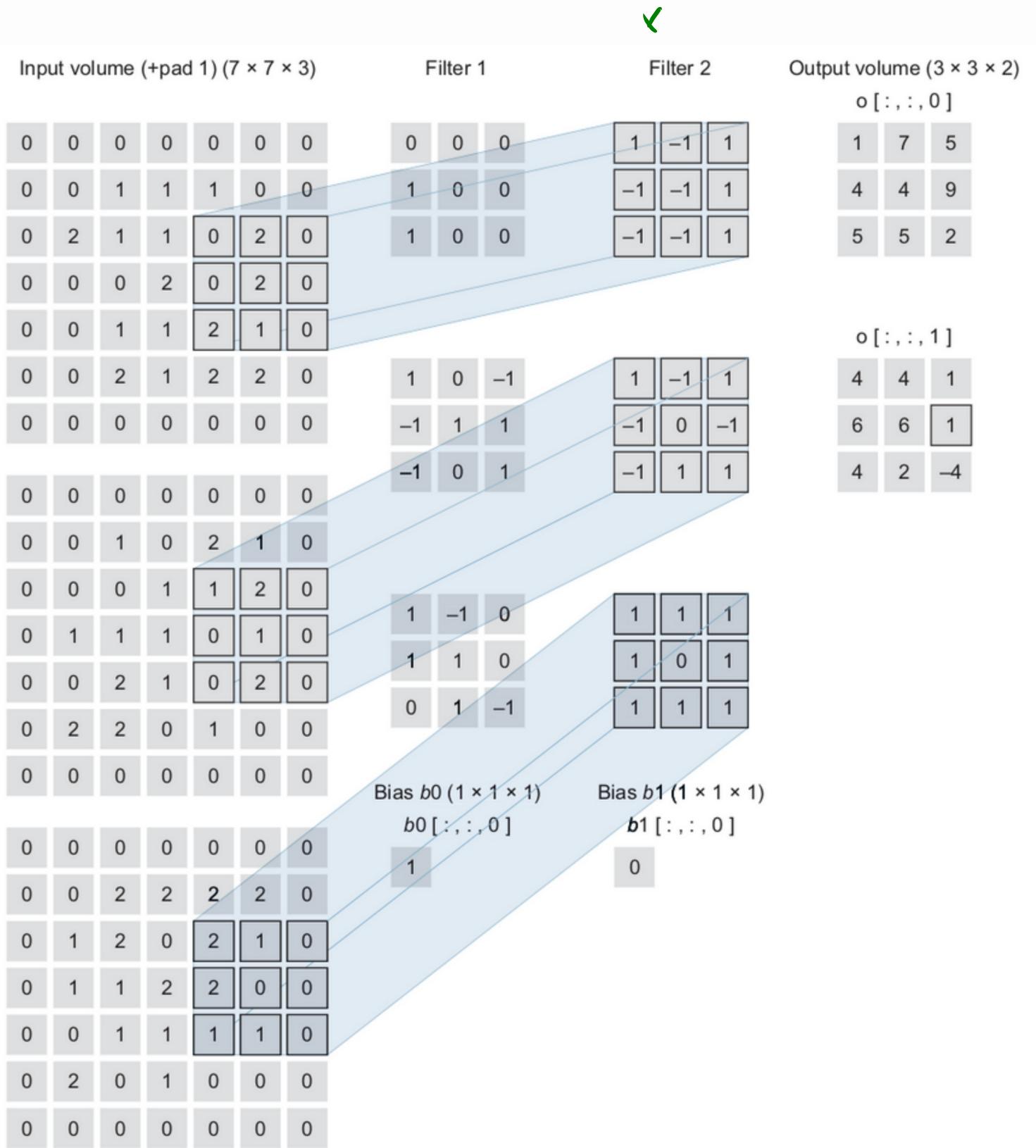
Blue

اعمال لایه کانولوشن روی ورودی با عمیق بیش از یک کanal (یک فیلتر یا کرنل):

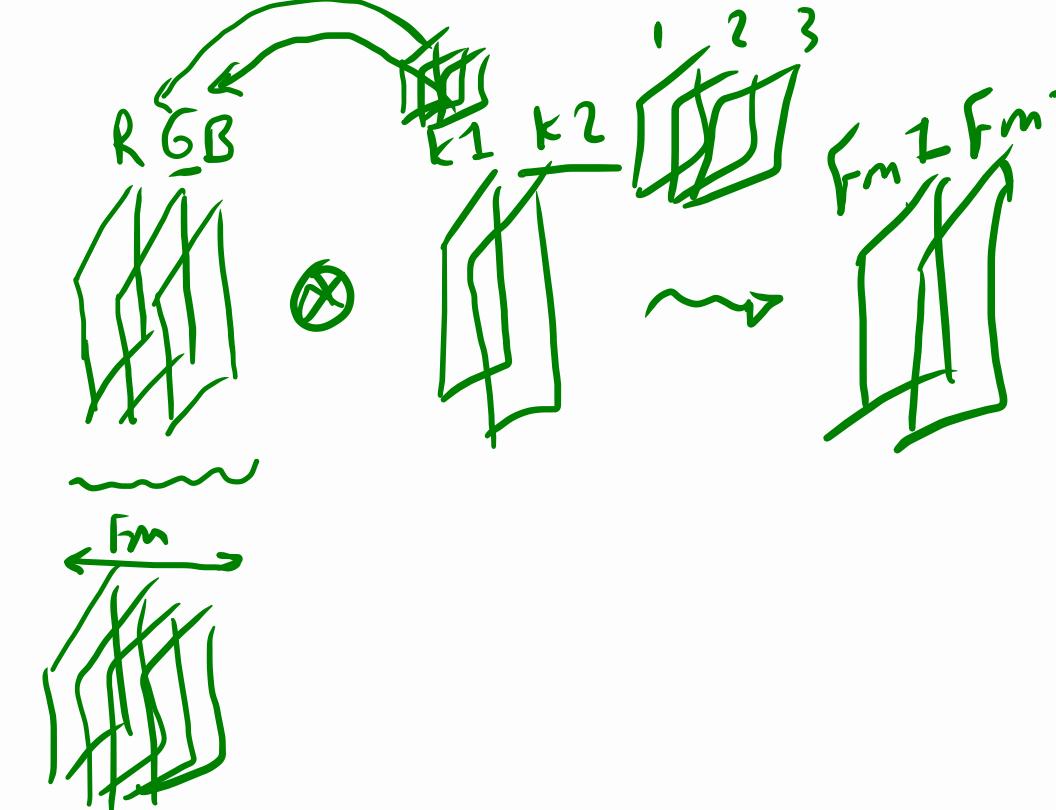
یک کرنل

output





اعمال لایه کانولوشن روی ورودی با عمق بیش از یک کanal (دو فیلتر یا کرنل):



No

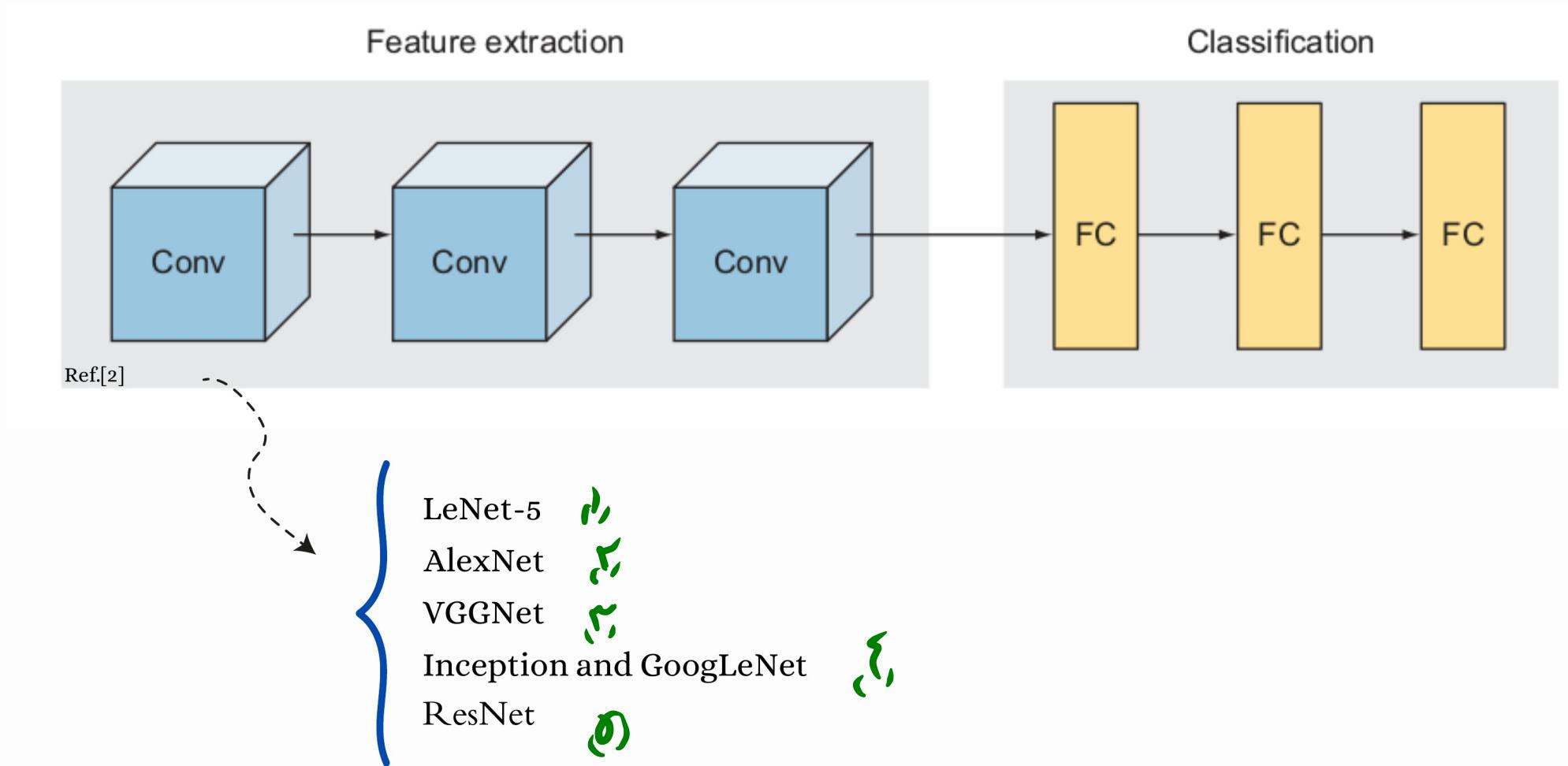
آیا شبکه های معمول کانولوشنی برای استخراج ویژگی های تصاویر کافیست؟

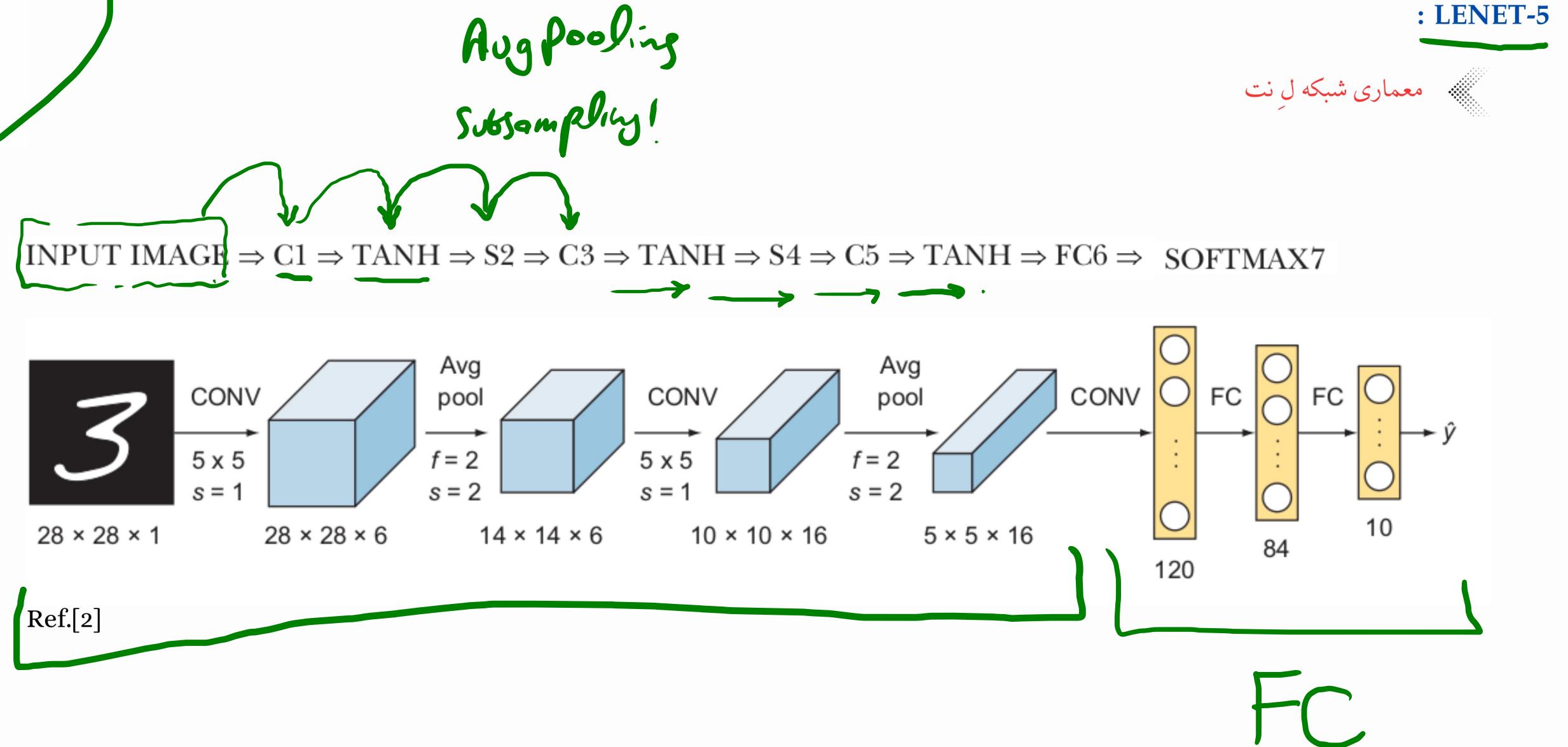
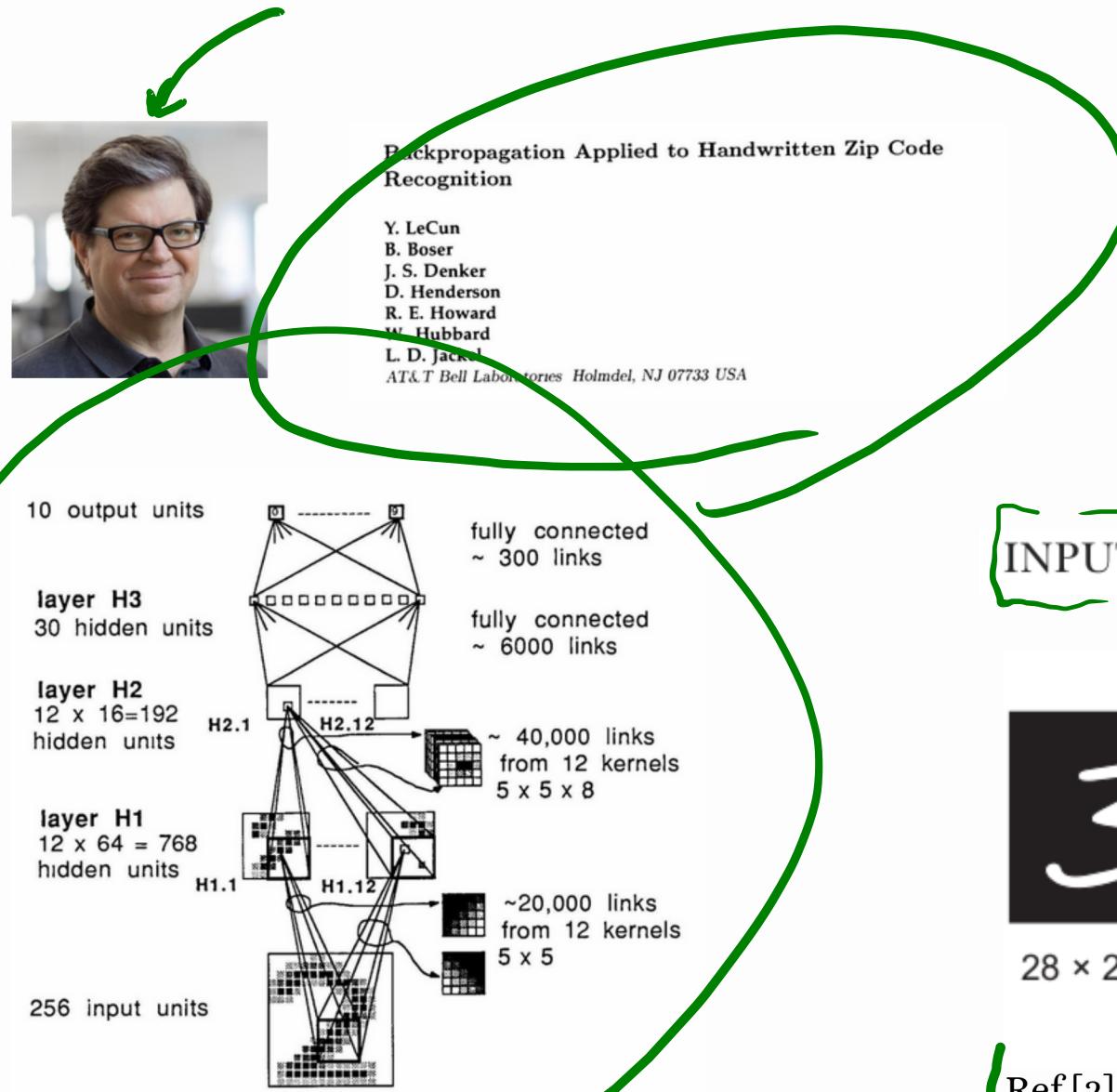
شبکه های پیشرفته کانولوشنی

Advanced CNN architectures

محل به کارگیری شبکه های پیشرفته

معماری های پیشرفته کانولوشنی، به منظور استخراج ویژگی طراحی شده اند!





: LENET-5

معماری شبکه لِ نت

: LENET-5

نوآوری های شبکه لِنت

- ✓ ۱. استفاده از لایه های پیچشی (Convolutional Layers)
- ✓ ۲. استفاده از لایه های تجمیع (Pooling Layers)
- ✓ ۳. استفاده از تابع فعال سازی غیر خطی (Non-Linearity) tanh
- ✗ ۴. آموزش مدل با پسانشار (Backpropagation) Backpropagation

CNN ~> Back propagation → vanishing gradient



ReLU

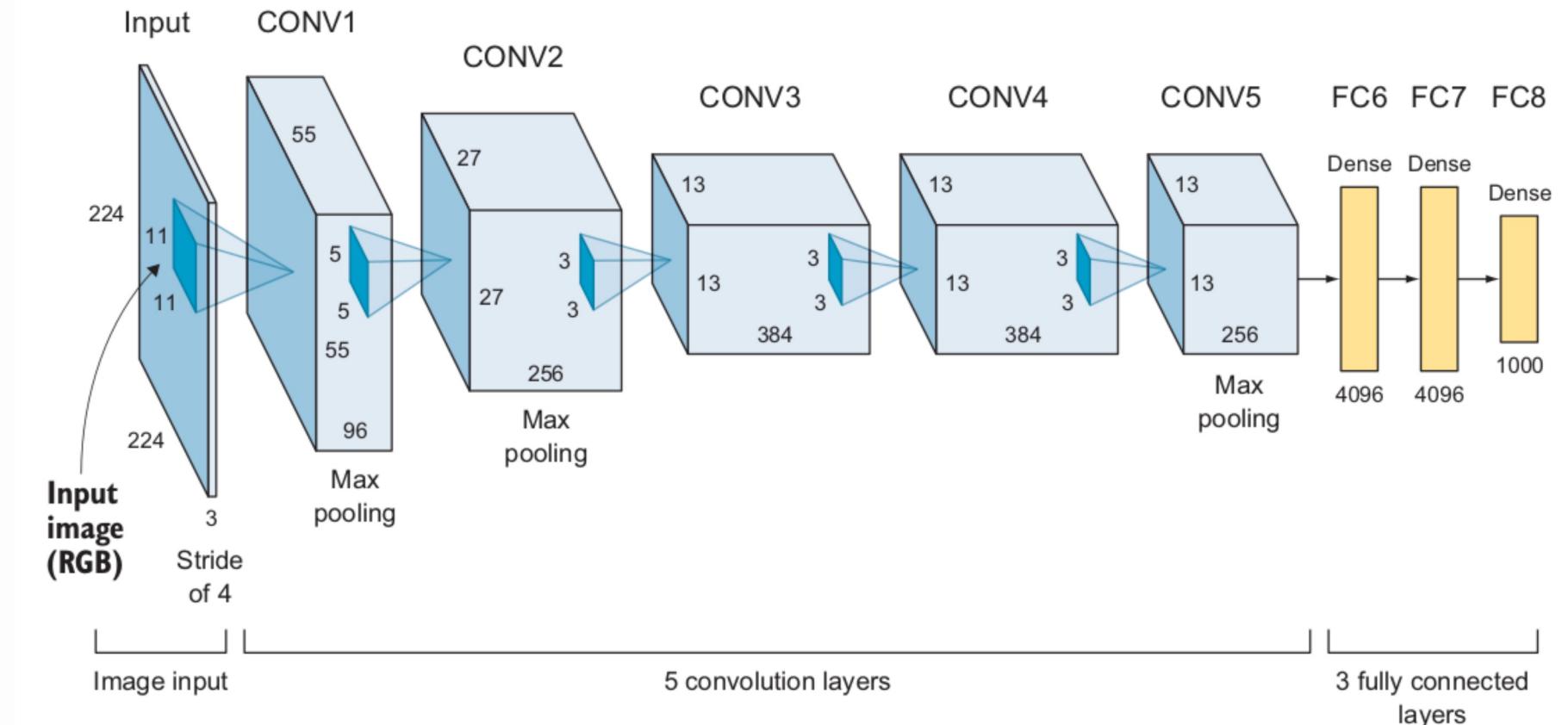
ImageNet Classification with Deep Convolutional Neural Networks

Alex Krizhevsky
University of Toronto
kriz@cs.utoronto.ca Ilya Sutskever
University of Toronto
ilya@cs.utoronto.ca Geoffrey E. Hinton
University of Toronto
hinton@cs.utoronto.ca

INPUT IMAGE \Rightarrow CONV1 \Rightarrow POOL2 \Rightarrow CONV3 \Rightarrow POOL4 \Rightarrow CONV5 \Rightarrow CONV6 \Rightarrow CONV7 \Rightarrow POOL8 \Rightarrow FC9 \Rightarrow FC10 \Rightarrow SOFTMAX7

: AlexNet

معماری شبکه الکس نت



Ref.[2]

: AlexNet

نوآوری های الکس نت



GPU

parallel programming.

Nvidia

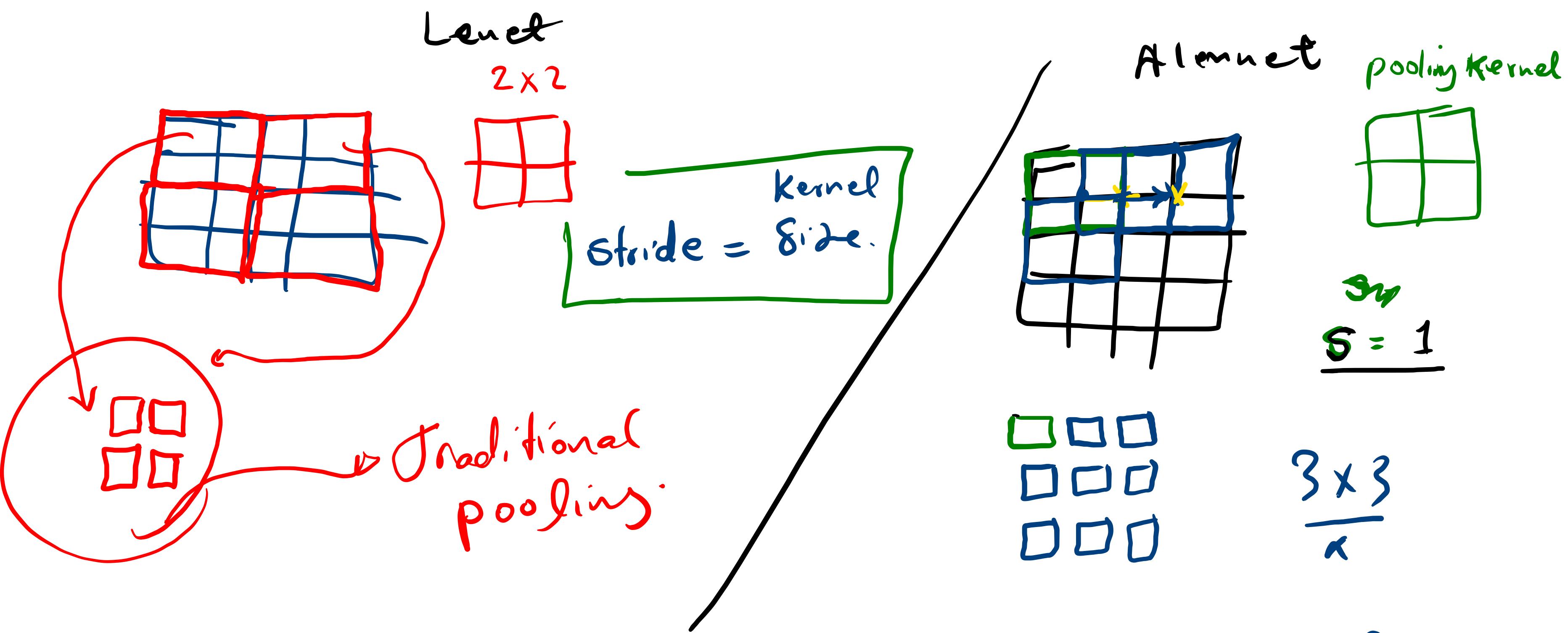
Cuda C++

pycuda py

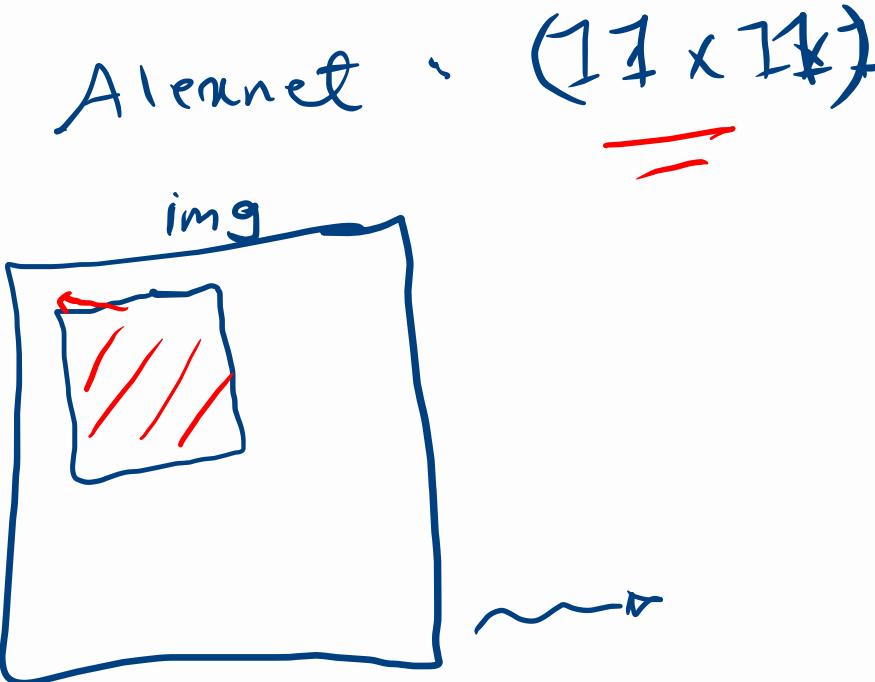
GPO

آموزش پردازه موارد

۱. افزایش عمق شبکه (8 لایه)
 ۲. استفاده از ReLU به جای Tanh و Sigmoid
 ۳. استفاده از Dropout برای جلوگیری از Overfitting
 ۴. استفاده از دو GPU برای آموزش موازی
 ۵. استفاده از Data Augmentation برای بهبود عملکرد
 ۶. استفاده از Overlapping Pooling به جای Traditional Pooling
- clip
rotate
brightness



(مُفهوم سُلْطَنِي بِعْدِ اسْتِرْجَاهِ الْمُؤْلِمَاتِ) Overlapping pooling

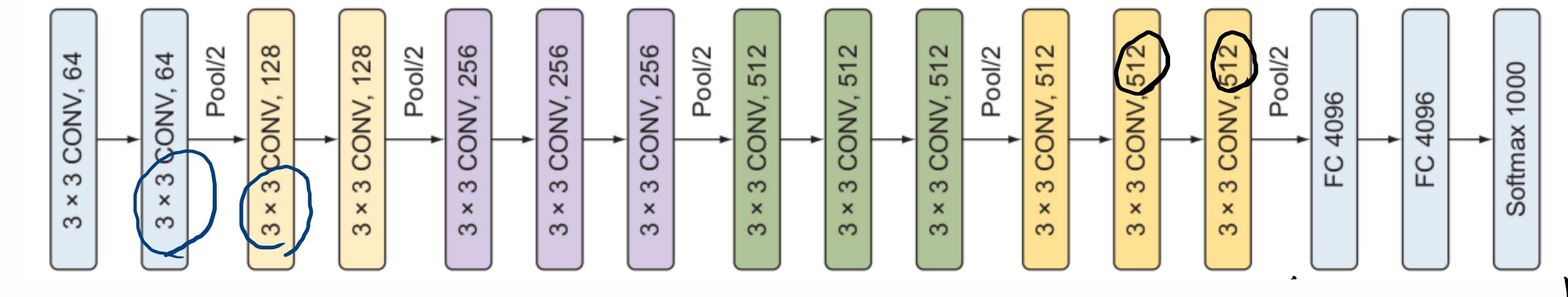


\rightarrow large, 11×11
 نیز سعیت رخراخه دارد.
 حین در مسافت افزایش کی شود!
 feature

$\times 3 \times 3$
 : VGGNet

معماری شبکه وی جی جی نت (16)

138 μm



dense

$\frac{1 \times 1}{?}$ 3×3 ✓ 5×5 ✓ 7×7
 maybe

8	3	-1	1
-	-	-	-
-	-	-	-
-	-	-	-

$$\textcircled{X} \quad \begin{matrix} 1 \\ 2 \end{matrix} \quad 1 = \alpha \quad \begin{array}{|c|c|c|c|} \hline & 8 & 3 & -2 & 1 \\ \hline \end{array}$$

! مفهوم feature extraction — عرضه سطح دریل \rightarrow جزء 1×1

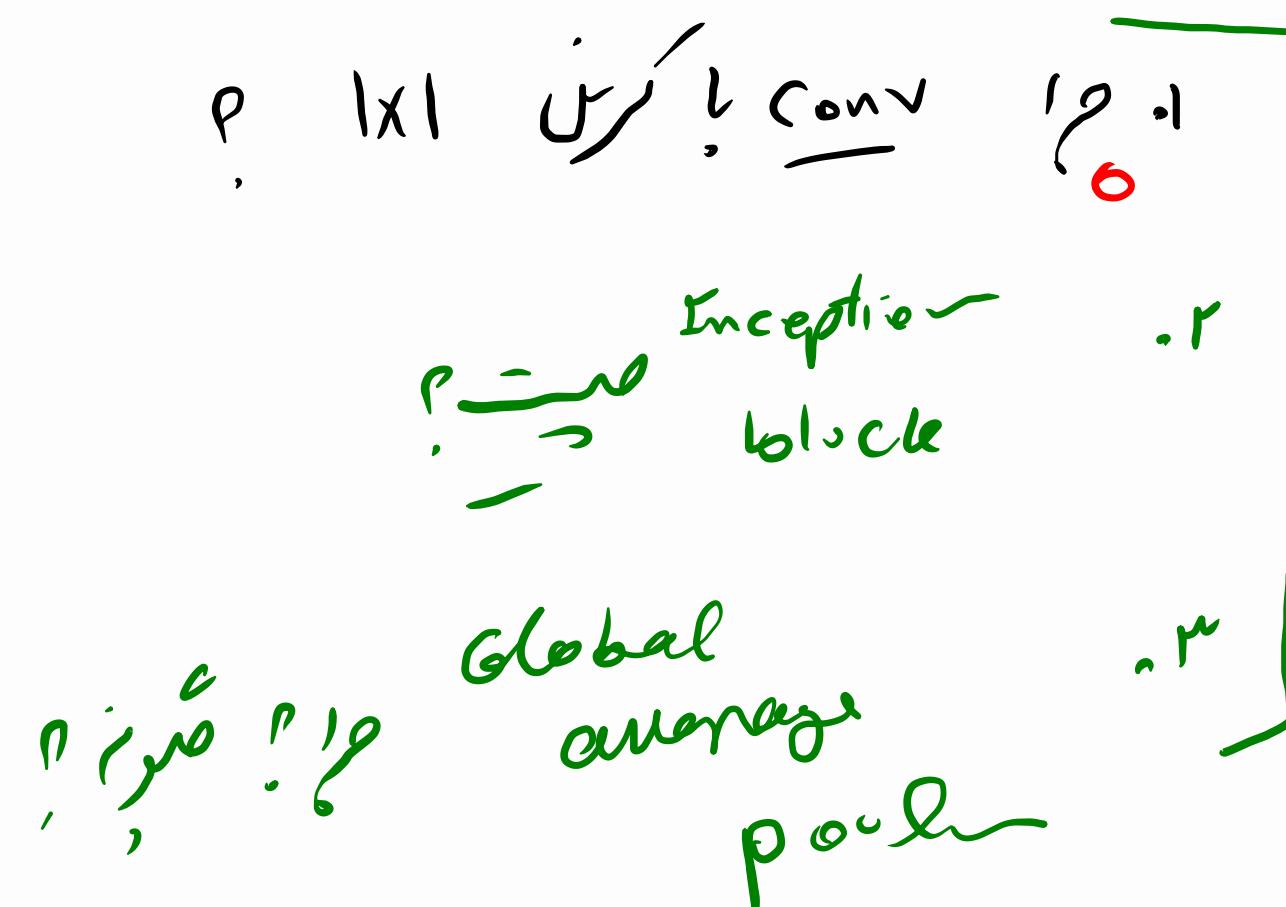
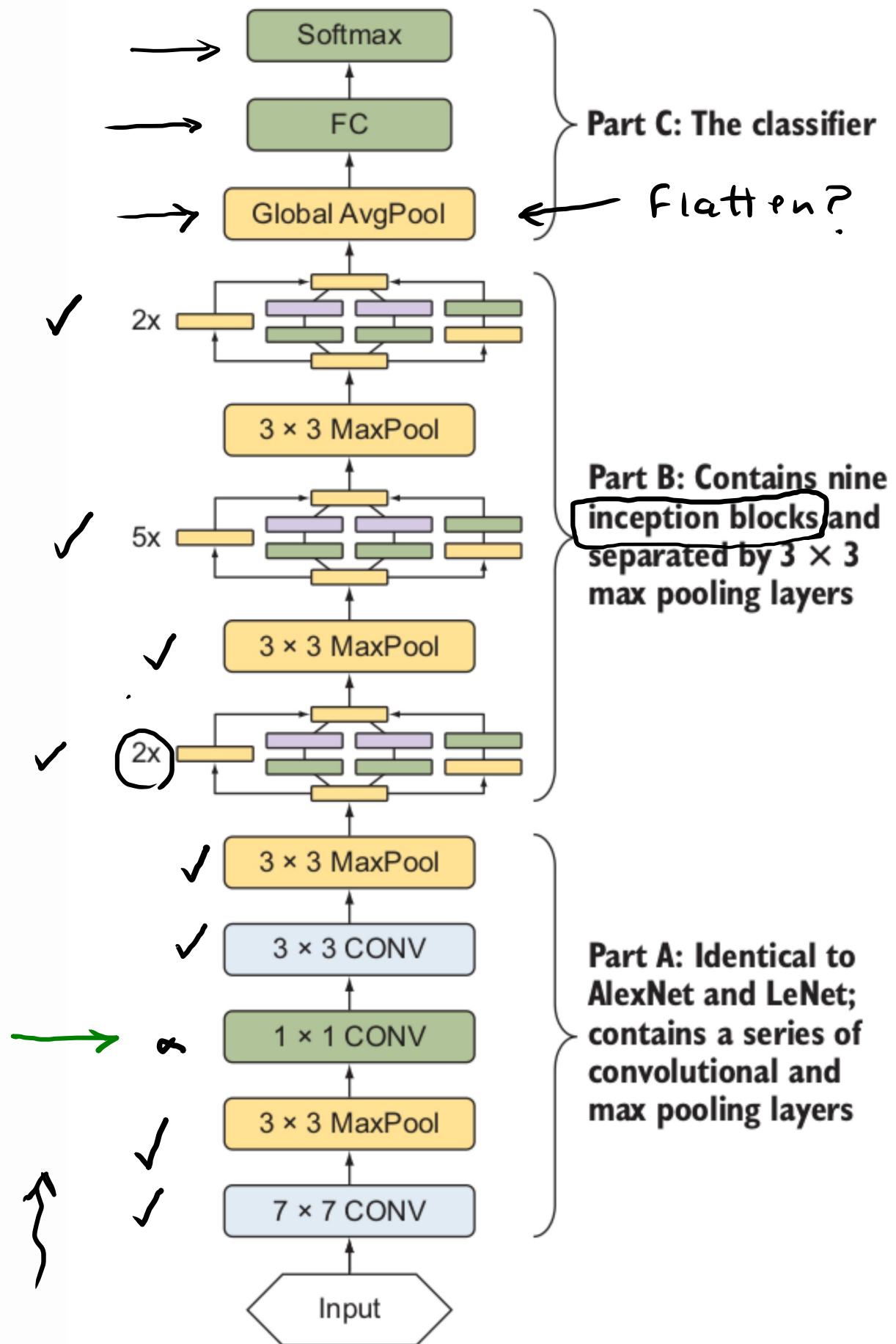
کسر نمودن

how? و نون sliders.

: VGGNet

نوآوری ها شبکه وی جی جی (۱۶)

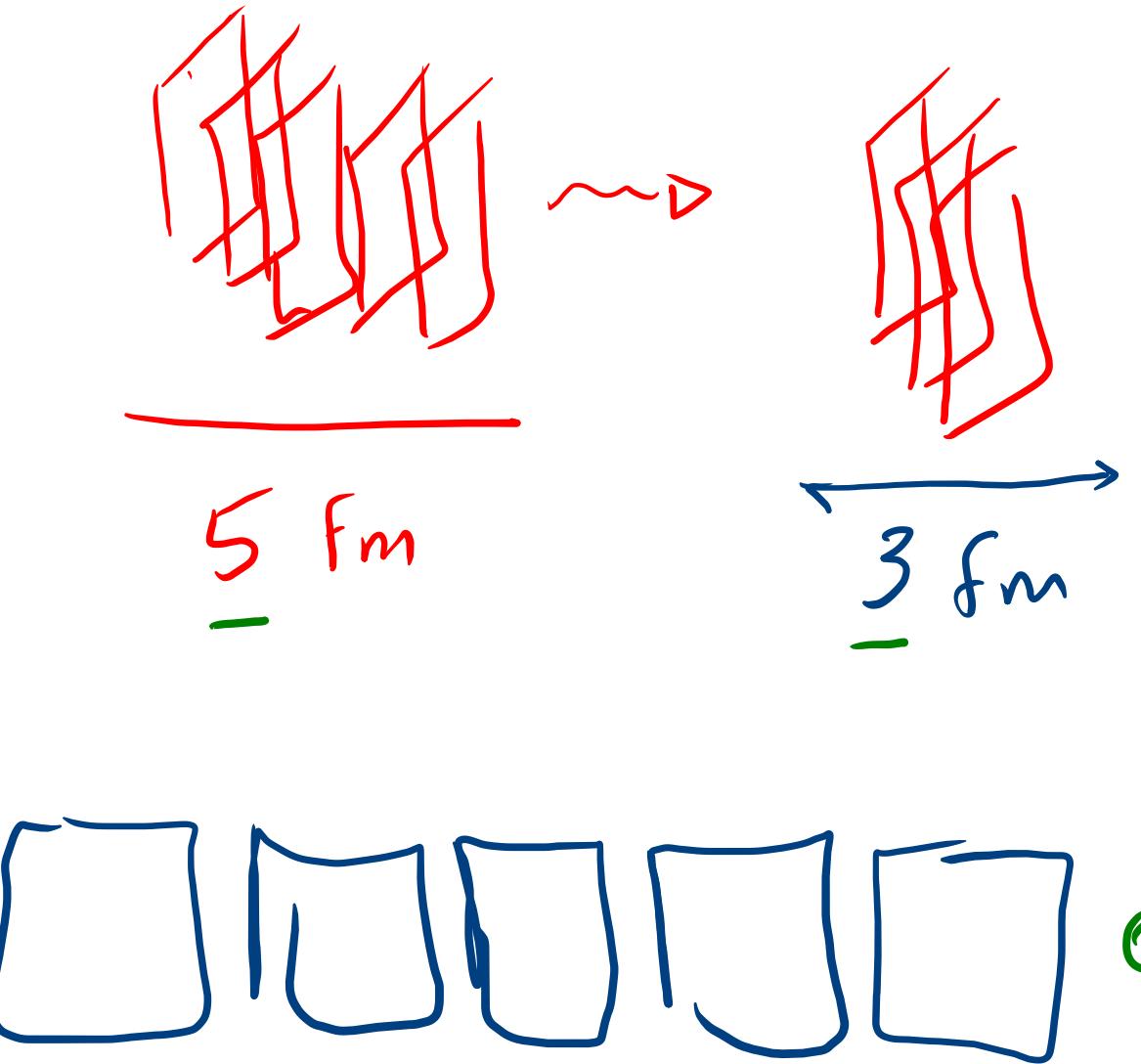
- ✓ ۱. استفاده از لایه های کانولوشنی کوچک (3×3)
- ✓ ۲. افزایش عمق شبکه (تا ۱۹ لایه)
- ✓ ۳. استفاده از ساختار تکرار شونده
- ✓ ۴. استفاده از ماس پولینگ 2×2 برای کاهش ابعاد



: GoogLeNet

معماری شبکه گوگل نت :





هر آن کار را سه مرتبه نمایش می‌خواهیم
و نیز هر ترتیب معکوس را داریم!

"برنجفه سه از گزینه‌ها باشد، انتخاب کنید"

: GoogLeNet

نوآوری ها شبکه گوگل نت

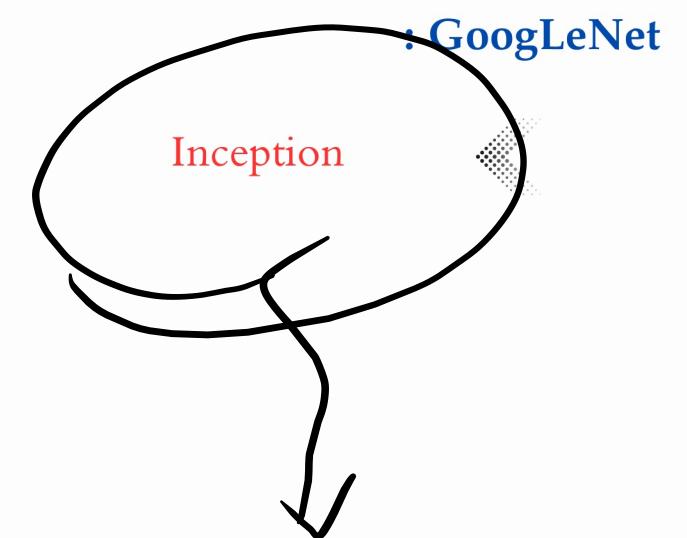
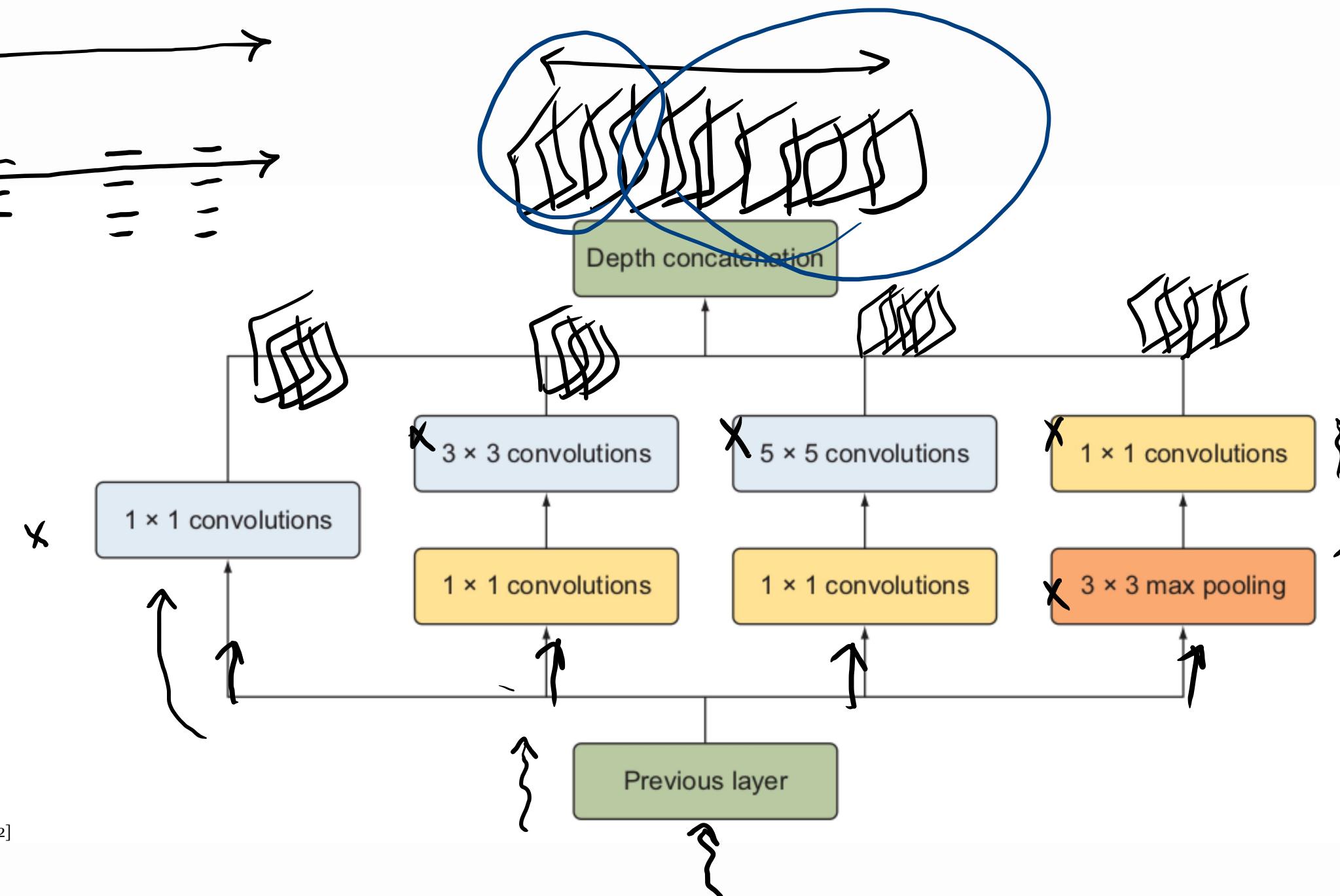
- معرفی ماژول Inception
- کاهش تعداد پارامترها 4 میلیون در مقابل 138 میلیون در VGGNet
- حذف لایه های Fully Connected سنگین
- اضافه کردن Auxiliary Classifiers

LeNet $f \rightarrow \text{output}$

Alexnet $f \rightarrow \text{output}$

VGG $f \rightarrow \text{output}$

Googlenet $f = \text{Depth concatenation}$

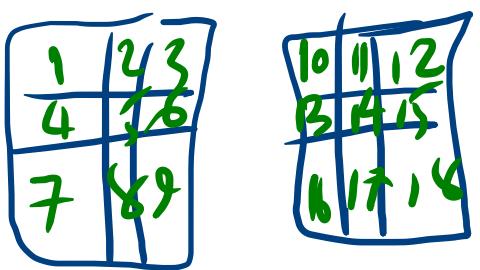
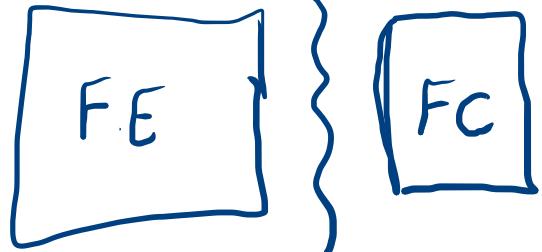


VG GNet ; 138m
Googlenet: 4m

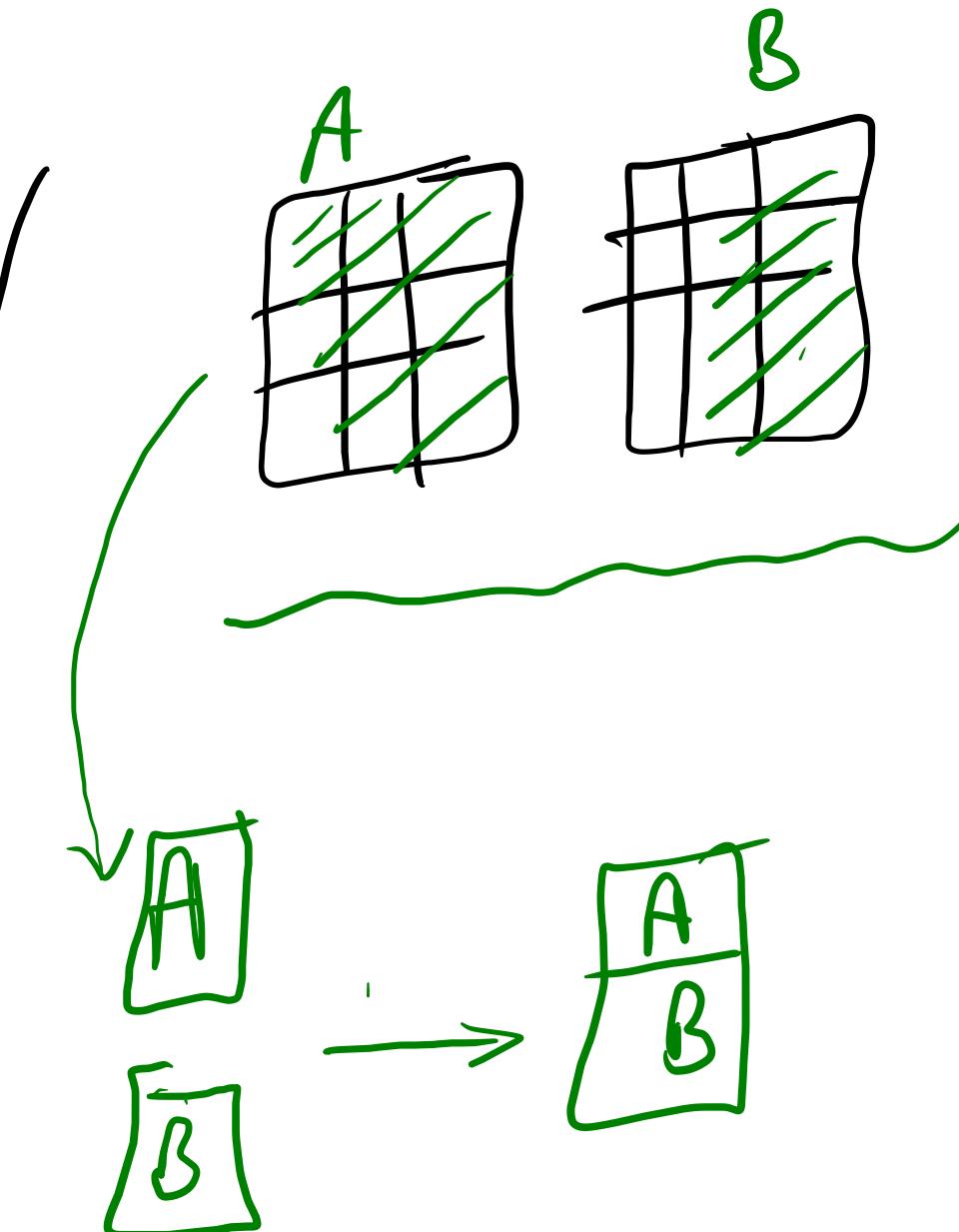
Global average pooling

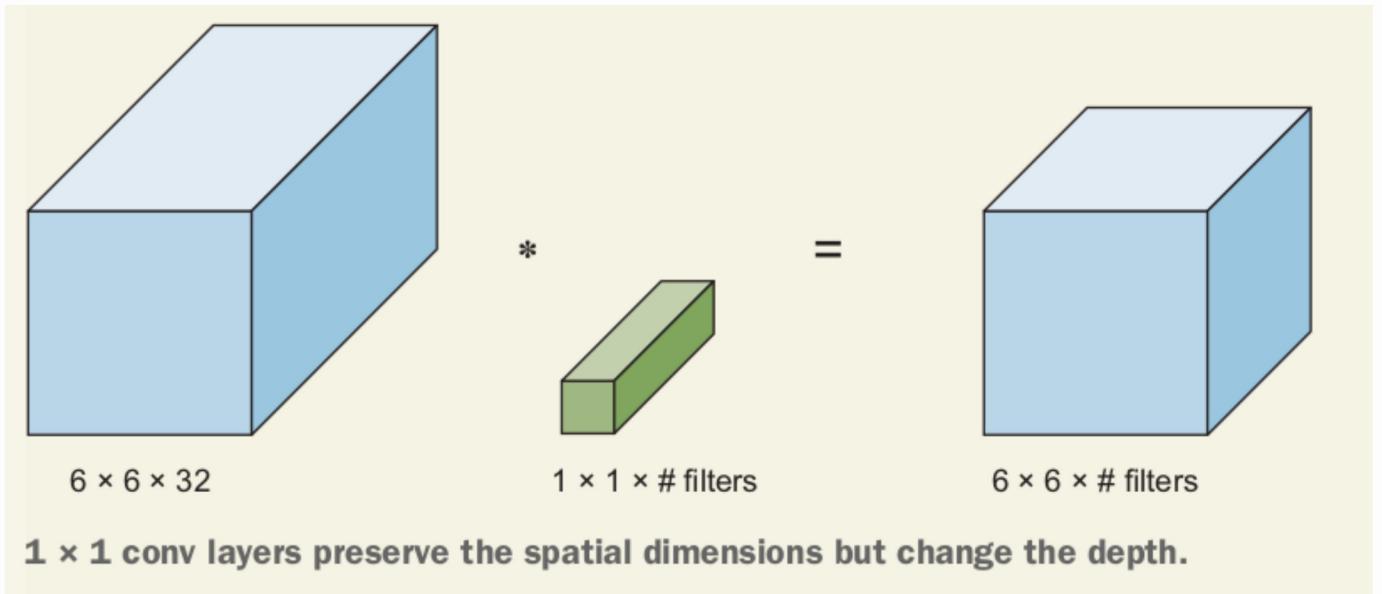
=

flatten



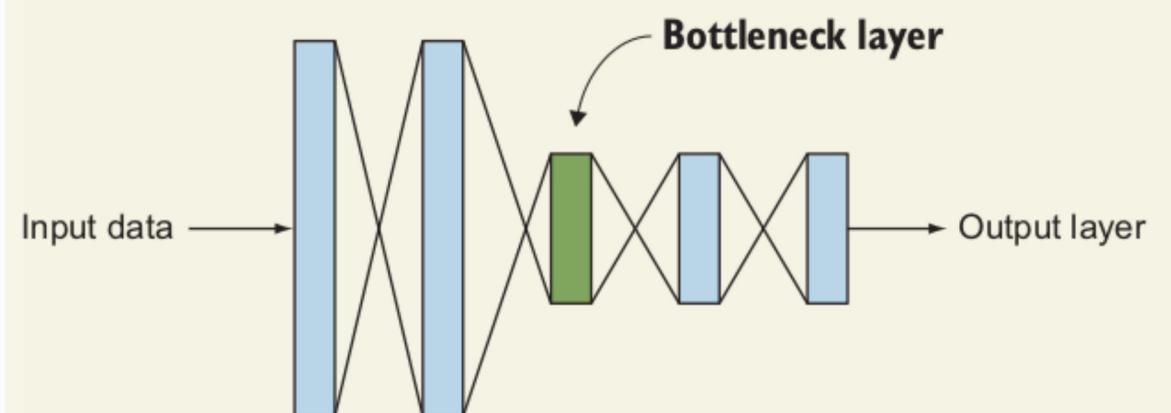
~ Flatten →





Ref[2]

The 1×1 convolutional layers are also known as bottleneck layers because the bottleneck is the smallest part of the bottle and reduce layers reduce the dimensionality of the network, making it look like a bottleneck:



1 × 1 convolutional layers are called bottleneck layers.

Ref[2]

: GoogLeNet

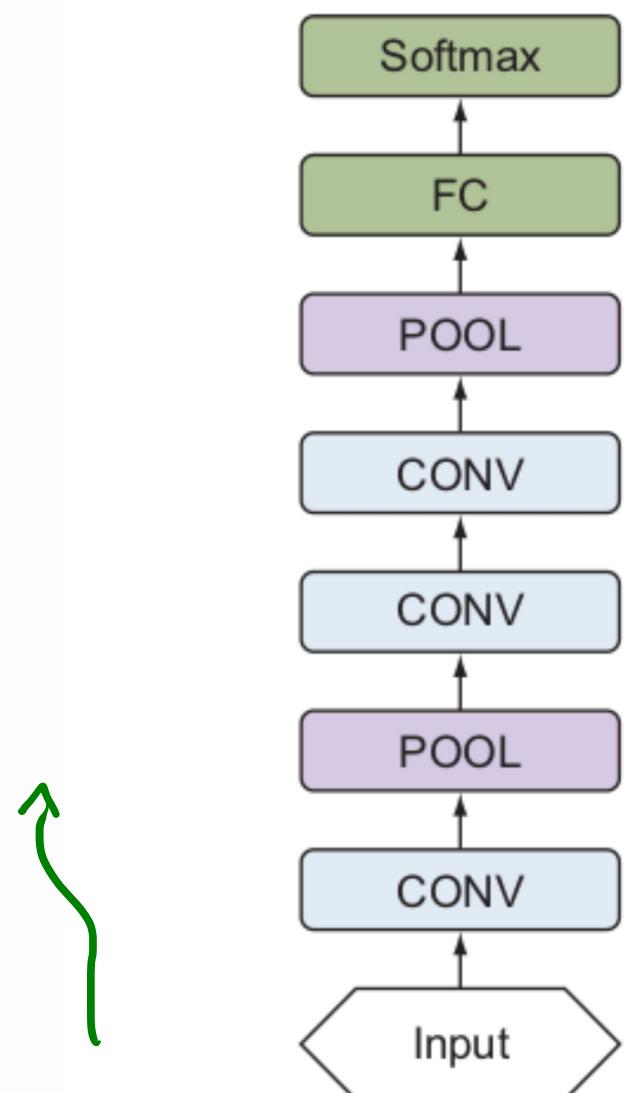
کرنل یک در یک

1×1

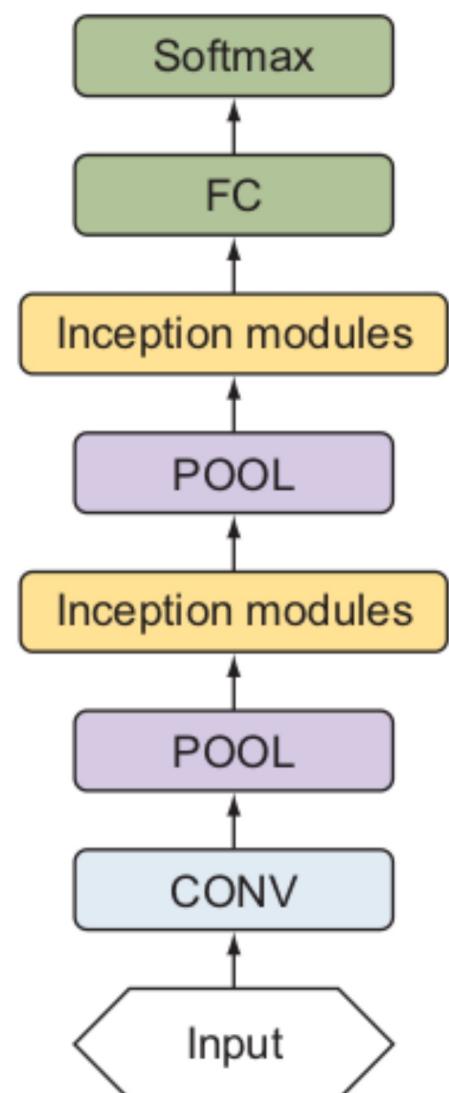
$\text{conv}(1 \times 1) \rightarrow$ bottleneck layer

Ref.[2]

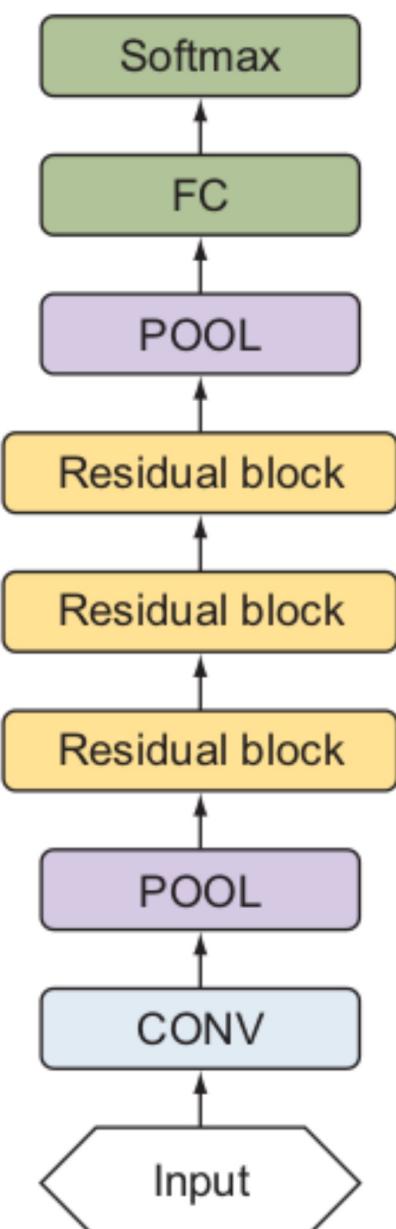
Classical CNN architecture



Inception modules

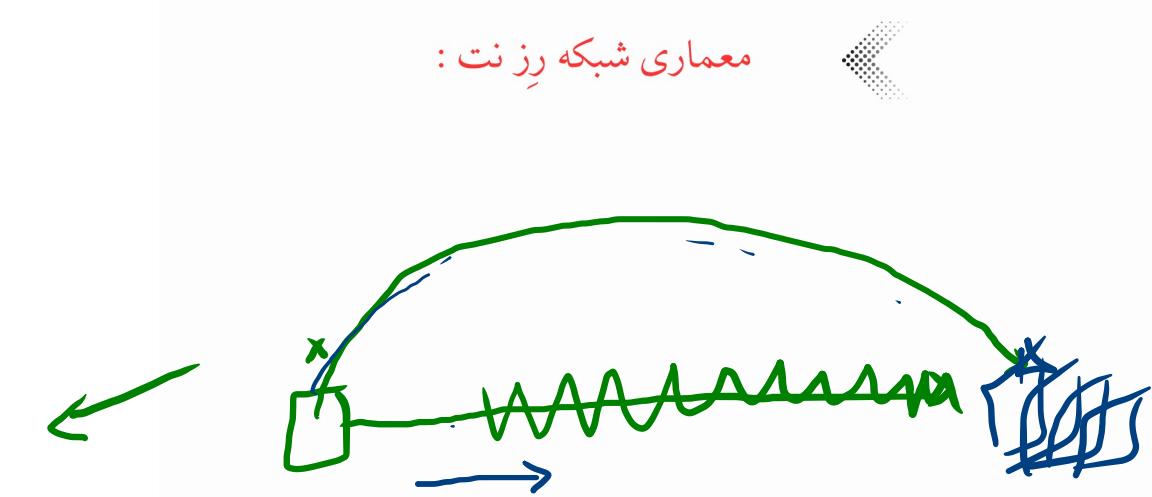


Residual blocks



: ResNet

معماری شبکه رز نت :



: ResNet

نوآوری های شبکه رز نت :

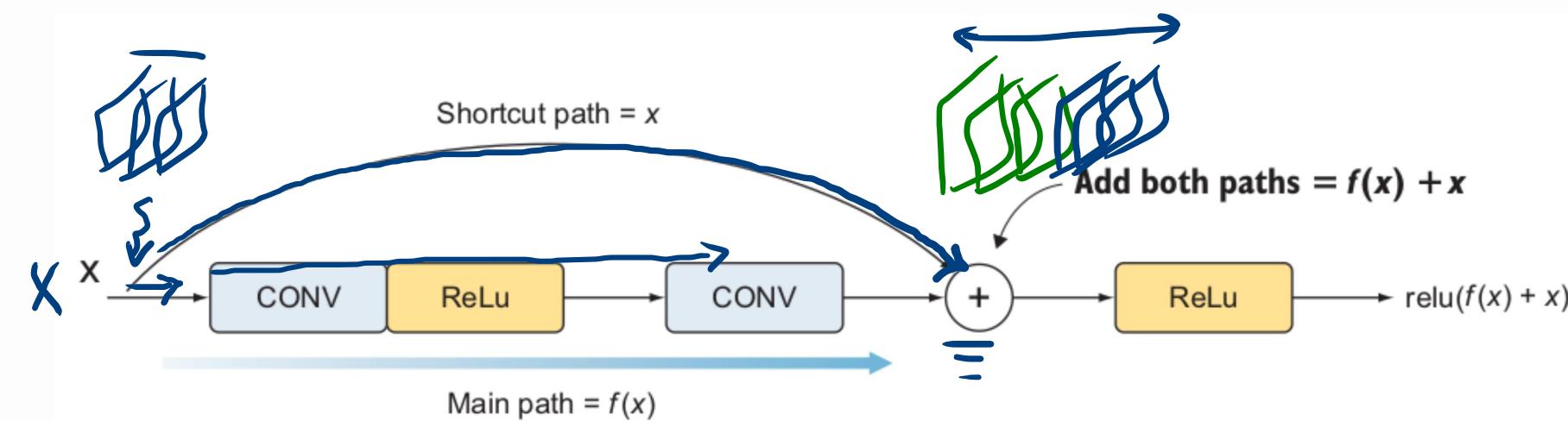


- معرفی اتصال های باقی مانده (Residual Connections)
- امکان طراحی شبکه های بسیار عمیق (50، 101 و 152 لایه)
- کاهش پارامترها نسبت به VGGNet

افزایش شمع دارکه های افزایش خوب برای سری

: ResNet

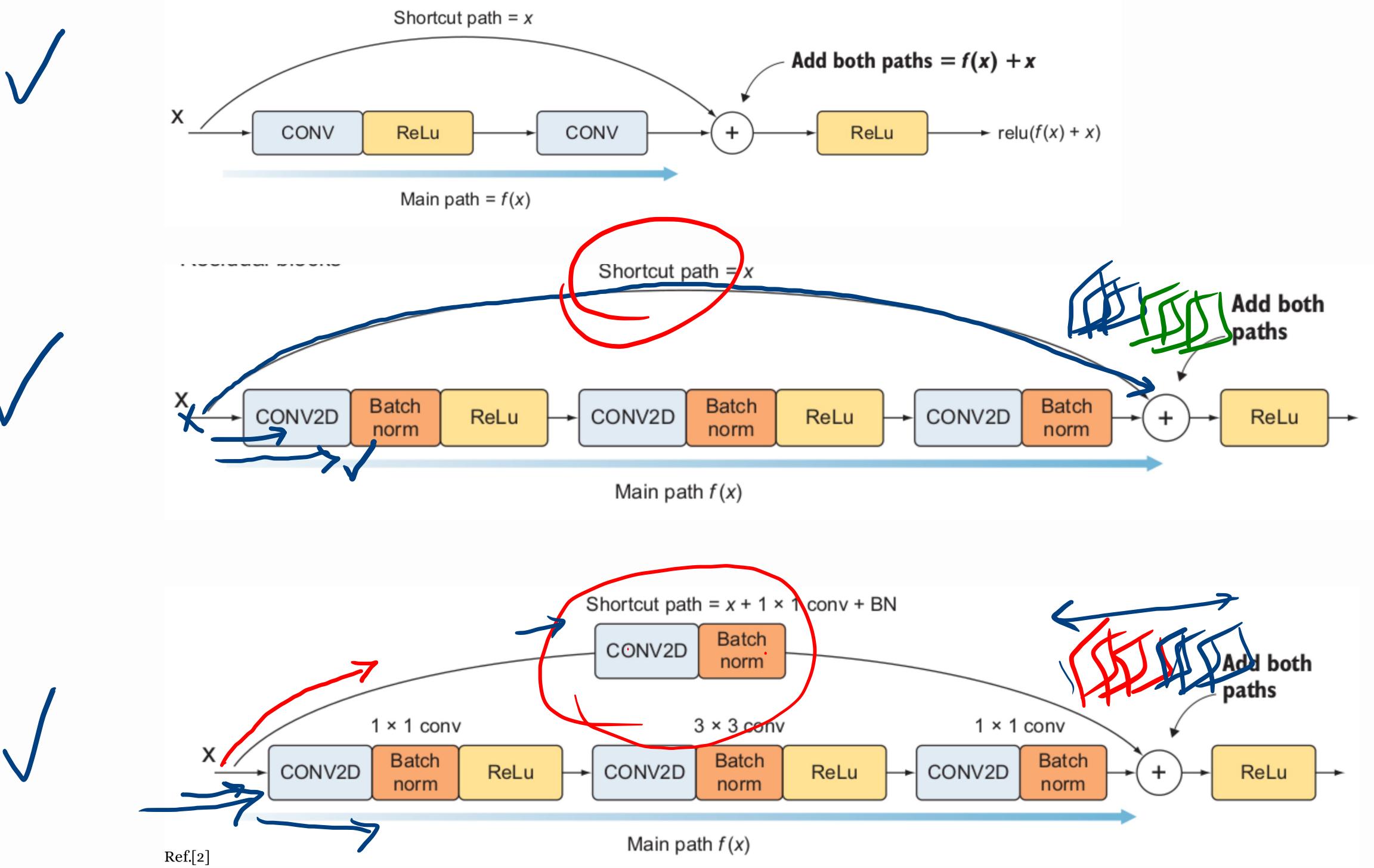
Residual Block (2)



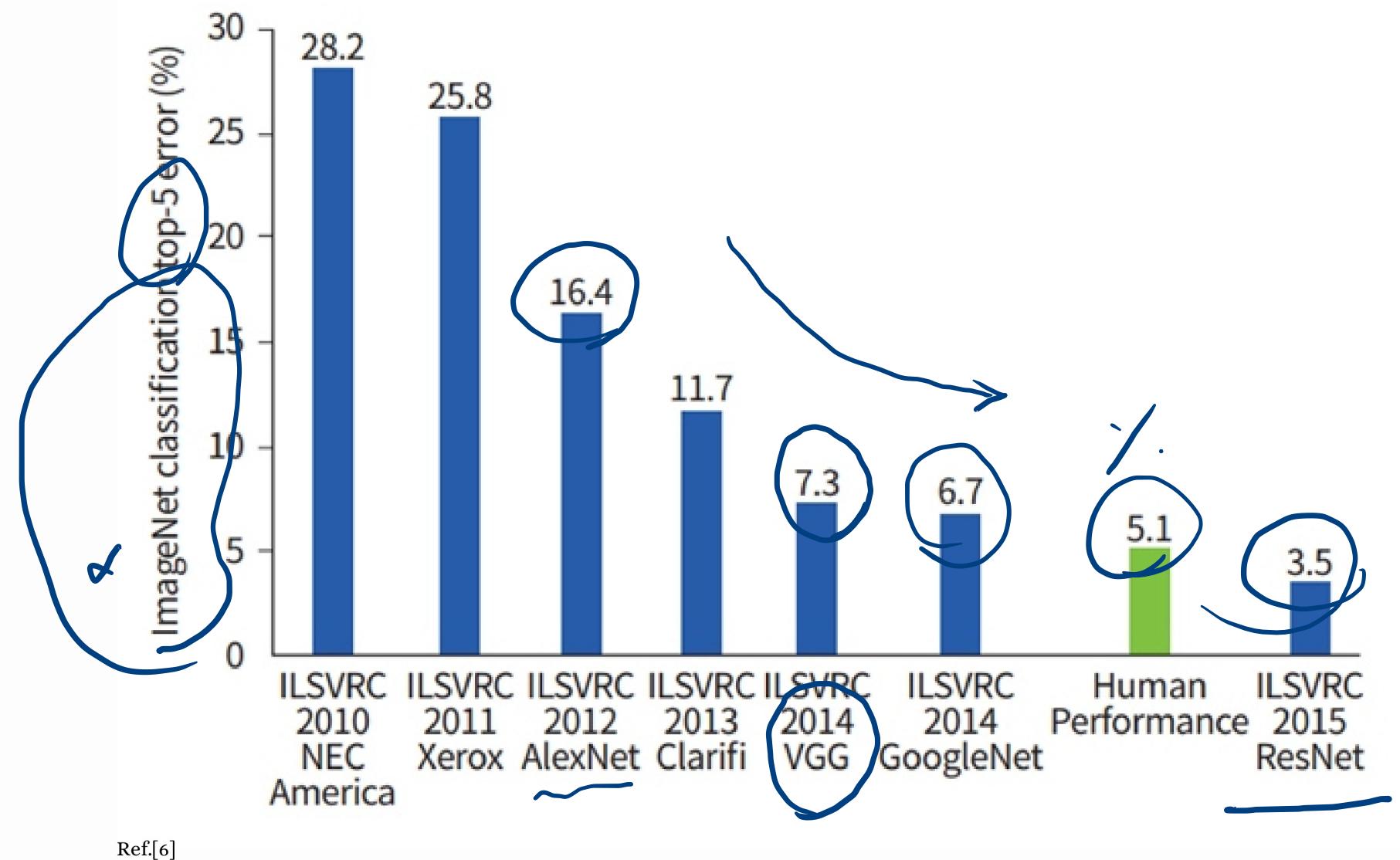
Ref.[2]



Residual Block (2)



مقایسه شبکه های پیشرفته معرفی شده:

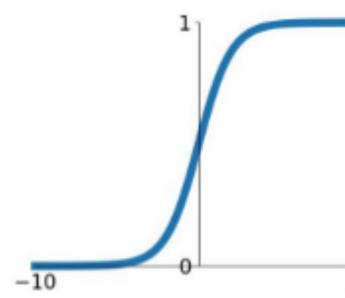


توابع فعال ساز

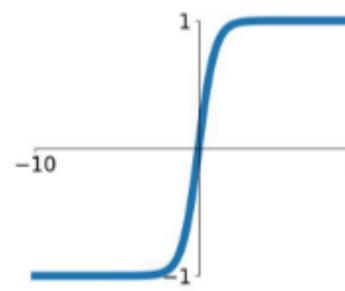
(Activation Functions)

Sigmoid

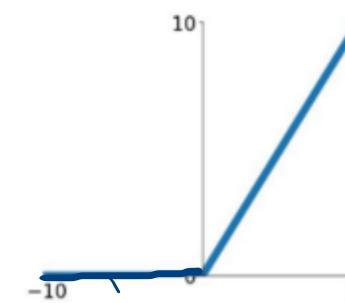
$$\sigma(x) = \frac{1}{1+e^{-x}}$$

**tanh**

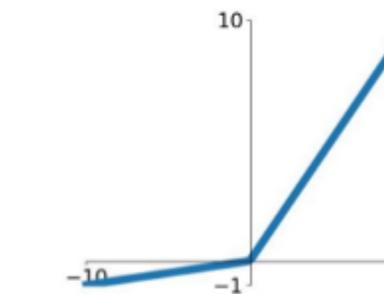
$$\tanh(x)$$

**ReLU**

$$\max(0, x)$$

**Leaky ReLU**

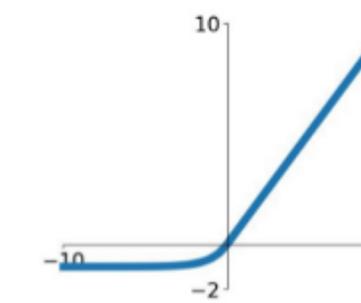
$$\max(0.1x, x)$$

**Maxout**

$$\max(w_1^T x + b_1, w_2^T x + b_2)$$

ELU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$

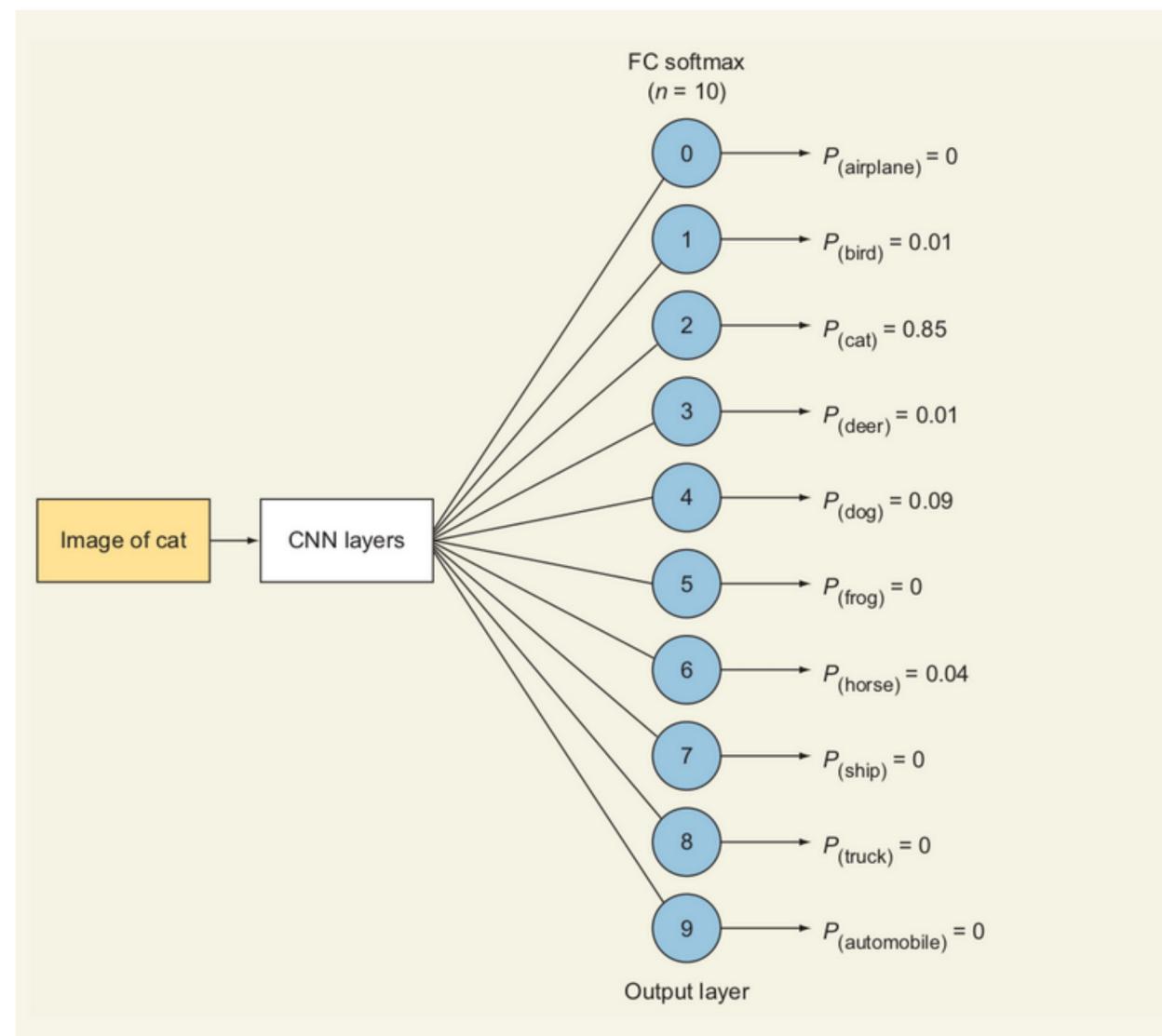


multi-class classification

$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}$$

تابع فعال ساز سافتمنکس (Softmax)

برای بیشتر از یک کلاس



Ref[2]

معیارهای ارزیابی

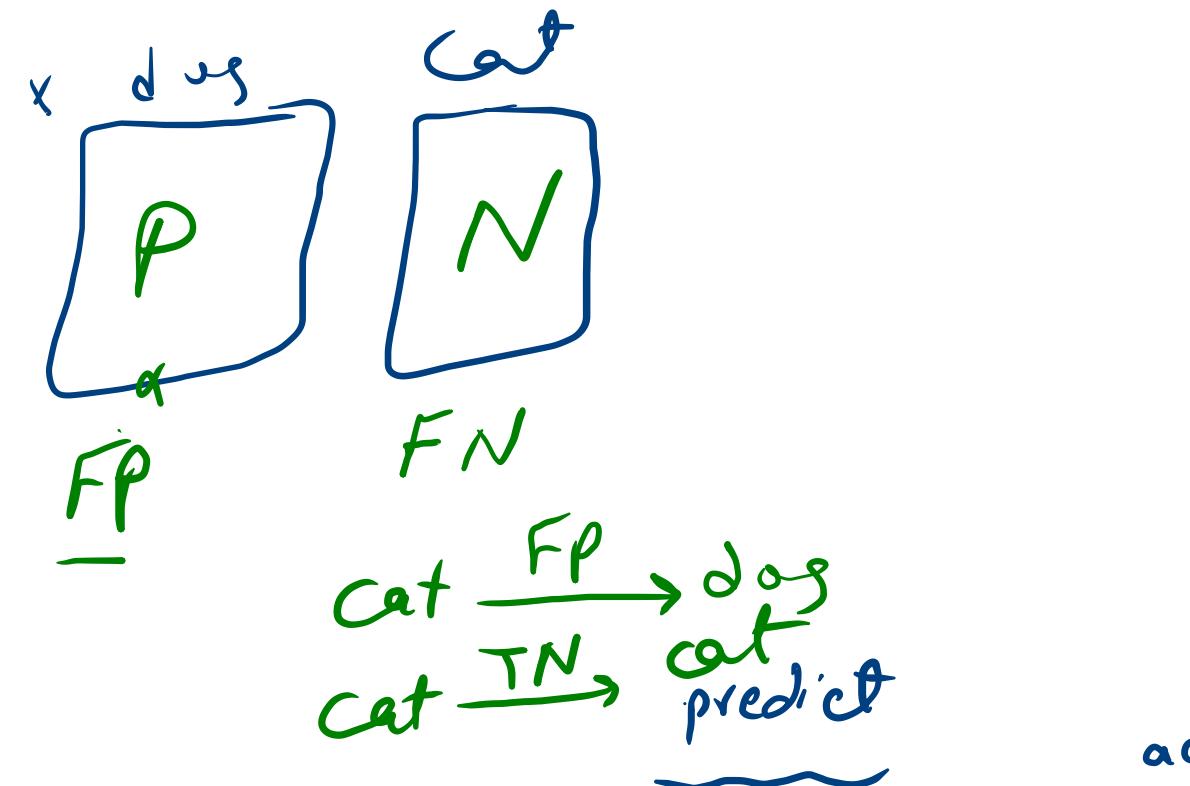
کلاسیفیکیشن (دسته بندی)

Predicted	
Actual	
True Positives TP	False Negatives FN
False Positives FP	True Negatives TN



Predicted	
Actual	
unicorn	non-unicorn
non-unicorn TP 90	FN 10
unicorn FP 60	TN 940

Ref.[7]



ماتریس درهم ریختگی (Confusion Matrix)

نمونه‌هایی که واقعاً مثبت هستند و مدل آن‌ها را مثبت پیش‌بینی کرده است. **TP (True Positive)**

منجذب
منجذب

نمونه‌هایی که واقعاً منفی هستند ولی مدل آن‌ها را مثبت پیش‌بینی کرده است (مثلاً فرد سالم را بیمار تشخیص دهد). **FN (False Negative)**

منجذب
منجذب

نمونه‌هایی که واقعاً منفی هستند ولی مدل آن‌ها را منفی پیش‌بینی کرده است (مثلاً فرد بیمار را سالم تشخیص دهد). **FP (False Positives)**

منجذب
منجذب

نمونه‌هایی که واقعاً منفی هستند و مدل آن‌ها را منفی پیش‌بینی کرده است. **TN (True Negative)**

منجذب
منجذب

دقت (Accuracy)

نسبت نمونه‌های درست طبقه‌بندی شده به کل نمونه‌ها

$$\frac{TP + TN}{TP + FP + TN + FN}$$

معارضه
معارضها

: Positive Predictive Value – PPV یا Precision

در صد پیش‌بینی‌های مثبت که واقعاً مثبت هستند.



- زمانی که هزینه خطای نوع اول (False Positive) بالاست و مهم است که پیش‌بینی‌های مثبت مدل دقیق باشند.

$$\frac{TP}{TP + FP}$$

• True Positive Rate – TPR یا Sensitivity یا Recall

درصد نمونه‌های مثبت که درست شناسایی شده‌اند.



- زمانی که هزینه خطای نوع دوم (False Negative) بالاست و مهم است که همه نمونه‌های مثبت به درستی شناسایی شوند.

$$\alpha \frac{TP}{TP + FN}$$

: F1 score

میانگین هارمونیک Precision و Recall

- زمانی که بین Precision و Recall تعادل برقرار کنیم و داده‌ها نامتوازن هستند.

$$\frac{\frac{2}{precision + recall}}{\frac{1}{precision} + \frac{1}{recall}} = \frac{2 * precision * recall}{precision + recall}$$

مثال:

- اگر Precision بالا باشد اما Recall پایین، ممکن است افراد مجرم از سیستم عبور کنند.
- اگر Recall بالا باشد اما Precision پایین، ممکن است افراد بی‌گناه اشتباه شناسایی شوند.

مشکلات مدل های پیشرفته گفته شده :

Data Hungry

۱. نیاز به حجم زیاد داده برای جلوگیری از Overfitting

✓ ۲. مشکل جمع آوری و برچسب گذاری داده

۳. مشکل Generalization (تعمیم پذیری) در صورت داده های کم

داده

Underfit

Overfit

پایان

بخش مقدماتی بخش کلاسیفیکیشن

The GrC

سوال

اگر تصاویر کافی برای آموزش مدل نداشته باشم ،
راه حل چیست ؟

کلاسیفیکیشن (دسته بندی)