

Universidade Federal da Paraíba

Centro de Informática

---

Departamento de Informática

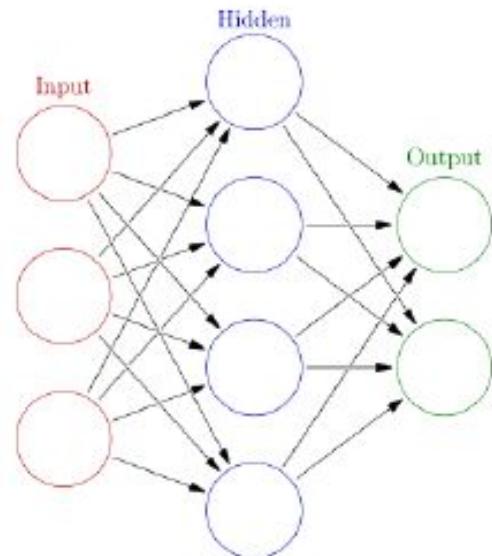
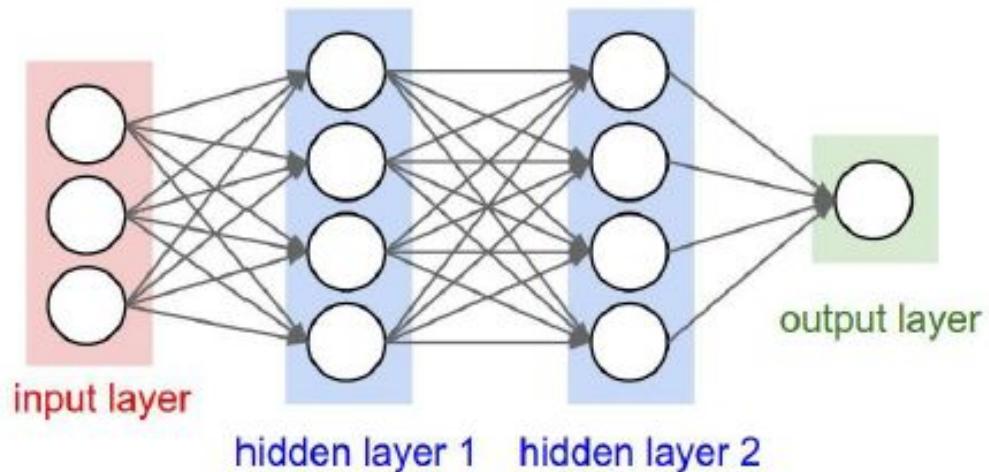
# Aprendizado Profundo Redes Neurais Convolucionais (Adaptado do Material do Prof. Leonardo Batista)

Thais Gaudencio

Tiago Maritan

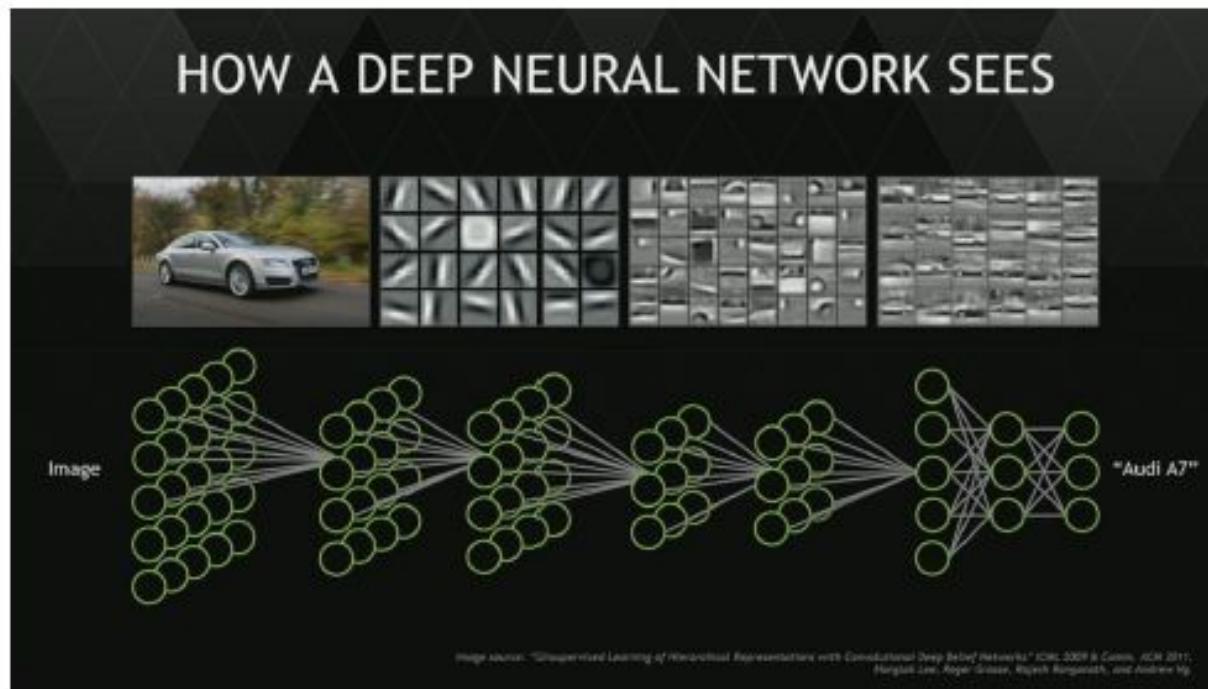
# Redes Neurais Clássicas

- De dezenas a dezenas de milhares de “neurônios”



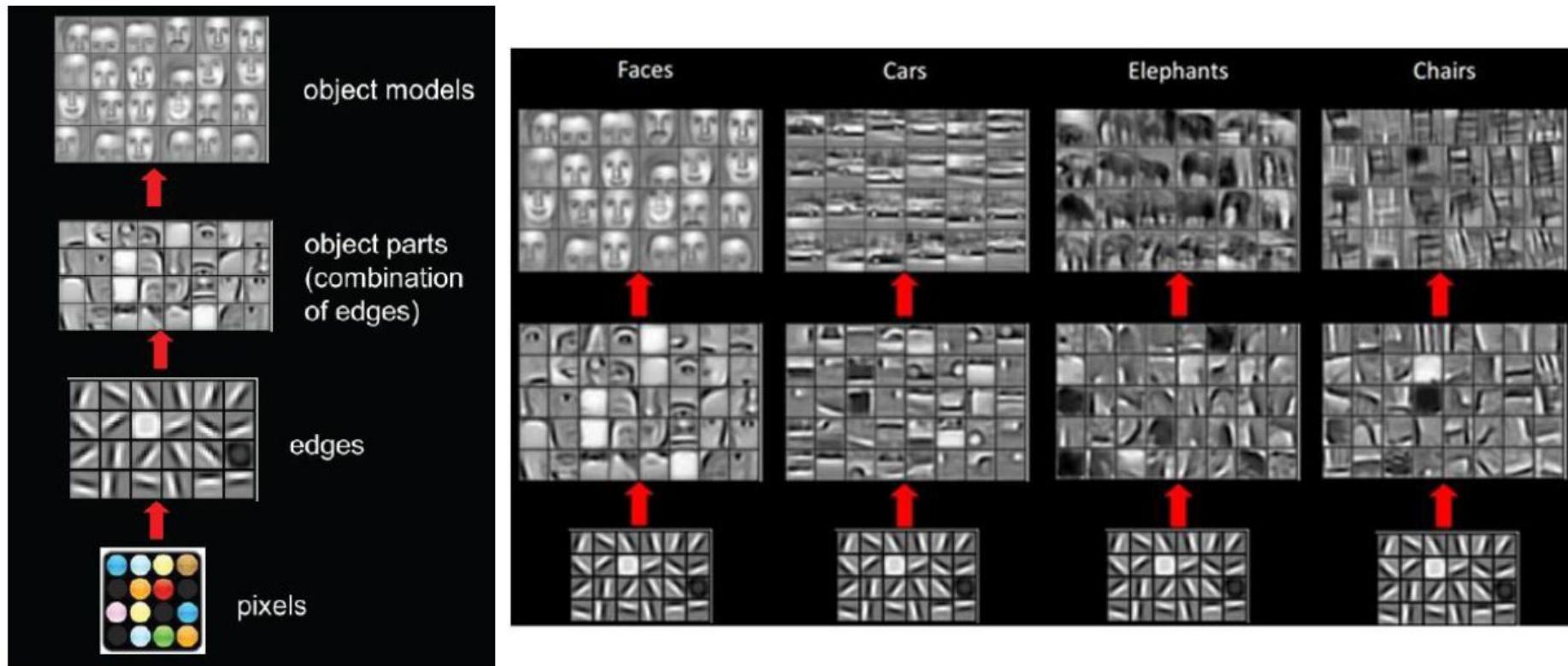
# Redes Neurais Artificiais Profundas

- ▶ Múltiplas camadas de **aprendizagem hierarquicamente organizadas**
- ▶ **Redes Convolucionais (CNN):** Camadas convolucionais seguidas por camadas tradicionais (PDI + IA)



# Redes Neurais Artificiais Profundas

- Múltiplas camadas de aprendizagem hierarquicamente organizadas



# Arquitetura do Córtex Visual

---

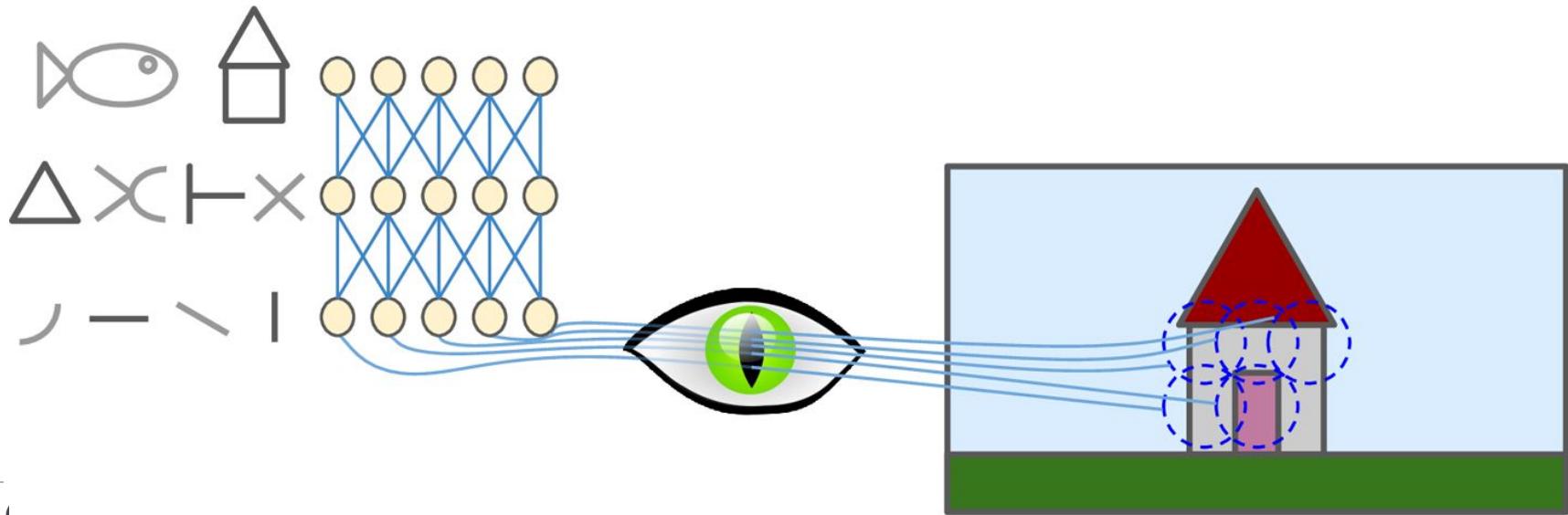
- ▶ Em 1959, David Hubel e Torsten Wiesel realizaram experimentos em gatos que forneceram informações importantes sobre a estrutura do córtex visual<sup>1, 2</sup>.
- ▶ Receberam o Prêmio Nobel de Fisiologia e Medicina em 1981 por seu trabalho.

1 David H. Hubel and Torsten N. Wiesel, "Single Unit Activity in Striate Cortex of Unrestrained Cats", *The Journal of Physiology*, 147, 226-238, 1959

▶ 5 2 David H. Hubel and Torsten N. Wiesel, "Receptive Fields Single Neurons in the Cat's Striate Cortex", *The Journal of Physiology*, 148, 574-591, 1959

# Arquitetura do Córtex Visual

- ▶ Eles demonstraram que muitos neurônios no córtex visual tem **campo receptivo local**
  - ▶ Reagem apenas a estímulos visuais localizados em uma região limitada do campo visual.
- ▶ **Campos receptivos de diferentes neurônios podem se sobrepor e revestir todo o campo visual.**



# Arquitetura do Córtex Visual

---

- ▶ Outras descobertas:
  - ▶ Alguns neurônios têm campos receptivos maiores e reagem a padrões mais complexos que são combinações dos padrões do nível inferior.
  - ▶ Neurônios de nível superior tomam como base as saídas dos neurônios vizinhos de nível inferior.
  - ▶ Cada neurônio está conectado a apenas alguns vizinhos de nível inferior
- ▶ Esses estudos foram inspiração para o surgimento das Redes Neurais Convolucionais

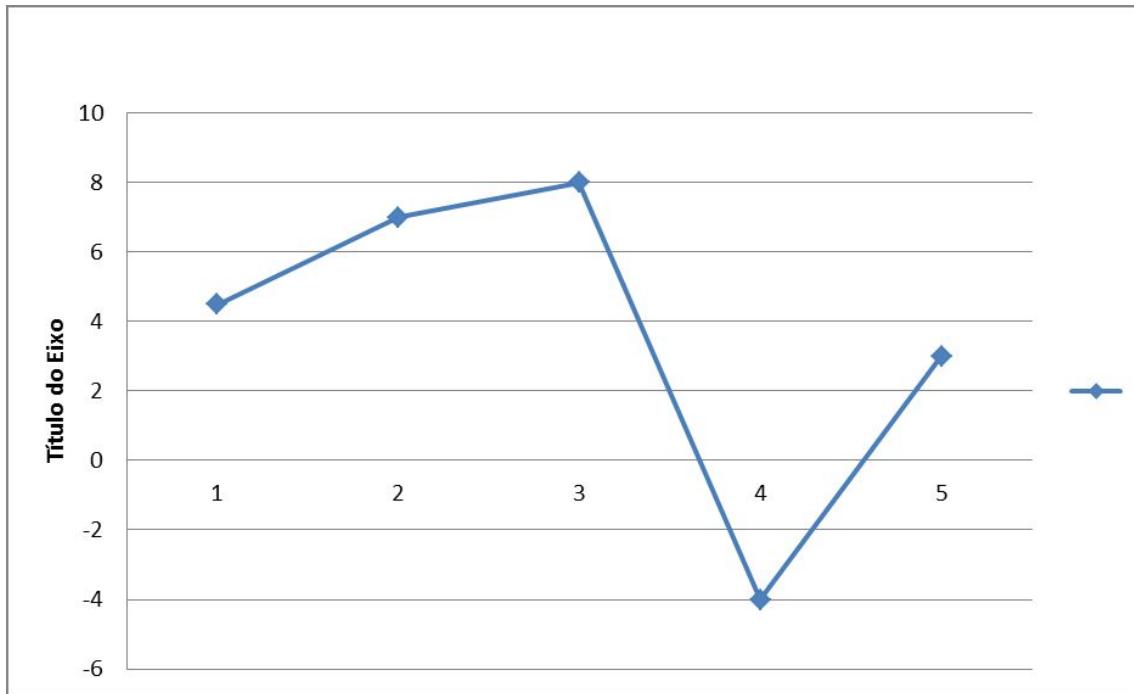
(

---

# Sinais e Imagens Digitais

# Sinais

- ▶ Sinal 1D (Tensor de ordem 1): sequência numérica ou vetor
- ▶ Ex:  $s = (4.5, 7, 8, -4, 3)$ 
  - ▶ Amostra do sinal colhida em  $n = 1$ : 4.5
  - ▶  $s[n]$ :  $s[1] = 4.5, s[2] = 7, s[3] = 8, \dots$



# Sinais

---

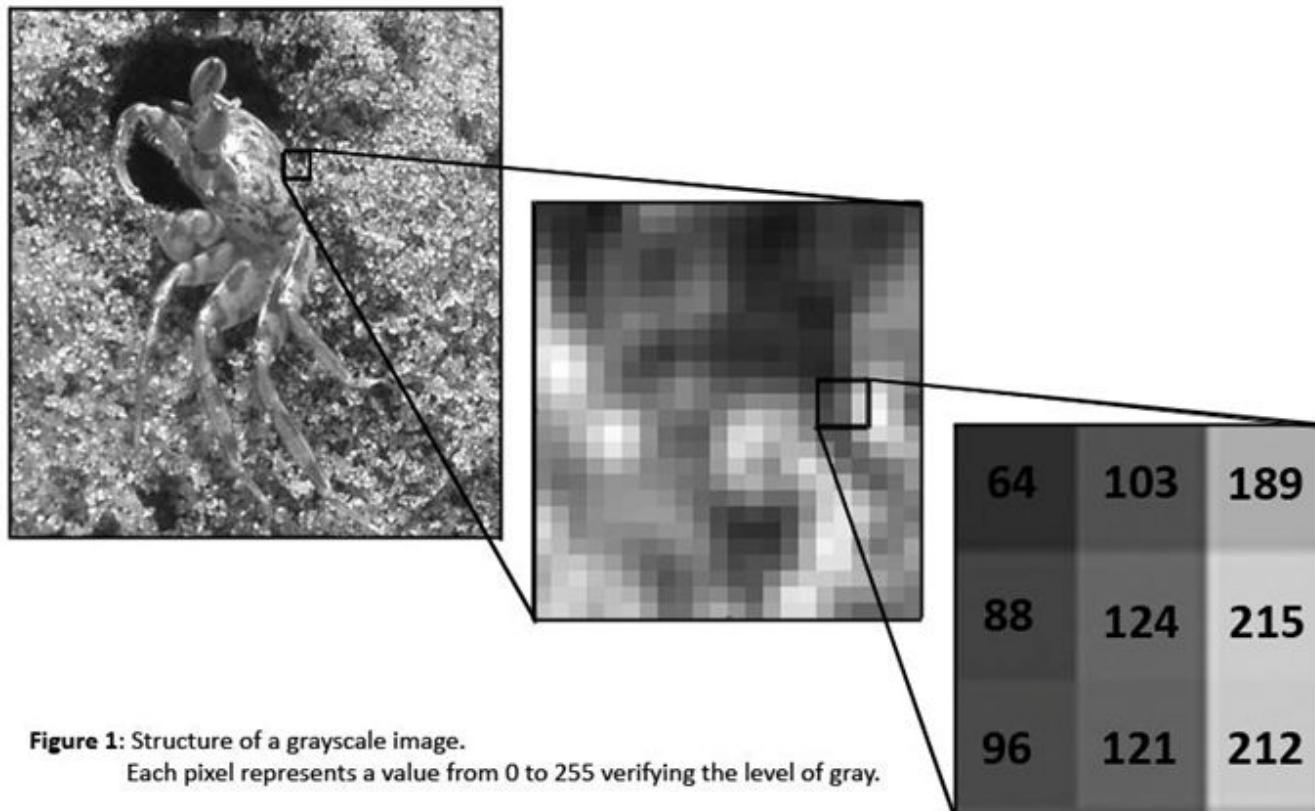
- ▶ Sinal 2D (tensor de ordem 2) ou imagem ou matriz

$$\text{Ex: } I = \begin{bmatrix} 4 & 7 & 6 \\ 2 & 0 & 0 \\ 7 & 1 & 1 \end{bmatrix}$$

- ▶ Pixel na posição (1,1): 4
- ▶  $I(i, j)$ :  $I(1,1) = 4, I(1,2) = 7, I(1,3) = 6, I(2,1) = 2\dots$

# Imagen Digital em Tons de Cinza

- ▶ É uma **matriz de valores numéricos**, onde cada um deles representa um valor de **intensidade de luz quantizado**.
- ▶ Os pontos (valores) de uma imagem são denominados ***pixels***



# Imagens RGB

- ▶ **Imagens coloridas** podem ser geradas a partir da **combinação aditiva** de componentes R, G e B.
  - ▶ 3 matrizes, R, G, B = 1 “matriz 3D” (tensor de ordem 3)

Ex:

$$I_R = \begin{bmatrix} 255 & 128 & 0 \\ 255 & 0 & 0 \\ 128 & 0 & 128 \end{bmatrix}, I_G = \begin{bmatrix} 255 & 128 & 0 \\ 0 & 255 & 0 \\ 128 & 1 & 1 \end{bmatrix}, I_B = \begin{bmatrix} 255 & 128 & 0 \\ 0 & 255 & 0 \\ 7 & 254 & 1 \end{bmatrix}$$

$$I = \begin{bmatrix} (255, 255, 255) & (128, 128, 128) & (0, 0, 0) \\ (255, 0, 0) & (0, 255, 255) & (0, 0, 0) \\ (128, 128, 7) & (0, 1, 254) & (128, 1, 1) \end{bmatrix}$$

Pixel na posição (1,1):  $I(1,1)=(255, 255, 255)$

$I(i,j)$ :  $I(1,1)=(255, 255, 255)$ ,  $I(1,2)=(128, 128, 128)$ ,  
 $I(1,3)=(0, 0, 0)$ ,  $I(2,1)=(255, 0, 0)$ ...

# Imagens RGB

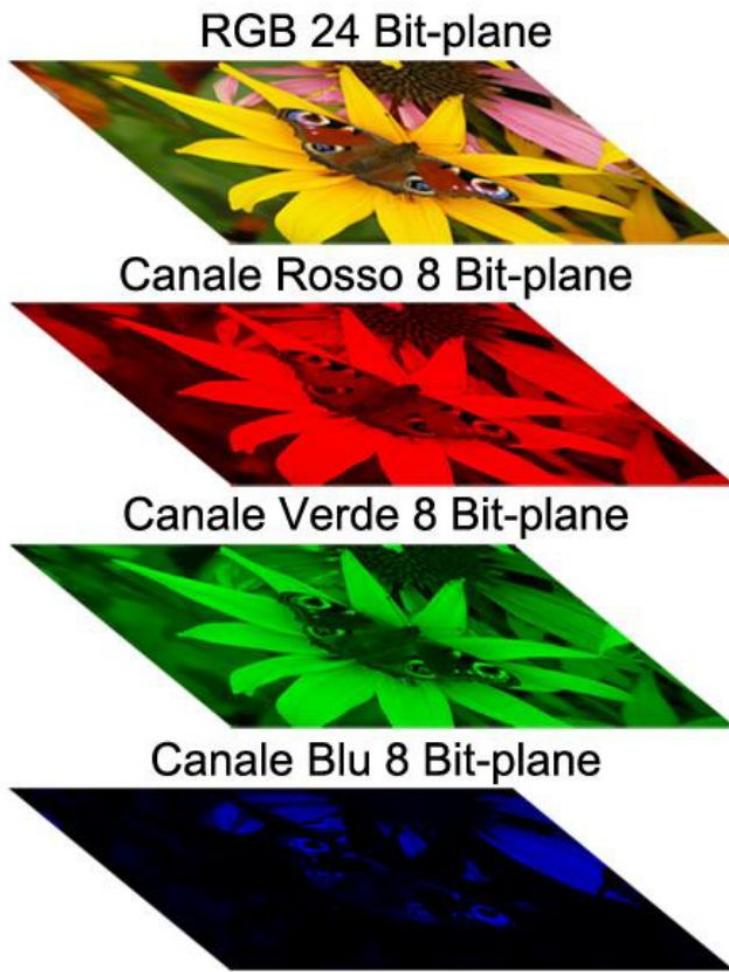


Imagen H x W x 3

Imagenes H x W x C

C: no. de canais, ou profundidade

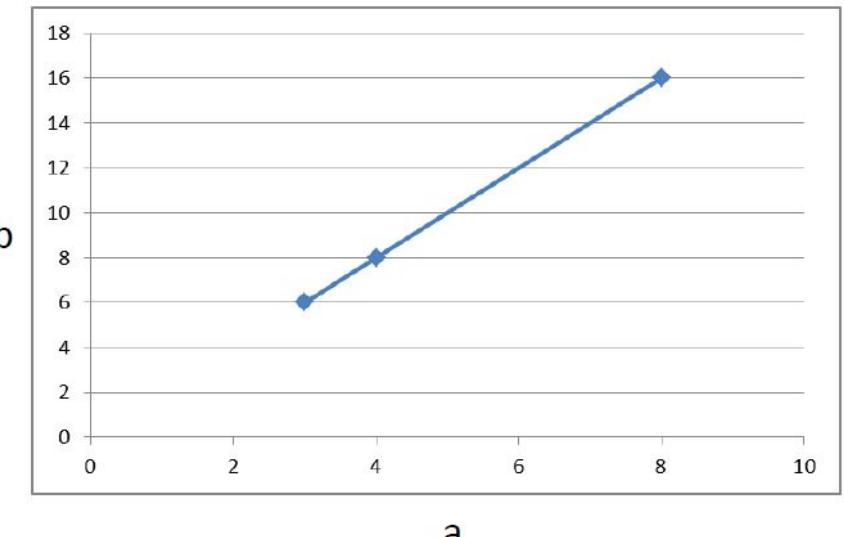
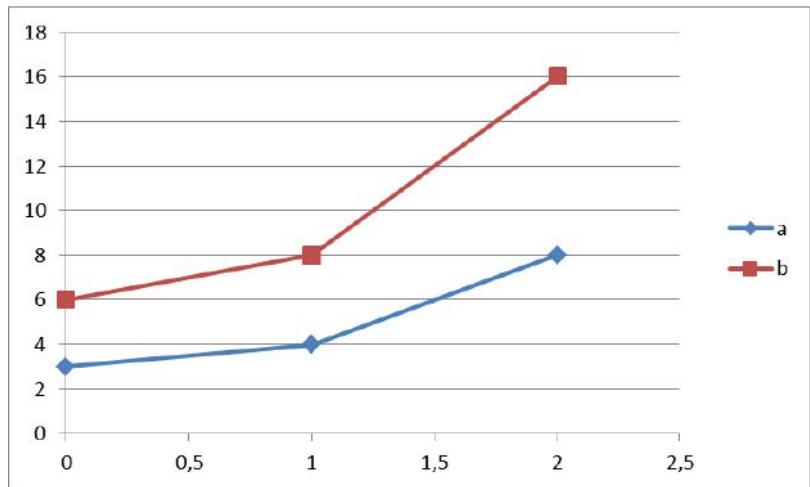
---

# Correlação e Convolução

# Correlação

Tempo	a	b
0	3	6
1	4	8
2	8	16

- Existe alguma correlação entre os sinais 'a' e 'b'?

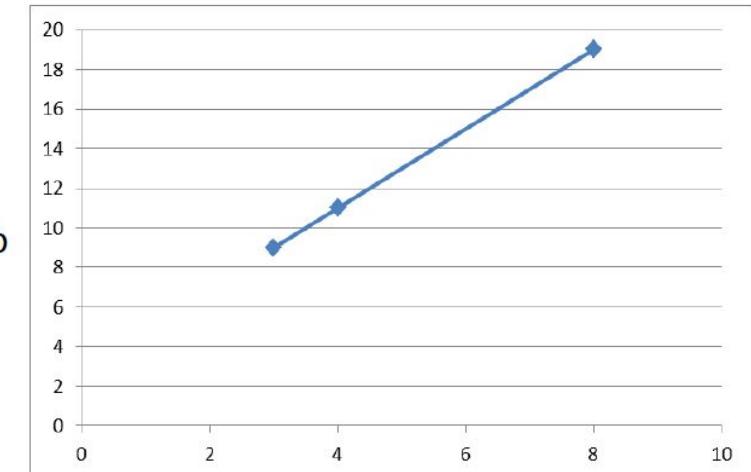
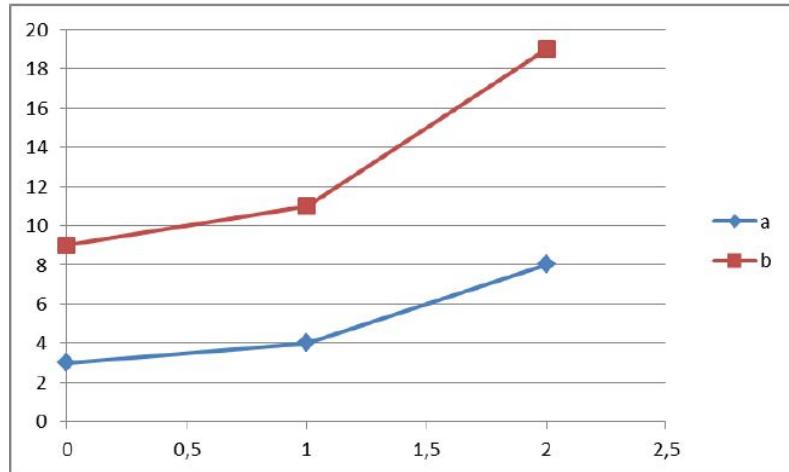


Correlação linear máxima (=1)

# Correlação

Tempo	a	b
0	3	8
1	4	10
2	8	18

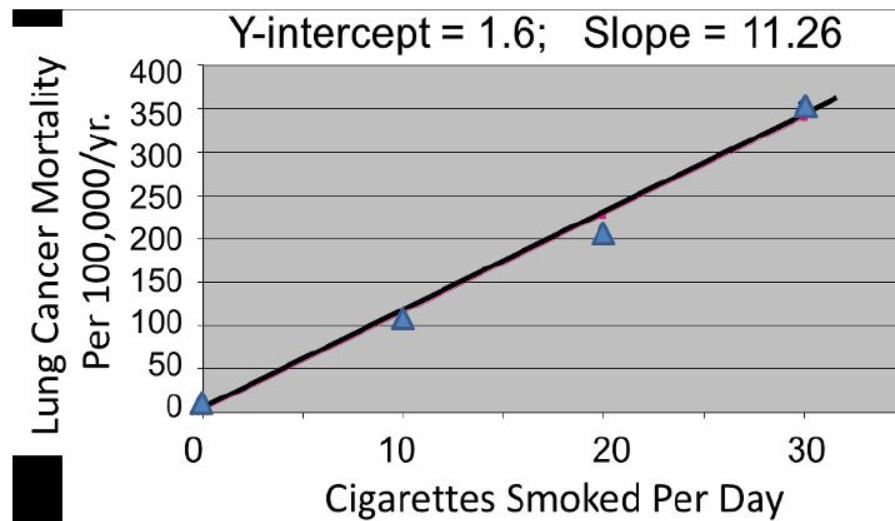
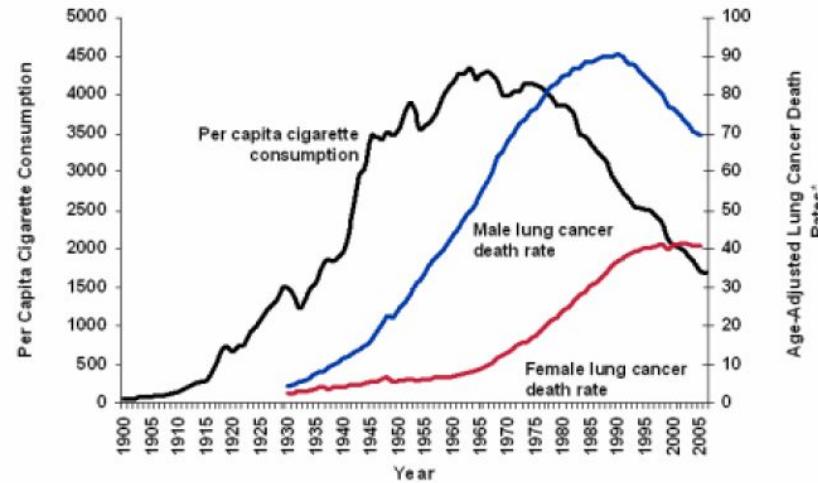
- Existe alguma correlação entre os sinais 'a' e 'b'?



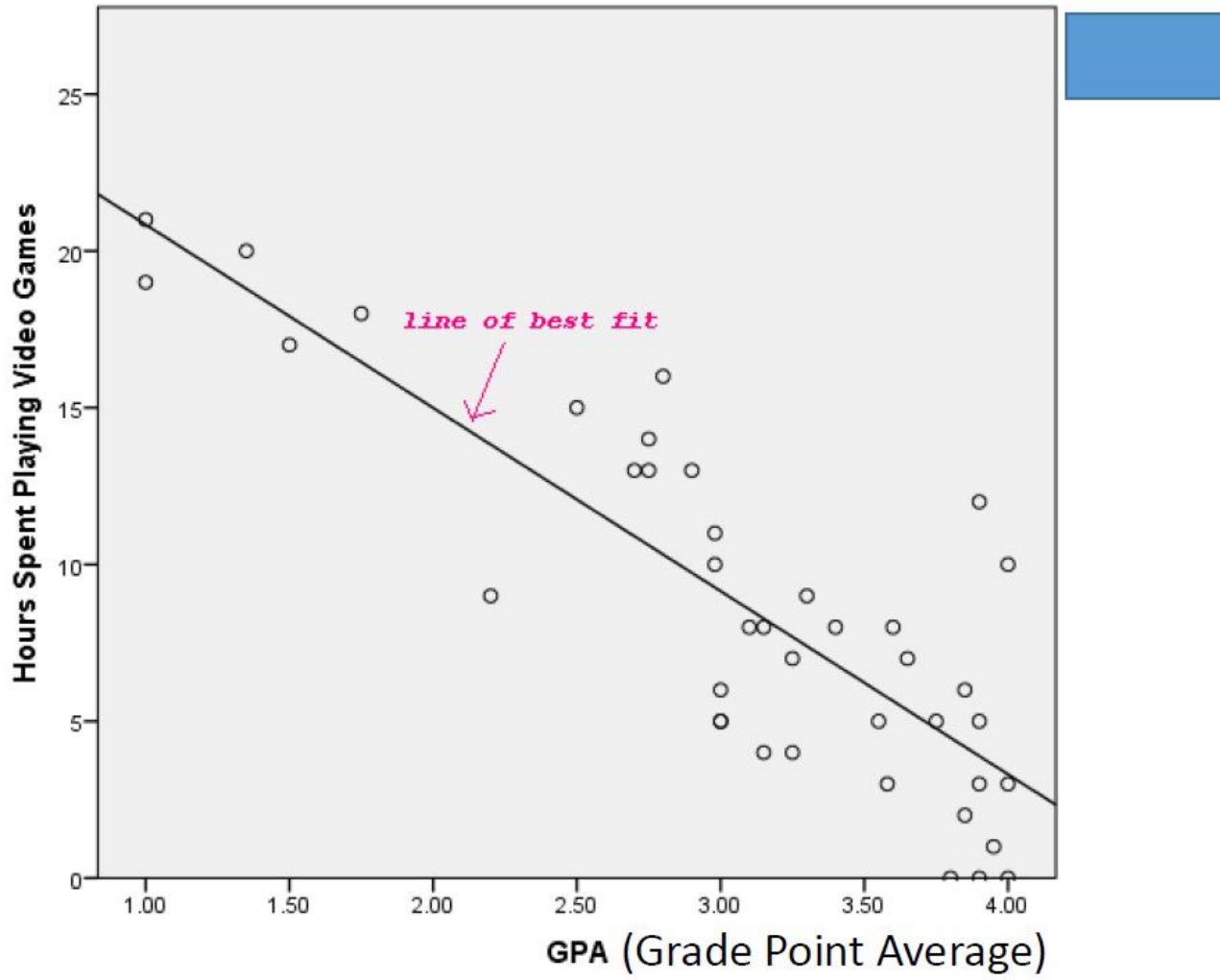
Correlação linear máxima (=1)

# Correlação (Positiva)

Tobacco Use in the US, 1900-2005



# Correlação (Negativa)



# Coeficiente de Correlação

$r = 1$ : correlação máxima

$r = -1$ : correlação negativa máxima

$r = 0$ : correlação nula

Strength of correlation

Perfect	+1	-1
Strong	+0.9	-0.9
Moderate	+0.6	-0.6
Weak	+0.3	-0.3
Zero	0	0

# Coeficiente de Correlação

---

- ▶ Calcular a correlação entre dois sinais é computacionalmente caro.
- ▶ Suponha dois sinais **a** e **b**, o coeficiente de correlação de Pearson (**r**) é calculado da seguinte forma:

$$r = \frac{(a - \mu_a) \cdot (b - \mu_b)}{|a - \mu_a| |b - \mu_b|} =$$

Necessário calcular as médias (*ua* e *ub*) e a norma do sinal.

# Coeficiente de Correlação

---

Ex:  $\mathbf{a} = (1, 2)$  e  $\mathbf{b} = (2, 4) - 4 = (-2, 0)$

$$\mu_a = 1,5 \text{ e } \mu_b = -1$$

$$\mathbf{a} - \mu_a = (-0,5, 0,5) \text{ e } \mathbf{b} - \mu_b = (-1, 1)$$

$$\cos\theta = \frac{0,5+0,5}{\sqrt{0,5}\sqrt{2}} = 1$$

# Correlação Normalizada

- ▶ Imagine duas sequências numéricas,  $s$  e  $w$ 
  - ▶ Sinal  $s$ , máscara  $w$  (filtro correlacional)

$g = s \bullet w$  (correlação normalizada entre  $s$  e  $w$ )

Ex:  $w = (3, 7, 5)$

$s = (4, 1, 3, 8, 4, 0, 3, 8, 0, 7, 7, 7, 1, 2)$

$s$	4	1	3	8	4	0	3	8	0	7	7	7	1	2
$w$	3	7	5											
$g$		?												

# Correlação Normalizada

s	4	1	3	8	4	0	3	8	0	7	7	7	1	2
w	3	7	5											
g		-.98												

s	4	1	3	8	4	0	3	8	0	7	7	7	1	2
w		3	7	5										
g		-.98	?											

s	4	1	3	8	4	0	3	8	0	7	7	7	1	2
w														
g		-.98	0.28											

s	4	1	3	8	4	0	3	8	0	7	7	7	1	2	
w													3	7	5
g		-.98	0.28	0.94	-0.5	-.96	0.37	0.62	-.92	0.87	?				

# Correlação Normalizada

---

s	4	1	3	8	4	0	3	8	0	7	7	7	1	2	
w													3	7	5
g		-.98	0.28	0.94	-0.5	-.96	0.37	0.62	-.92	0.87	NaN	0	-.93		

- ▶ **Filtro casado:** resposta máxima na posição do sinal mais “semelhante” (correlacionada com) à máscara

# Correlação Não-Normalizada

- ▶ Uma alternativa viável e computacionalmente mais leve é utilizar a **correlação não-normalizada**

$$g = s \circ w \text{ (correlação entre } s \text{ e } w\text{)}$$

s	4	1	3	8	4	0	3	8	0	7	7	7	1	2
w	3	7	5											
g		34												

$$(3, 7, 5) \cdot (4, 1, 3) = 12 + 7 + 15 = 34$$

# Correlação Não-Normalizada

s	4	1	3	8	4	0	3	8	0	7	7	7	1	2	
w													3	7	5
g		34	64	85	52	27	61	65	59	84	105	75	38		

- ▶ Correlação produz sinal g com dimensão inferior à de s  
 $\dim(g) = \dim(s) - \dim(w) + 1$
- ▶ Para evitar essa redução, geralmente usa-se extensão por zero (zero padding):
  - ▶ Anexar zeros no início e no final de s, conforme necessário

# Correlação Não-Normalizada (Zero Padding)

0	4	1	3	8	4	0	3	8	0	7	7	7	1	2	0
3	7	5											3	7	5
g	33	34	64	85	52	27	61	65	59	84	105	75	38		17

- ▶ Correlação produz sinal  $g$  com dimensão inferior à de  $s$   
$$\dim(g) = \dim(s) - \dim(w) + 1$$
- ▶ Para evitar essa redução, geralmente usa-se extensão por zero (zero padding):
  - ▶ Anexar zeros no início e no final de  $s$ , conforme necessário

# Convolução

- ▶ Sinal  $s$ , máscara  $w$  (filtro convolucional)
- ▶ Convolução entre  $s$  e  $w$ :

$$g = s * w = s \bullet w_r$$

$w_r$  é  $w$  rebatido (“flipado”)

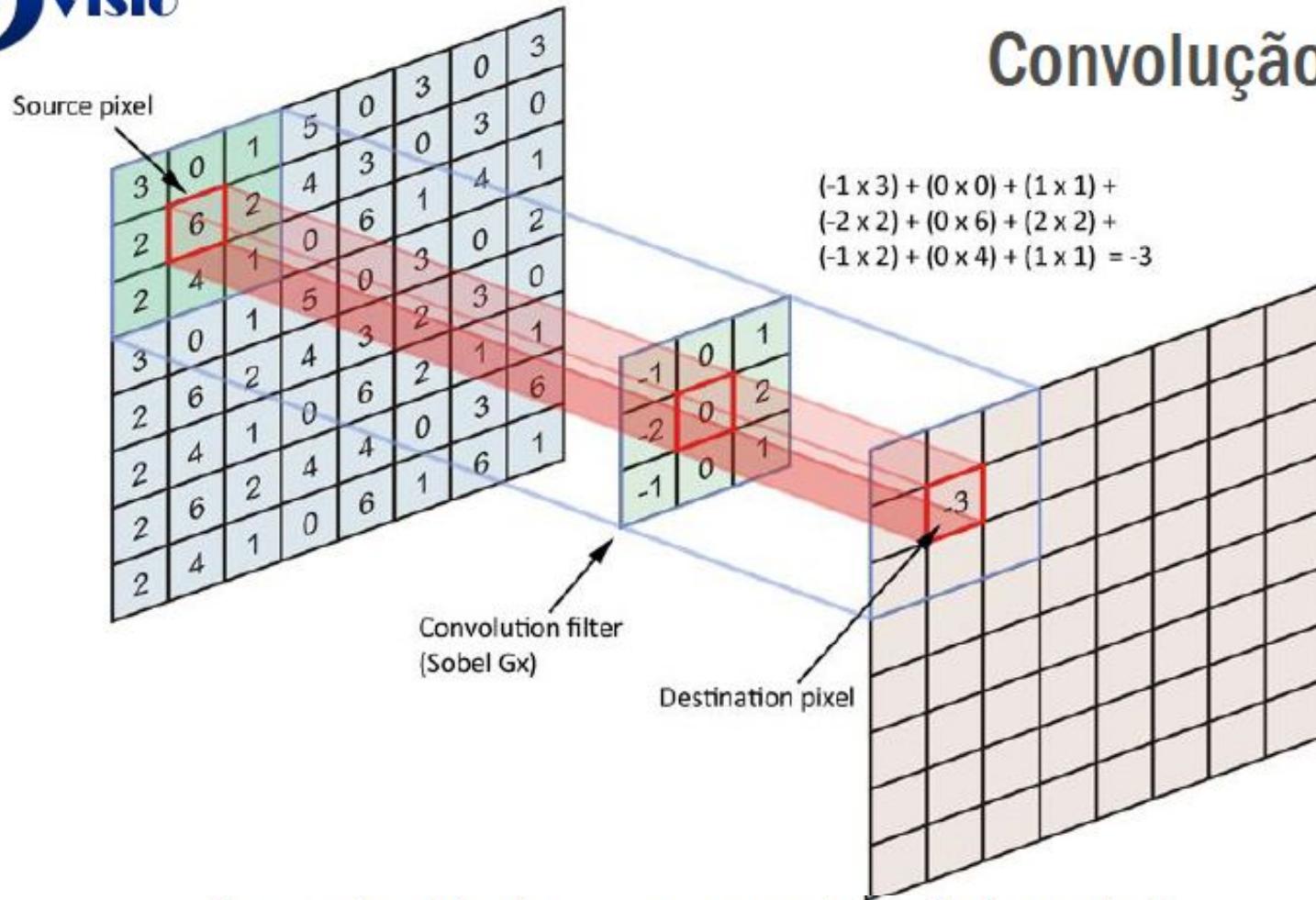
$$w = (h_1, h_2, \dots, h_n)$$

$$w_r = (h_n, h_{n-1}, \dots, h_1)$$

# Convolução 2D



## Convolução 2D



Complexidade computacional elevada!

$$D_w^2 \times D_f^2$$

# Convolução 2D

Ex: **Filtro de Sobel:** Usado para detectar contornos

## Convolução com Filtro de Sobel

10	10	10
10	10	10
10	10	10

0	10	10
0	10	10
0	10	10

10	10	10
0	10	10
0	0	10

10	10	10
10	10	10
0	0	0

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

0

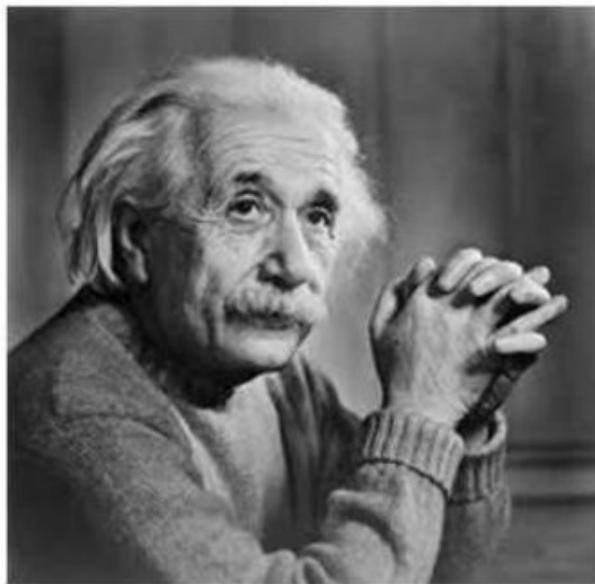
40

30

0

# Convolução 2D

Ex: Aplicação do Filtro de Sobel



Mapa de Ativação

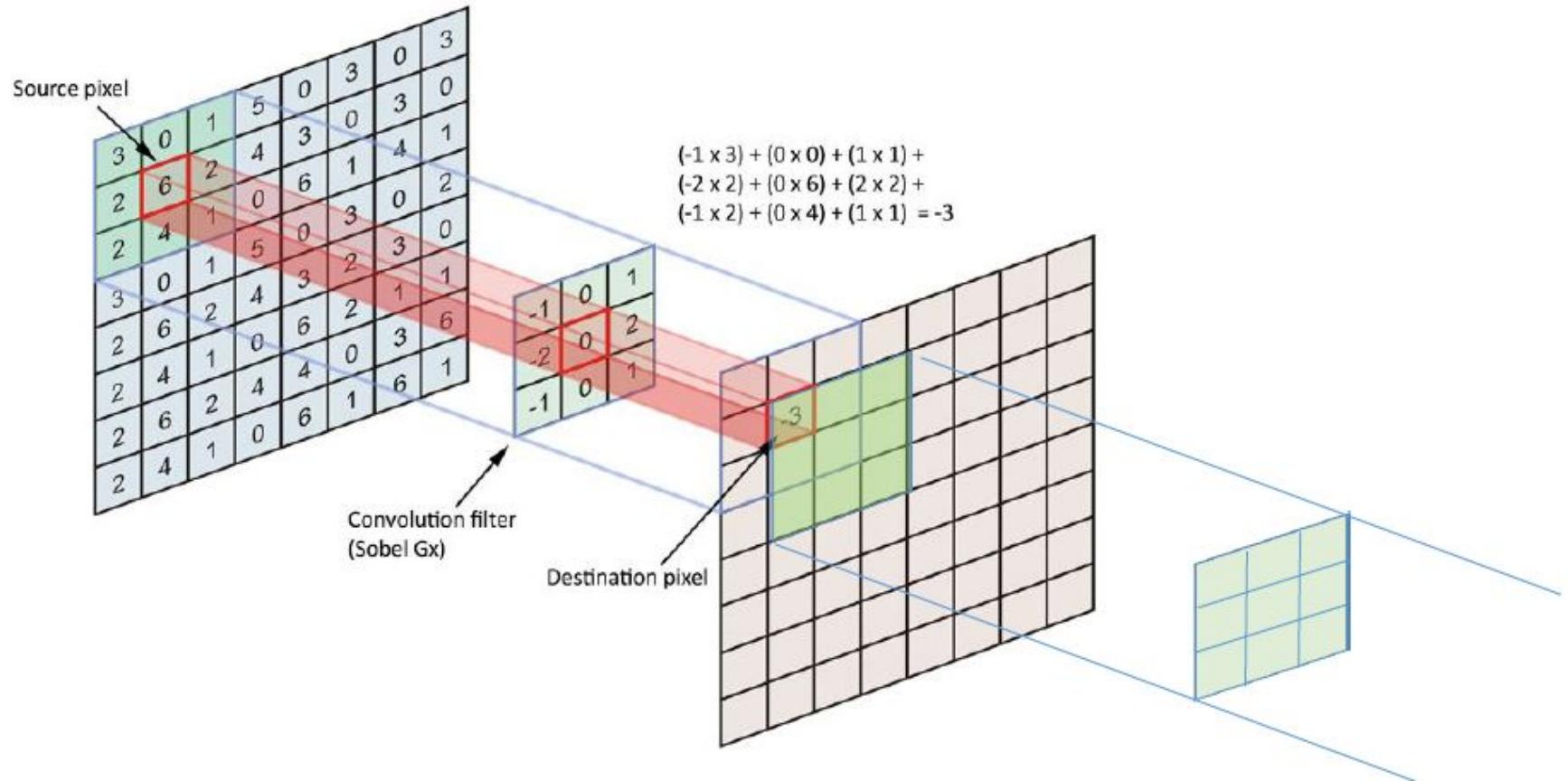


Sobel Gx



Sobel Gy

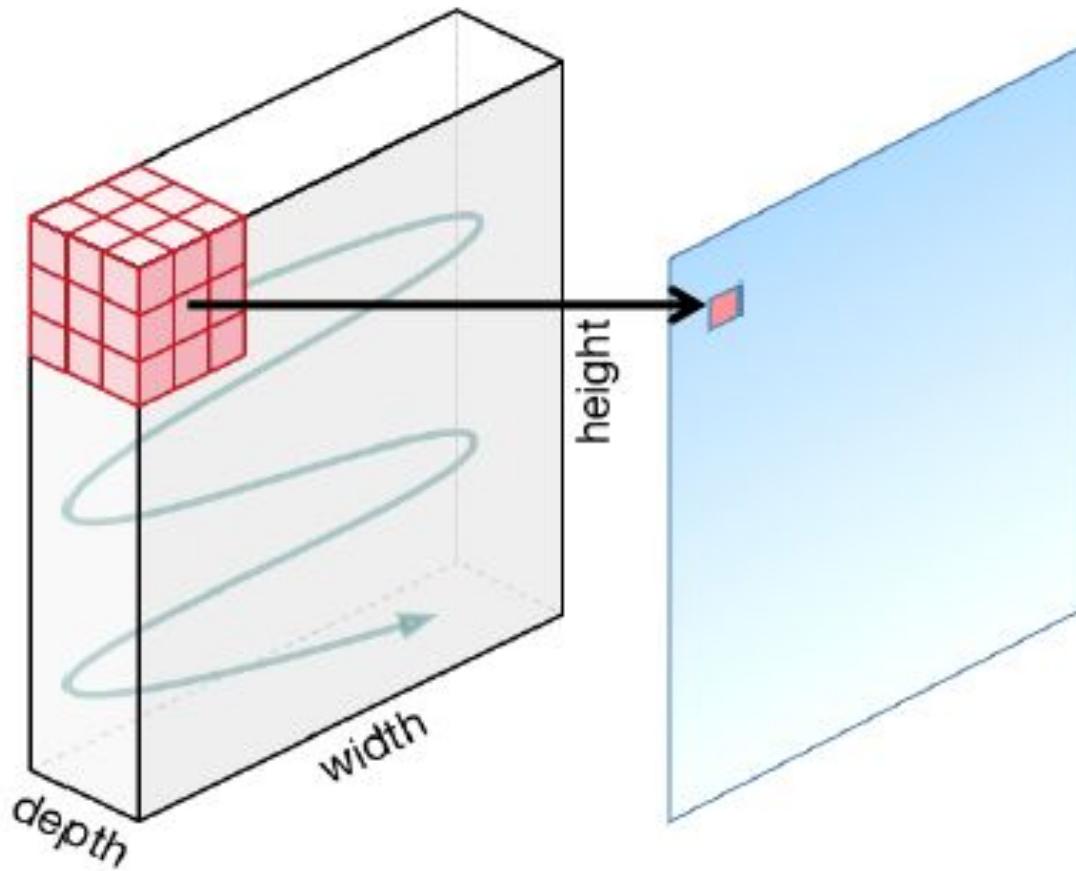
# Convolução de Convolução



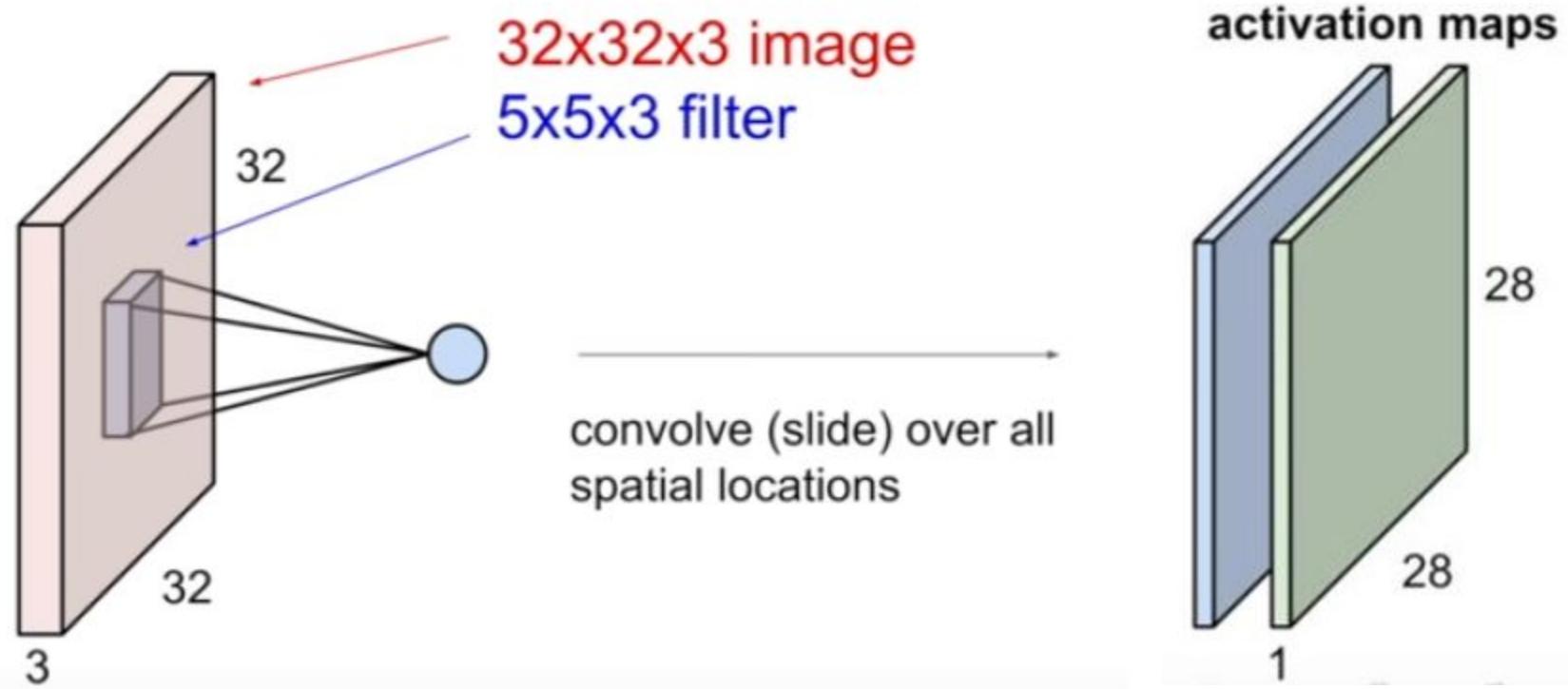
Campo receptivo da segunda máscara, em relação à imagem de entrada?

R: 5x5

# Convolução 3D

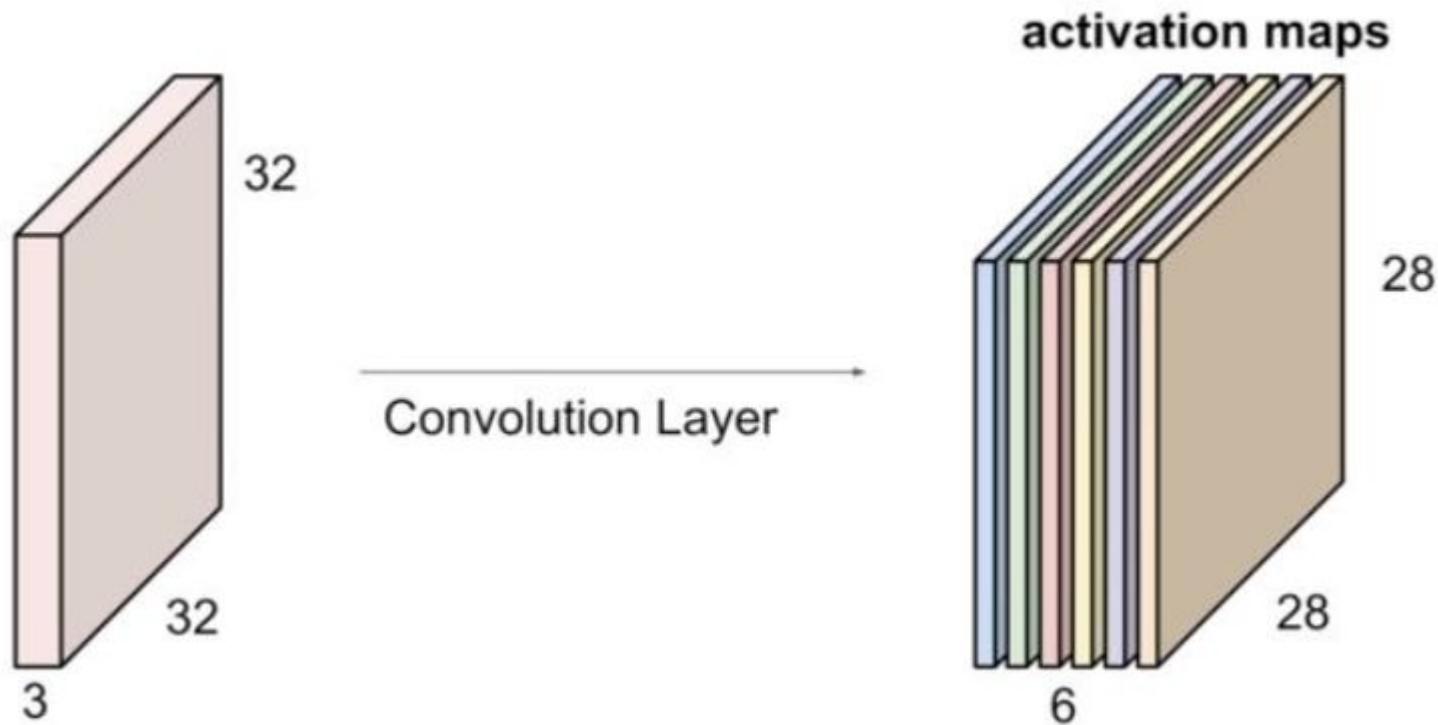


# Convolução 3D

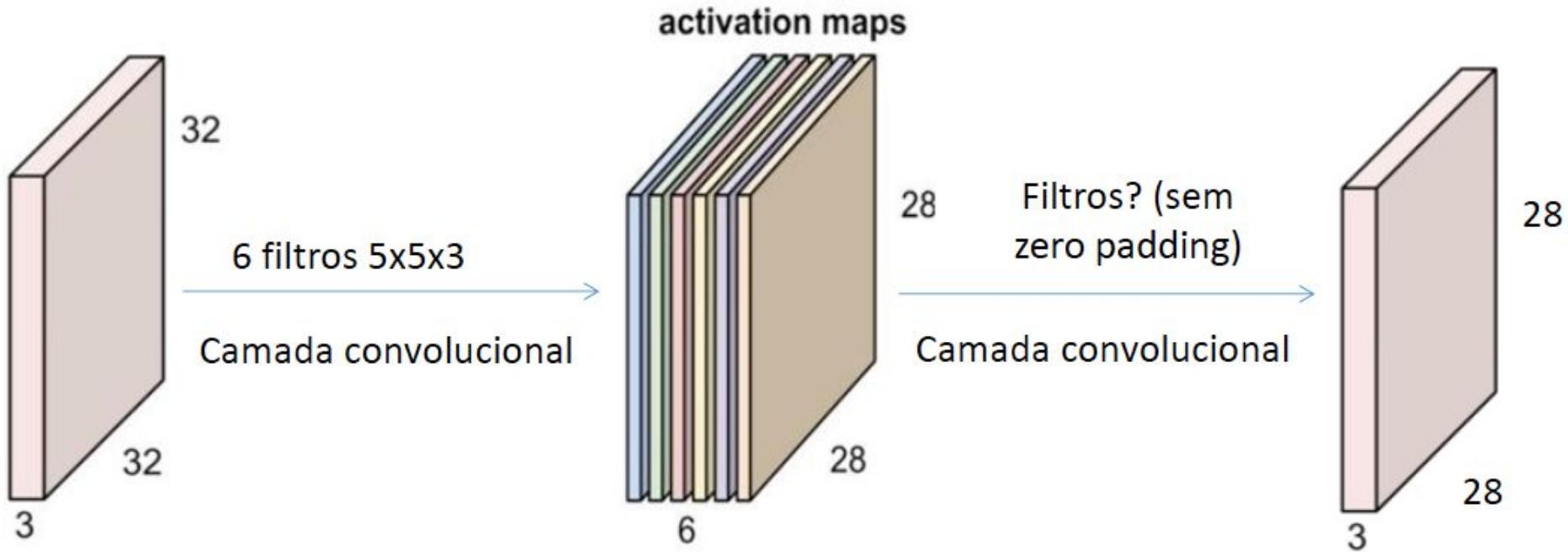


# Convolução 3D

- ▶ Seis filtros  $5 \times 5 \times 3$  -> seis mapas de ativação: volume de dimensões  $28 \times 28 \times 6$



# Convolução 3D



- ▶ Quantos filtros são aplicados e qual é o tamanho da máscara?

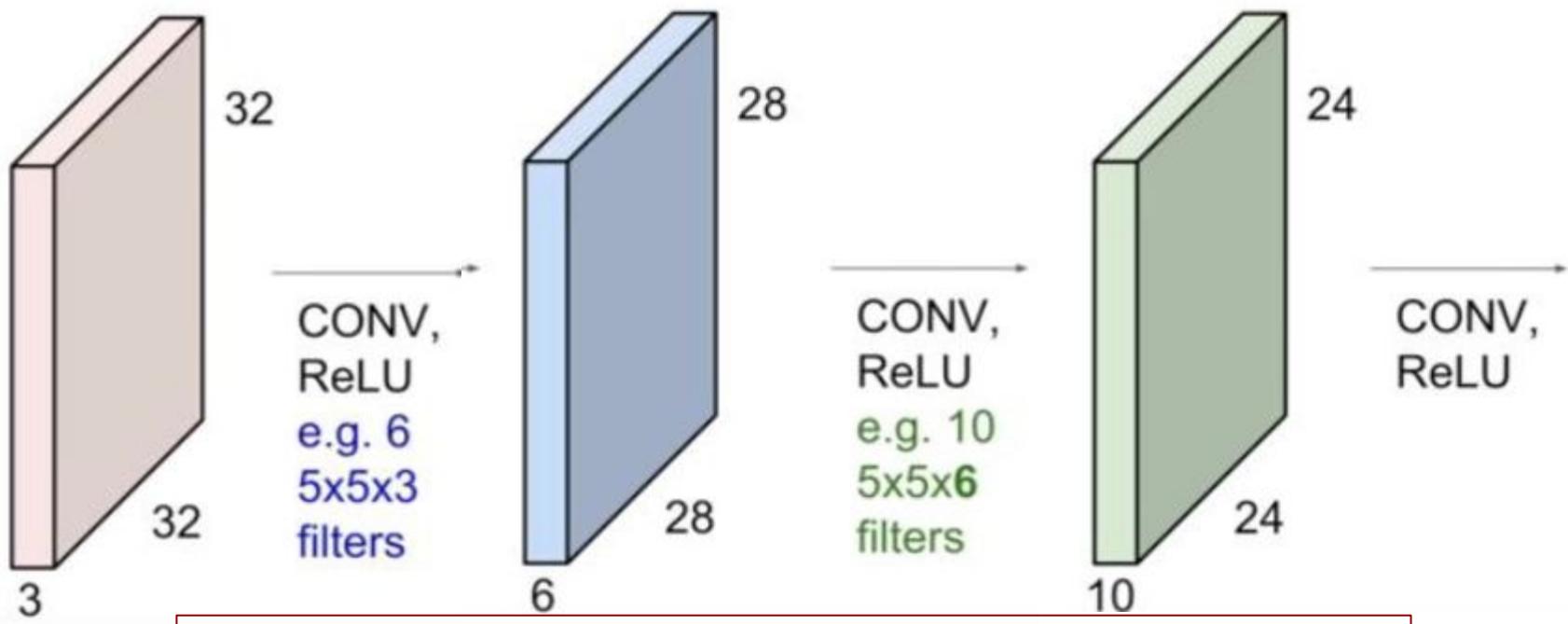
**R: 3 filtros 1x1x6**

)

---

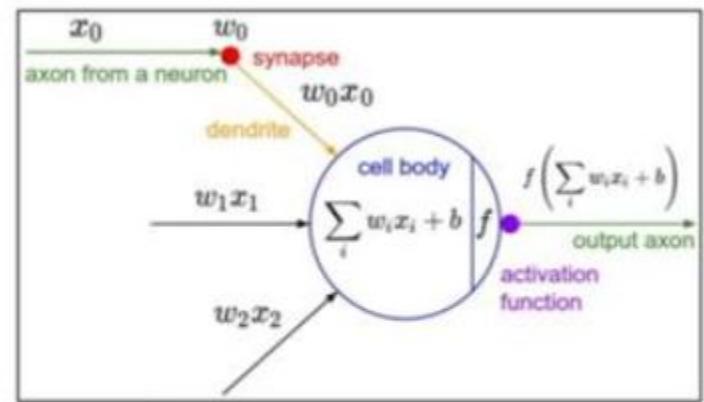
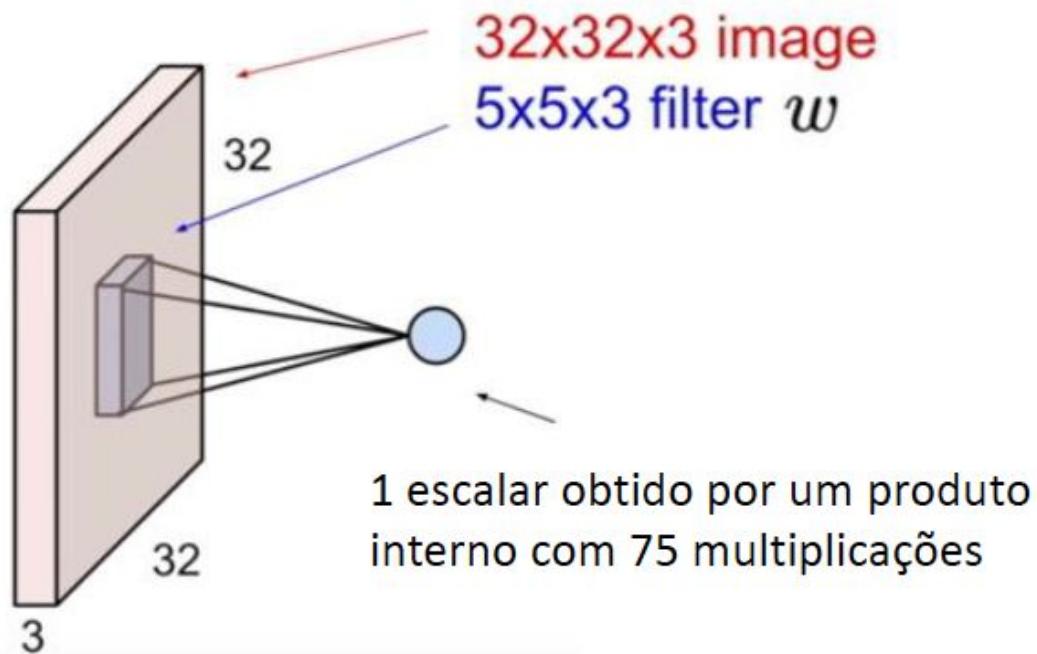
# Redes Neurais Convolucionais

# Redes Neurais Convolucionais (CNNs)



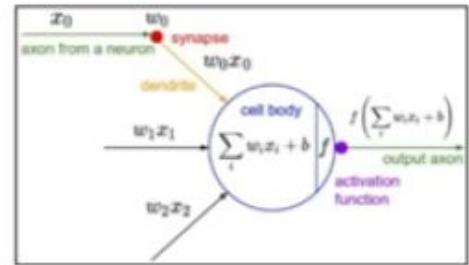
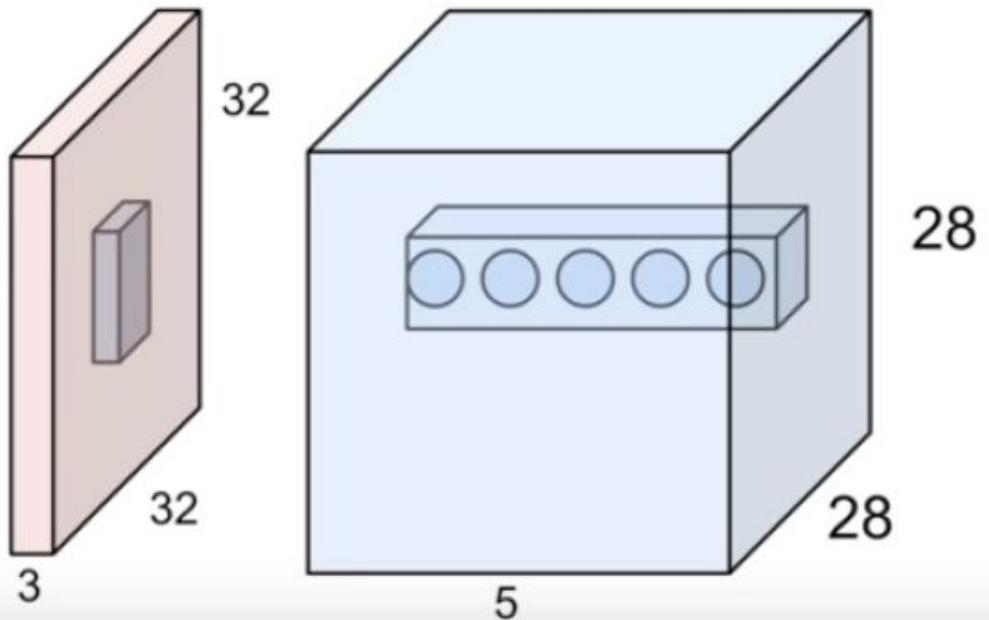
Coeficientes (pesos) de w aprendidos  
(e.g. backpropagation)

# Redes Neurais Convolucionais (CNNs)



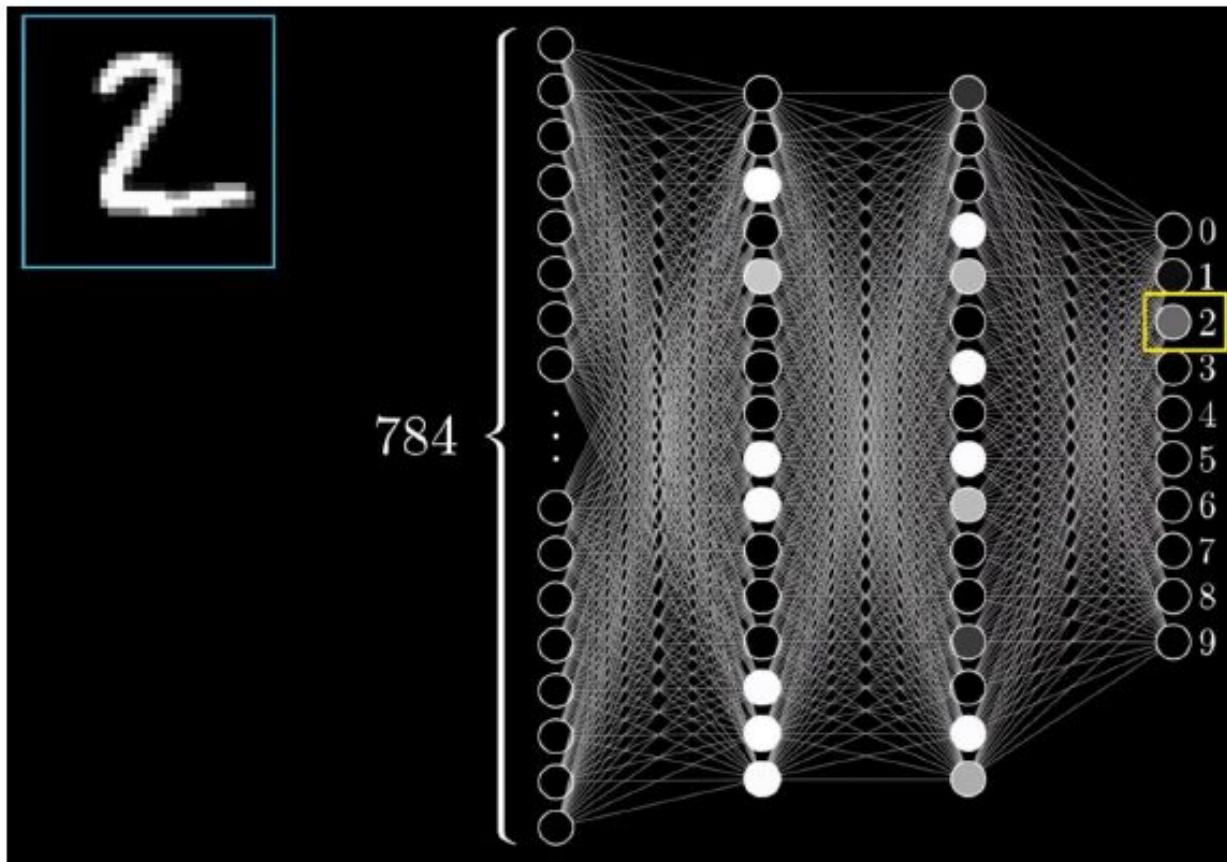
Neurônio com  
conectividade local

# Redes Neurais Convolucionais



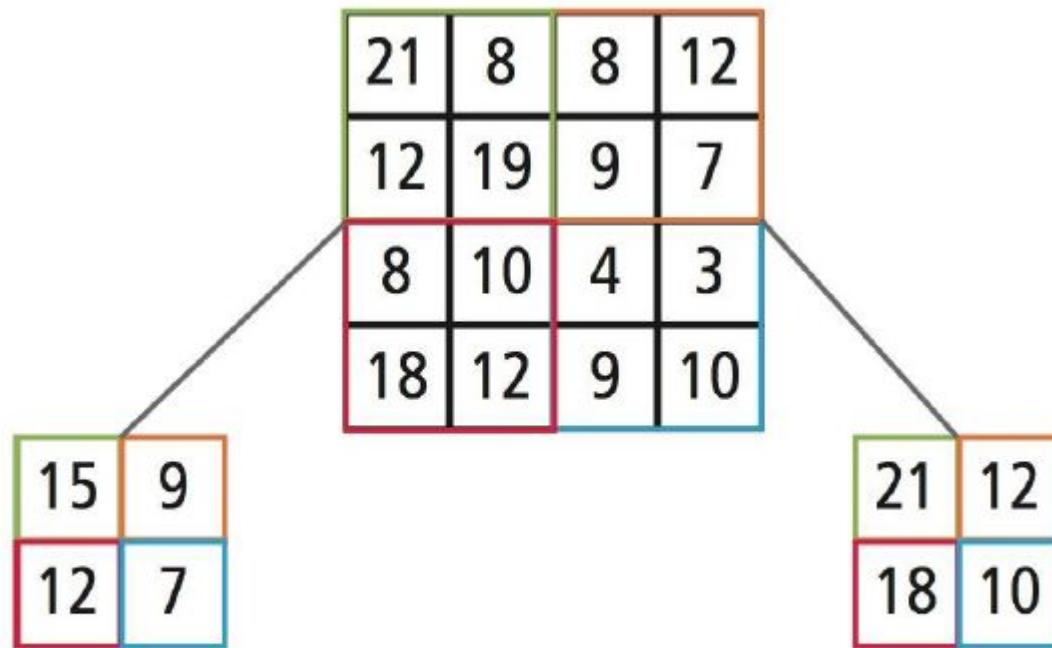
Com cinco filtros,  
haverá cinco  
neurônios “olhando”  
para a mesma região  
no volume de entrada

# CNNs vs ANNs



- ▶ Fully connected:  $16 * 784 = 12544$  pesos na 1<sup>a</sup> camada
- ▶ Convolucional (16 filtros 3x3x1):
  - ▶  $16 * 9 = 144$  pesos na 1<sup>a</sup> camada

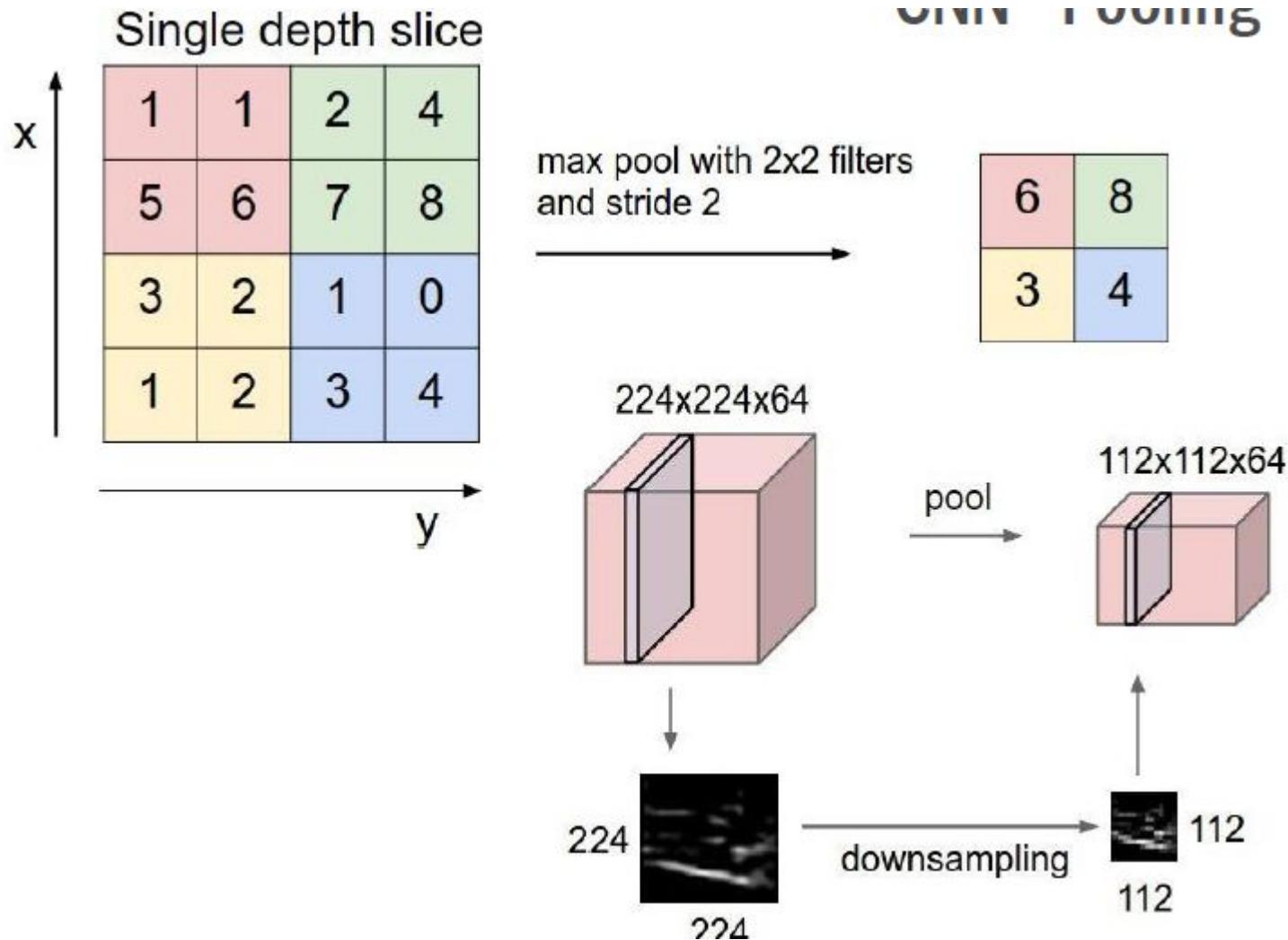
# CNN - Pooling



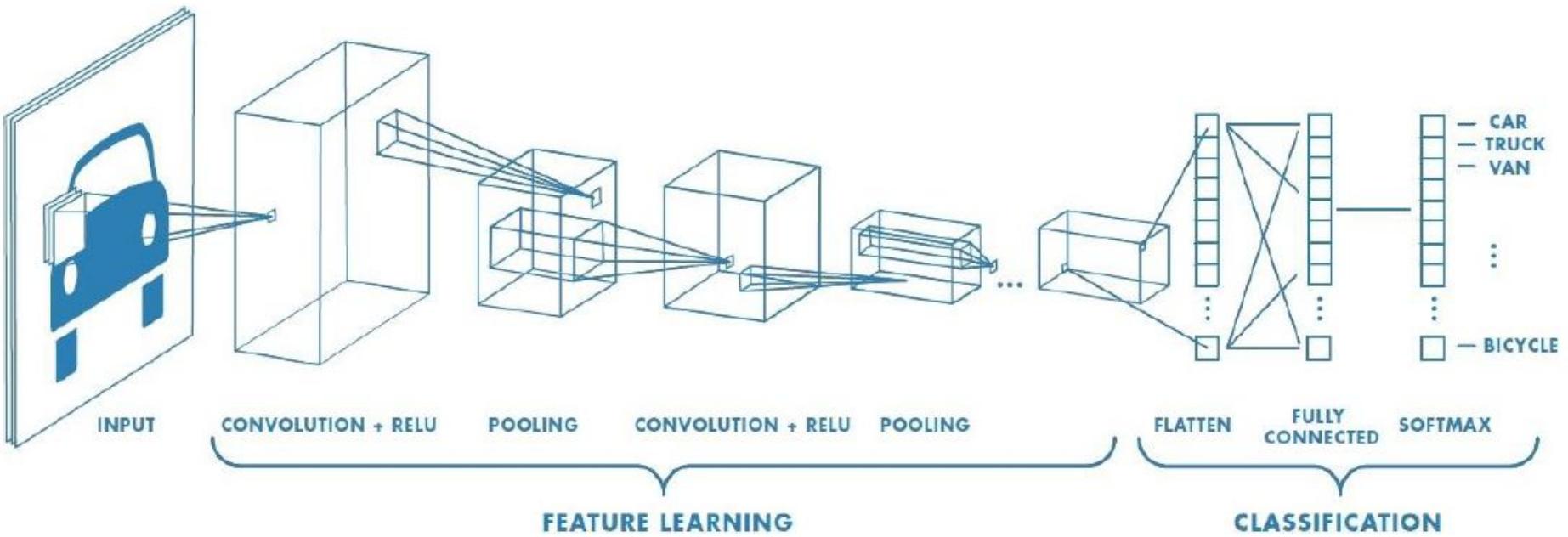
Average Pooling

Max Pooling

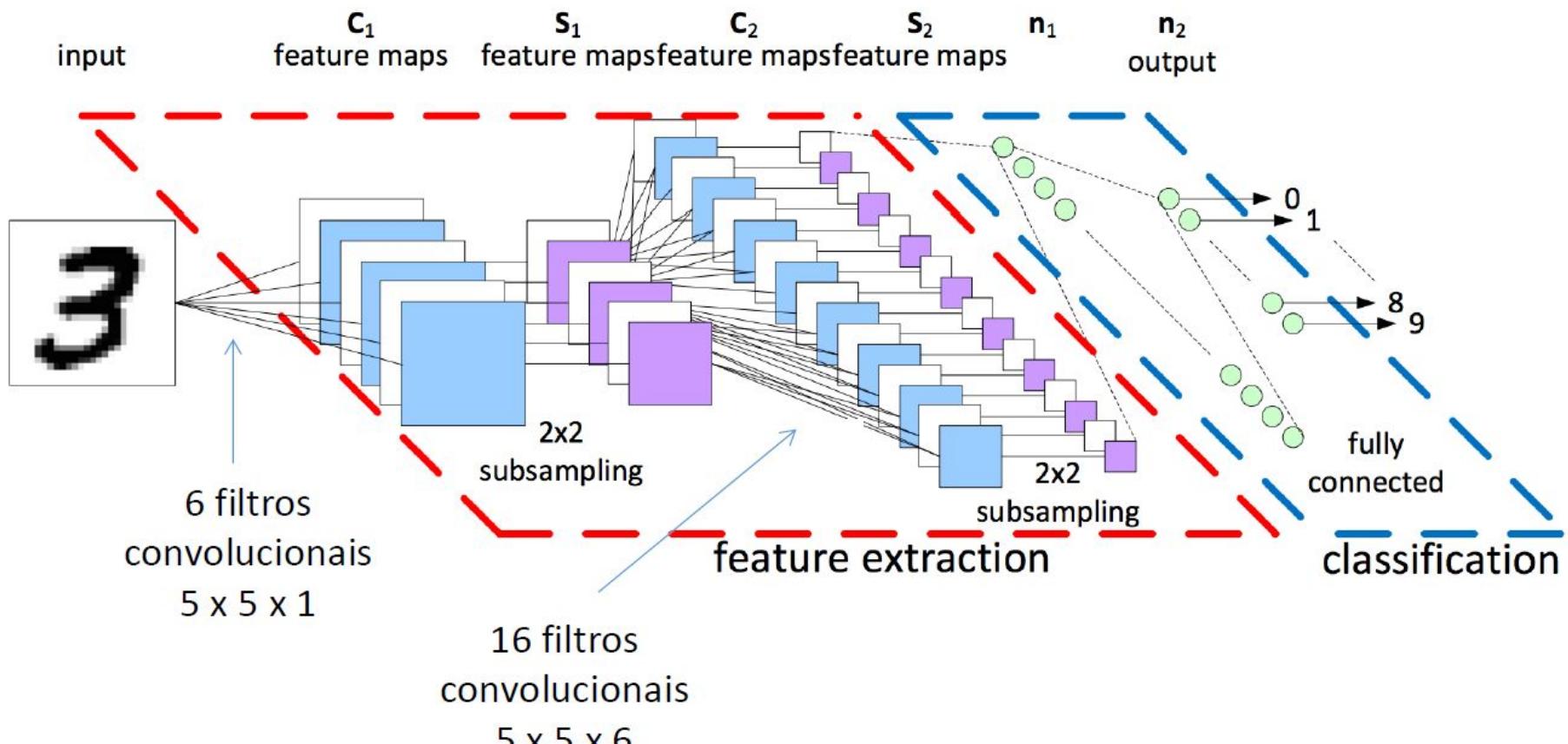
# CNNs - Pooling



# CNNs



# CNN LeNet-5 (1998)



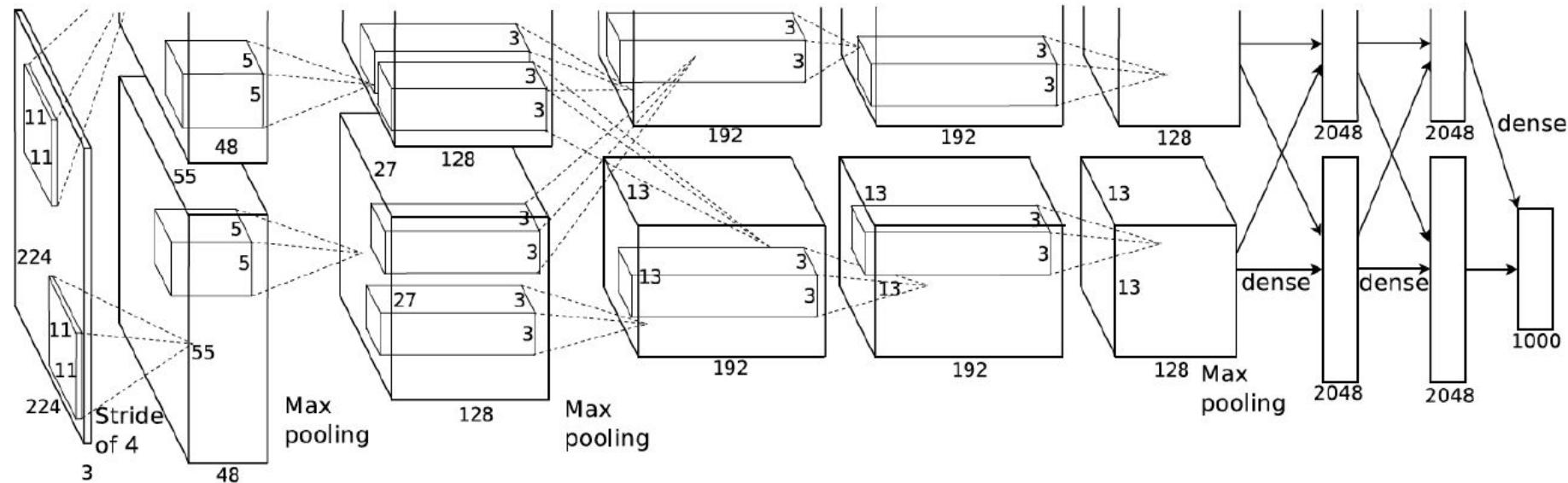
Conv-pool-conv-pool-FC-FC

# CNN LeNet 5 (1998)

---

- ▶ Entrada em níveis de cinza
- ▶ Camada1: 6 filtros  $5 \times 5$ , stride 1, no zero padding, tanh
- ▶ Pooling  $2 \times 2$
- ▶ Camada 2:16 filtros  $5 \times 5 \times 6$ , stride 1, no zero padding, tanh
- ▶ Pooling  $2 \times 2$

# CNN – AlexNet (2012)



Conv1: 96 11x11 filtros, stride 4, no zero padding  
... (8 camadas)

Treinada em GPUs GTX 580 com 3 GB de RAM

# Rede dividida entre duas GPUs

# Primeira CNN vencedora na ILSVRC

## CNN - AlexNet (2012)

## Case Study: AlexNet

[Krizhevsky et al. 2012]

Full (simplified) AlexNet architecture:

[227x227x3] INPUT

[55x55x96] CONV1: 96 11x11 filters at stride 4, pad 0

[27x27x96] MAX POOL1: 3x3 filters at stride 2

[27x27x96] NORM1: Normalization layer

[27x27x256] CONV2: 256 5x5 filters at stride 1, pad 2

[13x13x256] MAX POOL2: 3x3 filters at stride 2

[13x13x256] NORM2: Normalization layer

[13x13x384] CONV3: 384 3x3 filters at stride 1, pad 1

[13x13x384] CONV4: 384 3x3 filters at stride 1, pad 1

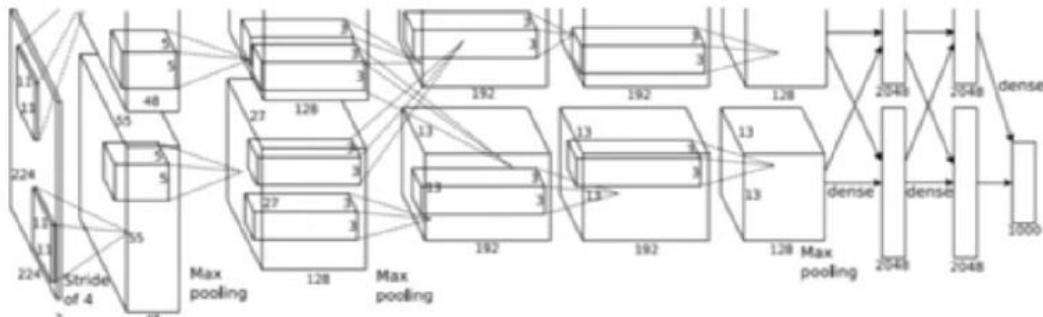
[13x13x256] CONV5: 256 3x3 filters at stride 1, pad 1

[6x6x256] MAX POOL3: 3x3 filters at stride 2

[4096] FC6: 4096 neurons

[4096] FC7: 4096 neurons

[1000] FC8: 1000 neurons (class scores)

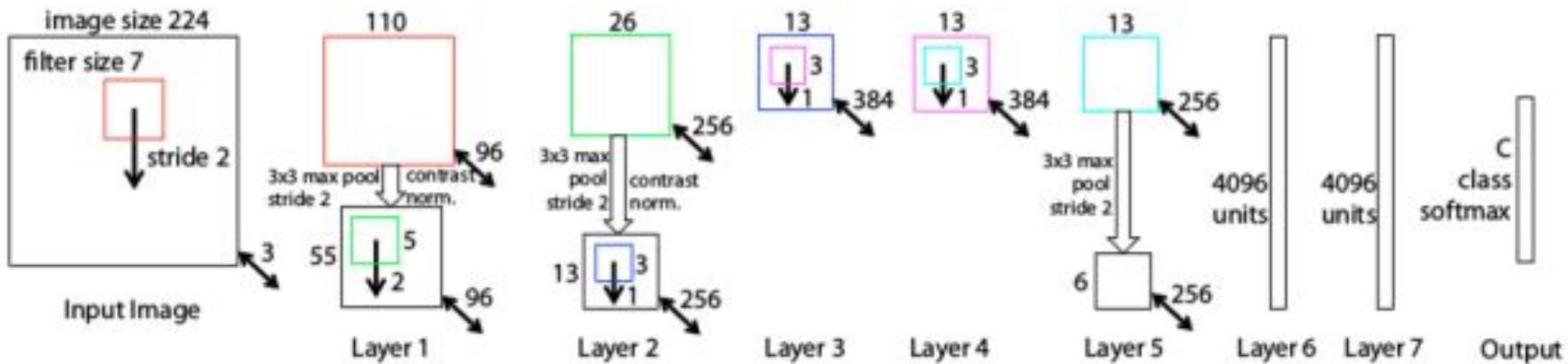


## Details/Retrospectives:

- first use of ReLU
- used Norm layers (not common anymore)
- heavy data augmentation
- dropout 0.5
- batch size 128
- SGD Momentum 0.9
- Learning rate 1e-2, reduced by 10 manually when val accuracy plateaus
- L2 weight decay 5e-4
- 7 CNN ensemble: 18.2% -> 15.4%

# CNN ZF-Net (2013)

- ▶ Semelhante à AlexNet, mas...
- ▶ Conv1: filtros 7x7, stride 2
- ▶ Convs 3, 4, 5: 512, 1024 e 512 filtros, respectivamente



ZF Net Architecture

# Case Study: VGGNet

[Simonyan and Zisserman, 2014]

Small filters, Deeper networks

8 layers (AlexNet)

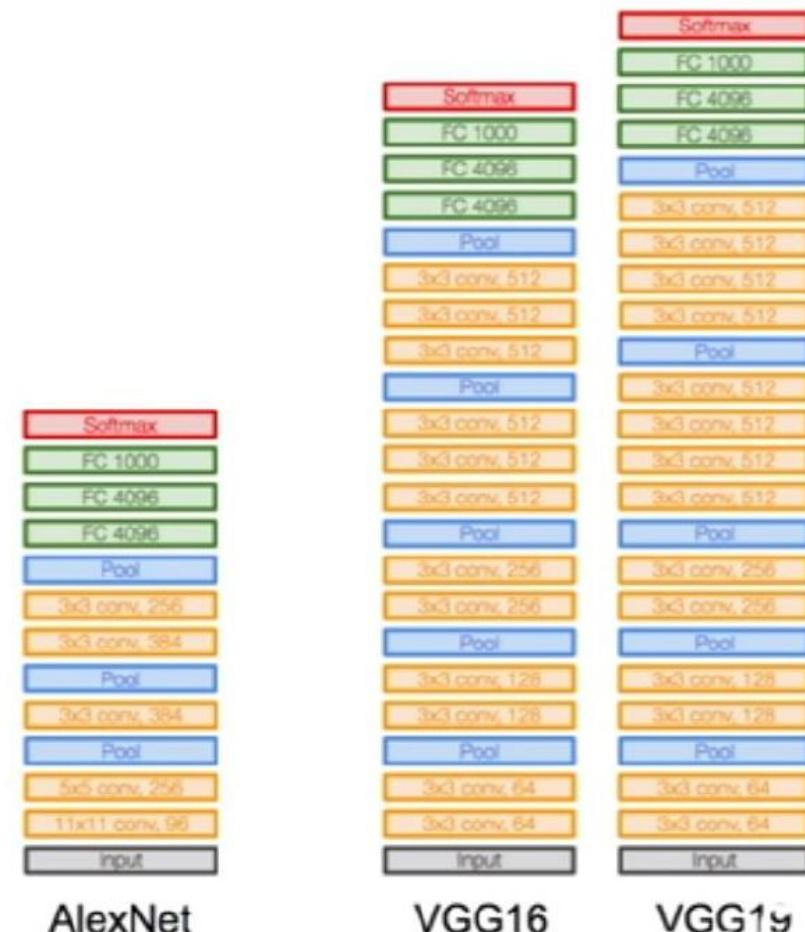
-> 16 - 19 layers (VGG16Net)

Only 3x3 CONV stride 1, pad 1  
and 2x2 MAX POOL stride 2

11.7% top 5 error in ILSVRC'13

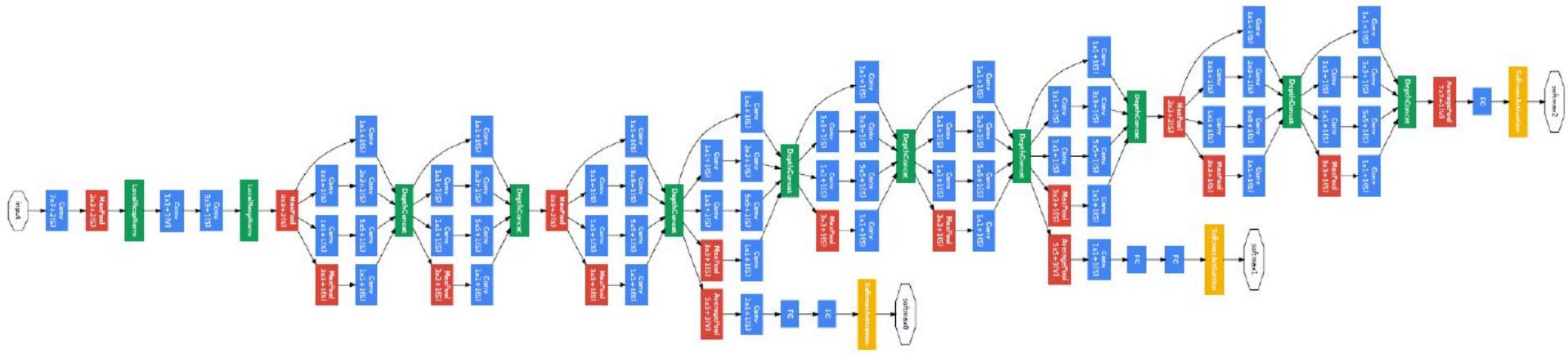
(ZFNet)

-> 7.3% top 5 error in ILSVRC'14



Não venceu o desafio em 2014, mas ficou famosa por ser muito simples e baseada na AlexNet!!!

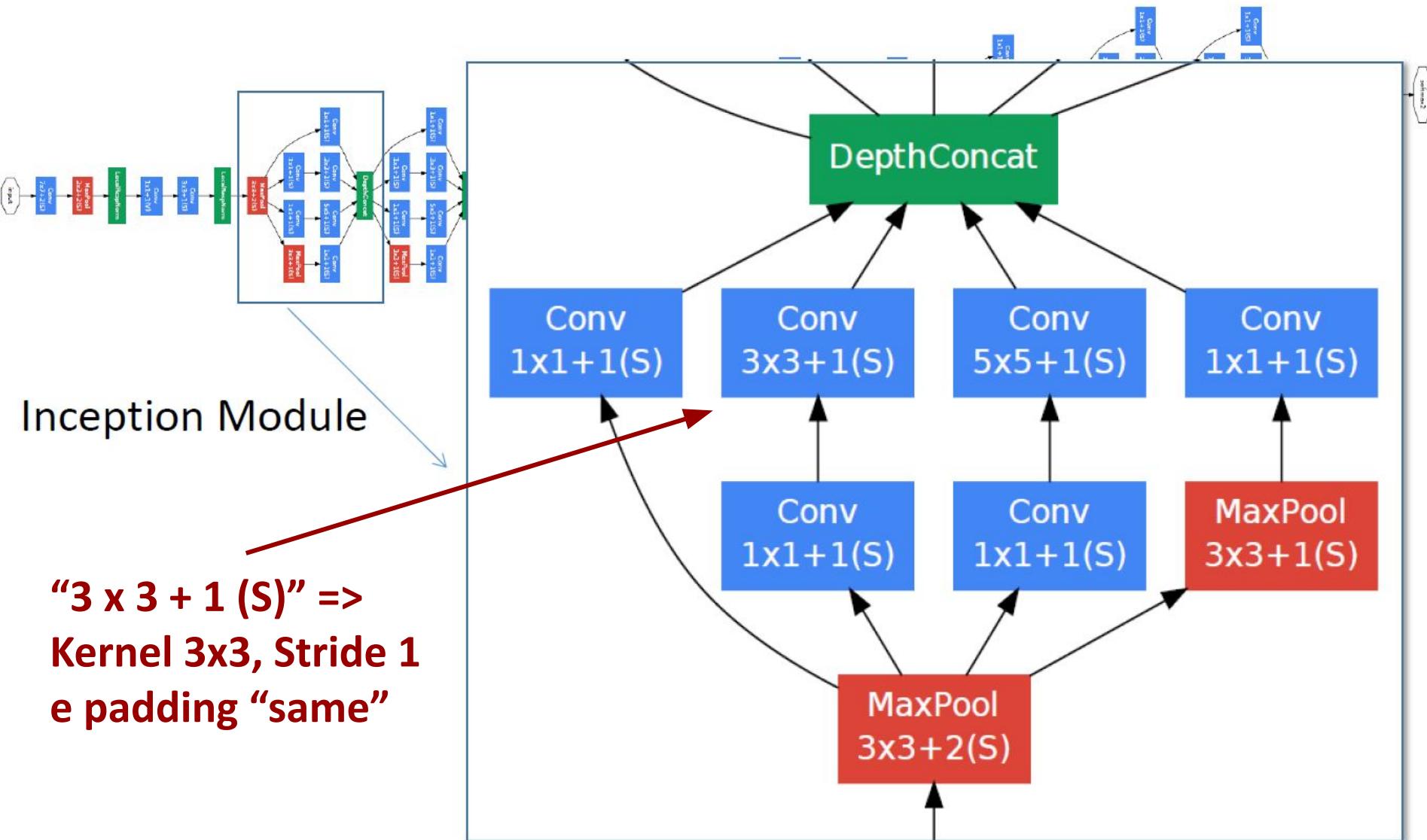
# CNN – GoogLeNet (2014)



<b>Convolution</b>	<b>Pooling</b>	<b>Softmax</b>	<b>Other</b>
--------------------	----------------	----------------	--------------

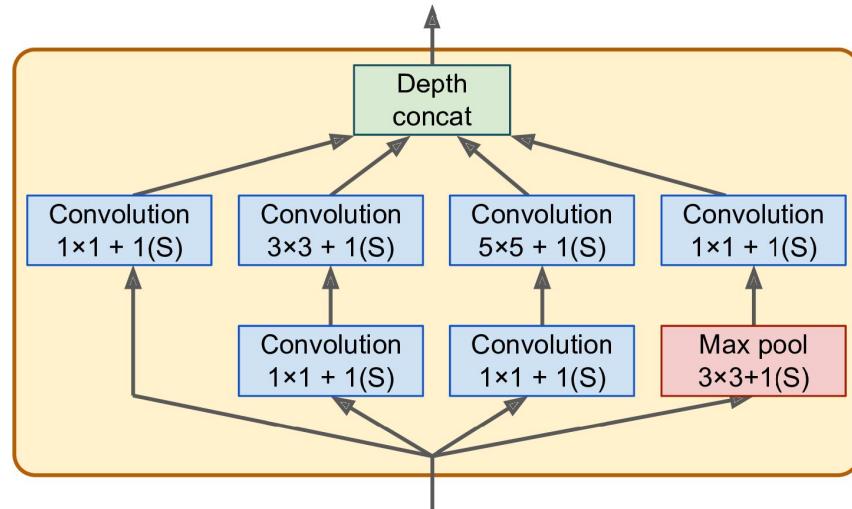
C. Szegedy et al., [Going deeper with convolutions](#), CVPR 2015

# CNN - GoogLeNet (2014)



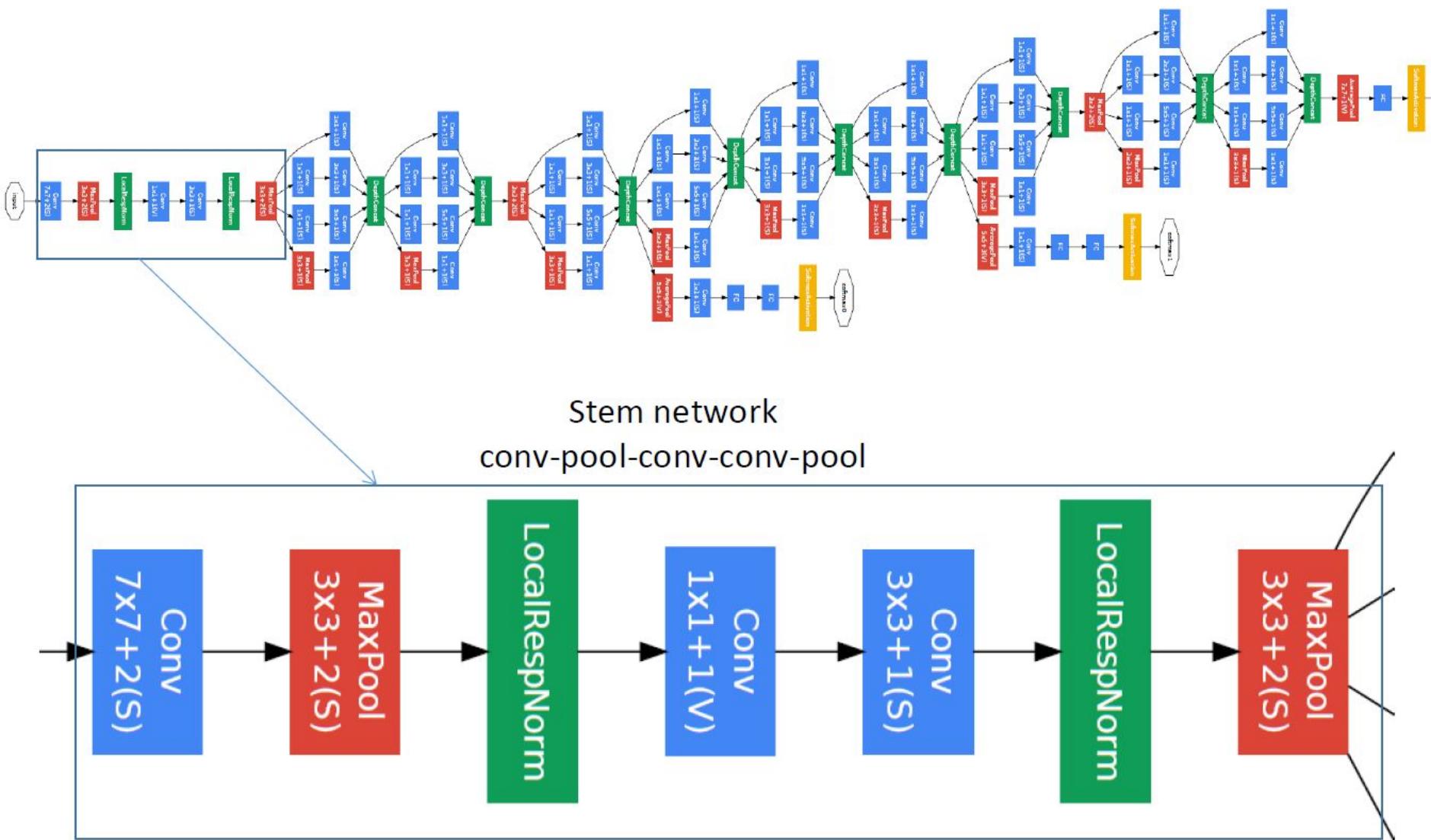
# GoogleLeNet (2014) - Inception Module

- ▶ Sinal de entrada é copiado e alimenta 4 camadas diferentes;
- ▶ Tamanhos diferentes de kernels ( $1 \times 1$ ,  $3 \times 3$  e  $5 \times 5$ ) para identificar padrões em escalas diferentes.
- ▶ Stride 1 e padding “same” para que as saídas tenham a mesma largura e altura da entrada
  - ▶ Permite a concatenação das saídas das 4 camadas na profundidade
- ▶ Camada de concatenação - empilha/concatena os mapas de características das 4 camadas anteriores

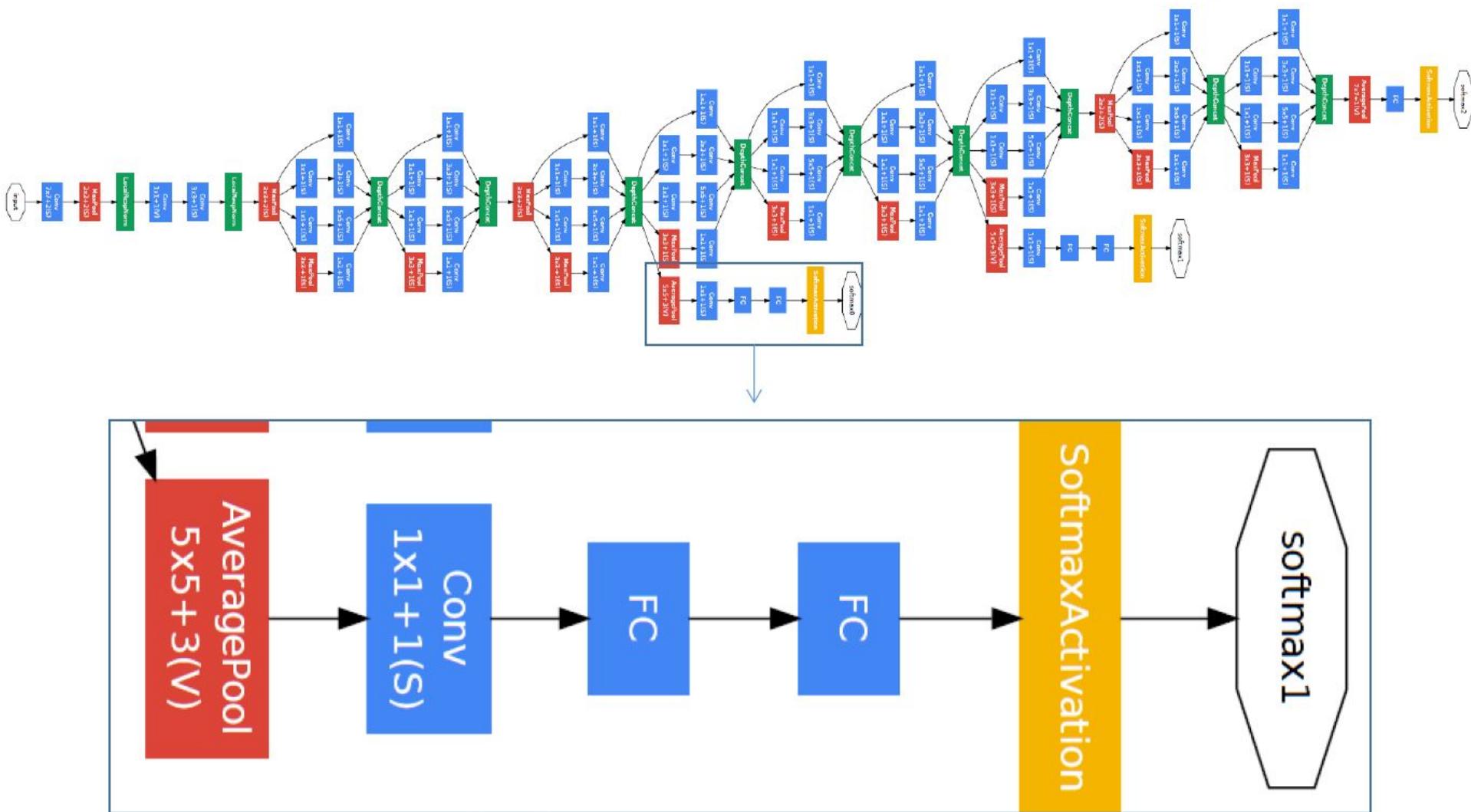




# CNN – GoogLeNet (2014)



# CNN - GoogLeNet (2014)



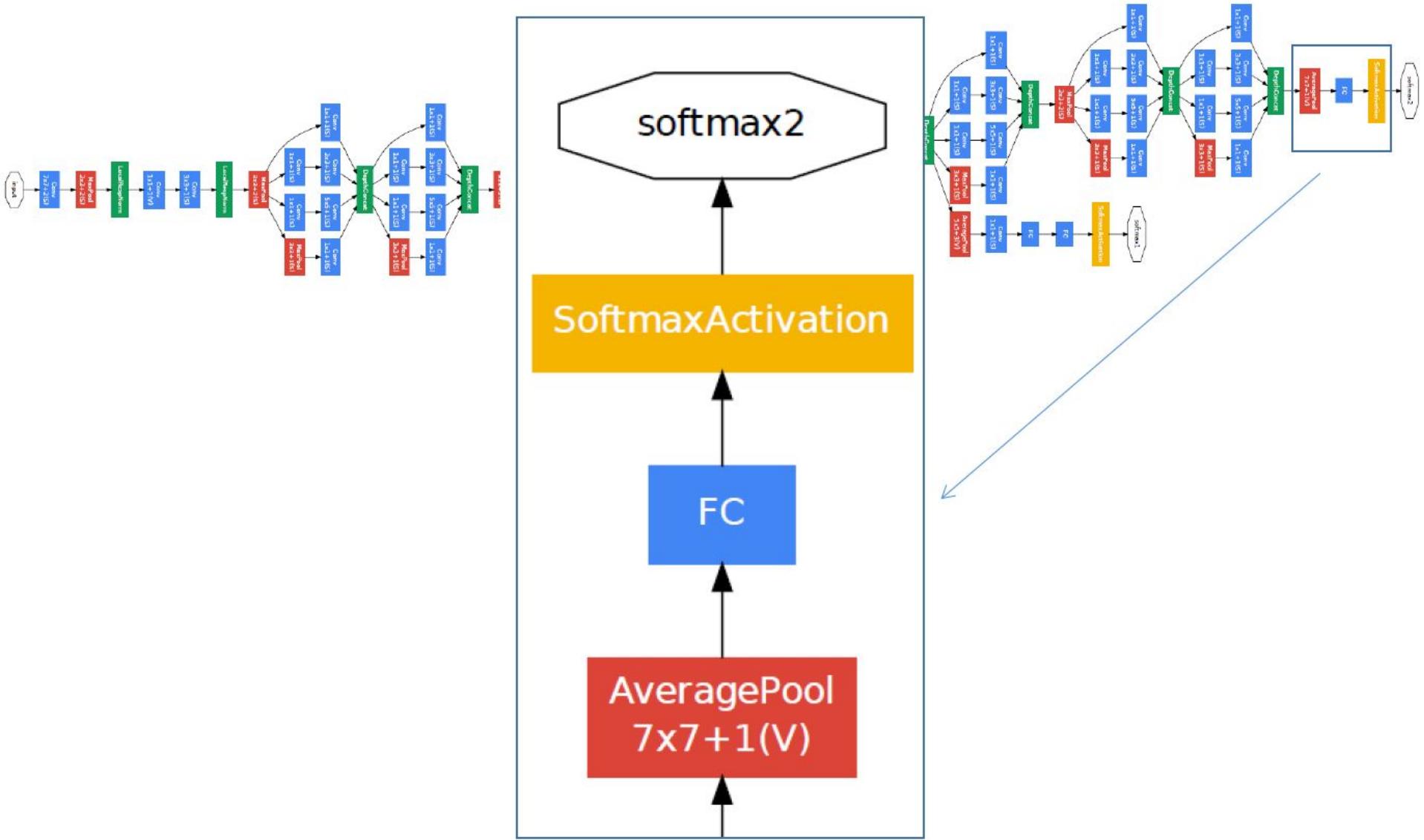
Classificações parciais para auxiliar no  
backpropagation

# GoogleLeNet (2014) -

---

- ▶ **Classificadores auxiliares/parciais:**
  - ▶ Durante o treinamento, o erro (reduzido em 70%) das classificações parciais era adicionada ao erro geral.
  - ▶ Objetivo era evitar que os gradientes desaparecessem e regularizar a rede;
  - ▶ Mais tarde, comprovou-se que seu efeito era bem menor.

# CNN - GoogLeNet (2014)



# CNN - GoogleLeNet (2014)

---

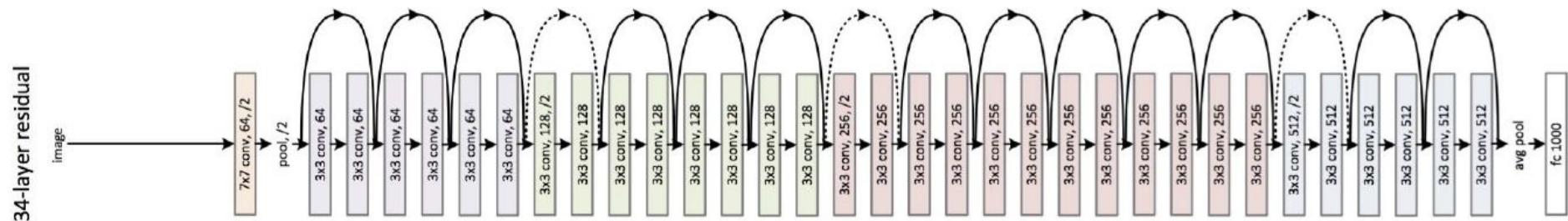
- ▶ 22 camadas
- ▶ 5 milhões de parâmetros
- ▶ 12x menos que AlexNet



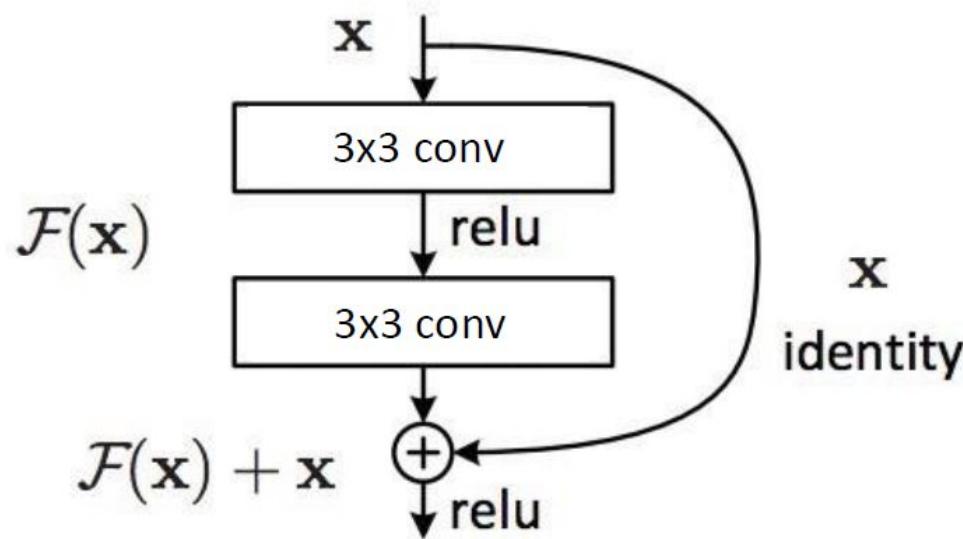
# CNN – GoogLeNet (2014)

# CNN – ResNet (2015)

## Residual Network

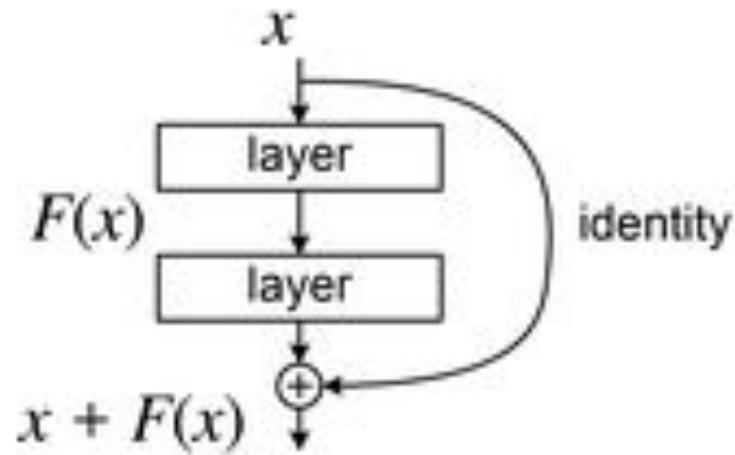


152 Camadas na ILSVRC, 1202 no CIFAR



# CNN - ResNet (2015)

- ▶ Modelos cada vez mais profundos e com menos parâmetros
- ▶ Skip connections: segredo para treinar uma rede tão profunda

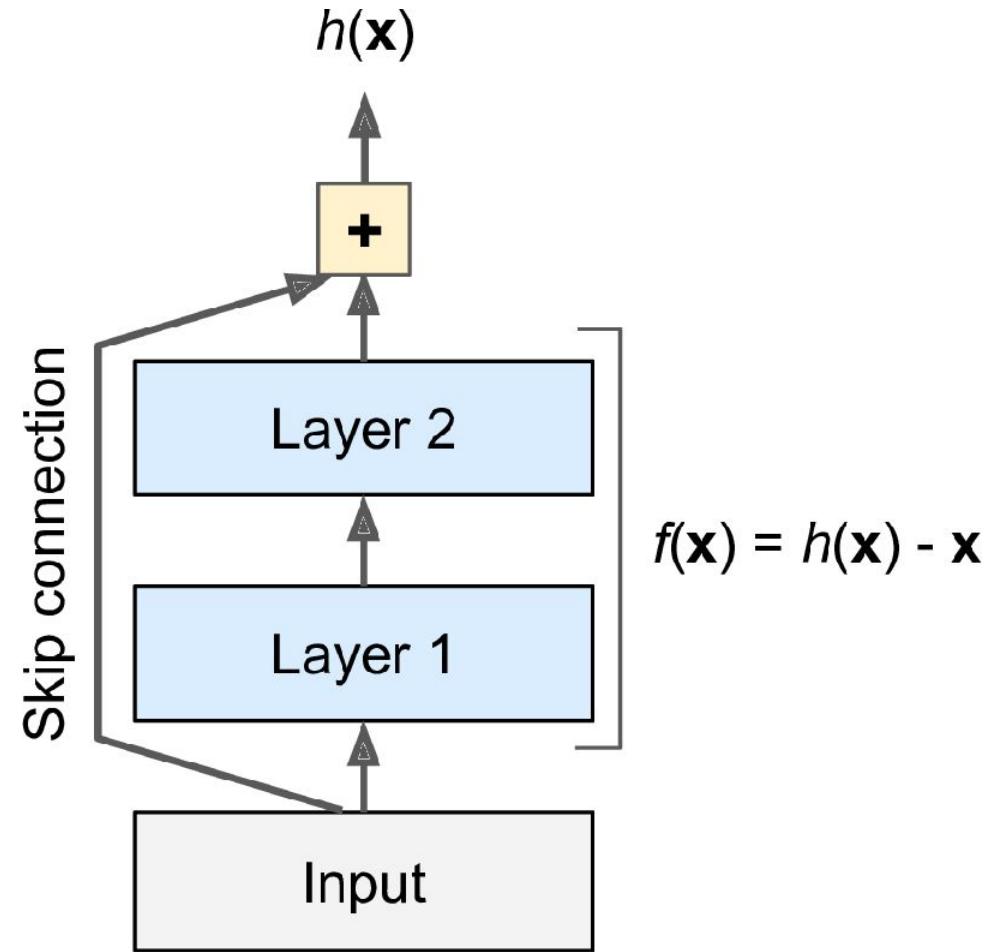
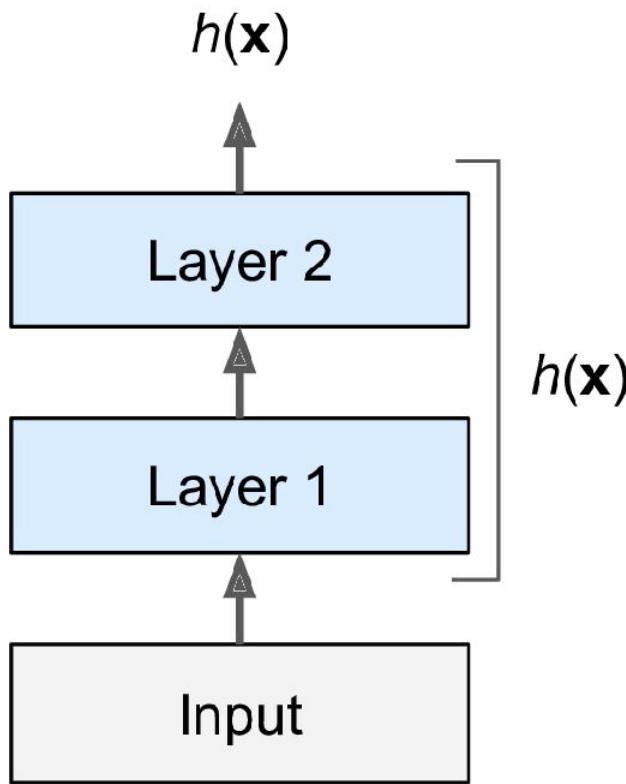


# ResNet (2015) - Skip Connections

---

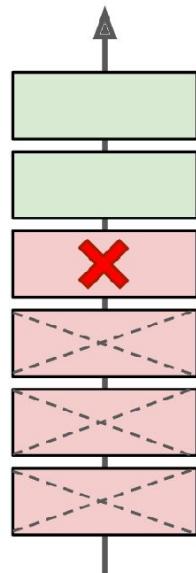
- ▶ Por quê?
  - ▶ Ao treinar uma rede, o objetivo é fazer com que ela aprenda uma função alvo  $h(x)$
  - ▶ Se você adicionar a entrada  $x$  à saída (skip connection), a rede será forçada a modelar  $f(x) = h(x) - x$ , em vez de  $h(x)$
  - ▶ Ao inicializar a rede neural, pesos são próximos de zero, e portanto, a rede só gera valores próximos de zero na saída.
  - ▶ **Ao adicionar uma skip connection, a inicialmente modela a função de identidade:  $f(x) = x$ .**
  - ▶ Se a função alvo estiver próxima da função de identidade (geralmente é o caso), **isso vai acelerar o treinamento.**

# ResNet (2015) - Skip Connections



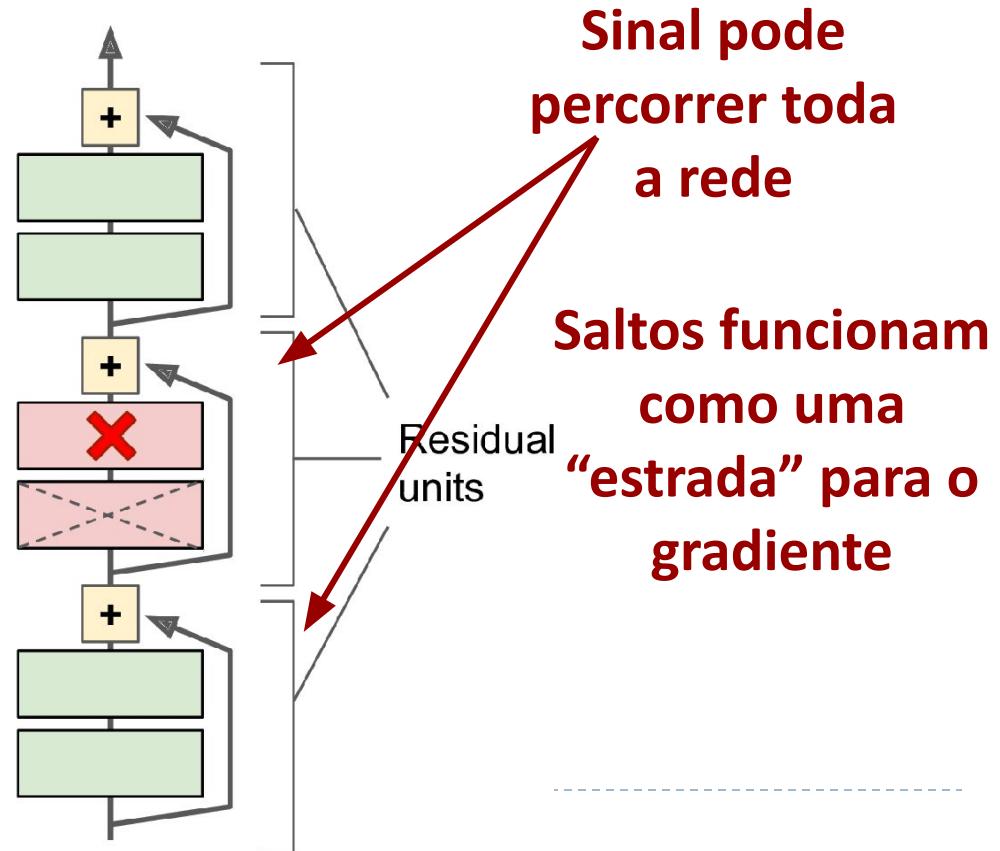
# ResNet (2015) - Skip Connections

- ▶ Com a adição de várias skip connections, a rede pode começar a progredir ainda que diversas camadas não tenham iniciado o aprendizado.



**✗** = Layer blocking backpropagation

= Layer not learning



Sinal pode percorrer toda a rede  
Saltos funcionam como uma “estrada” para o gradiente  
Residual units

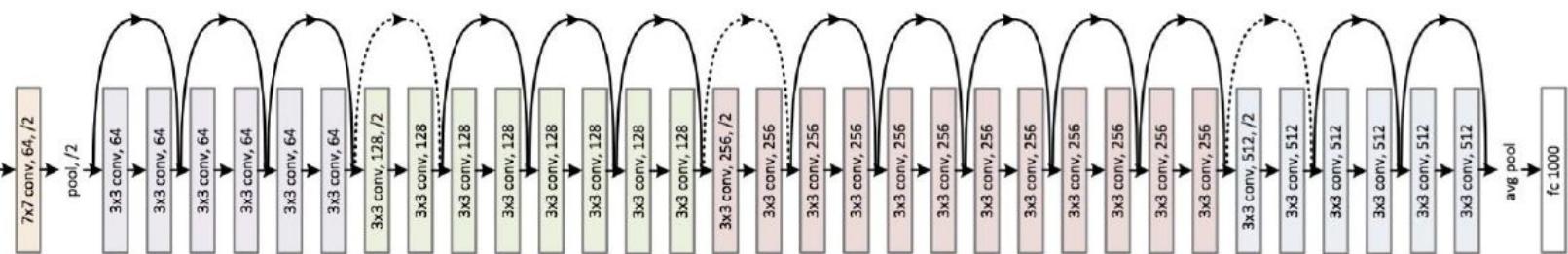
# CNN - ResNet (2015)

---

- ▶ Por que os blocos residuais melhoram a classificação?
  - ▶ Uma razão intuitiva é que **os saltos** (de uma camada para outra) formam uma "**estrada para o gradiente**"
  - ▶ Gradientes calculados podem afetar diretamente os pesos das primeiras camadas fazendo atualizações ter mais efeito.

# CNN - ResNet (2015)

34-layer residual



64 convs 7x7, stride 2

MaxPool 2x2

Blocos residuais com 64 convs 3x3

Bloco residual com 128 convs 3x3 stride 2 e 128 convs 3x3

Blocos residuais com 128 convs 3x3

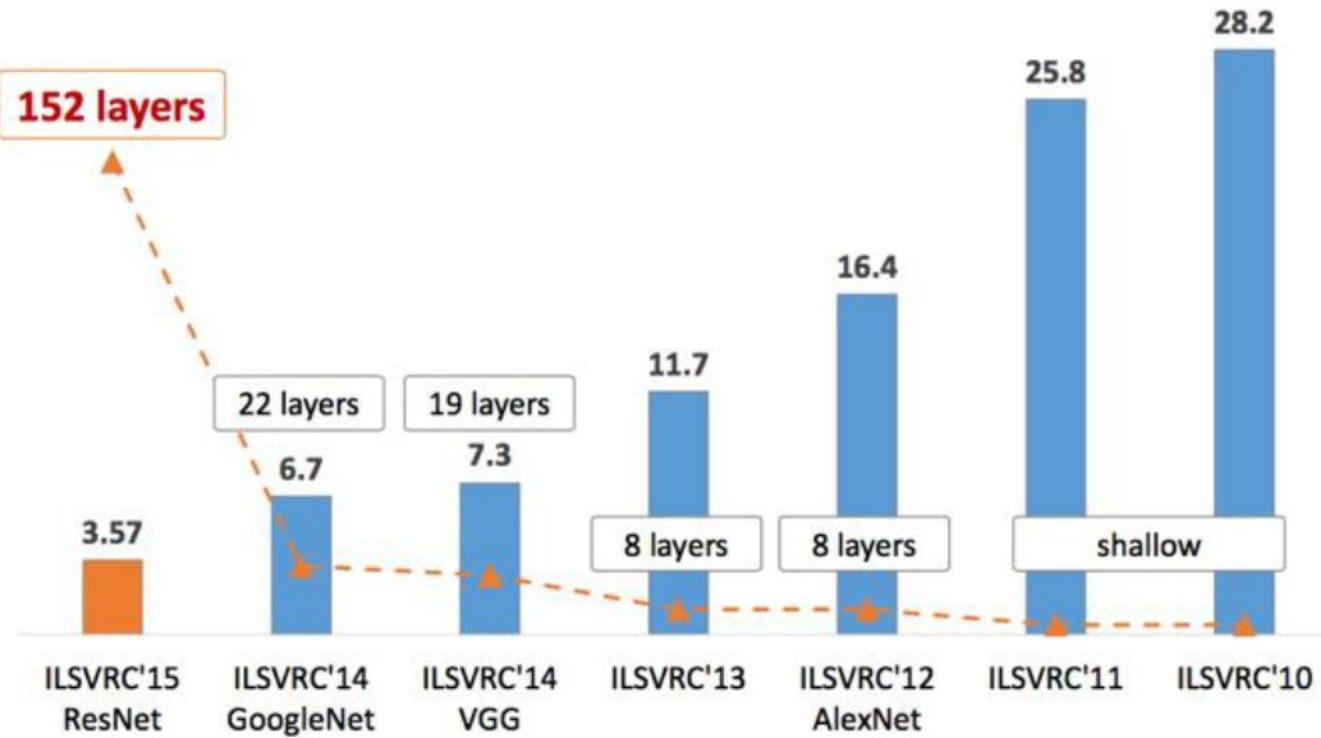
Bloco residual com 256 convs 3x3 stride 2 e 256 convs 3x3

Blocos residuais com 256 convs 3x3

Bloco residual com 512 convs 3x3 stride 2 e 512 convs 3x3

Blocos residuais com 512 convs 3x3

AvgPool e FC 1000 (apenas para os escores)



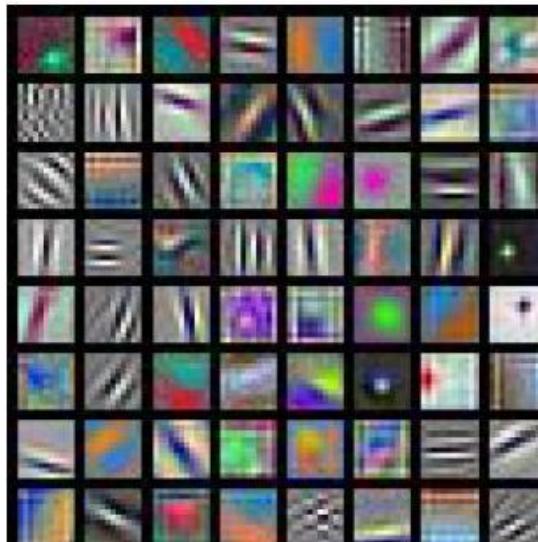
2016

Ensamble of Inception, ResNet and Inception/ResNet

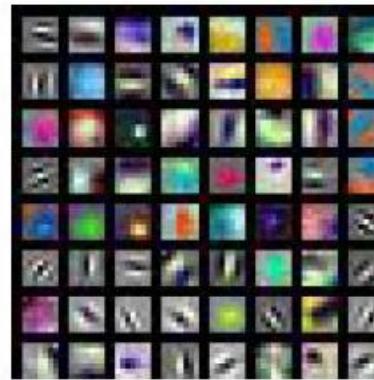
Erro de 2,991 (próximo ao nível de ruído)

No Google, Microsoft, Baidu...

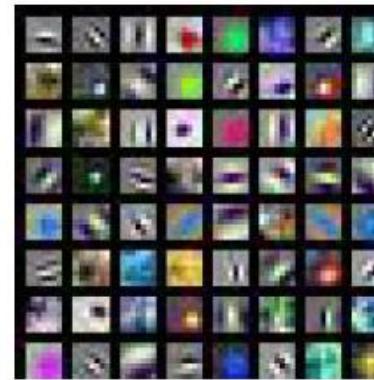
# Visualização - Filtros das Primeiras Camadas



AlexNet:  
 $64 \times 3 \times 11 \times 11$



ResNet-18:  
 $64 \times 3 \times 7 \times 7$



ResNet-101:  
 $64 \times 3 \times 7 \times 7$



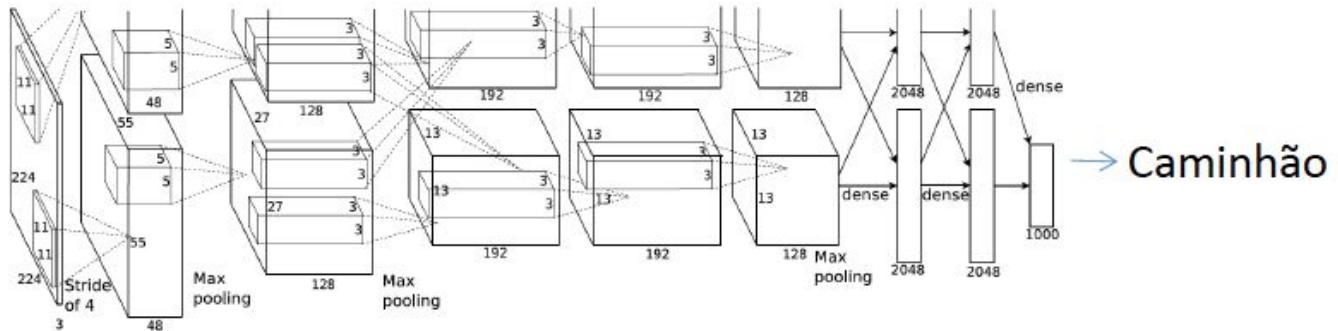
DenseNet-121:  
 $64 \times 3 \times 7 \times 7$

---

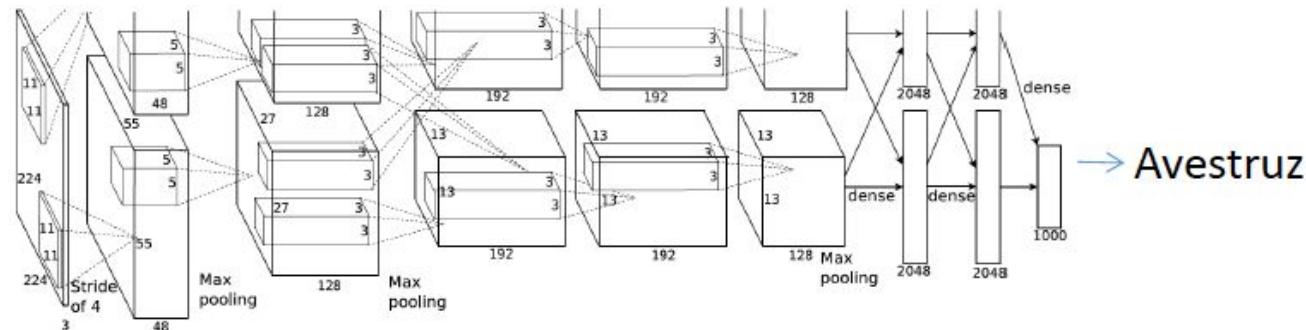
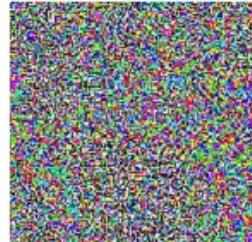
# Redes Adversárias (*Generative Adversarial Networks* - GANs)

# Redes Adversárias (GANs)

- ▶ Ajusta a imagem de entrada para **enganar** a rede.



+

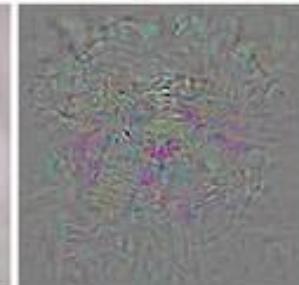
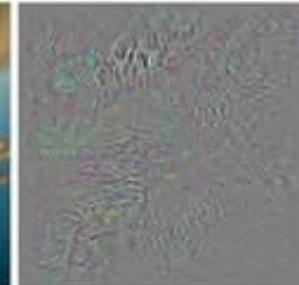
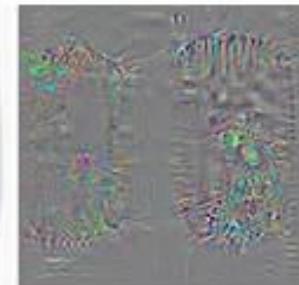
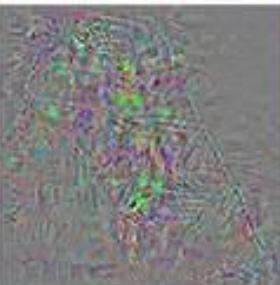
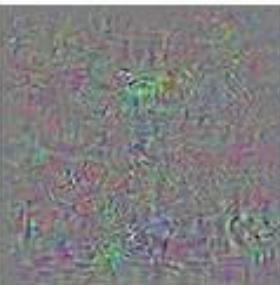


# Redes Adversárias (GANs)

- ▶ Ajustar (por backpropagation) os pixels da imagem de entrada, até obter a saída desejada, com a confiança que se deseja

$$\begin{array}{ccc} \text{panda} & + .007 \times & \text{gibbon} \\ x & \text{sign}(\nabla_x J(\theta, x, y)) & \xrightarrow{\epsilon \text{sign}(\nabla_x J(\theta, x, y))} \\ \text{"panda"} & \text{"nematode"} & \text{"gibbon"} \\ 57.7\% \text{ confidence} & 8.2\% \text{ confidence} & 99.3 \% \text{ confidence} \end{array}$$

# Redes Adversárias (GANs)



correct

+distort

ostrich

correct

+distort

ostrich

# Redes Adversárias (GANs)

African elephant



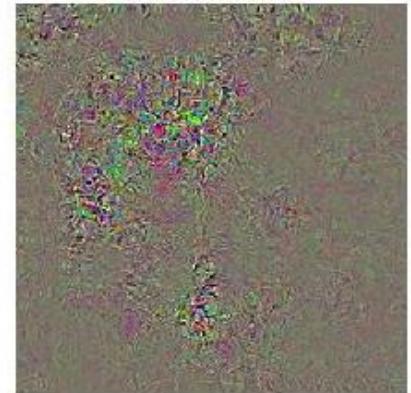
koala



Difference



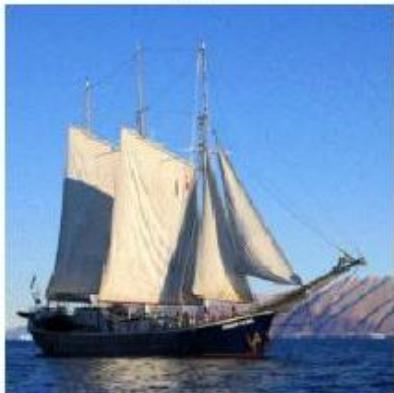
10x Difference



schooner



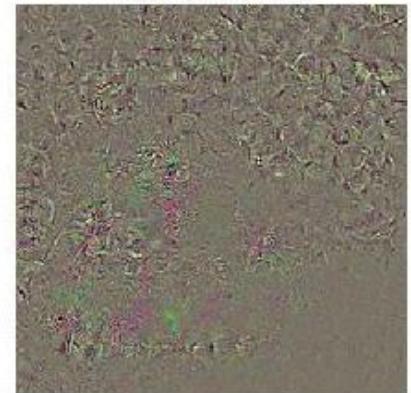
iPod



Difference

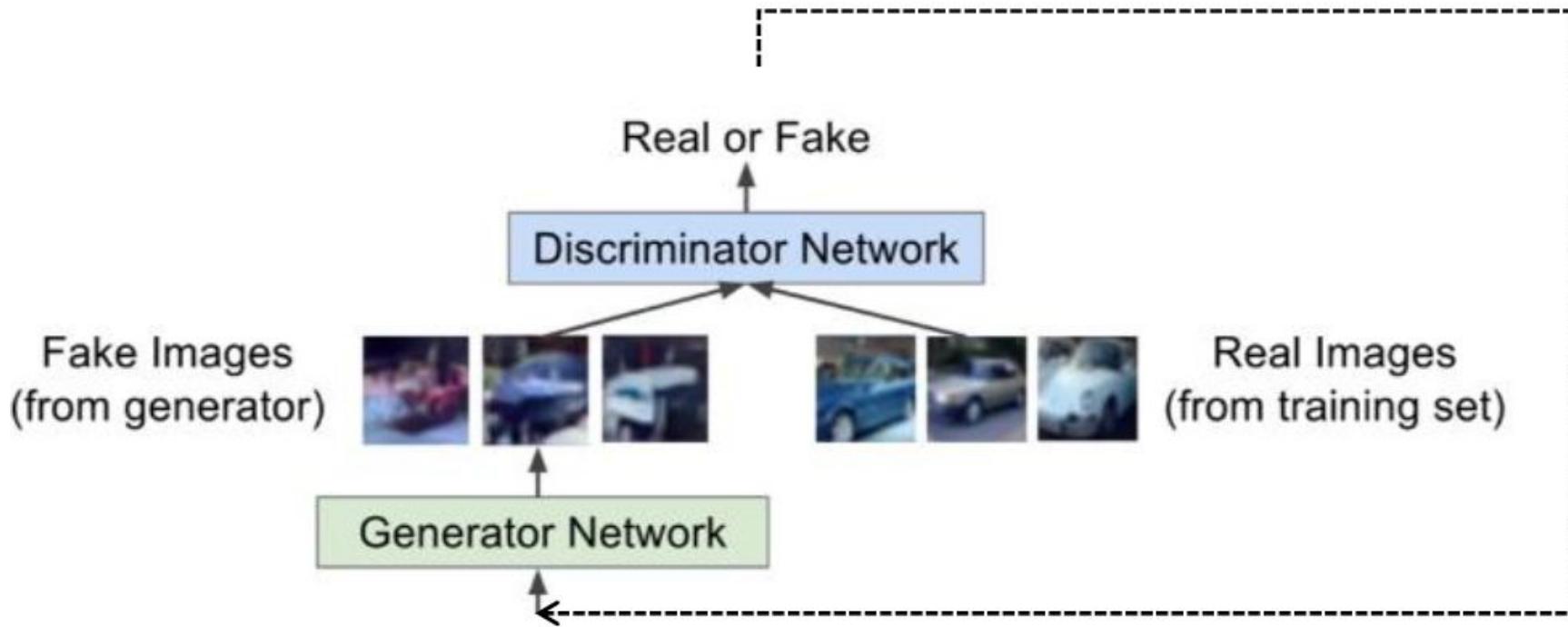


10x Difference



# Redes Adversárias (GANs)

- ▶ Solução: Treinar rede com exemplos adversários
- ▶ Jogos de dois jogadores



Universidade Federal da Paraíba

Centro de Informática

---

Departamento de Informática

# Aprendizado Profundo Redes Neurais Convolucionais (Adaptado do Material do Prof. Leonardo Batista)

Thais Gaudencio

Tiago Maritan