

# (4) Redes Neurais Convolucionais - CNNs

## Redes Neurais e Aprendizado Profundo

Moacir Ponti

*ICMC, Universidade de São Paulo*

[www.icmc.usp.br/~moacir](http://www.icmc.usp.br/~moacir) — [moacir@icmc.usp.br](mailto:moacir@icmc.usp.br)

## Redes Neurais Convolucionais (CNNs)

- Convolução

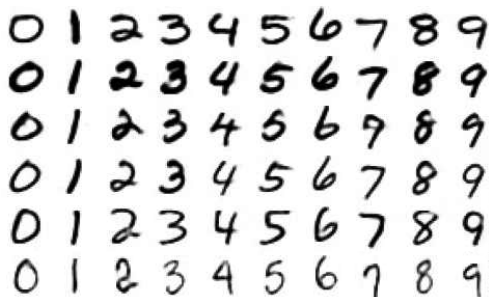
- Camada convolucional para redes neurais

- Exemplo

- Número de parâmetros

- Pooling

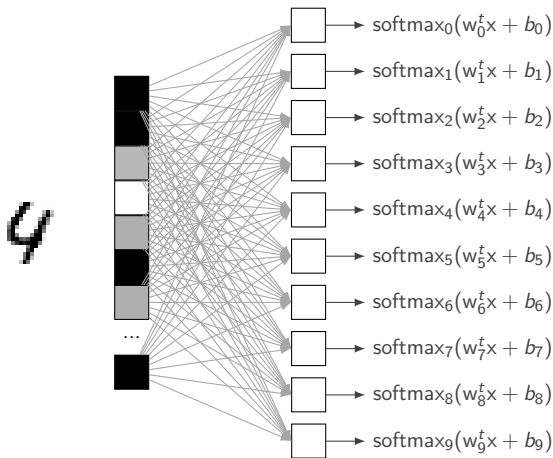
## Exemplo de problema: classificação de dígitos



- ▶ Imagens com  $28 \times 28 = 784$  pixels,

# Questões importantes sobre as redes MLP (densas)

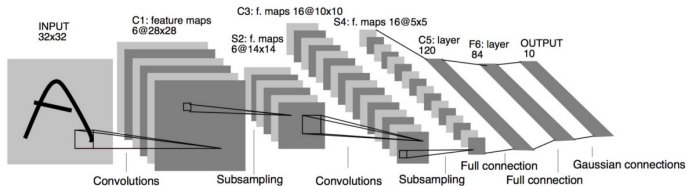
1. Valores de entrada (atributos) são considerados independentes
2. Não são aproveitadas relações locais entre os dados



# Questões importantes sobre as redes MLP (densas)

1. Grande número de parâmetros: memória e processamento
  - ▶ Exemplo: entrada imagem de  $28 \times 28 = 784$
  - ▶ Uma camada com 100 neurônios teria..
  - ▶  $78400 + 100 = 78500$  parâmetros a serem aprendidos e mantidos na memória durante o treinamento

# Redes Neurais Convolucionais (CNNs)



(Arquitetura LeNet)

Nova terminologia:

- ▶ Camada convolucional (convolutional layer)
- ▶ Subamostragem (pooling)
- ▶ Mapas de Ativação (activation/feature maps)
- ▶ Camada densa (dense/fully connected, tipo MLP)

# Convolução

- ▶ Operador que visa realizar uma combinação linear de valores locais da entrada
- ▶ Centrado em uma posição, e.g.  $(x, y)$ , gera como saída um único valor de saída

# Convolução

Volume de entrada 7 x 7								Volume de saída										
	0	1	2	3	4	5	6	Filtro W (3 x 3)				0	1	2	3	4	5	6
0	2	2	2	2	3	3	3	-1	0.5	1	0							
1	1	0	1	1	1	1	0	-1	0	0	1							
2	1	1	3	3	0	0	0	0	0	0.5	2							
3	1	1	3	2	0	0	3				3							
4	1	1	3	2	0	0	3				4							
5	1	3	3	2	0	0	3				5							
6	3	3	3	2	0	0	3				6							



# Convolução

Volume de entrada 7 x 7								Volume de saída										
	0	1	2	3	4	5	6	Filtro W (3 x 3)				0	1	2	3	4	5	6
0	2	2	2	2	3	3	3	-1	0.5	1	0							
1	1	0	1	1	1	1	0	-1	0	0	1	1.5						
2	1	1	3	3	0	0	0	0	0	0.5	2							
3	1	1	3	2	0	0	3				3							
4	1	1	3	2	0	0	3				4							
5	1	3	3	2	0	0	3				5							
6	3	3	3	2	0	0	3				6							



























- ▶ **Zero-padding:** para compensar a impossibilidade de computar todos os valores;
  - ▶ Amplia-se a entrada de forma que o volume de saída seja igual ao de entrada

















- ▶ **Convolução em profundidade:** quando a entrada possui mais do que 1 canal
  - ▶ O filtro terá  $k \times k \times p$ , onde  $p$  é a quantidade de canais de entrada

# Convolução

Volume de entrada 6 x 6 x 3 (RGB) + zero padding									Filtro W (3 x 3 x 3)				Volume de saída 6 x 6													
	0	1	2	3	4	5	6	7												0	1	2	3	4	5	
0	0	0	0	0	0	0	0	0	-1	0.5	1									0						
0	0	2	2	2	2	3	3	0	-1	0	0									1						
1	0	1	0	1	1	1	1	0	0	0	0.5									2						
2	0	1	1	3	3	0	0	0				1	0	1						3						
3	0	1	1	3	2	0	0	0				-1	1	0						4						
4	0	1	1	3	2	0	0	0				0	0	-0.5						5						
5	0	1	3	3	2	0	0	0							1	0	1									
7	0	0	0	0	0	0	0	0							1	0.5	-1									
															0	1	0									
	0	1	2	3	4	5	6	7		0	1	2	3	4	5	6	7									
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0									
1	0	3	3	1	1	1	1	0	1	0	0	3	2	2	3	2	0									
2	0	3	0	3	1	1	1	0	2	0	1	1	1	1	1	0										
3	0	3	3	3	3	0	0	0	3	0	1	2	3	3	0	0	0									
4	0	0	0	3	2	0	0	0	4	0	1	2	3	1	1	1	0									
5	0	1	2	2	2	0	0	0	5	0	1	3	3	2	1	1	0									
6	0	2	2	2	2	0	0	0	6	0	3	3	0	2	0	2	0									
7	0	0	0	0	0	0	0	0	7	0	0	0	0	0	0	0	0									





# Convolução

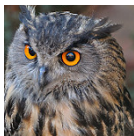
Volume de entrada 6 x 6 x 3 (RGB) + zero padding								Filtro W (3 x 3 x 3)			Volume de saída 6 x 6										
	0	1	2	3	4	5	6	7													
0	0	0	0	0	0	0	0	0	-1	0.5	1										
0	0	2	2	2	2	3	3	0	-1	0	0	0	0	0	0	0	0	0	0		
1	0	1	0	1	1	1	1	0	0	0	0.5	0	1	-2.5							
2	0	1	1	3	3	0	0	0		1	0	1	1								
3	0	1	1	3	2	0	0	0		-1	1	0	2								
4	0	1	1	3	2	0	0	0		0	0	-0.5	3								
5	0	1	3	3	2	0	0	0			1	0	1	4							
7	0	0	0	0	0	0	0	0			1	0.5	-1	5							
											0	1	0								
	0	1	2	3	4	5	6	7													
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
1	0	3	3	1	1	1	1	0	1	0	0	3	2	2	3	2	0				
2	0	3	0	3	1	1	1	0	2	0	1	1	1	1	1	0					
3	0	3	3	3	3	0	0	0	3	0	1	2	3	3	0	0	0				
4	0	0	0	3	2	0	0	0	4	0	1	2	3	1	1	1	0				
5	0	1	2	2	2	0	0	0	5	0	1	3	3	2	1	1	0				
6	0	2	2	2	2	0	0	0	6	0	3	3	0	2	0	2	0				
7	0	0	0	0	0	0	0	0	7	0	0	0	0	0	0	0	0				

# Convolução

Volume de entrada 6 x 6 x 3 (RGB) + zero padding									Filtro W (3 x 3 x 3)			Volume de saída 6 x 6																																																																																																																																																																																																																																																																																																																																																																																																																																							
	0	1	2	3	4	5	6	7											0	0	0	0	0	0	0	0	0	-1	0.5	1						0	1	2	3	4	5	1	0	2	2	2	2	3	3	0	-1	0	0						0	1	-2.5	-1	-1	0.5	2	2	0	1	0	1	1	1	1	0	0	0	0.5						1	11.5	5.5	16.5	9.5	8	3	3	0	1	1	3	3	0	0	0				1	0	1			-1	4.5	8	7	9	2.5	2.5	4	0	1	1	3	2	0	0	0									0	5.5	14	22	5	0.5	2.5	5	0	1	1	3	2	0	0	-2										1	5.5	12.5	9	7	-0.5	4.5	6	0	1	3	3	2	0	0	-1										1	4	11	6	7	-0.5	4.5	7	0	0	0	0	0	0	0	0										0								0	1	2	3	4	5	6	7																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	3	3	1	1	1	1	0	1	0	0	3	2	2	3	2	0								2	0	3	0	3	1	1	1	0	2	0	1	1	1	1	1	1	0								3	0	3	3	3	3	0	0	0	3	0	1	2	3	3	0	0	0								4	0	0	0	3	2	0	0	0	4	0	1	2	3	1	1	1	0								5	0	1	2	2	2	0	0	0	5	0	1	3	3	2	1	1	0								6	0	2	2	2	2	0	0	0	6	0	3	3	0	2	0	2	0								7	0	0	0	0	0	0	0	0	7	0	0	0	0	0	0	0	0							
0	0	0	0	0	0	0	0	0	-1	0.5	1						0	1	2	3	4	5																																																																																																																																																																																																																																																																																																																																																																																																																													
1	0	2	2	2	2	3	3	0	-1	0	0						0	1	-2.5	-1	-1	0.5	2																																																																																																																																																																																																																																																																																																																																																																																																																												
2	0	1	0	1	1	1	1	0	0	0	0.5						1	11.5	5.5	16.5	9.5	8	3																																																																																																																																																																																																																																																																																																																																																																																																																												
3	0	1	1	3	3	0	0	0				1	0	1			-1	4.5	8	7	9	2.5	2.5																																																																																																																																																																																																																																																																																																																																																																																																																												
4	0	1	1	3	2	0	0	0									0	5.5	14	22	5	0.5	2.5																																																																																																																																																																																																																																																																																																																																																																																																																												
5	0	1	1	3	2	0	0	-2										1	5.5	12.5	9	7	-0.5	4.5																																																																																																																																																																																																																																																																																																																																																																																																																											
6	0	1	3	3	2	0	0	-1										1	4	11	6	7	-0.5	4.5																																																																																																																																																																																																																																																																																																																																																																																																																											
7	0	0	0	0	0	0	0	0										0																																																																																																																																																																																																																																																																																																																																																																																																																																	
	0	1	2	3	4	5	6	7																																																																																																																																																																																																																																																																																																																																																																																																																																											
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																																																																																																																																																																																																																																																																																																																																																																																																																												
1	0	3	3	1	1	1	1	0	1	0	0	3	2	2	3	2	0																																																																																																																																																																																																																																																																																																																																																																																																																																		
2	0	3	0	3	1	1	1	0	2	0	1	1	1	1	1	1	0																																																																																																																																																																																																																																																																																																																																																																																																																																		
3	0	3	3	3	3	0	0	0	3	0	1	2	3	3	0	0	0																																																																																																																																																																																																																																																																																																																																																																																																																																		
4	0	0	0	3	2	0	0	0	4	0	1	2	3	1	1	1	0																																																																																																																																																																																																																																																																																																																																																																																																																																		
5	0	1	2	2	2	0	0	0	5	0	1	3	3	2	1	1	0																																																																																																																																																																																																																																																																																																																																																																																																																																		
6	0	2	2	2	2	0	0	0	6	0	3	3	0	2	0	2	0																																																																																																																																																																																																																																																																																																																																																																																																																																		
7	0	0	0	0	0	0	0	0	7	0	0	0	0	0	0	0	0																																																																																																																																																																																																																																																																																																																																																																																																																																		

# Camada convolucional

**Entrada** ( $m \times n \times p$ )



e.g.  $32 \times 32 \times 3$

**Filtro** (kernel ou neurônio convolucional)  $w$  com tamanho  $k \times k \times p$ , e.g.  $5 \times 5 \times 3$

- ▶ Cada neurônio realiza a convolução da entrada e gera um volume (matriz/tensor) de saída

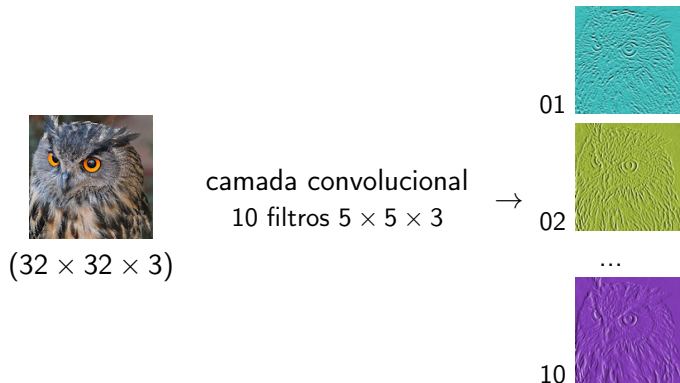
**Centrado** em um pixel específico, temos, matematicamente

$$w^t x + b$$

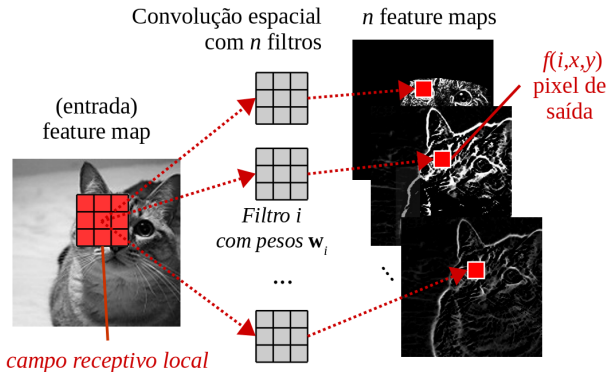
- sim, há a soma de **bias** para além dos pesos da convolução.

# Camada convolucional

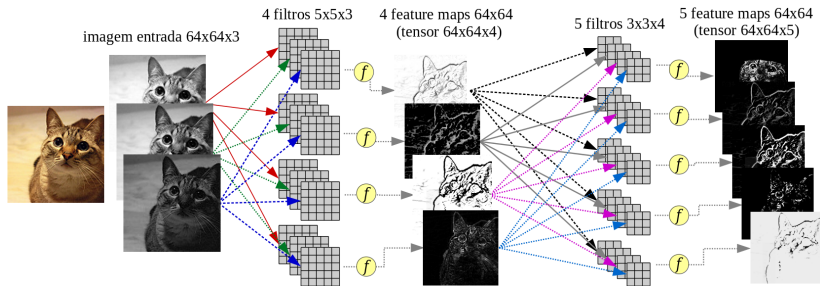
- ▶ Mapas de ativação (ou características) são obtidos após convolução e função de ativação (e.g. ReLU);
- ▶ Empilhados formam um tensor que será a entrada da próxima camada.



# Camada convolucional: campo receptivo local



# Camada convolucional: feature maps



## Camada convolucional: entrada, filtro, passo

A camada convolucional tem que levar em conta:

- ▶ tamanho da entrada (largura, altura, profundidade)
- ▶ tamanho do filtro
  - ▶ a profundidade deve ser igual à da entrada
  - ▶ altura e largura afetam o *campo receptivo local*
- ▶ stride (passo)
  - ▶ 1 : todos os pixels são filtrados pelo neurônio
  - ▶ > 1 : salta um número de pixels em determinada direção, a cada convolução.
    - ▶ nesse caso o volume de saída tem tamanho reduzido, ex. com passo 2





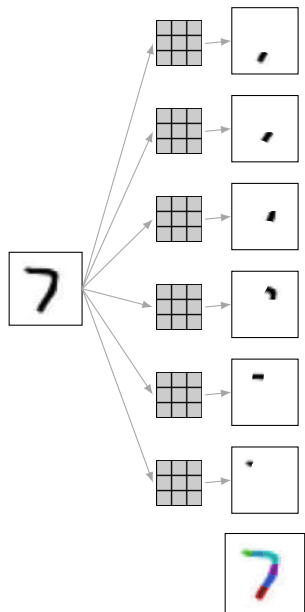




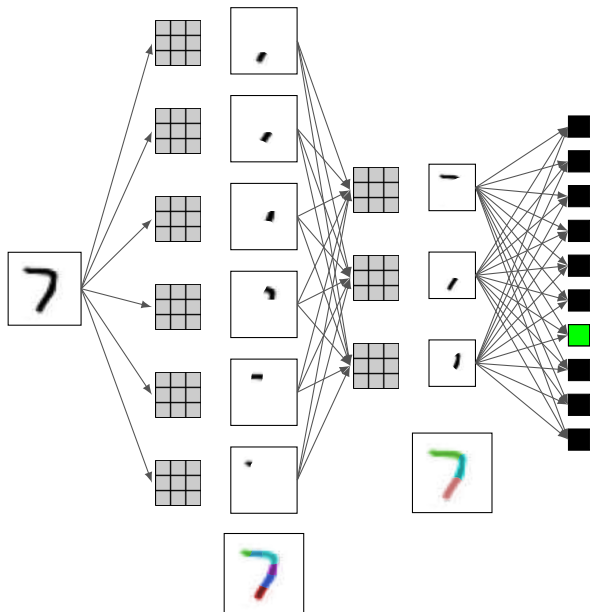




# Classificação de dígitos com conv.layers



# Classificação de dígitos com conv.layers



# Número de parâmetros em CNNs

$[(k \times k \times p) + 1] \times d$ :

- ▶ pesos dos filtros:  $k \times k \times p$ ,  $p$  é dado pela profundidade da entrada
- ▶ número de filtros/neurônios:  $d$  (cada um gera um mapa de ativação)
- ▶ +1 é o termo bias de cada filtro

Ex: entrada  $32 \times 32 \times 3$  e 3 camadas:

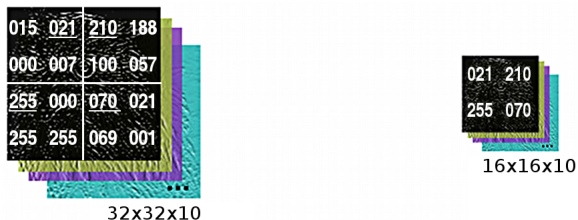
- ▶ Conv.Layer 1:  $k = 5$ ,  $d = 8$
- ▶ Conv.Layer 2:  $k = 3$ ,  $d = 16$
- ▶ Conv.Layer 3:  $k = 1$ ,  $d = 32$
- ▶ # parâmetros Conv.layer 1:  $[(5 \times 5 \times 3) + 1] \times 8 = 608$
- ▶ # parameters Conv.layer 2:  $[(3 \times 3 \times 8) + 1] \times 16 = 1168$
- ▶ # parameters Conv.layer 3:  $[(1 \times 1 \times 16) + 1] \times 32 = 544$

## Subamostragem: Pooling layer

Opera sobre cada mapa de ativação, reduzindo a dimensão lateral

- ▶ max pooling: aplica a operação de máximo local
- ▶ average pooling: aplica operação de média local

Ex.: max pooling com tamanho de pool 2 e passo 2.



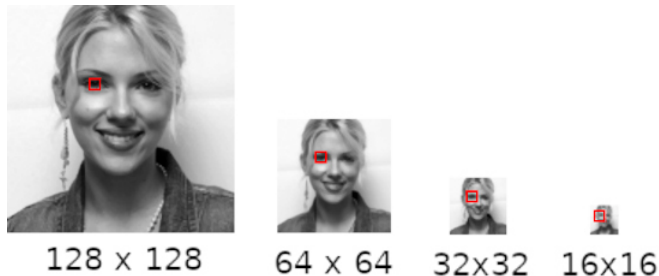
Usar camadas convolucionais com passo/stride  $> 1$  pode substituir pooling



# Pooling layer

Reduzir o tamanho da entrada permite que o filtro opere em regiões maiores da imagem.

Empilhamento de camadas convolucionais aumenta o campo receptivo local não necessitando manter a resolução de entrada



(uso de filtro de mesmo tamanho em imagens progressivamente menores)

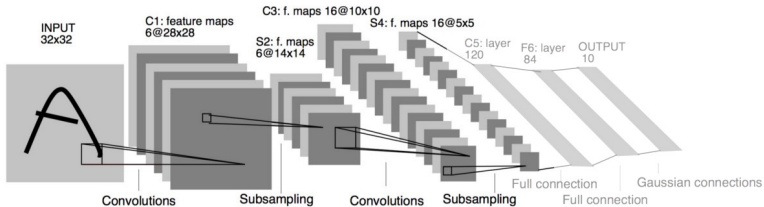
## Global pooling

Obtém um valor por canal, como se o tamanho de pool fosse igual às dimensões laterais

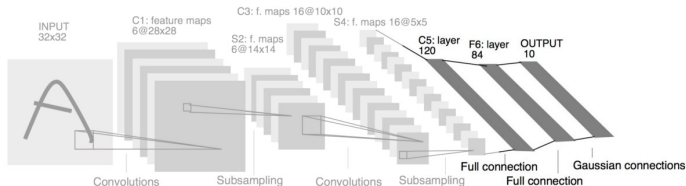
Ex. numa entrada com  $40 \times 40 \times 100$ , a saída será 100 dimensões.

# Voltando à arquitetura: camadas convolucionais

– Extraem características espaciais



# Camadas densas e saída



## Dense/fully connected (FC) layer:

- ▶ similar à de uma MLP
- ▶ pode ser vista como uma **projeção** dos dados em uma dimensionalidade arbitrária

## Saída: comumente densa (ex: classificação e regressão)

- ▶ pode ser vista como um vetor de distribuição de probabilidades
- ▶ não é densa em redes completamente convolucionais (Fully Convolutional Networks)

# Bibliography I



Moacir A. Ponti, Gabriel Paranhos da Costa. **Como funciona o Deep Learning**

SBC, 2017. Book chapter.

<https://arxiv.org/abs/1806.07908>



Moacir A. Ponti, Leo Ribeiro, Tiago Nazaré, Tu Bui, John Collomosse. **Everything You Wanted to Know About Deep Learning for Computer Vision but were Afraid to Ask.**

SIBGRAPI-T, 2017. Tutorial.