

# **Reti Neurali Convoluzionali per il riconoscimento di immagini**

**PCTO Addestramento di Reti Neurali con Linguaggio Python**

**Laura Nenzi, Gloria Pietropolli, Gaia Saveri**



**LICEO SCIENTIFICO  
GALILEO GALILEI**



**UNIVERSITÀ  
DEGLI STUDI  
DI TRIESTE**

# Come fanno i computer a vedere ed interpretare ciò che è un'immagine?

La **visione artificiale** è un campo dell'informatica  
che si occupa di permettere ai computer di  
interpretare comprendere il mondo visivo.

# AI e Computer Vision

## Come insegnare ai computer a capire le immagini?



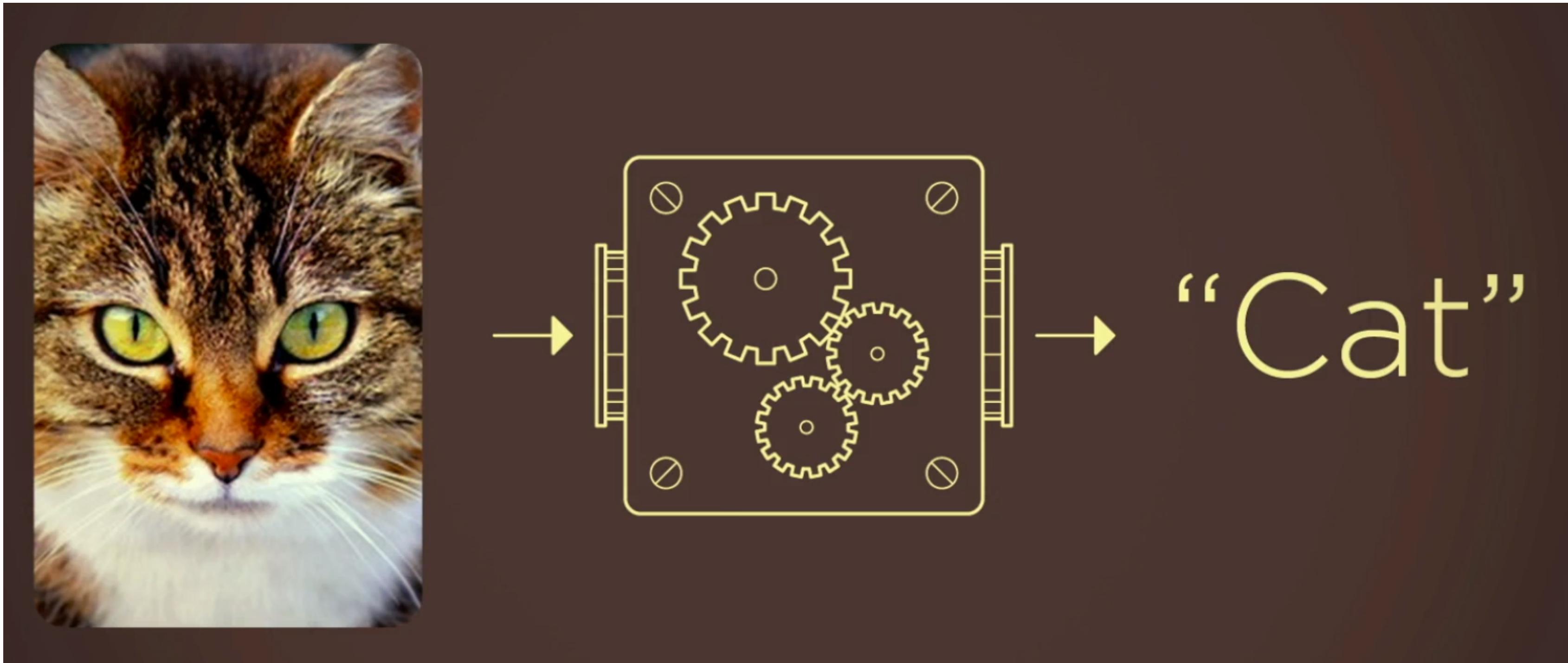
0	2	15	0	0	11	10	0	0	0	0	9	9	0	0	0	0
0	0	0	4	60	157	236	255	255	177	95	61	32	0	0	29	
0	10	16	119	238	255	244	245	243	250	249	255	222	103	10	0	
0	14	170	255	255	244	254	255	253	245	255	249	253	251	124	1	
2	98	255	228	255	251	254	211	141	116	122	215	251	238	255	49	
13	217	243	255	155	33	226	52	2	0	10	13	232	255	255	36	
16	229	252	254	49	12	0	0	7	7	0	70	237	252	235	62	
6	141	245	255	212	25	11	9	3	0	115	236	243	255	137	0	
0	87	252	250	248	215	60	0	1	121	252	255	248	144	6	0	
0	13	113	255	255	245	255	182	181	248	252	242	208	36	0	19	
1	0	5	117	251	255	241	255	247	255	241	162	17	0	7	0	
0	0	0	4	58	251	255	246	254	253	255	120	11	0	1	0	
0	0	4	97	255	255	255	248	252	255	244	255	182	10	0	4	
0	22	206	252	246	251	241	100	24	113	255	245	255	194	9	0	
0	111	255	242	255	158	24	0	0	6	39	255	232	230	56	0	
0	218	251	250	137	7	11	0	0	0	2	62	255	250	125	3	
0	173	255	255	101	9	20	0	13	3	13	182	251	245	61	0	
0	107	251	241	255	230	98	55	19	118	217	248	253	255	52	4	
0	18	146	250	255	247	255	255	249	255	240	255	129	0	5	0	
0	0	23	113	215	255	250	248	255	255	248	248	118	14	12	0	
0	0	6	1	0	52	153	233	255	252	147	37	0	0	4	1	
0	0	5	5	0	0	0	0	0	14	1	0	6	6	0	0	

0	2	15	0	0	11	10	0	0	0	0	9	9	0	0	0	0
0	0	0	4	60	157	236	255	255	177	95	61	32	0	0	29	
0	10	16	119	238	255	244	245	243	250	249	255	222	103	10	0	
0	14	170	255	255	244	254	255	253	245	255	249	253	251	124	1	
2	98	255	228	255	251	254	211	141	116	122	215	251	238	255	49	
13	217	243	255	155	33	226	52	2	0	10	13	232	255	255	36	
16	229	252	254	49	12	0	0	7	7	0	70	237	252	235	62	
6	141	245	255	212	25	11	9	3	0	115	236	243	255	137	0	
0	87	252	250	248	215	60	0	1	121	252	255	248	144	6	0	
0	13	113	255	255	245	255	182	181	248	252	242	208	36	0	19	
1	0	5	117	251	255	241	255	247	255	241	162	17	0	7	0	
0	0	0	4	58	251	255	246	254	253	255	120	11	0	1	0	
0	0	4	97	255	255	255	248	252	255	244	255	182	10	0	4	
0	22	206	252	246	251	241	100	24	113	255	245	255	194	9	0	
0	111	255	242	255	158	24	0	0	6	39	255	232	230	56	0	
0	218	251	250	137	7	11	0	0	0	2	62	255	250	125	3	
0	173	255	255	101	9	20	0	13	3	13	182	251	245	61	0	
0	107	251	241	255	230	98	55	19	118	217	248	253	255	52	4	
0	18	146	250	255	247	255	255	249	255	240	255	129	0	5	0	
0	0	23	113	215	255	250	248	255	255	248	248	118	14	12	0	
0	0	6	1	0	52	153	233	255	252	147	37	0	0	4	1	
0	0	5	5	0	0	0	0	0	14	1	0	6	6	0	0	

Le immagini vengono viste come matrici di **pixels**

# L'uso dei Big Data

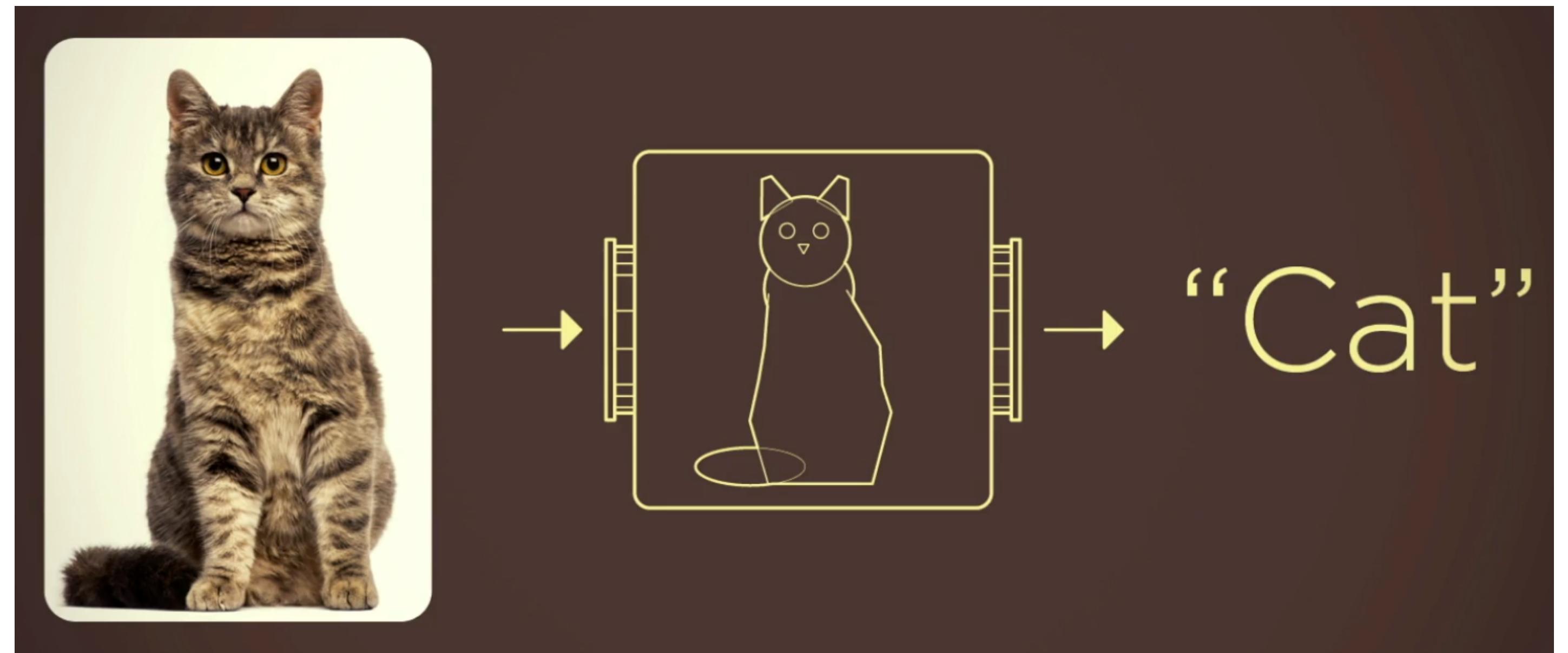
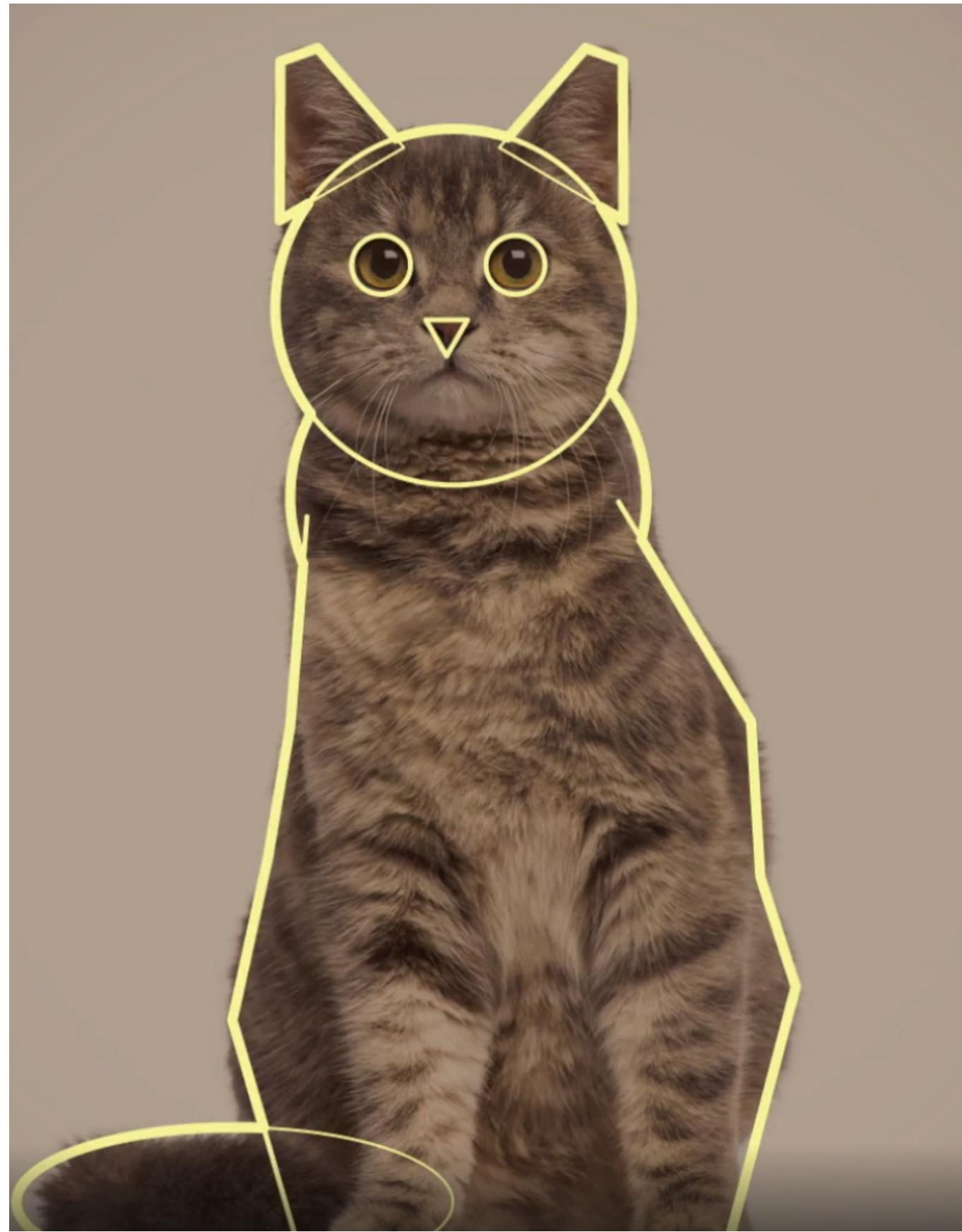
## Come le AI imparano



Come insegnamo all'AI che si tratta di un gatto?

# L'uso dei Big Data

Come le Intelligenze Artificiali imparano



Potremmo insegnarli le forme fondamentali che caratterizzano un gatto...

# L'uso dei Big Data

Come le Intelligenze Artificiali imparano



...ma alcuni gatti potrebbero ingannare la nostra AI

# L'uso dei Big Data

Come le Intelligenze Artificiali imparano



Utilizziamo tantissime immagini di gatti per far apprendere al nostro algoritmo di AI cosa sia un gatto

**Per far imparare un AI servono:**

**DATI, DATI E  
DATI**

**Per un AI nel campo del Computer Vision:**

**IMMAGINI, IMMAGINI  
E IMMAGINI**

**E nella vostra vita quotidiana,  
usate mai la visione artificiale?**

# Reti Neurali per il riconoscimento di immagini

## Classificare immagini utilizzando tecniche di Deep Learning



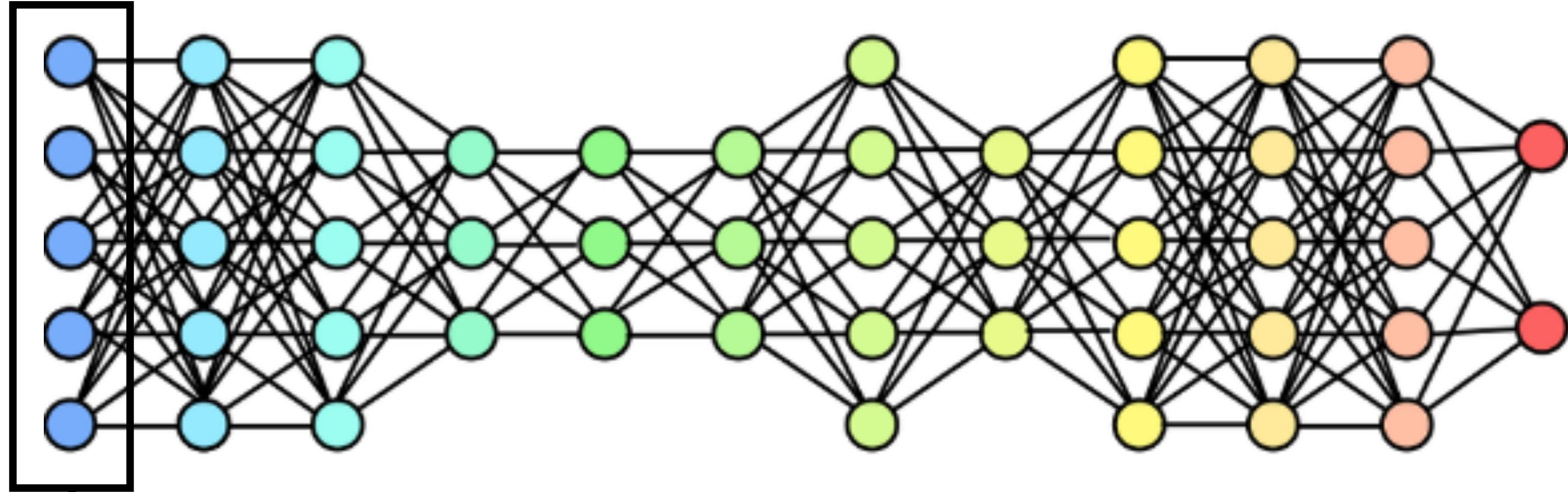
08	02	22	97	38	15	00	40	00	75	04	05	07	78	52	12	50	77	91	04
49	49	99	40	17	81	18	57	60	87	17	40	98	43	69	48	09	56	62	00
81	49	31	73	55	79	14	29	93	71	40	67	53	85	30	03	49	13	36	65
52	70	95	23	04	60	11	42	63	94	56	01	32	56	71	37	02	36	91	
22	31	16	71	51	62	03	89	41	92	36	54	22	40	40	28	66	33	13	80
24	47	38	60	99	03	45	02	44	75	33	53	78	36	84	20	35	17	12	50
32	98	81	28	64	23	67	10	26	38	40	67	59	54	70	66	18	38	64	70
67	26	20	68	02	62	12	20	95	63	94	39	63	08	40	91	66	49	94	21
24	55	58	05	66	73	99	26	97	17	78	78	96	83	14	88	34	89	63	72
21	36	23	09	75	00	76	44	20	45	35	14	00	61	33	97	34	31	33	95
78	17	53	28	22	75	31	67	15	94	03	80	04	62	16	14	09	53	56	92
16	39	05	42	96	35	31	47	55	58	88	24	00	17	54	24	36	29	85	57
86	56	00	48	35	71	89	07	05	44	44	37	44	60	21	58	51	54	17	58
19	80	81	68	05	94	47	69	28	73	92	13	86	52	17	77	04	89	55	40
04	52	08	83	97	35	99	16	07	97	57	32	16	26	26	79	33	27	98	66
09	34	68	87	57	62	20	72	03	46	33	67	46	55	12	32	63	93	53	69
04	42	16	73	36	05	39	11	24	94	72	18	08	46	29	32	40	62	76	36
20	69	36	41	72	30	23	88	39	89	69	82	67	59	85	74	04	36	16	
20	73	35	29	78	31	90	01	74	31	49	71	48	66	61	16	23	57	05	54
01	70	54	71	83	51	54	69	16	92	33	48	61	43	52	01	89	23	67	40

Cosa vede il computer

→ 85% gatto  
15% cane  
Classificazione  
(apprendimento supervisionato)

# Perché non si usano gli MLP per le immagini?

In breve: gli MLP non hanno la nozione di spazio



L'input degli MLP ha 1 dimensione, ma **una delle caratteristiche principali delle immagini è l'informazione spaziale!**



*Invarianza per traslazione*

Per usare un'immagine come input per un MLP, questa andrebbe appiattita:

- Perdita di informazione;
- Aumento eccessivo di parametri (uno per ogni pixel)

# Cosa sono le Reti Neurali Convoluzionali?

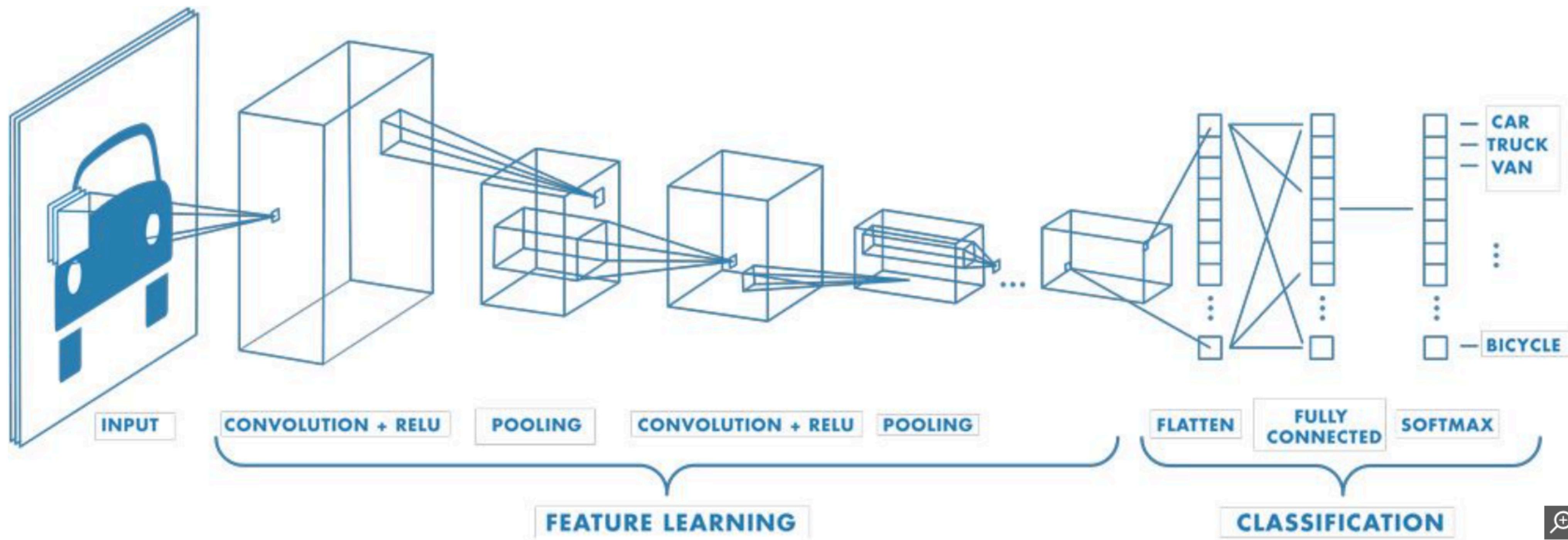
Le Reti Neurali Convoluzionali (CNN) sono una particolare architettura pensata per **preservare le correlazioni spaziali nell'input.**

Sí basano sull'operazione di convoluzione, in cui un filtro (kernel) scorre sull'immagine per astrarre delle caratteristiche (features).



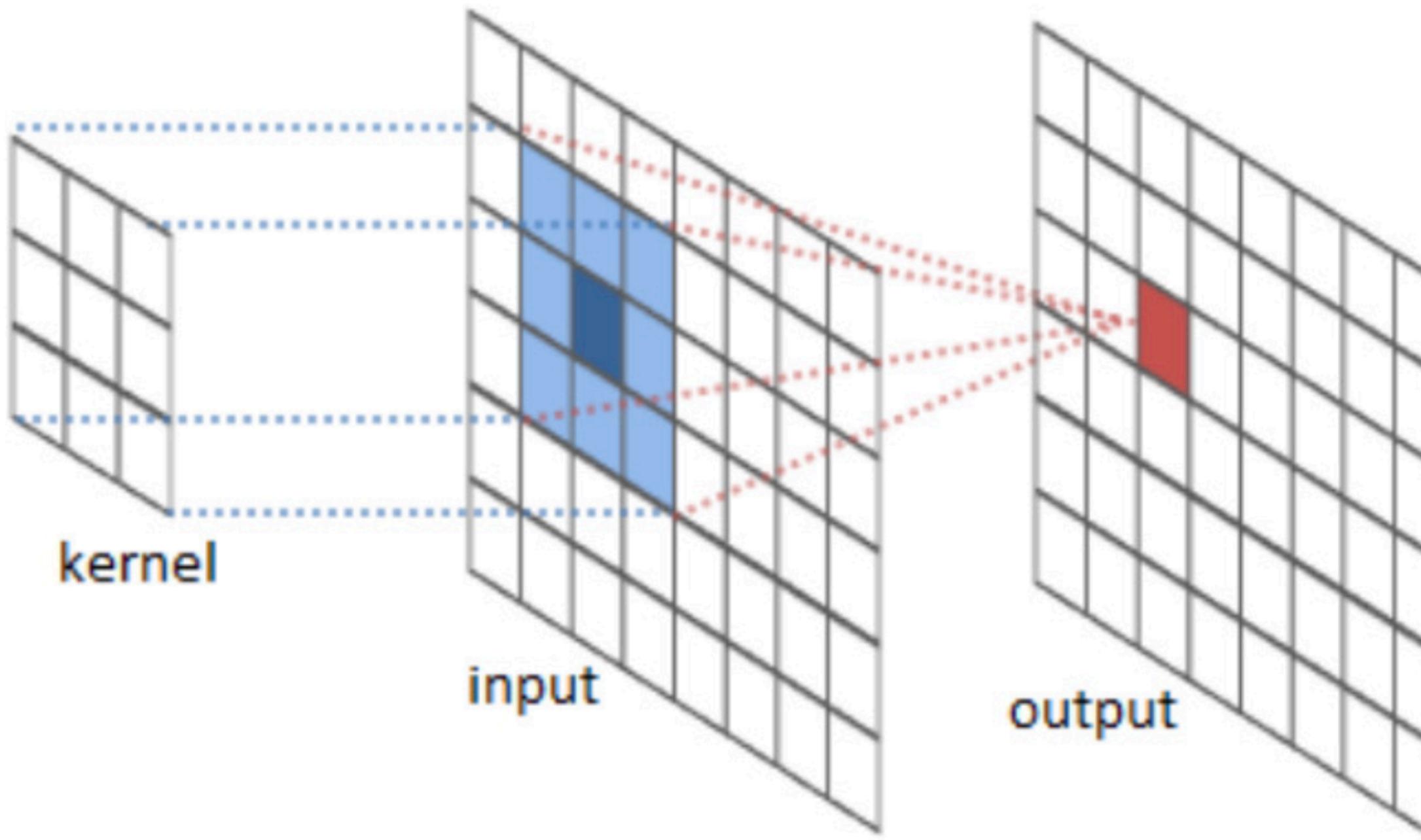
# Reti Neurali Convoluzionali

Come appaiono di solito



# Componenti di una CNN: kernel

Le CNN usano **molte copie dello stesso neurone**, applicati a parti delle immagini diverse, per **calcolare la stessa caratteristica in posizioni diverse dell'input**.

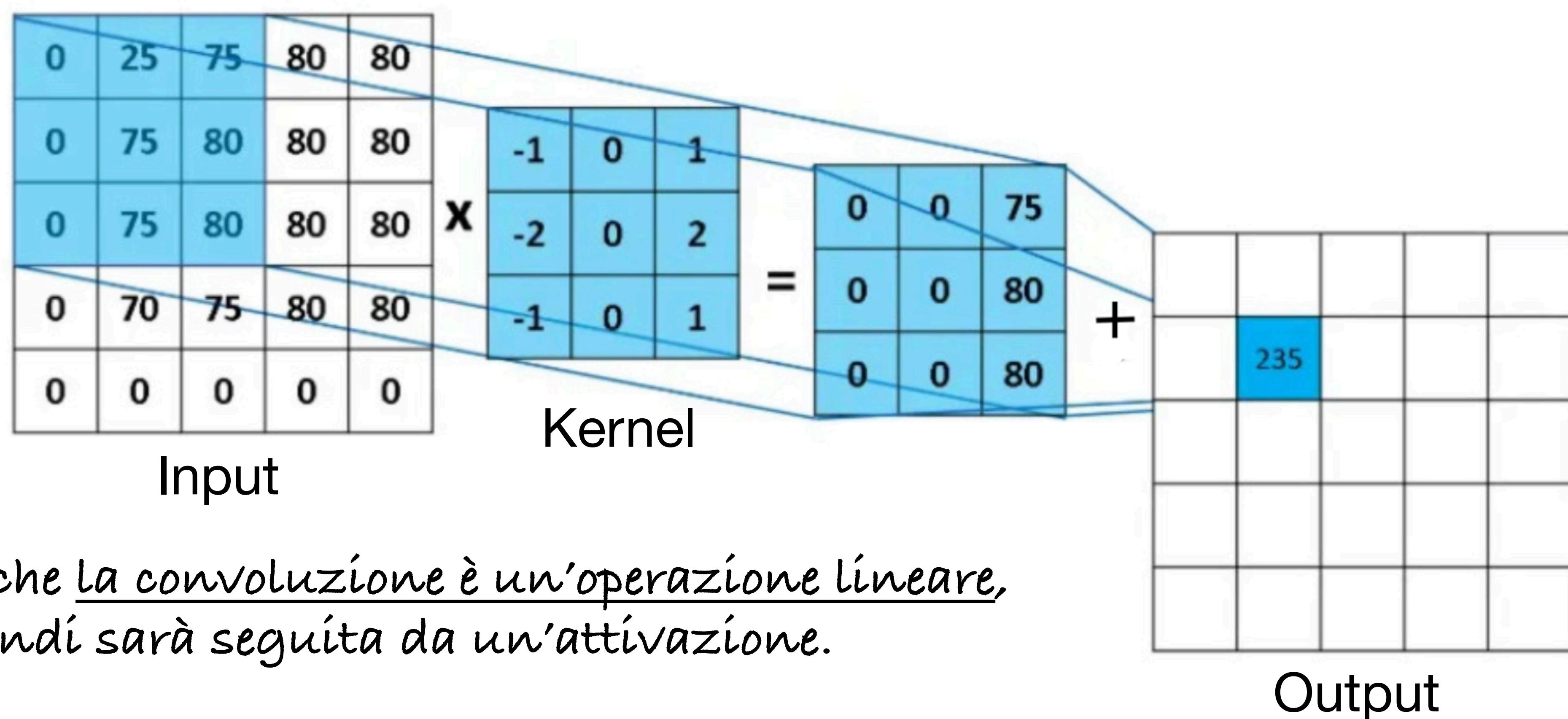


Il kernel è una matrice i cui componenti vengono imparati con l'addestramento.

Le dimensioni del kernel determinano la porzione di immagine considerata ad ogni step.

# Componenti di una CNN: kernel

Le CNN usano **molte copie dello stesso neurone**, applicati a parti delle immagini diverse, per **calcolare la stessa caratteristica in posizioni diverse dell'input**.

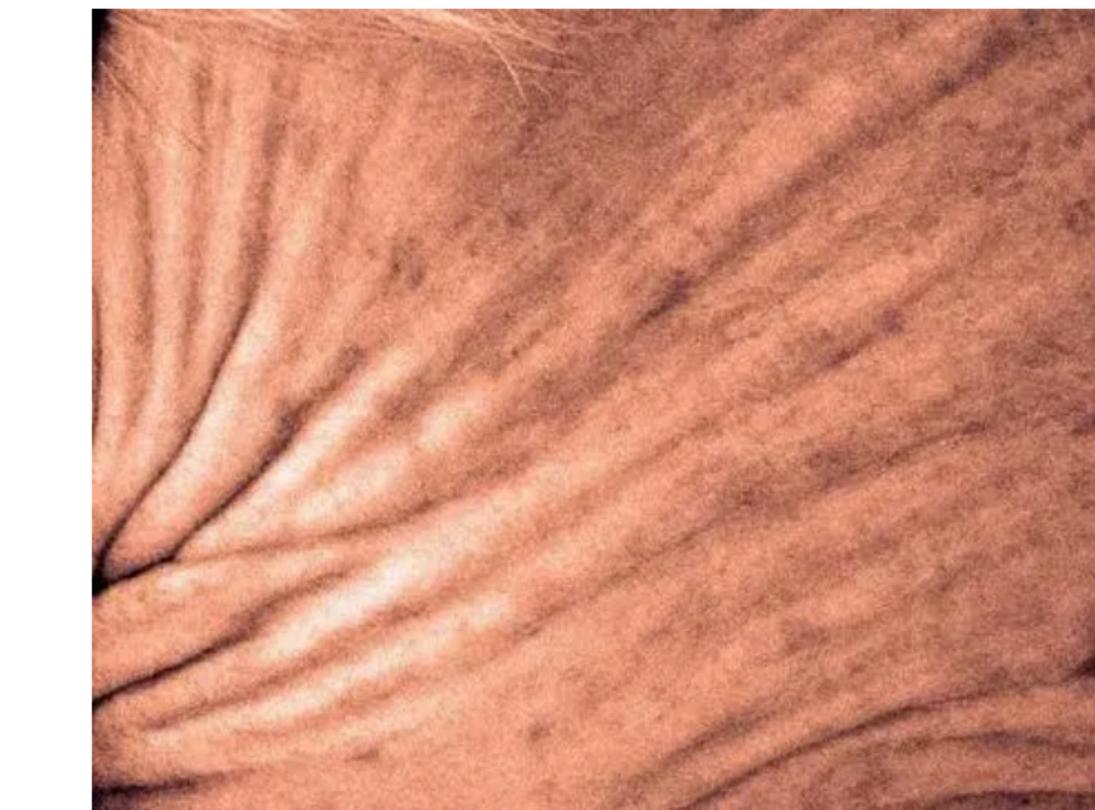
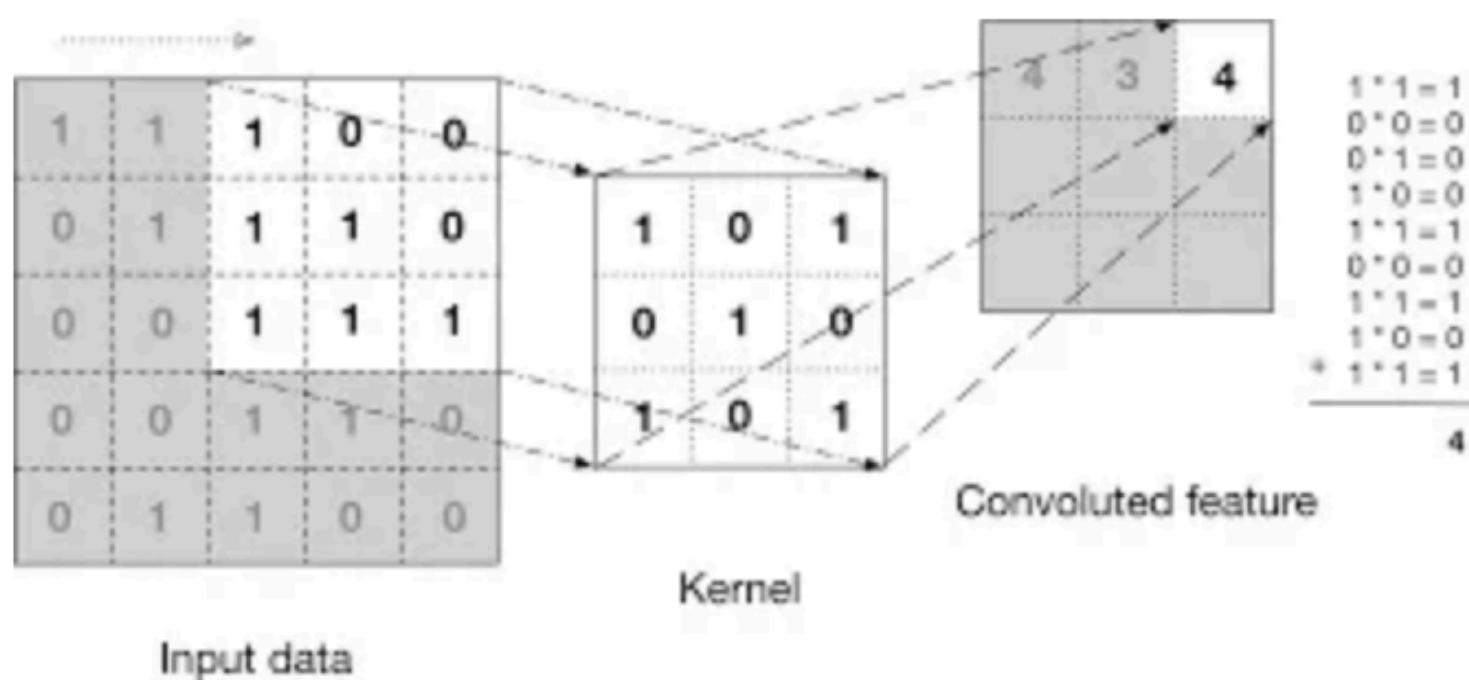


Anche la convoluzione è un'operazione lineare, quindi sarà seguita da un'attivazione.

# Componenti di una CNN: kernel

vogliamo capire che età ha una persona tramite la sua fotografia

→ Guardare i segni sul volto



1	-1	-1
-1	1	-1
-1	-1	1

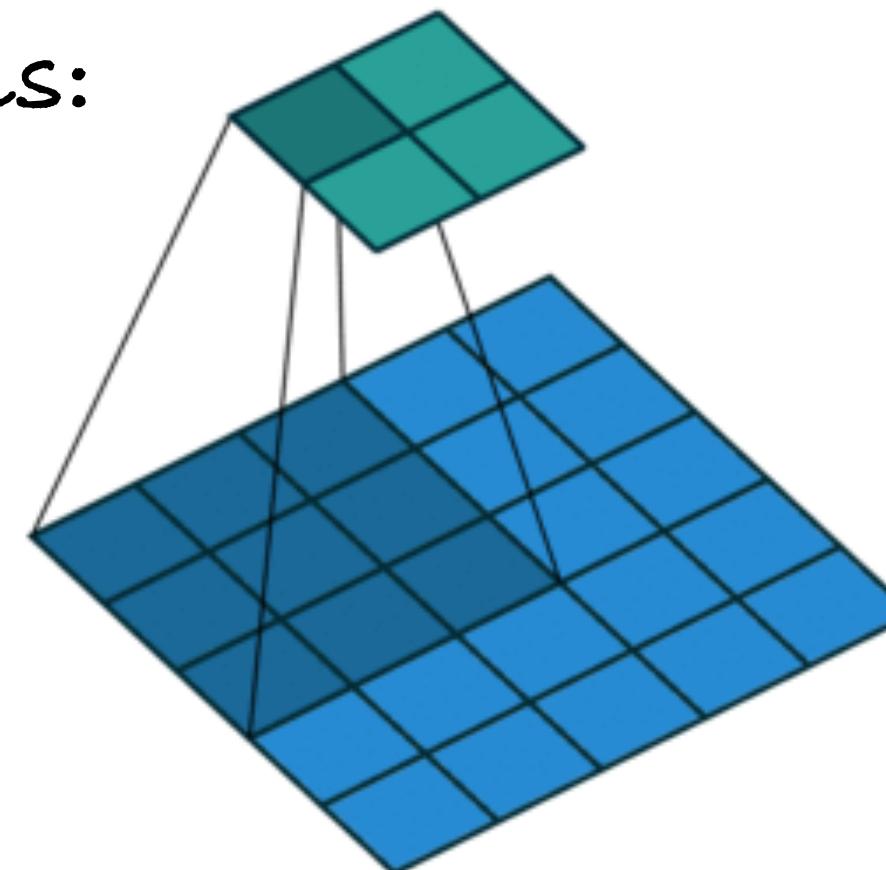
-1	-1	1
-1	1	-1
1	-1	-1

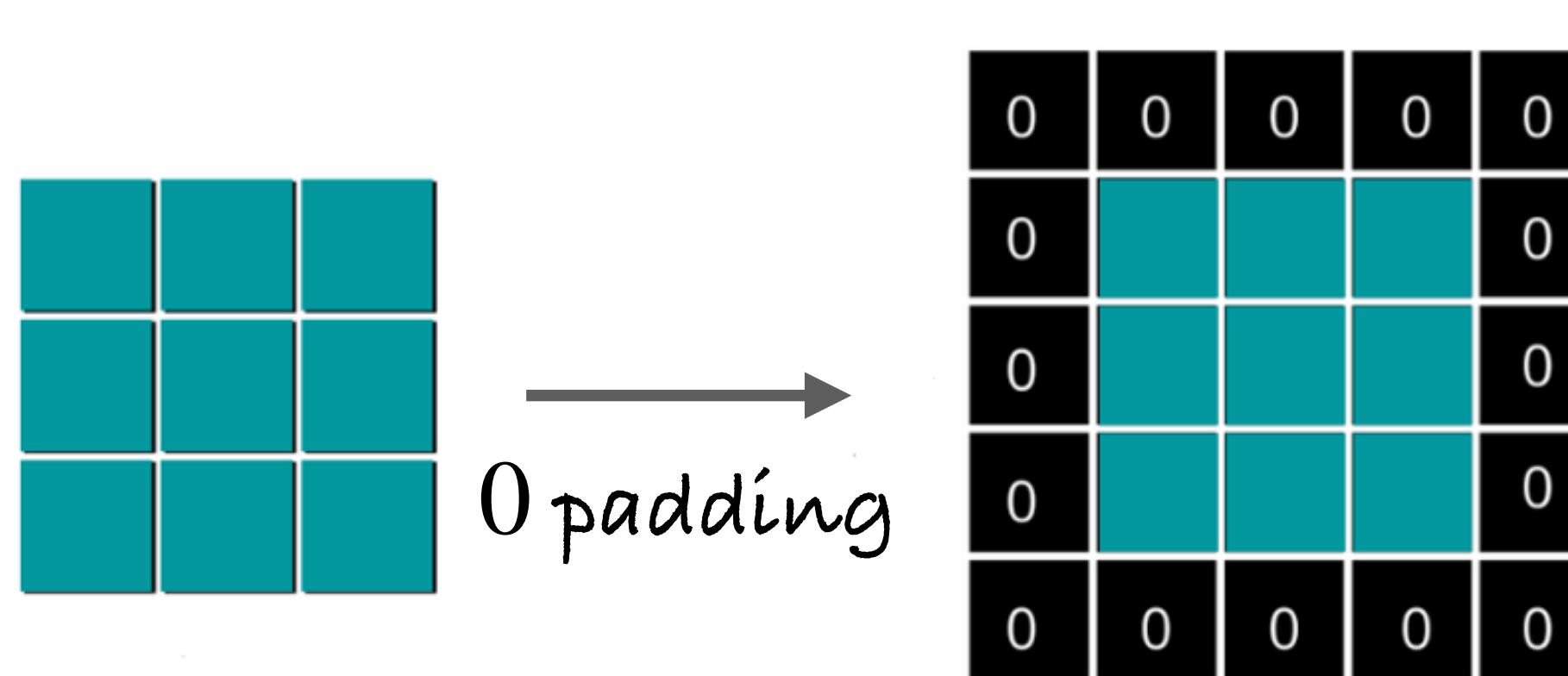
-1	-1	-1
1	1	1
-1	-1	-1

# Componenti di una CNN: stride e padding

Stride of 2 pixels:



**Stride:** numero di pixel di cui ci spostiamo quando facciamo scorrere il kernel

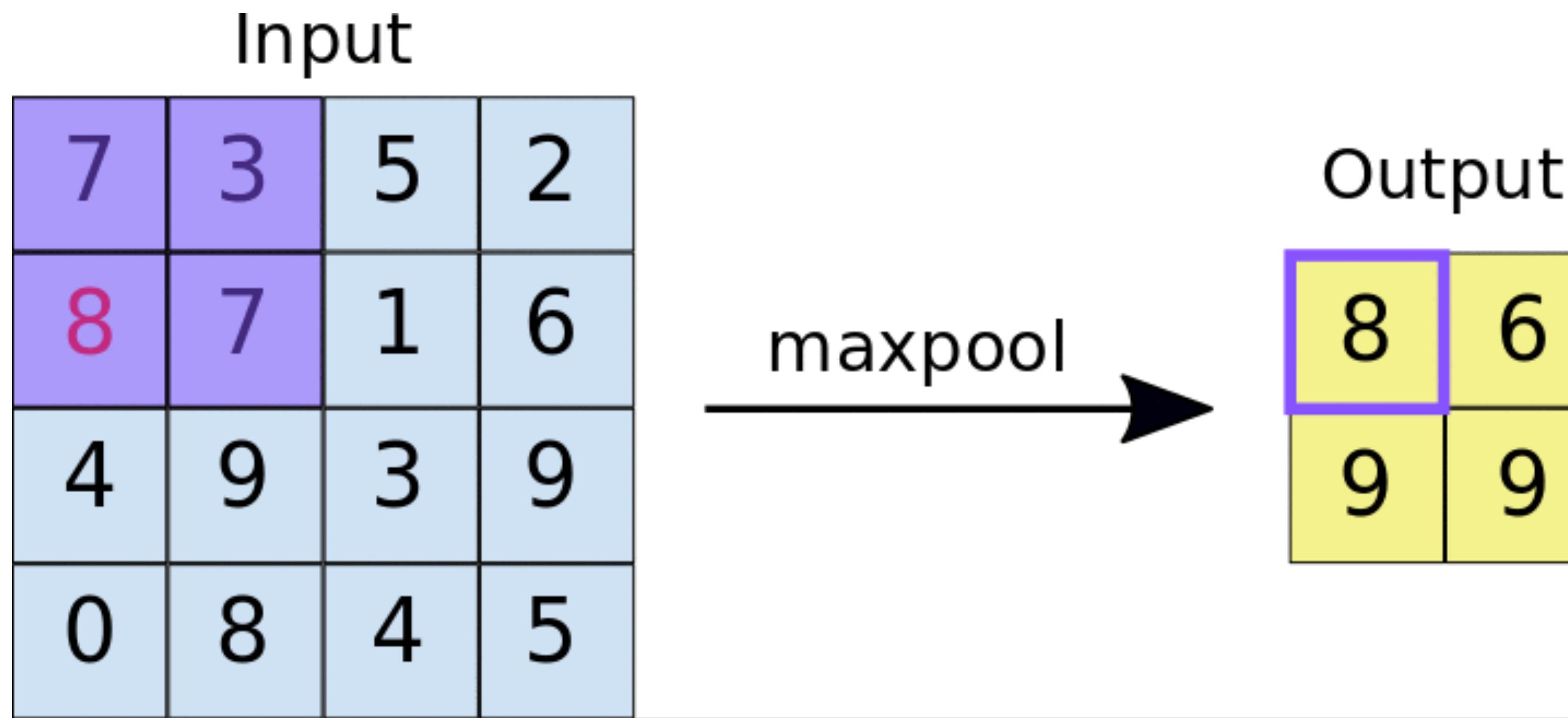


**Padding:** aggiunta di pixel ai bordi dell'immagine, principalmente per:

- Considerare tutti i pixel lo stesso numero di volte;
- Evitare di ridurre eccessivamente la dimensione dell'immagine.

# Componenti di una CNN: pooling

Il **pooling** serve a distillare le caratteristiche salienti del pattern imparato dal kernel.



Resilienza a piccole trasformazioni dell'input.

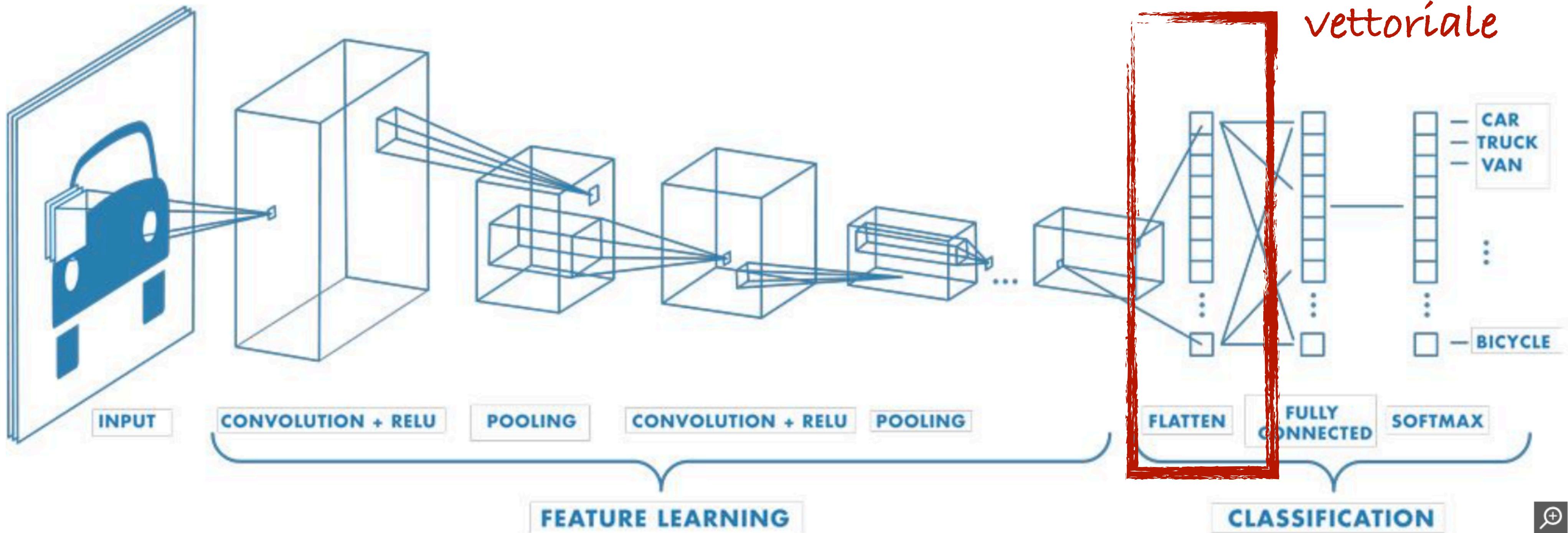
Riduzione del numero di parametri della rete.

I più tipi comuni di pooling sono il **massimo** e la **media aritmetica**.

Come il kernel, il pooling ha dimensione e stride.

# Componenti di una CNN: flattening

Perche ho bisogno di classificare, quindi mi serve un output vettoriale

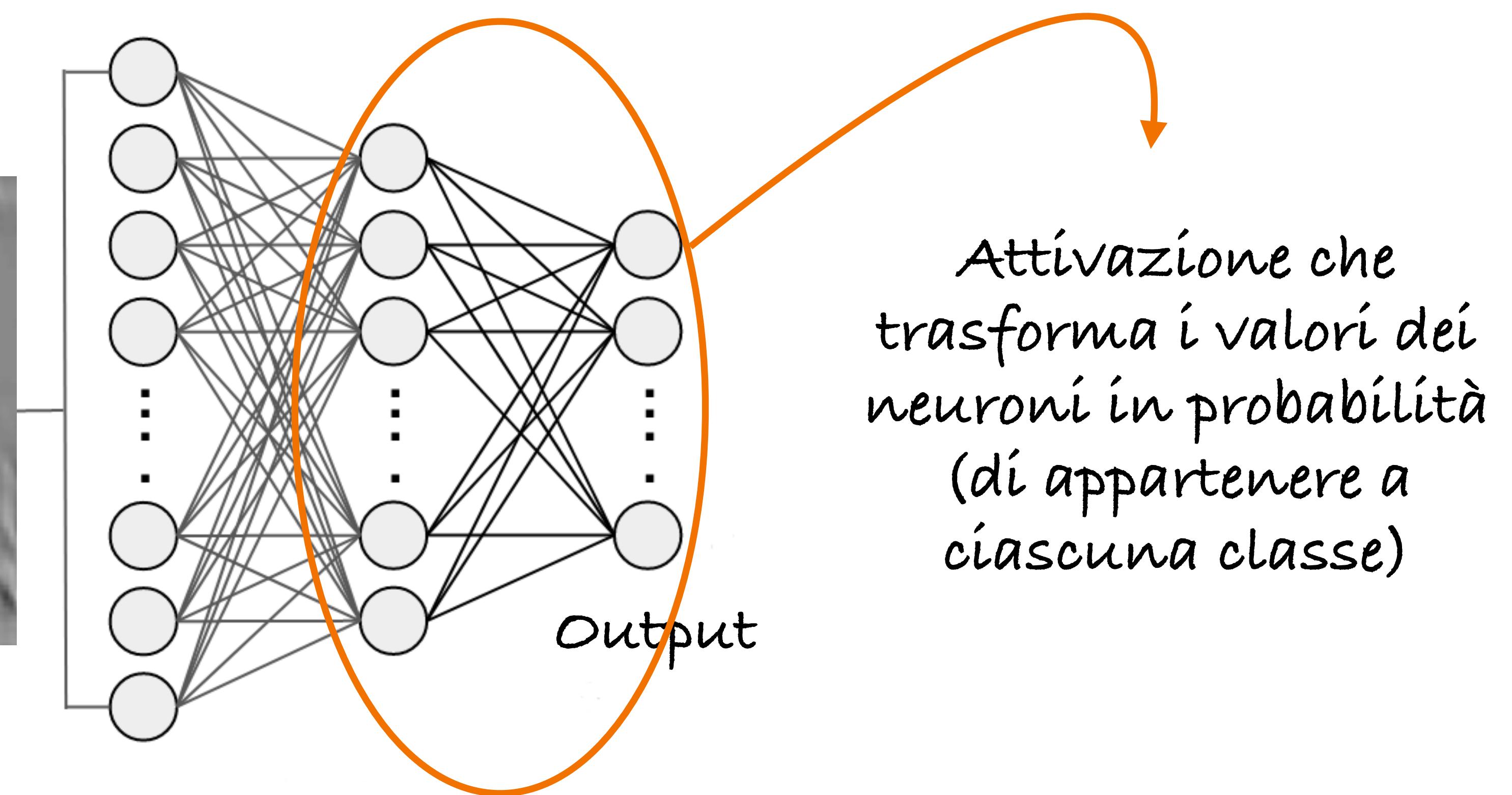


# Componenti di una CNN: strato di classificazione

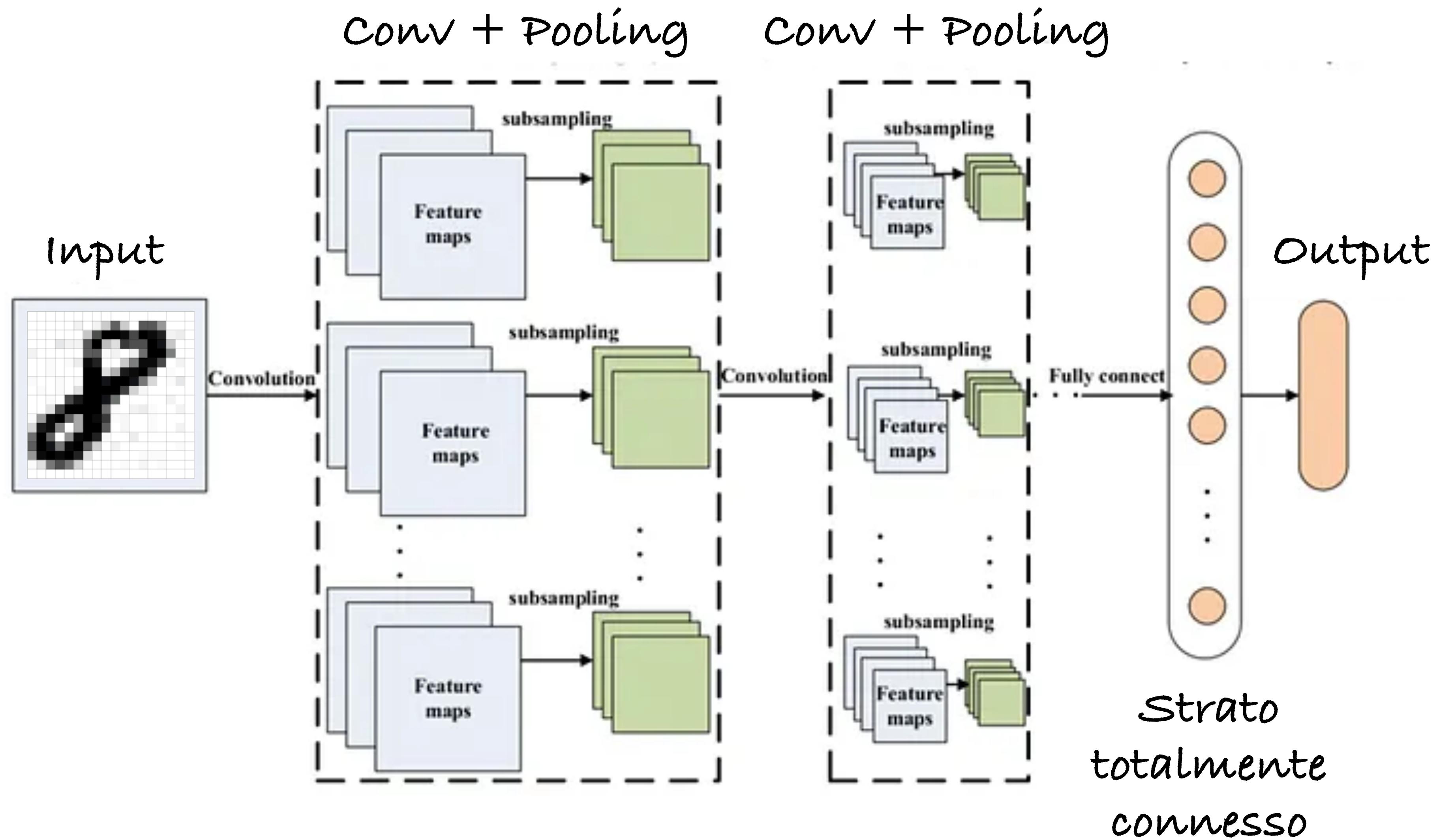
L'ultimo strato è quello permette di classificare l'immagine in input utilizzando le features estratte dagli strati convolutivi. Solitamente è implementato tramite uno strato densamente/completamente connesso.



Features estratte dalle  
convoluzioni



# Le operazioni in una CNN:



# Normalizzazione dei dati di input

La normalizzazione è il processo di standardizzazione dei dati di input per migliorare l'efficienza dell'addestramento delle reti neurali.

## Perché Normalizzare?:

- Riduce le discrepanze di scala tra le caratteristiche, permettendo alla rete di apprendere più velocemente.
- Aiuta a prevenire l'**overfitting** mantenendo i pesi della rete in un intervallo gestibile.

## Normalizzazione nelle Immagini:

- Per le immagini, tipicamente si divide ciascun pixel per 255 (il massimo valore di un pixel in scala di grigi o in RGB), convertendo i dati in un intervallo  $[0, 1]$

## Effetti Visivi:

- Questo processo non altera l'aspetto visivo delle immagini ma rende i dati più omogenei e facilmente interpretabili dalla rete.

# Confusion Matrix

		ACTUAL VALUES	
		Positive	Negative
PREDICTED VALUES	Positive	TP	FP
	Negative	FN	TN

The predicted value is positive and its positive

Type I error : The predicted value is positive but it False

Type II error : The predicted value is negative but its positive

The predicted value is Negative and its Negative

La matrice di confusione è una tabella che viene usata per descrivere le prestazioni di un modello di classificazione.

Gli elementi principali sono:

- 1) **VP** veri positivi
- 2) **FP** falsi positivi
- 3) **VN** veri negativi
- 4) **FN** falsi negativi