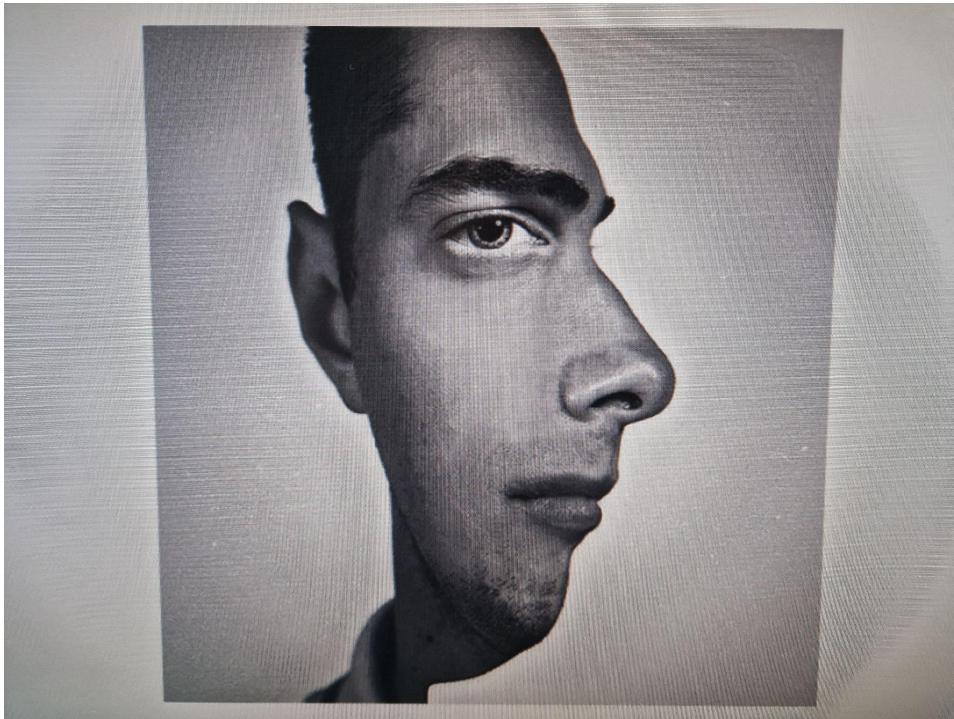


Redes Convolucionales

U-TAD

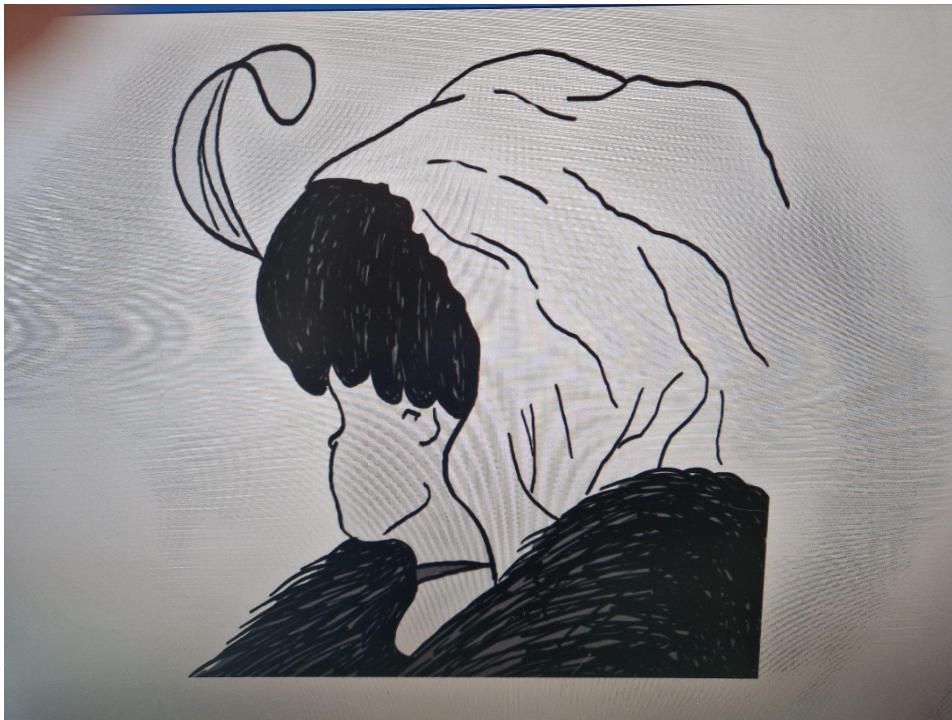
¿Qué es una red neuronal convolucional?



¿Está mirando al frente o a su izquierda?

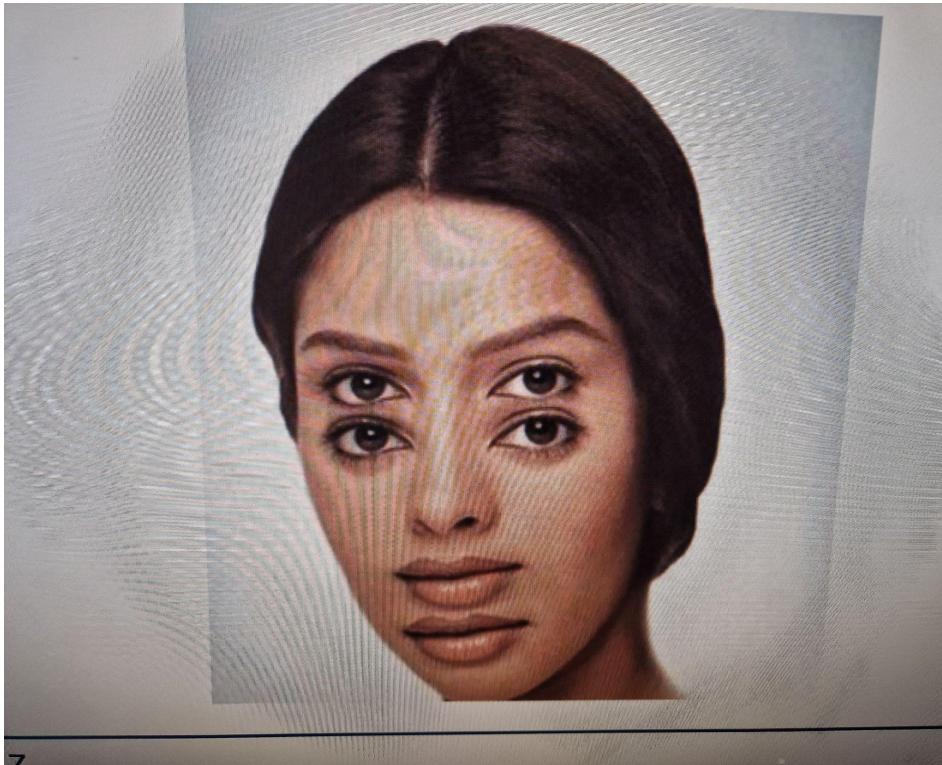
En función de dónde miremos podemos creer una cosa u otra. Si miramos al lado derecho nuestro cerebro verá, por ejemplo, la nariz y nuestro cerebro creerá que está mirando a un lado, y si miramos al centro, verá, por ejemplo, el ojo y creerá que mira de frente.

¿Qué es una red neuronal convolucional?



¿Joven mirando hacia atrás o anciana mirando hacia abajo?
En función de qué características procesa vuestro cerebro, veréis una u otra

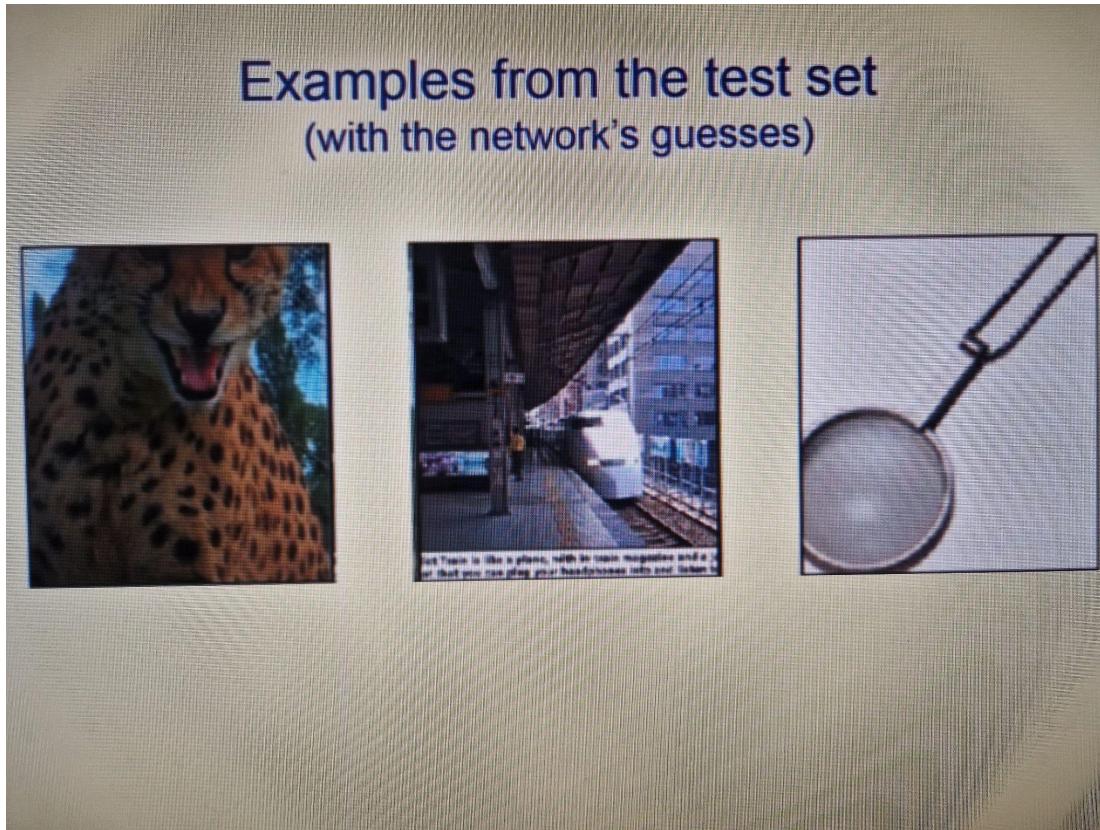
¿Qué es una red neuronal convolucional?



Y este es un ejemplo clásico de que ciertas características podrían estar ahí, son reales, pero el cerebro no es capaz de decidir qué ojo es correcto (puede llegar incluso a marear).

Con las RNAs convolucionales trataremos de hacer algo similar, tratar de simplificar las características para catalogar un objeto.

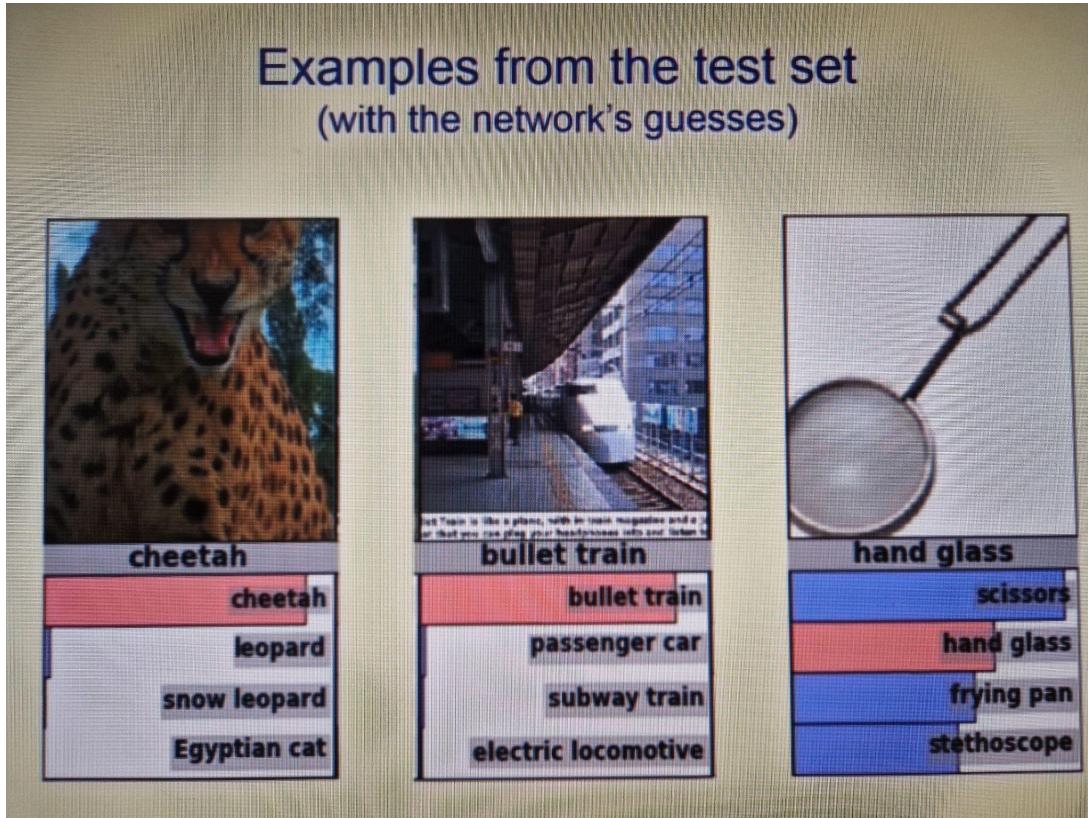
¿Qué es una red neuronal convolucional?



Experimento hecho en ordenador con una red neuronal evolucionada (diapositiva de Geoffrey Hinton, uno de los padres de las redes neuronales).

Tenemos un felino, un tren y algo difícil de catalogar. Se le suministra estas imágenes a un modelo y ¿qué predijo?

¿Qué es una red neuronal convolucional?



Las dos primeras (guepardo y tren bala), lo decide con una muy alta probabilidad. Y la última, que nos costaría a nosotros (es una lupa), lo predijo como tijera, pero no lo tenía demasiado claro.

La conclusión es que el modelo (del 2005/2006) está algo indeciso al igual que nosotros.

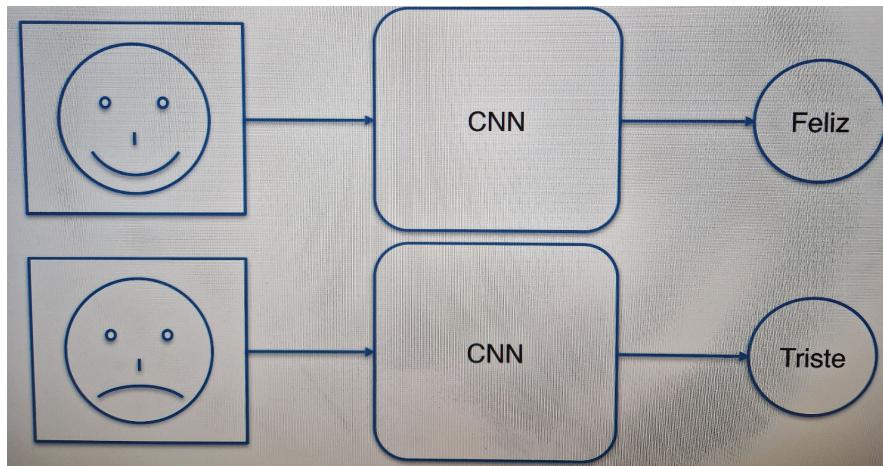
¿Qué es una red neuronal convolucional?

Han ganado gran popularidad de la IA.

Gracias a cosas como la conducción autónoma, reconociendo señales de tráfico y entendiendo en tiempo real qué ocurre a su alrededor.

También, las redes sociales son capaces de reconocer el rostro en fotos y asociarlo con el nombre entre tus contactos. Otro de los padres de la IA, Yann Lecun, ha trabajado para Facebook.

¿Qué es una red neuronal convolucional? Arquitectura



Nos devolverá una
probabilidad de Feliz o Triste

¿Qué es una red neuronal convolucional? Arquitectura

Imagen en B/N 2x2 px

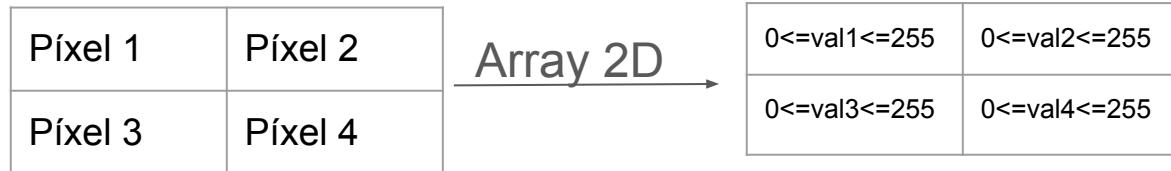
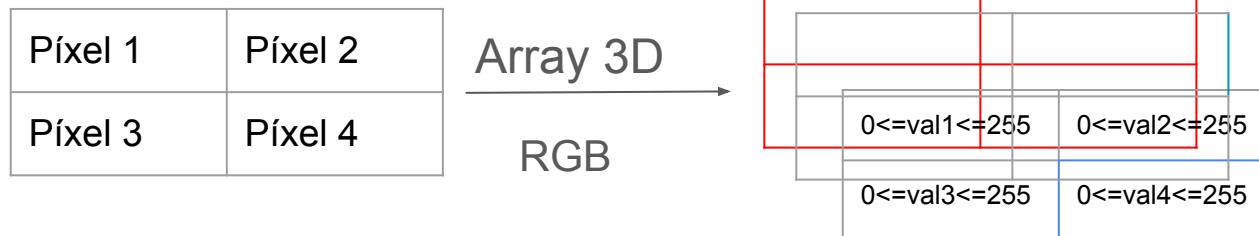


Imagen en Color 2x2 px



¿Qué es una red neuronal convolucional? Pasos

Paso 1: Convolución

Paso 2: Max Pooling

Paso 3: Flattering

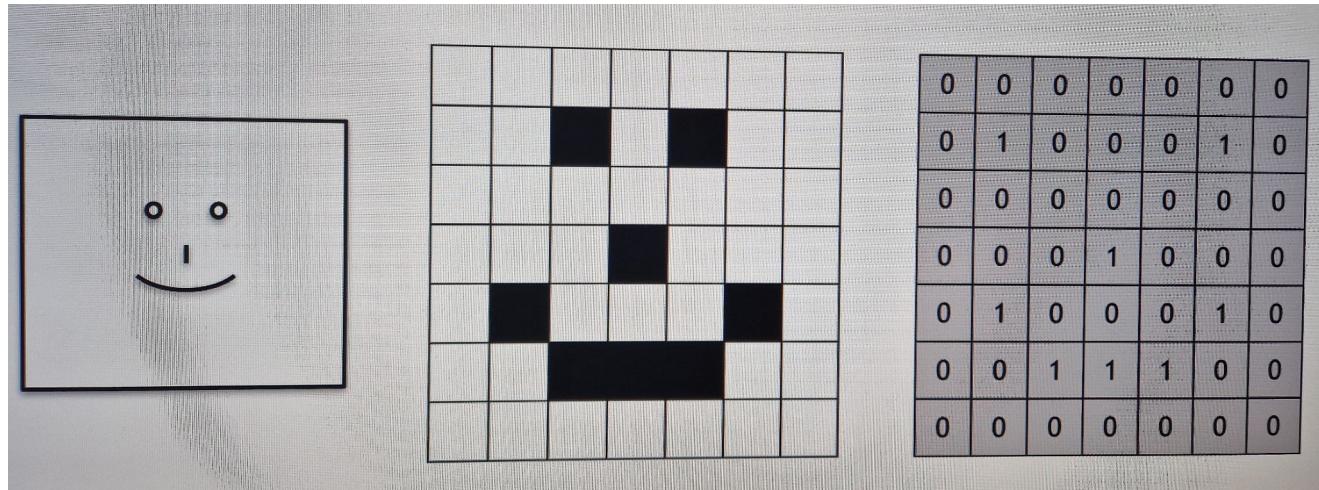
Paso 4: Full Connection

Lectura adicional:

[Gradient-Based Learning](#)

[Applied Document Recognition](#)

[By Yann LeCun \(1998\)](#)



Paso 1 - Convolución

Fórmula: $(f * g)(t) \doteq \int_{-\infty}^{\infty} f(\eta)g(t - \eta)d\eta$

Una convolución es una integral (dada una función modifica la otra).

Extra: *Introduction to Convolutional Neural Networks by Jianxin Wu (2017)*

Imagen de entrada

0	0	0	0	0	0	0
0	1	0	0	0	1	0
0	0	0	0	0	0	0
0	0	0	1	0	0	0
0	1	0	0	0	1	0
0	0	1	1	1	0	0
0	0	0	0	0	0	0

X Detectores de Rasgos

(mayor dimensión, con imágenes más grandes)

0	0	1
1	0	0
0	1	1

(Creamos varios mapas de características para obtener así la primera capa de convolución)

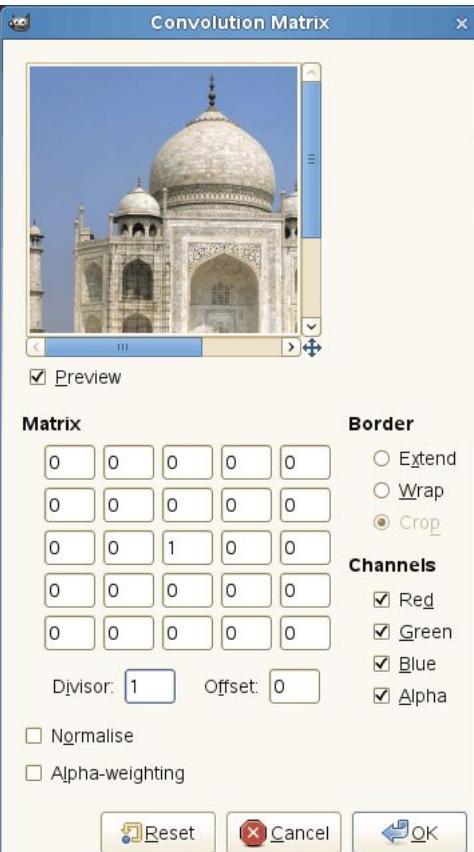
= Mapa de características

(resume la información)

0	1	0	0	0
0	1	1	1	0
1	0	1	2	1
1	4	2	1	0
0	0	1	2	1

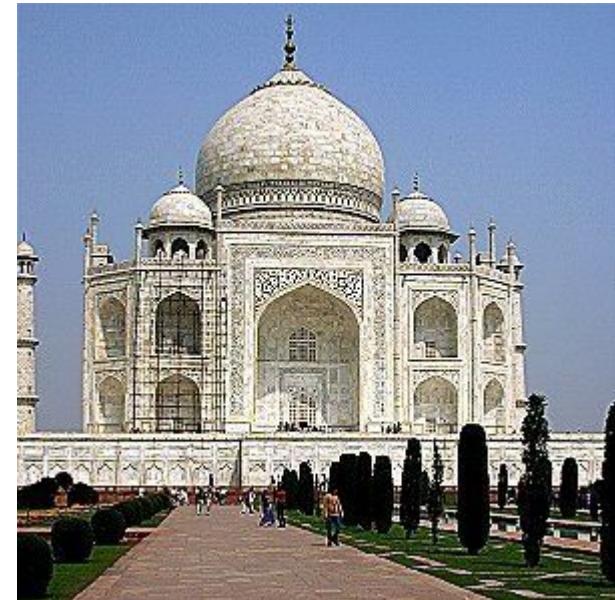
Paso 1 - Convolución

<https://docs.gimp.org/2.8/en/plug-in-convmatrix.html>



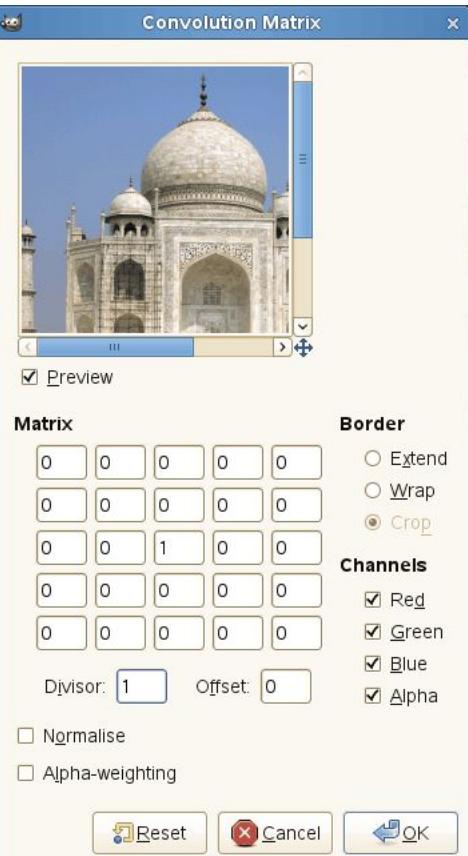
Sharpen:

0	0	0	0	0
0	0	-1	0	0
0	-1	5	-1	0
0	0	-1	0	0
0	0	0	0	0



Permite agudizar los contornos, es decir, agudizar la presencia de objetos en la imagen

Paso 1 - Convolución



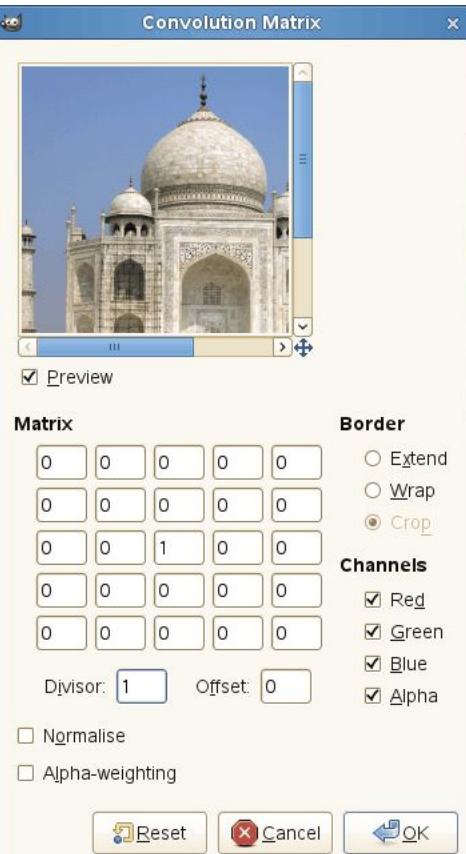
Blur:

0	0	0	0	0
0	1	1	1	0
0	1	1	1	0
0	1	1	1	0
0	0	0	0	0



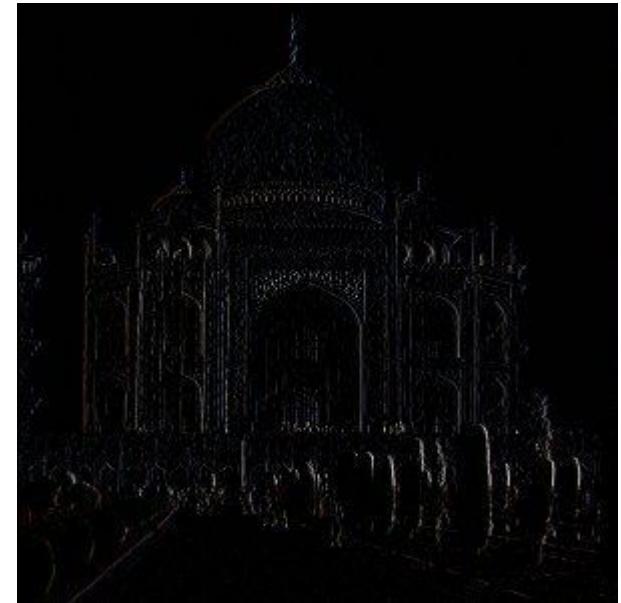
Difumina la imagen

Paso 1 - Convolución



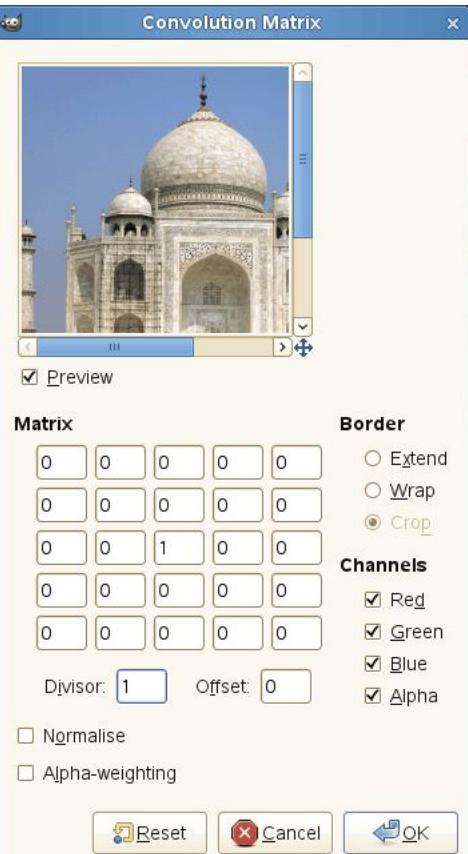
Edge Enhance:

0	0	0		
-1	1	0		
0	0	0		



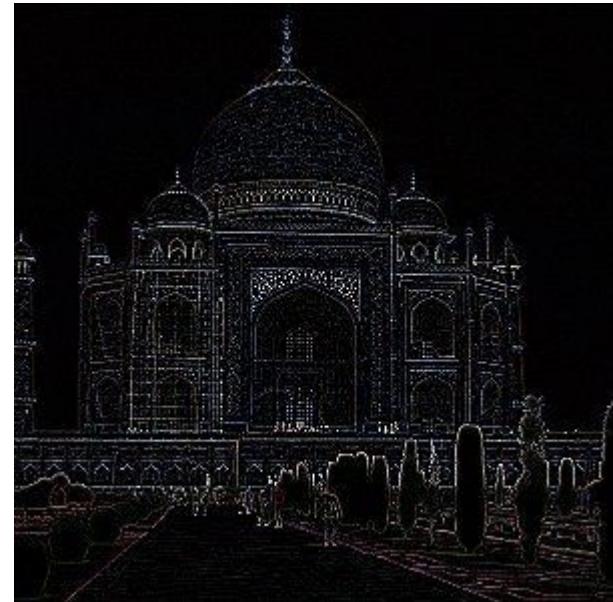
Mejora de los límites de la imagen. A cada píxel le elimina su vecino horizontal anterior, quedando las formas verticales

Paso 1 - Convolución



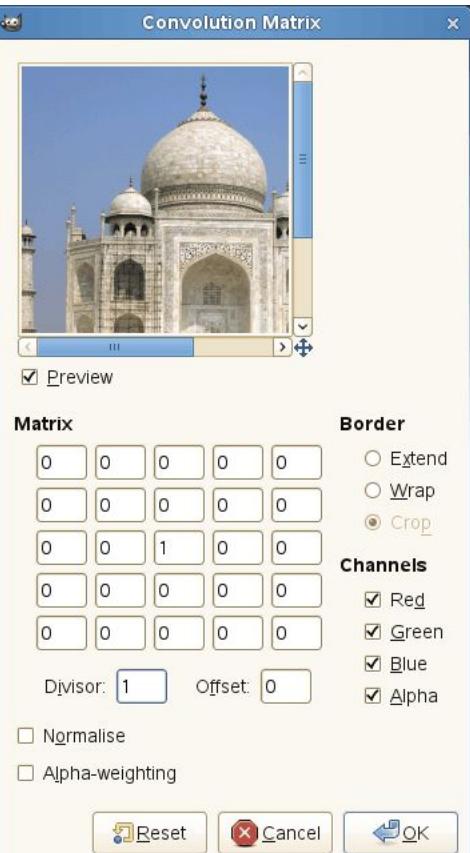
Edge Detect:

0	1	0		
1	-4	1		
0	1	0		



Detecta los bordes. Hace lo mismo pero en horizontal y vertical: dado un píxel elimina todas las vecinas Norte, Sur, Este y Oeste.

Paso 1 - Convolución



Emboss:

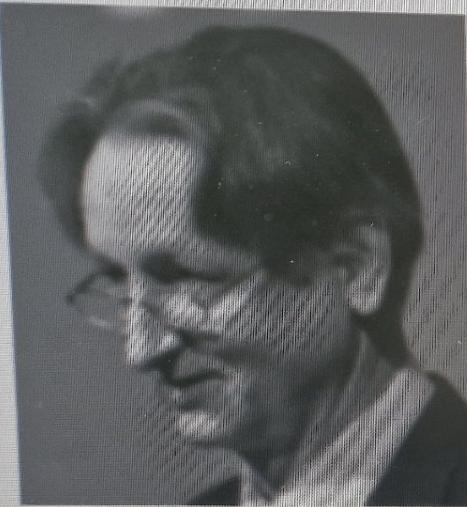
	-2	-1	0	
	-1	1	1	
	0	1	2	



Intenta mejorar los píxeles centrales y de derecha abajo restándole los anteriores. En definitiva, **cada uno de los filtros permite extraer información que la red neuronal sea capaz de buscar (formas, bordes, etc), encargándose de detectar o explicar una parte u otra de la imagen.**

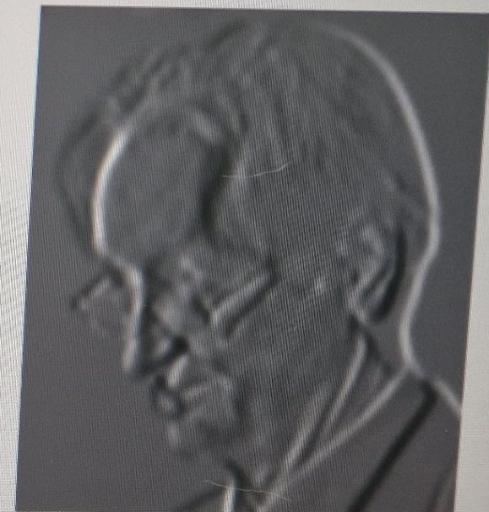
Paso 1 - Convolución

Ejemplo del propio Jeffry Hinton que pasó por uno de estos filtros permitiendo detectar rasgos característicos de él, como las gafas o el pelo



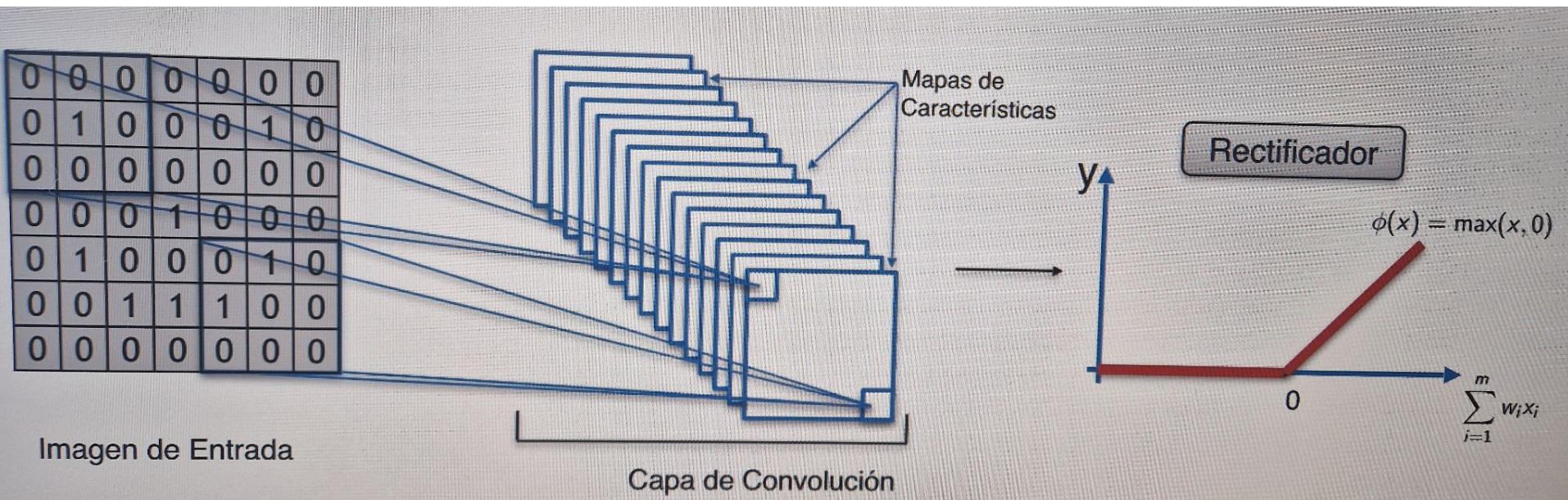
*

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



Paso 1.b - Capa de ReLU (Rectificadora Lineal Unitaria)

A la capa de convolución se le aplica ReLU:



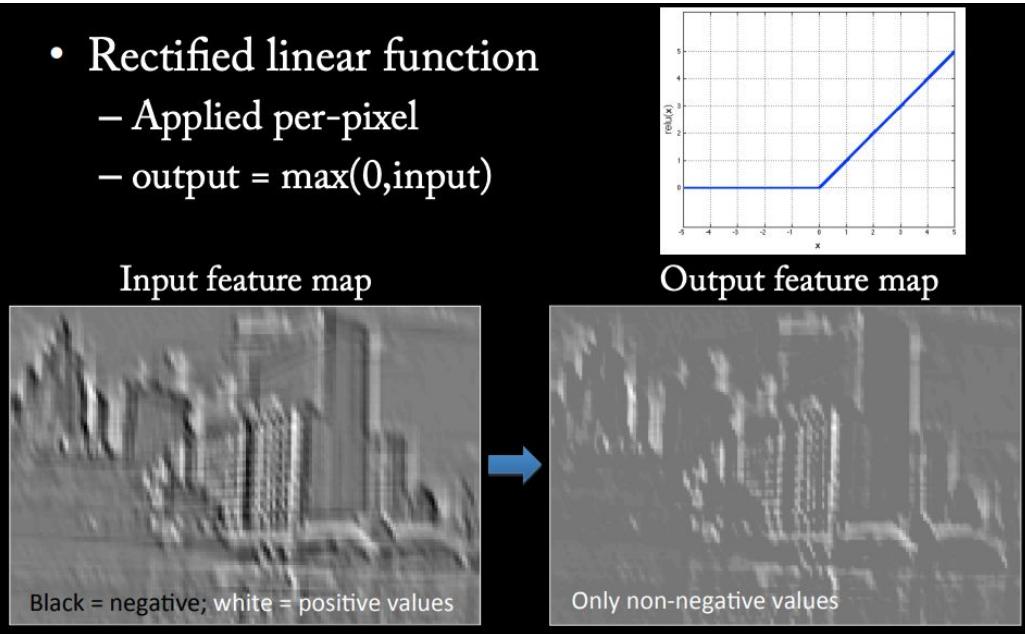
El filtro es una operación lineal de multiplicaciones y sumas, y con ReLU, rompe la linealidad dentro de nuestra red neuronal porque en los rasgos de las imágenes son altamente no lineales (la imagen contendrá muchos contrastes y bordes) y querremos acentuarlos. Es por esto que para hacer diverger (separar) las fronteras entre elementos cercanos dentro de la imagen podemos aplicar este rectificador lineal unitario

Paso 1.b - Capa ReLU

http://mlss.tuebingen.mpg.de/2015/slides/fergus/Fergus_1.pdf



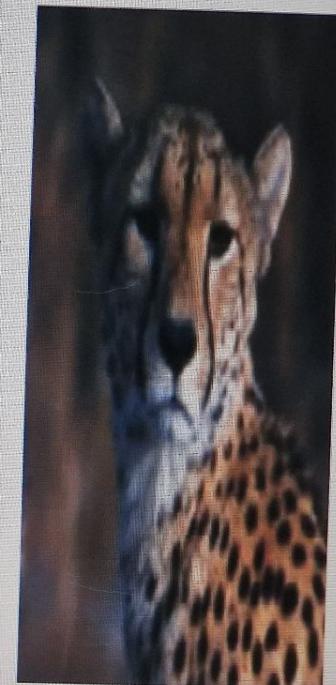
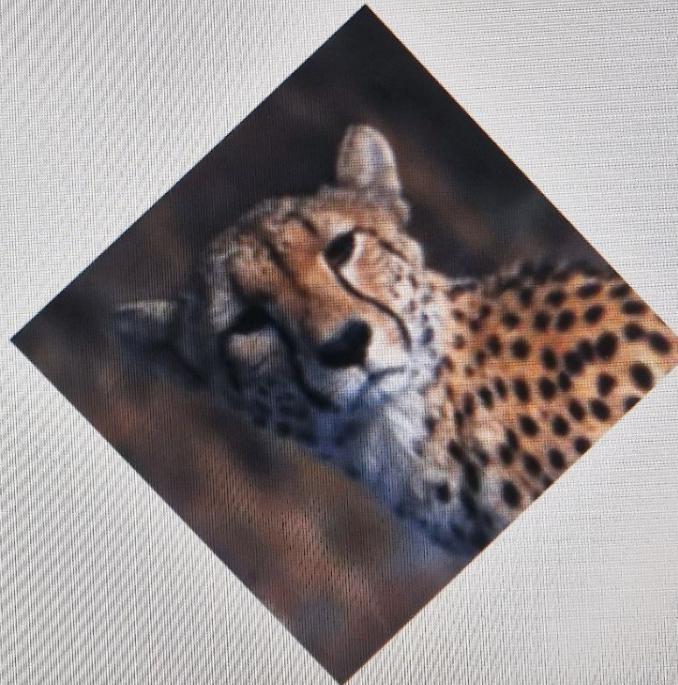
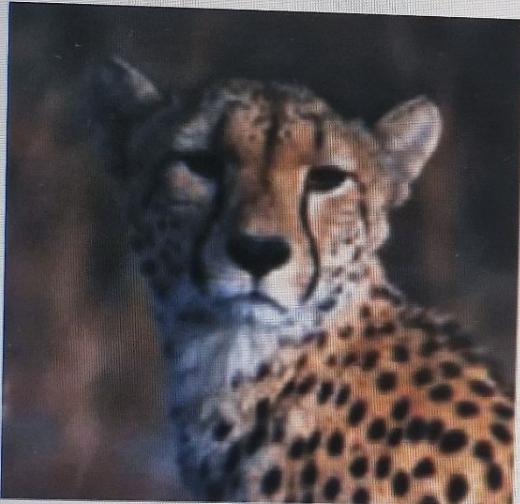
Input



Con ReLU se elimina la linealidad. Fijaros en el edificio del centro, donde sobresale la parte central que después se convierte en una sombra en el “input feature map” y, al aplicar ReLU, corrige el exceso de valores negativos tal que el negro intenso pasará a ser un gris neutro (valor nulo) de modo que ya no parece dos edificios separados como en la imagen intermedia. Al final, ayuda a la convergencia y a la detección de rasgos dentro de las redes convolucionales

Paso 2 - Max Pooling

Con la misma imagen transformada:



Paso 2 - Max Pooling

Y nuevas imágenes de guepardo en distintas posiciones, trataré de identificar ciertos rasgos: como los ojos color avellana, las manchas de color negro, etc.

Es decir, en base a ciertas características con distintas sombras y brillos, deberá generalizar y detectar guepardos en las imágenes donde los haya.

Esto nos lo va a facilitar el método de agrupación máxima (max pooling)



Paso 2 - Max Pooling

Mapa de características

0	1	0	0	0	
0	1	1	1	0	
1	0	1	2	1	
1	4	2	1	0	
0	0	1	2	1	

Mapa de características Pooled

1	1	0
4	2	1
0	2	1

Max Pooling

(resumen de la información)

De cada subconjunto de, por ejemplo, 2x2, obtengo el máximo valor de los 4

El algoritmo de max pooling preserva los máximos de la figura original, en posiciones similares

Paso 2 - Max Pooling

Las manchas del guepardo, los ojos, etc, un poco más a la izquierda o a la derecha, un poco más o menos girados, serán corregidas con una ventana de Max Pooling acorde al tamaño de la imagen

Intentamos agrupar la información lo máximo posible preservando las características y teniendo en cuenta distorsiones espaciales, rotaciones o cambios de escala con respecto a la imagen, a la vez que reducimos el tamaño. Por tanto, esto nos ayudará en términos de procesamiento a la hora de suministrar píxeles a la red neuronal final

Además de simplificar, ayuda a evitar el overfitting.

Se puede resumir en que desecha la información no relevante. Se puede agrupar de distintos modos cambiando el tamaño de ventana (o haciendo el promedio en vez de quedarnos con el máximo)

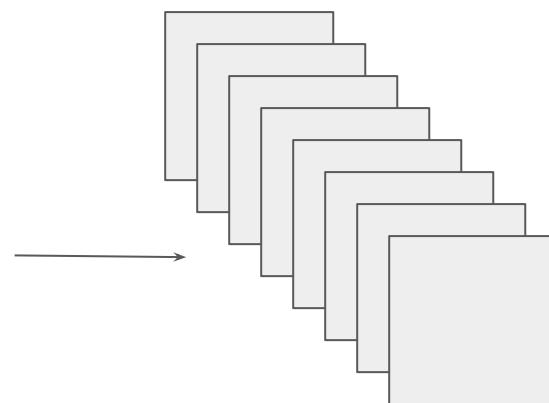
Extra: *Evaluation of Pooling Operations in Convolutional Architectures for Objects Recognition by Dominik Scherer (2010)*

Paso 2 - Max Pooling

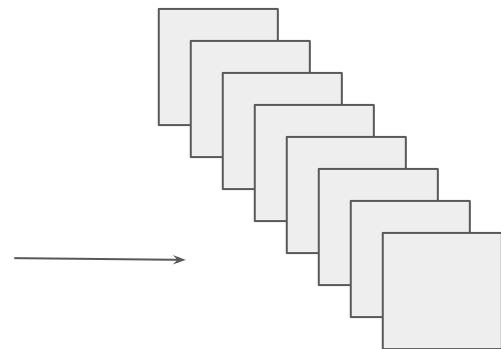
Imagen de Entrada

0	0	0	0	0	0	0
0	1	0	0	0	1	0
0	0	0	0	0	0	0
0	0	0	1	0	0	0
0	1	0	0	0	1	0
0	0	1	1	1	0	0
0	0	0	0	0	0	0

Capa de convolución



Capa de Pooling



Paso 3 - Flattering (operación de aplanado)

Mapa de características pooled

1	1	0
4	2	1
0	2	1

Flattering



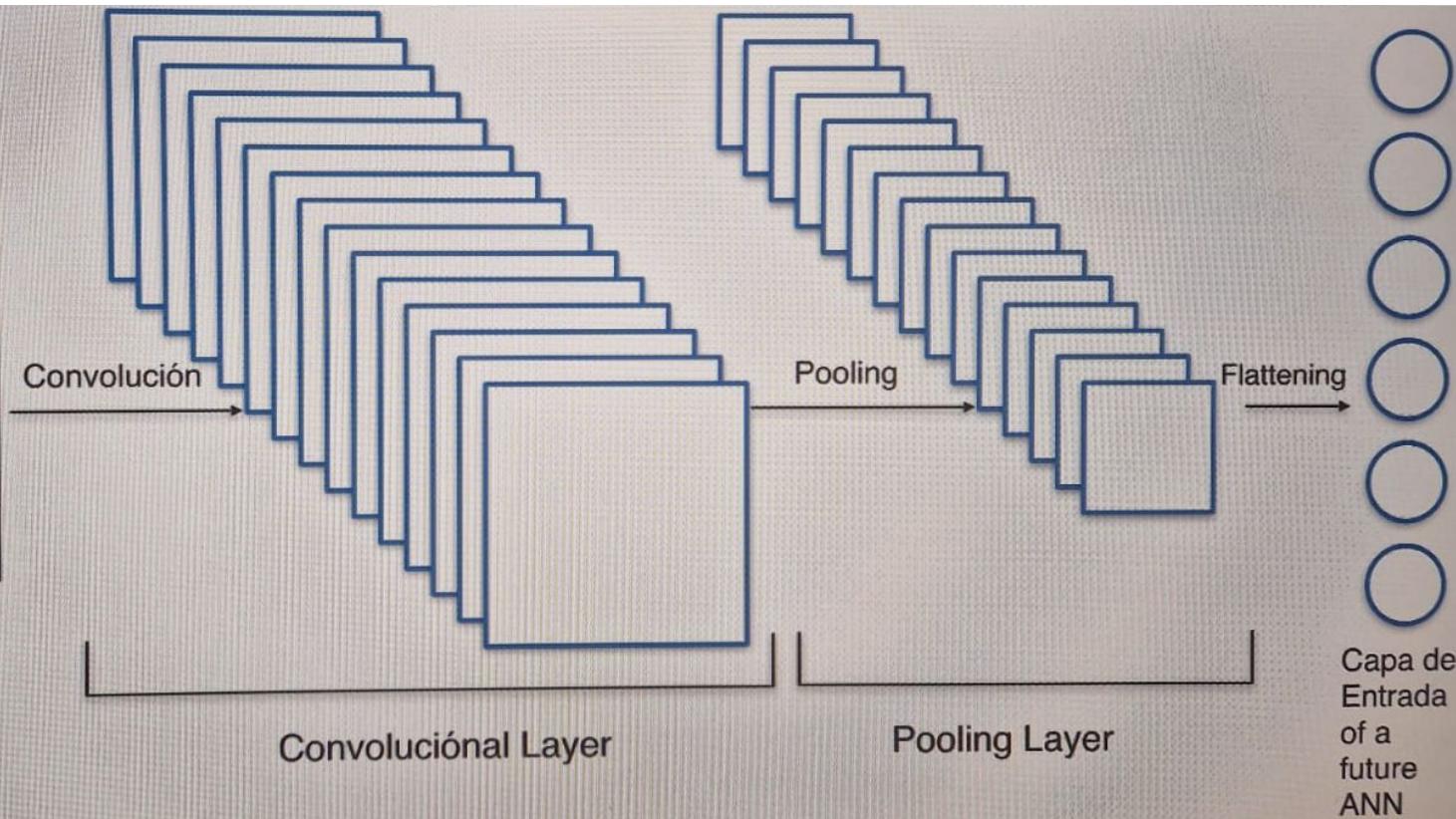
1
1
0
4
2
1
0
2
1

Capa de entrada de la
RNA

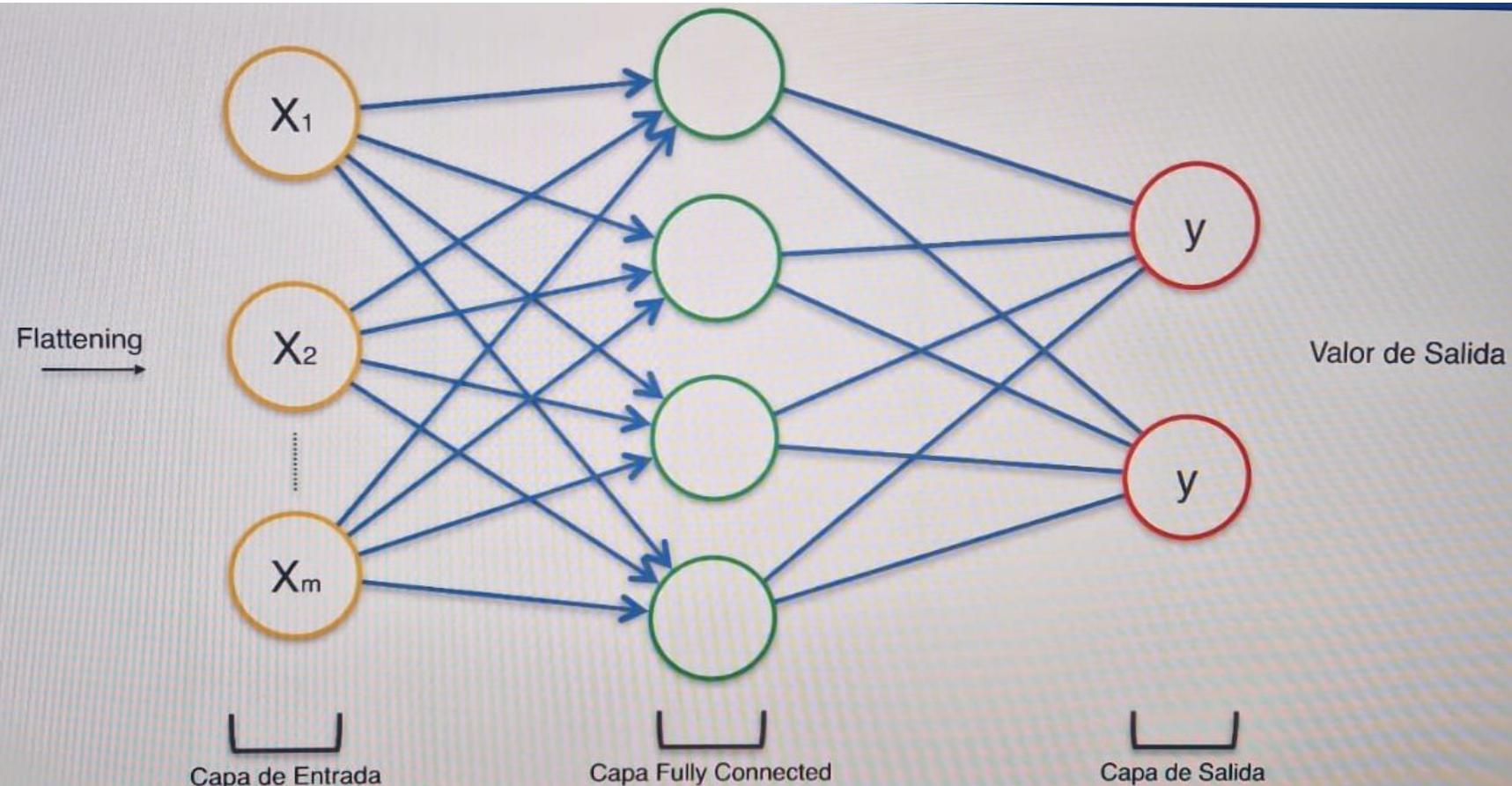
Paso 3 - Flattering

0	0	0	0	0	0	0	0
0	1	0	0	0	1	0	
0	0	0	0	0	0	0	
0	0	0	1	0	0	0	
0	1	0	0	0	1	0	
0	0	1	1	1	0	0	
0	0	0	0	0	0	0	

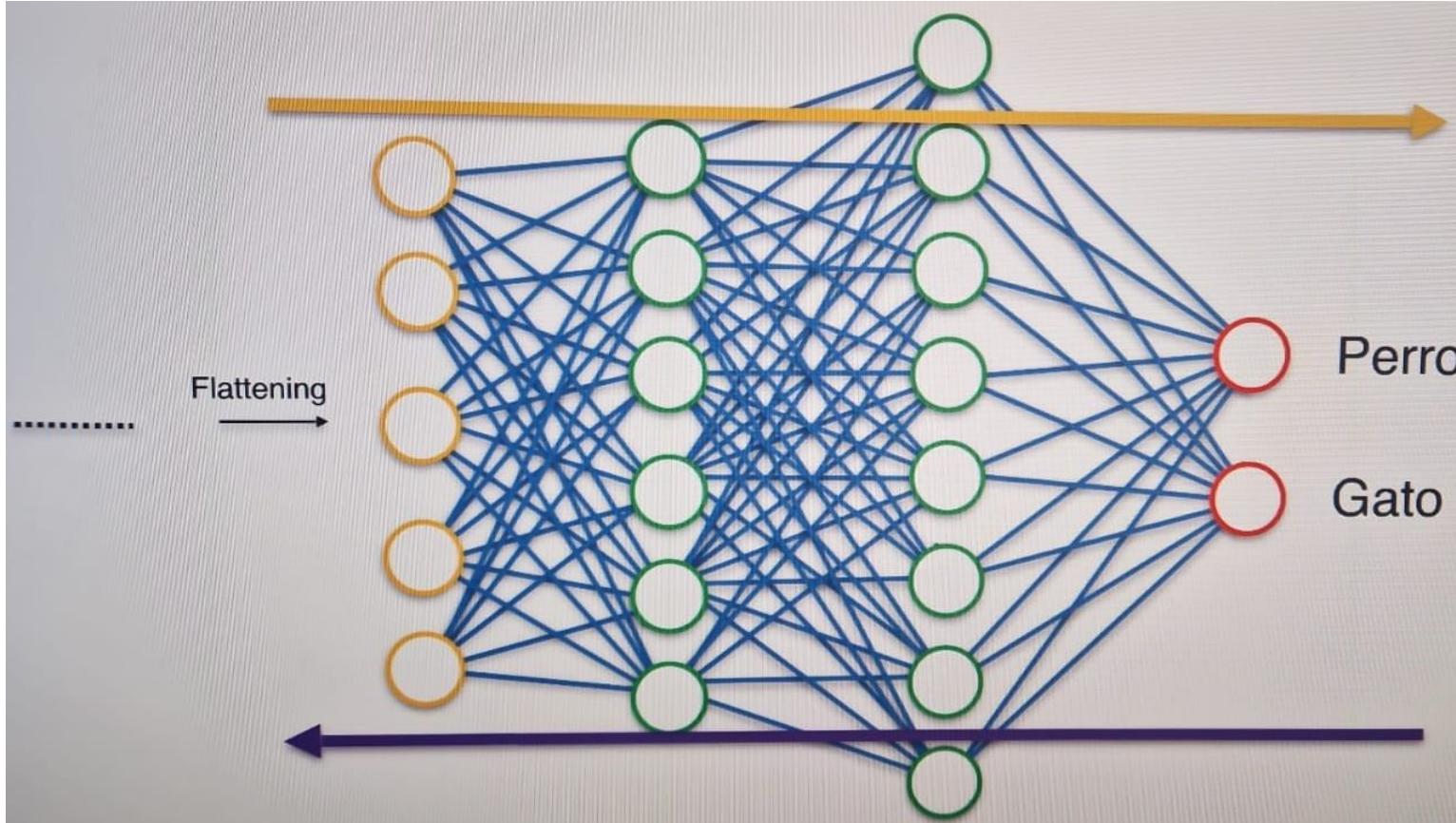
Imagen de Entrada



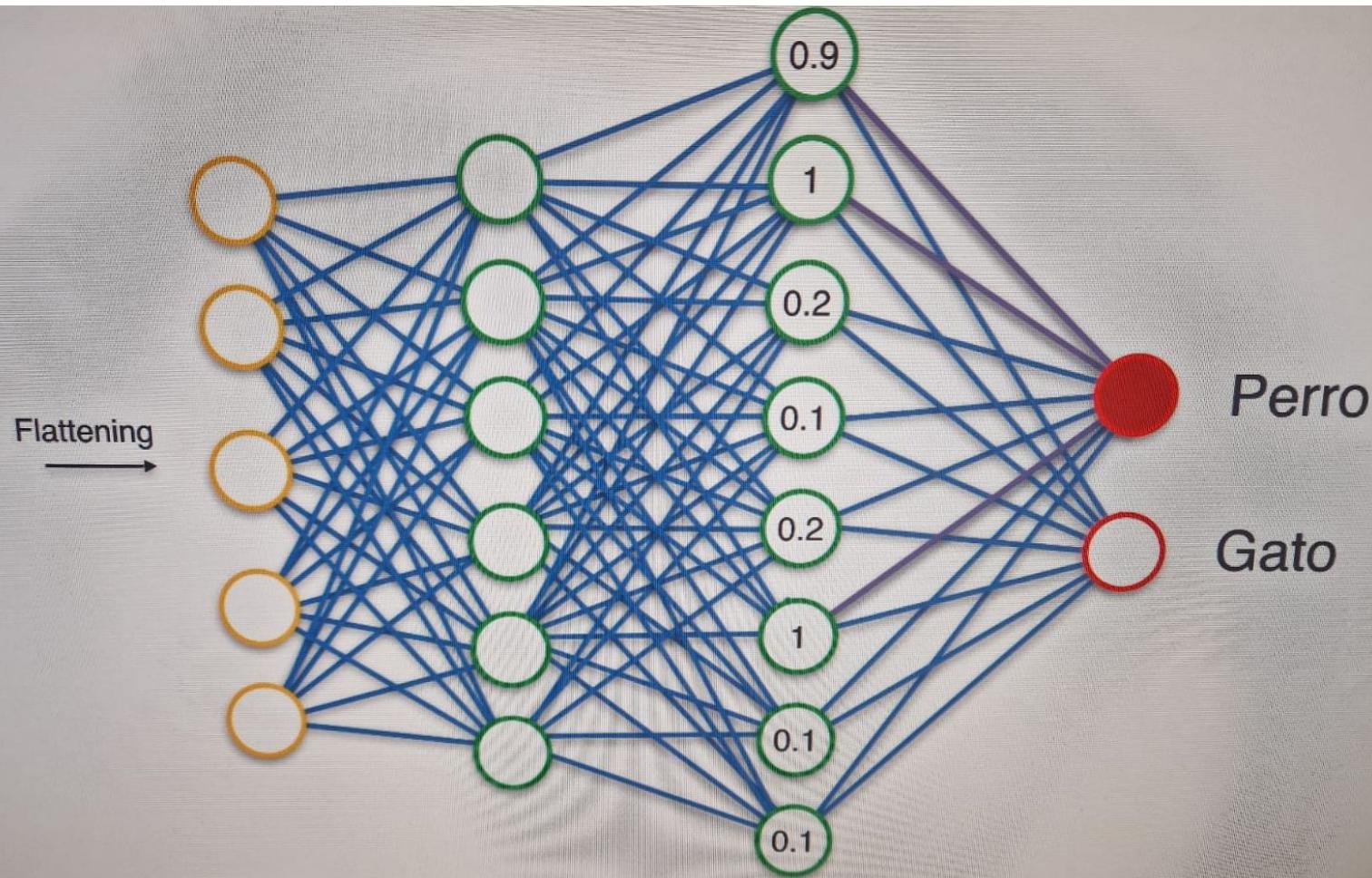
Paso 4 - Full Connection (Capa totalmente conectada)



Paso 4 - Full Connection

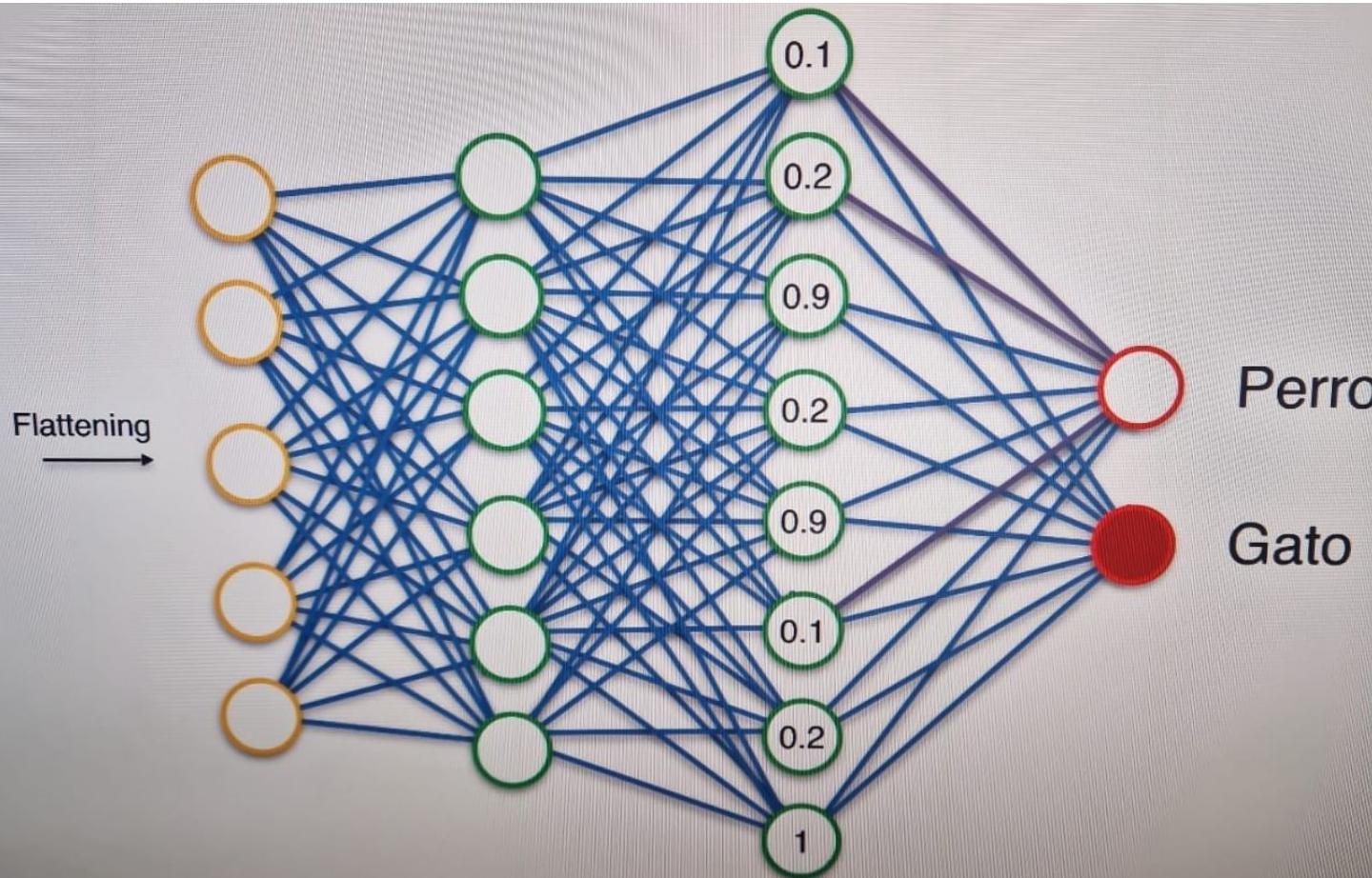


Paso 4 - Full Connection



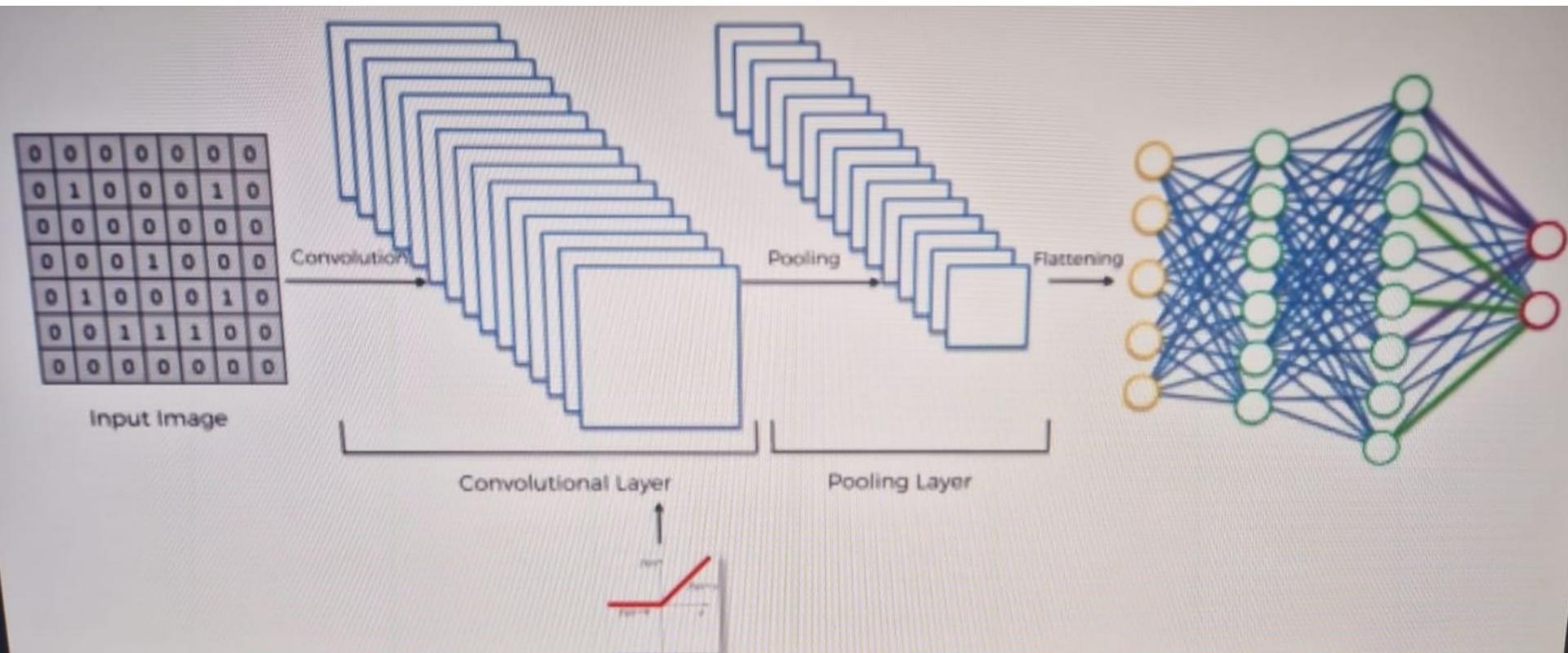
Por ejemplo,
La primera,
segunda y
cuarta
neurona
habrán
detectado
cierta
característica
de perro,
como las
cejas, con
alta
probabilidad.
Y las bajas,
quizás
indicarán
característica
de gatos

Paso 4 - Full Connection



Cada neurona se especializa en detectar una forma, rasgo de perro o rasgo de gato. Y la capa de salida es la que devolverá el resultado en función de la mayor probabilidad

Resumen

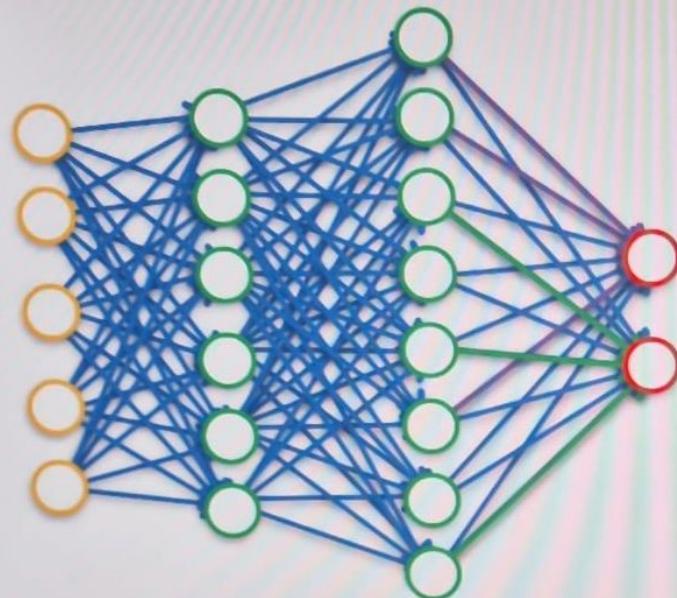


Softmax y cross-entropy

Softmax:



.....
Flattening



$$f_j(z) = \frac{e^{z_j}}{\sum_k e^{z_k}}$$

↓
Perro $\longrightarrow z_1 \longrightarrow 0.95$
Gato $\longrightarrow z_2 \longrightarrow 0.05$

Softmax y cross-entropy

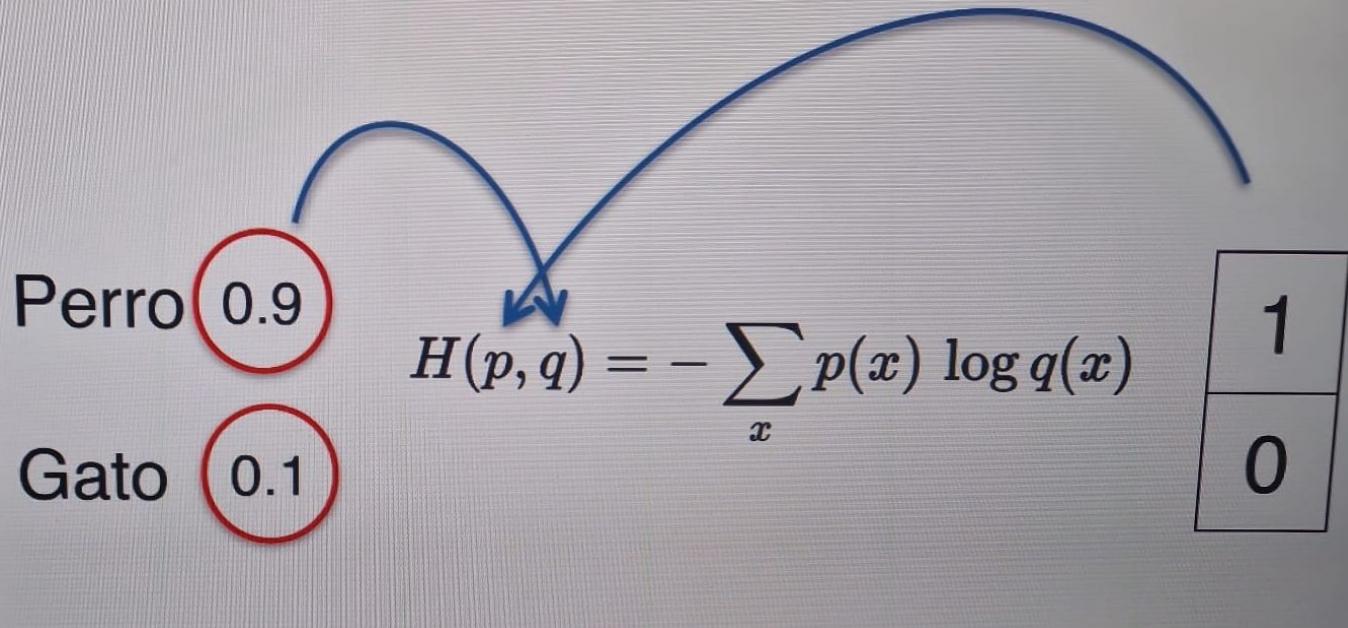
Función de coste o error: entropía-cruzada (mejora al minimizar el error):

$$L_i = - \log \left(\frac{e^{f_{y_i}}}{\sum_j e^{f_j}} \right)$$

$$H(p, q) = - \sum_x p(x) \log q(x)$$

Softmax y cross-entropy

Cross-entropy: El valor real por el logaritmo de la probabilidad devuelta



Ejercicio: Construir una Red Neuronal de Convolucional

- 1) Pre-procesado de los datos
- 2) Construir la RNC
- 3) Entrenar y evaluar la RNC

Cuestionario

1 ¿Cuál es la función de activación más comúnmente utilizada en las capas convolucionales cuando trabajamos con imágenes?

- 1.Sigmoide
- 2.ReLU
- 3.Softmax

2 ¿Cómo se añade una capa convolucional a un modelo con TensorFlow 2.0?

- 1. model.add(tf.keras.optimizers.Conv2D())
- 2.model.add(tf.keras.layers.Conv2D())
- 3.model.add(tf.jeras.models.Conv2D())

3 ¿Qué capa se utiliza después de la última capa convolucional en una Red Neuronal Convolucional tradicional?

- 1. tf.keras.layers.Dense()
- 2. model.compile()
- 3. tf.keras.layers.Flatten()

4 ¿Qué son los *strides*?

1. Es un argumento de la capa de convolución que define el número de píxeles que el kernel puede moverse en cualquier dirección analizando la capa de entrada
2. Es un argumento de la capa de convolución que define el número de píxeles del kernel de dicha capa