

# Inteligencia Artificial II

Juan Pablo Restrepo Uribe

Ing. Biomedico - MSc. Automatización y Control Industrial

[jprestrepo@correo.iue.edu.co](mailto:jprestrepo@correo.iue.edu.co)

2023

Institución Universitaria de Envigado

# Imágenes

Una imagen es una representación visual, que manifiesta la apariencia visual de un objeto real o imaginario.



Algunas de ellas son el dibujo, el diseño, la pintura, la fotografía o el vídeo, entre otras.

# Imágenes

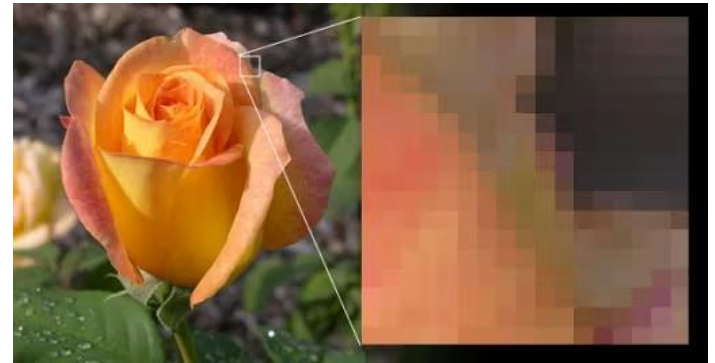
Las imágenes de **mapa de bits** cuentan con una retícula perfectamente definida y, por tanto, su calidad se mantiene fija. Si se amplía o disminuye la imagen, vemos como los píxeles que la forman se multiplican o dividen, lo que provoca una pérdida de calidad. Cualquier programa en el cual podamos trabajar con imágenes para retocarlas, por ejemplo, Adobe Photoshop

Las **imágenes vectoriales** al estar basadas en fórmulas matemáticas tienen una resolución infinita, y por tanto se pueden ampliar o reducir sin riesgo de que su calidad baje. Cualquier programa que sirva para dibujar, por ejemplo, Adobe Illustrator

# Imágenes

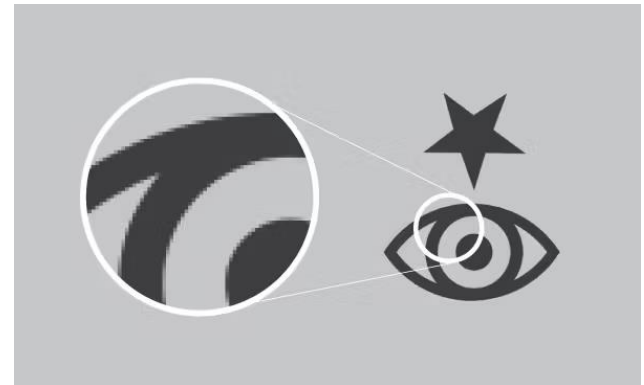
## Formatos de imagen para los mapas de bits

- Formato JPG
- Formato PNG
- Formato GIF
- Formato TIFF
- Formato PSD
- Formato BMP



## Formatos de imagen vectoriales

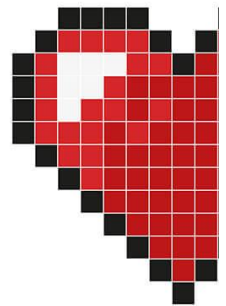
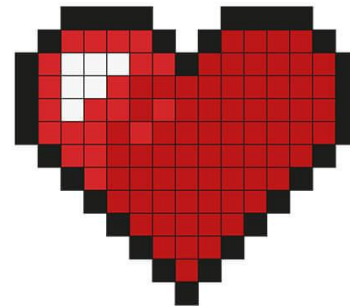
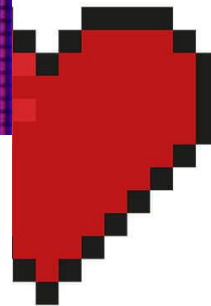
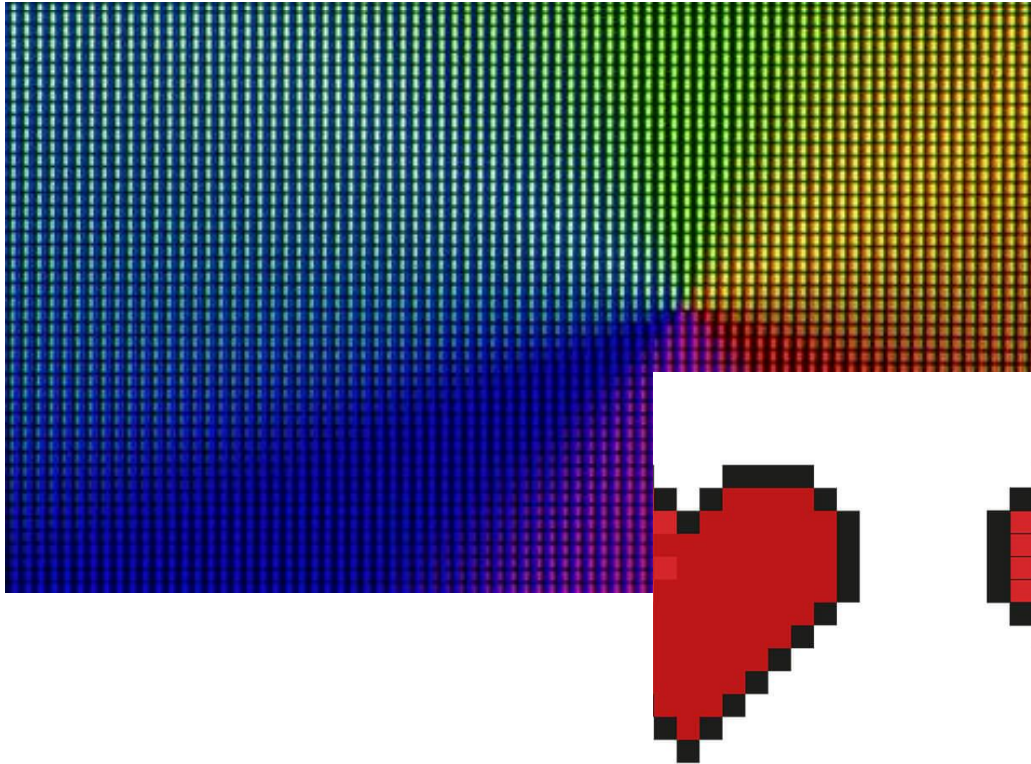
- Formato EPS
- Formato SVG



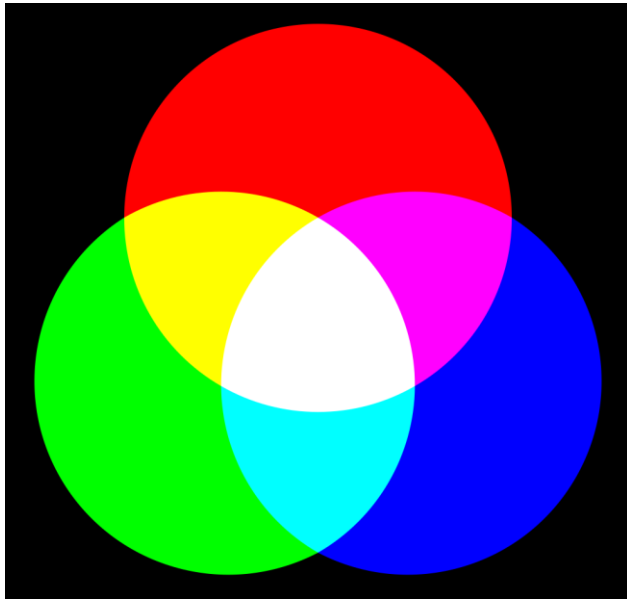
# Imágenes



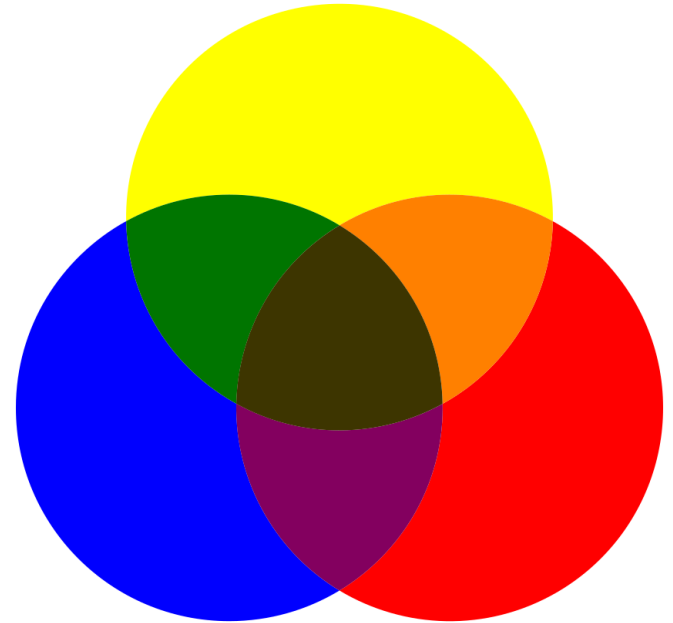
# Pixel



## Colores primarios

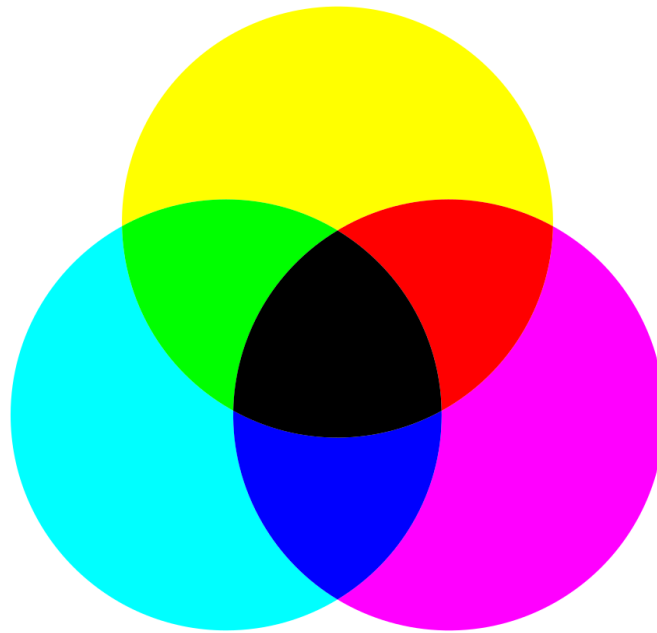


**RGB**



**RYB**

# Colores primarios



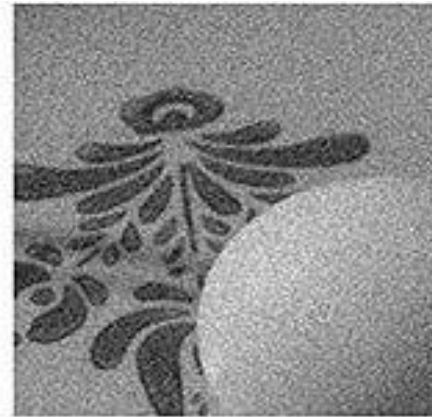
**CMY**



# Procesamiento de imágenes



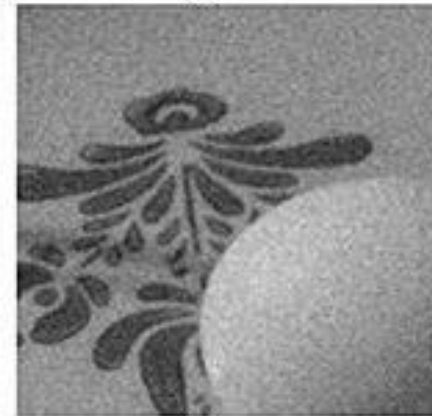
(a)



(b)



(c)



(d)

# Imágenes

- **Filtro paso bajo (suavizamiento):** utilizados para eliminar ruido o detalles pequeños de poco interés puesto que solo afecta a zonas con muchos cambios.
- **Filtro paso alto (atenuamiento):** intensifica los detalles, bordes y cambios de alta frecuencia, mientras que atenúa las zonas de tonalidad uniforme.
- **Realce de bordes mediante Laplace:** este tipo de filtros realza los bordes en todas direcciones
- **Resalte de bordes con gradiente direccional:** empleado para destacar y resaltar con mayor precisión los bordes que se localizan en una dirección determinada.
- **Detección de bordes y filtros de contorno (Prewitt y Sobel):** se centra en las diferencias de intensidad que se dan pixel a pixel. Son utilizados para obtener los contornos de objetos.

[https://claudiovz.github.io/scipy-lecture-notes-ES/advanced/image\\_processing/index.html](https://claudiovz.github.io/scipy-lecture-notes-ES/advanced/image_processing/index.html)

# Imágenes



Imagen original, y resultante tras filtros Laplaciano y Sobel.

# Convolución

35	40	41	45	50
40	40	42	46	52
42	46	50	55	55
48	52	56	58	60
56	60	65	70	75

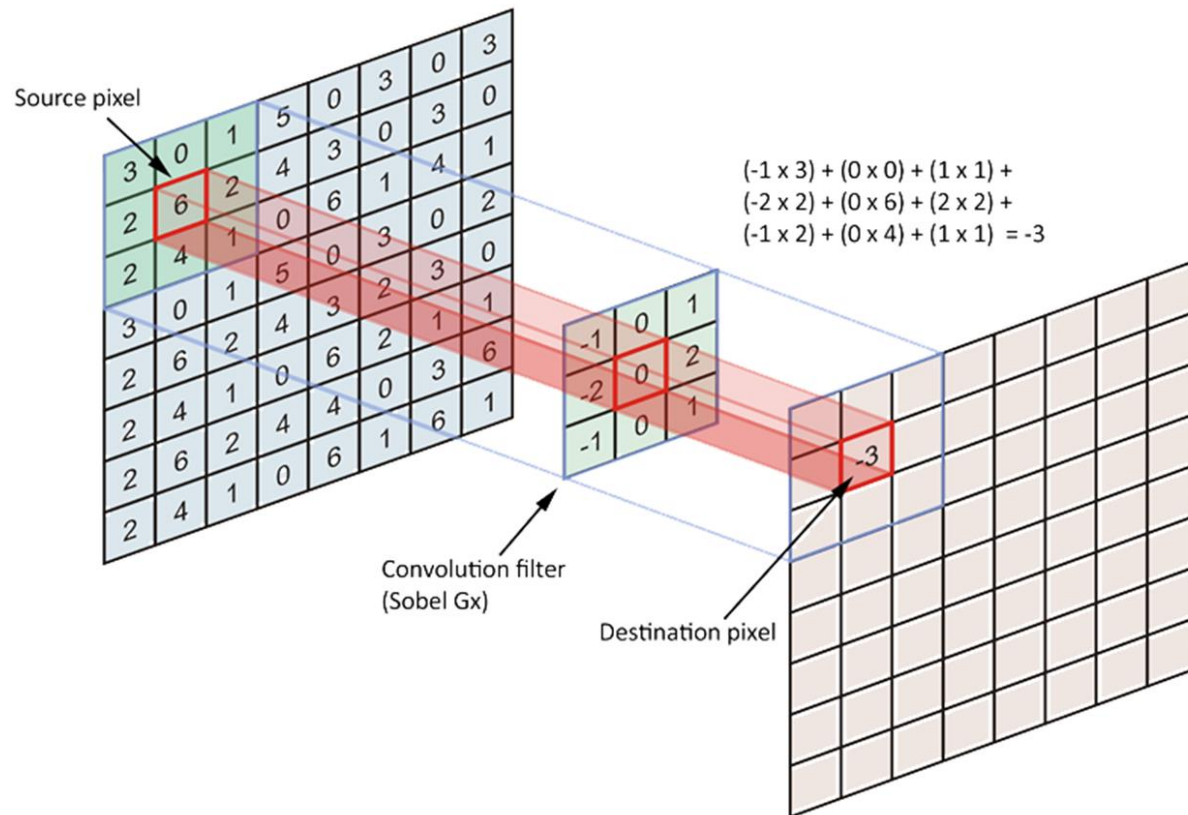
 $\times$ 

	0	1	0	
	0	0	0	
	0	0	0	

 $=$ 

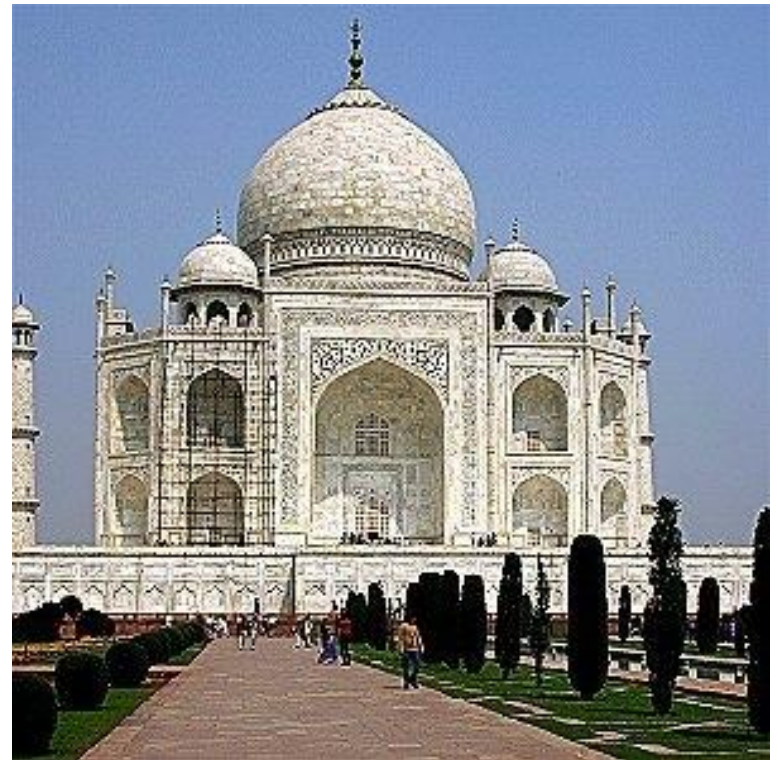
		42		

# Convolución



## Convolución: Enfocar

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





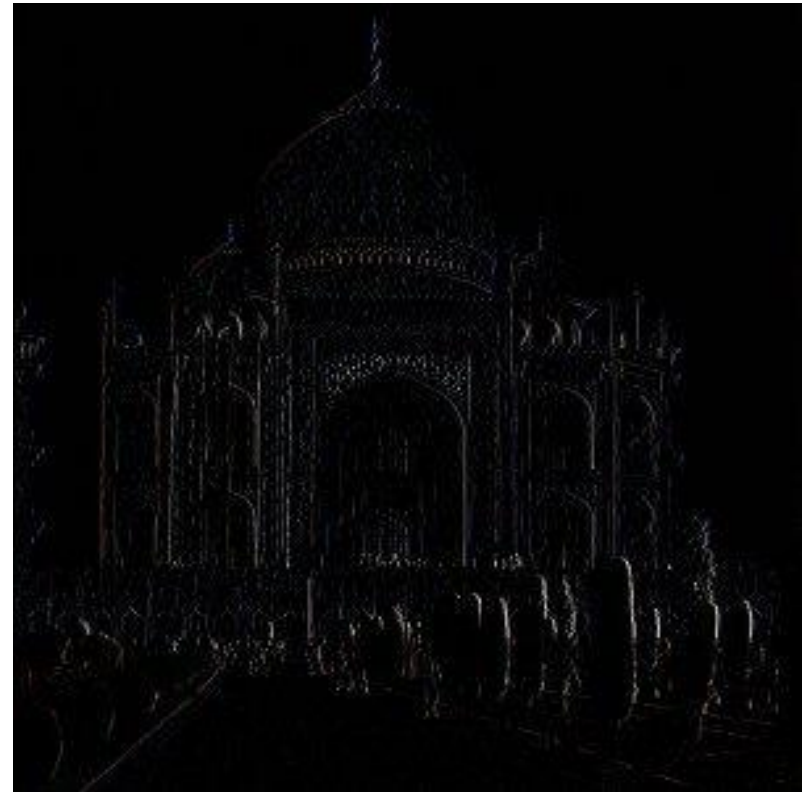
# Convolución: Desenfoque

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



## Convolución: Realzar bordes

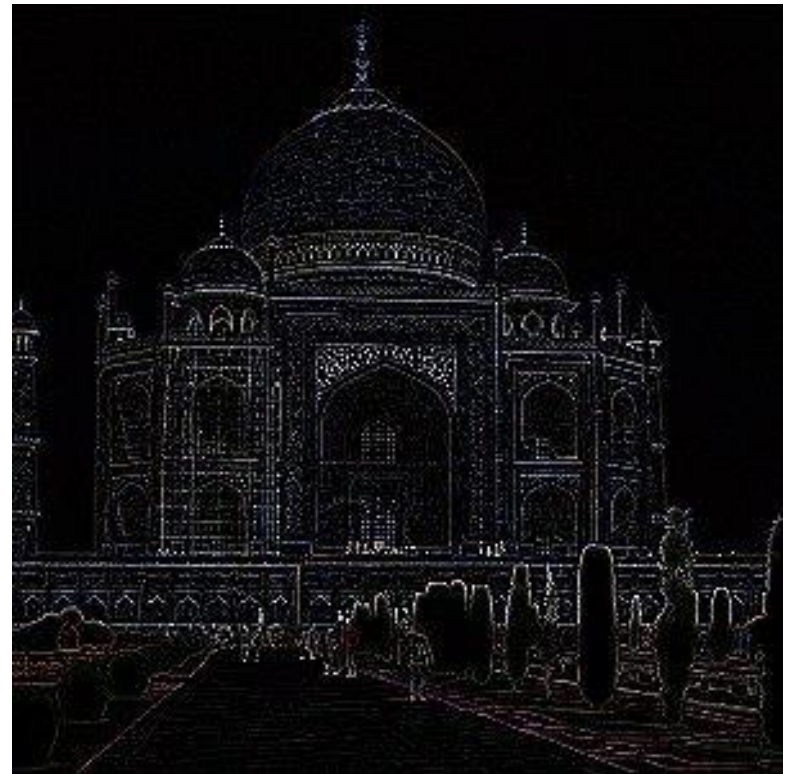
	0	0	0	
	-1	1	0	
	0	0	0	





## Convolución: Detectar bordes

	0	1	0	
	1	-4	1	
	0	1	0	



## Convolución: Repujado

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



## Filtrando imágenes



## Visión

La visión es –argumentable– uno de los sentidos más importantes para el humano. En donde confiamos muchas de las acciones que tomamos. Sin embargo, el reconocimiento de imágenes ha sido, y en algunos casos sigue siendo, uno de los problemas más difíciles en ciencias de la computación.

No es simple explicar a una máquina las características que observamos, sin embargo, podemos hacer que la máquina aprenda por si misma a reconocer esta información mediante DL.

## Justificación para CNN

Los pixels en una imagen están organizados en una grilla bidimensional, y si no fuera gris, tendríamos una tercera dimensión asociada al color.

- Existen imágenes en 3D (Resonancia magnética (MRI) usa un espacio tridimensional).
- Hasta ahora, hemos usado imágenes mediante la conversión de una malla 2D a una 1D (reconocer números, u otras imágenes).
- La idea fundamental de CNN es no necesitar la conversión a 1D de nuestra imagen original.
- En CNN las neuronas que se encuentran más cerca, poseen información más relevante para la neurona, que aquellas que se encuentran lejos. En términos de imágenes esto se traduce en que los pixeles cercanos a cierto pixel, son más relevantes en información que aquellos que se encuentran lejos.

## El problema CIFAR

El sistema CIFAR que veremos en un modelo secuencial presenta poca precisión utilizando capas fully-connected.

Una de las razones principales en estos casos es el overfit. La primera capa oculta, tiene 1024 neuronas. El input size fue de  $32 \times 32 \times 3 = 3072$ . Por lo tanto la primera capa tiene un total de  $3072 \cdot 1024 = 3145728$  pesos. Este no es un número pequeño, y no sólo es fácil que ocurra overfit en estos casos, sino que en términos de memoria es altamente ineficiente.

Adicionalmente cada input esta conectado a cada neurona en la capa oculta, lo que no toma ventaja de la proximidad espacial de los píxeles.



# CIFAR

**airplane**



**automobile**



**bird**



**cat**



**deer**



**dog**



**frog**



**horse**



**ship**

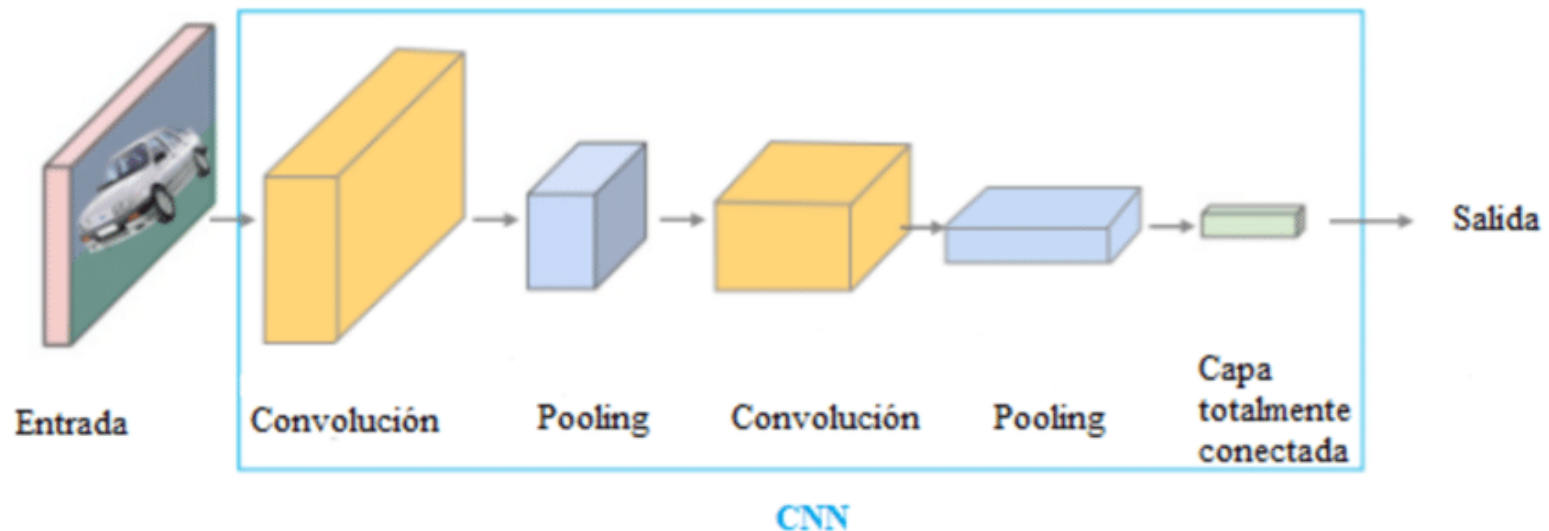


**truck**



## CIFAR y CNN

- Las neuronas se conectan sólo con un vecindario de píxeles de la imagen. Así las neuronas están obligadas a tomar inputs de neuronas que están cerca. Esto también reduce el número de pesos.
- Una CNN usa parámetros compartidos. En otras palabras, un numero limitados de pesos son compartidos entre todas las neuronas en una capa





# CIFAR y CNN

<sup>1</sup> 3	<sup>0</sup> 0	<sup>-1</sup> 1	2	7	4
<sup>1</sup> 1	<sup>0</sup> 5	<sup>-1</sup> 8	9	3	1
<sup>1</sup> 2	<sup>0</sup> 7	<sup>-1</sup> 2	5	1	3
0	1	3	1	7	8
4	2	1	6	2	8
2	4	5	2	3	9

Imagen: 6x6

\*

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

Filtro (kernel): 3x3

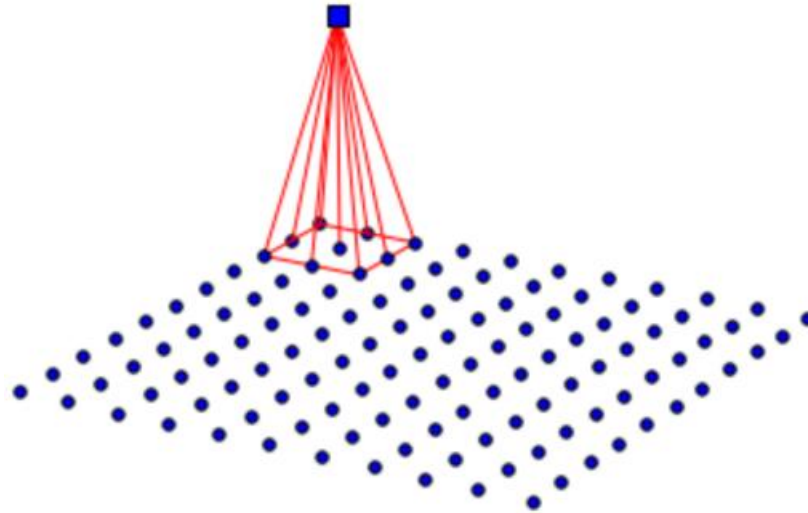
=

-5			

Resultado: 4x4

# Capas convolucionales

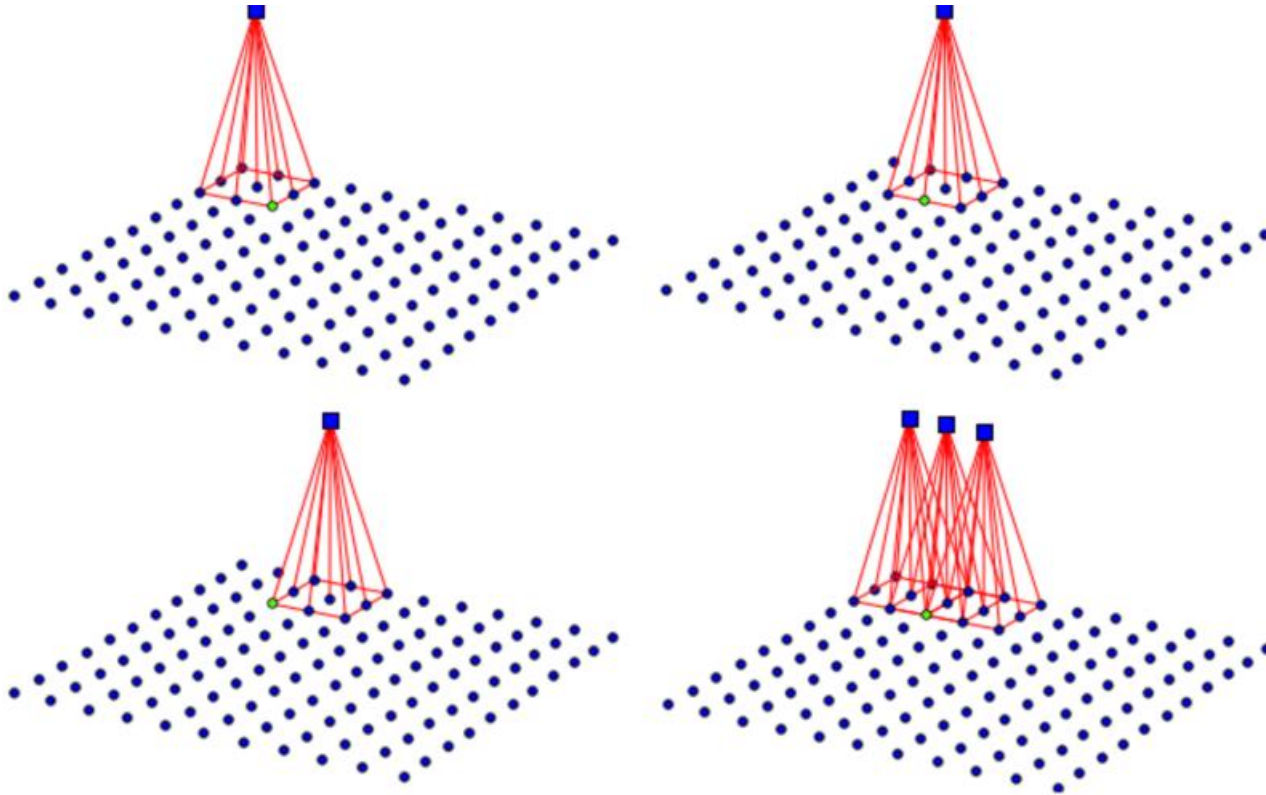
La capa convolucional es el elemento más importante de una CNN. Esta compuesta de un set de filtros (conocidos también como kernels o detectores de características), que es aplicado sobre todas las áreas del dato de entrada. Un filtro es definido como un set de pesos de aprendizaje.



## Capas convolucionales

- Aplicamos un filtro 3x3 en la imagen.
- Cada input es asociado con un peso de el filtro. En este caso hay nueve pesos, debido a las nueve neuronas de entrada. Pero en general el tamaño es arbitrario.
- La salida de el filtro es una suma pesada de estos inputs. Así el propósito es destacar una característica específica del input.
- El grupo de neuronas cercanas, que participan en el input, son llamadas el campo receptivo.
- En el contexto de la red, la salida del filtro representa el calor de activación de una neurona en la siguiente capa.

# Capas convolucionales

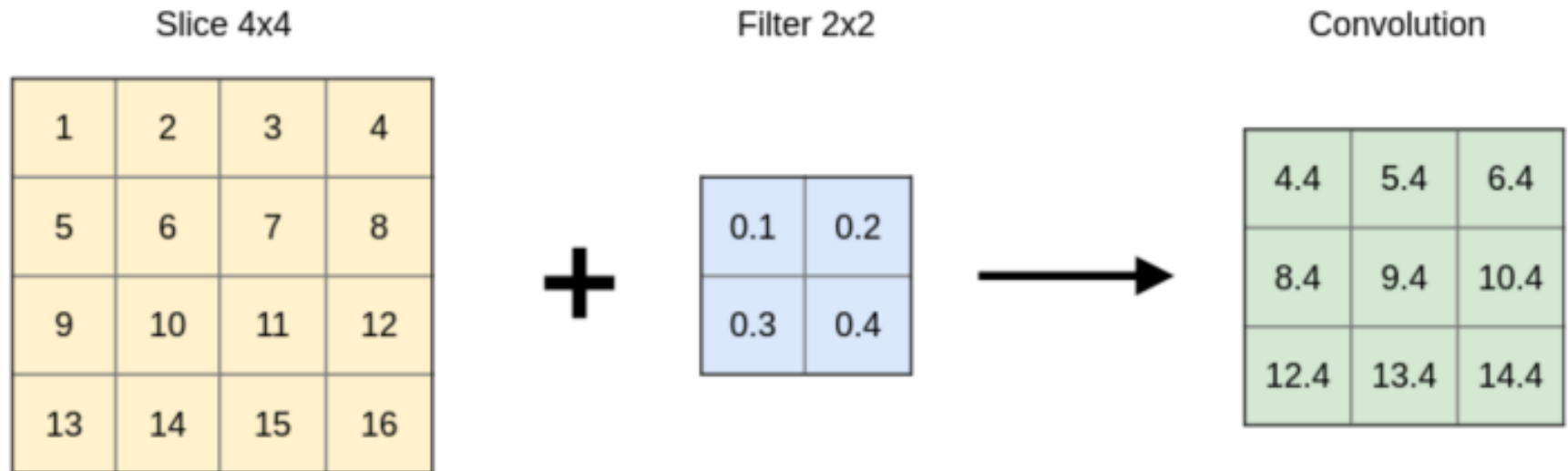


# Capas convolucionales

- Desplazamos el filtro por toda la imagen.
- Los pesos del filtro no cambian
  - Reduce el overfitting
  - El filtro destaca características especiales. Podemos asumir que la característica es útil a pesar de la posición en la imagen.
- Para calcular todas las salidas, repetimos el proceso hasta que nos hemos desplazado por todo el input.
- Las neuronas ordenadas espacialmente son llamadas depth slices (o feature map).
- El slice puede servir como input a otras capas en la red.

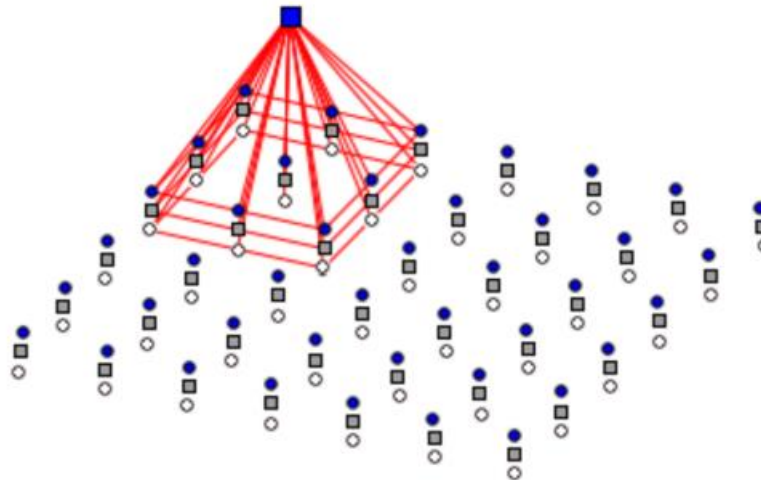
## Activación

Al igual que en capas regulares, podemos usar funciones de activación después de cada neurona. ReLU es una de las más comunes.



## ¿Y los colores?

Uno podría separar ahora una imagen en colores, por ejemplo, en un sistema RGB añadiendo 3 nuevas capas. Esta combinación de slices es llamada volumen de entrada con una profundidad de 3. Aplicando un filtro 3x3, la activación será la suma de los pesos de los  $3 \times 3 \times 3 + 1 = 28$  pesos (se añadió un bias).



# Qué dimensiones tiene que tener el Kernel de una red neuronal convolucional

Pues bien, para la primera pregunta no hay una respuesta exacta. Sí que parece que una red con muchas capas y kernels más pequeños es más eficiente que una red neuronal convolucional que tenga menos capas y Kernels más grandes. De hecho, lo más normal es que el Kernel tenga una dimensión de  $3 \times 3$ .

**$3 \times 3$**

0.91	0.32	0.07
0.73	0.26	0.81
0.53	0.68	0.14

**$5 \times 5$**

0.27	0.64	0.44	0.84	0.29
0.28	0.06	0.89	0.99	0.33
0.64	0.67	0.08	0.38	0.03
0.04	0.31	0.16	0.57	0.08
0.87	0.85	0.97	0.71	0.96



# ¿Hay un límite de capas convolucionales que puedes meter?

Con cada convolución, se reduce el tamaño del resultado, por lo que, si no se hace nada, sí que habría un número máximo de capas convolucionales que se pueden aplicar. Pero, esto mismo lo podemos evitar incluyendo lo que se conoce como padding o relleno a la imagen.

