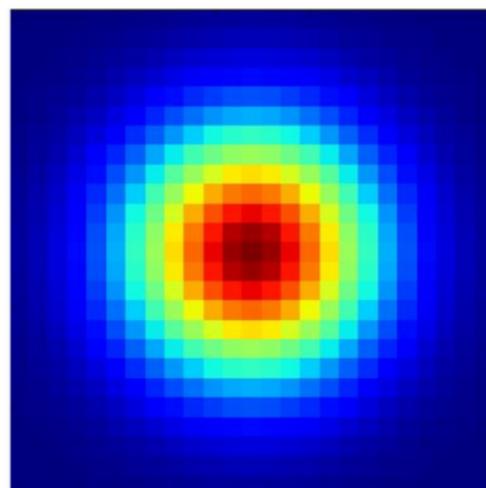


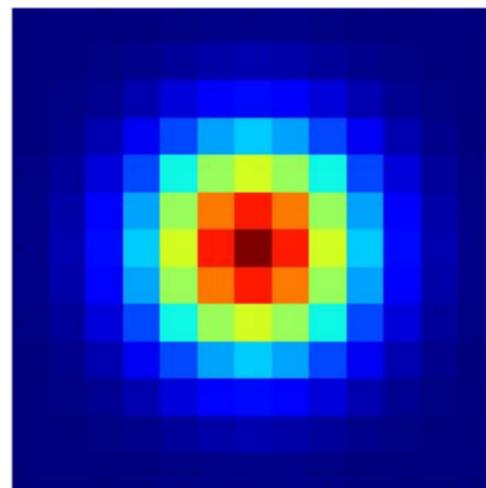


Procesamiento de Imágenes

Convolución



$r=12, \sigma=4.5$



$r=6, \sigma=2.0$

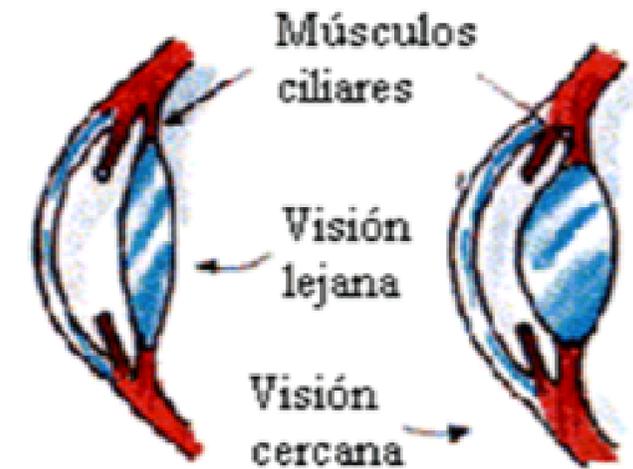
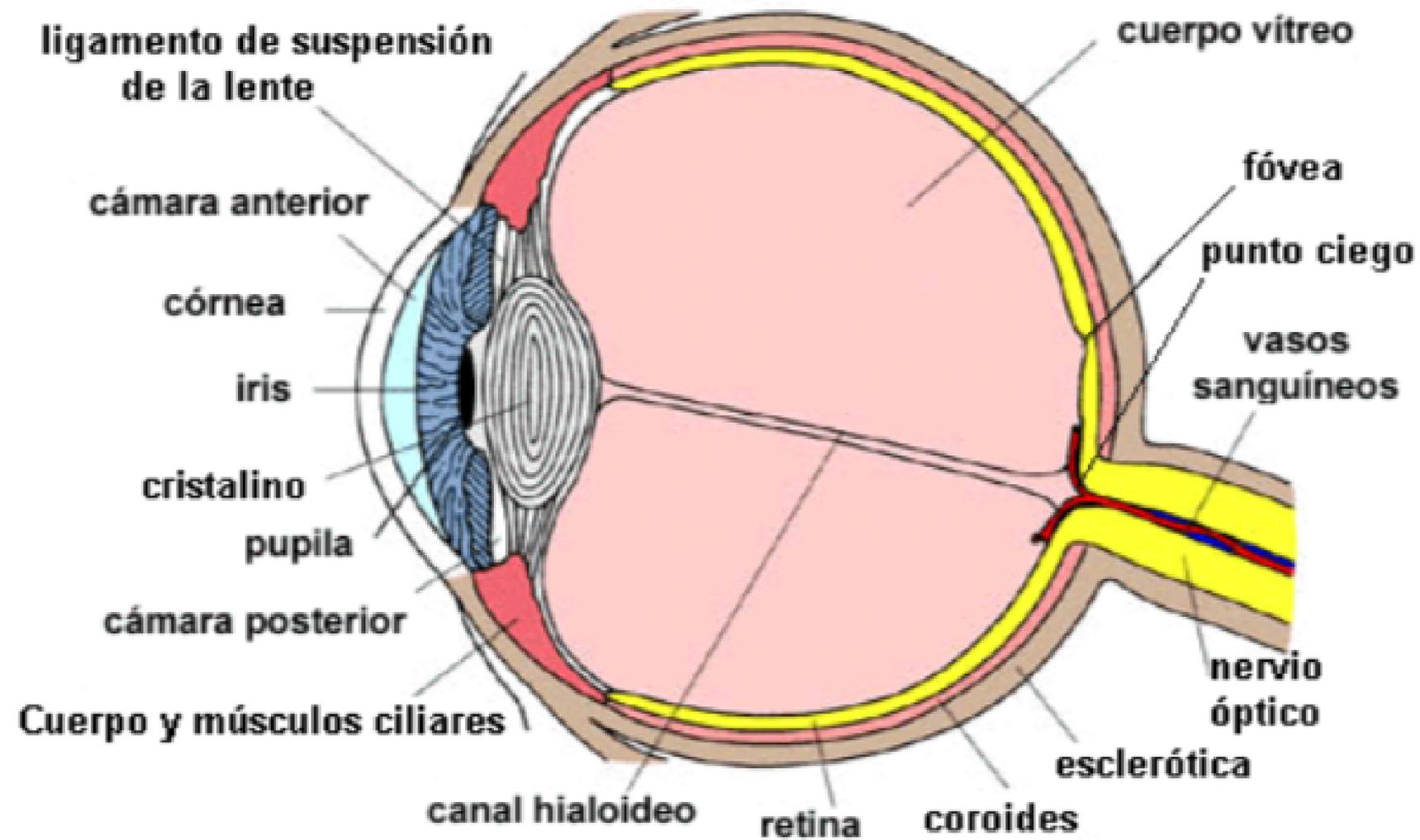
CLASE 2

José M. Saavedra R.
Profesor Asistente

jmsaavedrar@miuandes.cl

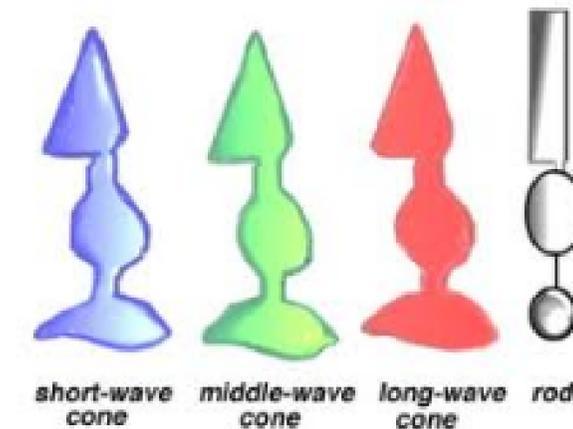
Ed. Ingeniería - Oficina 315

La Imagen y sus Propiedades



Modelo Biológico

La Imagen y sus Propiedades

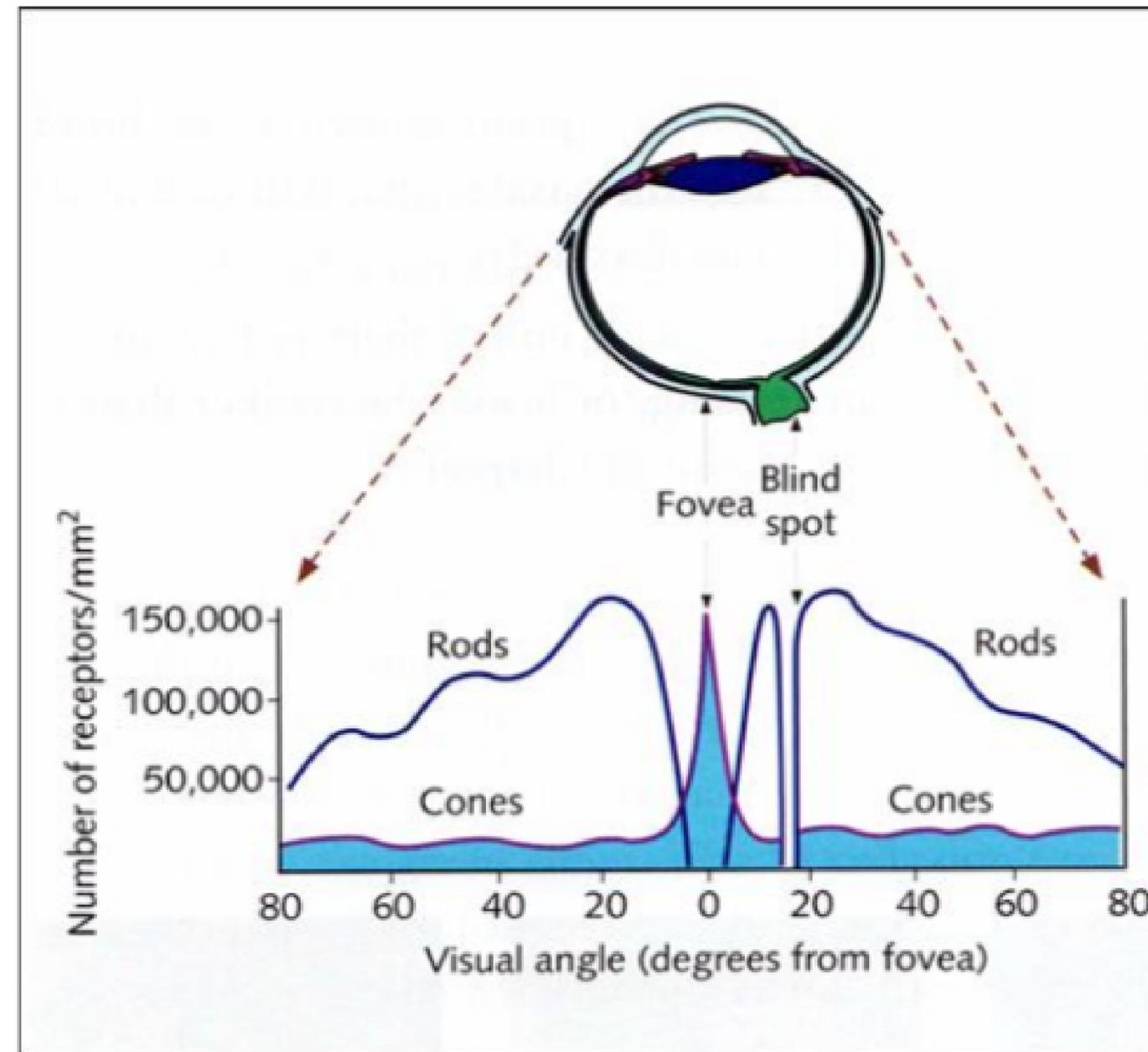


Short ~450nm
Medium ~560nm
Long ~700nm

En la retina encontramos dos tipos de células fotosensibles:

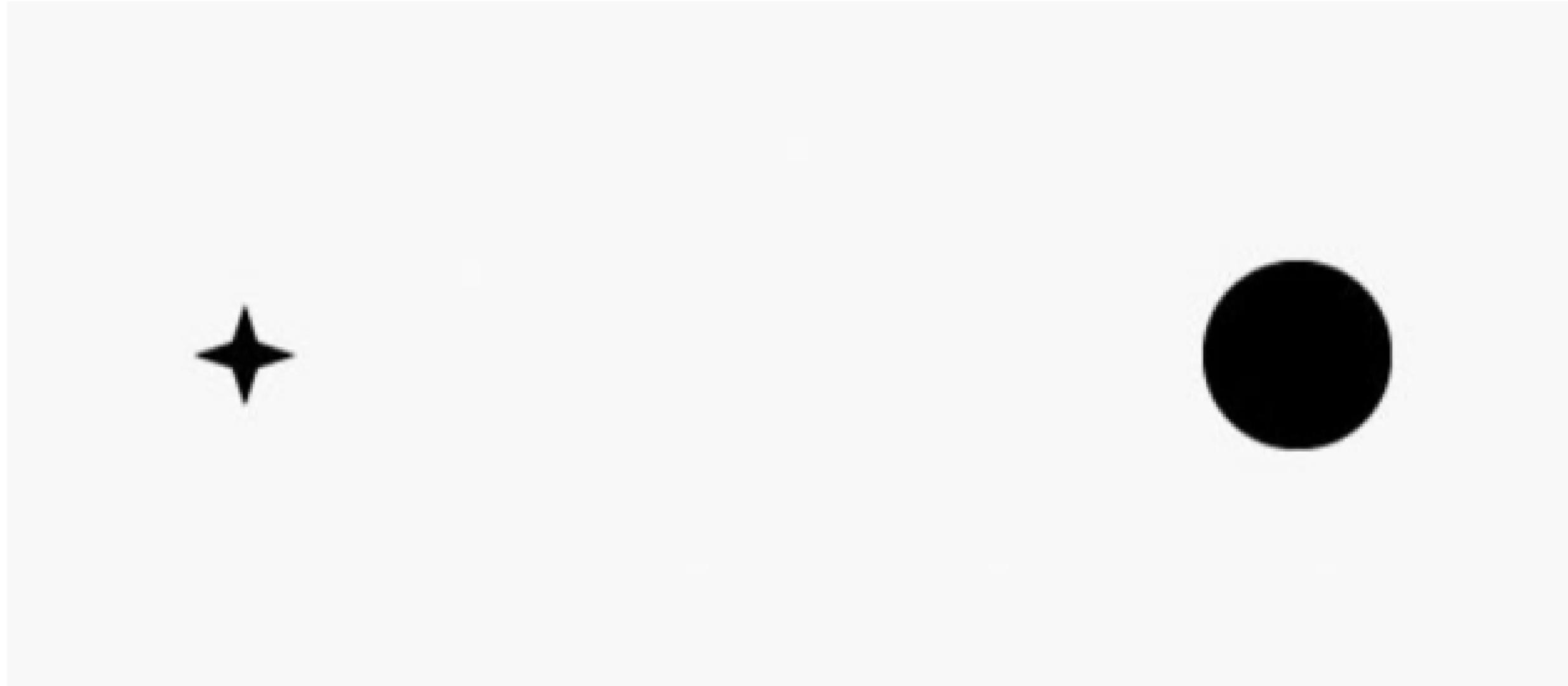
- **Conos:** Visión a color (6 ~ 7 millones).
- **Bastones:** Altamente sensibles a ambientes oscuros (120 millones).

La Imagen y sus Propiedades



Modelo Biológico

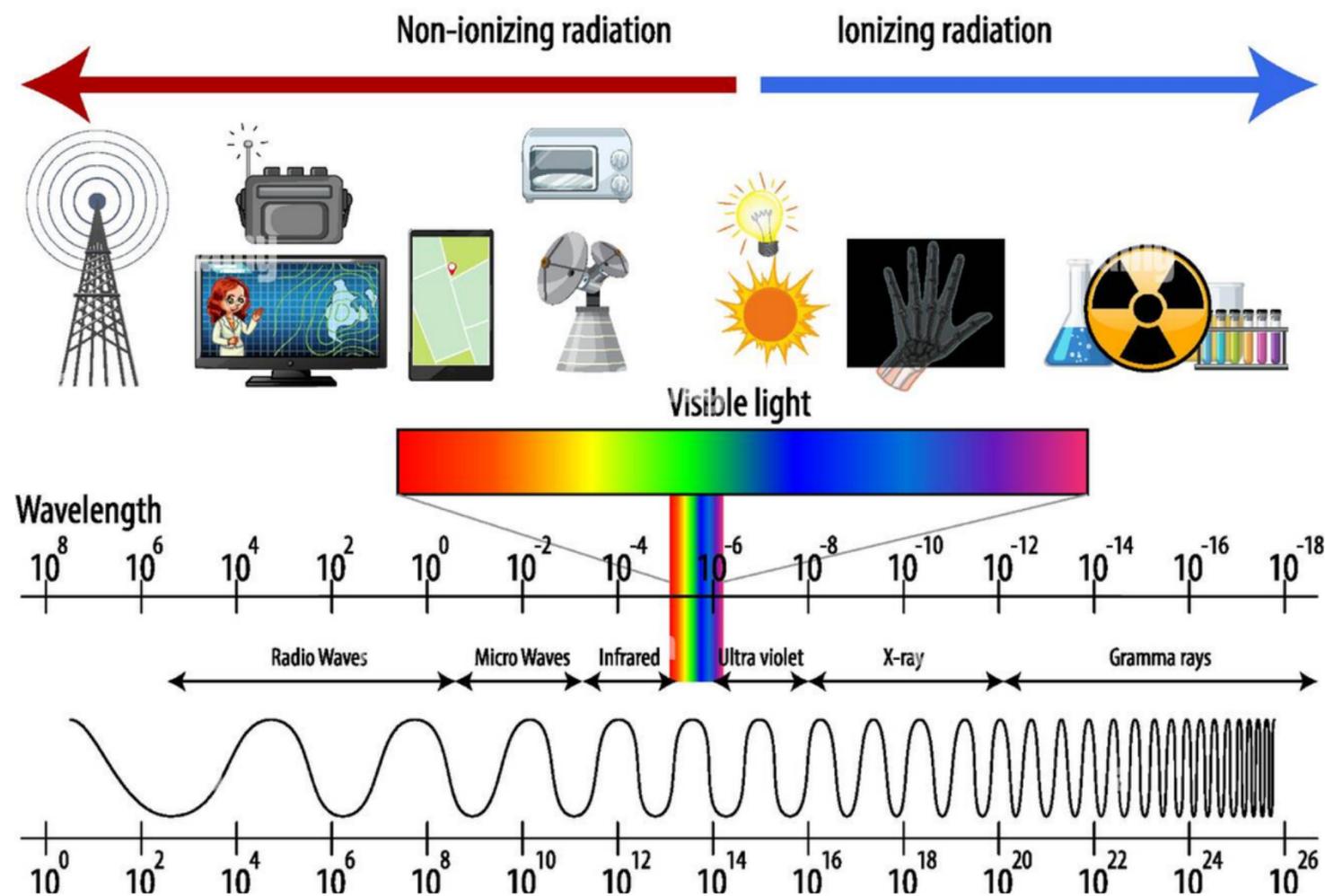
La Imagen y sus Propiedades



Modelo Biológico (punto cieto)

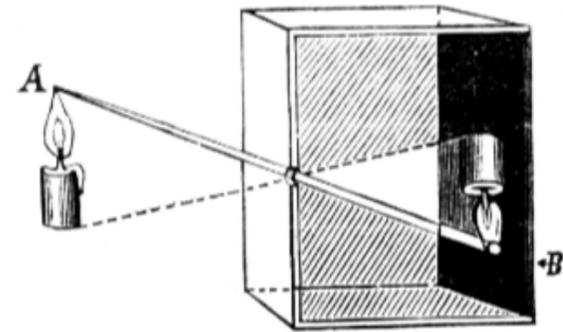
La Imagen y sus Propiedades

Espectro Electromagnético: Es el conjunto de longitudes de onda de todas las radiaciones electromagnéticas. Un subconjunto es el espectro visible (luz).



La Imagen y sus Propiedades

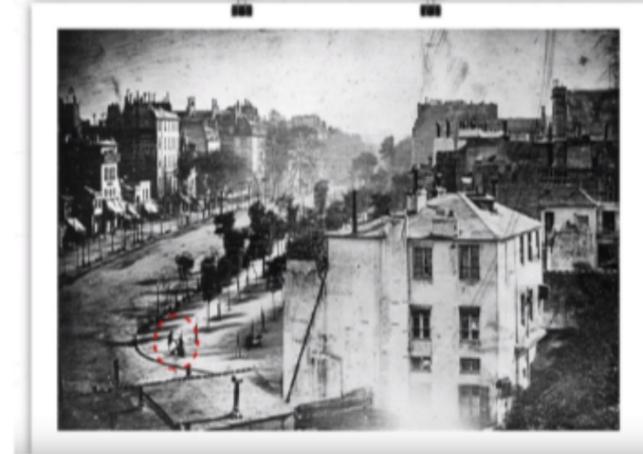
La Fotografía



Modelo "Pinhole"
(cámara oscura)



DAGUERROTIPO
Louis Daguer
Joseph Niépce ~1830

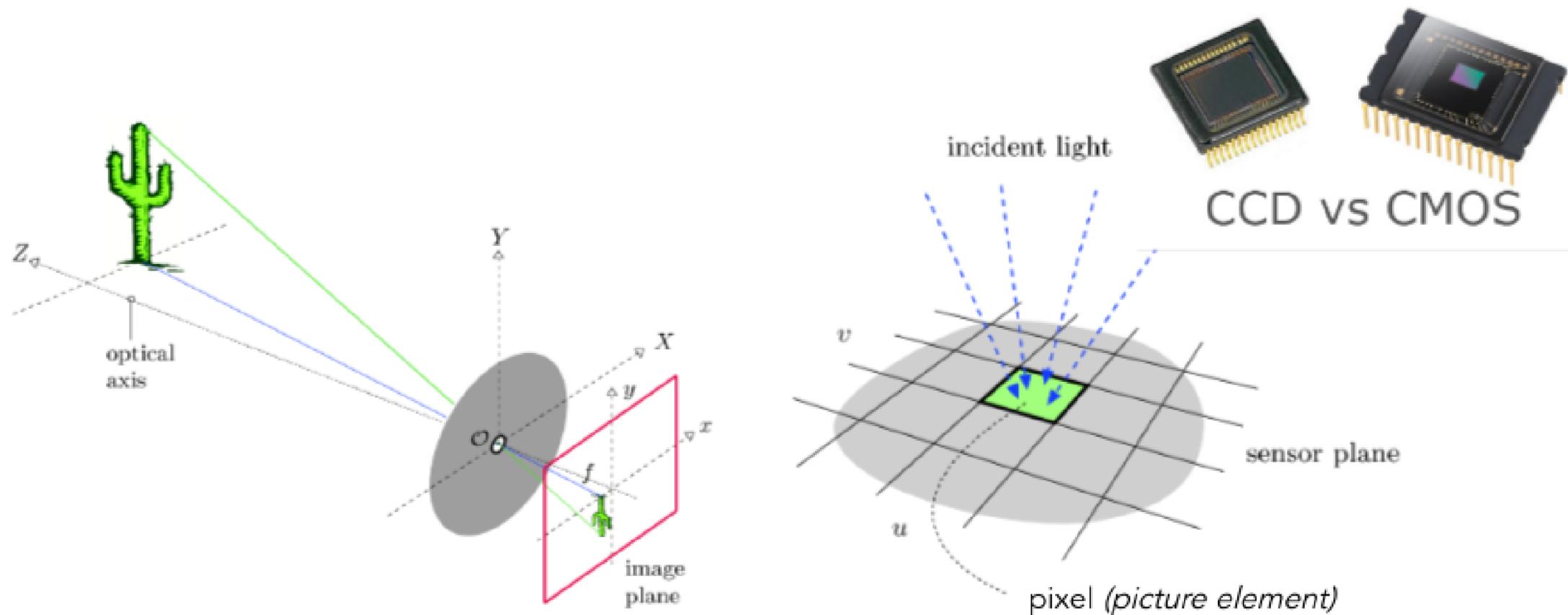


Cámara Oscura (Modelo a gran escala - Museo de Ciencias Granada)

Cámara Oscura

La Imagen y sus Propiedades

La Imagen Digital



CCD: Charge-Coupled Device

CMOS: Complementary Metal Oxide Semiconductor

La Imagen y sus Propiedades

Muestreo y Cuantización

Muestreo: Discretiza el dominio de la función imagen.

Cuantización: Discretiza el rango de la función.

MUESTREO + CUANTIZACIÓN = IMAGEN DIGITAL

La Imagen y sus Propiedades

Muestreo



Tamaño de una image es diferente a su resolución

La Imagen y sus Propiedades

Muestreo



RESOLUCIÓN

dpi : dots per inch

- Tamaño de una image: pixeles
- Resolución: dots per inch (dpi)



1250 dpi, 300 dpi

150 dpi, 72 dpi

La Imagen y sus Propiedades

Cuantización

Determinar el número de niveles de intensidad de luz (niveles de voltaje), L .

Si usamos b bits para representar cada nivel, entonces

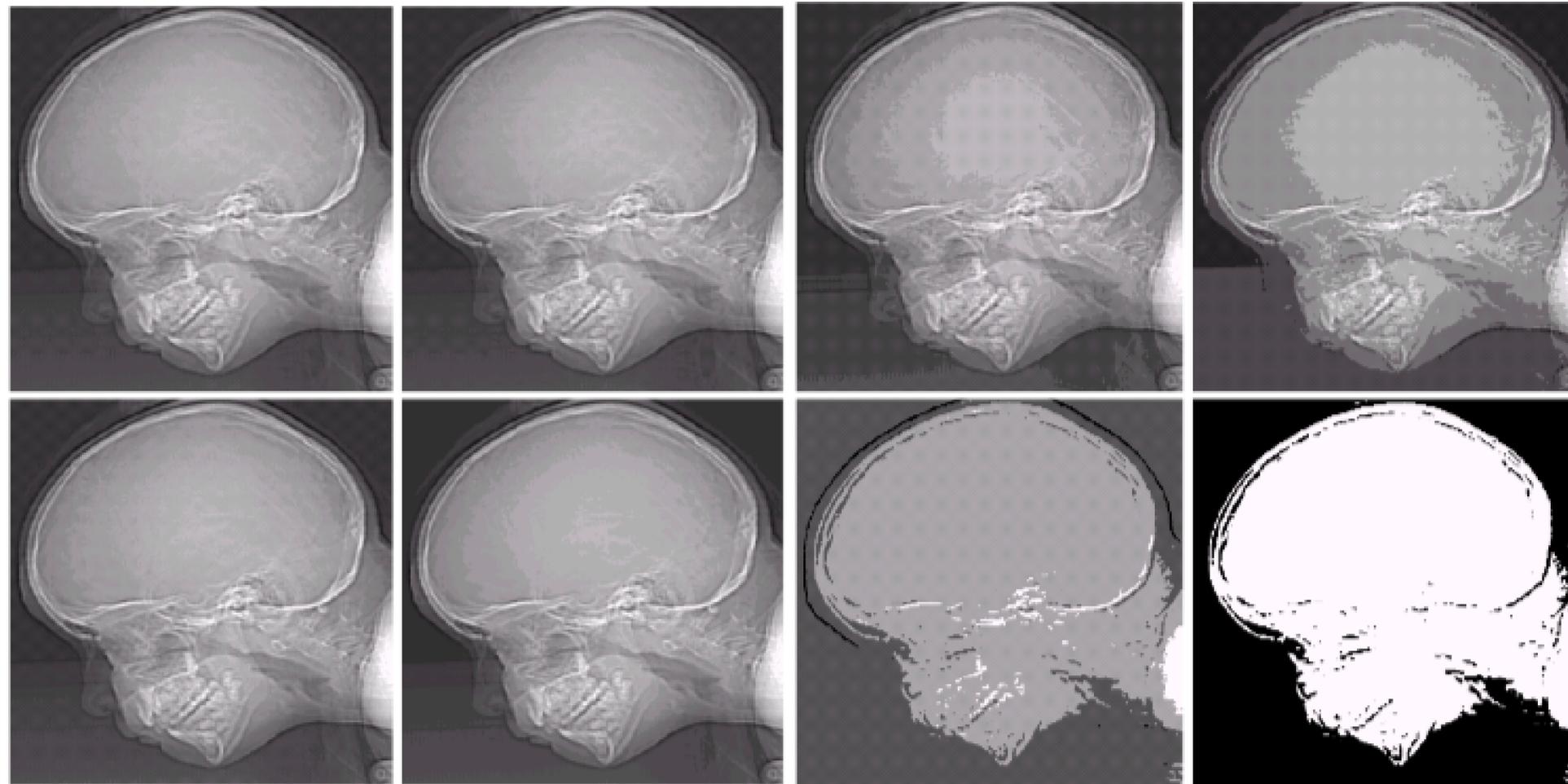
$$L = 2^b \quad b = 8 \text{ (byte), } L = 256$$



En el campo médico se usa imágenes de 16 bits (xR, CTs, MRI)

La Imagen y sus Propiedades

Cuantización



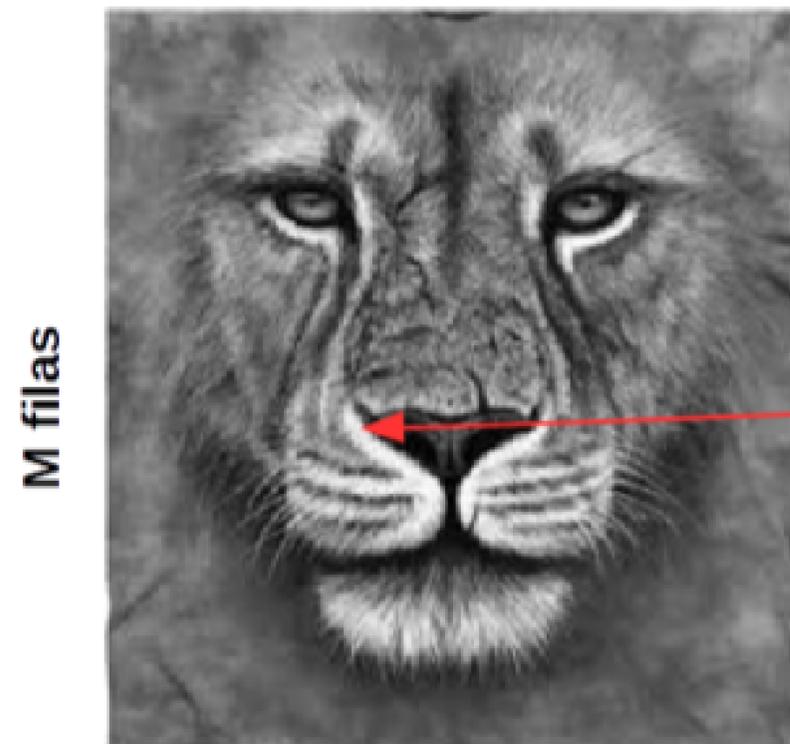
Reduciendo la cantidad de bits por pixel

La Imagen y sus Propiedades

La Imagen

$$I : \{0, \dots, M - 1\} \times \{0, \dots, N - 1\} \rightarrow \{0, \dots, L - 1\}$$

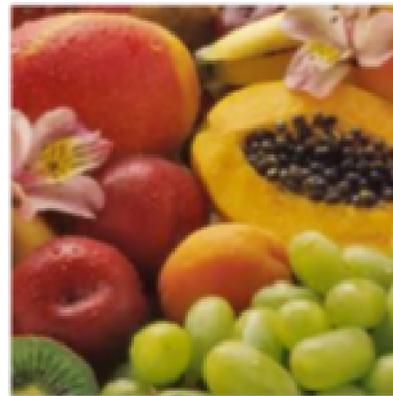
$$I : \{0, \dots, M - 1\} \times \{0, \dots, N - 1\} \rightarrow \{0, \dots, 255\} \quad 0: \text{ negro, } 255: \text{ blanco}$$



114	127	129	124	121	120	114	122	180	207
142	164	153	141	143	142	157	171	202	187
157	169	160	155	146	119	119	158	193	169
148	165	158	138	123	124	126	171	204	185
171	176	164	139	126	138	122	161	202	197
164	155	147	139	144	162	135	165	200	197
182	169	150	132	131	152	153	185	189	167
178	169	152	130	122	144	168	213	201	168
172	159	151	144	139	143	164	215	211	184
189	156	142	144	147	141	158	204	201	177

La Imagen y sus Propiedades

La Imagen



$$I : \{0, \dots, M - 1\} \times \{0, \dots, N - 1\} \rightarrow \{0, \dots, L - 1\}^3$$

Si $L=256$, el número de posibles colores es $256^3 \sim 16.7$ millones de colores.



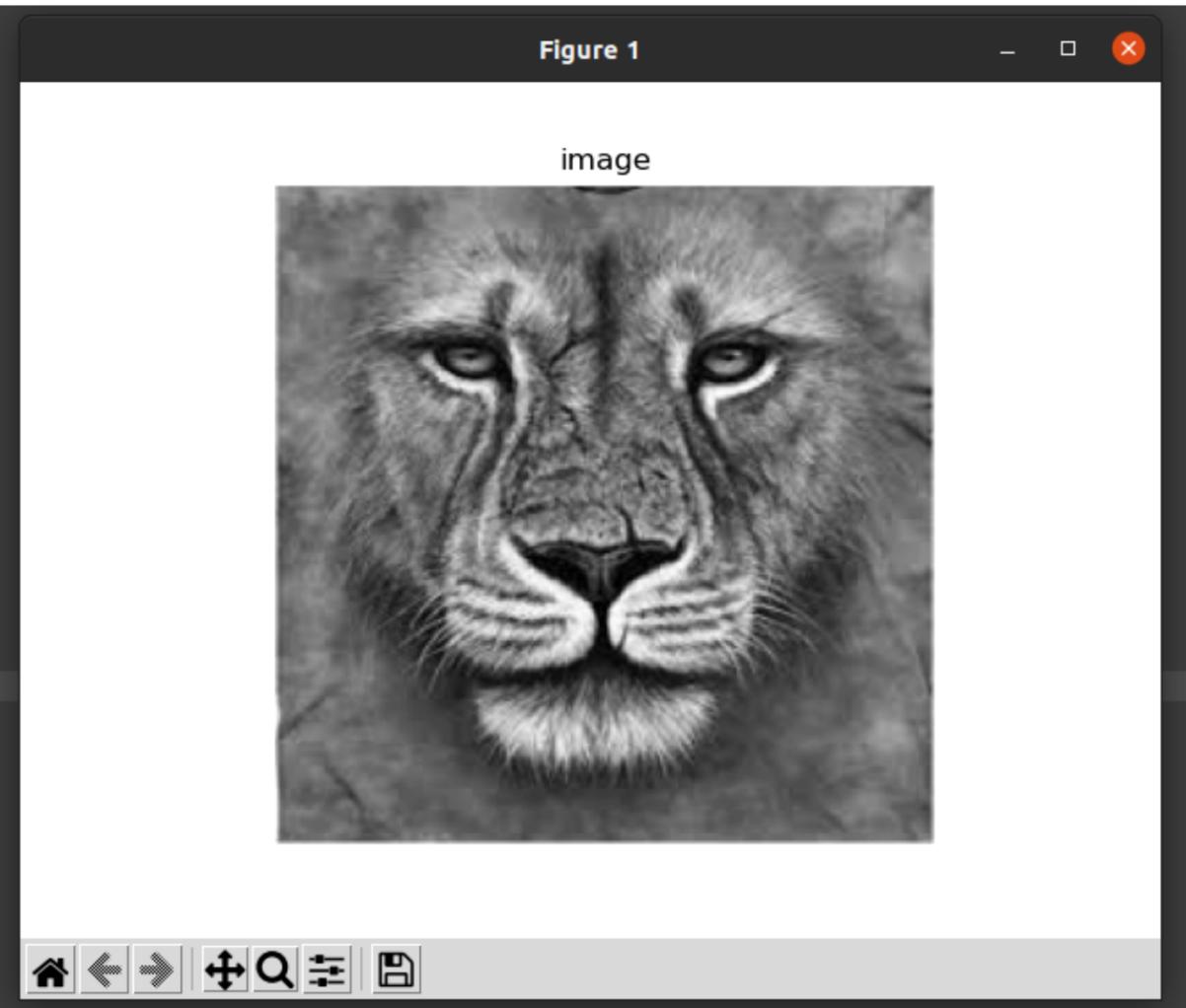
La Imagen y sus Propiedades

La Imagen

```
author: jsaavedr

A simple example for reading images with skimage
scikit-image
pip install scikit-image
'''
import skimage.io as io
import matplotlib.pyplot as plt

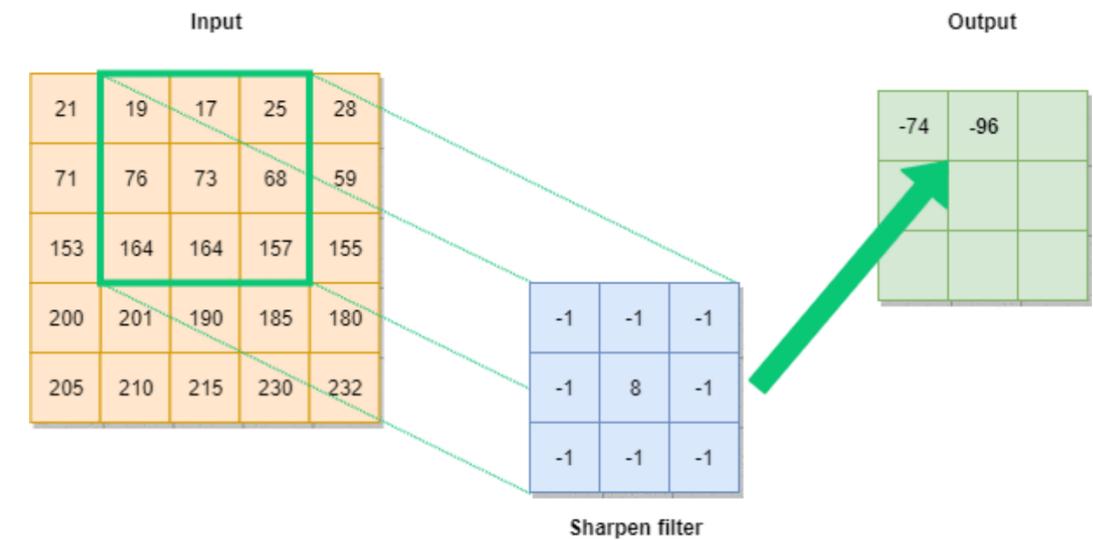
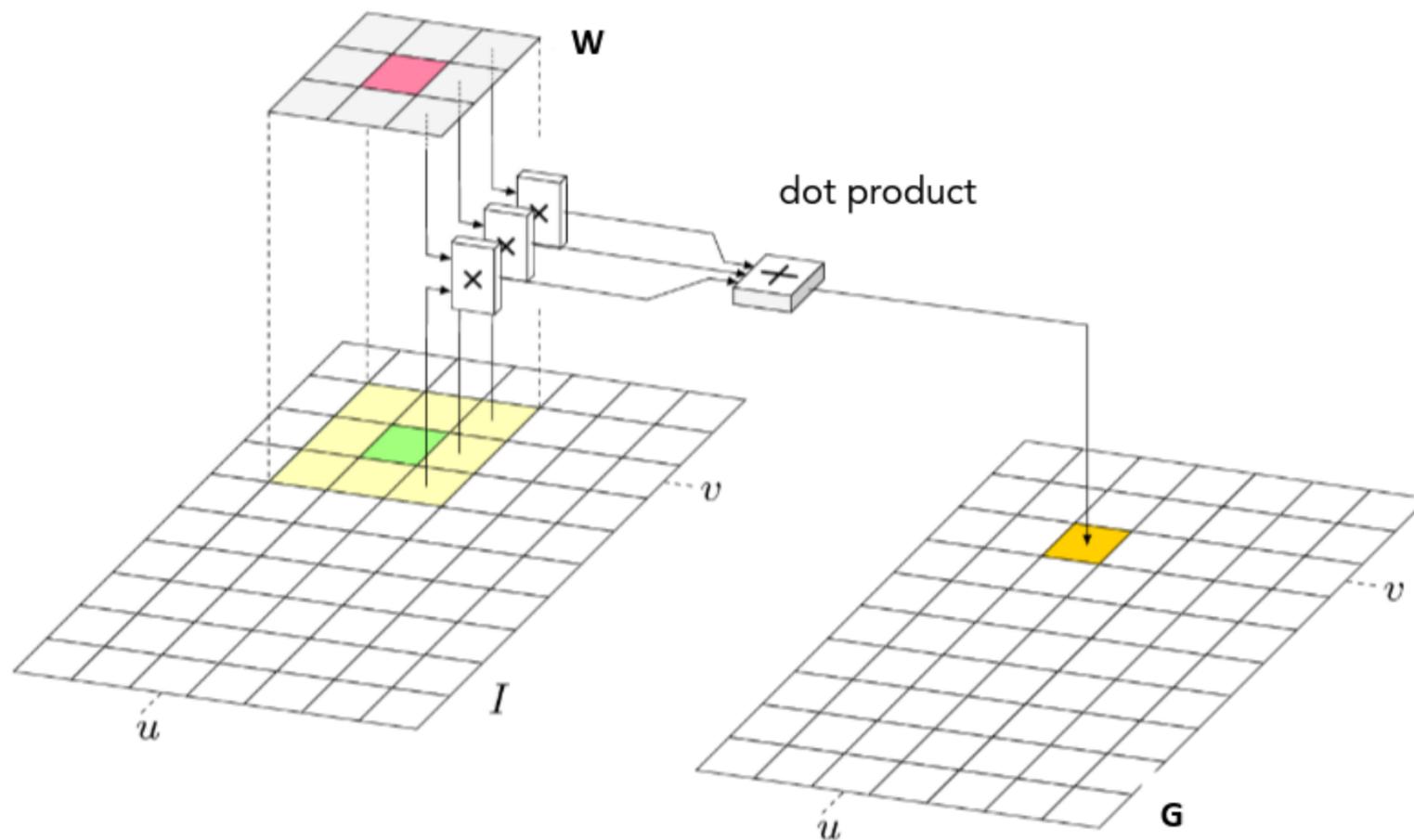
if __name__ == '__main__':
    filename = '../images/gray/lion_gray.jpg'
    #filename = '../images/color/fichas.jpg'
    image = io.imread(filename, as_gray = 'True')
    print(image[10:15,10:15])
    #0..249 x 0..249
    print('shape: {} dtype: {}'.format(image.shape, image.dtype))
    ##showing image
    plt.imshow(image, cmap = 'gray')
    plt.title('image')
    plt.axis('off')
    plt.show()
```



<https://github.com/jmsaavedrar/cc5508>

La Imagen y sus Propiedades

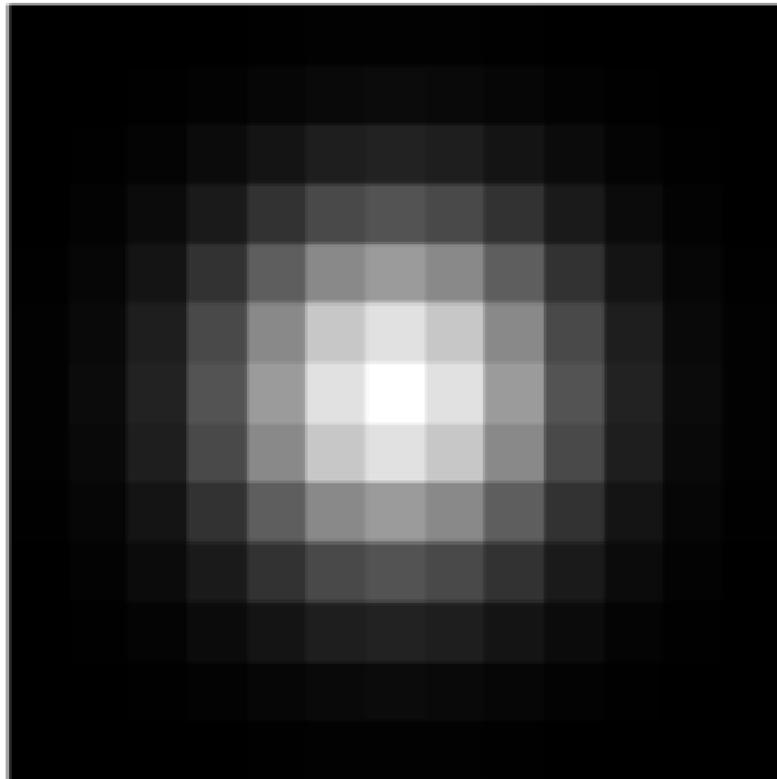
Filtrado Lineal (convolución)



Ejemplo de filtrado
[Notar la aplicación de dot-product en cada paso]

Filtros de Suavizamiento [reducción de ruido]

kernel



Imagen



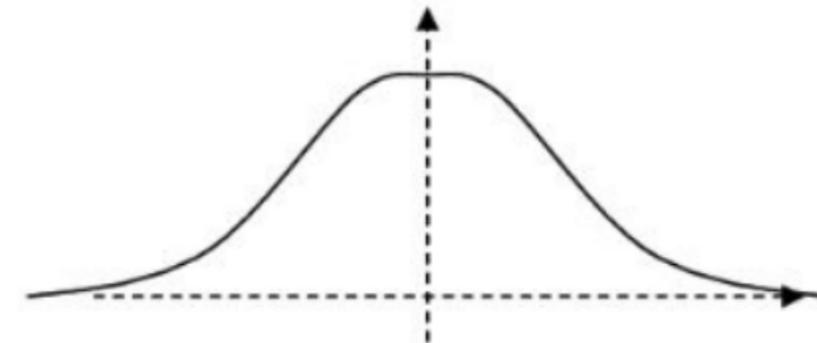
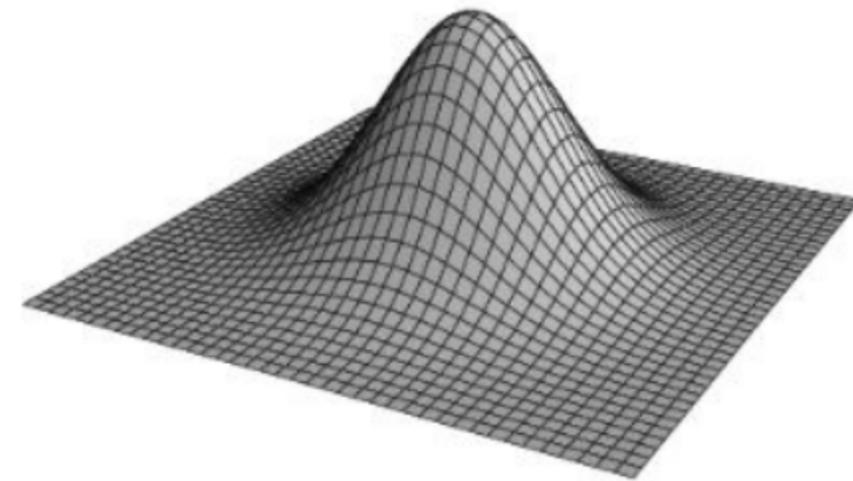
Imagen Filtrada



Filtros de Suavizamiento [reducción de ruido]

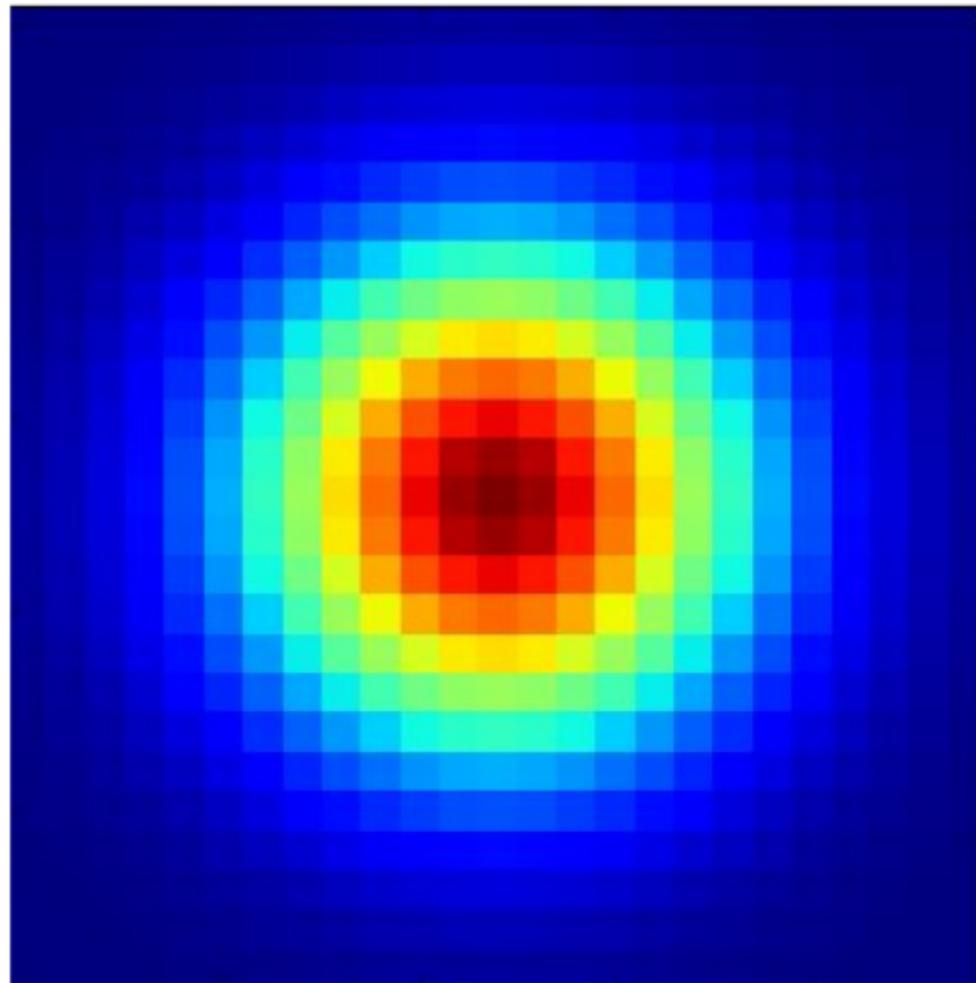
Filtro Gaussiano

$$W(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\left(\frac{x^2+y^2}{2\sigma^2}\right)}$$

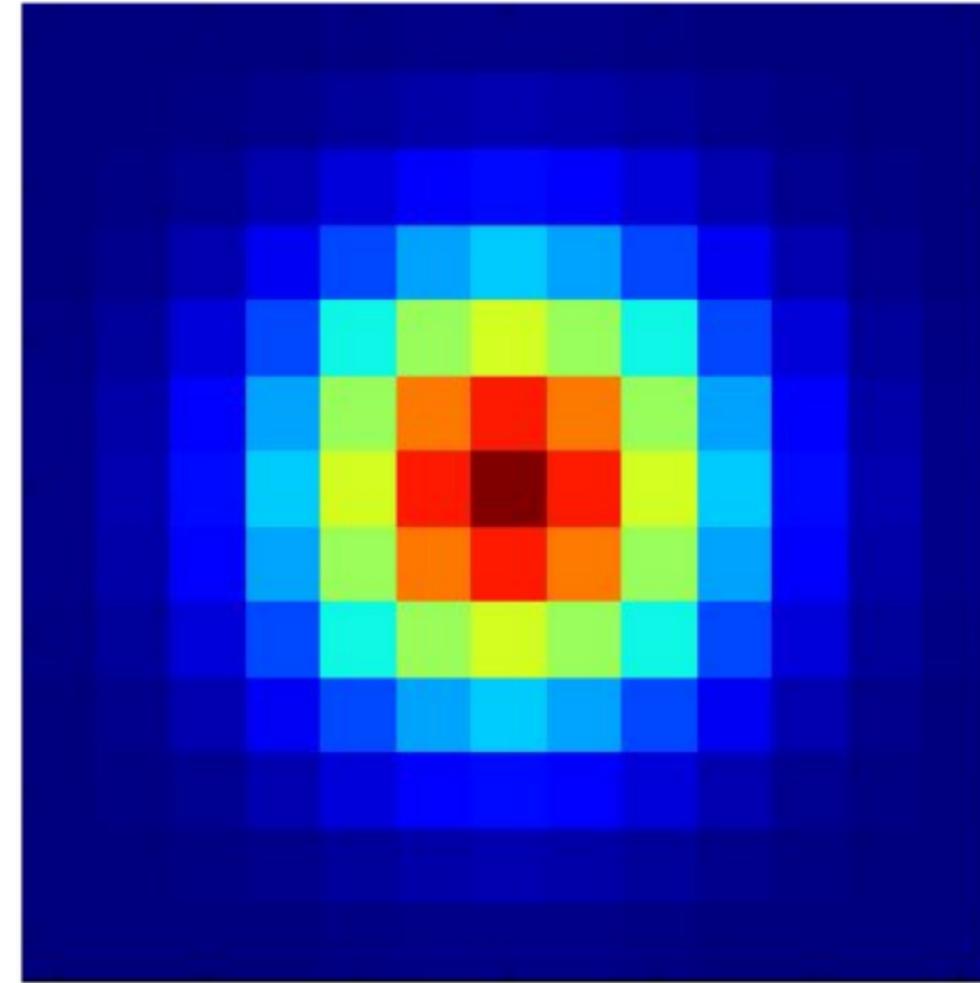


Filtros de Suavizado [reducción de ruido]

Filtro Gaussiano



$r=12$, $\sigma=4.5$



$r=6$, $\sigma=2.0$

Filtros de Suavizamiento [reducción de ruido]

Filtro Gaussiano



```
14 if __name__ == '__main__':
15
16     #filename = '../images/gray/rice.jpg'
17     #filename = '../images/gray/ruido.tif'
18     filename = '../images/gray/lenna_gray_noisy.png'
19     image = pai_io.imread(filename, as_gray = True)
20     g_kernel = utils.get_gaussian2d(sigma = 2, radius = 6)
21     image_g = nd_filters.convolve(image, g_kernel, mode='constant', cval=0)
22     print((g_kernel*100).astype(np.int32) / 100.0)
23     fig, xs = plt.subplots(1,3)
24     for i in range(3):
25         xs[i].set_axis_off()
26     xs[0].imshow(g_kernel, cmap = 'gray')
27     xs[0].set_title('kernel')
28     xs[1].imshow(image, cmap = 'gray', vmin = 0, vmax = 255)
29     xs[1].set_title('Imagen')
30     xs[2].imshow(image_g, cmap = 'gray', vmin = 0, vmax = 255)
31     xs[2].set_title('Imagen Filtrada')
32     plt.show()
```

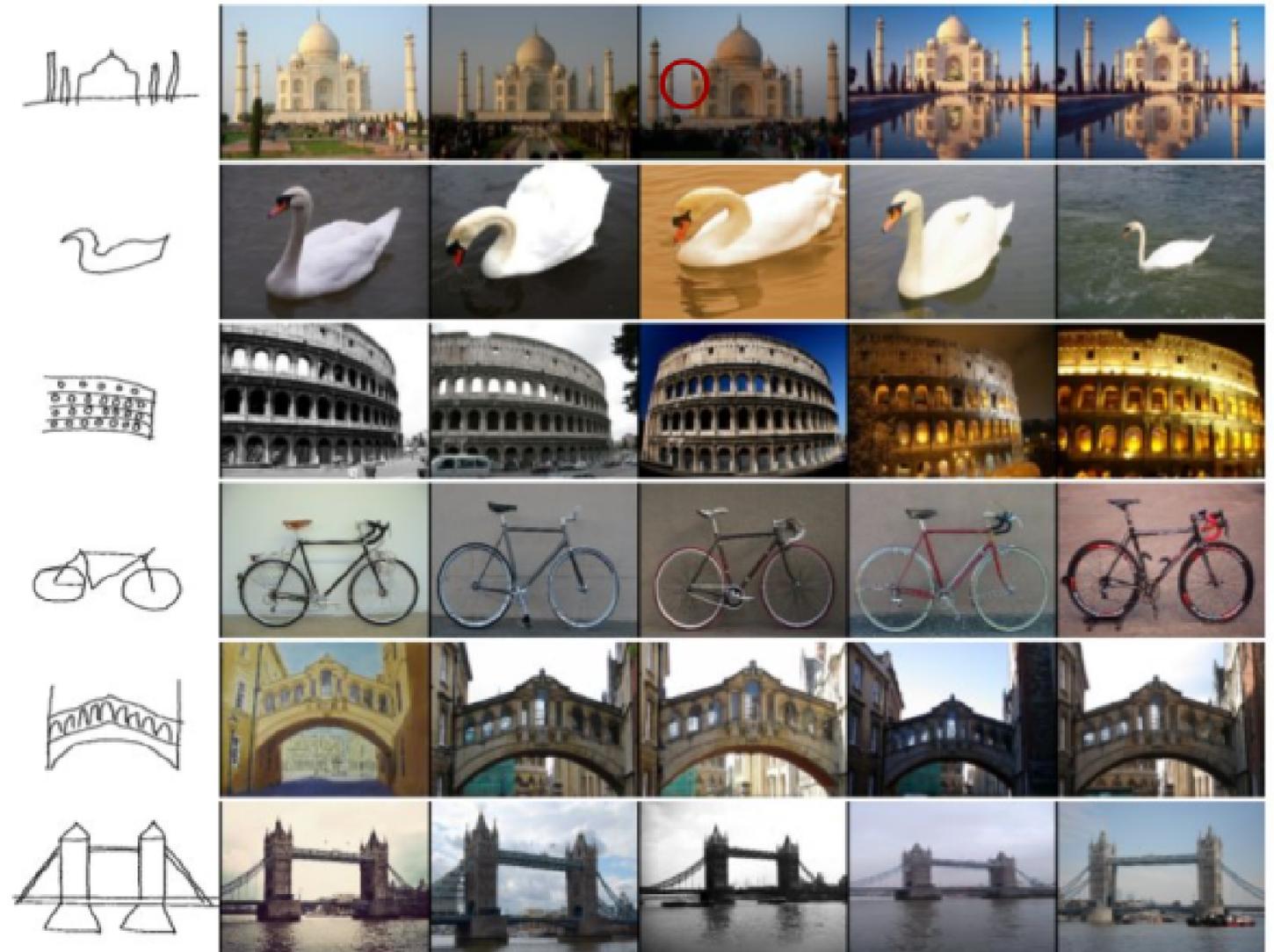
<https://github.com/jmsaavedrar/CC5508>

[pai_basic/example_14_gaussian.py]

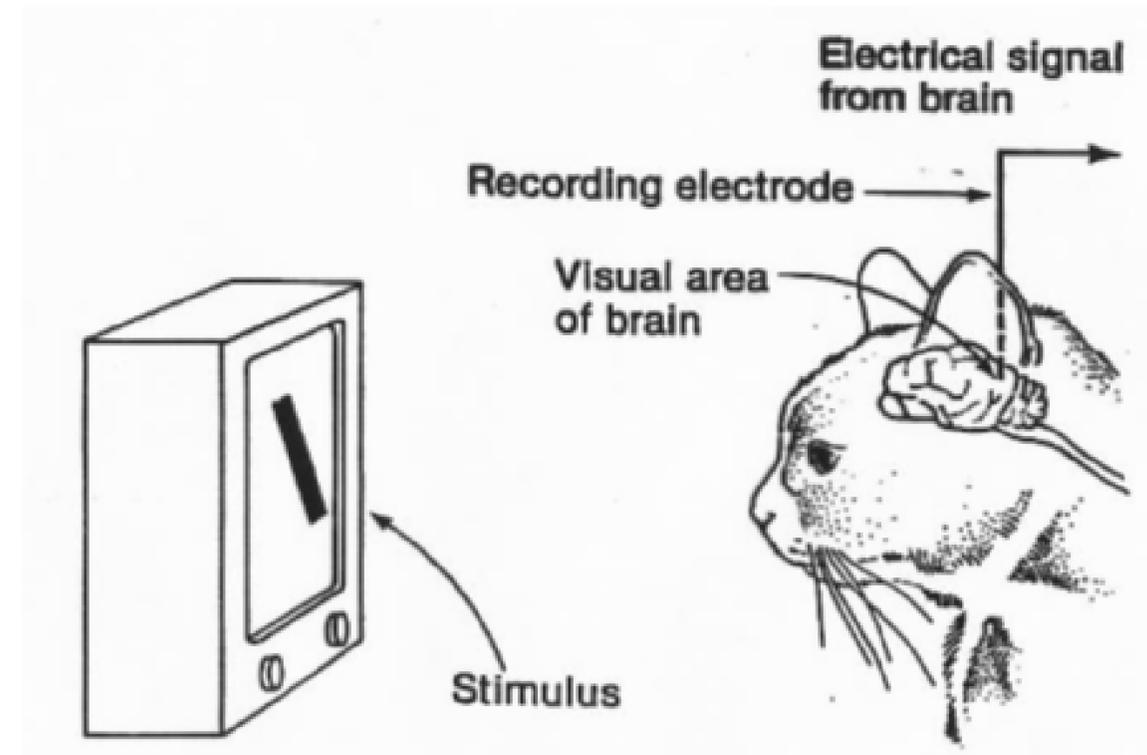
Filtros de Detección de Borde

Identificar cambios repentinos o discontinuidades en una imagen.

- Los bordes llevan alto valor semántico
- Permiten representar la forma de los objetos de una imagen
- El cerebro responde con mayor fuerza a los bordes.



Filtros de Detección de Borde



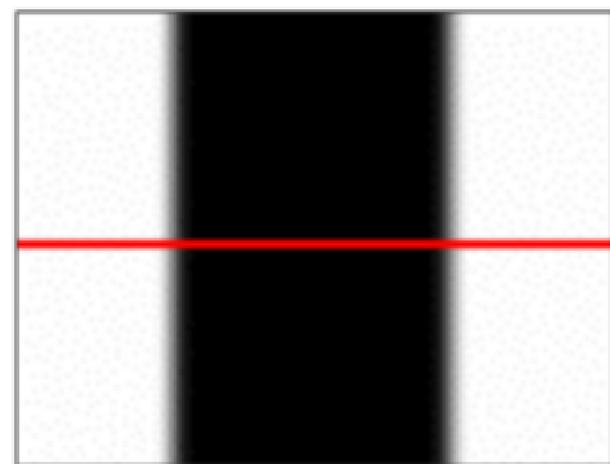
Experimento de Huebel & Wiesel

<https://www.youtube.com/watch?v=IOHayh06LJ4>

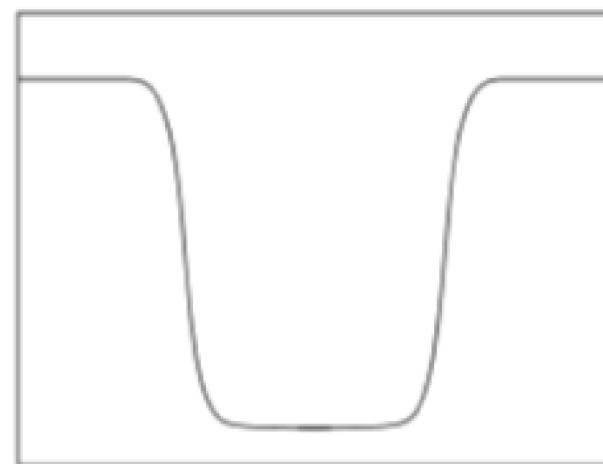
Filtros de Detección de Borde

Caracterización de un Borde

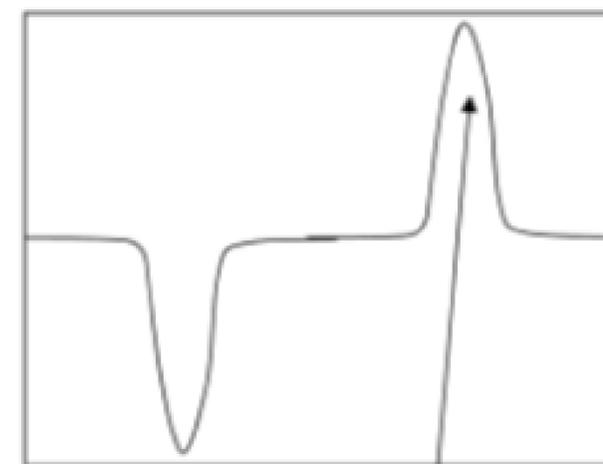
Un borde se detecta en el lugar de rápido cambio de intensidad en la imagen.



Imagen



Función de intensidad a lo largo de la línea de barrido



Puntos de borde (extremos)

Filtros de Detección de Borde

Gradiente de una imagen

$$\nabla I = [g_x, g_y]^T = \left[\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y} \right]^T$$

Magnitud y ángulo del Gradiente

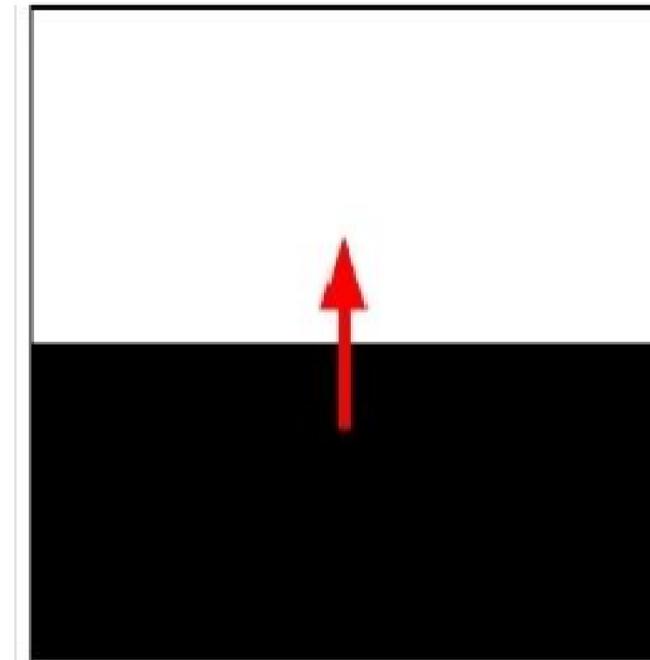
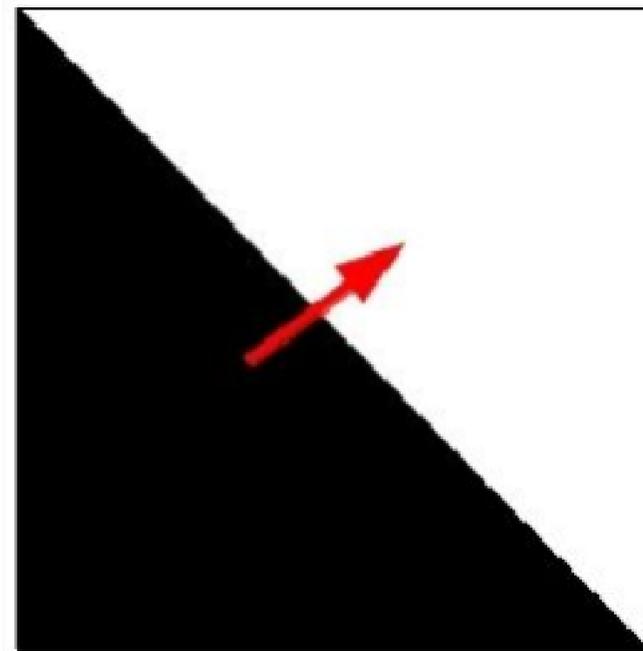
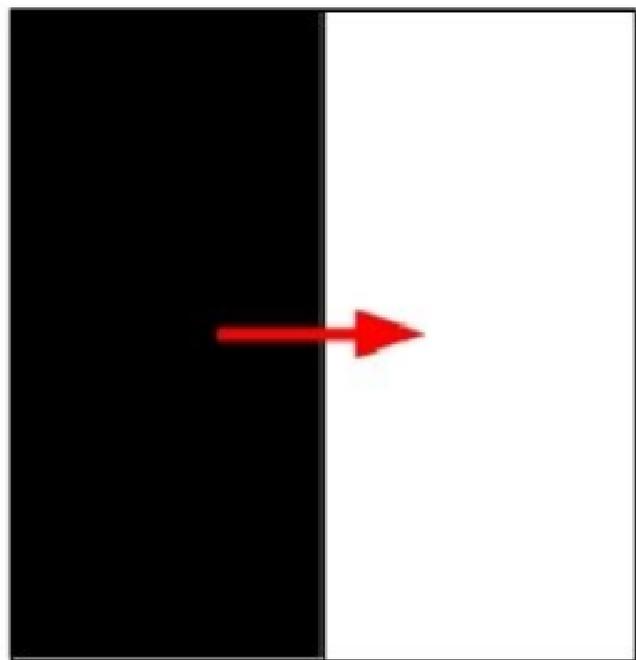
$$M(x, y) = \text{mag}(\nabla I) = \sqrt{g_x^2 + g_y^2}$$

Ángulo del Gradiente

$$\alpha(x, y) = \tan^{-1} \left[\frac{g_y}{g_x} \right]$$

Filtros de Detección de Borde

Dirección del Gradiente (dirección de máximo cambio)



Filtros de Detección de Borde

Sea $f(x,y)$ una función de dos variable, la derivada parcial es:

$$\frac{\partial f(x, y)}{\partial x} = \lim_{\varepsilon \rightarrow 0} \frac{f(x + \varepsilon, y) - f(x, y)}{\varepsilon}$$

Para el caso de imágenes :

$$\frac{\partial f(x, y)}{\partial x} \approx \frac{f(x + 1, y) - f(x, y)}{1}$$

Entonces es posible aplicar filtrado lineal [correlación o convolución]

Filtros de Detección de Borde

Filtro de Sobel

Filtro que aproxima el gradiente respecto a x e y.

mask_y

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

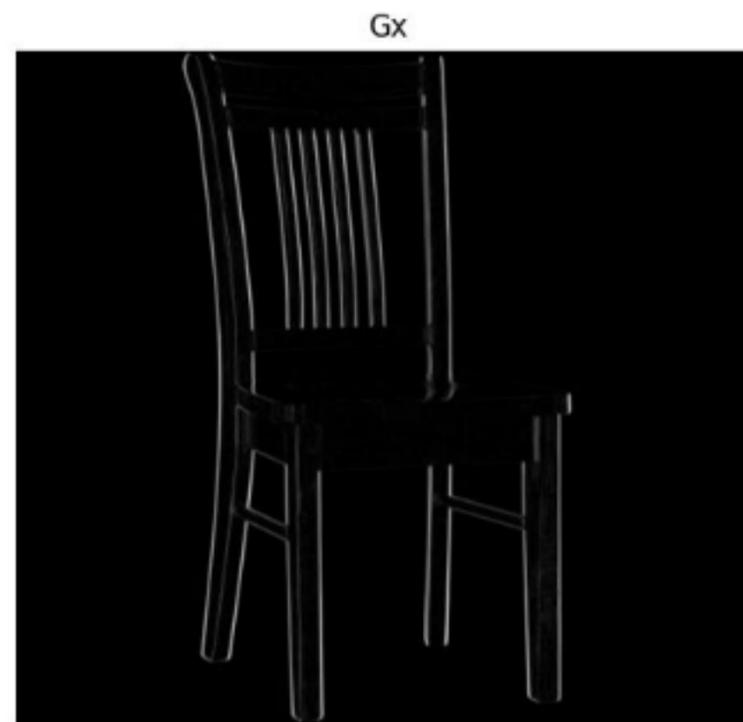
mask_x

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

Filtros de Detección de Borde

Filtro de Sobel

Filtro que aproxima el gradiente respecto a x e y.



Filtros de Detección de Borde

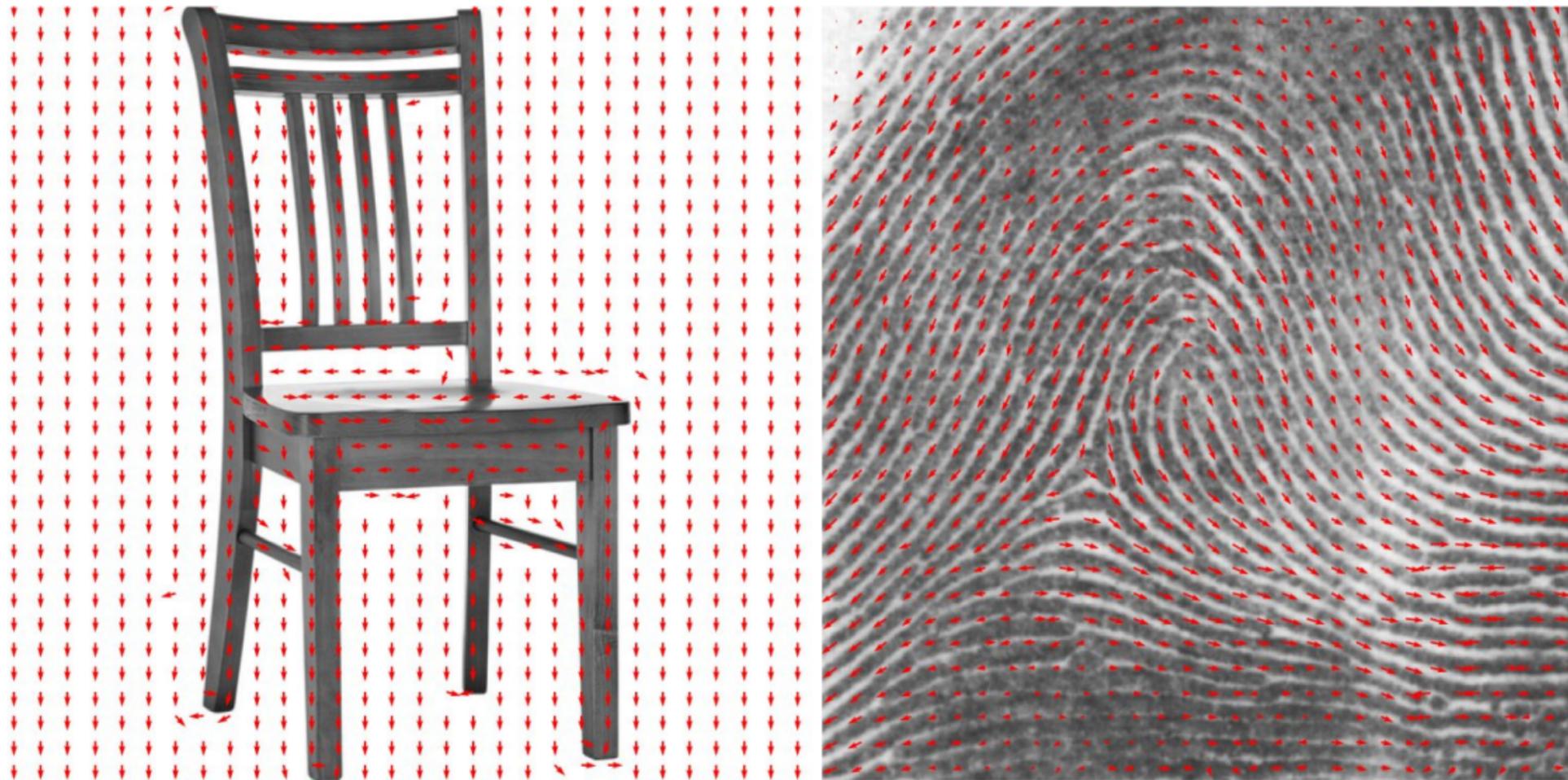
Filtro de Sobel



Imagen derecha: representación de la magnitud del gradiente calculado en cada punto de la imagen (es muestra la imagen en negativo para mejorar la visualización).

Filtros de Detección de Borde

Filtro de Sobel



Campo de Gradientes

Filtros y Aprendizaje de Máquinas

Machine Learning

A machine learning algorithm is an algorithm that is able to learn from data

$$y = H_{\mathbf{w}}(\mathbf{x})$$

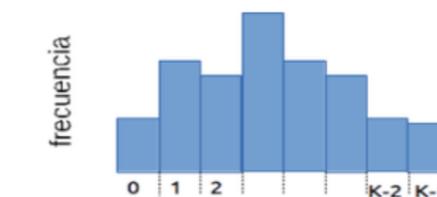
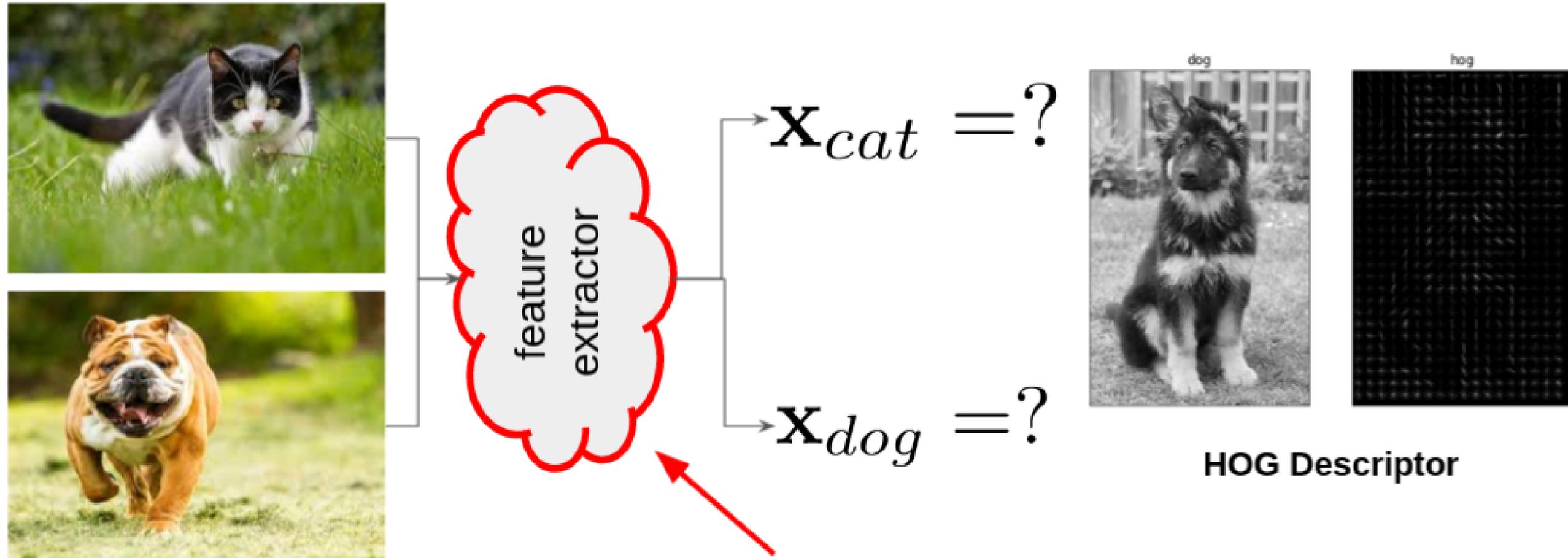
experience (data)

$$\mathbf{x} \in \mathbb{R}^d$$

Given : experience (E), task (T) and performance measure (P).
Learning allows P to improve with E in T

Feature Vector (description)

Unstructured data



distribución de orientaciones de los graientes

Histograma de Orientaciones

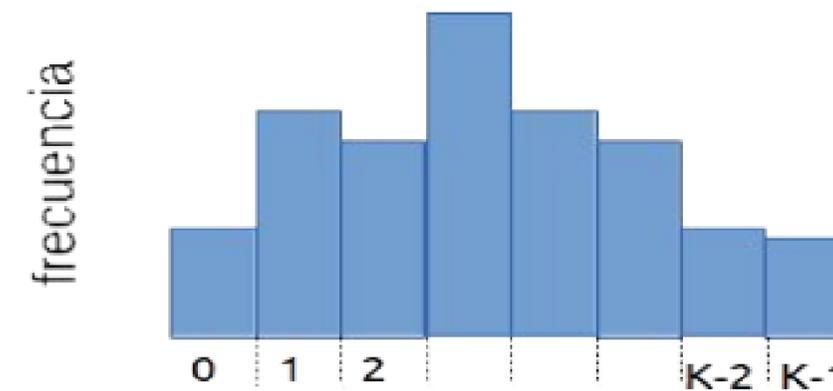
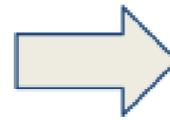
Representa de distribución de orientaciones de los gradientes calculados en cada pixel de una imagen.



$$\alpha = \tan^{-1} \left(\frac{g_y}{g_x} \right)$$

$$0 \leq \alpha \leq \pi$$

cuantizar orientación en K bins



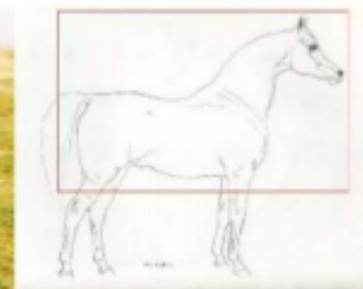
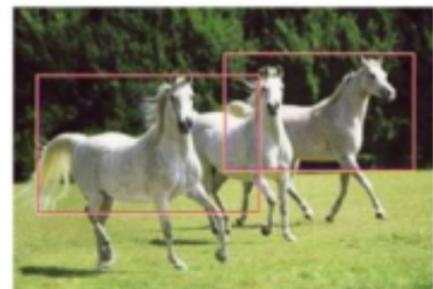
$$\text{round}[(\alpha/\pi) \cdot K]$$

HARD ASSIGNMENT

Histogram of Oriented Gradients [HOG]



Detección
de Objetos



[Navneet Dalal and Bill Triggs. 2005. Histograms of Oriented Gradients for Human Detection. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01 (CVPR '05), Vol. 1. IEEE Computer Society, Washington, DC, USA, 886-893]

Fácil



Difícil



Fácil

7210414959
0690159784
9665407401
3134727121
1742351244

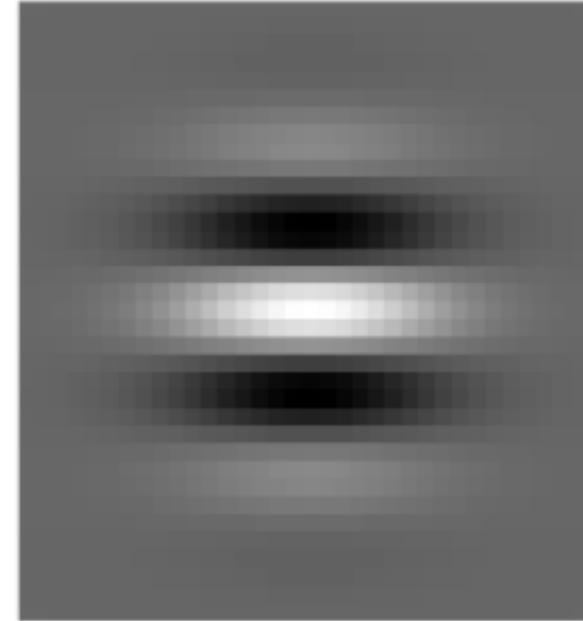
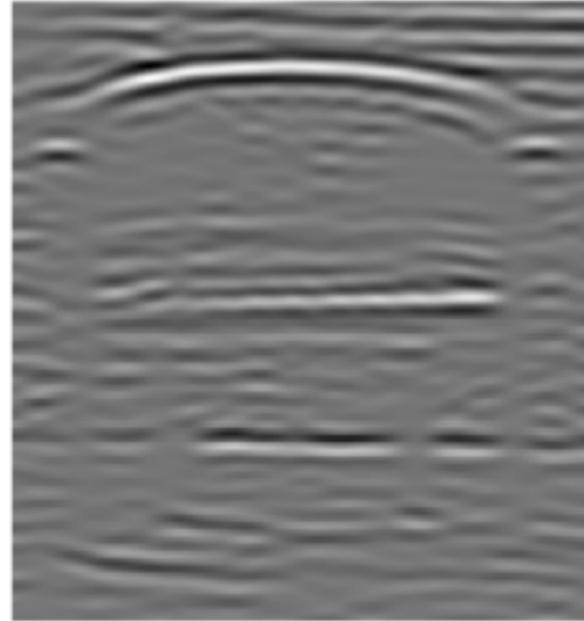
Difícil

117.718 / 5.006.33
√106 3636 111.000
22.000. — 93.922 60

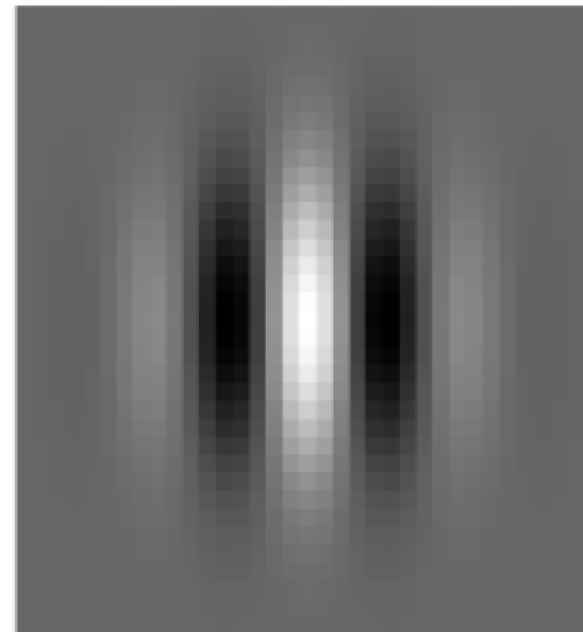
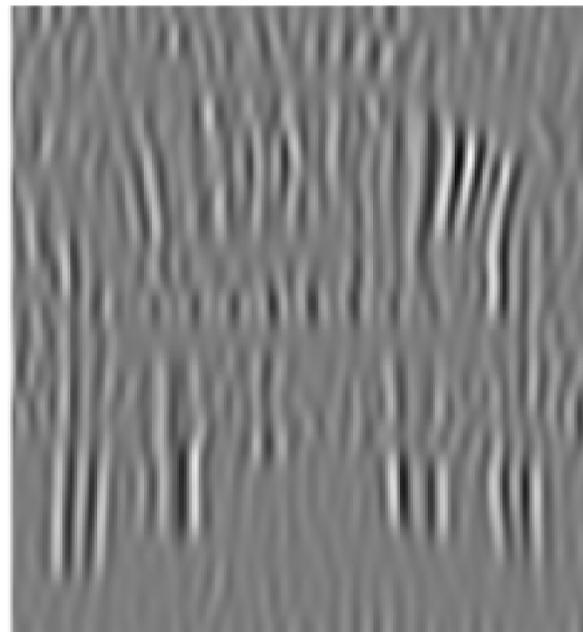
200.000 = 250.000 —
42.000 = 178.200 —
40.000 / 56.000 14.250,5

Filtros como Extractores de Características

Filtros para Detectar Bordes

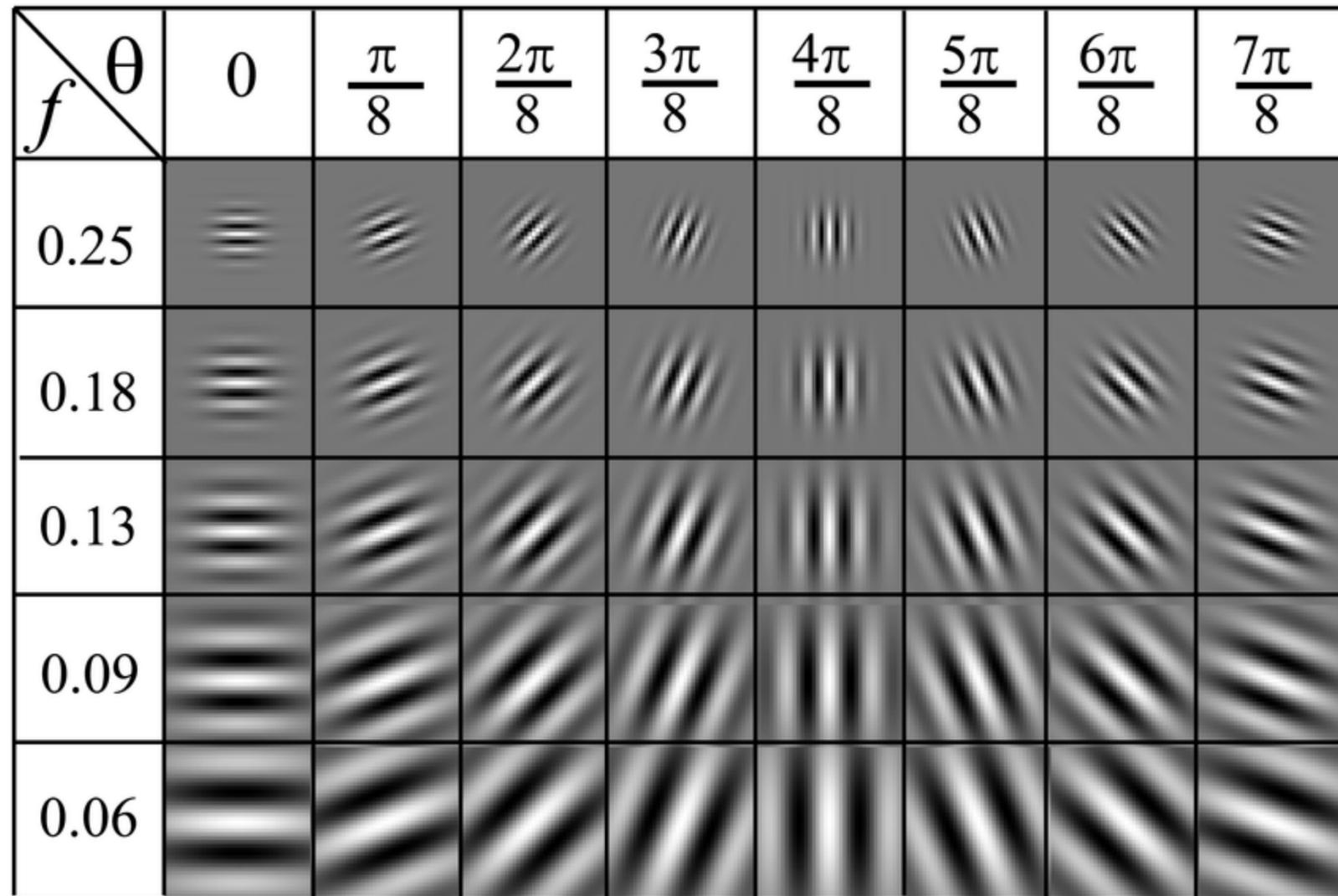


Podemos entender la operación de convolución como un extractor de características

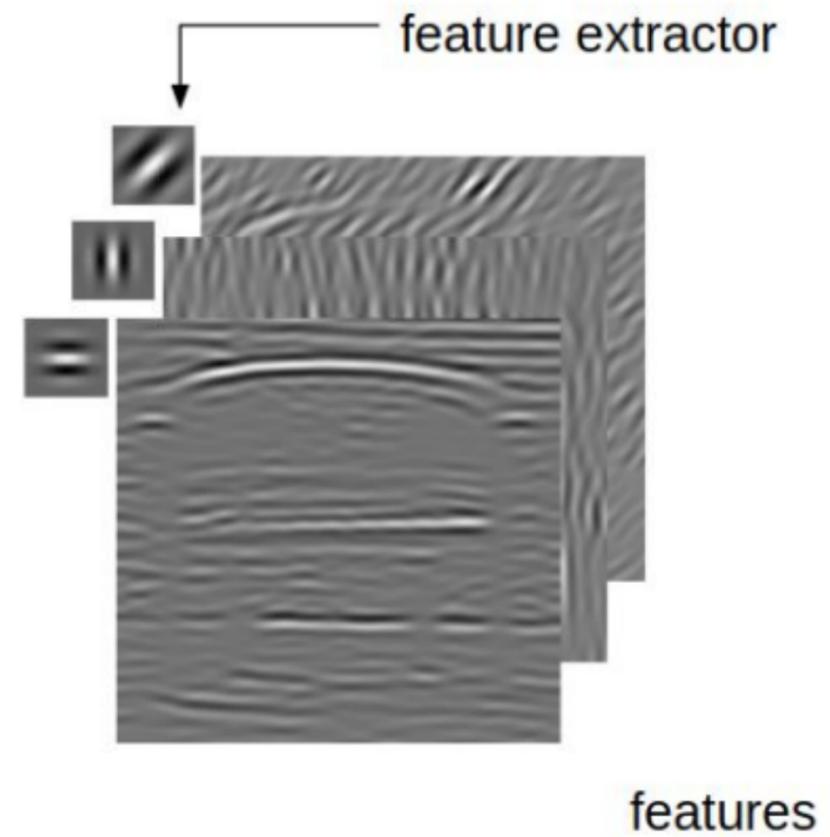


Banco de Filtros [Gabor]

$$F(u_1, u_2) = \exp\left(-\frac{(\hat{u}_1^2 + \gamma^2 \hat{u}_2^2)}{2\sigma^2}\right) \times \cos\left(\frac{2\pi}{\lambda} \hat{u}_1\right),$$
$$\hat{u}_1 = u_1 \cos \theta + u_2 \sin \theta \quad \text{and}$$
$$\hat{u}_2 = -u_1 \sin \theta + u_2 \cos \theta,$$

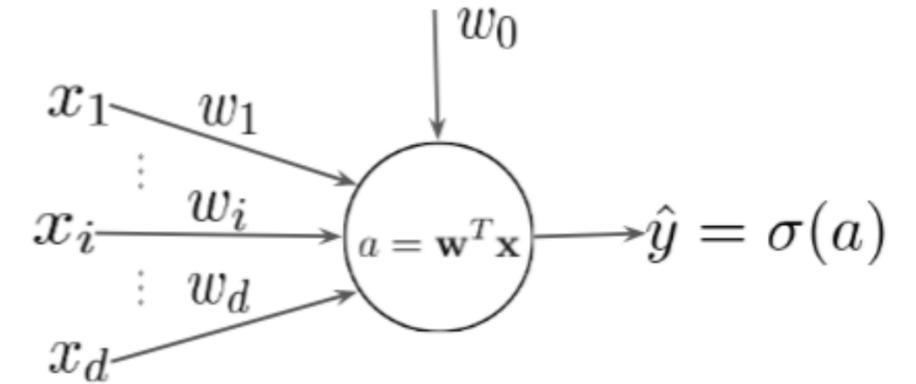


¿Podremos aprender los filtros?



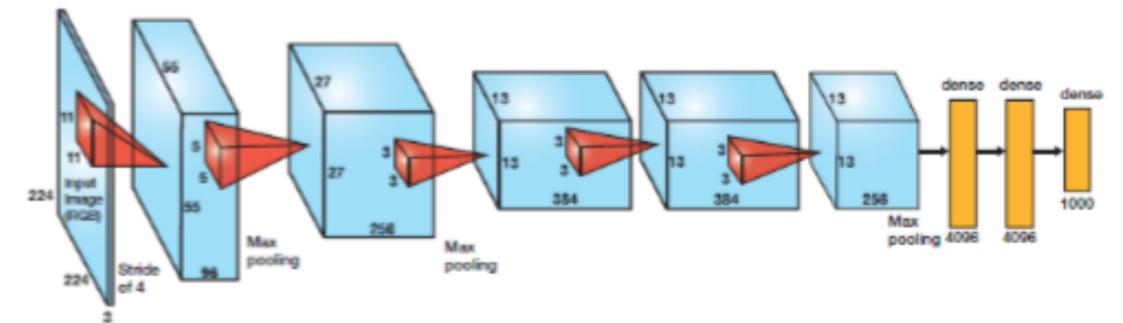
Deep Learning (DL)

Familia de modelos de redes neuronales, que contienen muchas capas de unidades de procesamiento llamadas **neuronas**. La primeras capas tratan de extraer información de bajo nivel, y mientras avanzamos en profundidad, el nivel semántico de las características extraídas aumenta.



Convolutional Neural Network (CNN)

Un tipo de red neuronal, formada principalmente por capas convolucionales. Estas capas se relacionan con la operación de convolución (procesamiento de imágenes), y permiten extraer características a través de filtros (kernels) aprendibles. Una convolución, al igual que una neurona, aplica una operación lineal (producto punto).





Clase 2.2

- Redes Neuronales Convolucionales