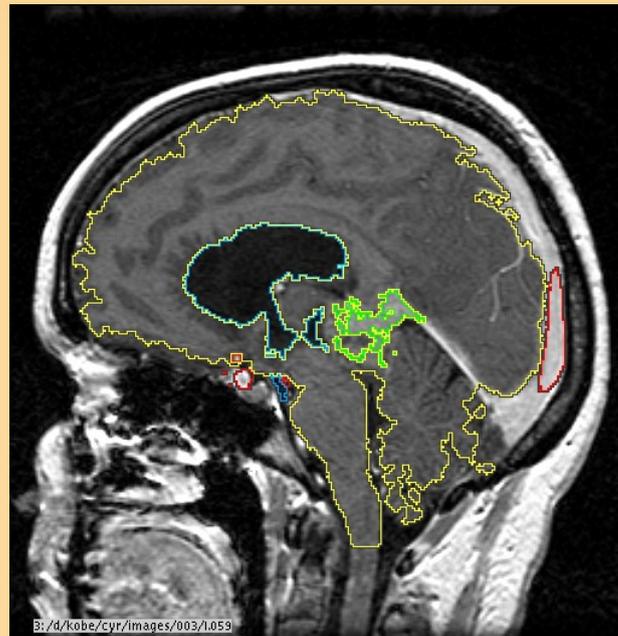


# Procesamiento Digital de Imágenes

Por: José M. Saavedra R.



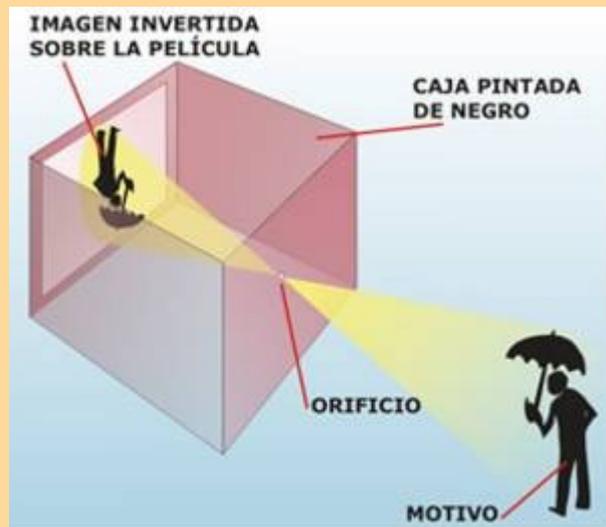
# Introducción

- Área de la Ciencia de la Computación que tiene como objetivo aplicar ciertas operaciones sobre una imagen digital para facilitar su análisis posterior.



# Introducción

- Acontecimientos preliminares:
  - **La Fotografía:** Obtener imagen por acción de la luz. Basado en el principio de la *caja oscura (pinhole camera)*.
  - 1839, Joseph Niépce y Louis Daguerre.



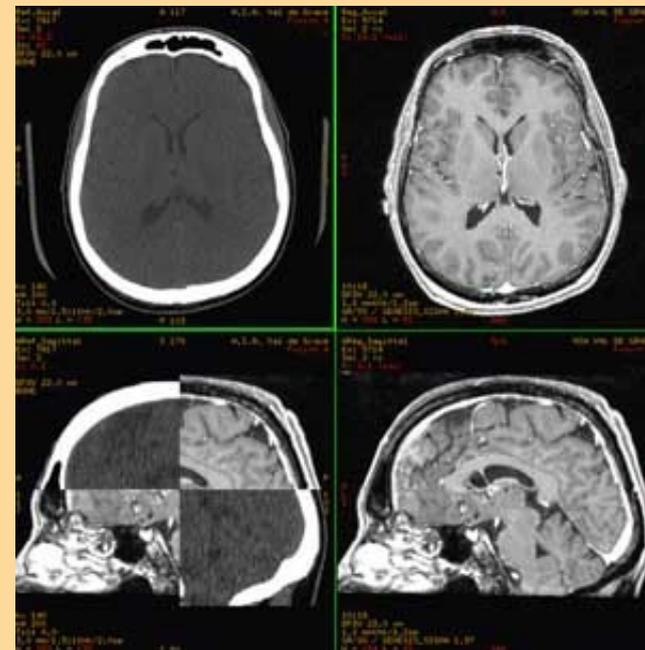
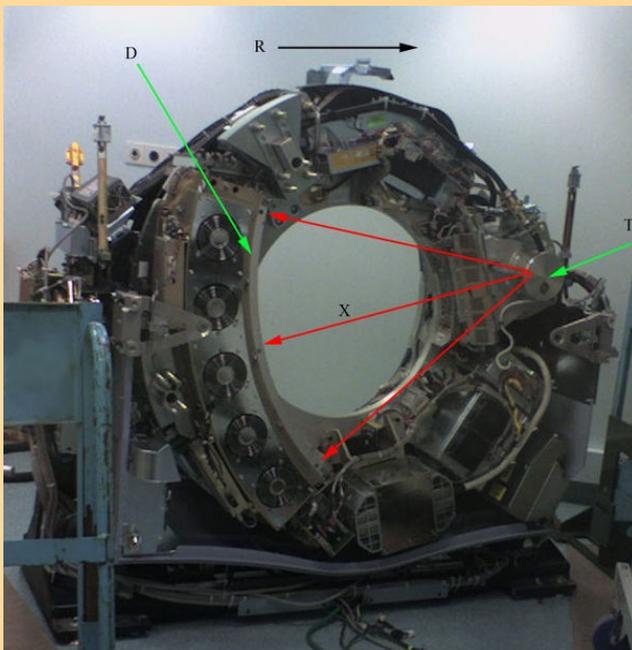
# Introducción

- Acontecimientos preliminares:
  - Rayos X
  - 1895, Wilhelm Conrad Roentgen



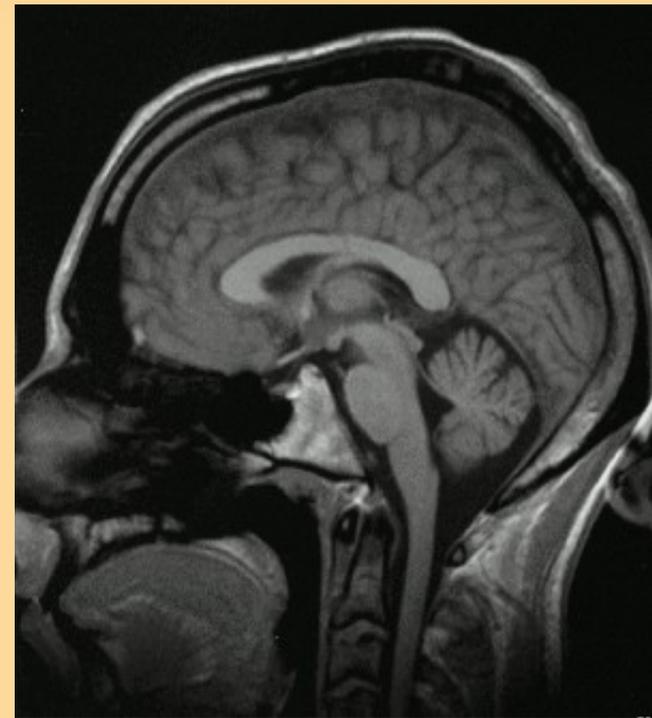
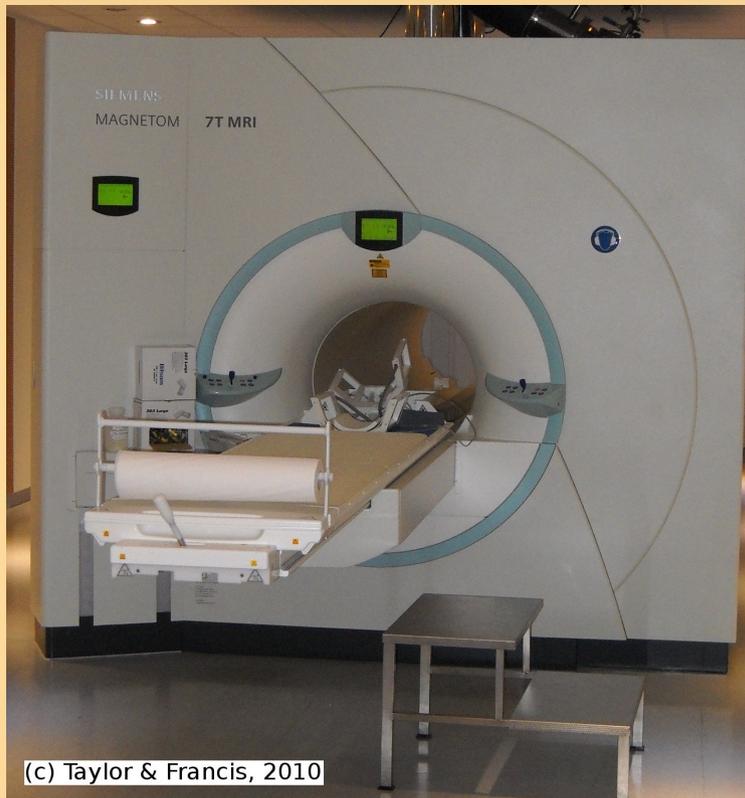
# Introducción

- Acontecimientos preliminares:
  - Tomografía Axial Computarizada: Extensión de los rayos X, varios rayos X en diferentes direcciones.
  - 1967, Allan McLeod Cormack & Godfrey Newbold Hounsfield.



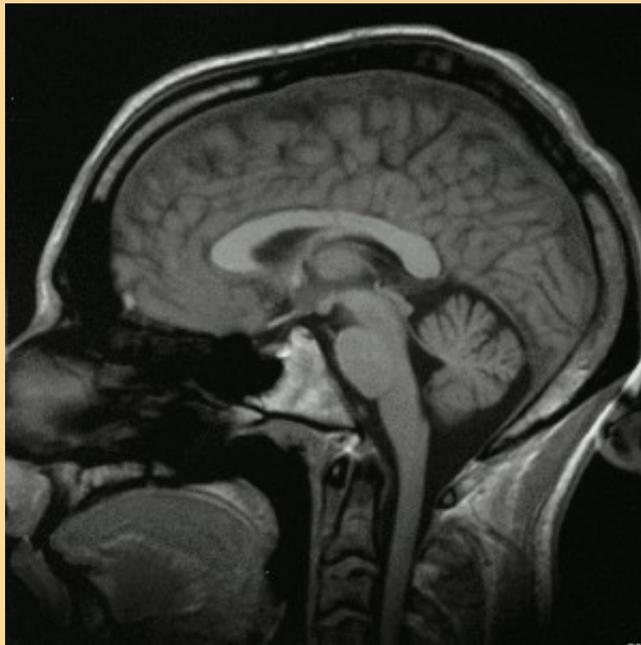
# Introducción

- Acontecimientos preliminares:
  - Resonancia Magnética
  - Introducido en el campo médico en 1980.



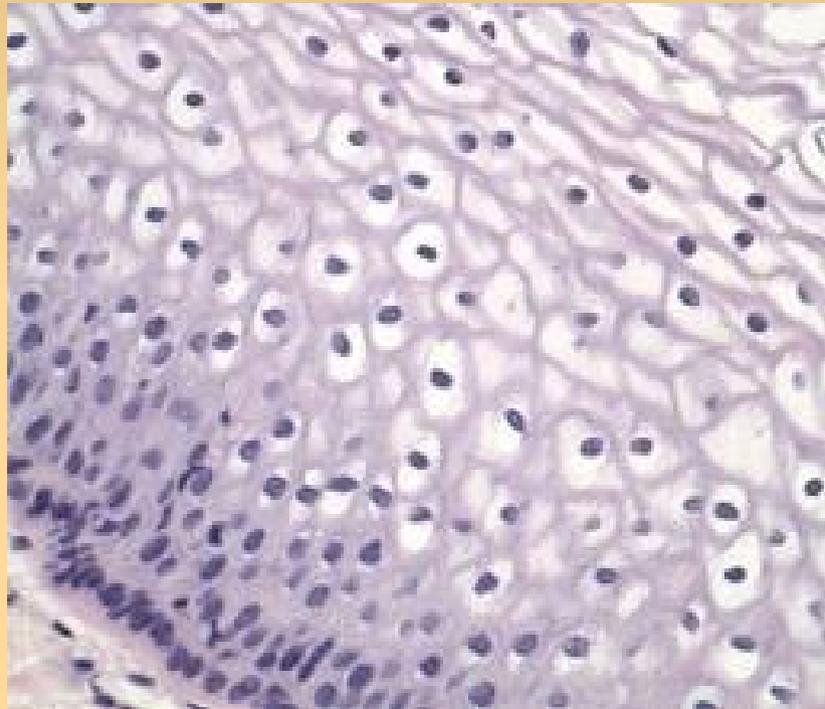
# Introducción

- Acontecimientos preliminares:
  - Ecografía Ultrasónica
  - Tomografía por emisión de protones, etc.

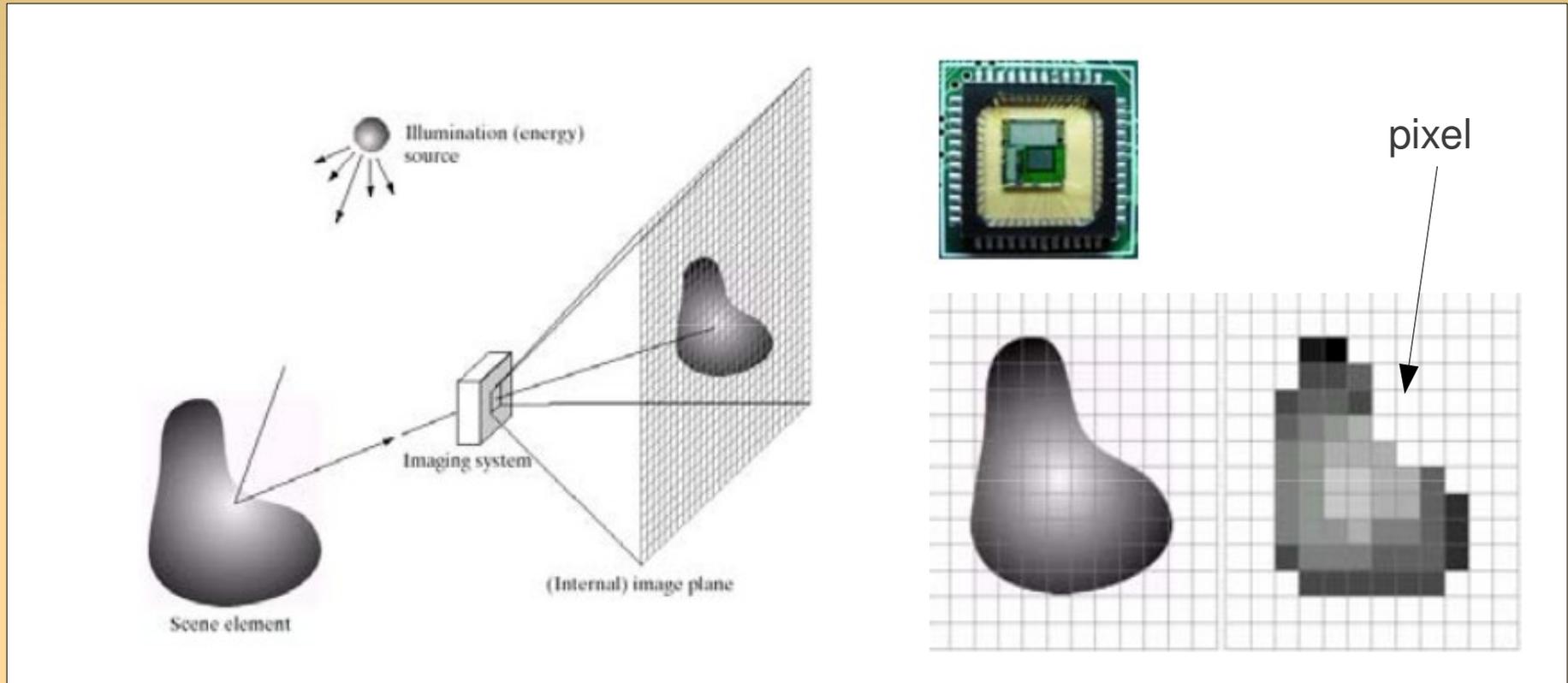


# Introducción

- En la actualidad (Bioanálisis)



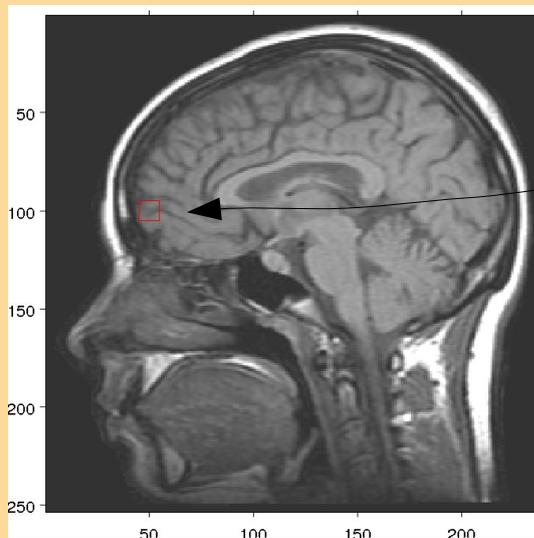
# La Imagen Digital



Analógica > Discreta

# La Imagen Digital

- **Muestrea** (samplig) el espacio 2D sobre una grilla.
- **Cuantiza** (quantization) cada muestra a un valor entero.
- Por lo tanto, una imagen digital puede ser representada por una matriz(grilla) de valores enteros.



77	79	84	92	100	106	113	121	124	119	112
74	82	95	109	116	118	120	127	131	127	118
81	86	96	108	118	124	127	131	130	124	116
91	95	99	105	114	122	125	123	118	112	107
95	100	102	104	106	109	110	108	105	100	98
97	103	106	107	108	107	104	101	97	94	94
93	104	110	113	110	105	100	100	105	112	118
92	108	114	111	106	102	104	111	120	127	131
92	110	116	114	111	111	115	122	127	128	128
87	107	118	121	119	117	119	123	128	131	132
84	106	118	120	119	118	119	121	122	123	122

# La Imagen Digital

- Así una imagen  $I$ , puede definirse como:

$$I = [1..M] \times [1..N] \rightarrow V$$

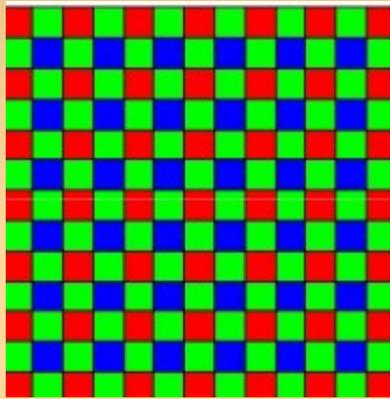
- Comúnmente  $V = \{0, 1, \dots, 255\}$ . Donde 0 representa el color negro y 255 el blanco.  $V$  corresponde a los diferentes valores de grises.
- Así cada valor de  $V$  se puede representar mediante un byte.
- $M$ : Número de filas de la grilla.
- $N$ : Número de columnas de las grilla

## MATLAB

```
>> y=imread('imagen.jpg');  
>> [M, N]=size(y);
```

# La Imagen Digital

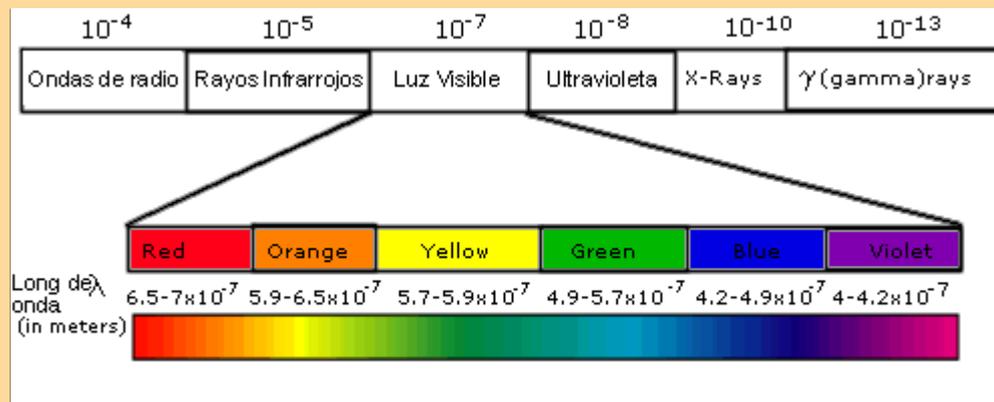
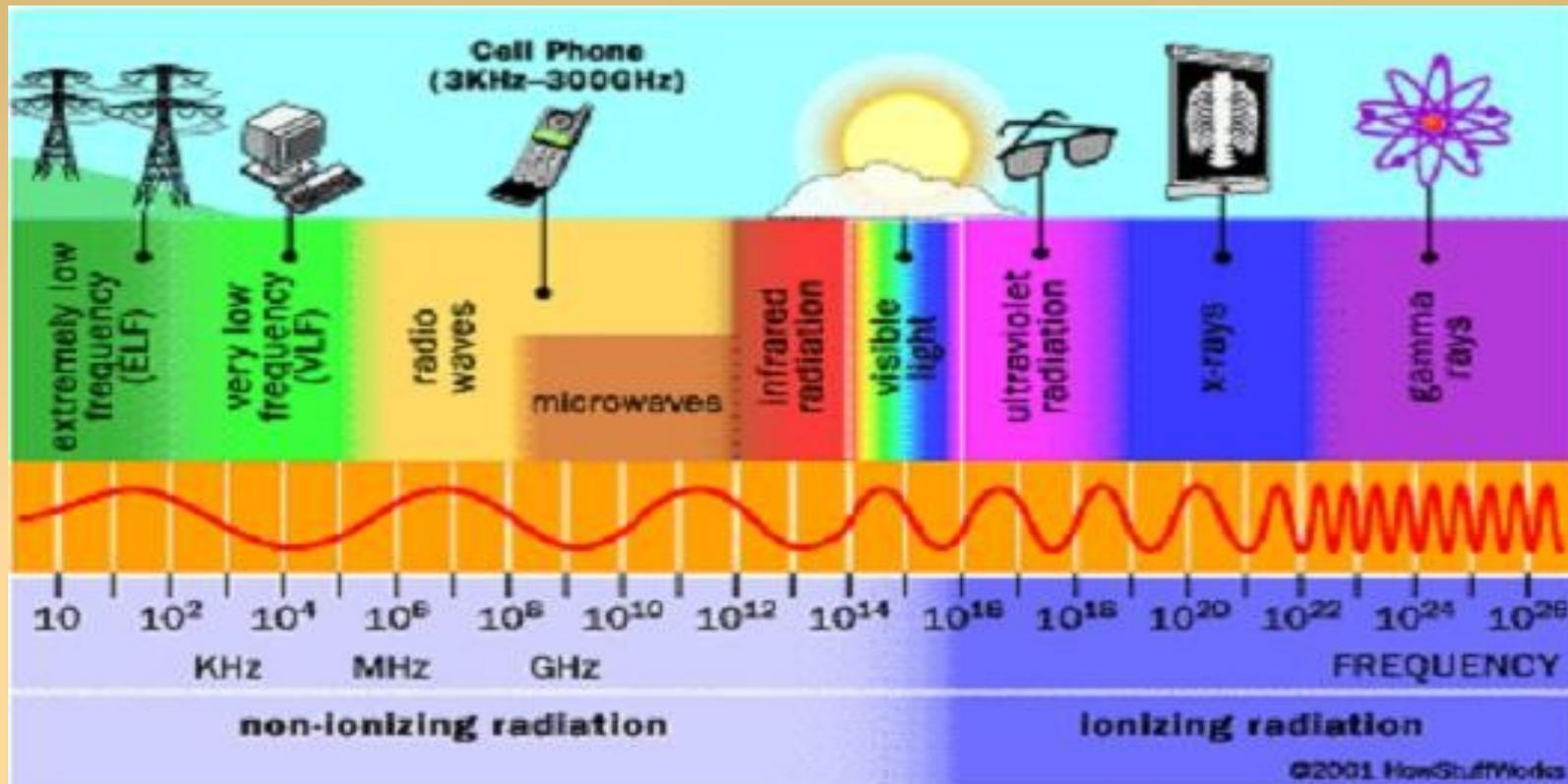
- Imágenes a colores



Filtro RGB

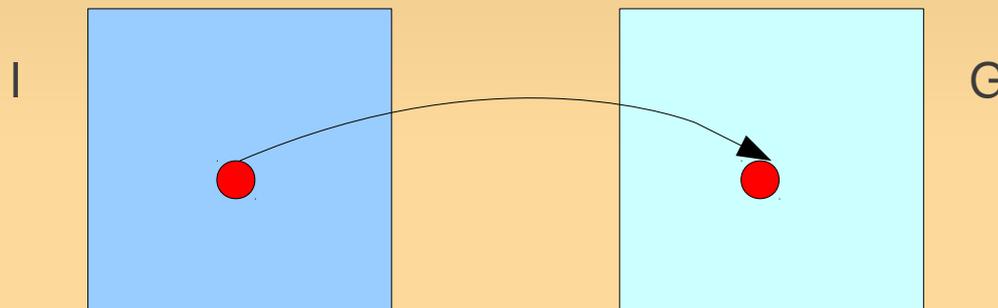


# La Imagen Digital



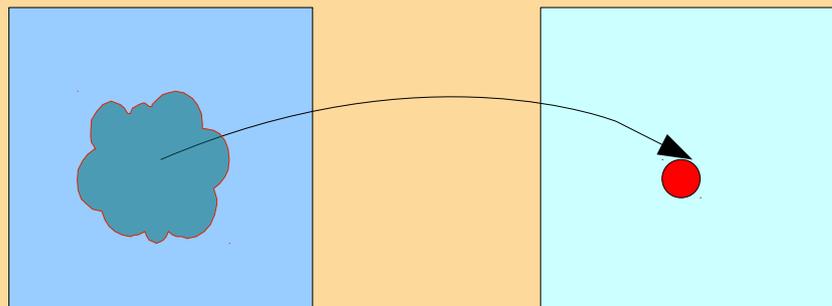
# Operaciones

- Una operación  $T$  sobre una imagen  $I$  permite generar otra imagen  $G$  procesando los pixels de  $I$ .
- Operaciones de Punto



$$G(x) = T(I(x))$$

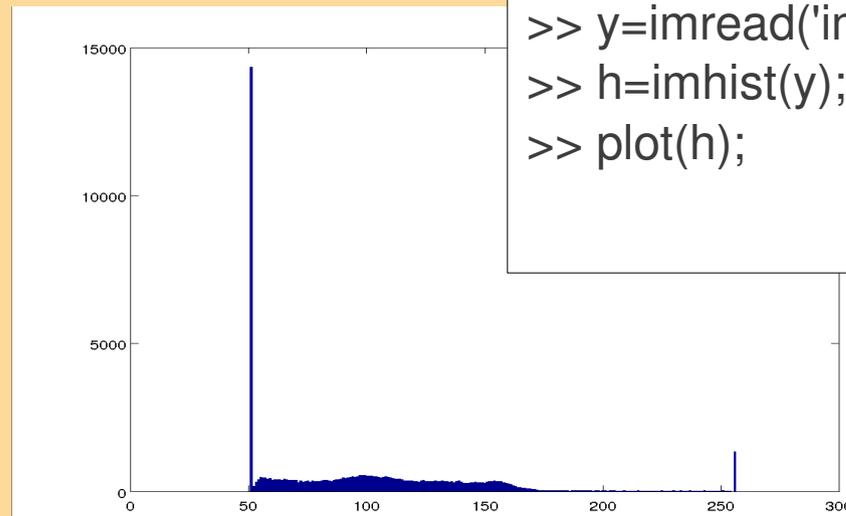
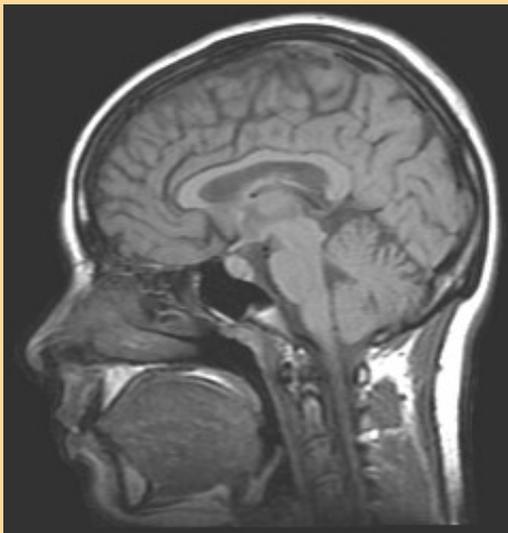
- Operaciones Locales



$$G(x) = T(\text{region}(I, x))$$

# Operaciones

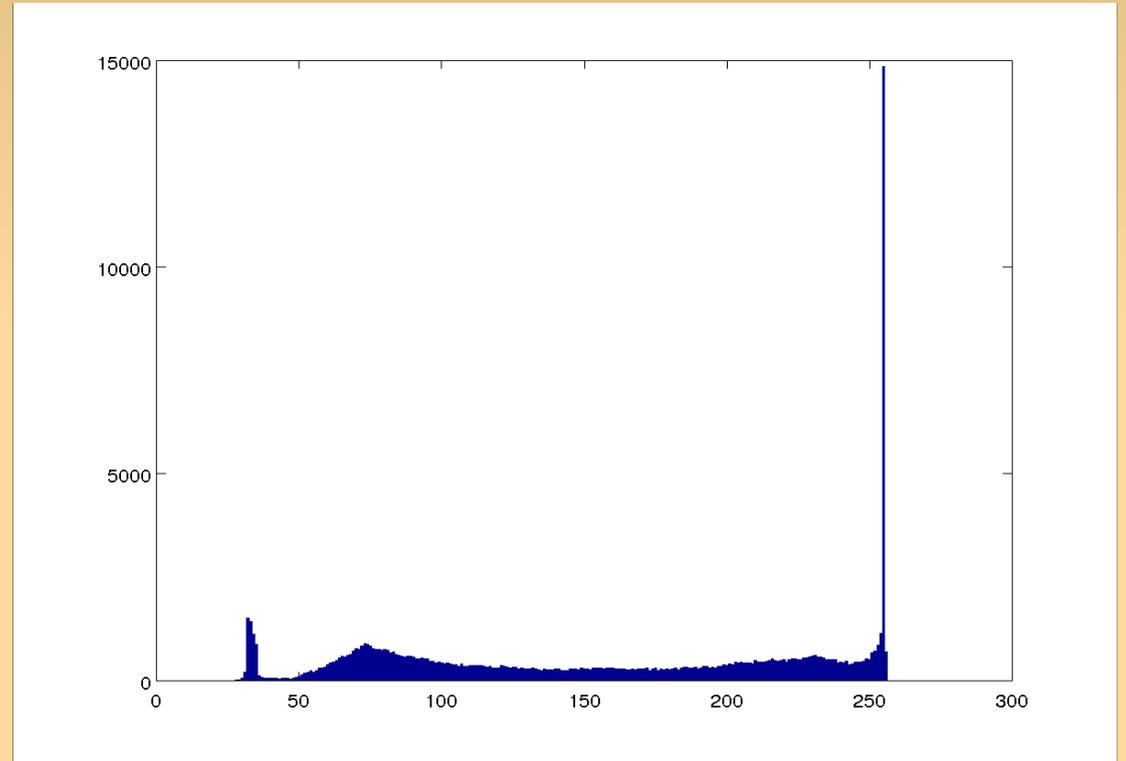
- **El Histograma:** Representa las frecuencias de ocurrencias de cada color [0..255] en una imagen.
- El histograma  $h$  es un vector con 256 entradas, cada entrada representa un color del espectro de colores de la imagen. El valor almacenado en  $h(i)$  indica el número de pixels que tienen el color  $i$ .



MATLAB

```
>> y=imread('imagen.jpg');  
>> h=imhist(y);  
>> plot(h);
```

# Operaciones



# Operaciones de Punto

- Operaciones de Punto
  - Adición (cambia brillo)
  - Multiplicación (cambia contraste)
  - Ajuste del Contraste
  - Binarización

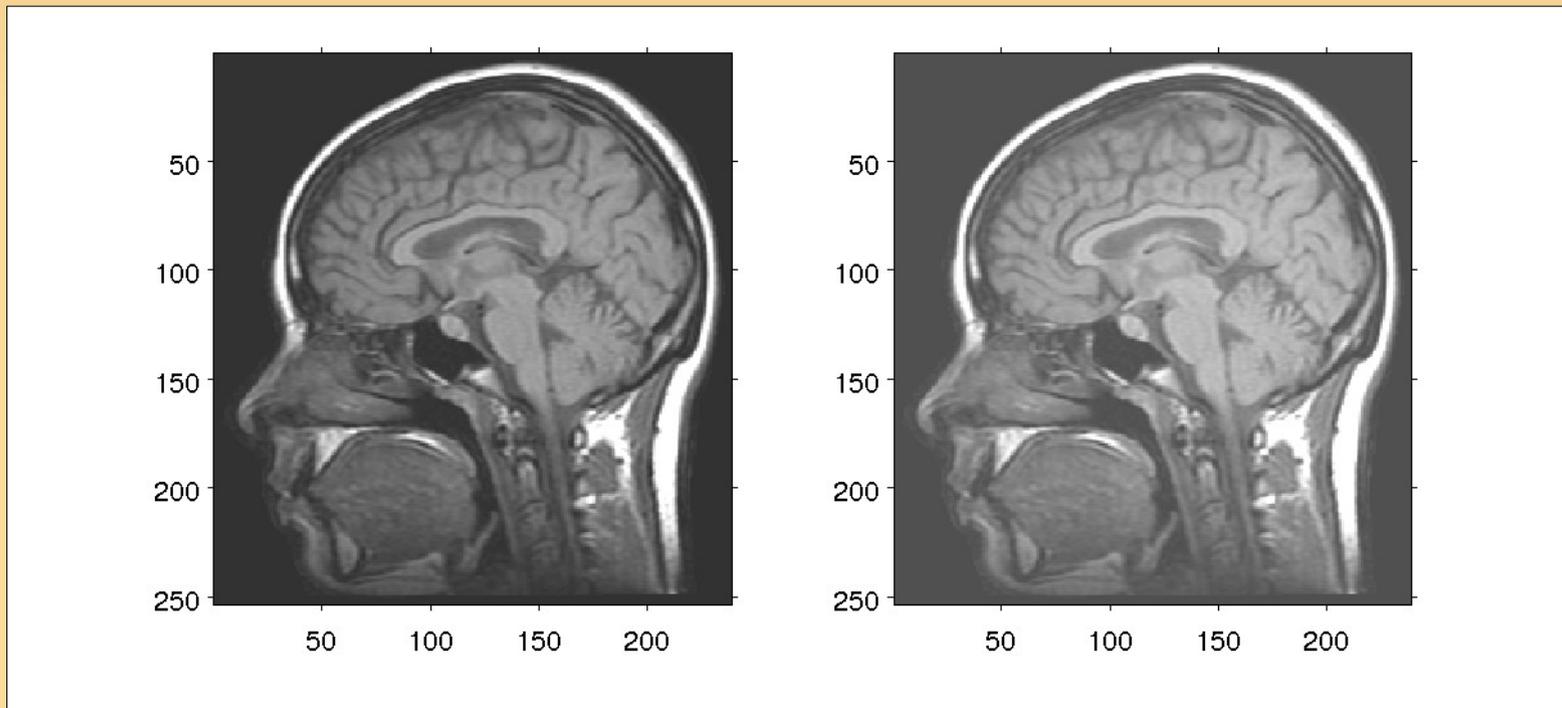
# Operaciones de Punto

- Operaciones de Punto

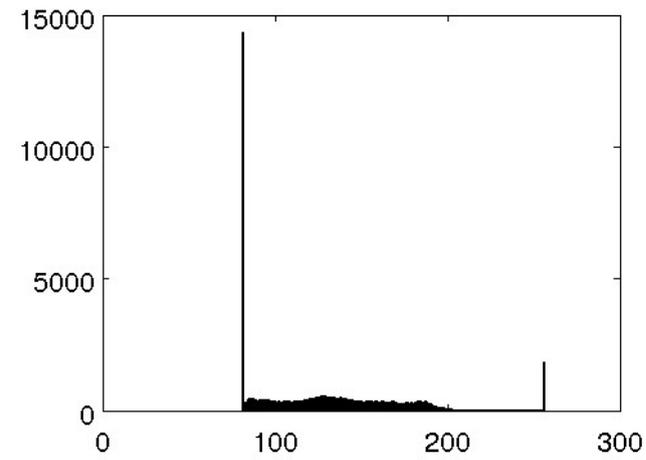
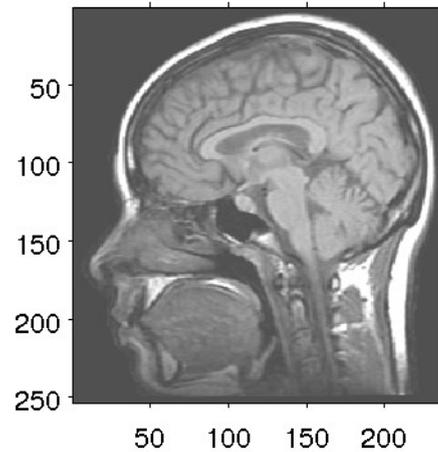
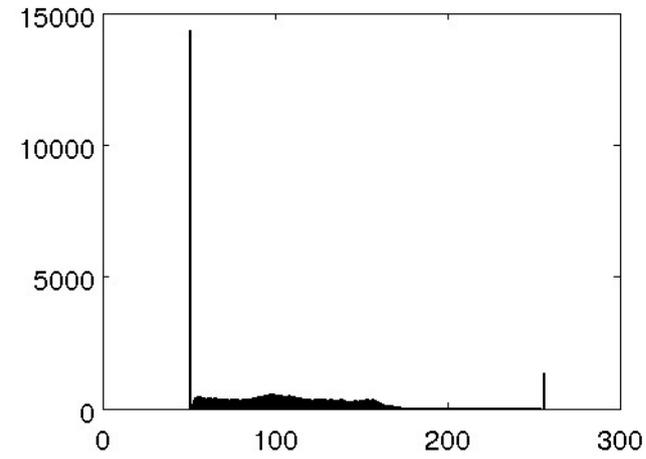
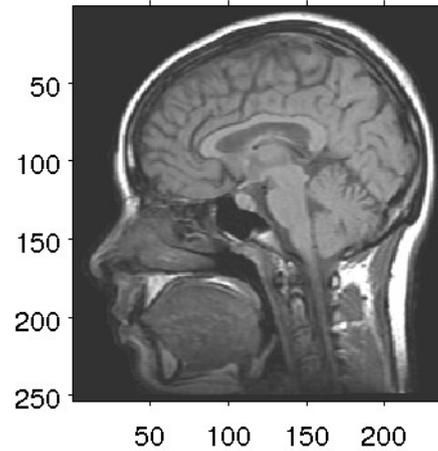
- Adición**

$$g(x)=f(x)+b;$$

Ejemplo:  $b=30$



# Operaciones de Punto

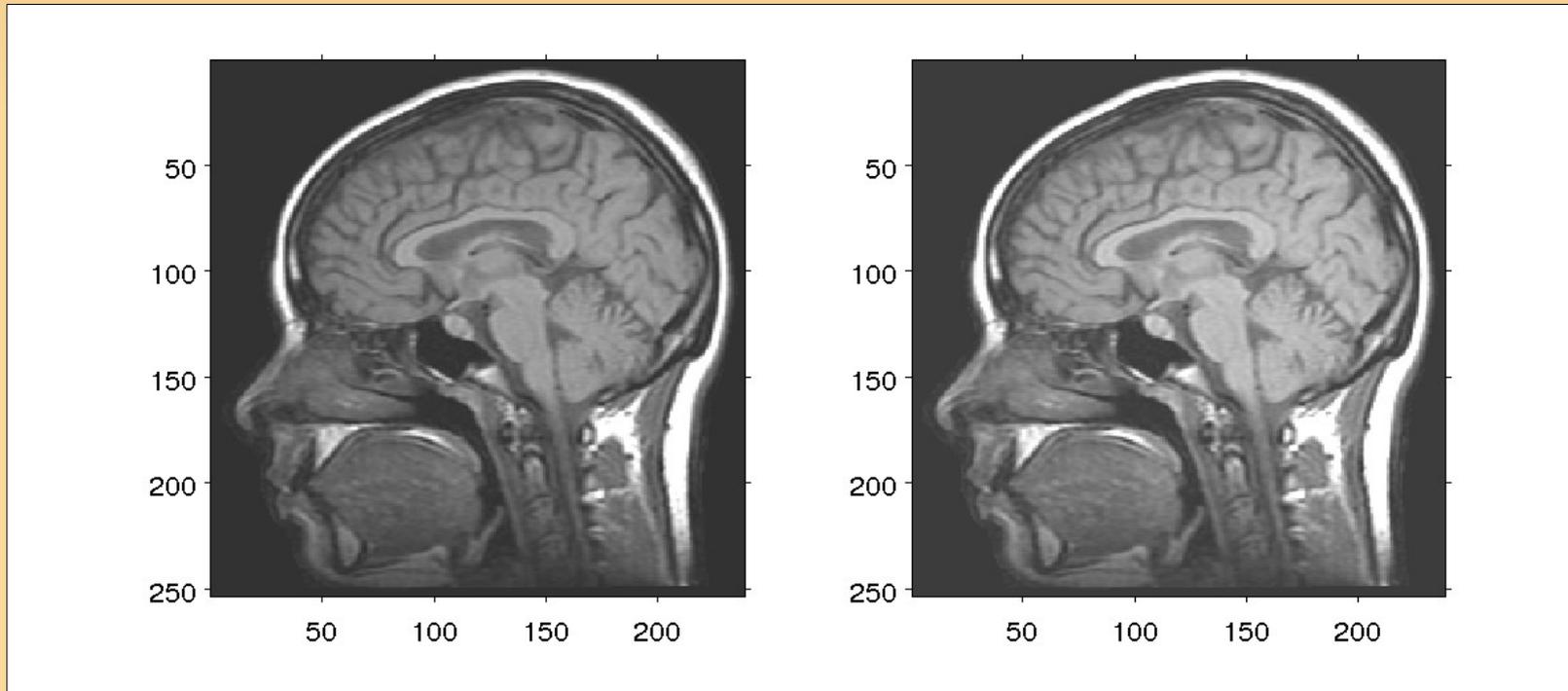


# Operaciones de Punto

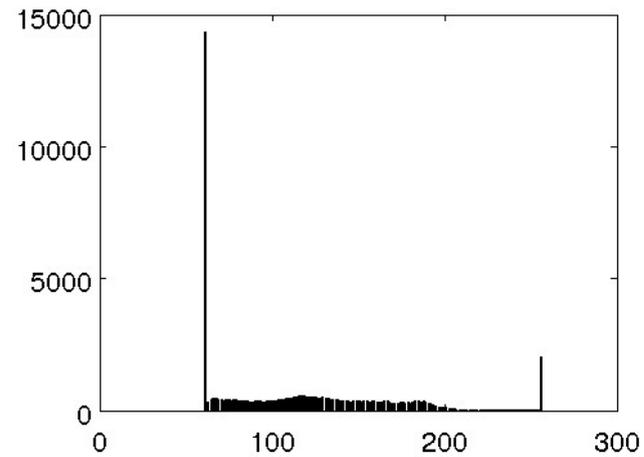
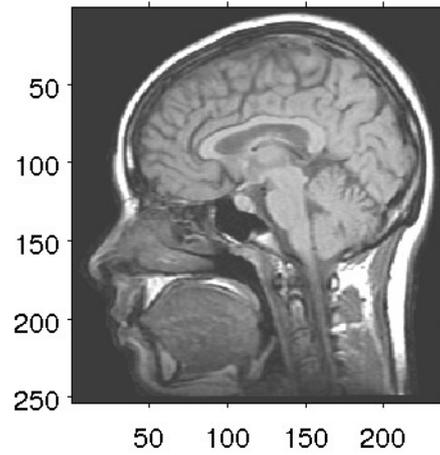
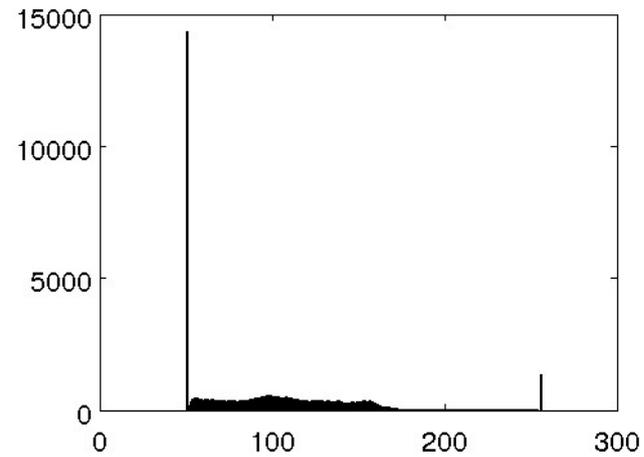
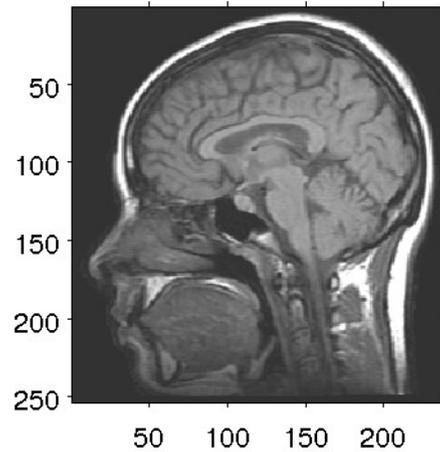
- Operaciones de Punto

- Multiplicacion**

$$g(x)=a*f(x);$$



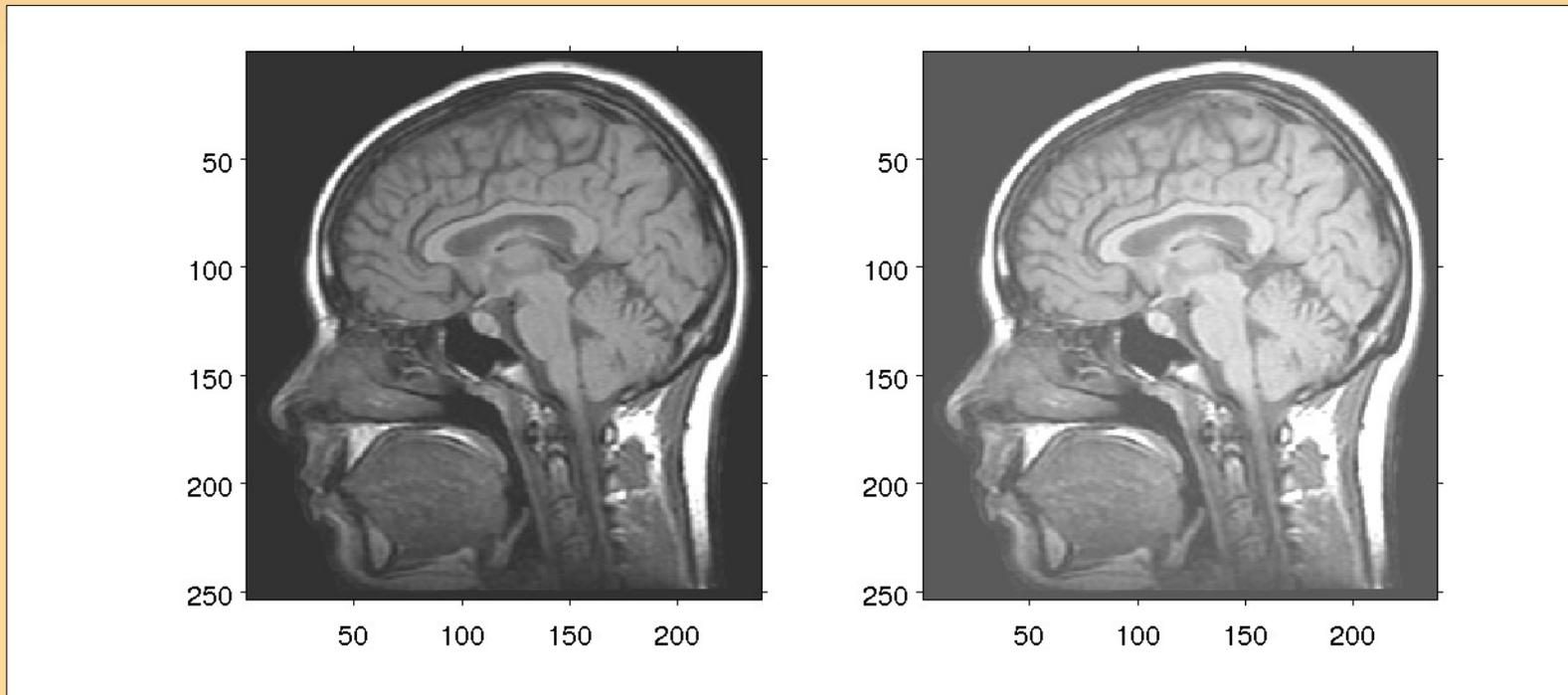
# Operaciones de Punto



# Operaciones de Punto

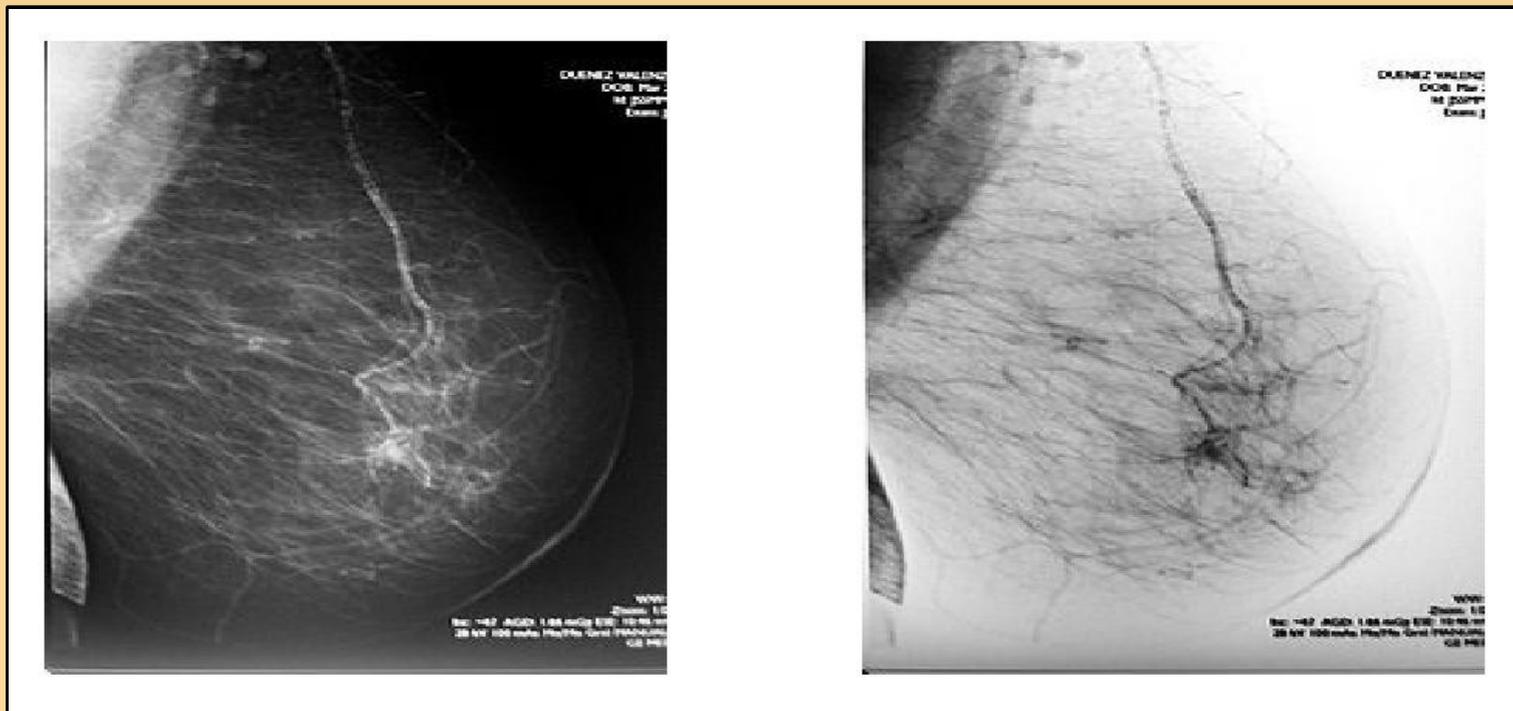
- Operaciones de Punto
  - Adicion+Multiplicacion

$$g(x)=a*f(x)+b; \quad (a: \text{contraste} \quad b: \text{brillo})$$



# Operaciones de Punto

- Operaciones de Punto
  - Negativo  
 $g(x) = \text{max\_tono} - f(x);$

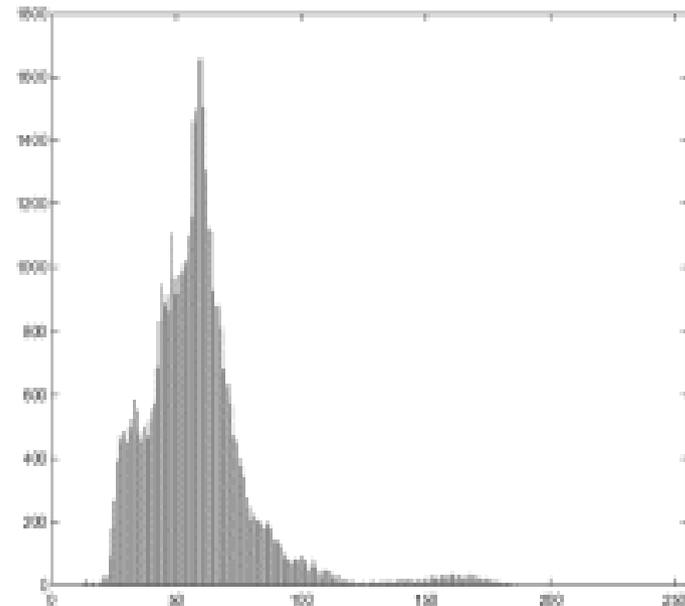


# Operaciones de Punto

- Operaciones de Punto
  - Ajuste de Contraste
  - Contraste: Qué tan bien distribuidos están los pixels en todo el espectro de colores.



Imagen con bajo contraste



Histograma

# Operaciones de Punto

- Operaciones de Punto
  - Ajuste de Contraste
  - Contraste: Qué tan bien distribuidos están los pixels en todo el espectro de colores.

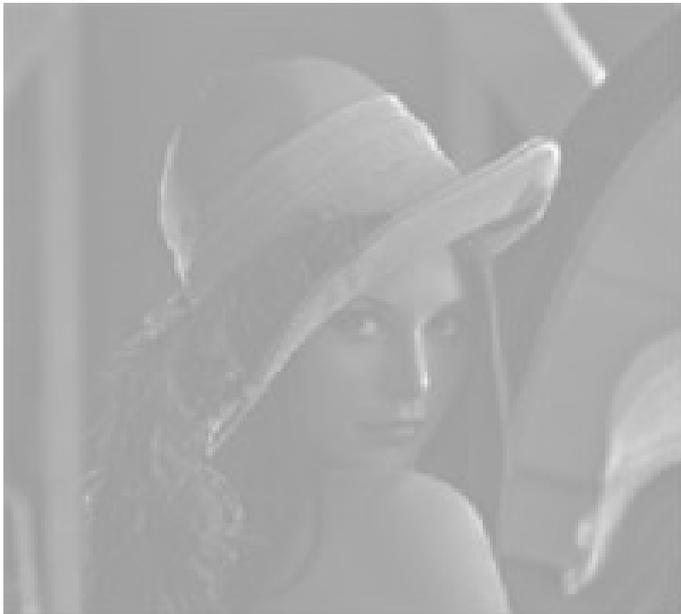
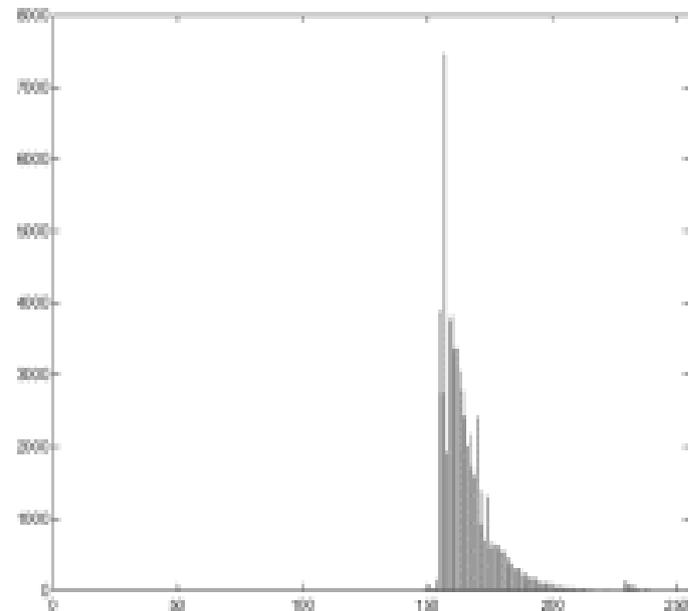


Imagen con bajo contraste



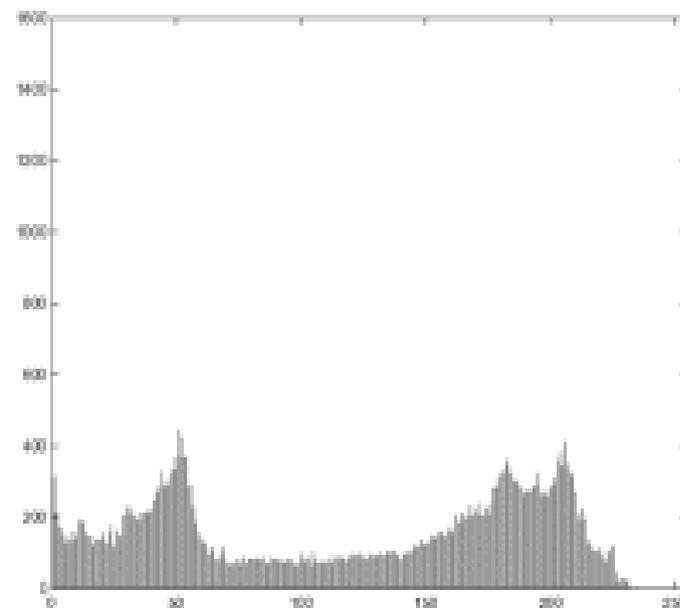
Histograma

# Operaciones de Punto

- Operaciones de Punto
  - Ajuste de Contraste
  - Contraste: Qué tan bien distribuidos están los pixels en todo el espectro de colores.



Imagen con alto contraste



Histograma

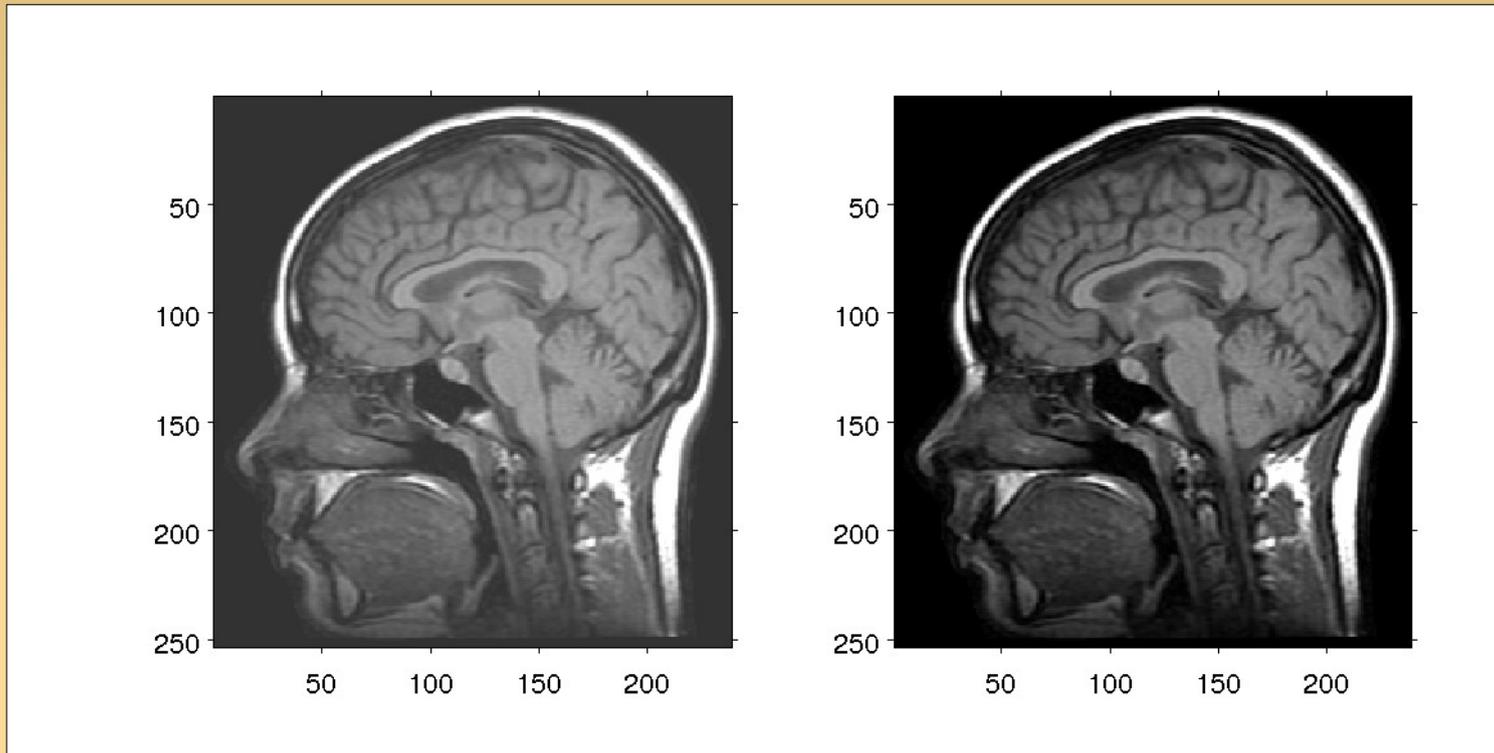
# Operaciones de Punto

- Operaciones de Punto
  - Ajuste de Contraste
  - **Escalamiento Lineal**

$$g(x)=255*[(f(x)-\text{min}_f)/\text{rango}_f];$$

- $\text{min}_f$ : valor del menor tono de gris (color).
- $\text{max}_f$ : valor del mayor tono de gris.
- $\text{rango}_f = \text{max}_f - \text{min}_f$

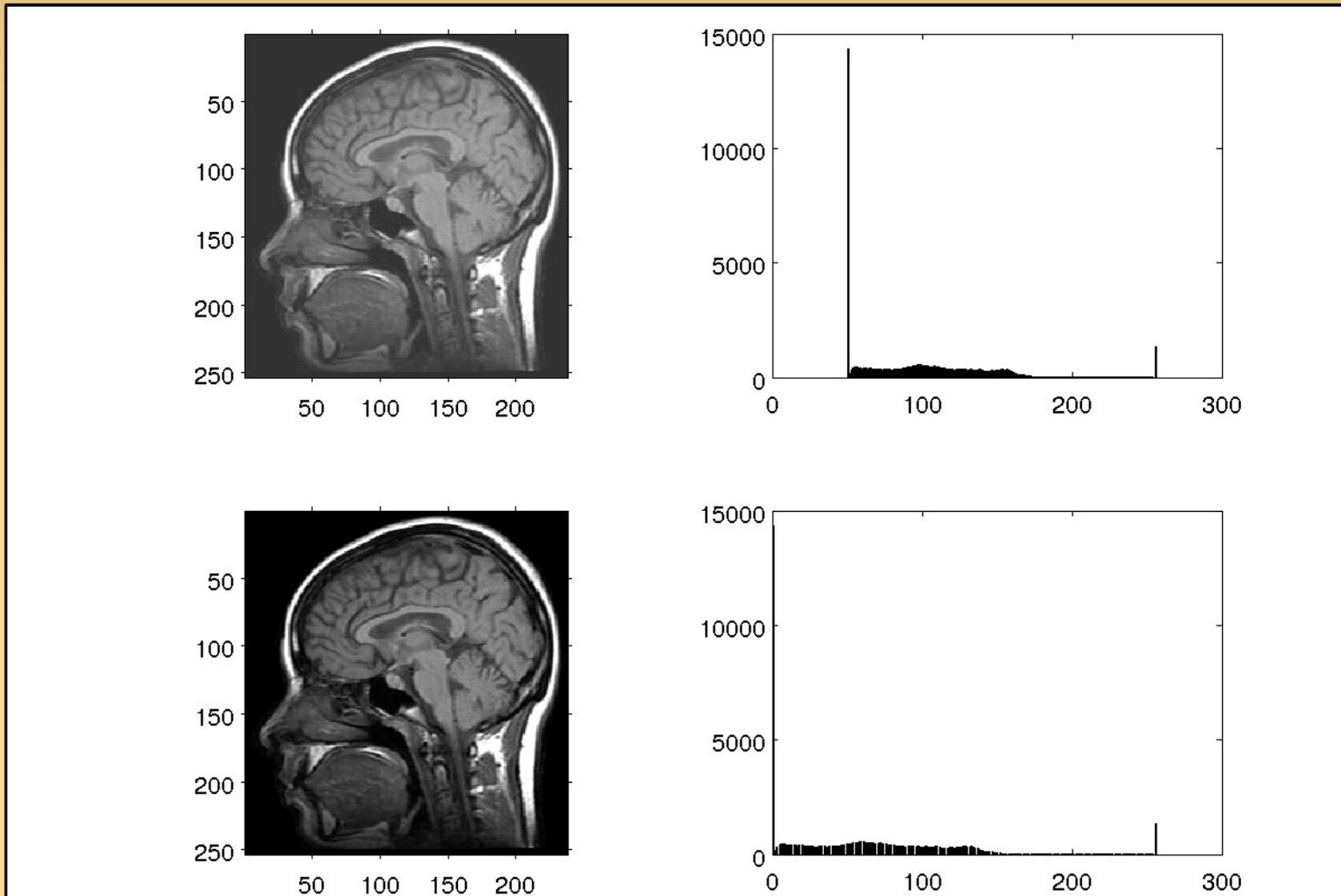
# Operaciones de Punto



Escalamiento Lineal

# Operaciones de Punto

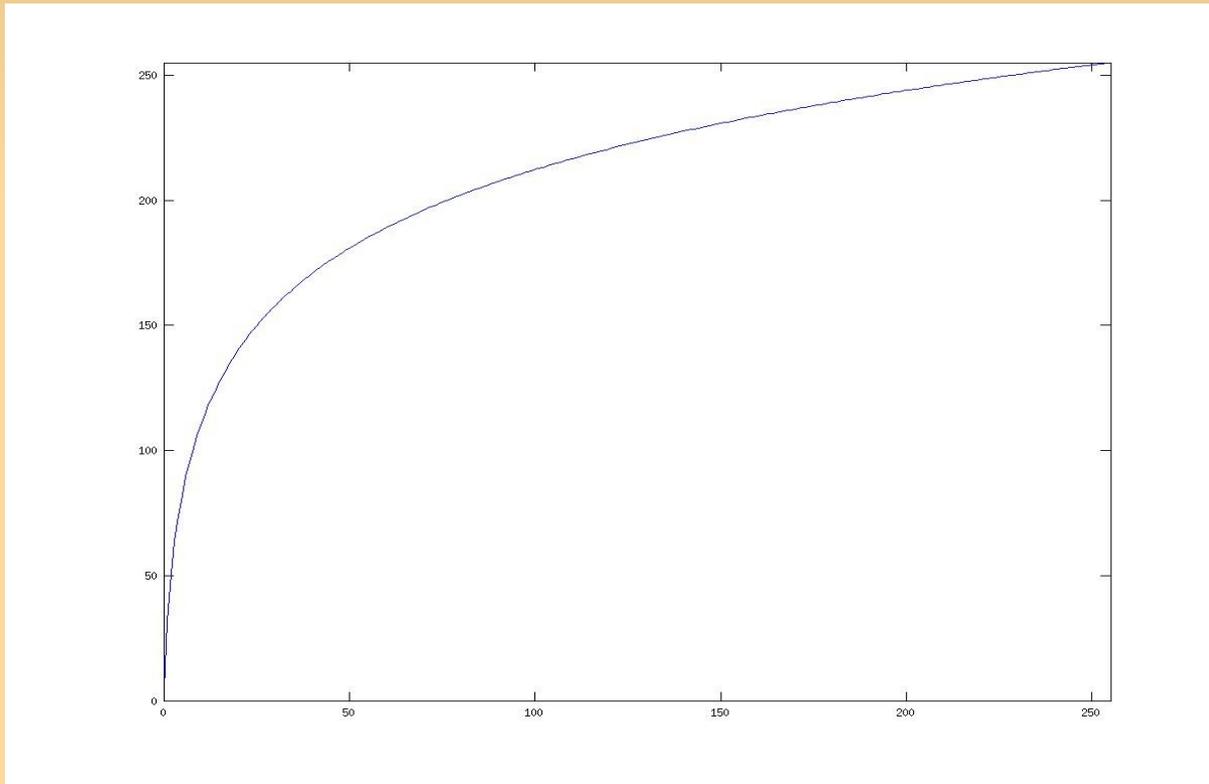
- Cómo se ven los histograma?



# Operaciones de Punto

- Operaciones de Punto
  - **Transformación Log**

$$g(x) = c * \log(1 + f(x))$$

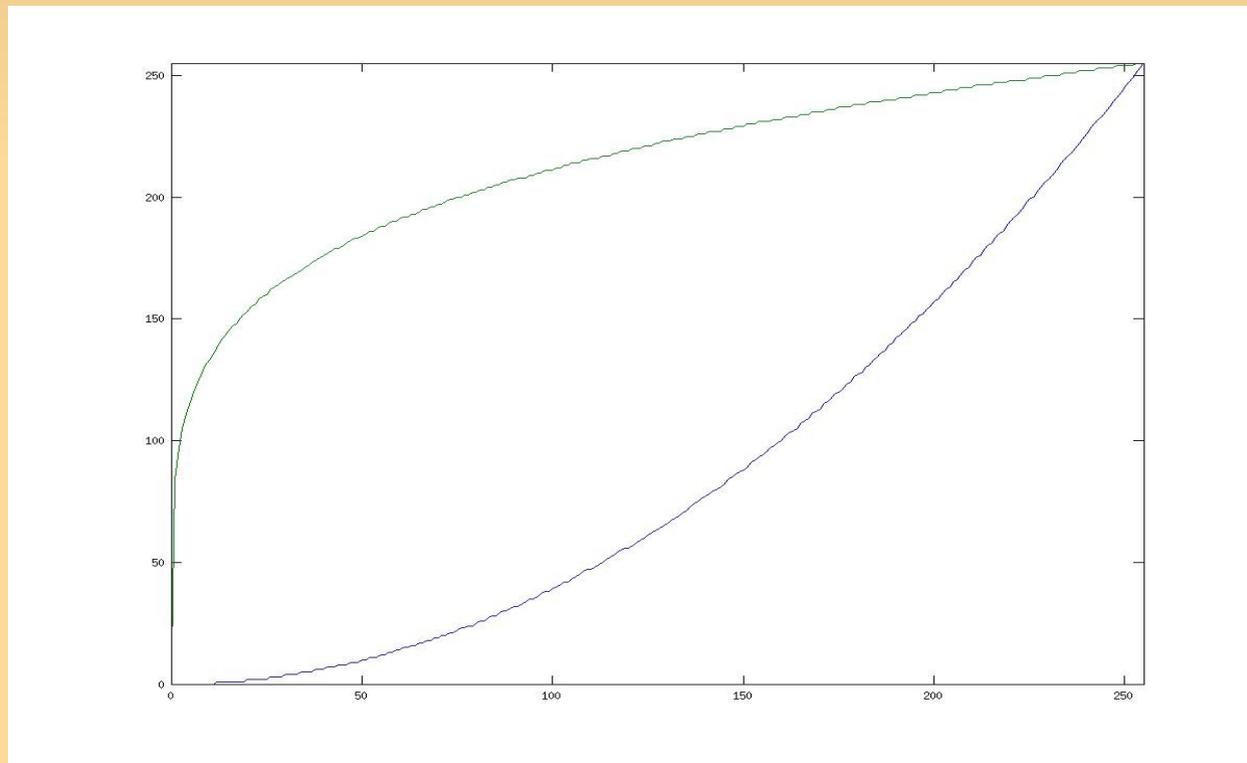


# Operaciones de Punto

- Operaciones de Punto

- Transformación Pow**

$g(x) = c * f(x)^z$  (Ejemplo  $c=1, z=0.2, 2$ )

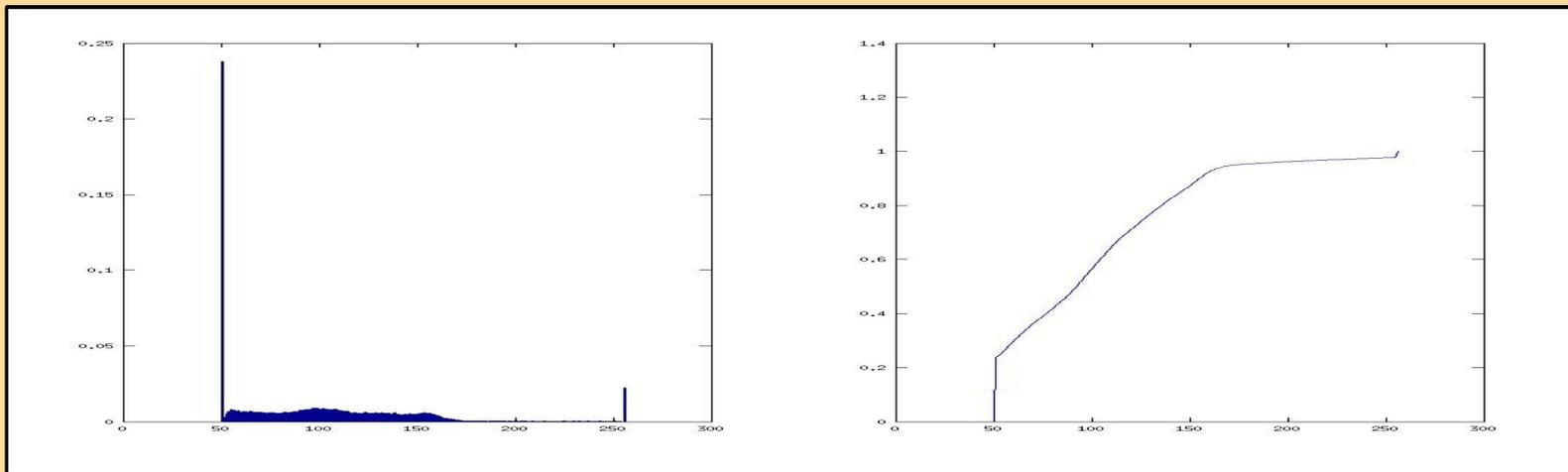


# Operaciones de Punto

- Operaciones de Punto
  - Ecualización del histograma**

$$g(x) = 255 * \text{cum}(f(x))$$

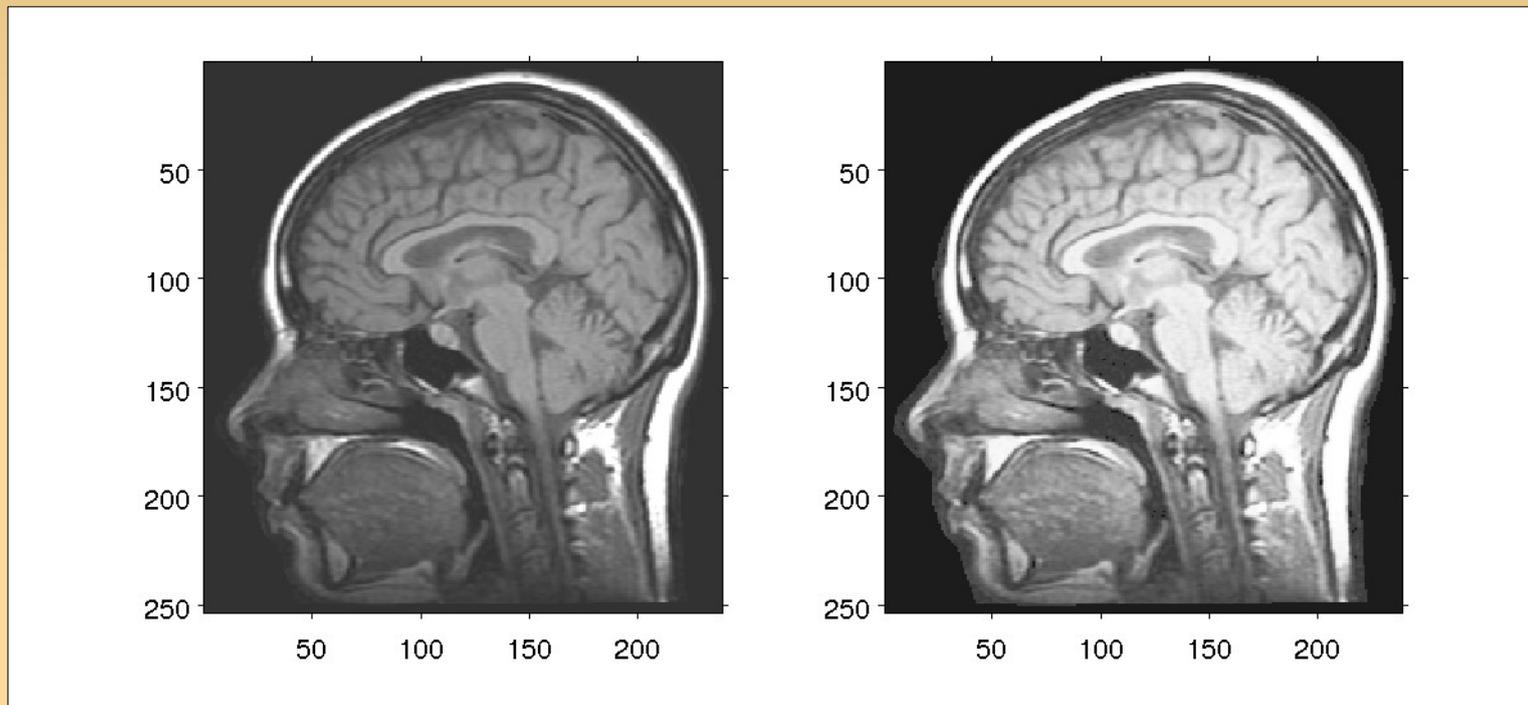
- $\text{cum}(f(x)) = h_n(0) + h_n(1) + \dots + h_n(f(x))$
- $h_n(i) = h(i) / \text{sum}(h)$



Histograma normalizado

Acumulación

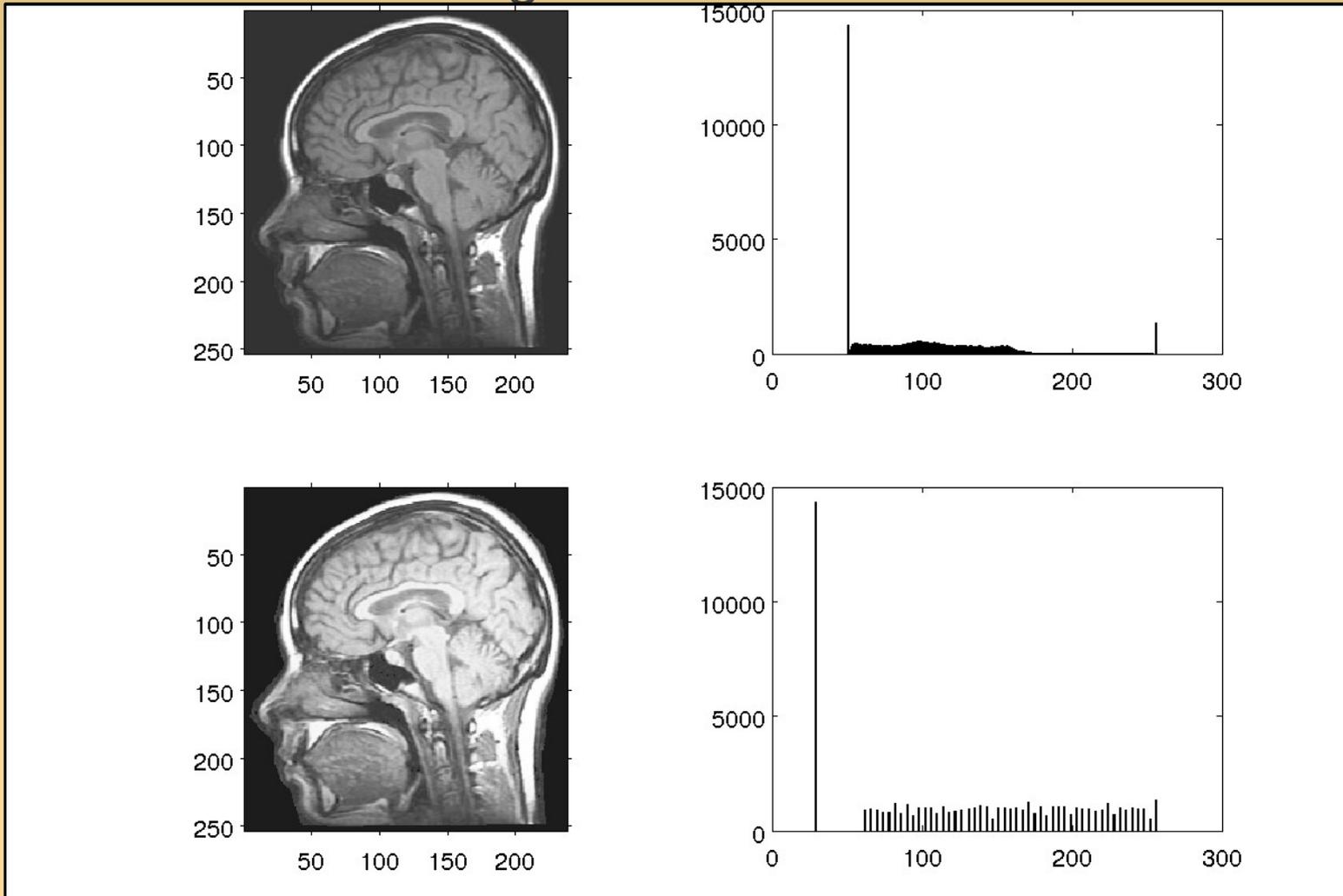
# Operaciones de Punto



Ecuación del  
histograma

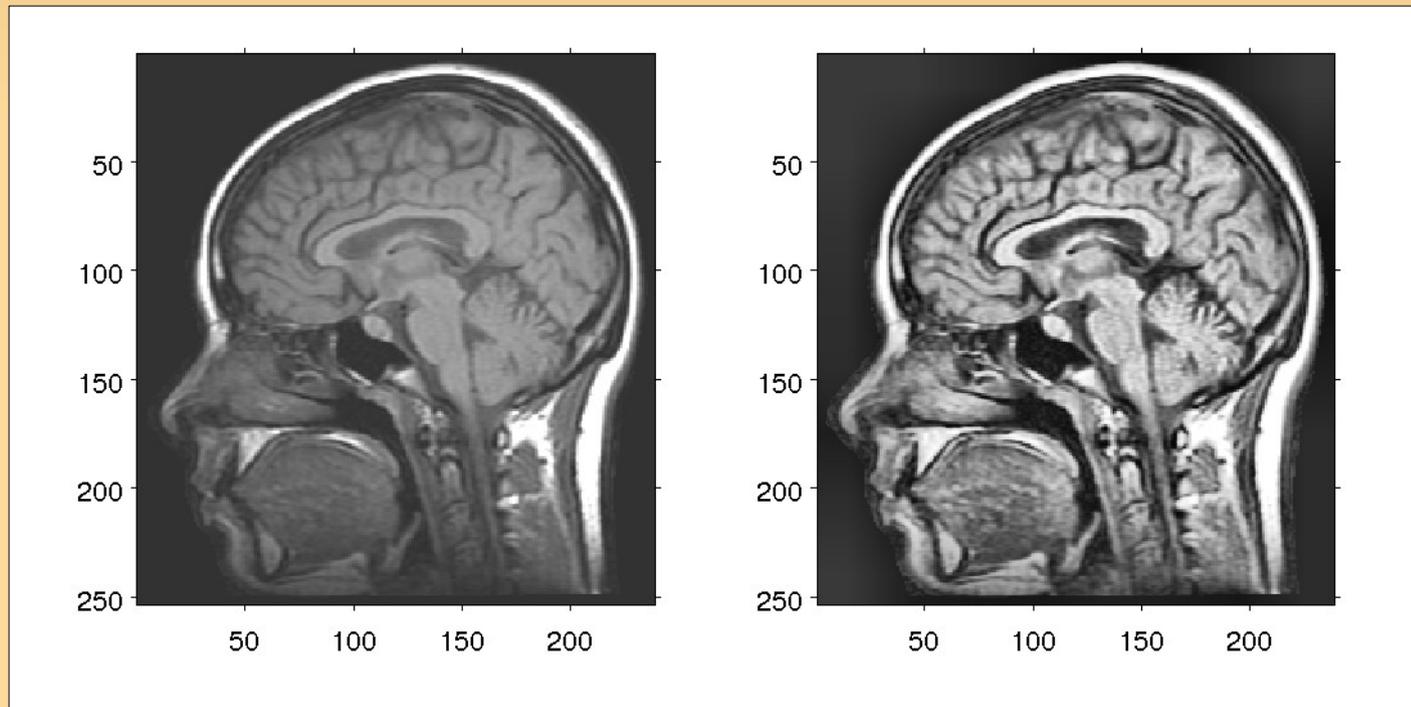
# Operaciones de Punto

- Cómo se ven los histograma?



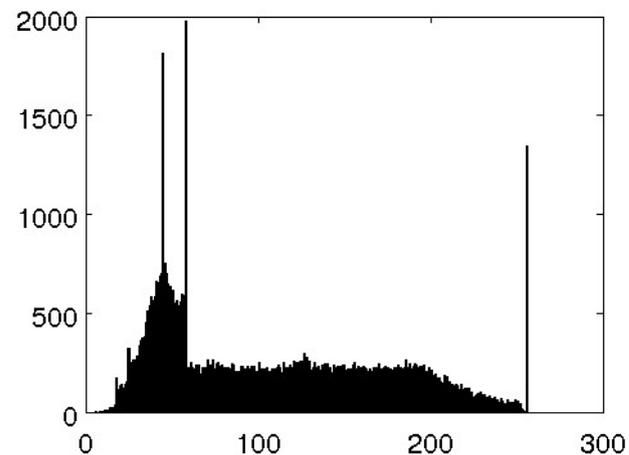
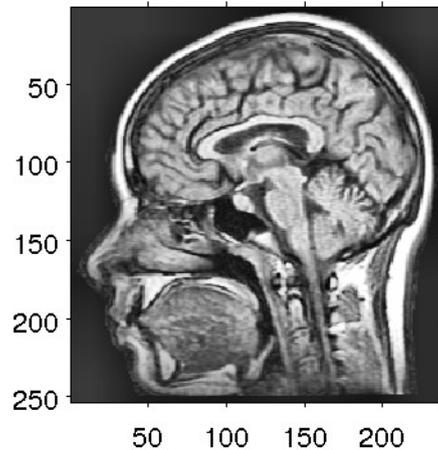
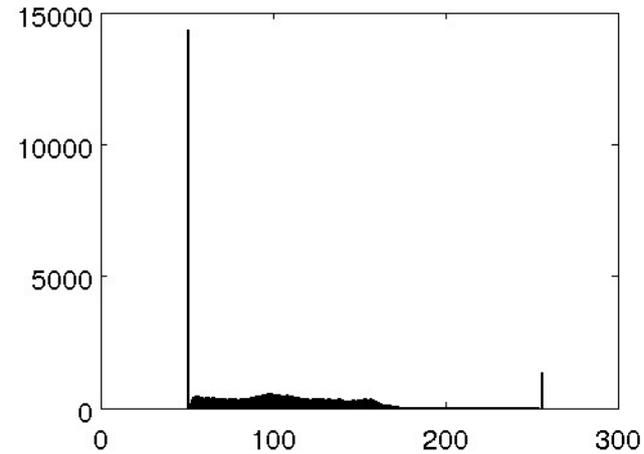
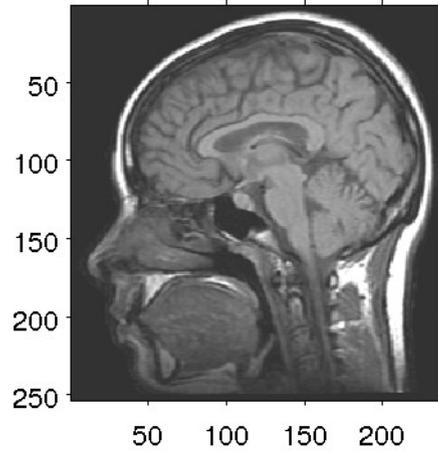
# Operaciones de Punto

- Operaciones de Punto
  - **Ecualización del histograma Adaptativo**  
Ecualiza el histograma por regiones



# Operaciones de Punto

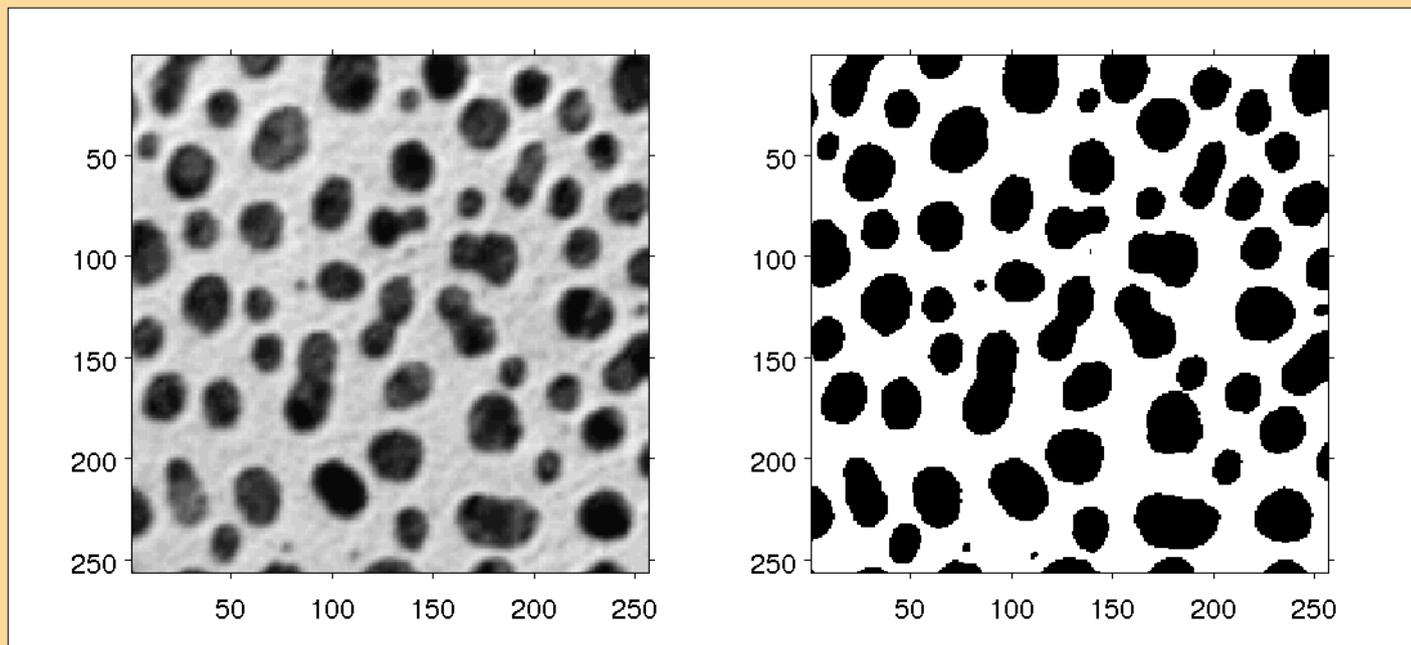
- Cómo se ven los histograma?



# Operaciones de Punto

- Operaciones de Punto
  - Binarización:** Separa objetos del fondo, los objetos de interés toman el valor 1, mientras que el fondo 0 (o viceversa).

$g(x) = \begin{cases} 1 & \text{si } (f(x) < u) \\ 0 & \text{en otro caso} \end{cases}$ ;



# Operaciones Locales

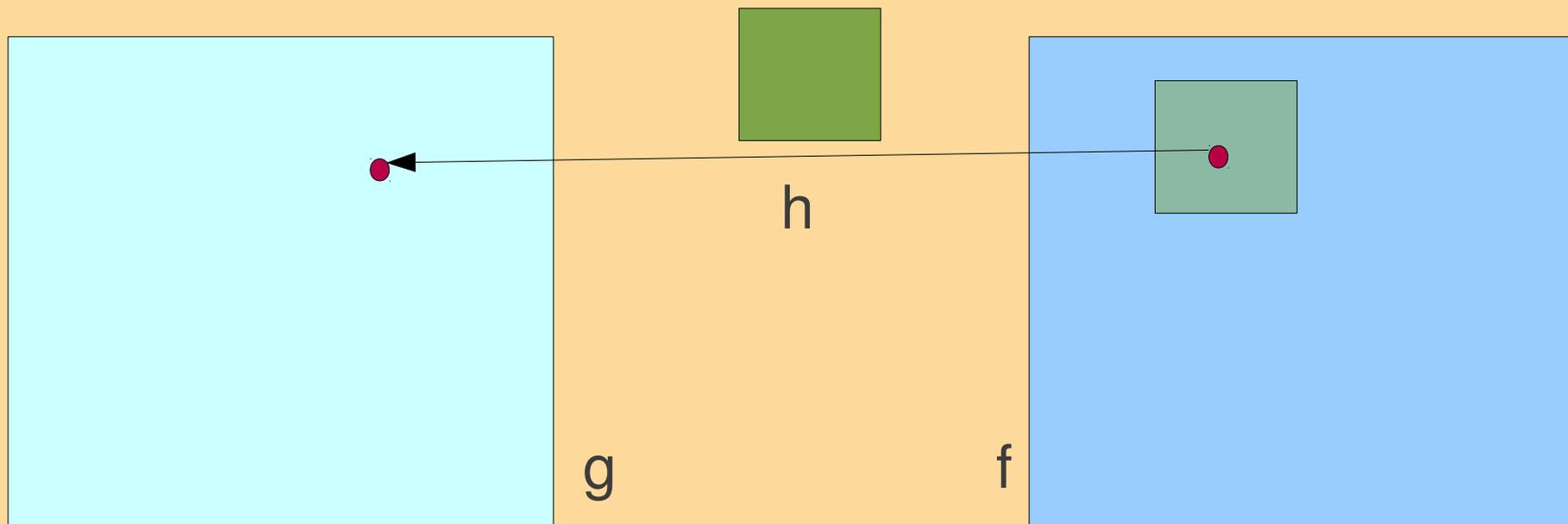
# Operaciones Locales

- Filtrado de Imágenes:
  - Calcula una función sobre una vecindad respecto a un pixel en la imagen.
    - La función se especifica por un filtro o 'máscara' que nos dice cómo combinar los valores de la vecindad.
- Usos del Filtro
  - Eliminación del ruido (smoothing)
  - Detección de bordes o fronteras.
  - Afinamiento de la imagen (sharpenig).

# Operaciones Locales

- Filtros Lineales: Convolución

$$g(i, j) = \sum_{k, l=-r}^r f(i-k, j-l)h(k, l)$$



# Operaciones Locales

- Eliminación de Ruido
  - **Ruido sal-pimienta:**  
Ocurriencia aleatoria de pixels blancos y negros.
  - **Ruido de Impulso:**  
Ocurriencia aleatoria de pixels blancos.
  - **Ruido Gaussiano:**  
Variación de intensidad dada por una distribución gaussiana.



Original



Salt and pepper noise



Impulse noise



Gaussian noise

# Operaciones Locales

- Eliminación de Ruido
  - Filtros Lineales
    - Box Filter: Promedio, todos los pixels se poderan de la misma manera (Filtro promedio).
    - Gaussiano: Filtro que sigue una ponderación gaussiana.
  - Filtro No Lineal
    - Mediana: Preserva bordes. Útil para eliminar ruido del tipo sal-pimienta.
  - Al definir el filtro se debe definir el tamaño (la vecindad).
  - Los valores del filtro deben sumar 1.

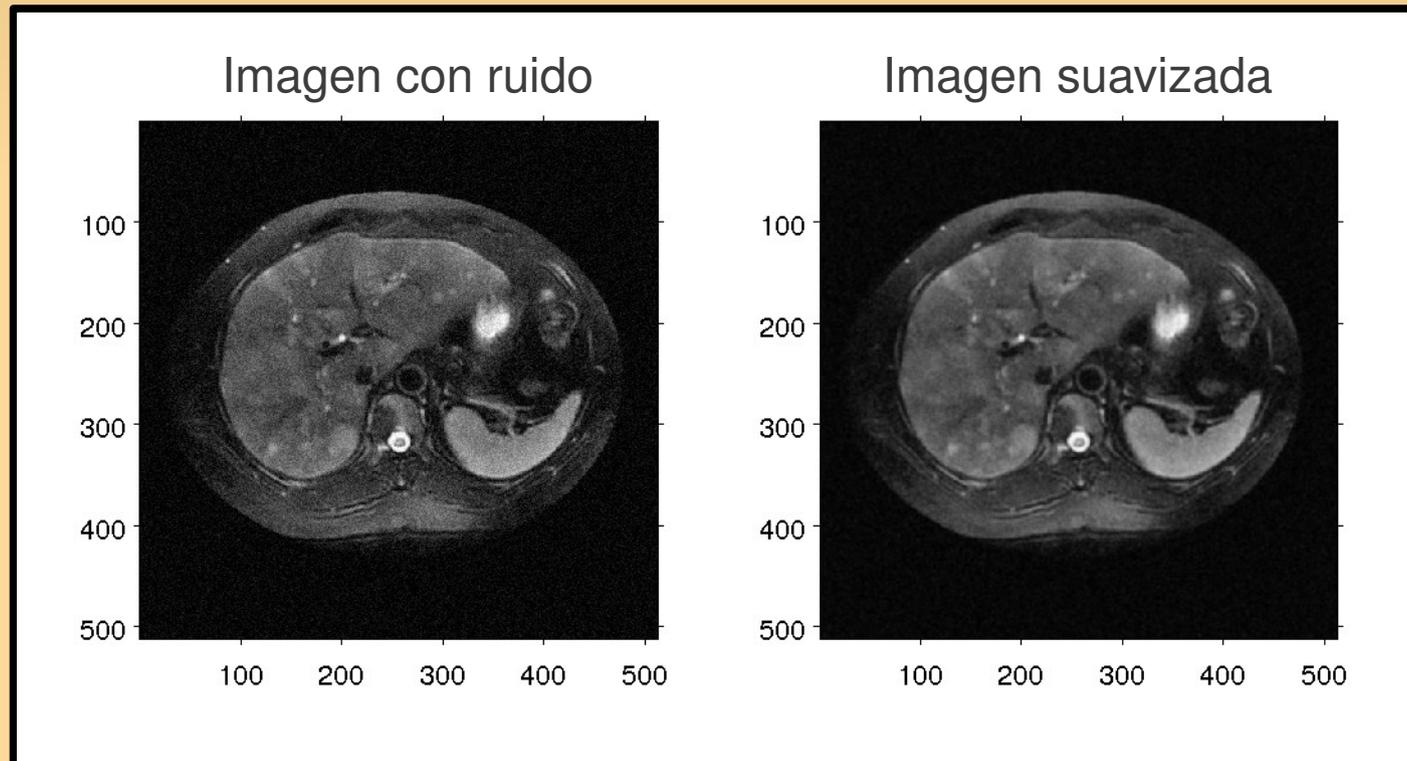
# Operaciones Locales

- Eliminación de ruido
  - Box Filter: El nuevo valor de un pixel es igual al promedio de su vecindad.
  - Ejemplo: Box Filter de 3x3 (<--Tamaño de la vecindad)

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

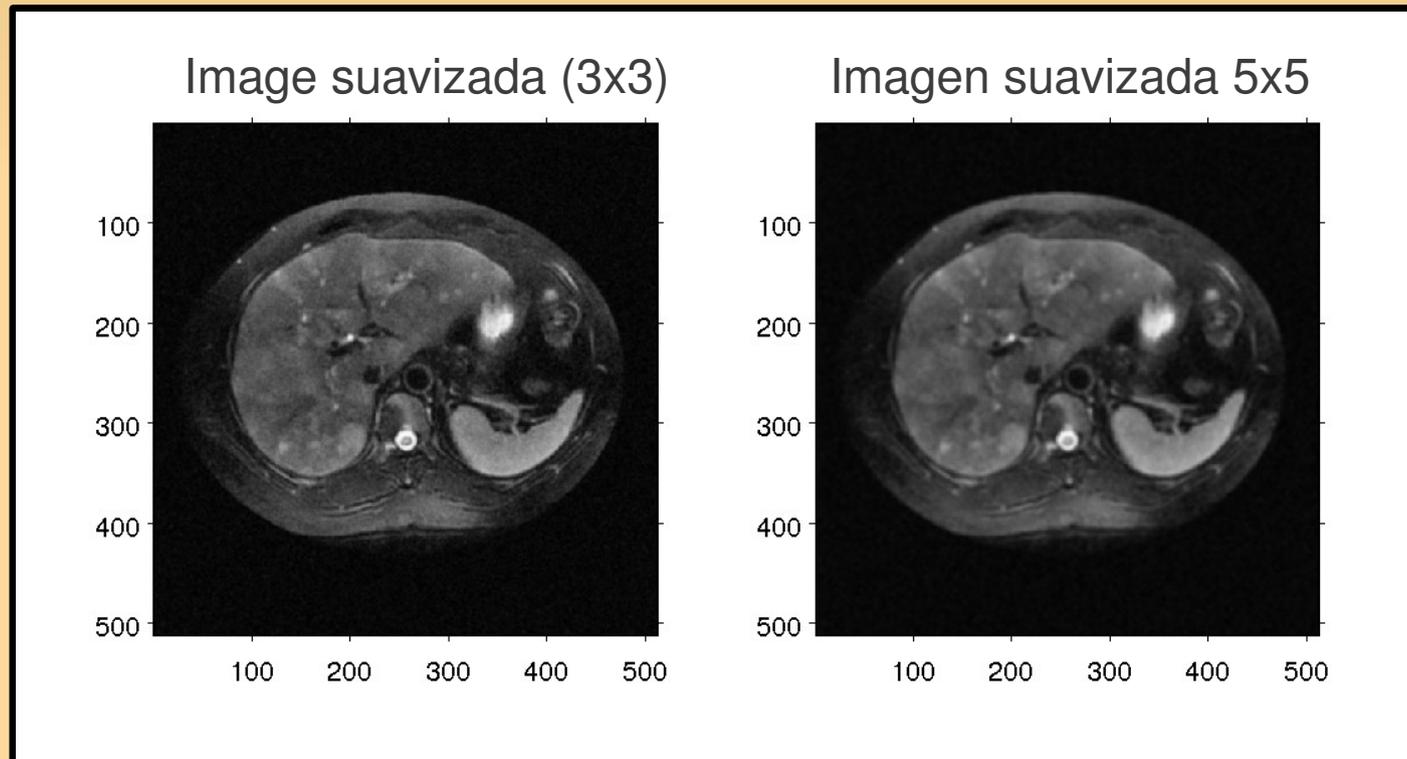
# Operaciones Locales

- Eliminación de ruido
  - Box Filter - Ejemplo



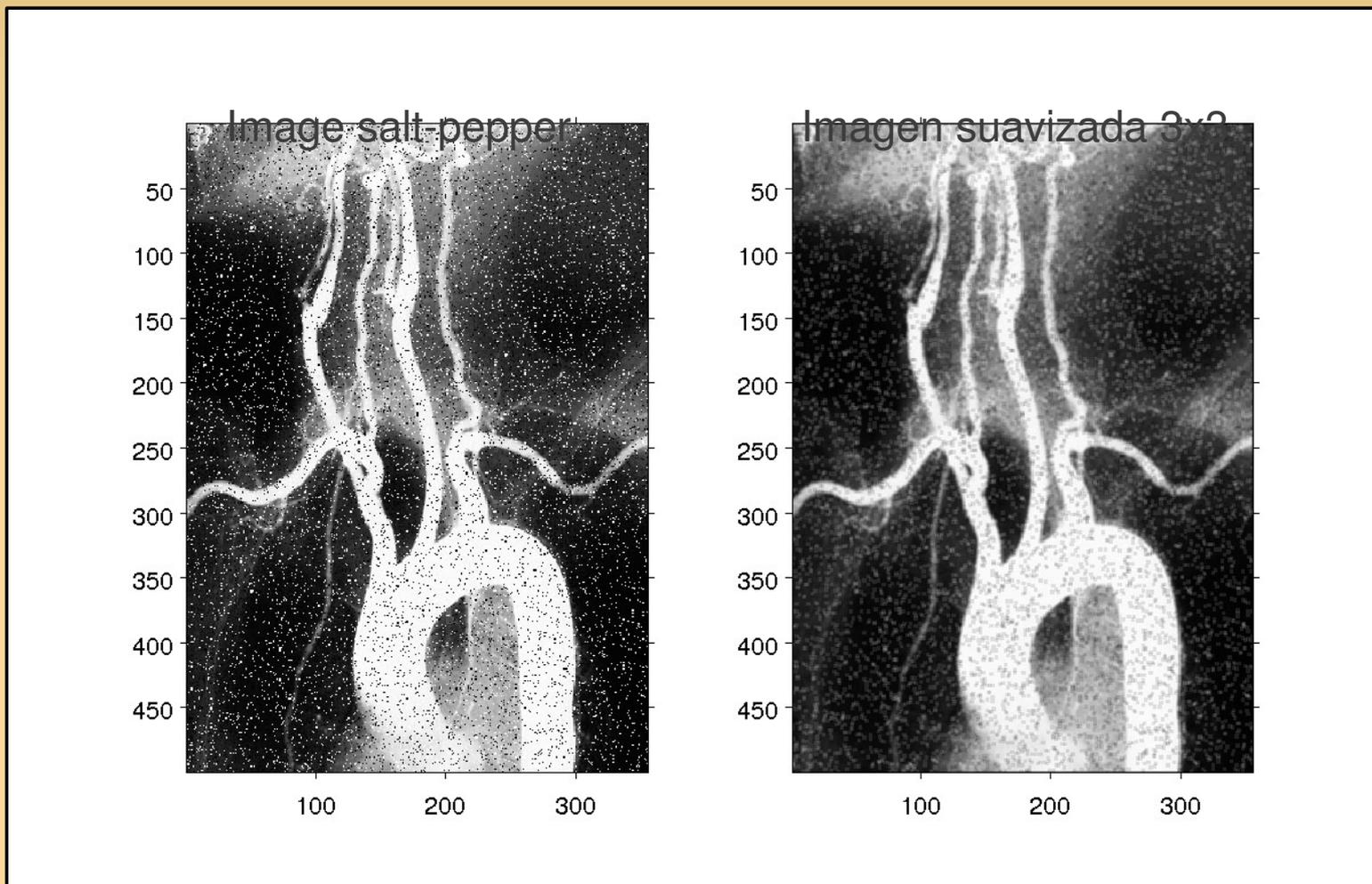
# Operaciones Locales

- Eliminación de ruido
  - Box Filter - Ejemplo



# Operaciones Locales

- Eliminación de ruido
  - Box Filter - Ejemplo



# Operaciones Locales

- Eliminación de ruido
  - Gaussiano: El nuevo valor de un pixel es igual al promedio de su vecindad. Aquí los pesos de la vecindad están dados por:

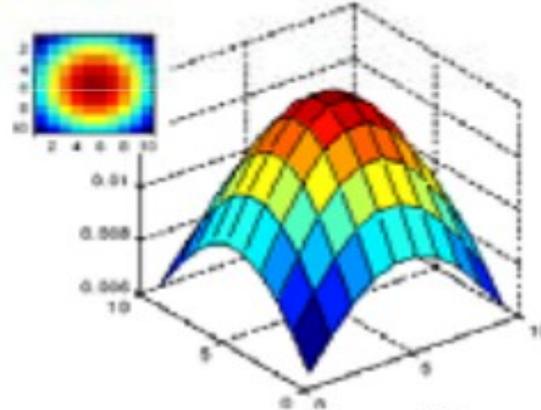
$$g(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

- Ejemplo: Gaussiano de 3x3, sigma=0.5

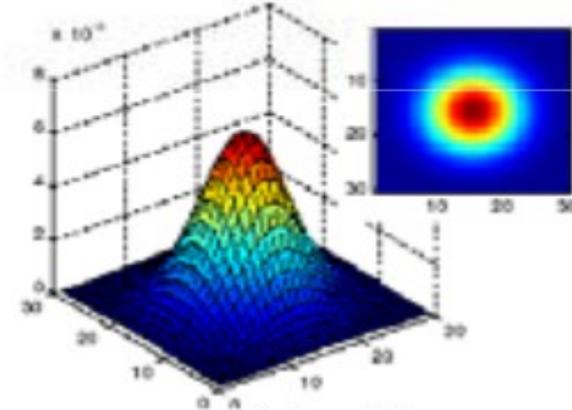
0.0113	0.0838	0.0113
0.0838	0.6193	0.0838
0.0113	0.0838	0.0113

# Operaciones Locales

- Filtro Gaussiano



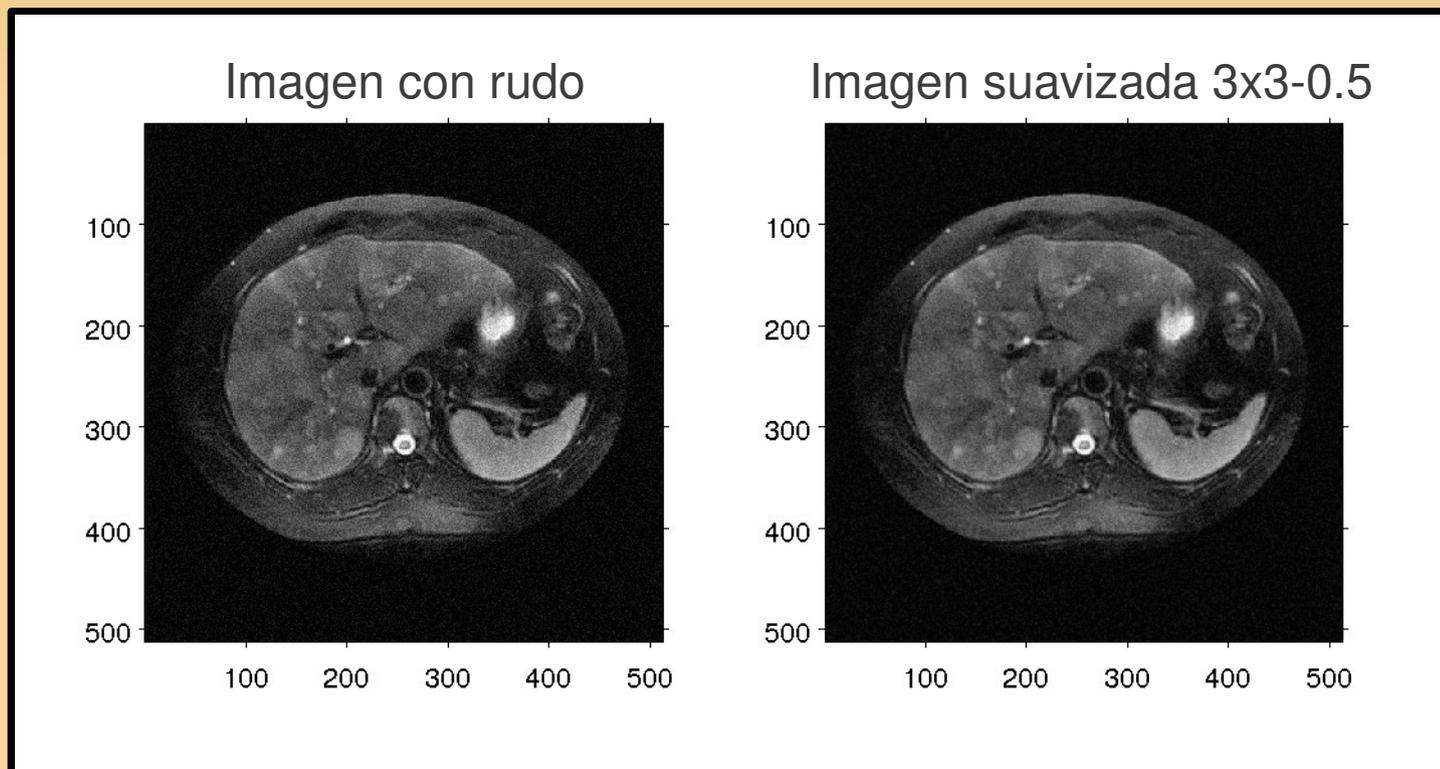
$\sigma = 5$  with  
10 x 10  
kernel



$\sigma = 5$  with  
30 x 30  
kernel

# Operaciones Locales

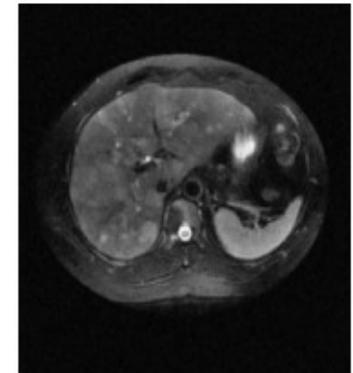
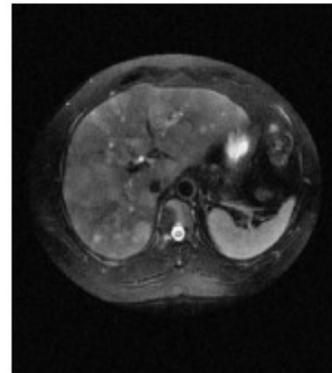
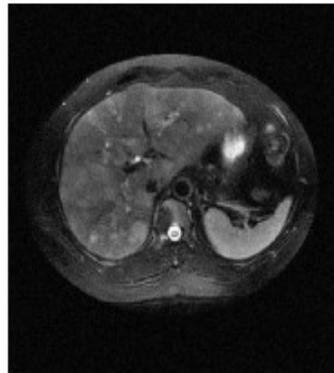
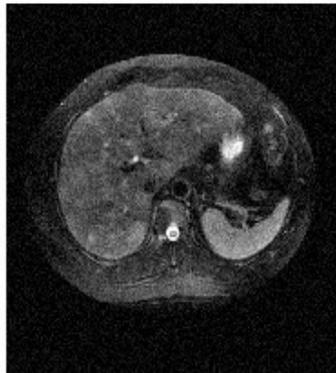
- Eliminación de ruido
  - Gaussiano - Ejemplo



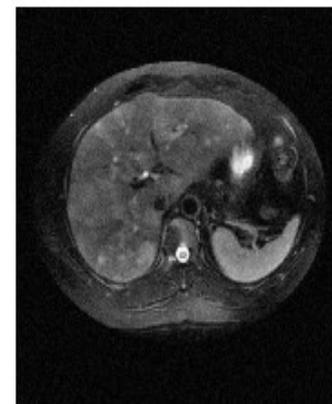
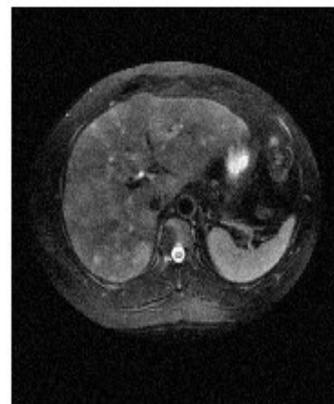
# Operaciones Locales

- Eliminación de Ruido: Box vs Gaussiano

box



gauss



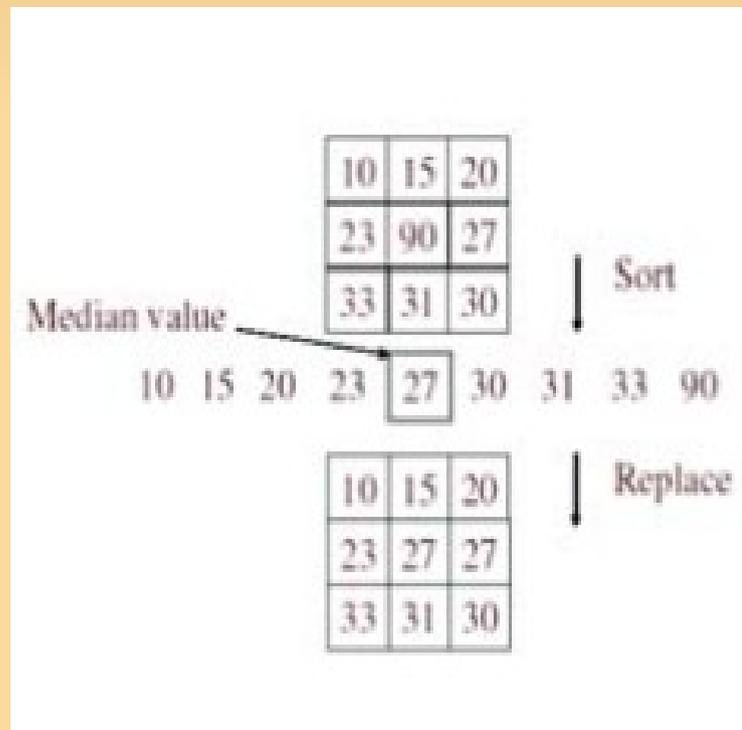
# Operaciones Locales

- Eliminación de ruido

Qué sucede con la imagen con ruido salt-pepper?

# Operaciones Locales

- Eliminación de ruido
  - Filtro no lineal de la mediana: El nuevo pixel es la mediana de la vecindad.



# Operaciones Locales

- Eliminación de ruido
  - Mediana - Ejemplo

Image salt-pepper

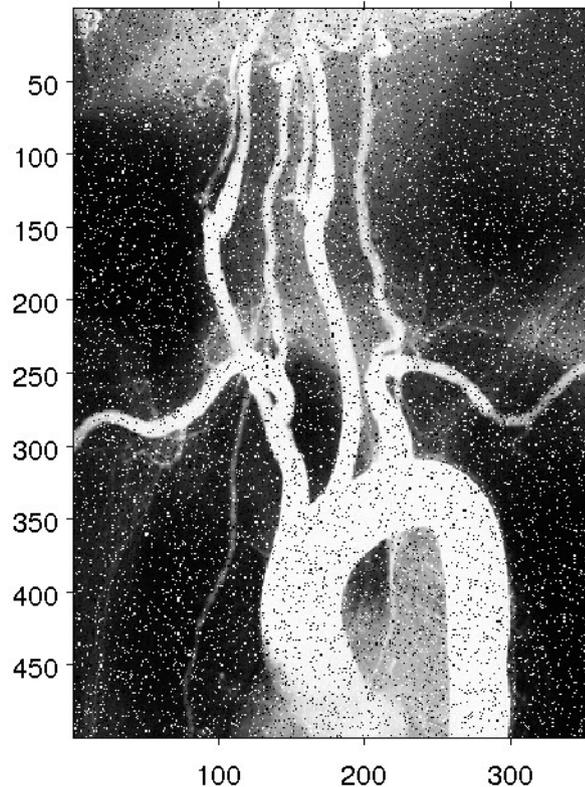
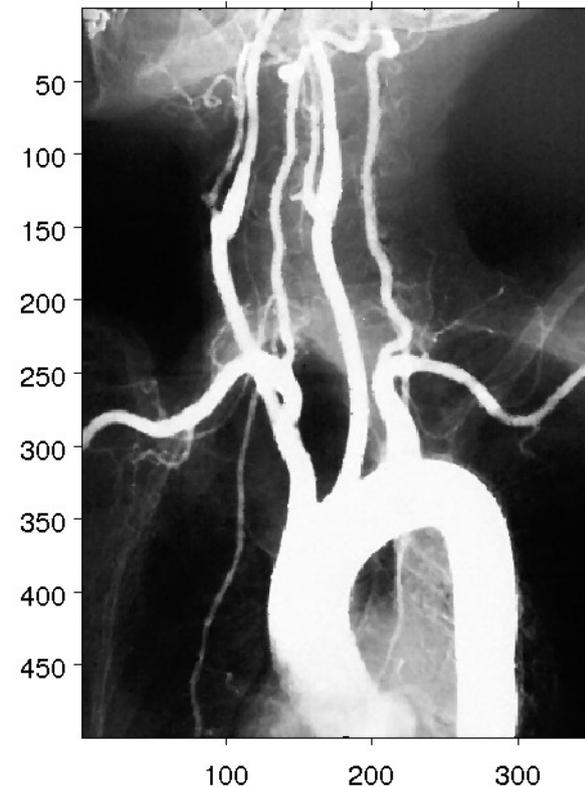


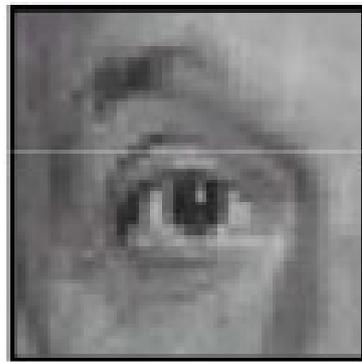
Imagen suavizada 3x3



# Operaciones Locales

- Resultados de las máscaras

Qué ocurre en este caso?



**Original**

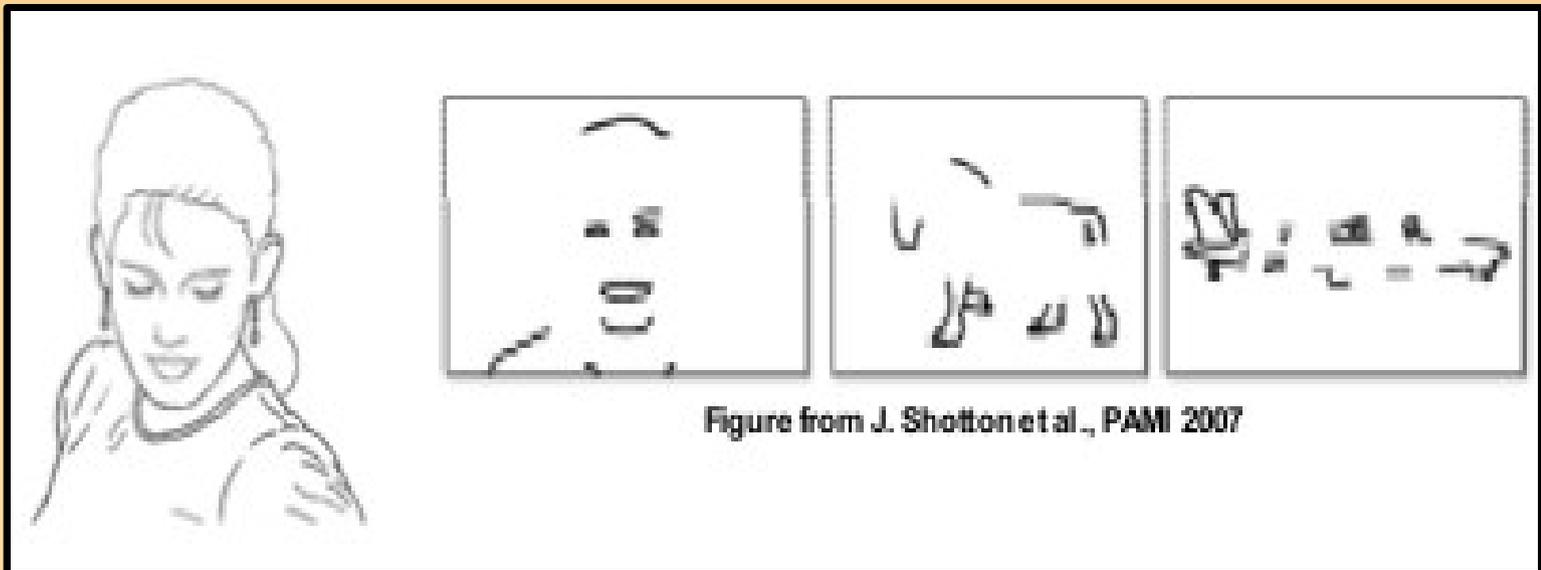
0	0	0
0	1	0
0	0	0

?

**BREAK (15min)**

# Operaciones Locales

- **Detección de Bordes**
  - Objetivo: Asociar una imagen con un conjunto de curvas o segmentos de línea o contornos.

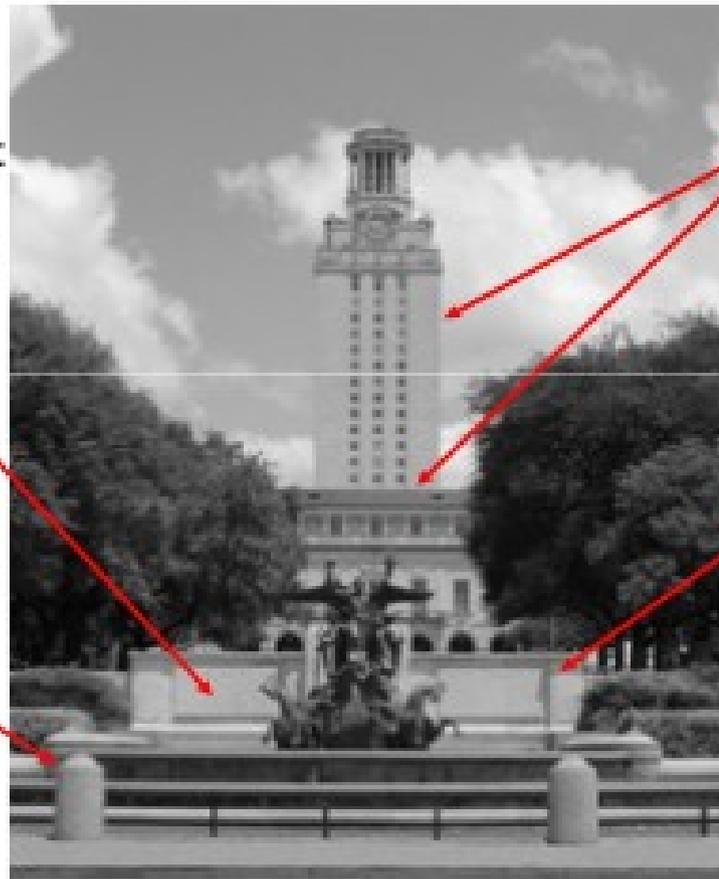


# Operaciones Locales

- Detección de Bordes
  - Qué genera los bordes?

Reflectance change:  
appearance  
information, texture

Change in surface  
orientation: shape

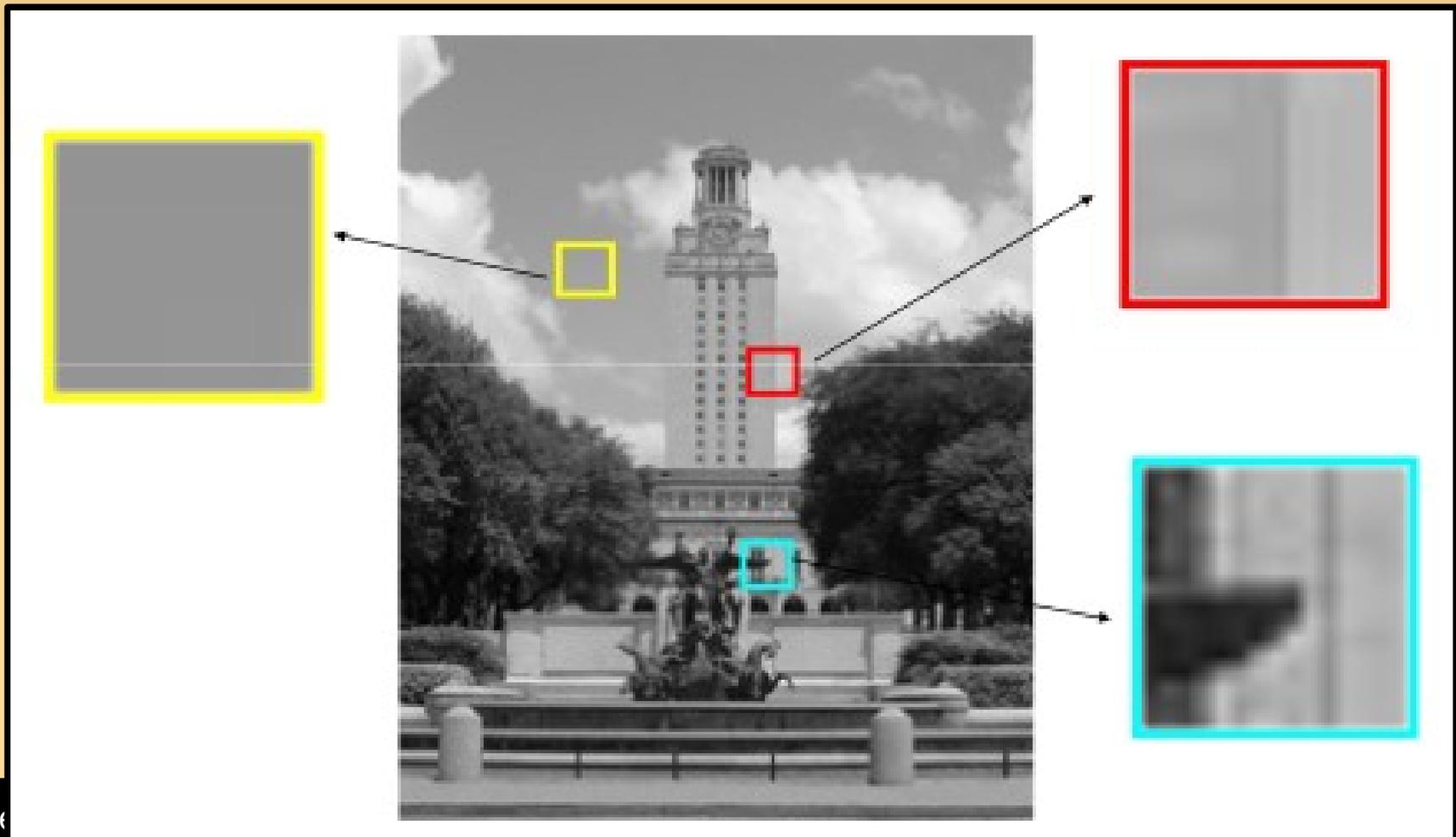


Depth discontinuity:  
object boundary

Cast shadows

# Operaciones Locales

- Detección de Bordes
  - Qué genera los bordes?



# Operaciones Locales

- Detección de Bordes
  - Gradiente: Mide la variación de intensidad local.

$$\nabla f = \left[ \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right]$$
$$\text{mag}(\nabla f) = \sqrt{\left( \frac{\partial f}{\partial x} \right)^2 + \left( \frac{\partial f}{\partial y} \right)^2}$$

# Operaciones Locales

- Detección de Bordos
  - Gradiente: Mide la variación de intensidad local.

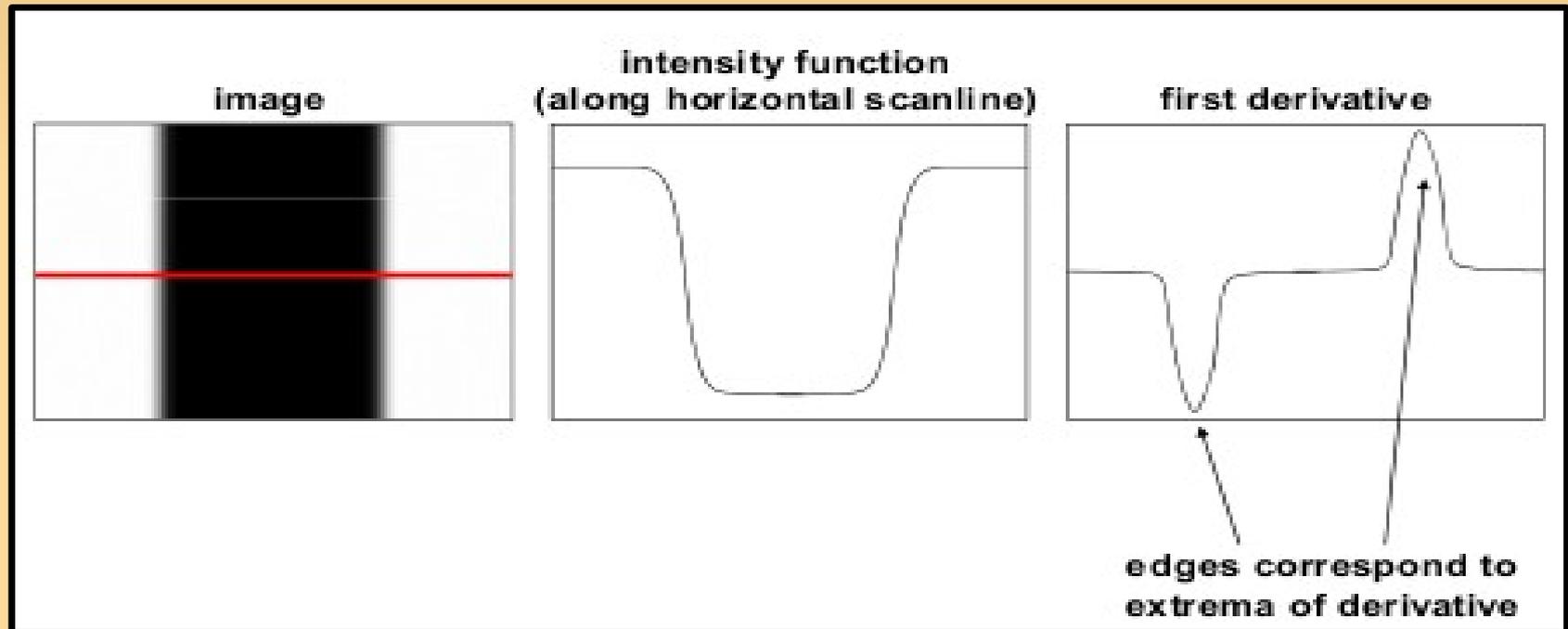
$$\frac{\partial f(x, y)}{\partial x} = \lim_{\varepsilon \rightarrow 0} \frac{f(x + \varepsilon, y) - f(x, y)}{\varepsilon}$$

En forma discreta:

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

# Operaciones Locales

- Detección de Bordes
  - Derivadas y Bordes



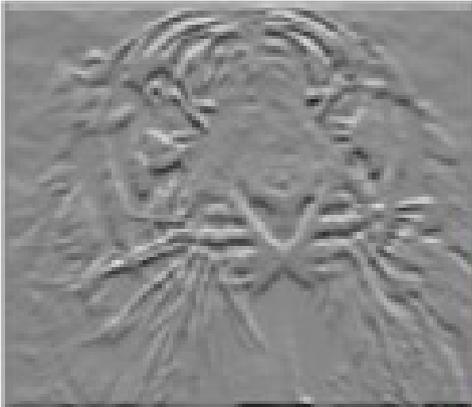
# Operaciones Locales

- Detección de Bordes
  - Derivadas y Bordes

$\frac{\partial f(x, y)}{\partial x}$



$\frac{\partial f(x, y)}{\partial y}$



$\begin{bmatrix} -1 & 1 \end{bmatrix}$

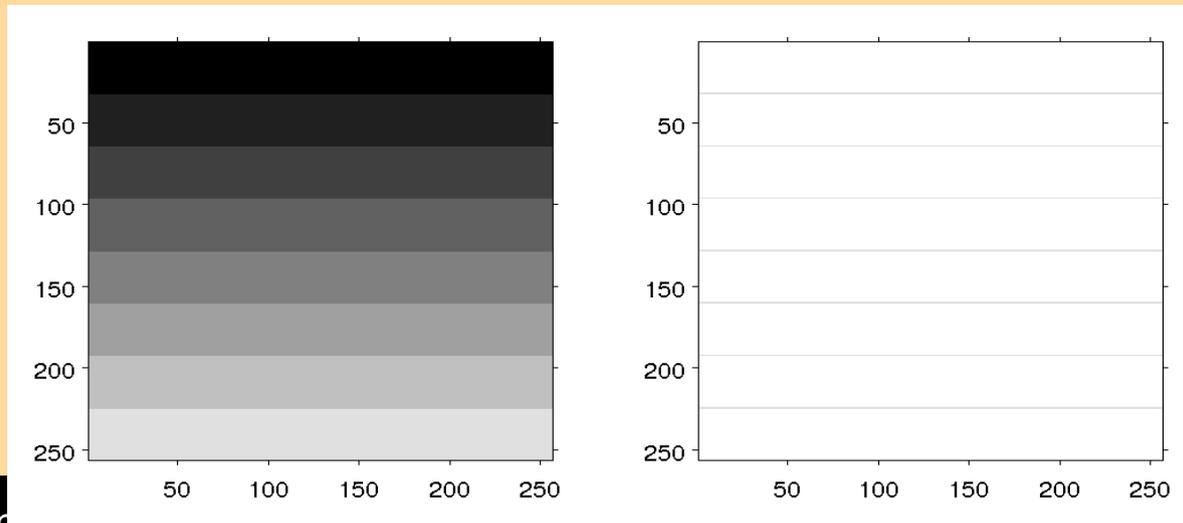
$\begin{bmatrix} -1 & ? & 1 \\ 1 & \text{or} & -1 \end{bmatrix}$

# Operaciones Locales

- Detección de Bordos
  - Aproximaciones
    - Simple

$G_x = [-1, 1]$  (gradiente en x)

$G_y = \begin{array}{|c|} \hline -1 \\ \hline 1 \\ \hline \end{array}$  (gradiente en y)

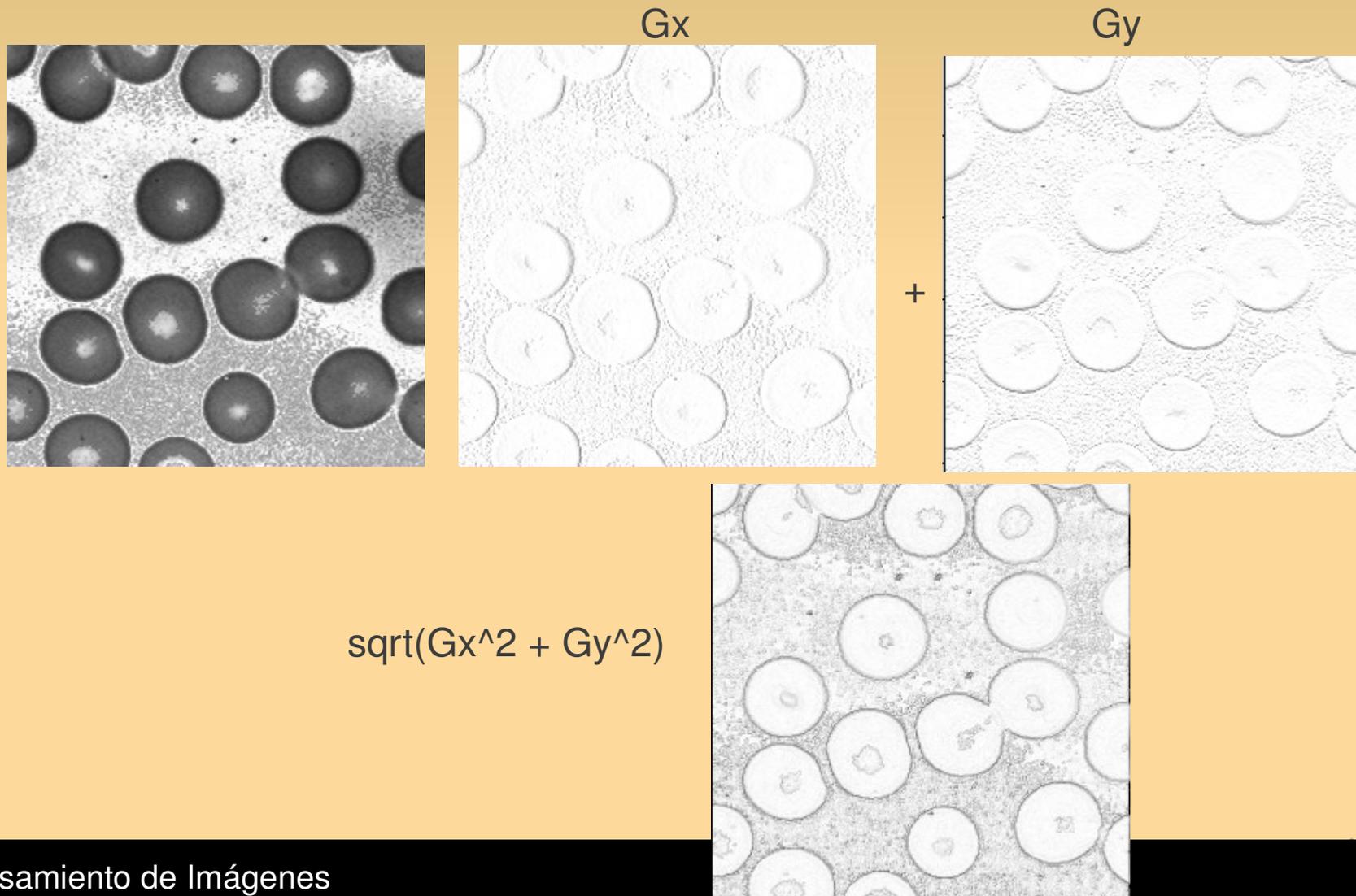


Grad\_y



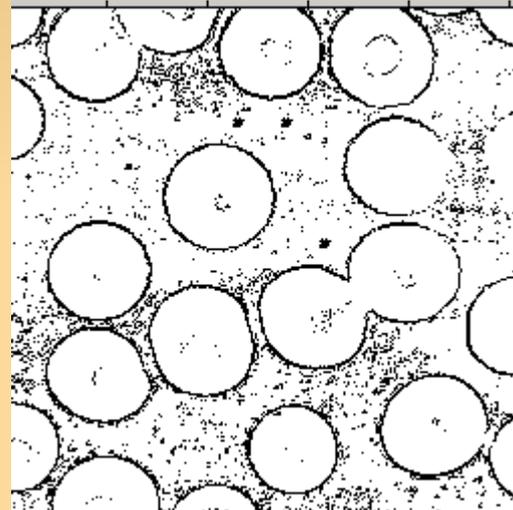
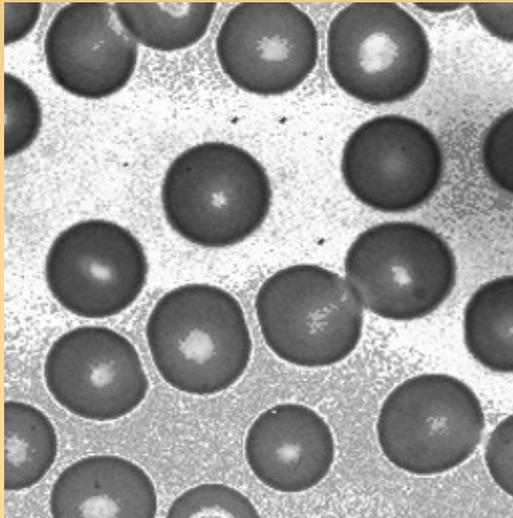
# Operaciones Locales

- Detección de Bordes



# Operaciones Locales

- Detección de Bordes



Binarización del resultado

# Operaciones Locales

## Detección de Bordos

**Prewitt:**  $M_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$  ;  $M_y = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$

**Sobel:**  $M_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$  ;  $M_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$

**Roberts:**  $M_x = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$  ;  $M_y = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$

# Operaciones Locales

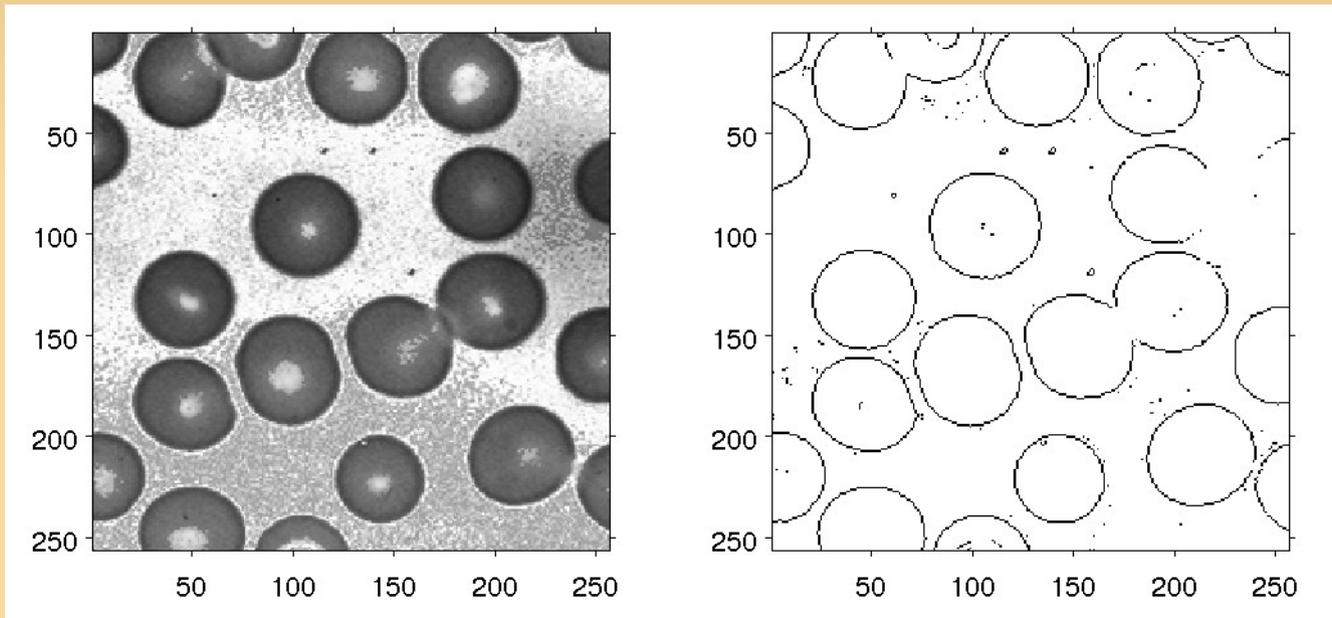
Ejemplo:

4	5	6	7
10	7	4	5
100	130	150	123
101	120	140	130

Cuál es el resultado de aplicar  $[-1 \ 1]$ ,  $[-1 \ 1]^T$  ?

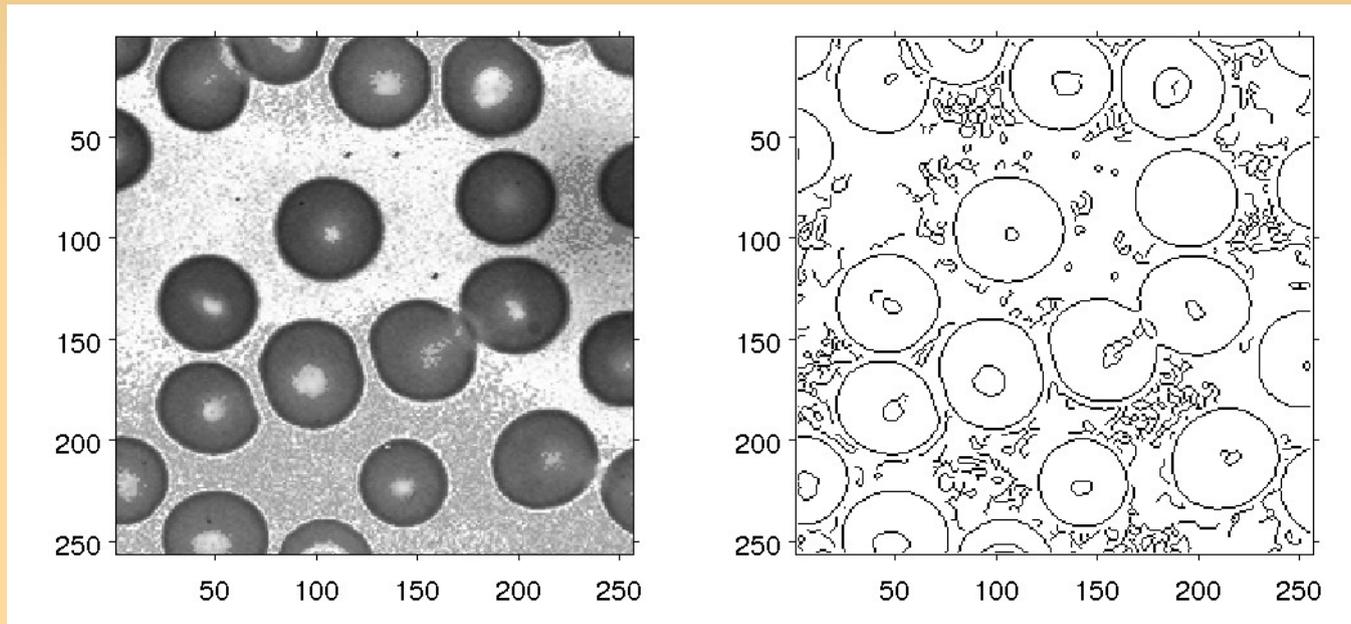
# Operaciones Locales

- Detección de Bordos: Ejemplo con sobel



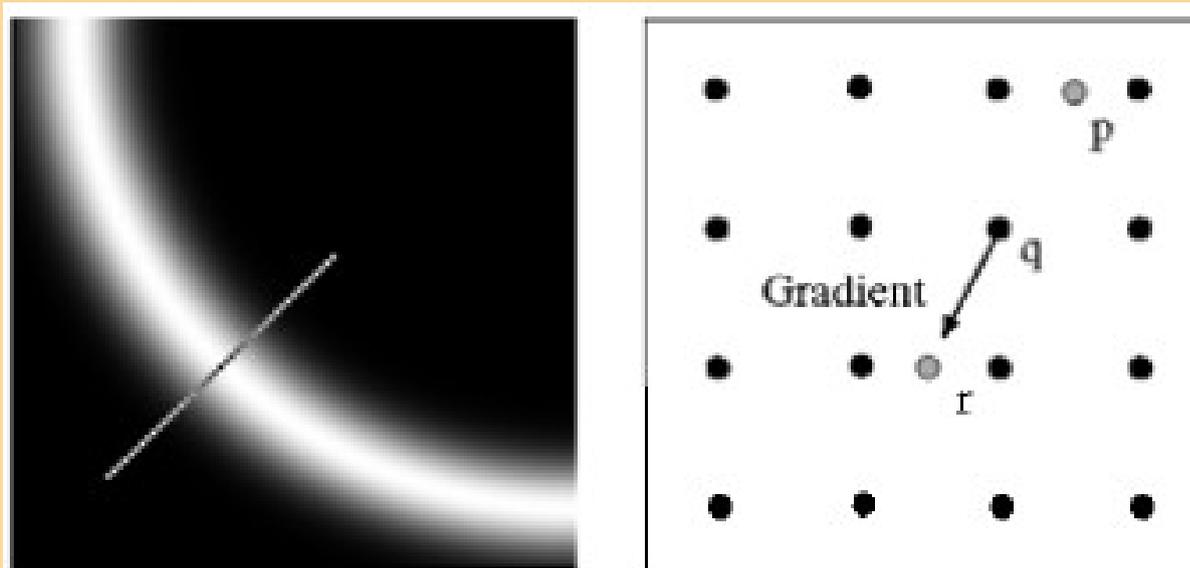
# Operaciones Locales

- Detección de Bordes: Canny

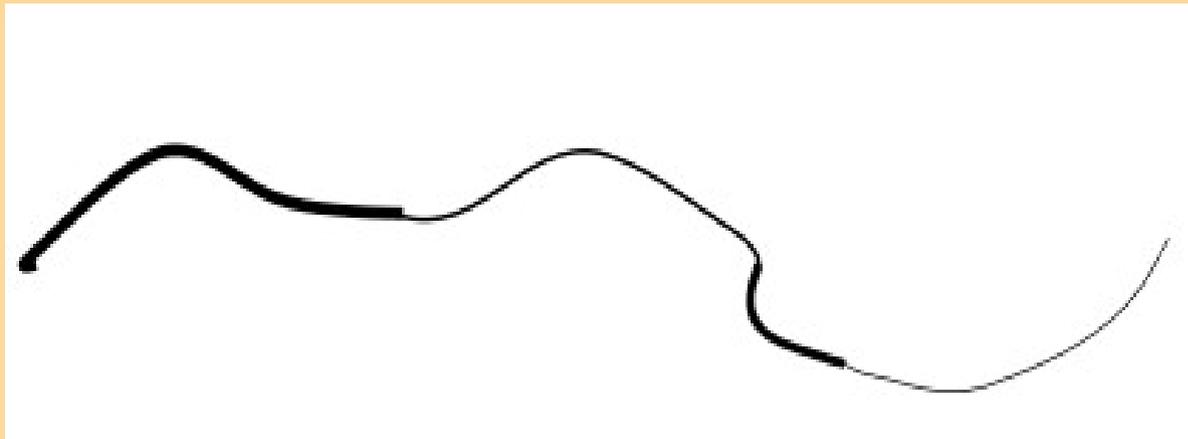


- Canny

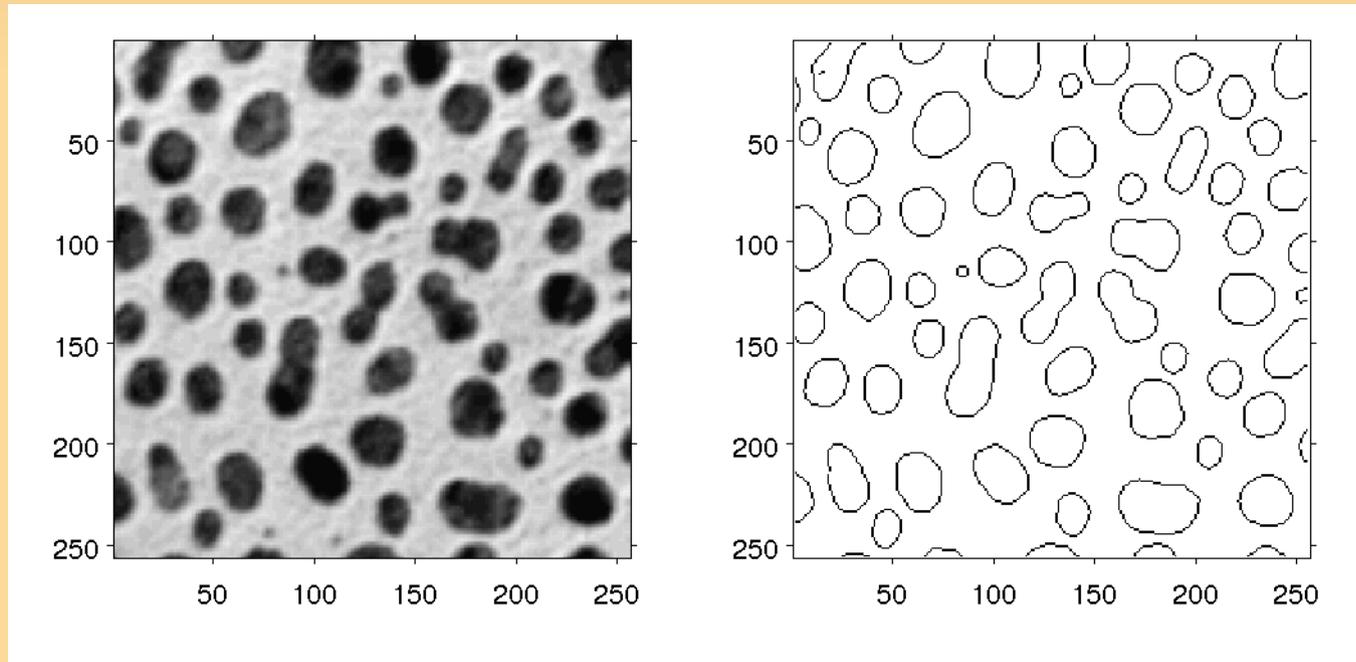
- Aplica supresión de los no-maximo en sentido del gradiente. Así permite encontrar bordes delgado.



- Canny
  - Une bordes débiles si están conectados a otros fuertes (histeresis)



- Canny
  - Otro ejemplo

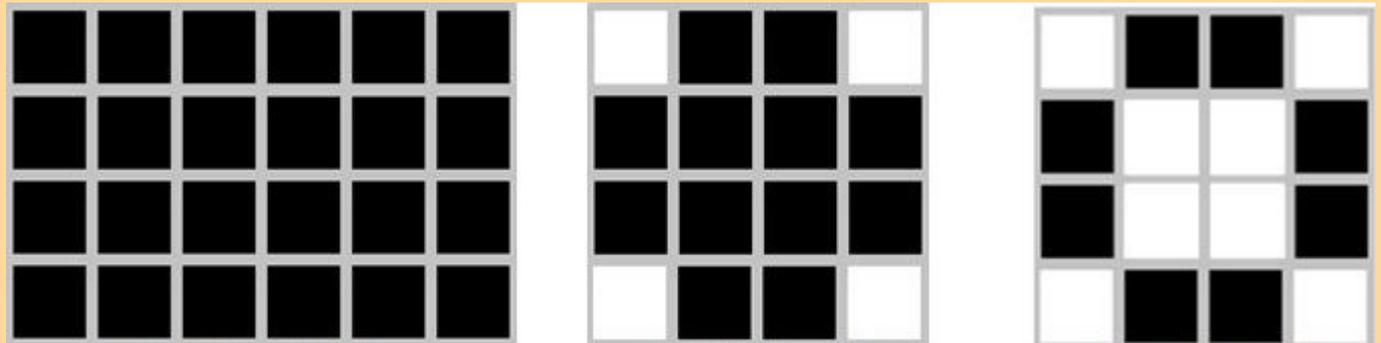


# Operaciones Morfológicas

- Sobre imágenes binarias.
- Se requiere de un elemento estructural que define la vecindad a procesar. Llamémoslo  $s$ .
- $c = f \otimes s$ , el número de 1's de  $f$  que caen dentro de  $s$ .
- $S = |s|$ , tamaño de  $s$ .
- Sea:

$$Q(x,t) = 1 \text{ si } x \geq t; 0 \text{ en otro caso;}$$

Ejemplo de elementos estructurales



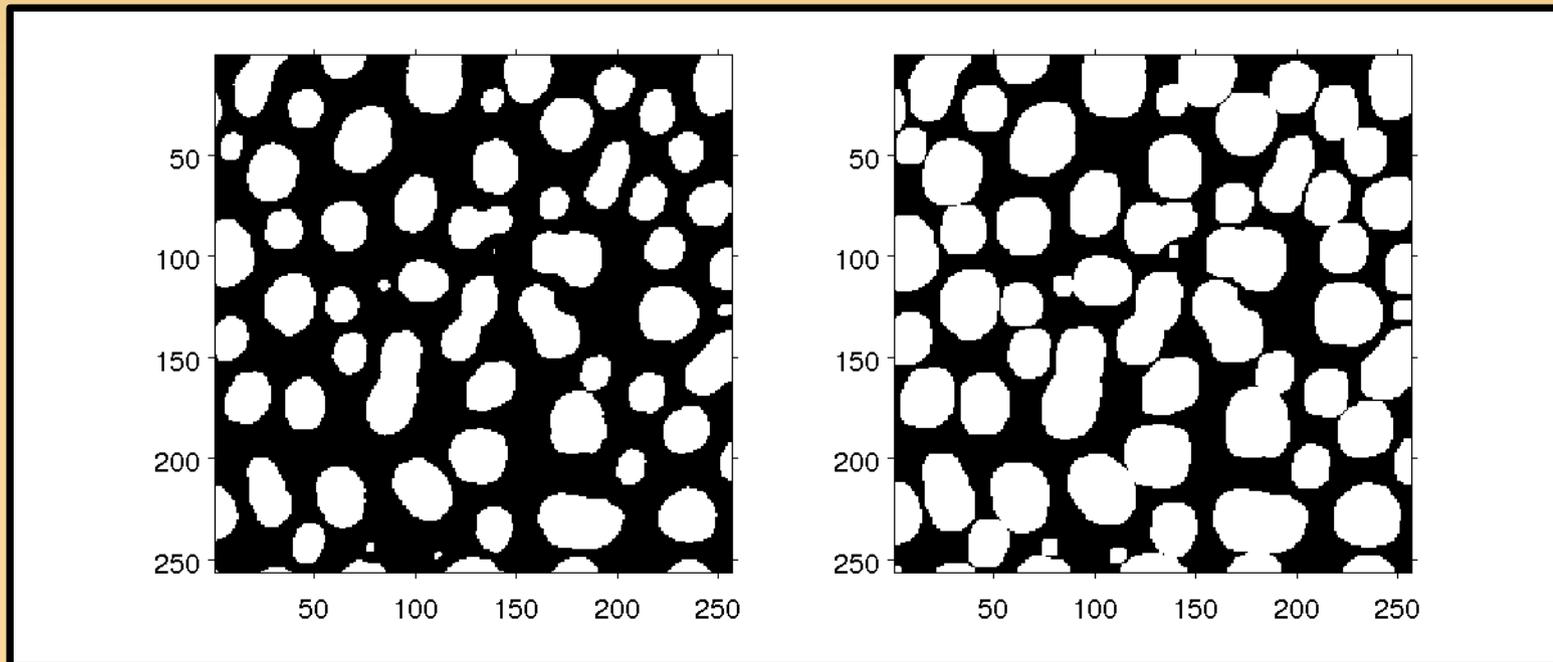
# Operaciones Morfológicas

- Mayoria:  $\text{maj}(f,s)=Q(c, S/2)$
- Dilatar:  $\text{dilate}(f,s)=Q(c,1)$
- Erosionar:  $\text{erode}(f,s)=Q(c,S)$
- Opening:  $\text{open}(f,s)=\text{dilate}(\text{erode}(f,s),s)$
- Closing:  $\text{close}(f,s)=\text{erode}(\text{dilate}(f,s),s)$

EJEMPLO

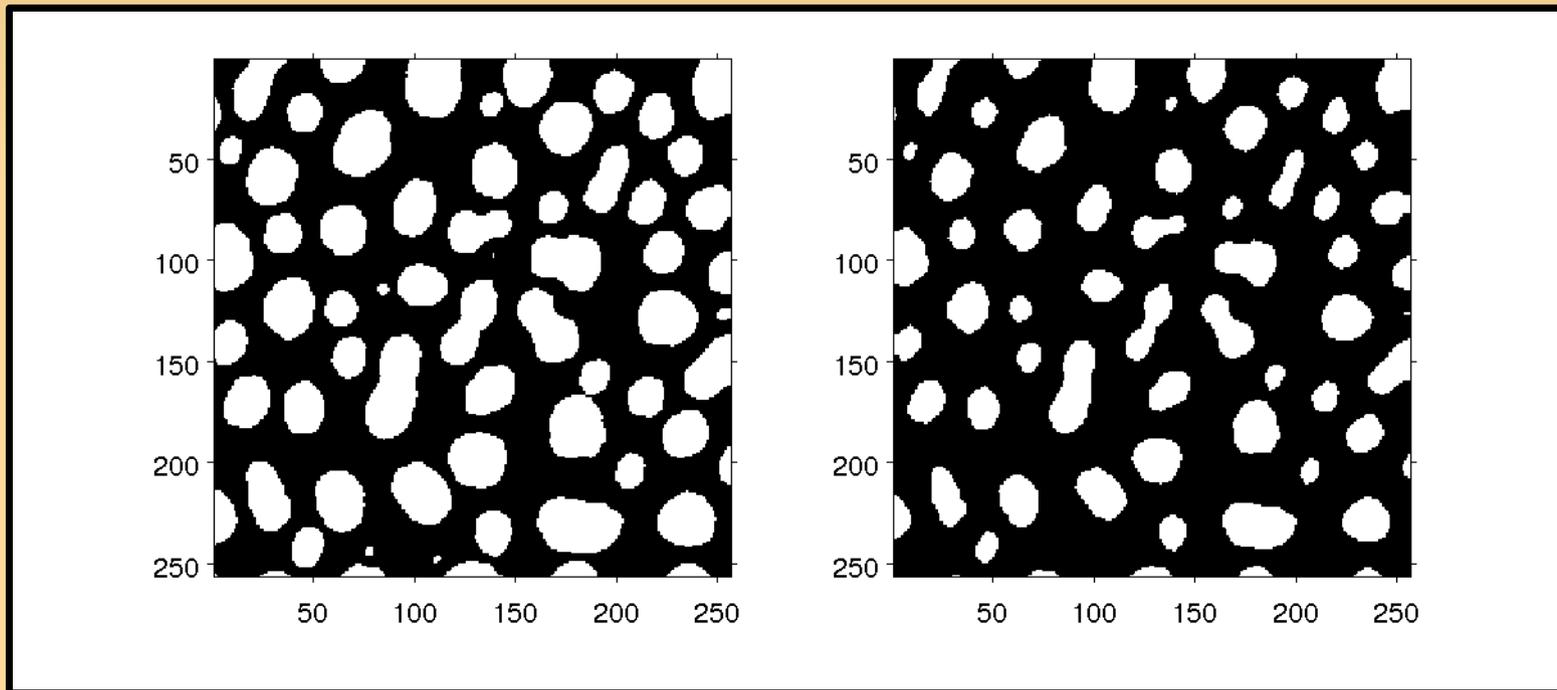
# Operaciones Morfológicas

- Ejemplos:
  - Dilatación



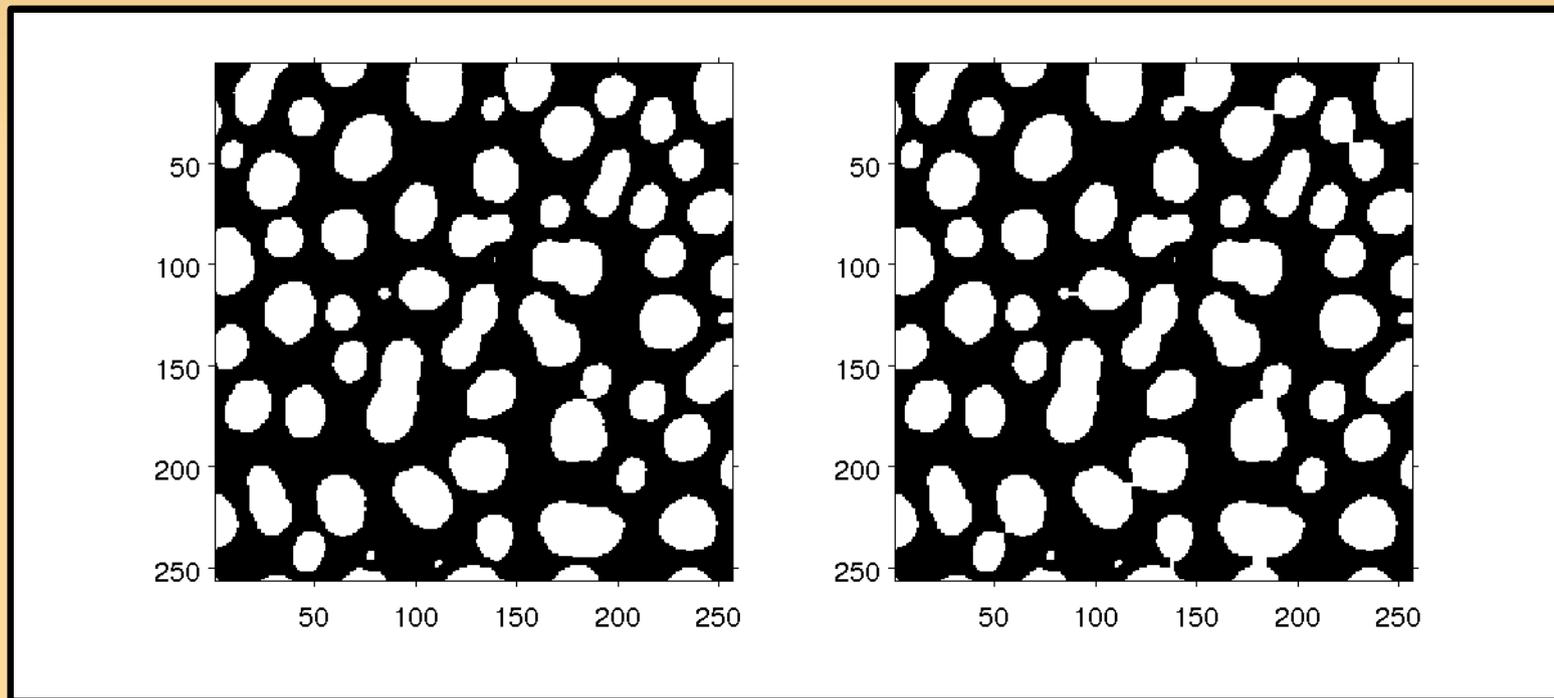
# Operaciones Morfológicas

- Ejemplos:
  - Erode



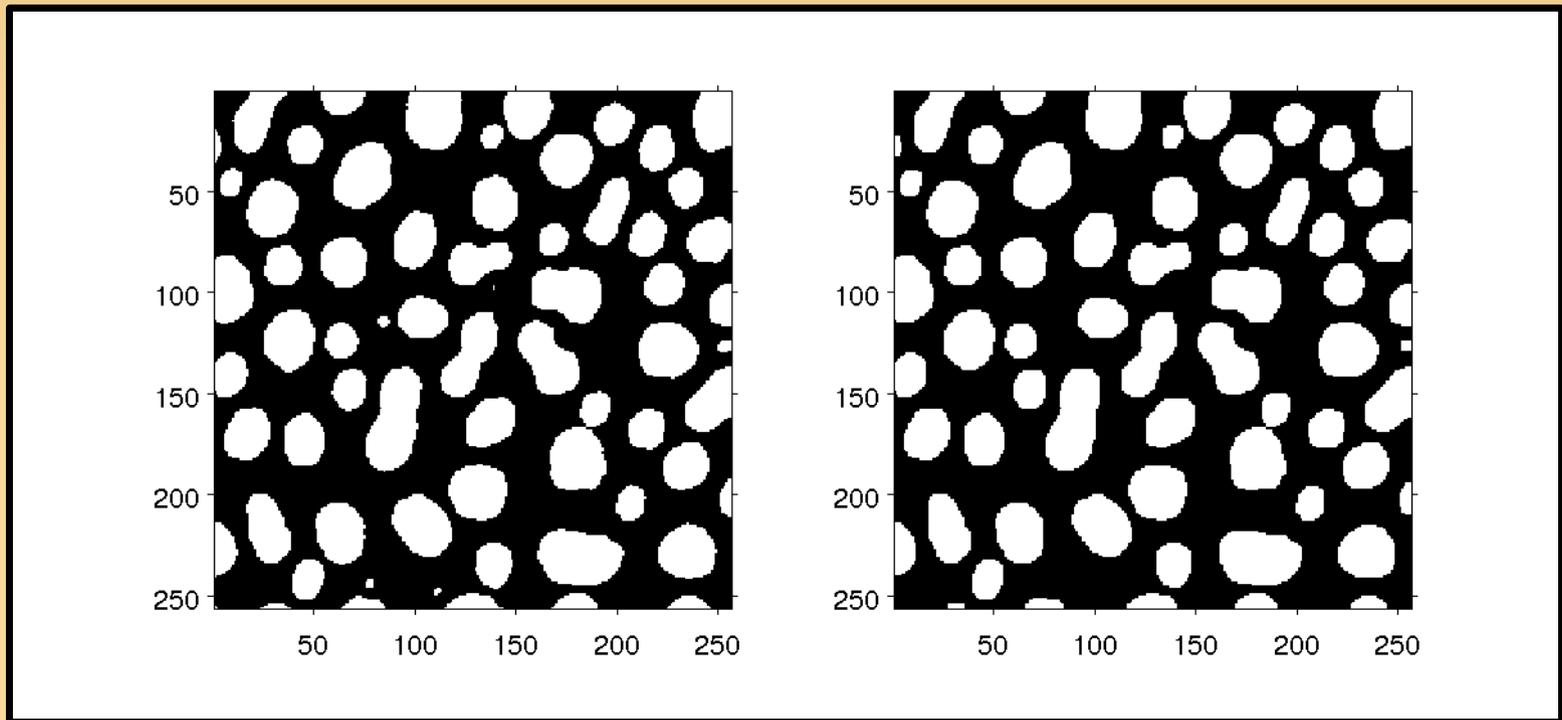
# Operaciones Morfológicas

- Ejemplos:
  - Closing



# Operaciones Morfológicas

- Ejemplos:
  - Opening



# Componentes Conexos

- Detecta regiones en las que desde cualquier pixel existe un camino de 1's hacia el resto de su región.
- En Matlab usamos `bwconncomp`



N=61

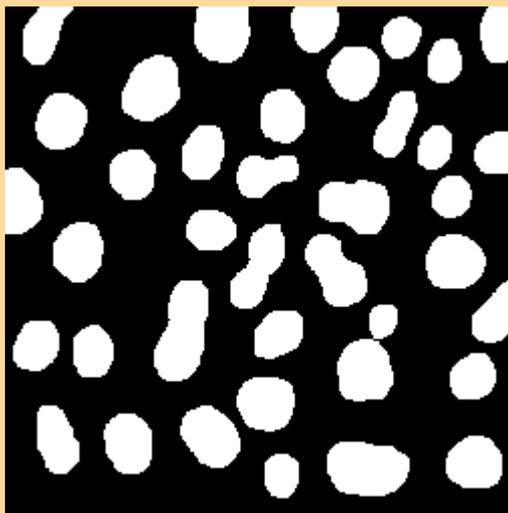
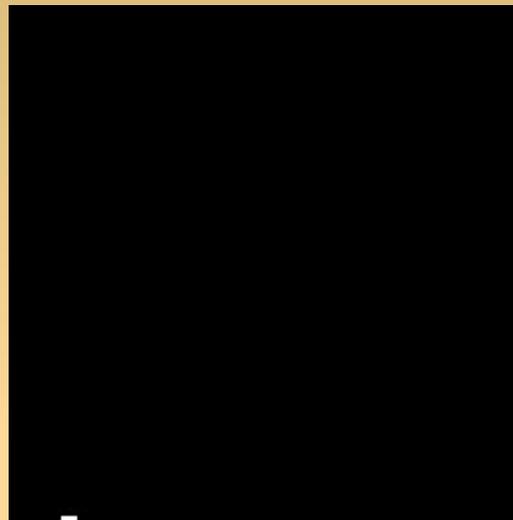
MATLAB

```
function [cc n]=detectCC(image)
    c=bwconncomp(image);
    n=length(c.PixelIdxList);
    cc=zeros(size(image));
    for i=1:n
        cc(c.PixelIdxList{i})=round(rand*255);
    end
end
```

# Componentes Conexos

- Ejemplo de usos:
  - Obtener los componentes conexos y detectar el de mayor tamaño.
  - Obtener los componentes conexos y obtener el más pequeño.
  - Obtener los componentes conexos y eliminar los mas pequeños que un tamaño dado.
  - ¿Cuántos componentes hay?

# Componentes Conexos



FIN