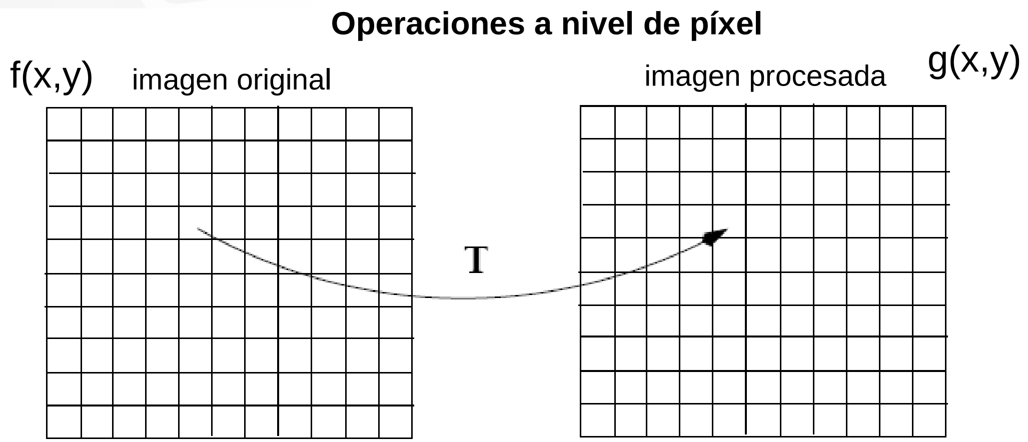
Visión Artificial

Tema 2: Procesamiento de imágenes

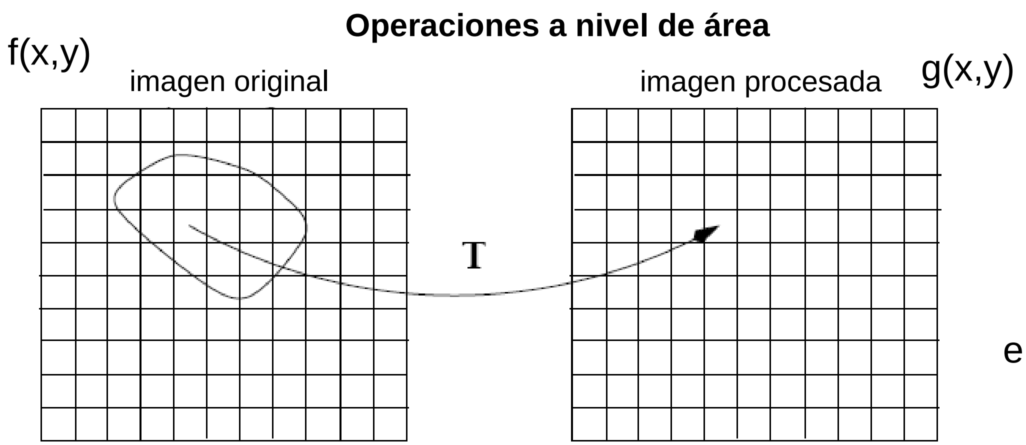
- Operaciones a nivel de píxel
- Operaciones a nivel de área
  - Filtros lineales
  - Filtros no lineales

Procesamiento de imágenes



$g(x, y) = \mathbf{T}[f(x, y)]$

T opera sobre cada píxel



$g(x, y) = \mathbf{T}[f(x, y)]$

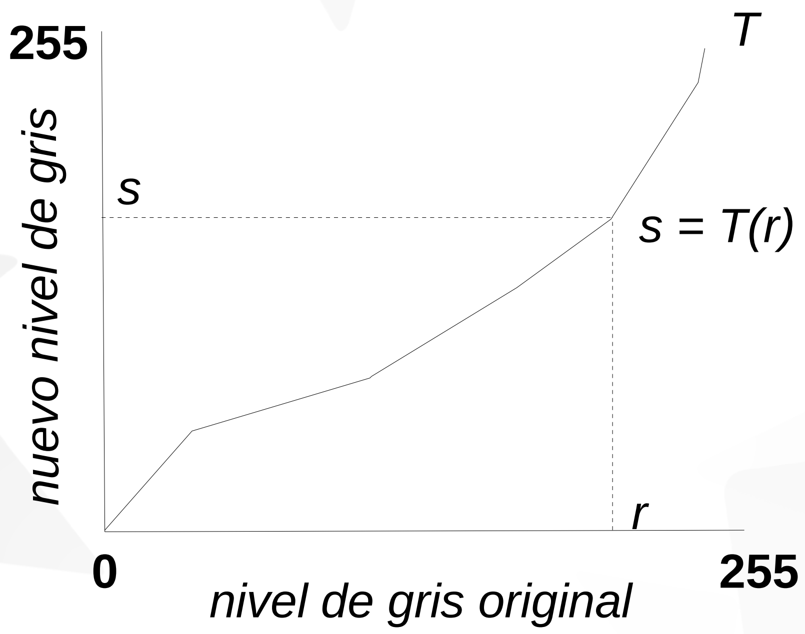
T opera sobre el entorno de cada píxel

Operaciones a nivel de píxel

- Operaciones básicas
- Ecualización del histograma

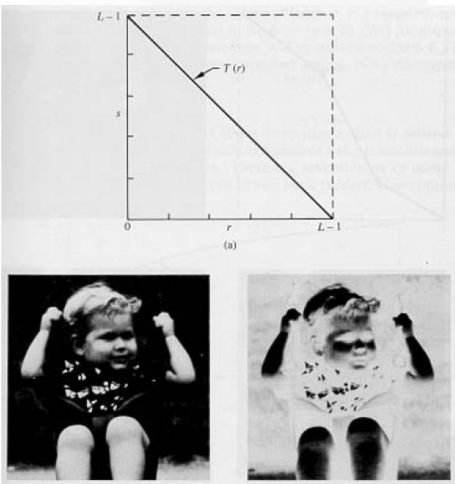
Operaciones a nivel de píxel

- Convierten el valor de cada píxel en un nuevo valor en base a una determinada función predefinida.

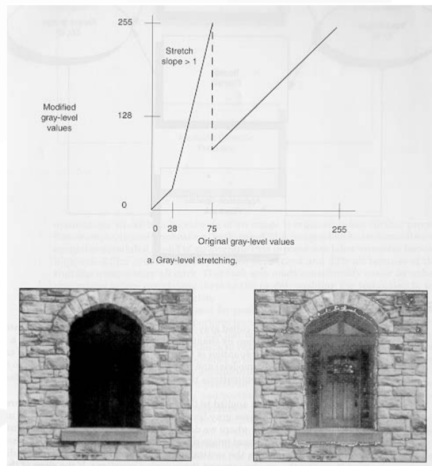


Operaciones a nivel de píxel

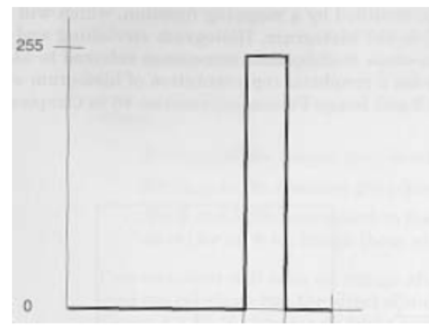
Negativo

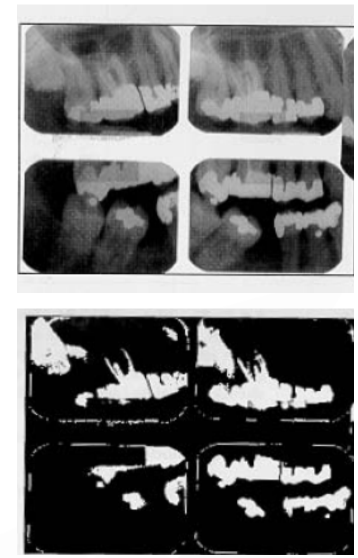


Ampliación de contraste



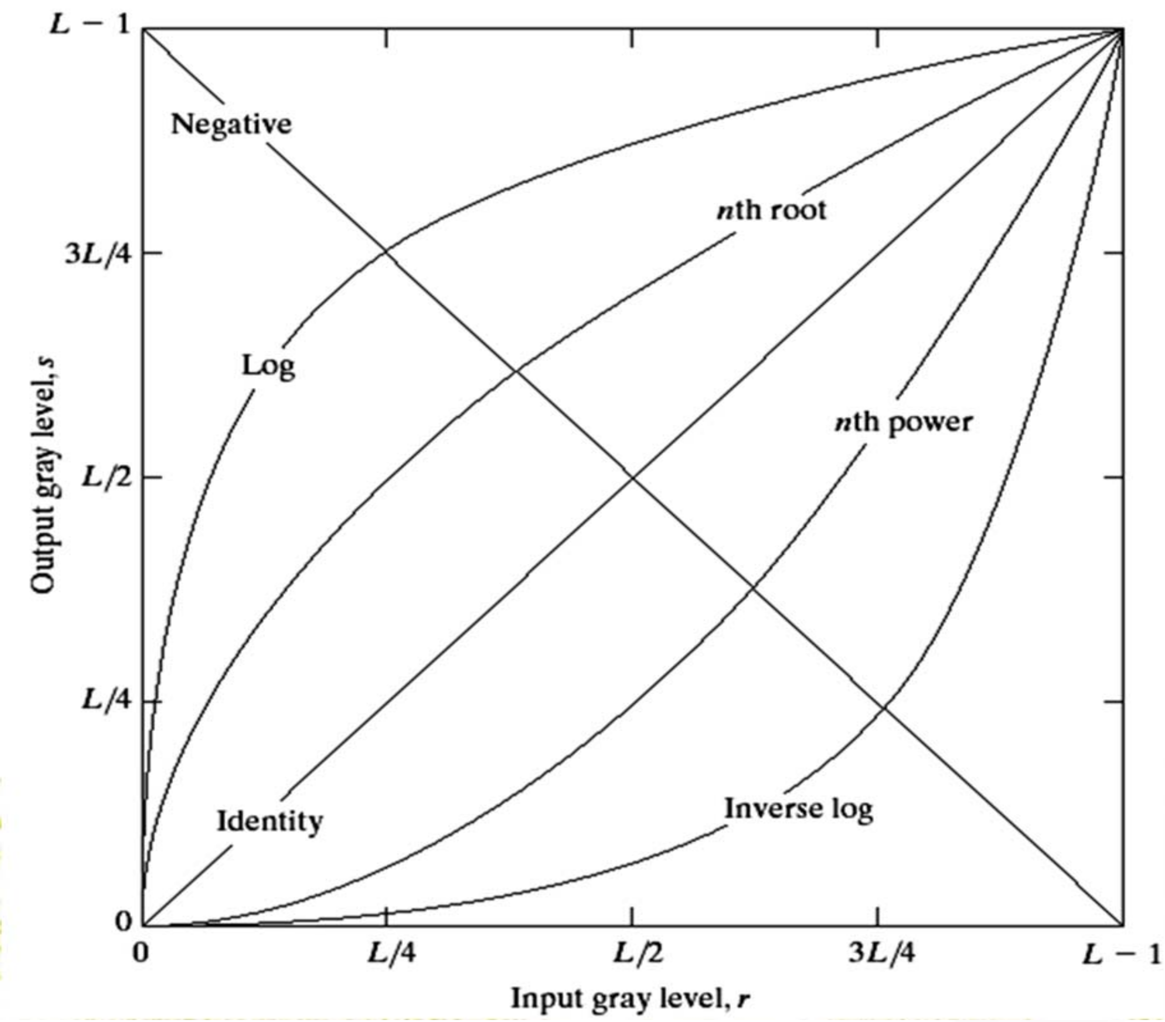
Umbralización





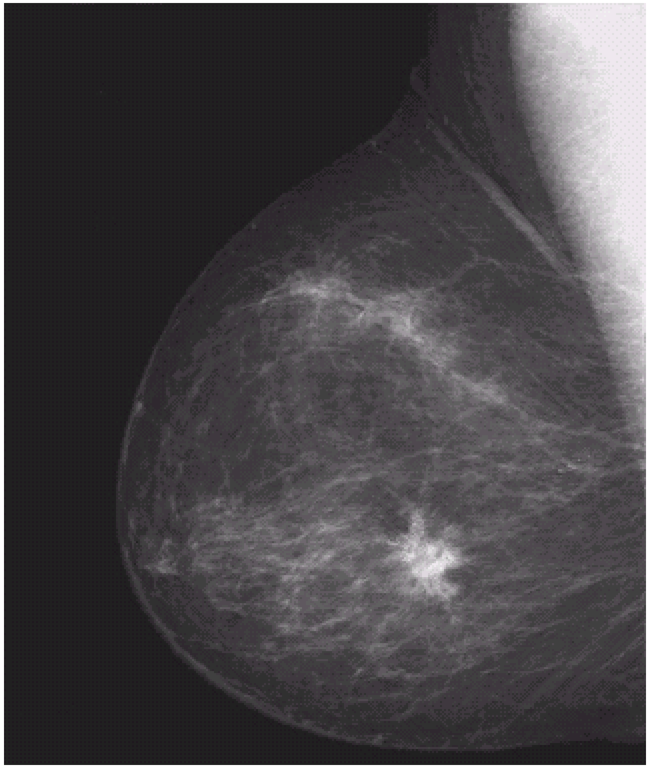
Ecualización
del histograma

Operaciones básicas



Operaciones básicas: negativo

- Las imágenes negativas son útiles para realzar detalles blancos o grises “incrustados” en regiones oscuras
- Este efecto puede apreciarse en la siguiente imagen y su imagen negativa.



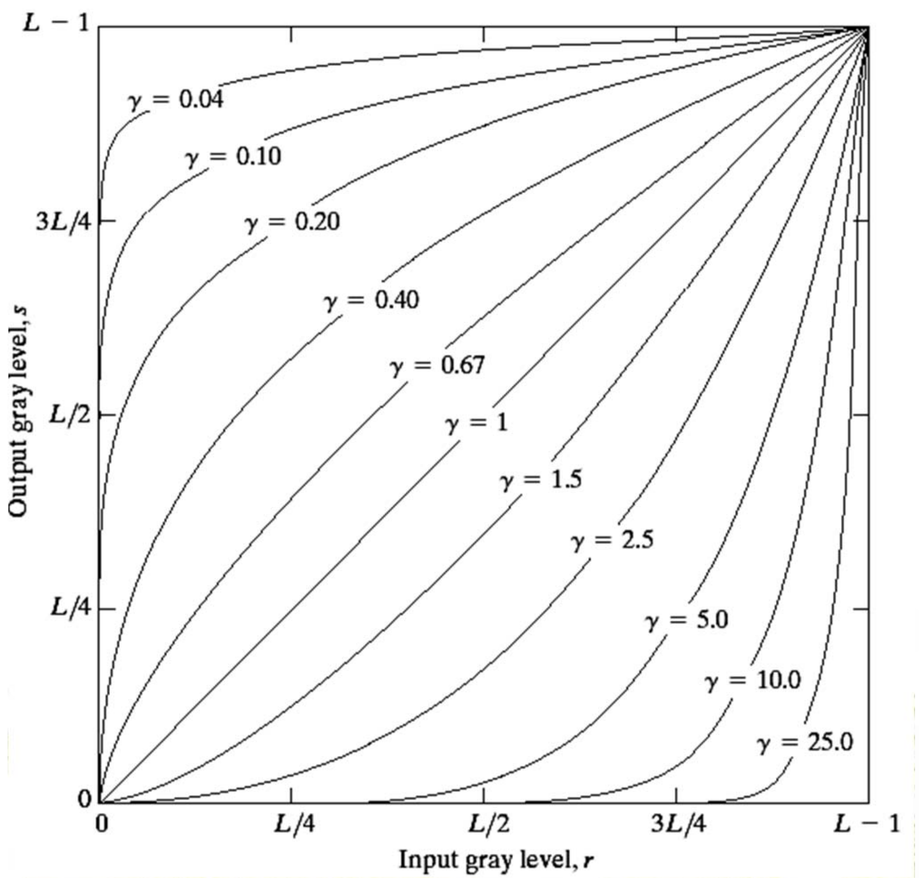
$$s = (L-1) - r$$

Operaciones básicas: función potencia

- La función potencia (filtro de corrección gamma) tiene la forma:

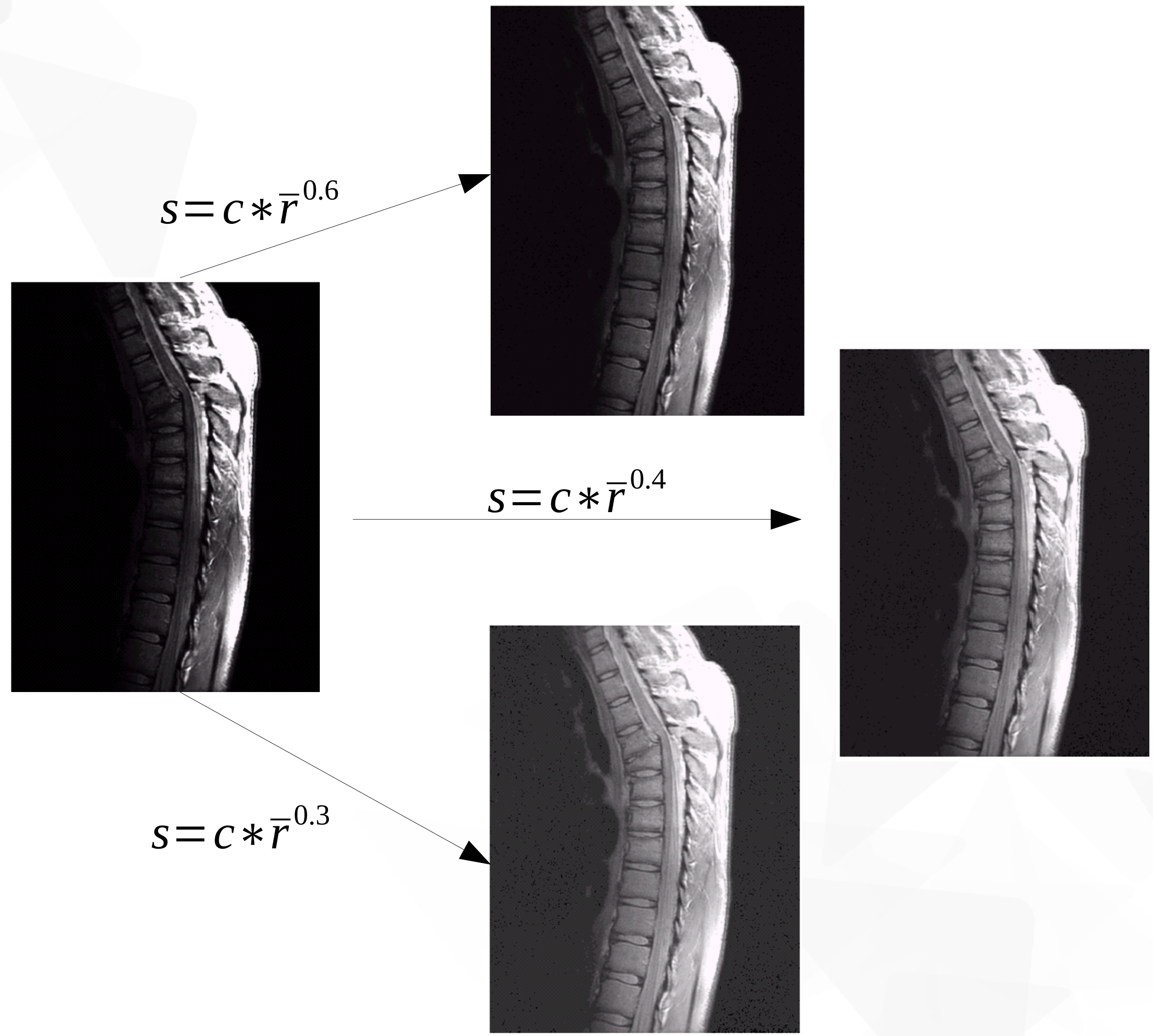
$$s = c * \bar{r}^{\gamma}$$
$$\bar{r} \rightarrow [0..1] \quad c = L - 1$$

- Transforman un rango reducido de valores de una determinada intensidad en un rango más amplio.
- Valores de *γ* inferiores a 1 permiten aclarar la imagen.
- Valores de *γ* superiores a 1 oscurecen la imagen.



Operaciones básicas: función potencia

- Las imágenes de la derecha muestran una resonancia magnética de una espina dorsal.
- Las diferentes curvas permiten resaltar distintos detalles.



Operaciones básicas: función potencia

- Foto aérea excesivamente iluminada.
- La función potencia en este caso se utiliza para oscurecer la imagen.
- Las diferentes curvas permiten resaltar distintos detalles.

Operaciones básicas: función logarítmica

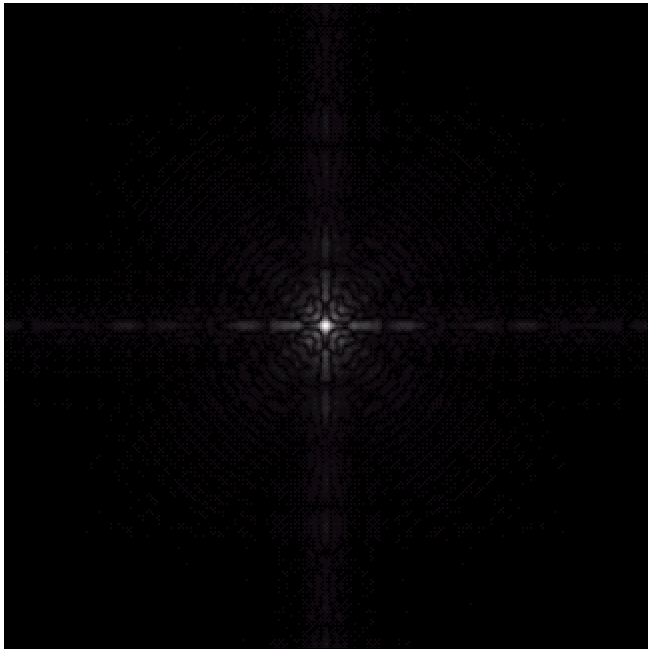
- La forma general de la función logarítmica es:

$$s = c * \log(1 + \bar{r})$$

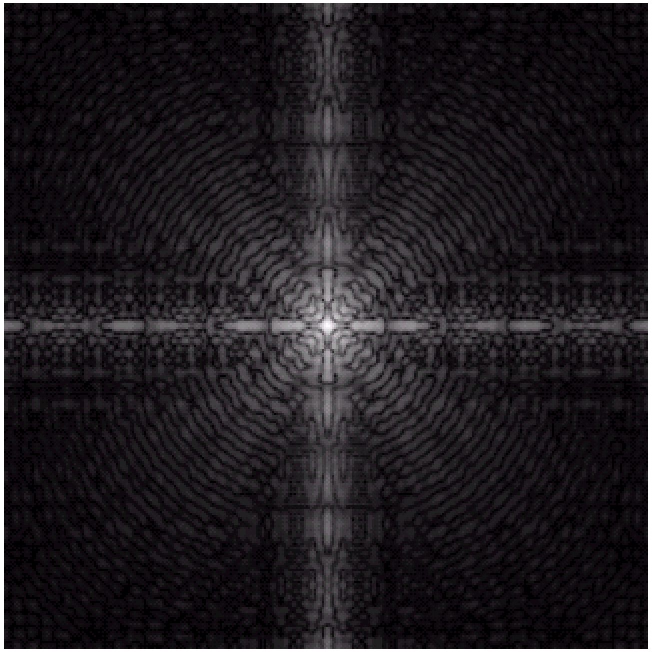
- La función logarítmica transforma un rango reducido de niveles de gris de baja intensidad en un rango más amplio de valores.
- Es un tipo de filtro de aclarado. La ampliación de contraste es mayor que en la función anterior.
- La función logarítmica inversa produce el efecto opuesto: amplía el rango de niveles de gris de alta intensidad.

Operaciones básicas: función logarítmica

- Las funciones logarítmicas son especialmente útiles para aclarar imágenes oscuras y ampliar el contraste.
- El siguiente ejemplo muestra la transformada de Fourier de una imagen y el resultado de aplicar sobre ella una función logarítmica.



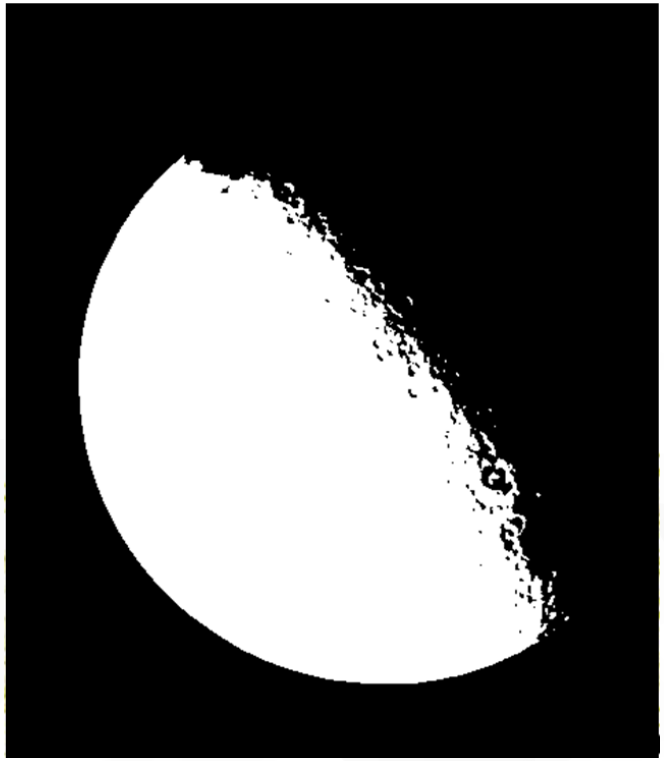
$$s = c * \log(1 + \bar{r}) \longrightarrow$$



Operaciones básicas: umbralización

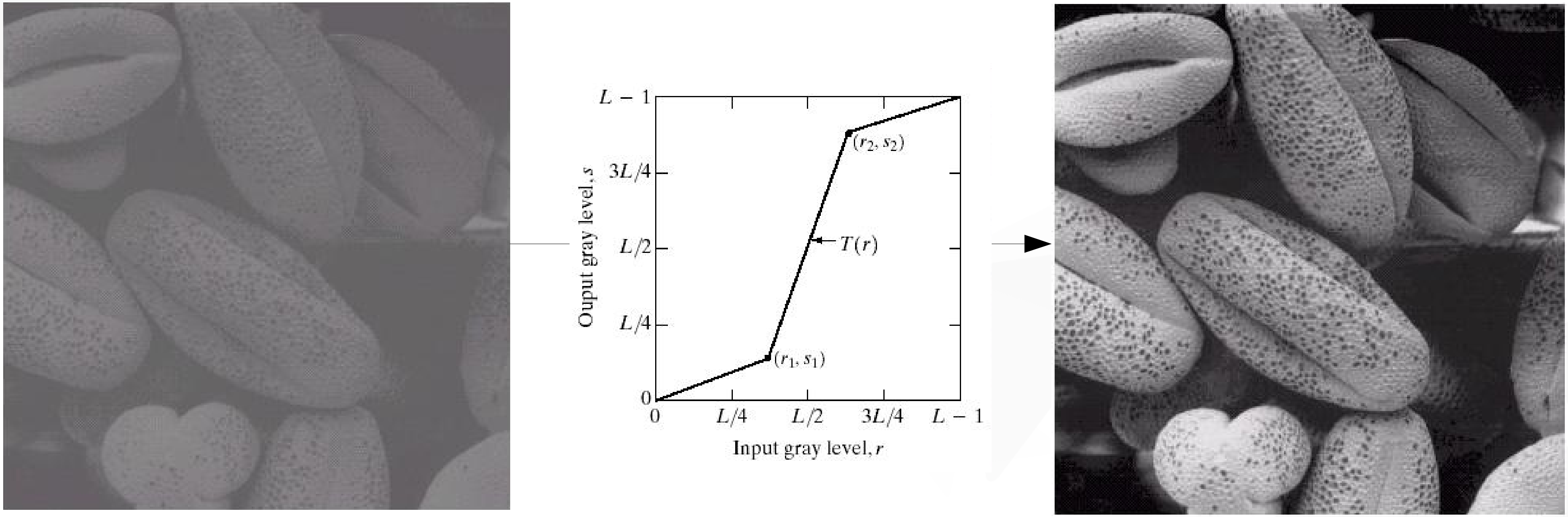
- La umbralización es un tipo de segmentación que permite separar los objetos del fondo en base a un valor de gris escogido como umbral entre ambas zonas.

$$s = \begin{cases} minG \text{ si } r <= \mu \\ maxG \text{ si } r > \mu \end{cases}$$



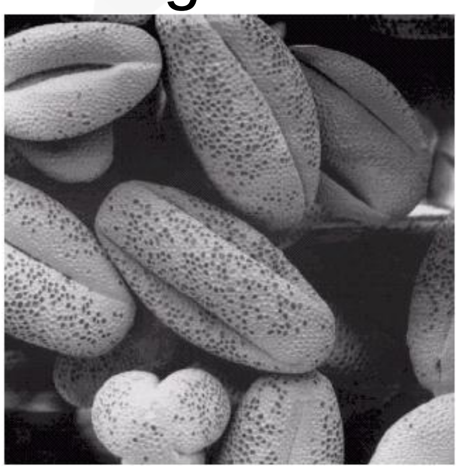
Operaciones básicas: otras funciones

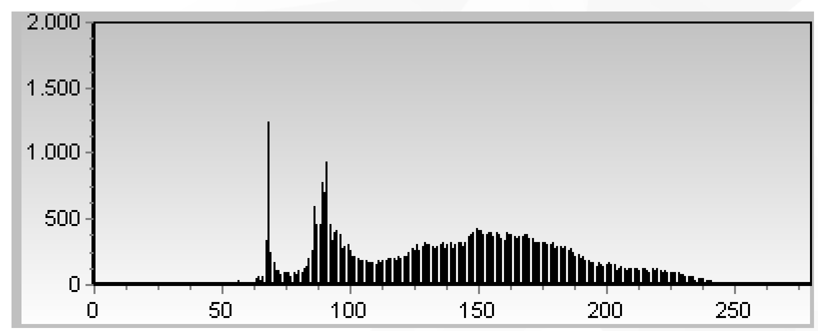
- En lugar de utilizar una función matemática bien definida, es posible aplicar una función definida por tramos.
- Las siguientes imágenes muestran una función de ampliación de contraste definida por tramos y su aplicación sobre una imagen con poco contraste.



Ecualización del histograma

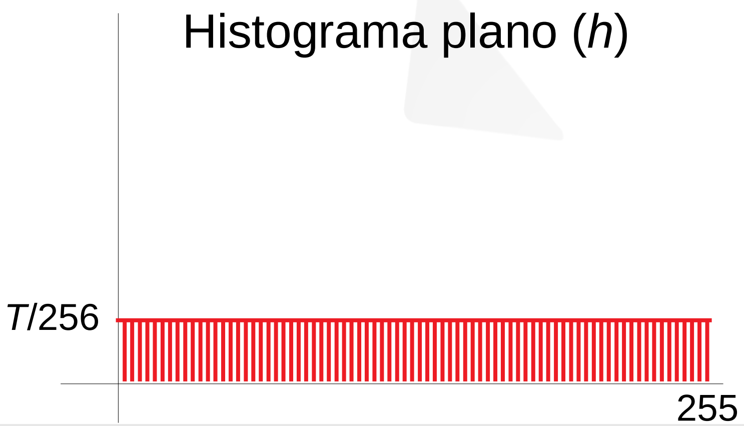
- El histograma de una imagen digital con un rango de niveles de gris [0,*L*-1] es una función discreta $h(r_k) = n_k$, siendo r_k el nivel de gris *k*-ésimo y n_k el número de píxels de la imagen con ese nivel de gris.
- Para una imagen de grises con 8 bits de profundidad, existen 256 posibles valores de intensidad. Así, la representación gráfica del histograma muestra 256 valores que representan la distribución de los píxels de la imagen entre los distintos niveles de gris.

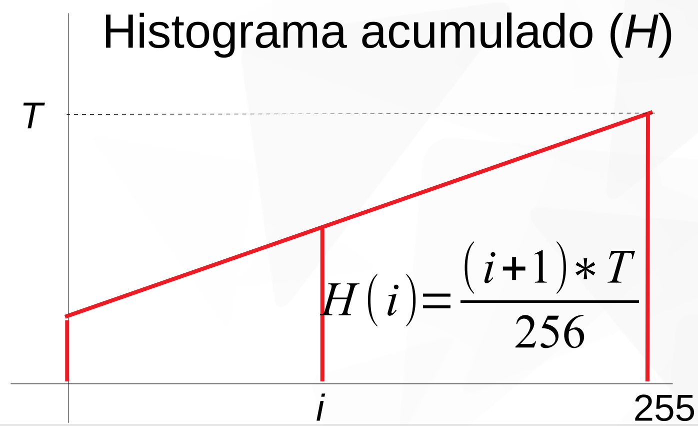




Ecualización del histograma

- El objetivo de la ecualización del histograma es que la imagen tenga una distribución uniforme sobre toda la escala de grises.
- Esto se consigue aproximando el histograma a un histograma plano.
- <u>Problema</u>: píxels con un mismo nivel de gris en el origen, tienen que tener el mismo nivel de gris en el destino.
- <u>Solución</u>: aproximar el histograma acumulado de la imagen al histograma acumulado de un histograma plano.





Ecualización del histograma

- Algoritmo:

1) Crear el histograma acumulado: $H(i) = \sum_{k=0}^{i} h(k)$

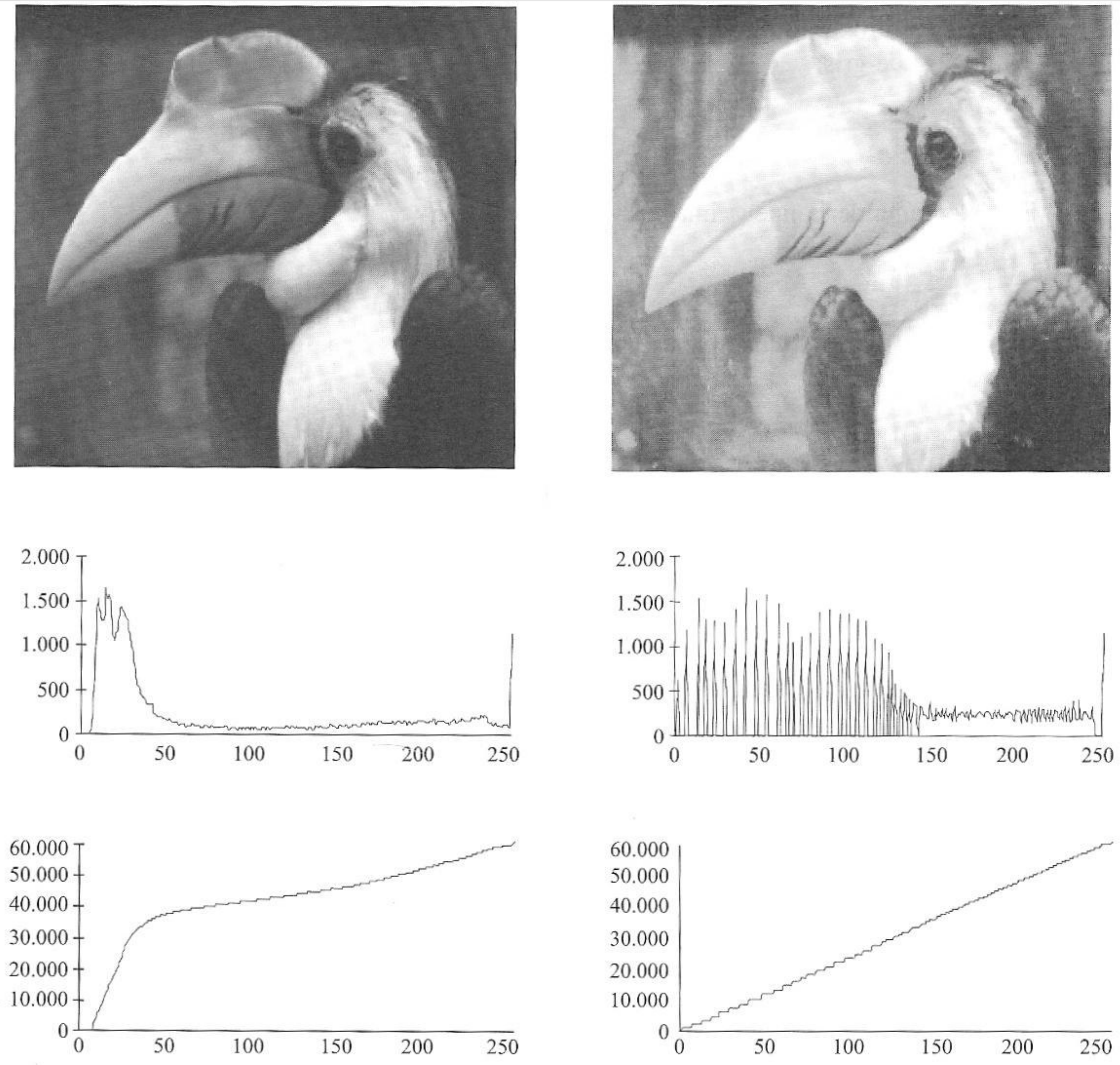
2) Igualar al modelo ideal: $H(i) = G(i') = (i'+1)\frac{NM}{n}$

   - *N* y *M* son las dimensiones de la imagen y *n* el número de niveles de gris.
   - A un nivel de gris ***i*** le corresponde un nivel de gris ***i'*** para el que ***H*(*i*) = *G*(*i'*)**

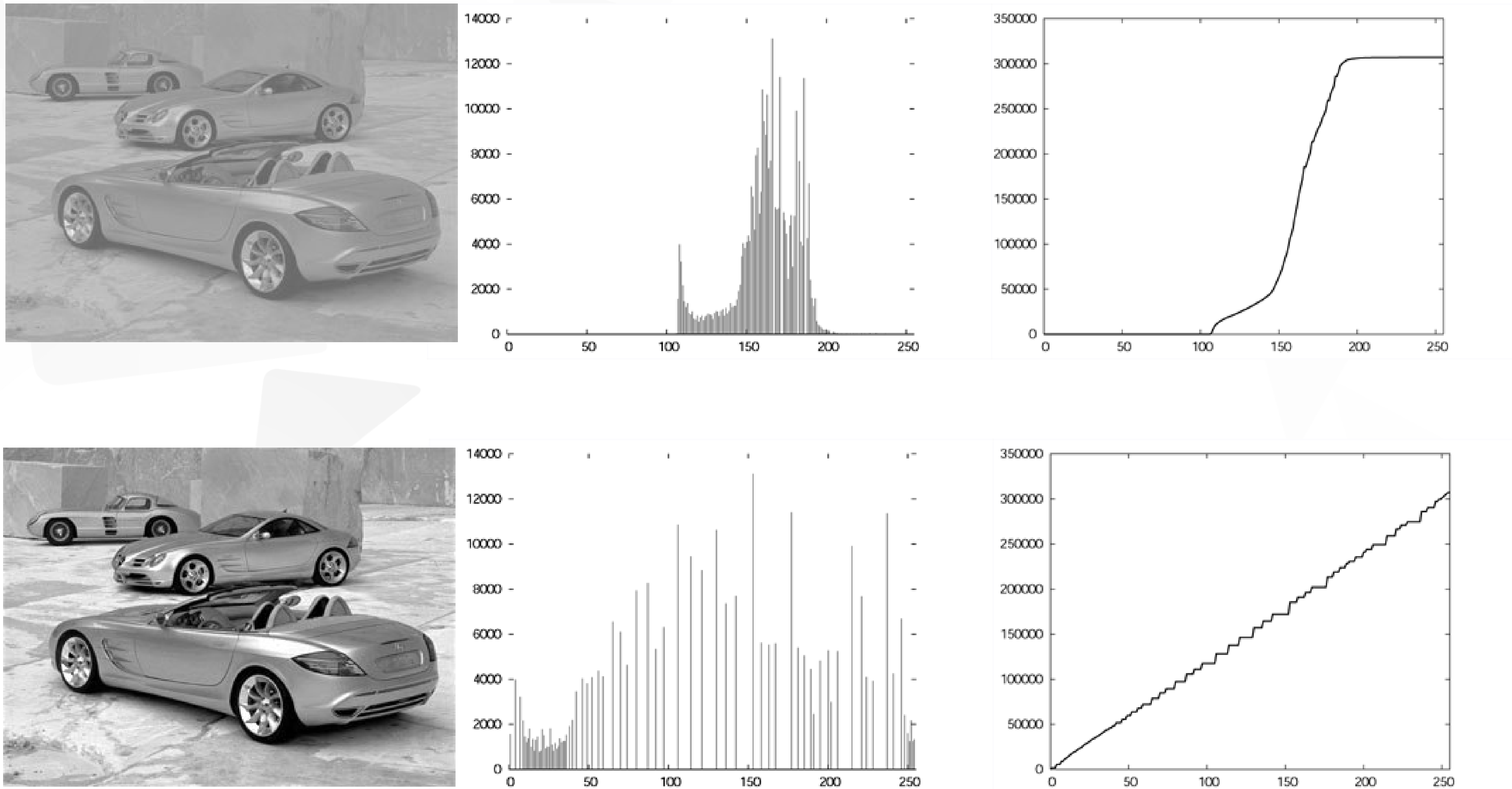
3) Calcular cuál es el nuevo nivel de gris *i'* asociado con cada nivel de gris actual *i*:

$$i' = \frac{n}{NM} H(i) - 1$$

Ecualización del histograma



Ecualización del histograma



Operaciones a nivel de área

- Filtros lineales
- Filtros no lineales

Filtros lineales

- Filtrado de imágenes: modificación de los píxels de la imagen aplicando una determinada función sobre el entorno local de cada píxel.
- Filtrado lineal: reemplaza el valor de cada píxel por una combinación lineal de sus vecinos (correlación cruzada, convolución)
- Los pesos de la combinación lineal del filtrado lineal se organizan en forma matricial. La matriz resultante se conoce como *kernel* (*núcleo*, *máscara*, *filtro*)

10	5	3
4	6	1
1	1	8

Entorno local de un píxel

0	0	0
0	0.5	0
0	1	0.5

kernel

	8	

Valor resultante del píxel

Filtros lineales: correlación cruzada

- Sea *F* una imagen de entrada, *H* un kernel de tamaño (2*k*+1)x(2*k*+1) y *G* la imagen resultante de aplicar el kernel *H* sobre *F*:

$$G[i,j] = \sum_{u=-k}^{k} \sum_{v=-k}^{k} H[u,v] F[i+u, j+v]$$

- Esta expresión se corresponde con la correlación cruzada entre *H* y *F* y se representa como:

$$G = H \otimes F$$

Filtros lineales: convolución

- Igual que la correlación cruzada, excepto que el kernel se aplica “volteado” en horizontal y vertical.

$$G[i,j] = \sum_{u=-k}^{k} \sum_{v=-k}^{k} H[u,v]F[i-u,j-v]$$

- Esta expresión se representa como:

$$G = H * F$$

- Si *H* es simétrica, la convolución equivale a una correlación cruzada.

Filtros lineales: convolución

$$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} *$$

H

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

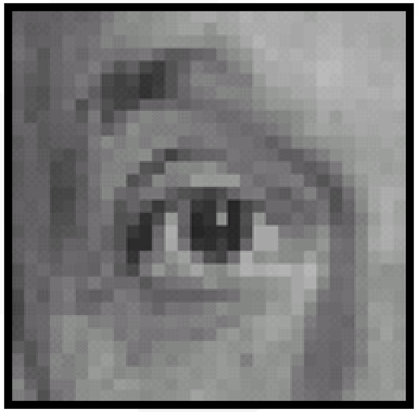
F

$=$

0	10	20	30	30	30	20	10
0	20	40	60	60	60	40	20
0	30	60	90	90	90	60	30
0	30	50	80	80	90	60	30
0	30	50	80	80	90	60	30
0	20	30	50	50	60	40	20
10	20	30	30	30	30	20	10
10	10	10	0	0	0	0	0

G

Filtros lineales: ejemplos



Original

$$* \quad \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad =$$

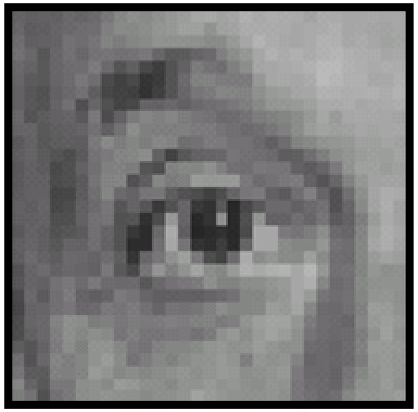
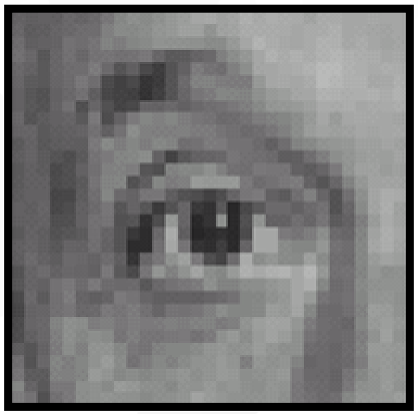


Imagen idéntica

Filtros lineales: ejemplos



Original

$$* \quad \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad =$$

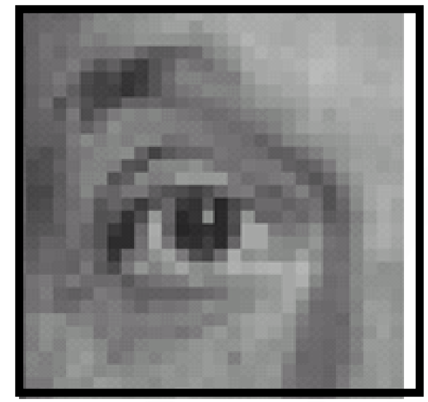
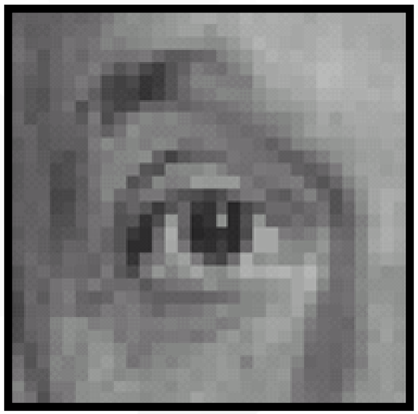


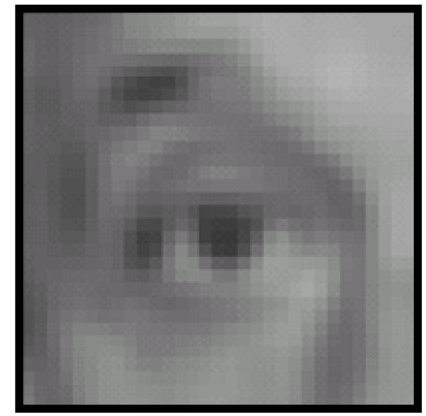
Imagen desplazada un píxel hacia la izquierda

Filtros lineales: ejemplos



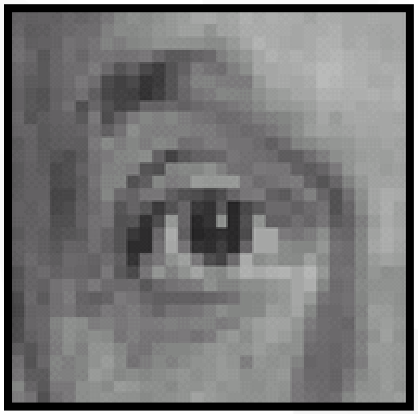
Original

$$* \ \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} =$$



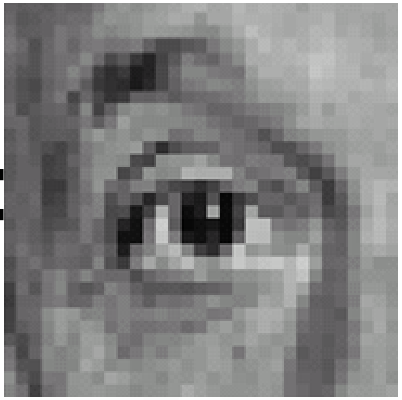
Suavizado (*blur*) con filtro media

Filtros lineales: ejemplos



Original

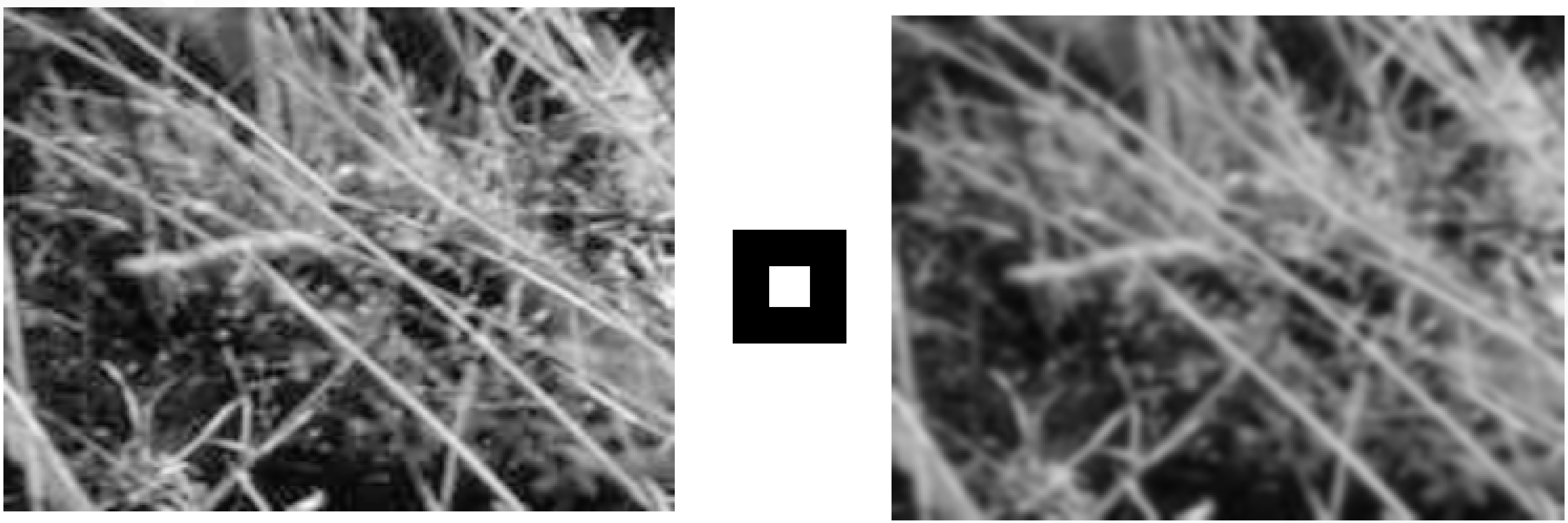
$$* \left(\begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \right) =$$



Filtro de realce
(Resalta los bordes – diferencias con la media local -)

Filtros lineales

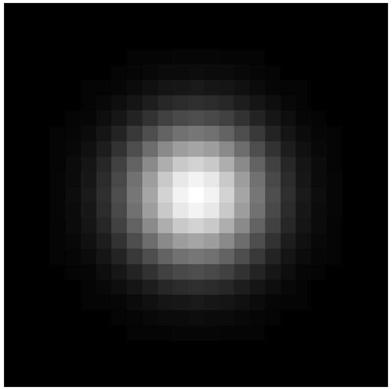
- Uso del filtro media para suavizado: ¿cuál es el problema de aplicar un filtro de este tipo en la imagen inferior?
- ¿Cómo realizar un suavizado en estos casos?



Filtros lineales

- ¿Cómo realizar un suavizado en estos casos?
  - Eliminar los efectos de emborronado, fijando los pesos del filtro asociados a cada píxel en función de su cercanía al centro.

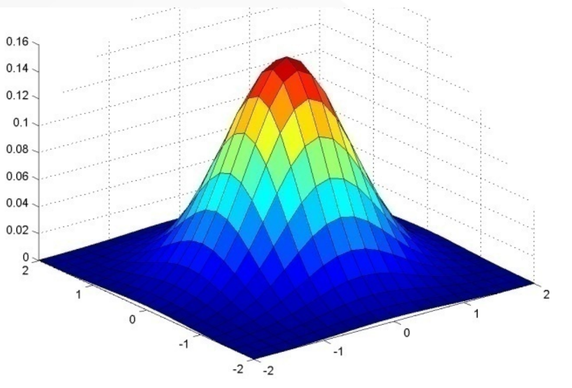
Kernel Gaussiano

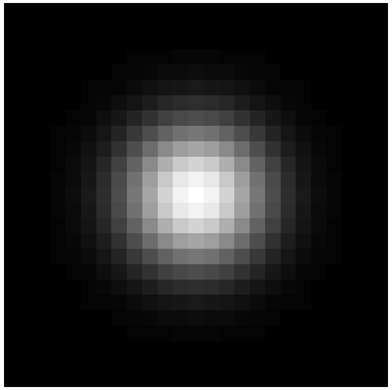


Filtros lineales: suavizado gaussiano

- El *kernel* se construye muestreando una función gaussiana.

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



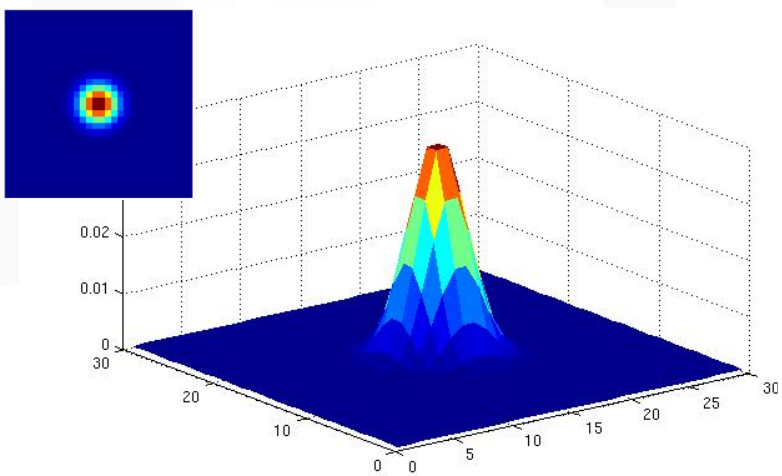


0.003	0.013	0.022	0.013	0.003
0.013	0.059	0.097	0.059	0.013
0.022	0.097	0.159	0.097	0.022
0.013	0.059	0.097	0.059	0.013
0.003	0.013	0.022	0.013	0.003

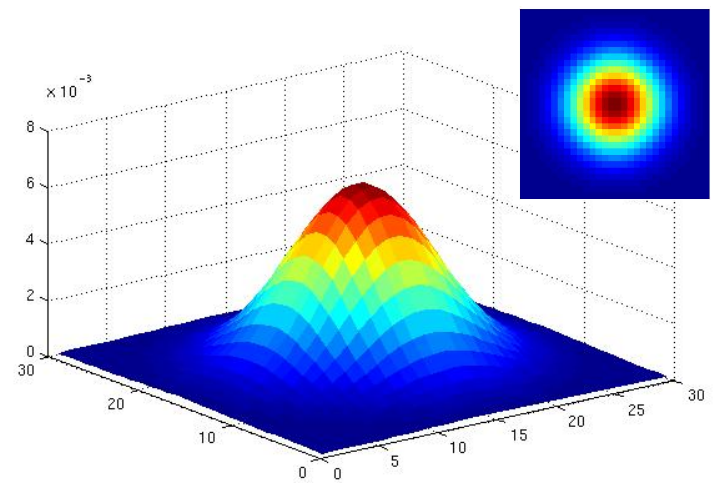
5 x 5, σ = 1

Filtros lineales: suavizado gaussiano

- Desviación típica σ: determina el efecto del suavizado. A mayor desviación típica, mayor contribución de los vecinos del píxel.

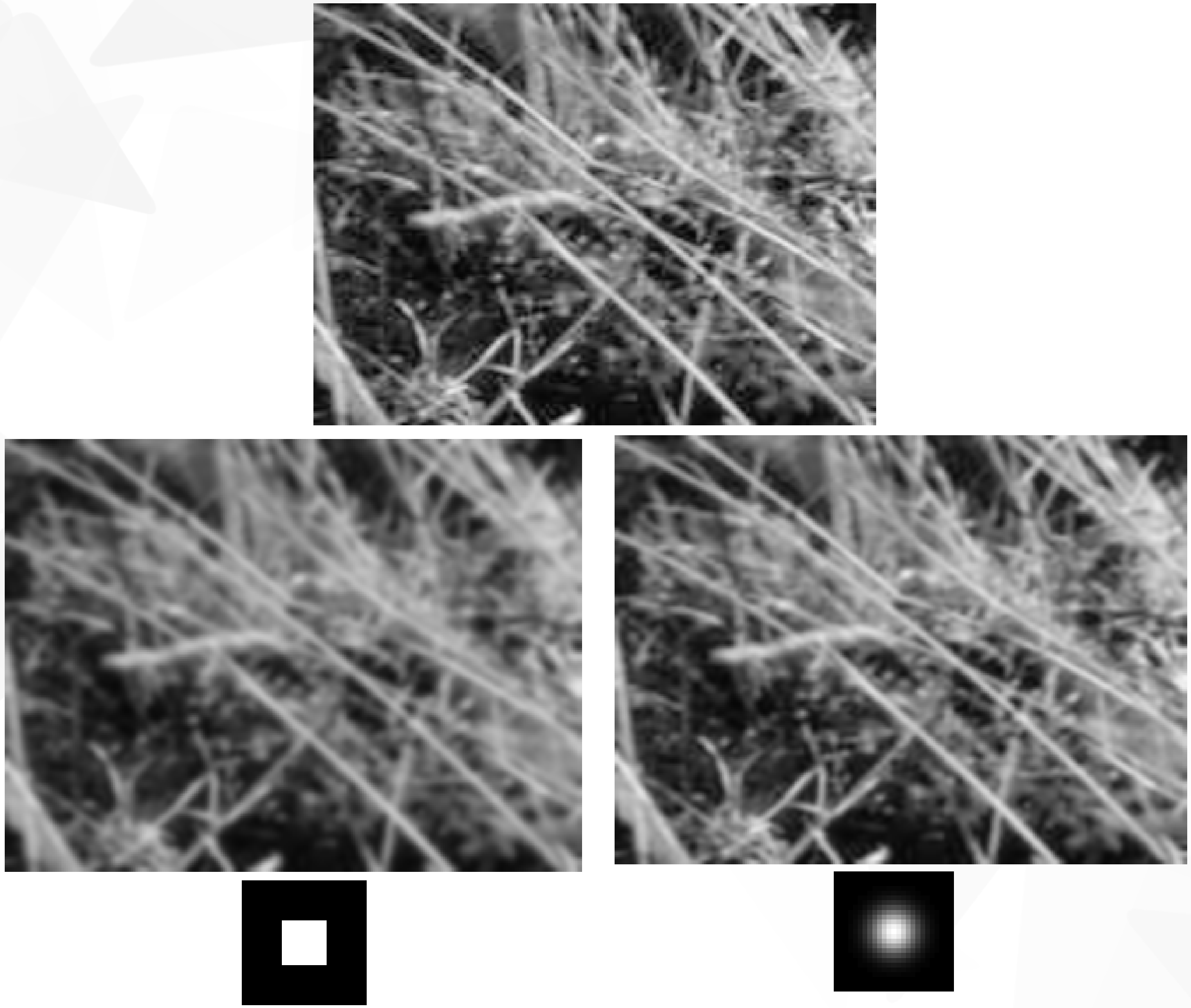


σ = 2



σ = 5

Filtros lineales: suavizado gaussiano



Filtros lineales: suavizado gaussiano

- Elimina las componentes de alta frecuencia de la imagen (filtro paso-bajo)
- Una convolución con otro filtro gaussiano produce a su vez un filtro gaussiano
  - Aplicar varias veces un filtro gaussiano con una desviación típica reducida produce el mismo efecto que aplicar una sola vez un filtro con una desviación típica mayor.
  - Convolucionar dos veces con un kernel gaussiano con desviación típica σ es lo mismo que convolucionar una sola vez con un kernel de desviación típica $\sigma\, 2^{1/2}$
- *Kernel separable*
  - Un kernel es separable si se puede expresar a través del producto entre dos vectores.
  - Reduce la complejidad.

Filtros lineales: suavizado gaussiano

- Separabilidad:
  - La función Gaussiana 2D puede expresarse como el producto de 2 funciones, una en *x* y otra en *y*.
  - Las dos funciones son gaussianas 1D

$$
\begin{aligned}
G_\sigma(x,y) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{x^2+y^2}{2\sigma^2}} \\
&= \left(\frac{1}{\sqrt{2\pi}\sigma} \exp^{-\frac{x^2}{2\sigma^2}} \right) \left(\frac{1}{\sqrt{2\pi}\sigma} \exp^{-\frac{y^2}{2\sigma^2}} \right)
\end{aligned}
$$

Filtros lineales: suavizado gaussiano

Convolución 2D

$$\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} * \begin{bmatrix} 2 & 3 & 3 \\ 3 & 5 & 5 \\ 4 & 4 & 6 \end{bmatrix}$$

$$\begin{aligned} &= 2 + 6 + 3 = 11 \\ &= 6 + 20 + 10 = 36 \\ &= 4 + 8 + 6 = 18 \\ &\overline{65} \end{aligned}$$

Descomposición
en 2 filtros 1D

$$\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \times \begin{bmatrix} 1 & 2 & 1 \end{bmatrix}$$

Convolución con
el filtro 1D por filas

$$\begin{bmatrix} 1 & 2 & 1 \end{bmatrix} * \begin{bmatrix} 2 & 3 & 3 \\ 3 & 5 & 5 \\ 4 & 4 & 6 \end{bmatrix} = \begin{bmatrix} & 11 & \\ & 18 & \\ & 18 & \end{bmatrix}$$

Convolución con
el filtro 1D por columnas

$$\begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} * \begin{bmatrix} & 11 & \\ & 18 & \\ & 18 & \end{bmatrix} = \begin{bmatrix} & & \\ & 65 & \\ & & \end{bmatrix}$$

Filtros no lineales

- Filtros lineales:
  - La respuesta del filtro para la suma de dos señales es equivalente a la suma de las dos respuestas del filtro:

$$f(I_1+I_2) = f(I_1)+f(I_2)$$

  - Más directo y eficiente.
- Filtros no lineales:
  - Filtros cuya respuesta no es una función lineal de sus entradas.
  - Mejores resultados en determinados casos.
  - Ejemplos:
    - Filtro mediana.
    - Operadores morfológicos

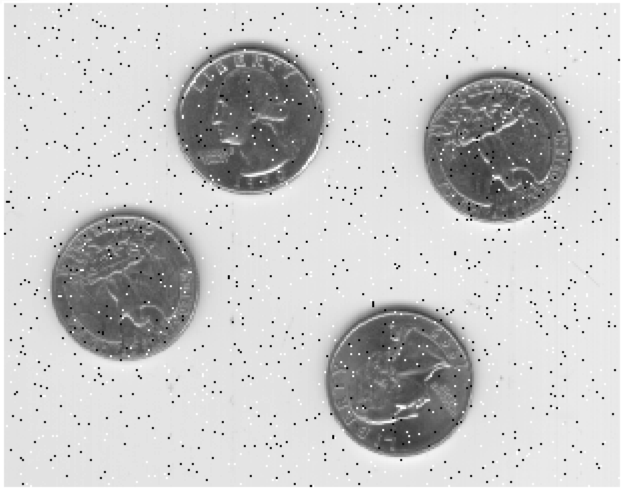
Filtros no lineales: filtro mediana

- Reemplaza el valor de cada píxel con la mediana de los valores de intensidad de sus vecinos.
- Se usa para reducir el ruido impulsivo. Este tipo de ruido se conoce como “sal y pimienta” por su apariencia.

- El filtro mediana es muy útil cuando el objetivo es reducir el ruido y preservar los bordes de la imagen al mismo tiempo.

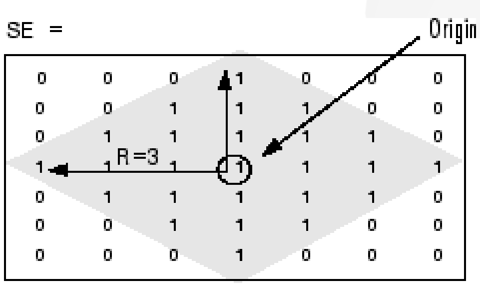
Filtros no lineales: filtro mediana

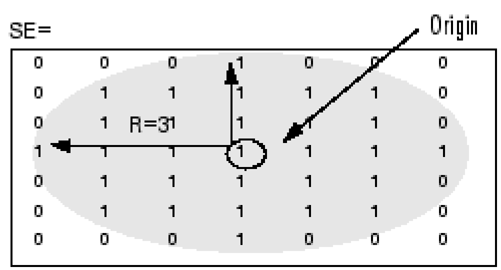
- Por cada píxel situado en una posición (*i*, *j*), tomando un entorno centrado en el píxel de tamaño *N*x*N*:
  - Ordenar los píxels de menor a mayor nivel de gris.
  - Escoger, como nuevo valor del píxel (*i*, *j*), el píxel central de la lista ordenada obtenida en el paso anterior.

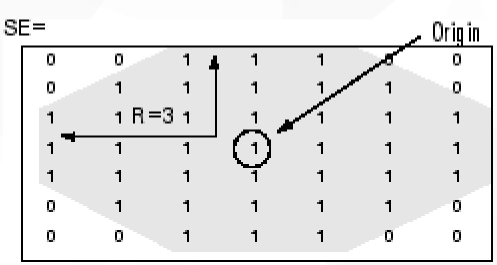


Filtros no lineales: operadores morfológicos

- Se aplican sobre imágenes binarias.
- Operación morfológica:
  - Usada en fases de post-procesamiento.
  - Procedimiento:
    - Convolucionar la imagen con un elemento estructural (filtro binario).
    - Umbralizar en base al número de 1's coincidentes con el elemento estructural
  - Elementos estructurales: diamante, disco, octágono, ...







Filtros no lineales: operadores morfológicos

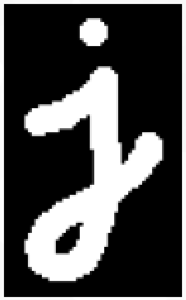
$c = f * s$: número de 1's coincidentes con el elemento estructural (s)

$$\theta(c, t) = \begin{cases} 1 & \text{si } c \geq t \\ 0 & \text{si } c < t \end{cases} \longrightarrow \quad \text{umbralización en base a } c$$

- Operaciones:
  - dilatación: *dilatación*(f, s) = $\theta(c, 1)$
  - erosión: *erosión*(f, s) = $\theta(c, S)$
  - mayoría: *mayoría*(f, s) = $\theta(c, S/2)$
  - apertura: *apertura*(f, s) = *dilatación*(*erosión*(f, s), s)
  - cierre: *cierre*(f, s) = *erosión*(*dilatación*(f, s), s)

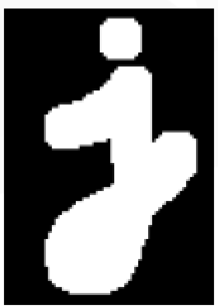
S : tamaño del elemento estructural (número de píxels)

Filtros no lineales: operadores morfológicos

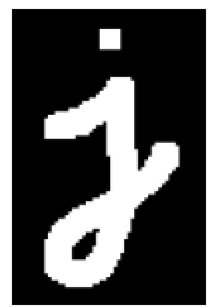


<u>Original</u>
Imagen binaria original

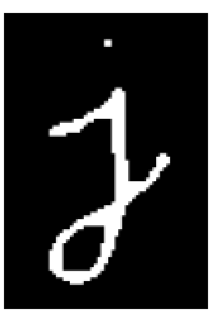
<u>Mayoría</u>
Suaviza los bordes del objeto



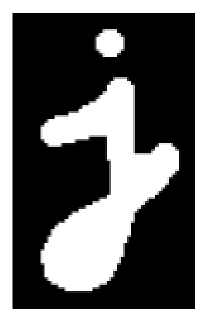
<u>Dilatación</u>
Ampliación (engorde) de objetos



<u>Apertura</u>
Separa partes de un objeto y elimina pequeñas regiones asociadas al ruido



<u>Erosión</u>
Reducción (adelgazamiento) de objetos



<u>Cierre</u>
Operación dual de la apertura. Cierra agujeros y rellena fisuras