

---

# Mejora de la Imagen Image Enhancement

---

---

# Fuentes

- Digital Image Processing – Gonzalez y Woods. Disponible en <http://www.imageprocessingplace.com/>
  - Digital Image Processing, An Algorithmic Approach – Burger y Burge. Disponible en <https://imagingbook.com/>
  - Computer Vision for Busy Developers, Vini Da Silva. Disponible en <https://medium.com/@vad710/cv-for-busy-devs-scale-space-9368e3be938b>
  - “A review of image denoising algorithms, with a new one”. A. Buades, B. Coll and J. M. Morel. SIAM Multiscale Modelisation and Simulation, vol. 4, No. 2, pp. 490–530
-

---

# Para que?

- Mejorar la visualización
  - Facilitar el procesamiento posterior
  - Adecuar la imagen para una aplicación específica
-

---

# Por que?

- Presencia de ruido
  - Pocos niveles de grises
  - Iluminación no uniforme
-

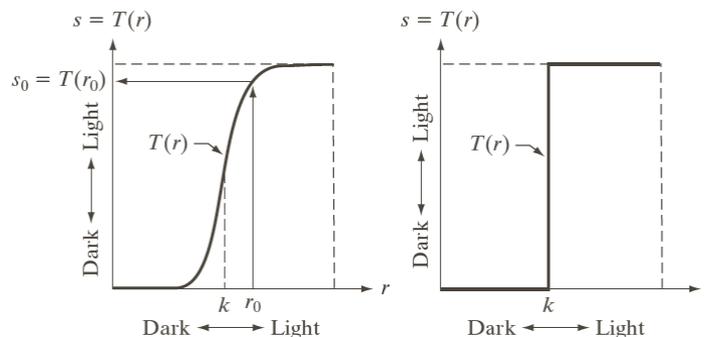
---

# Mejoramiento

- Realce de características de la imagen:
    - bordes
    - fronteras
    - contraste
  - No incrementa la información
  - Modifica el rango dinámico.
  - Facilita detección
  - Orientadas al problema, subjetivas, cualitativas
-

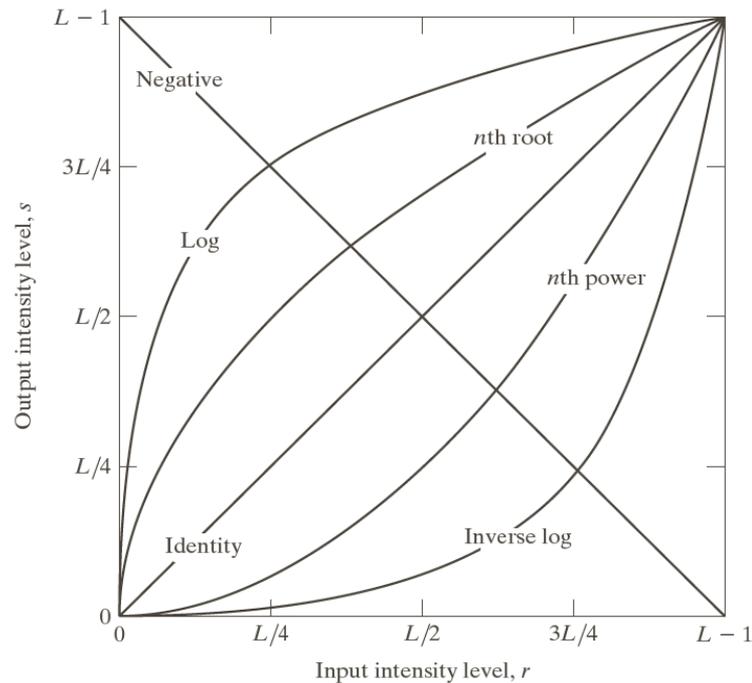
# Operaciones Puntuales

- Operación sin memoria que mapea cada nivel de gris en un nuevo valor de gris de acuerdo a la transformación  $v=f(u)$
- Ejemplos estiramiento de contraste y umbralización:



# Operaciones Puntuales

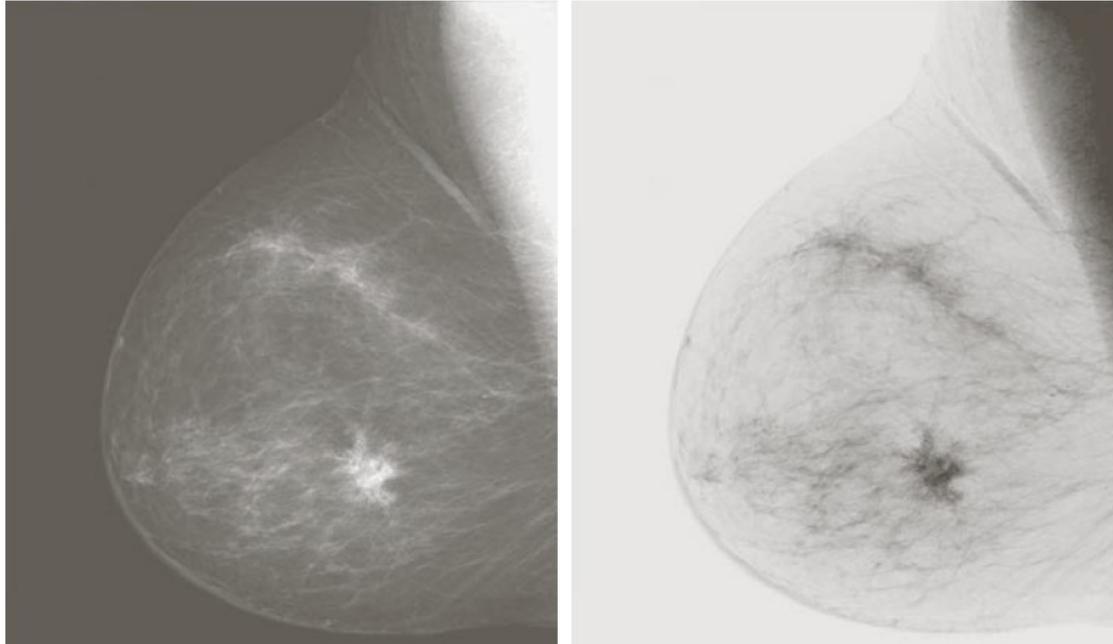
- Se puede usar una LUT



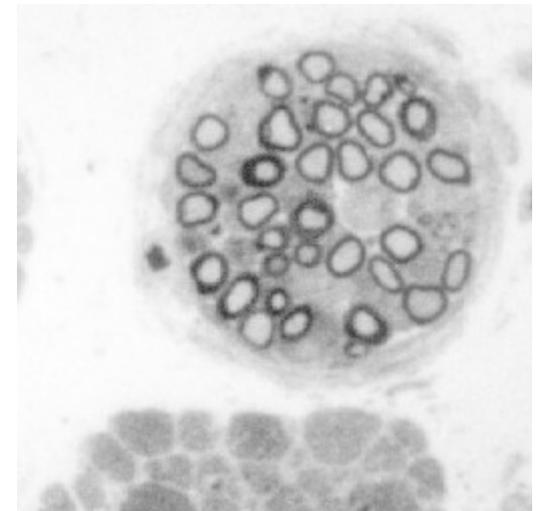
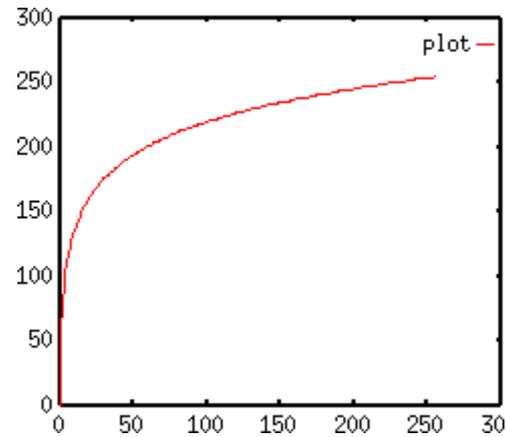
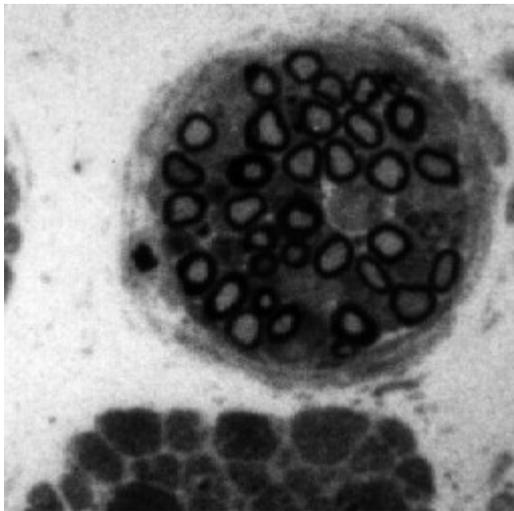
# Operaciones Puntuales

- Estiramiento de contraste
- Umbral
- Clipping
- Inversión (negativo)
- Compresión de Rango (logarítmico)
- Sustracción de Imágenes
- Modelado de Histograma

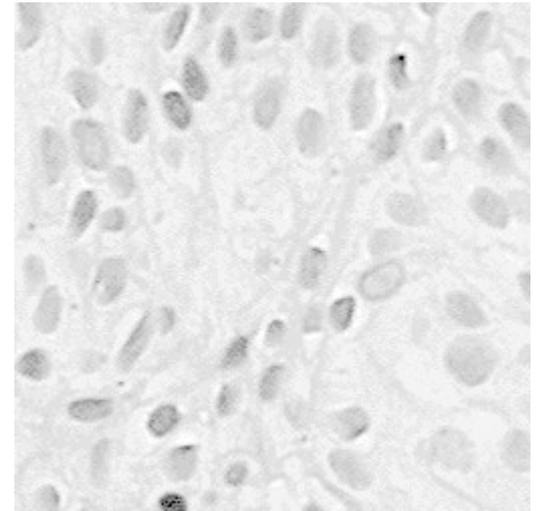
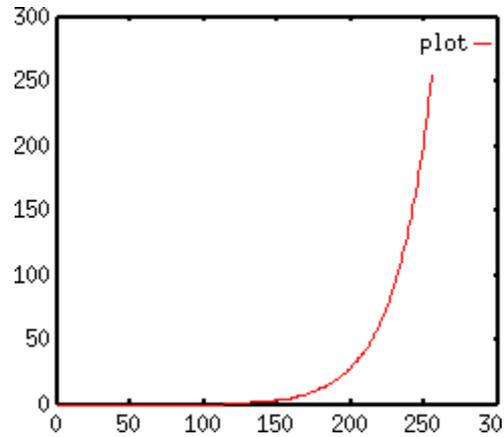
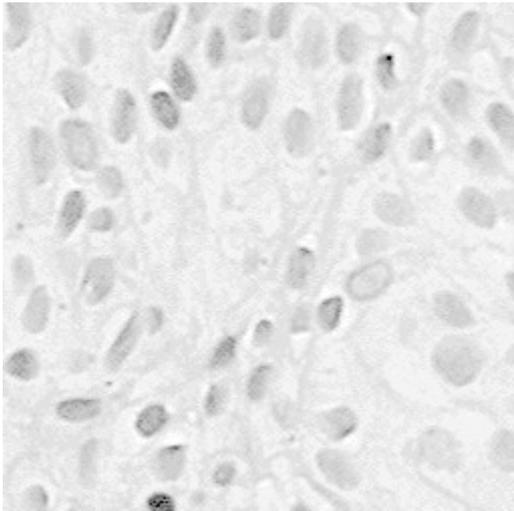
# Negativo



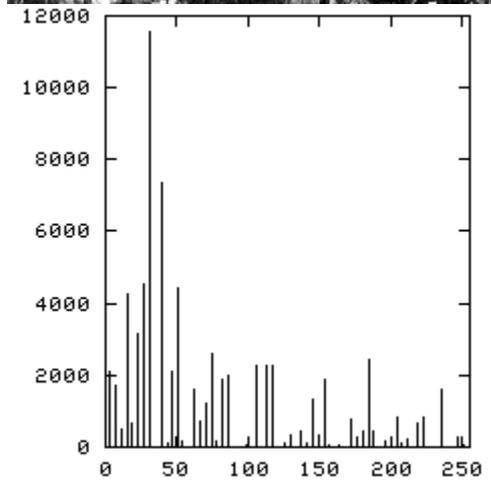
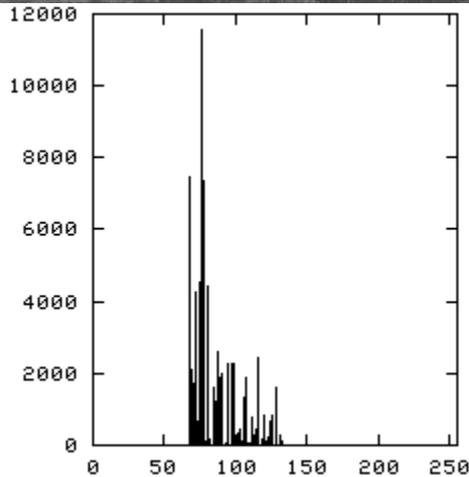
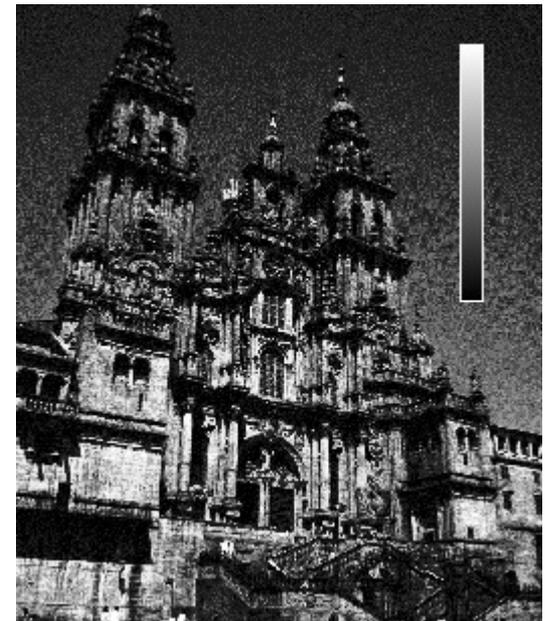
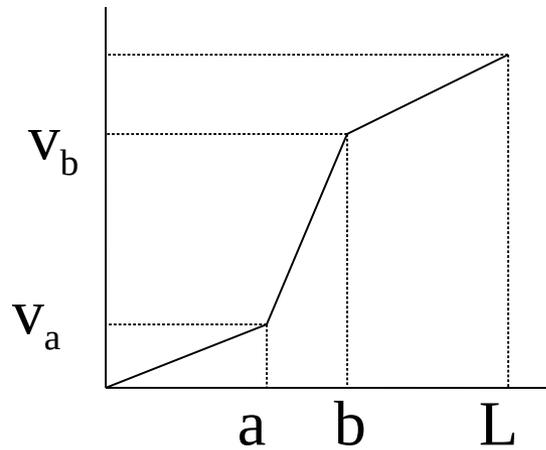
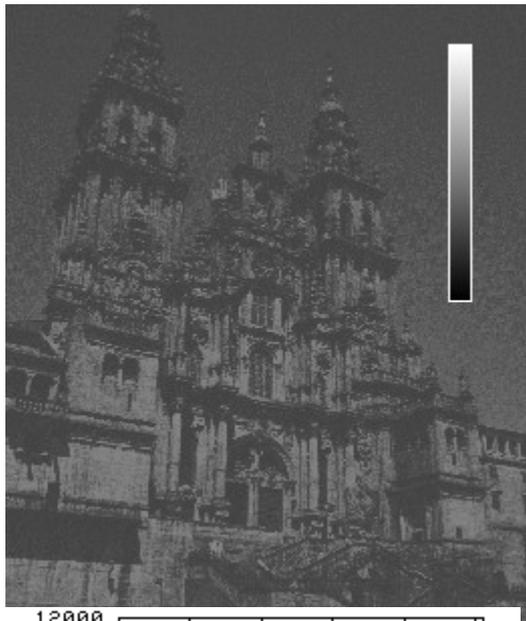
# Mejoramiento de contraste logarítmico

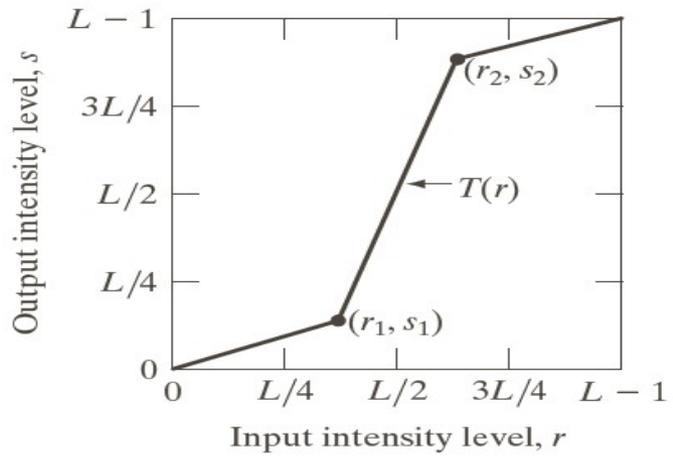


# Mejoramiento de contraste exponencial

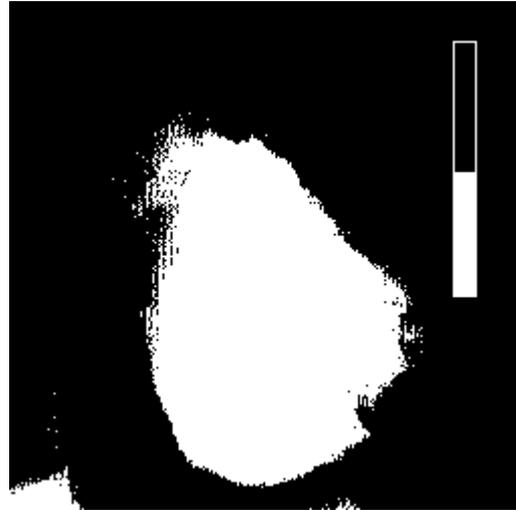
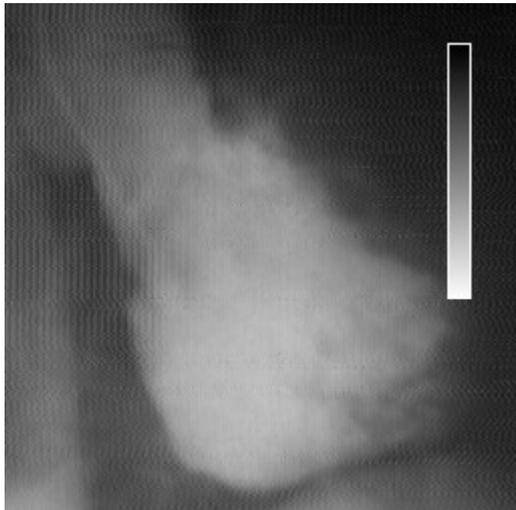


# Mejoramiento de contraste por estiramiento de histograma

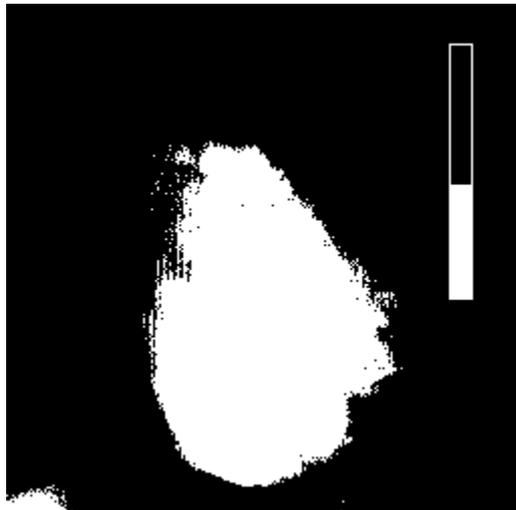
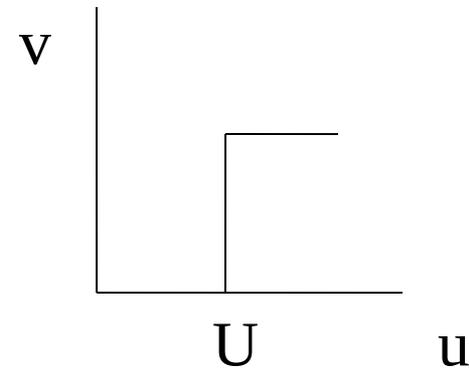




# Umbral(Thresholding)



118



Nivel	Area
118	19,670 pixels
128	16,969 pixels
138	14,462 pixels

Ref: Imagina

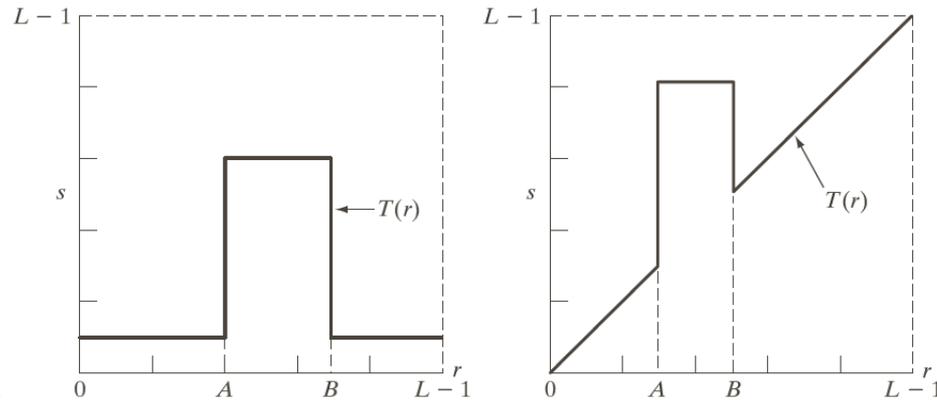
128

138

# Mejoramiento de contraste por determinación de ventana y nivel

a b

**FIGURE 3.11** (a) This transformation highlights intensity range  $[A, B]$  and reduces all other intensities to a lower level. (b) This transformation highlights range  $[A, B]$  and preserves all other intensity levels.



# Mejoramiento de contraste por determinación de ventana y nivel



a b c

**FIGURE 3.12** (a) Aortic angiogram. (b) Result of using a slicing transformation of the type illustrated in Fig. 3.11(a), with the range of intensities of interest selected in the upper end of the gray scale. (c) Result of using the transformation in Fig. 3.11(b), with the selected area set to black, so that grays in the area of the blood vessels and kidneys were preserved. (Original image courtesy of Dr. Thomas R. Gest, University of Michigan Medical School.)

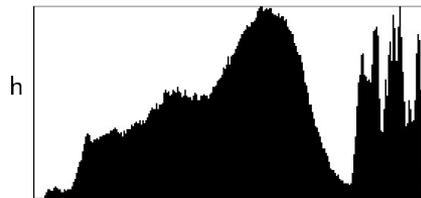
# Ecualización de Histograma



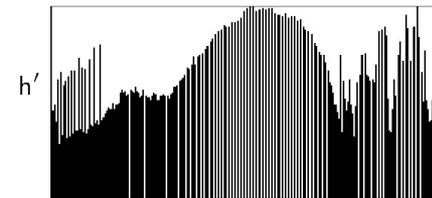
(a)



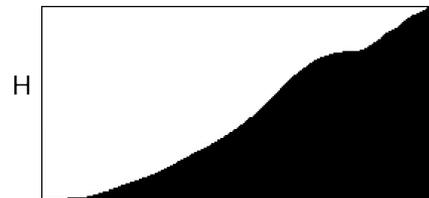
(b)



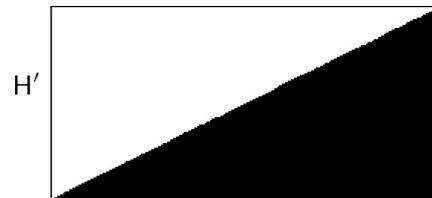
(c)



(d)

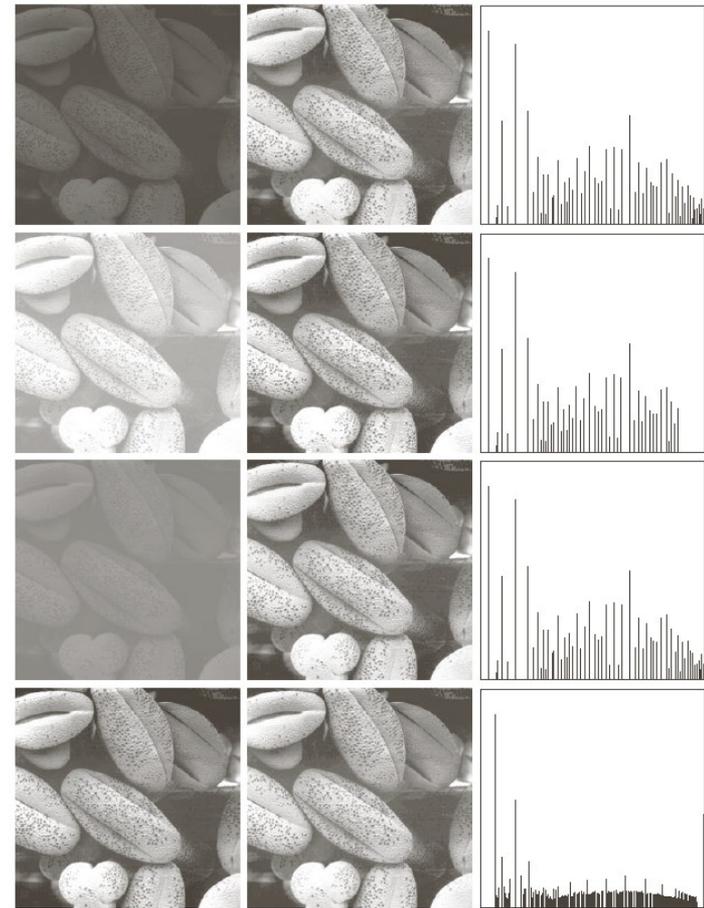
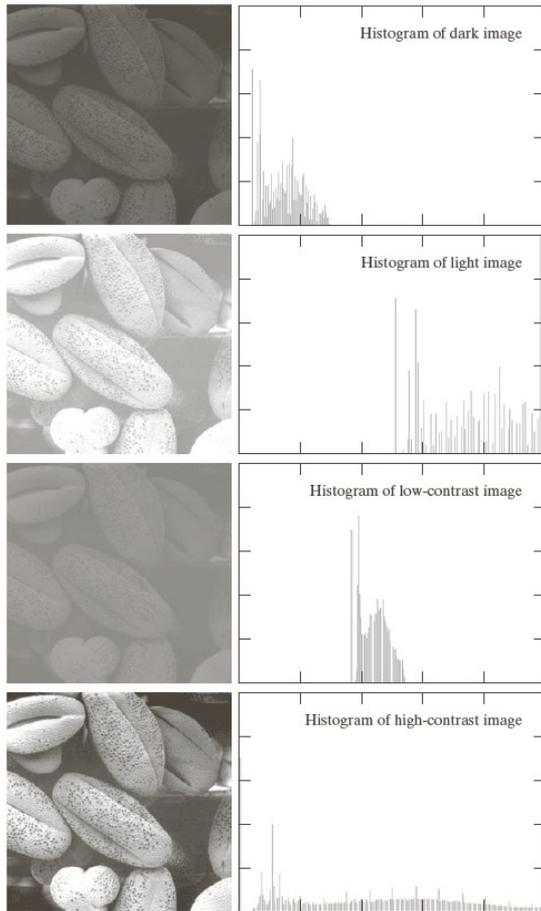


(e)



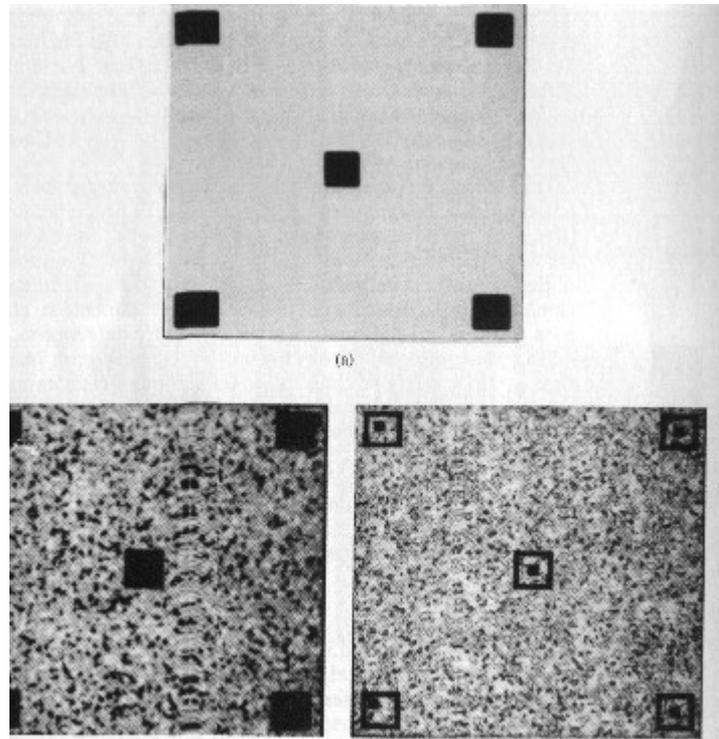
(f)

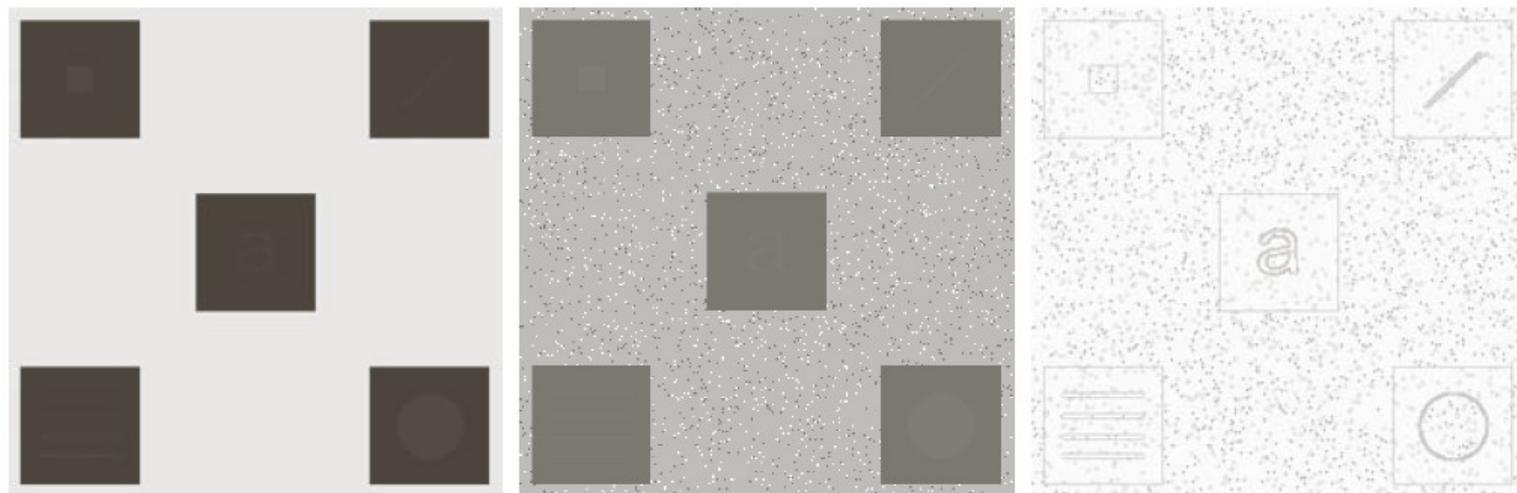
# Ecualización de histograma



# Ecualización local

- Aumenta visualización de detalles
- Vemos detalles pero ojo con el ruido...





a b c

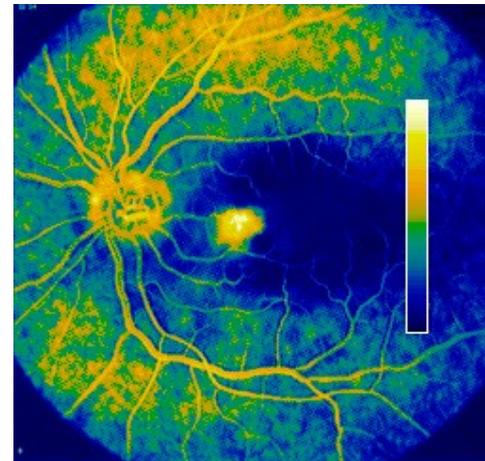
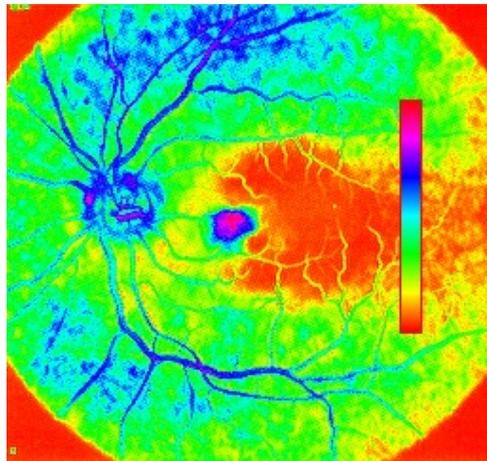
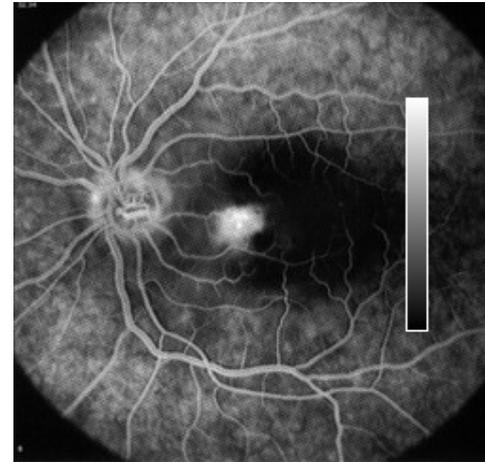
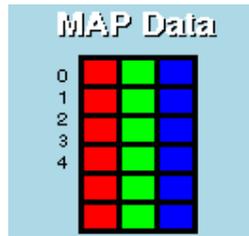
**FIGURE 3.26** (a) Original image. (b) Result of global histogram equalization. (c) Result of local histogram equalization applied to (a), using a neighborhood of size  $3 \times 3$ .

---

# Falso Color y Pseudocolor

- Falso Color : transforma una imagen color en otra imagen color. Se pueden usar mapas de colores que ayuden
  - Ej. Pseudocolor aplicado a una imagen gris.
    - Saturación constante.
    - Niveles de gris se mapean en el brillo.
    - Promedios espaciales en hue.
-

# Falso Color y Pseudocolor

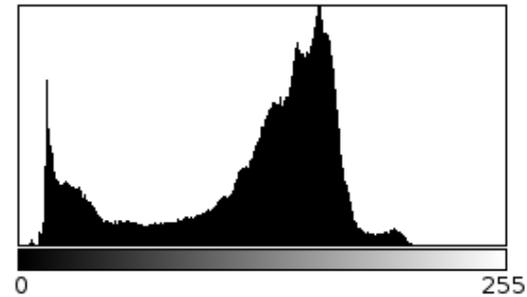


---

ruido

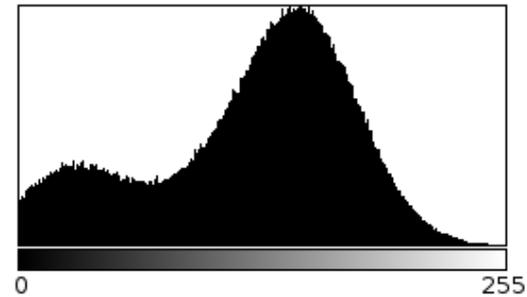
---

# Original



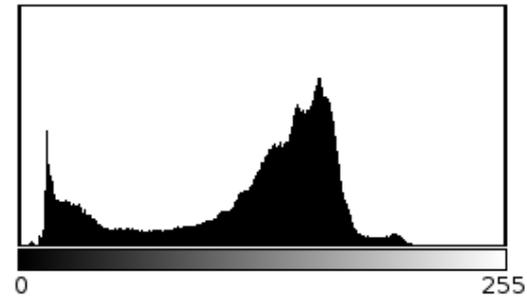
Count: 414720      Min: 3  
Mean: 120.027      Max: 220  
StdDev: 49.188      Mode: 157 (7480)

# Ruido gaussiano, std = 25



Count: 414720    Min: 0  
Mean: 121.186    Max: 255  
StdDev: 52.701    Mode: 152 (3933)

# Ruido sal y pimienta



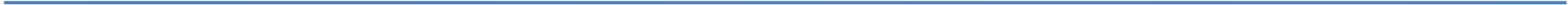
Count: 414720      Min: 0  
Mean: 120.374      Max: 255  
StdDev: 55.674      Mode: 0 (10135)

---

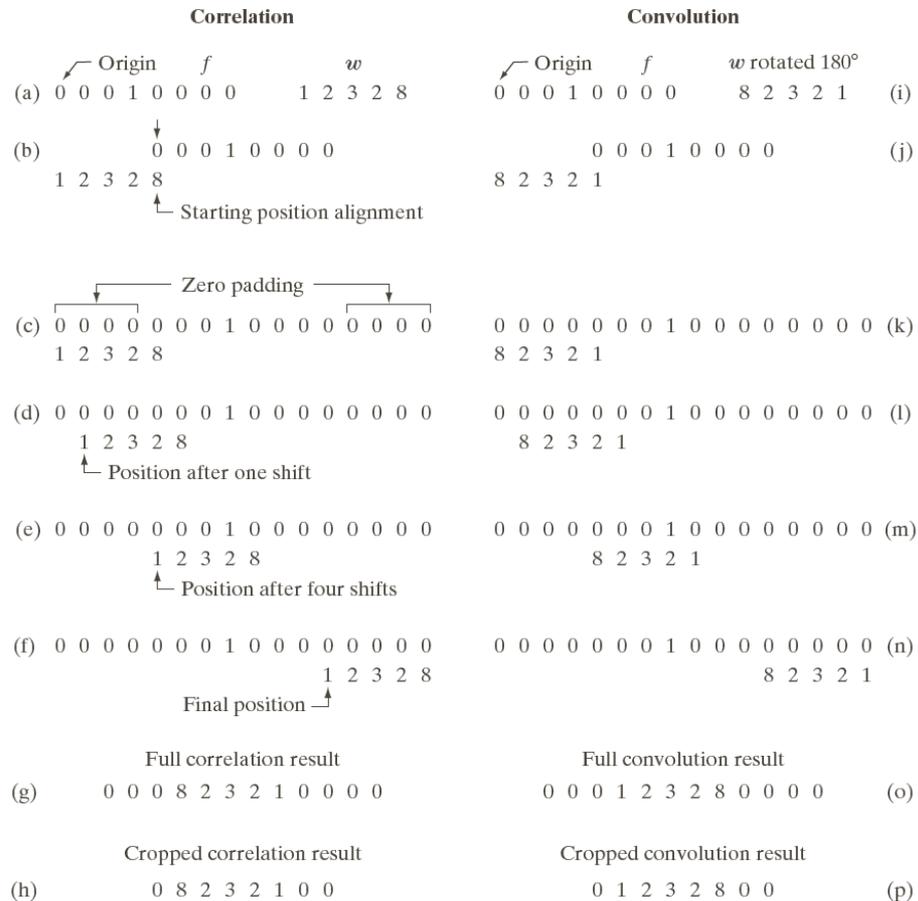
# Filtros Espaciales

- Involucran una vecindad local del pixel procesado.
- Convolución de la imagen original con un filtro caracterizado por su núcleo  $w$  o “mascara espacial”.

$$v(m,n) = \sum_{k,l \in W} w(k,l) y(m-k,n-l)$$

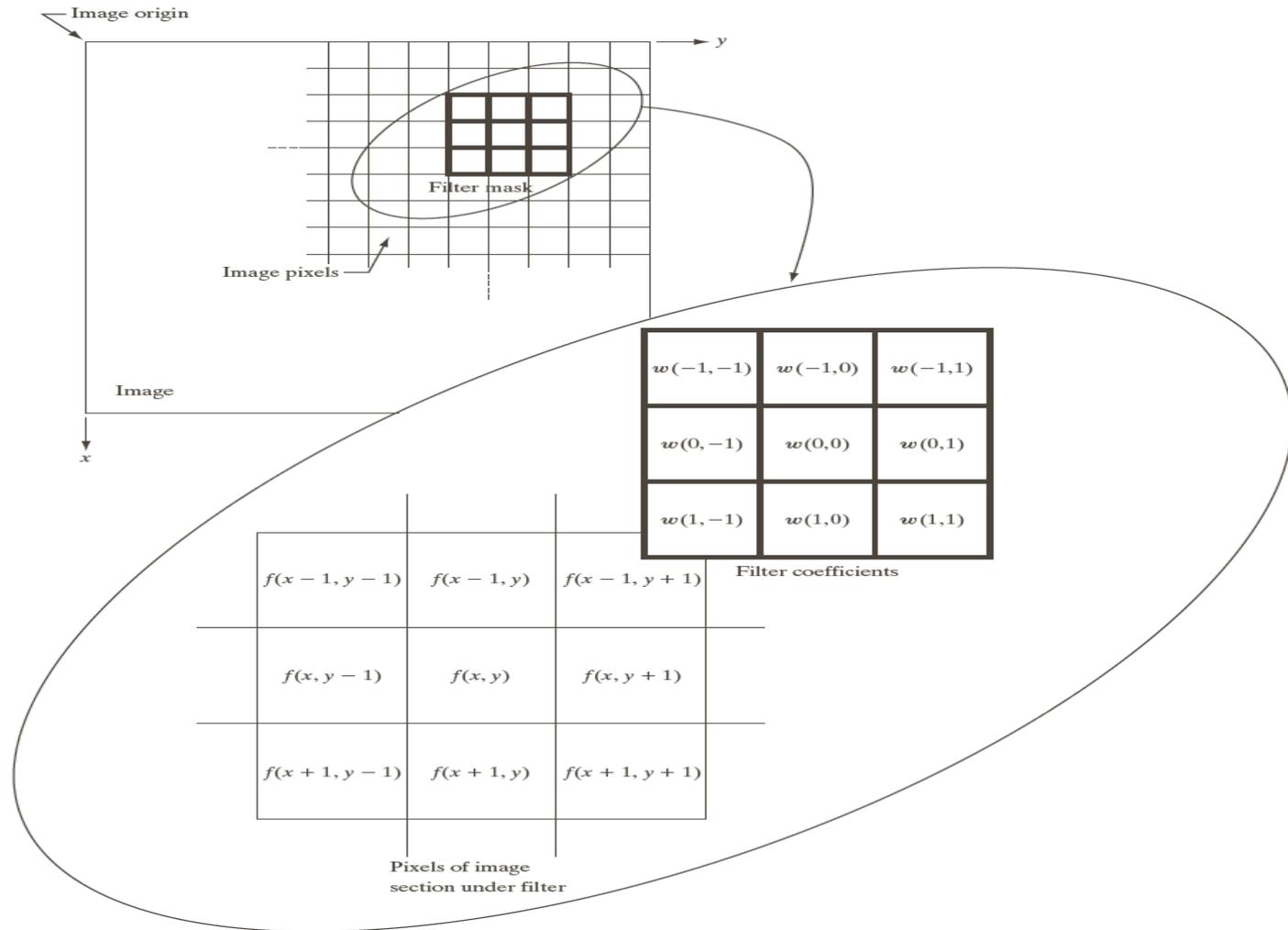


# Convolución vs correlación

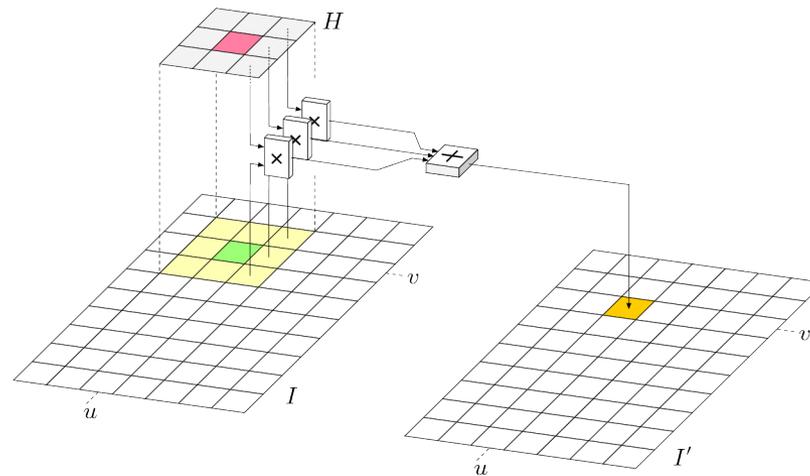


**FIGURE 3.29** Illustration of 1-D correlation and convolution of a filter with a discrete unit impulse. Note that correlation and convolution are functions of *displacement*.

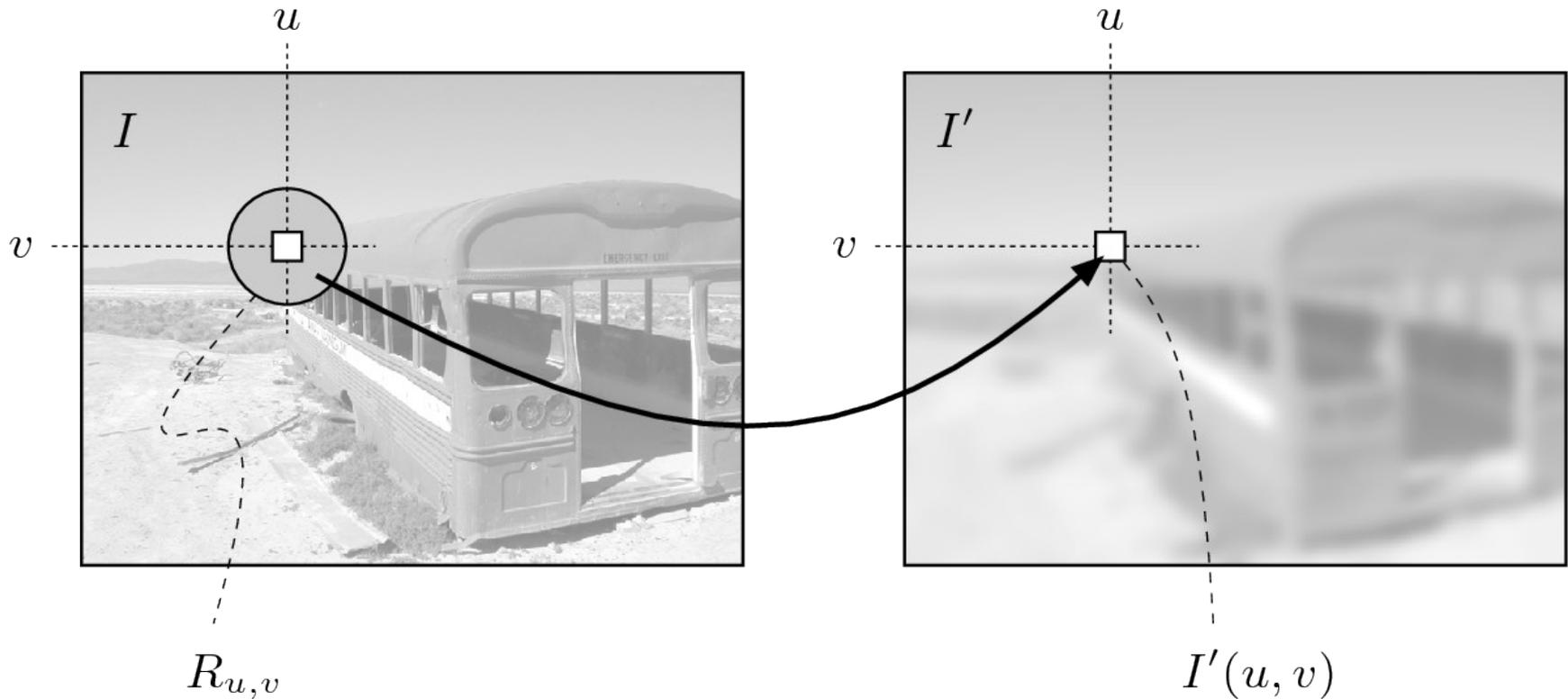




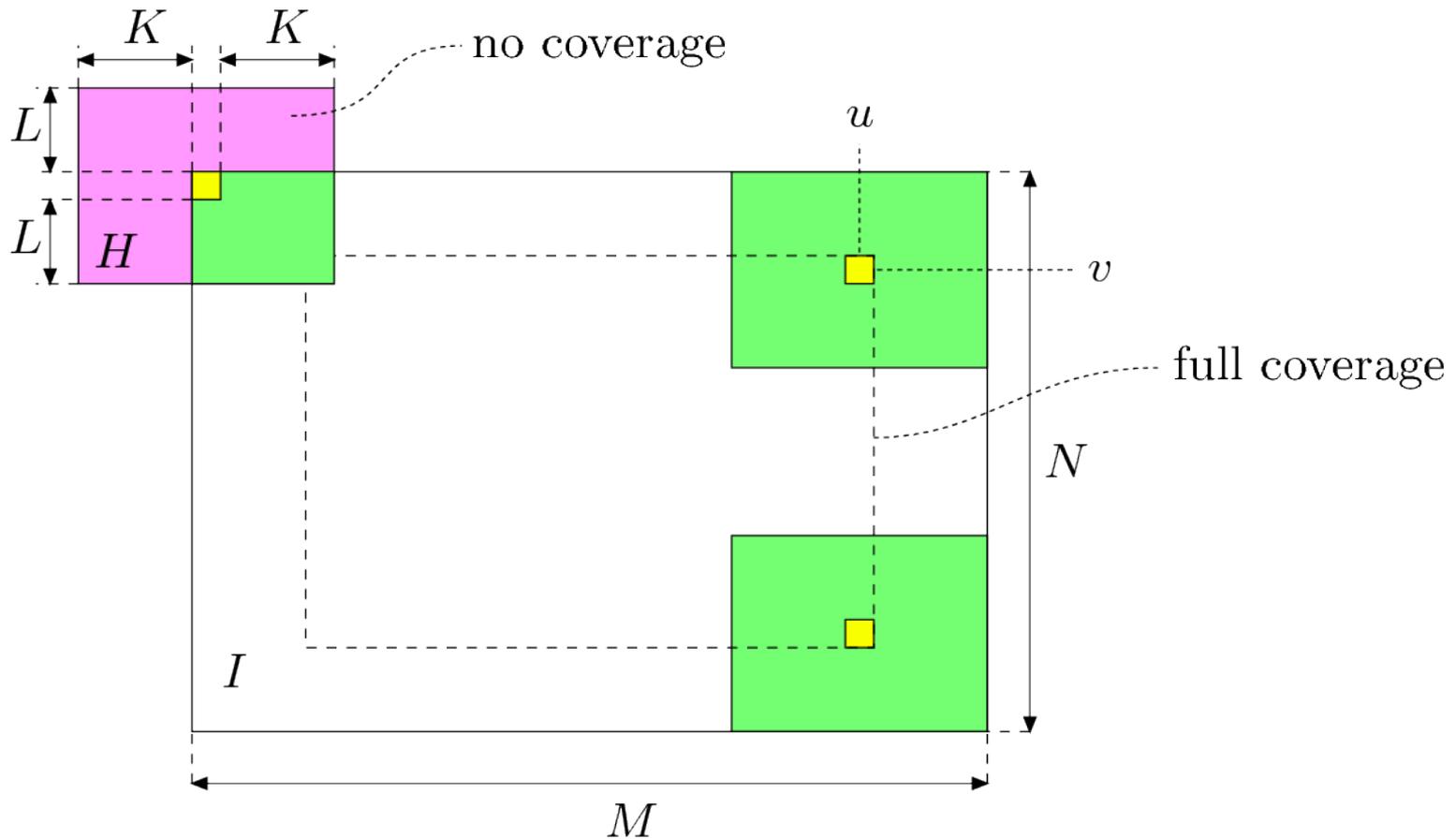
# Operaciones Espaciales



# Operaciones Espaciales



# Qué pasa en los bordes de la imagen?



# Estrategias de bordes



(a)



(b)



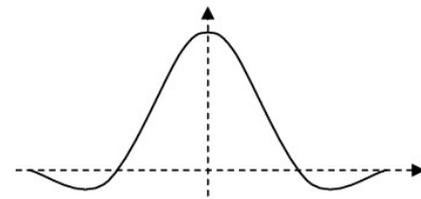
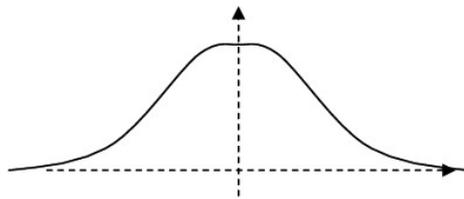
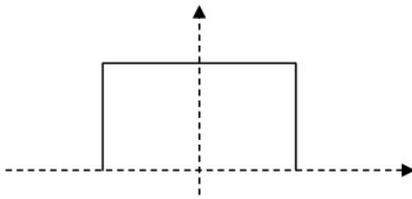
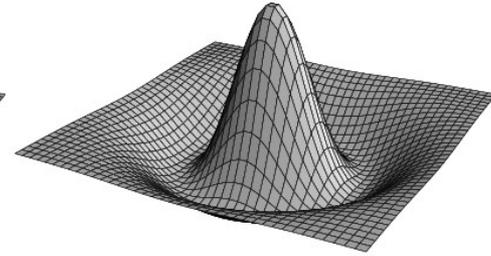
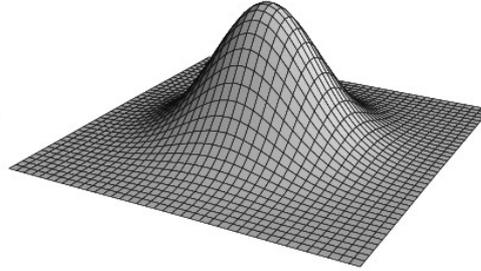
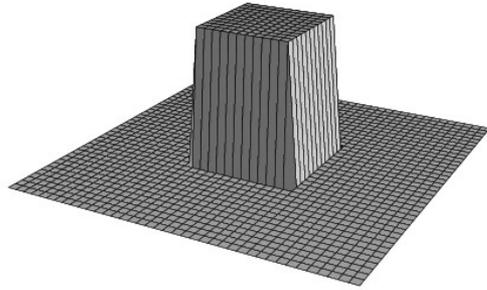
(c)



(d)

Digital Image Processing, An algorithmic introduction using Java. Burger and Burge

# Núcleos de filtros espaciales



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

(a)

0	1	2	1	0
1	3	5	3	1
2	5	9	5	2
1	3	5	3	1
0	1	2	1	0

(b)

0	0	-1	0	0
0	-1	-2	-1	0
-1	-2	16	-2	-1
0	-1	-2	-1	0
0	0	-1	0	0

(c)

---

# Filtrado de Ruido

- En espacio o tiempo!

Hipótesis:

- El tamaño de los pixels es mucho menor que las estructuras de interés
  - El valor de un pixel es parecido al de sus vecinos.
  - Ruido con media nula y varianza sigma cuadrado
-

# Promediado Espacial

1/9

1	1	1
1	1	1
1	1	1

1/16

1	2	1
2	4	2
1	2	1

aprox. gaussiana

$$v(m,n) = 1/N_w \sum_{k \in W} \sum_{l \in W} y(m-k, n-l)$$

Se utiliza :

- suavizado de ruido
- filtrado pasa bajos
- submuestreo de imágenes

# Qué pasa con un punto aislado?

0	0	0	0	0
0	0	0	0	0
0	0	225	0	0
0	0	0	0	0
0	0	0	0	0



0	0	0	0	0
0	25	25	25	0
0	25	25	25	0
0	25	25	25	0
0	0	0	0	0

# Qué pasa con una estructura menor que el orden del filtro?

0	0	225	0	0
0	0	225	0	0
0	0	225	0	0
0	0	225	0	0
0	0	225	0	0



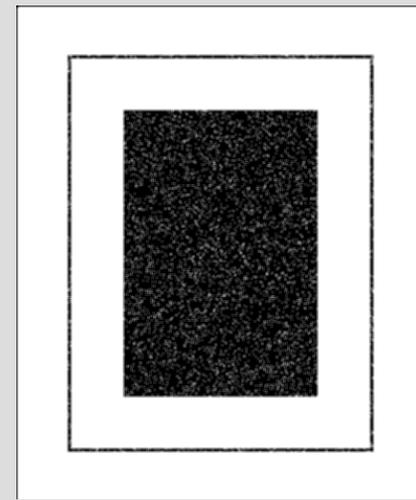
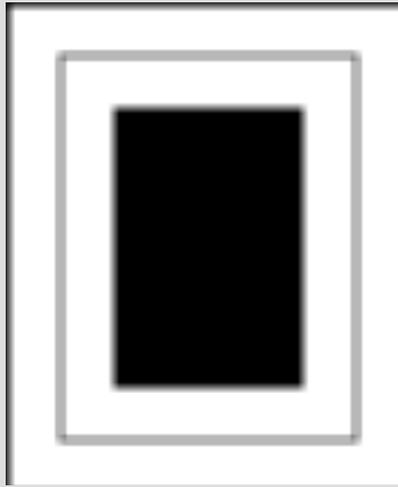
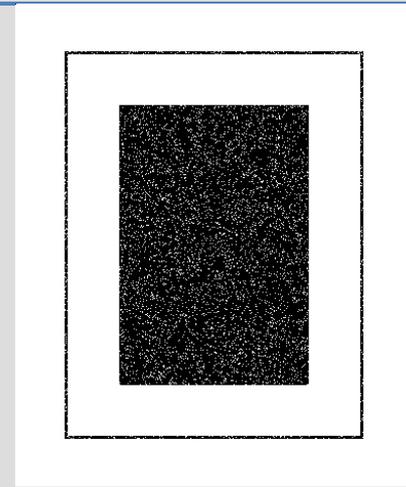
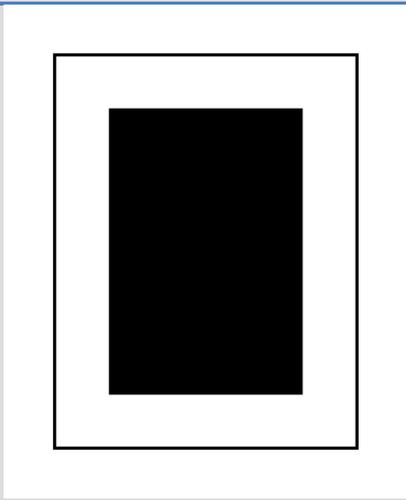
0	25	25	25	0
0	25	25	25	0
0	25	25	25	0
0	25	25	25	0
0	25	25	25	0

# Qué pasa con un borde?

0	0	225	225	225
0	0	225	225	225
0	0	225	225	225
0	0	225	225	225
0	0	225	225	225



0	75	150	225	225
0	75	150	225	225
0	75	150	225	225
0	75	150	225	225
0	75	150	225	225



m5

m3

**FIGURE 3.33** (a) Original image, of size  $500 \times 500$  pixels. (b)–(f) Results of smoothing with square averaging filter masks of sizes  $m = 3, 5, 9, 15,$  and  $35$ , respectively. The black squares at the top are of sizes 3, 5, 9, 15, 25, 35, 45, and 55 pixels, respectively; their borders are 25 pixels apart. The letters at the bottom range in size from 10 to 24 points, in increments of 2 points; the large letter at the top is 60 points. The vertical bars are 5 pixels wide and 100 pixels high; their separation is 20 pixels. The diameter of the circles is 25 pixels, and their borders are 15 pixels apart; their intensity levels range from 0% to 100% black in increments of 20%. The background of the image is 10% black. The noisy rectangles are of size  $50 \times 120$  pixels.

a b  
c d  
e f



---

# Promedio

- Reduce ruido
  - Destruye los bordes
  - Desplaza fronteras
  - Reduce contraste
-

# Filtros no lineales en el dominio espacial

## □ Filtros de ranking

- Máximo:  $u(i, j) = \max\{I(i, j) \text{ t. q. } I(i, j) \in w\}$
  - Mínimo:  $u(i, j) = \min\{I(i, j) \text{ t. q. } I(i, j) \in w\}$
  - Mediana
  - Filtro híbrido
- Tienen mayor complejidad algorítmica!
-

---

# Filtros máximo y mínimo

- ❑ En una imagen de gris con ruido impulsivo de tipo sal y pimienta
  - ❑ Filtro de mínimo: elimina puntos blancos y expande puntos negros
  - ❑ Filtro de máximo: elimina puntos negros y expande puntos blancos
-

# Filtros de máximo y mínimo

Salt & Pepper



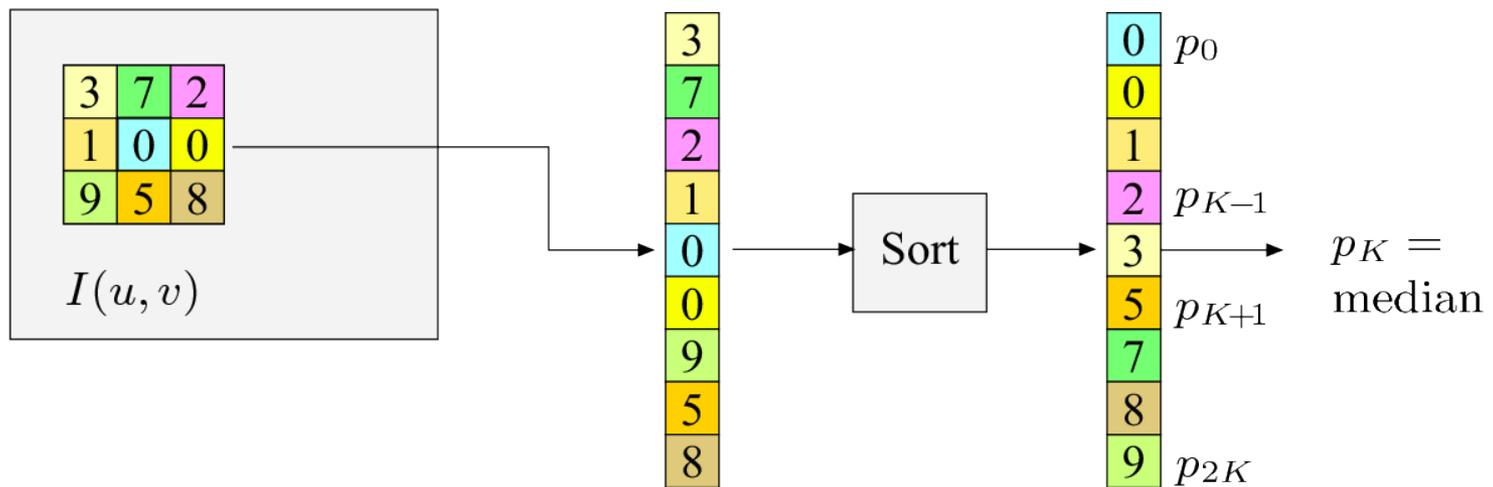
Mínimo



Máximo



# Filtro de mediana



---

# Filtro de mediana

- elimina ruido impulsivo (sal y pimienta), mala performance con ruido gaussiano.
  - no corre los bordes, no reduce diferencia de brillo en cada etapa
  - elimina líneas de ancho menor que el orden de la vecindad
  - redondea esquinas
-

# Qué pasa con un punto aislado?

0	0	0	0	0
0	0	0	0	0
0	0	225	0	0
0	0	0	0	0
0	0	0	0	0



0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

# Qué pasa con una estructura menor que el orden del filtro?

0	0	225	0	0
0	0	225	0	0
0	0	225	0	0
0	0	225	0	0
0	0	225	0	0



0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

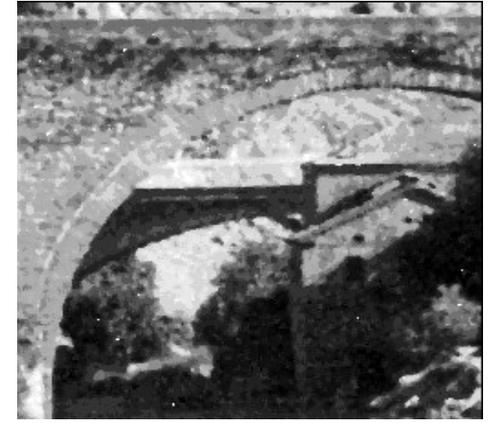
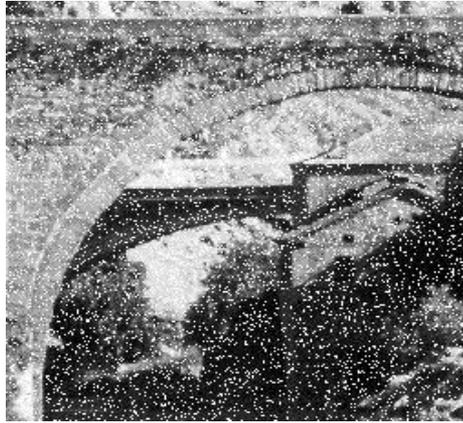
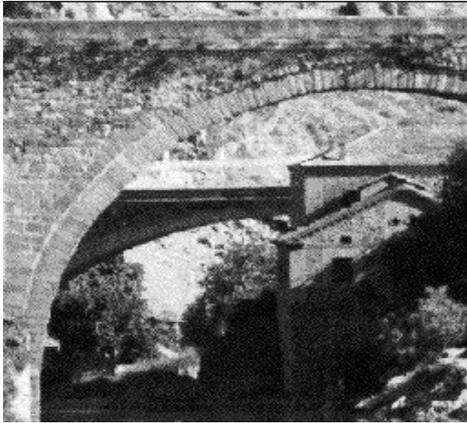
# Qué pasa con un borde?

0	0	225	225	225
0	0	225	225	225
0	0	225	225	225
0	0	225	225	225
0	0	225	225	225

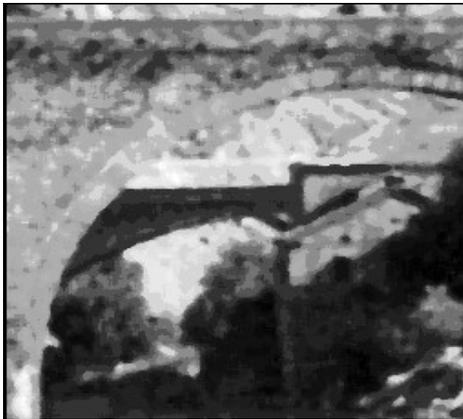


0	0	225	225	225
0	0	225	225	225
0	0	225	225	225
0	0	225	225	225
0	0	225	225	225

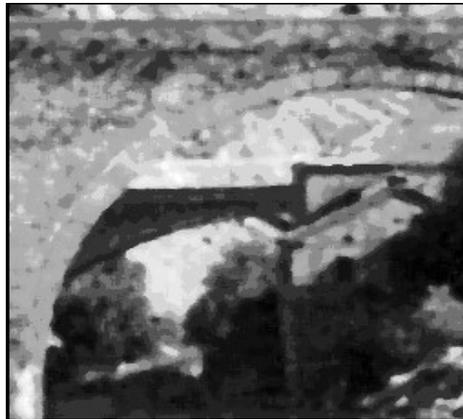
# Filtros de Mediana



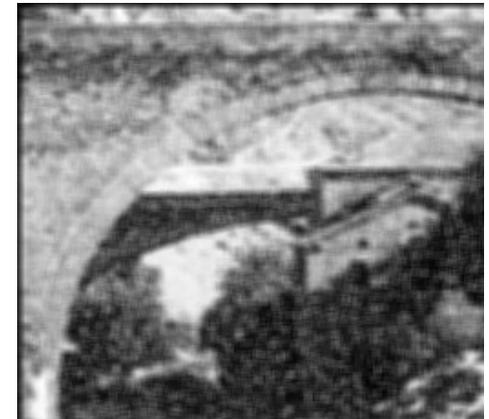
M3



3M3

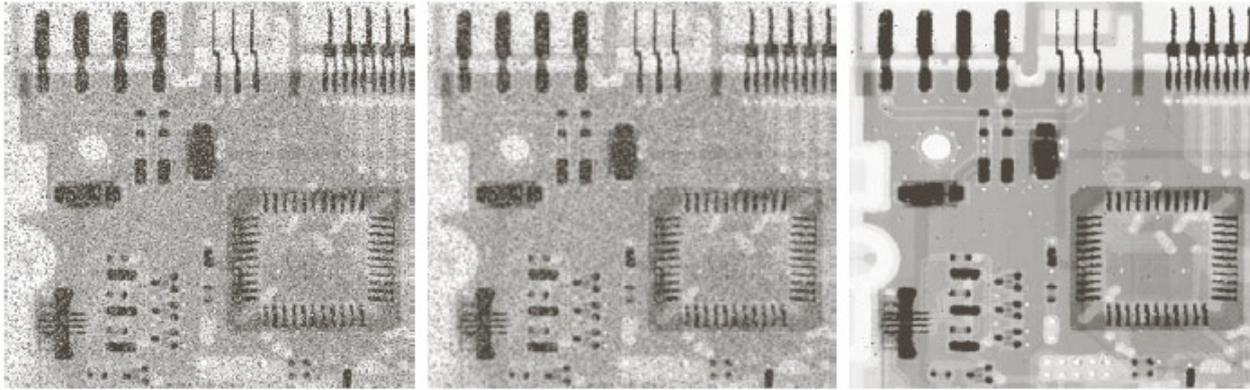


M5



m5

# Filtros de mediana



a b c

**FIGURE 3.35** (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a  $3 \times 3$  averaging mask. (c) Noise reduction with a  $3 \times 3$  median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

# Original.



# Ruido gaussiano std 25.



# Filtro promedio



# Sal y pimienta.



# Filtro de mediana



---

# Filtro híbrido

- La mediana del vecinaje  $V_4$
  - Mediana del vecinaje diagonal  $V_d$
  - Mediana entre los dos anteriores y el punto central.
-

0	0	0
0	1	1
0	1	1

**mediana** →

0	0	0
0	0	1
0	1	1

0	0	0
0	1	1
0	1	1

**híbrido**

mediana V4 [0 0 111]  
 mediana Vd [0 0 011]  
 punto central 1  
 mediana total [0 1 1]

---

# Qué estamos haciendo

- Objetivo: Limpiar ruido de la imagen reteniendo bordes.

$$I_o(x, y) = u(x, y) + \eta(x, y)$$

- Ruido gaussiano, media nula, independiente de la imagen, independiente pixel a pixel e idénticamente distribuido.
-

# Qué estamos haciendo?

- Un pixel observado podemos modelarlo como:  $v(i) = u(i) + \eta(i)$   
 $v(i)$  es el valor observado,  $u(i)$  el valor “real” y  $n(i)$  el ruido.
- Podemos modelar la relación señal a ruido usando la varianza de la señal y del ruido:  $SNR = \frac{\sigma(u)}{\sigma(n)}$   $\sigma(u) = \sqrt{\frac{1}{|I|} \sum_{i \in I} (u(i) - \bar{u})^2}$
- La varianza de la señal se puede medir y la del ruido puede surgir del modelo de ruido o medirse.
- “A review of image denoising algorithm, with a new one” Buades et al.

# Qué estamos haciendo?

- Al aplicar un filtro lineal (media, mediana) estamos reduciendo la varianza del ruido por un factor proporcional al orden del vecinaje.
- Por ejemplo en el caso de un filtro gaussiano  $\text{var}(G*n(0)) \simeq \frac{\varepsilon^2 \sigma^2}{8\pi h^2}$
- Donde  $h$  es el parámetro del filtro (el orden),  $\sigma$  es la varianza del ruido y  $\varepsilon$  es el tamaño del píxel.
- Es decir que la disminución de la varianza del ruido es proporcional al orden.

---

# Qué estamos haciendo?

- En una zona aproximadamente uniforme, promediamos pixeles similares y disminuimos su varianza en relación con el área del operador.
  - Hipótesis de base:
    - los pixeles cercanos son similares
    - Los detalles a conservar son mayores al soporte del filtro.
  - Y si la zona contiene cambios de valor de gris que corresponden a la estructura y no al ruido?
    - Bordes, texturas, etc...
-

---

# Non local means

- Promediar con los píxeles cercanos o con los parecidos?
  - Hacer una estadística de los patches de la imagen (métrica)
  - Promediamos los píxeles que comparten vecinajes (patches) similares.
-

# Non local means



Non Local Means Denoising. Buades et al.

# Original



# Con ruido gaussiano stdv 25



# NLM



# Diferencia



# Original



# Difusión Isotrópica

$$I(x, y, t) = I_0(x, y) * G(x, y, t)$$

- $I_0(x, y)$ : imagen original con ruido
- $G(x, y, t)$ : kernel gaussiano de varianza  $t$
- $I(x, y)$ : imagen filtrada.

## Interpretación:

Al variar el parámetro  $t$  de la gaussiana, la imagen de salida es la observación de la imagen original a una resolución dada.

# Original.



# Difusión Isotrópica $r=2$



# Difusión Isotrópica $r=5$



# Difusión Isotrópica $r=10$



# Difusión Isotrópica $r=15$



---

# Espacio de Escalas

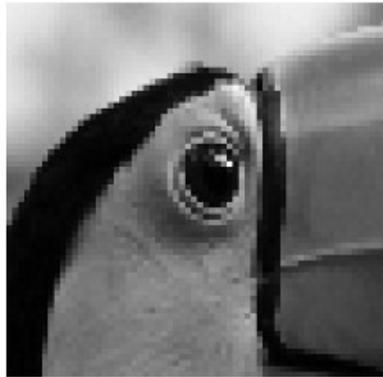
- Procesar la imagen con un banco de filtros con  $t$  creciente.
  - Al aumentar  $t$  : suavizo más, difundo detalles, imagen a escala más gruesa.
  
  - Köenderink y Witkin
-

# interludio

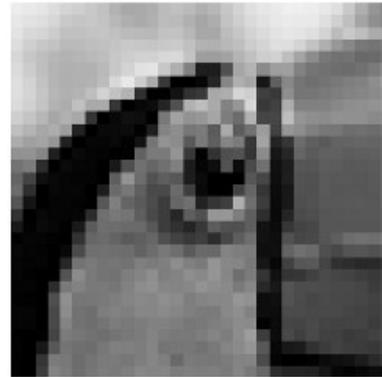
- Subsampling



Original Image



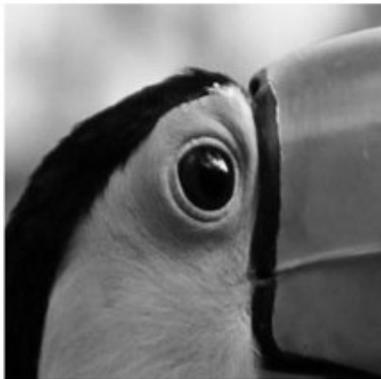
1 / 4 Subsample  
(and Magnified)



1 / 8 Subsample  
(and Magnified)

# interludio

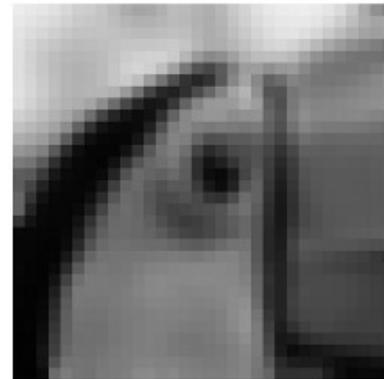
- Filtrado gaussiano y subsampling



Original Image



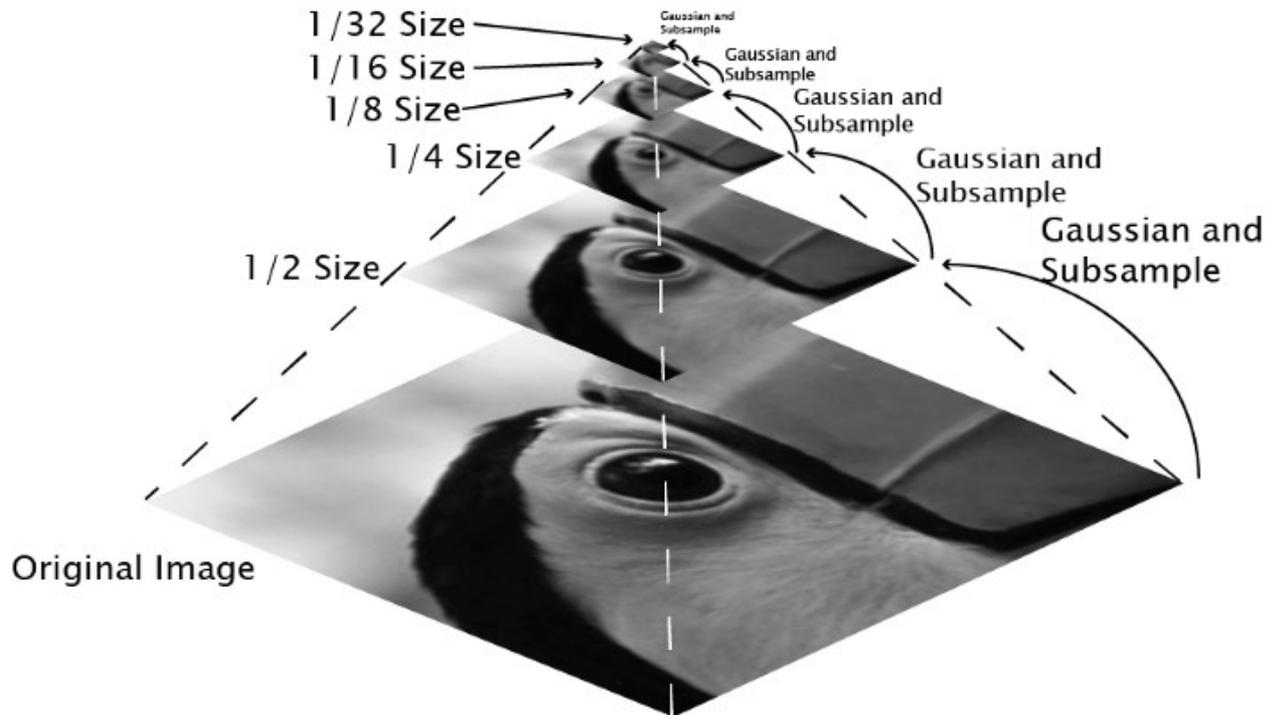
1/4 Subsample  
with Gaussian  
(and Magnified)



1/8 Subsample  
with Gaussian  
(and Magnified)

# interludio

- Pirámide gaussiana: espacio de escalas.



# Ecuación del calor

$$I(x, y, t) = I_o(x, y) * G(x, y, t)$$

**Interpretación:**  $I(x, y, t)$  es una familia paramétrica que se puede obtener como la solución de aplicar la ecuación del calor a la imagen.

$$I_t(x, y, t) = \nabla^2 I(x, y, t) = I_{xx} + I_{yy}$$

$$I(x, y, 0) = I_o(x, y)$$

---

# Ecuación del calor

- Permite resolver el problema de suavizado gaussiano con herramientas conocidas para la resolución de ecuaciones en derivadas parciales.
  - **Nuevo enfoque para la eliminación de ruido:** Encontrar nuevas ecuaciones del calor con coeficiente de conducción variable.
-

# Difusión anisotrópica

- Objetivo:
  - mejorar la imagen sin destruir bordes
  - suavizar dentro de las regiones limitadas por los bordes pero no a través de ellos.
- Ecuación de difusión anisotrópica:

$$I_t(x, y, t) = \nabla \cdot (c(x, y, t) \nabla I(x, y, t))$$

- $c(x, y, t) = \text{cte}$ , difusión isotrópica

# Difusión anisotrópica

$$I_t(x, y, t) = \nabla \cdot (c(x, y, t) \nabla I(x, y, t))$$

- Objetivo:  $c$  debe tener un comportamiento distinto si estoy en una región uniforme o en un borde:
  - $c = 1$  dentro de una región uniforme
  - $c = 0$  en la frontera entre dos regiones uniformes

# Qué función $c$ ?

- Necesito una función sensible a la presencia de fuertes variaciones de intensidad.
- Elección razonable: modulo del gradiente

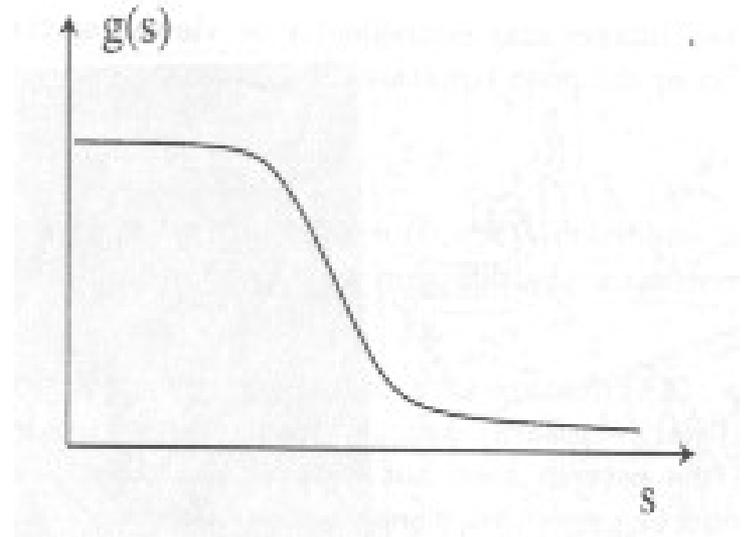
$$c(x, y, t) = g(|\nabla I(x, y, t)|)$$

# Elección del coeficiente

$$c(x, y, t) = g(|\nabla I(x, y, t)|)$$

$$g(s) \rightarrow 0 \text{ cuando } s \rightarrow \infty$$

$$g(s) \rightarrow 1 \text{ cuando } s \rightarrow 0$$

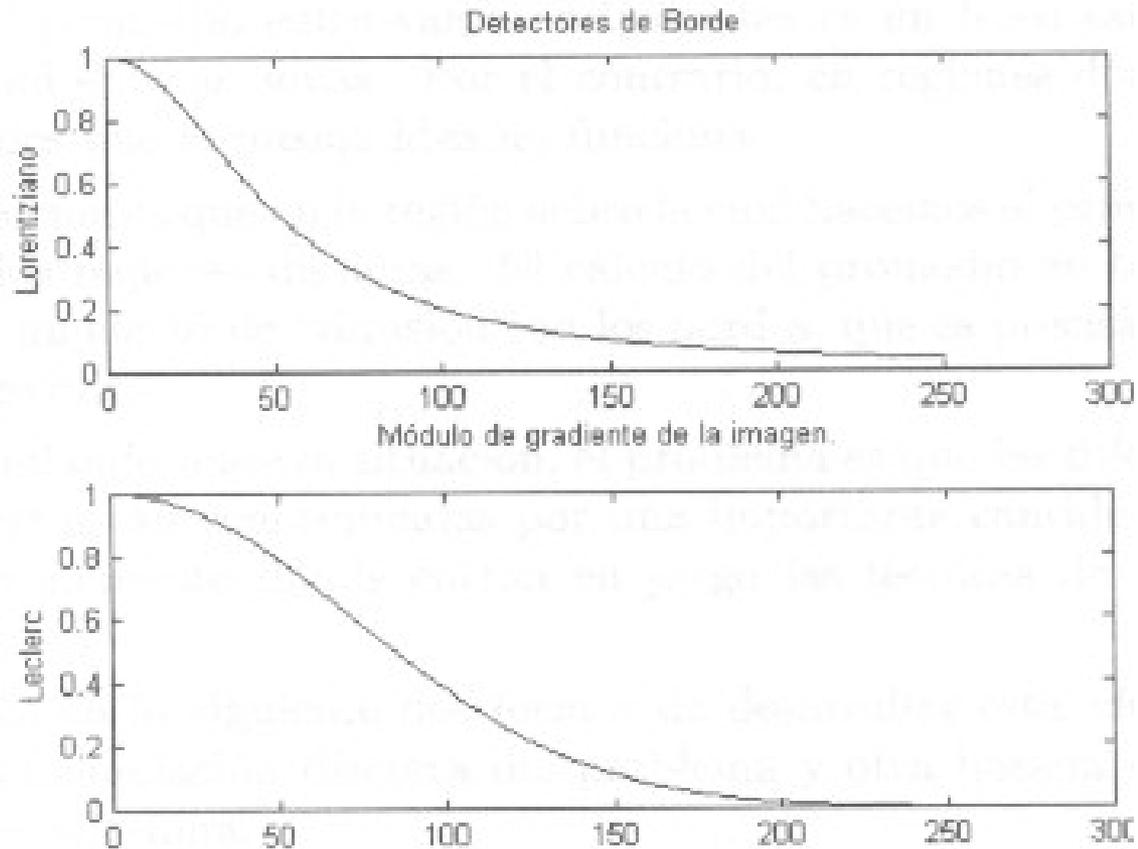


# Funciones del gradiente

- Detector de Leclerc:  $g(|\nabla I|) = e^{-|\nabla I|^2/k^2}$
- 
- Detector Lorentziano  $g(|\nabla I|) = 1/(1+|\nabla I|^2/k^2)$

-

# Funciones del gradiente



# Imagen original



# Difusión anisotrópica 10 iteraciones



# Difusión anisotrópica 20 iteraciones



# Difusión anisotrópica 40 iteraciones



# Original con ruido gaussiano std=25



# Difusión anisotrópica, 20 iteraciones.



# Original con ruido de sal y pimienta



# Difusión anisotrópica 20 iteraciones



# Discretización para la ecuación de difusión anisotrópica

$$I_{ij}^{t+1} = i_{ij}^t + \lambda (c_N D_N I + c_S D_S I + c_E D_E I + c_O D_O I)_{ij}^t$$

$$D_N = I_{i-1,j} - I_{i,j}$$

$$c_{N_{ij}} = g(|D_N I_{ij}^t|)$$