

# Segmentación

# Suavizado?

- Se suaviza previamente con una gaussiana.

$$G(x) \approx e^{-\|x\|^2 / 2\sigma^2}$$

$\sigma$ : desviación standard

- Se puede convolucionar con la derivada de la gaussiana:

$$\nabla(I \star G_\sigma) = I \star \nabla G_\sigma$$

$$\nabla^2(I \star G_\sigma) = I \star \nabla^2 G_\sigma$$

# Modelo lineal

- Hildreth, Marr, Witkin, Koenderink...
- Convolucionar la imagen con “detectores locales de borde”, detectar, filtrar y agrupar las respuestas.
  - Cambio de signo del laplaciano.
  - Extremas del gradiente.

# Ecuación del calor

- Koenderink propuso la ecuación del calor como una forma de análisis multiescala:

$$I_t(x, y, t) = \Delta I(x, y, t)$$

$$I(x, y, 0) = I_0(x, y)$$

- La solución es la convolución con gaussianas de sigma variable.

$$I_t(x, y, t) = G_t \star I_0 \quad G_t = \frac{1}{4\pi t} e^{-\frac{(x^2+y^2)}{4t}}$$

# Difusión Isotrópica

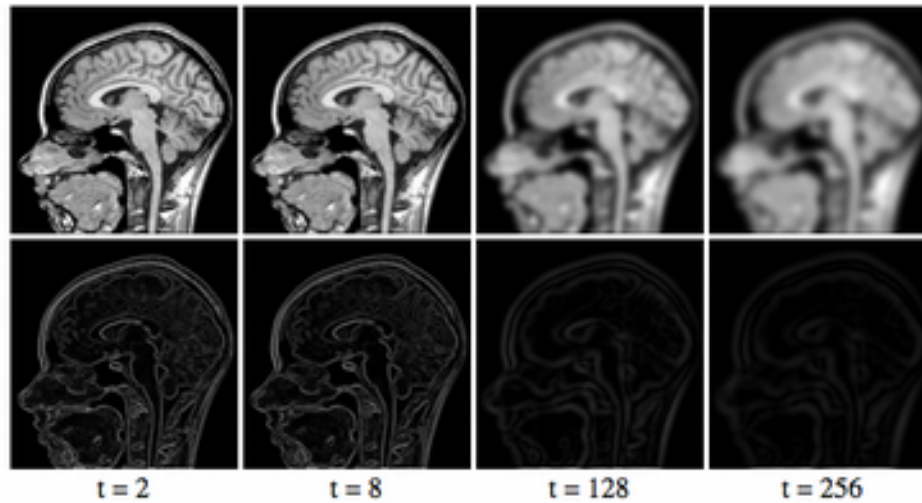
$$I(x, y, t) = I_0(x, y) * G(x, y, t)$$

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

## Interpretación:

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

# Difusión Isotrópica



# 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

# Difusión isotrópica





# Gaussiano $\sigma=2.0$



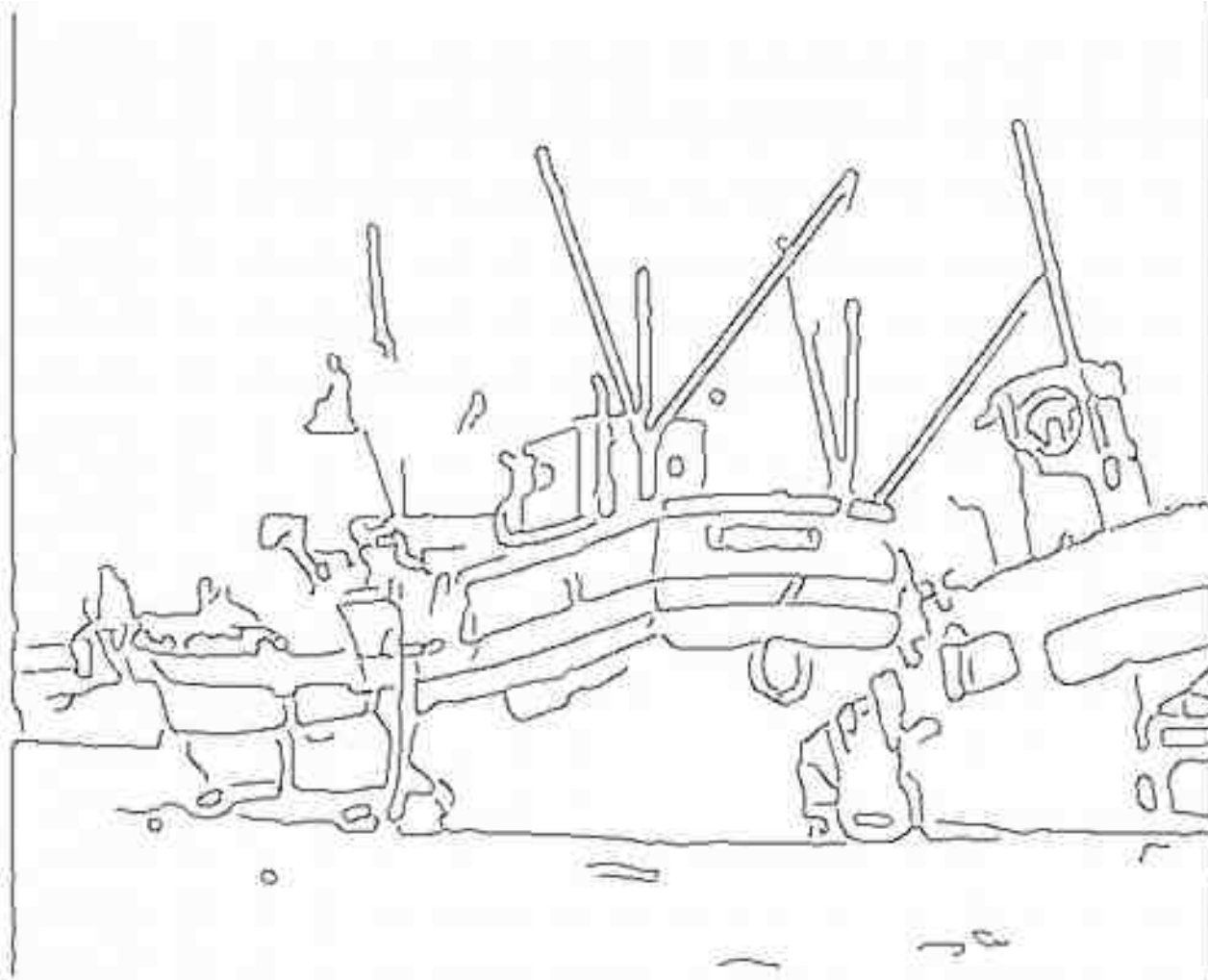
Gaussiano  $\sigma=5.0$



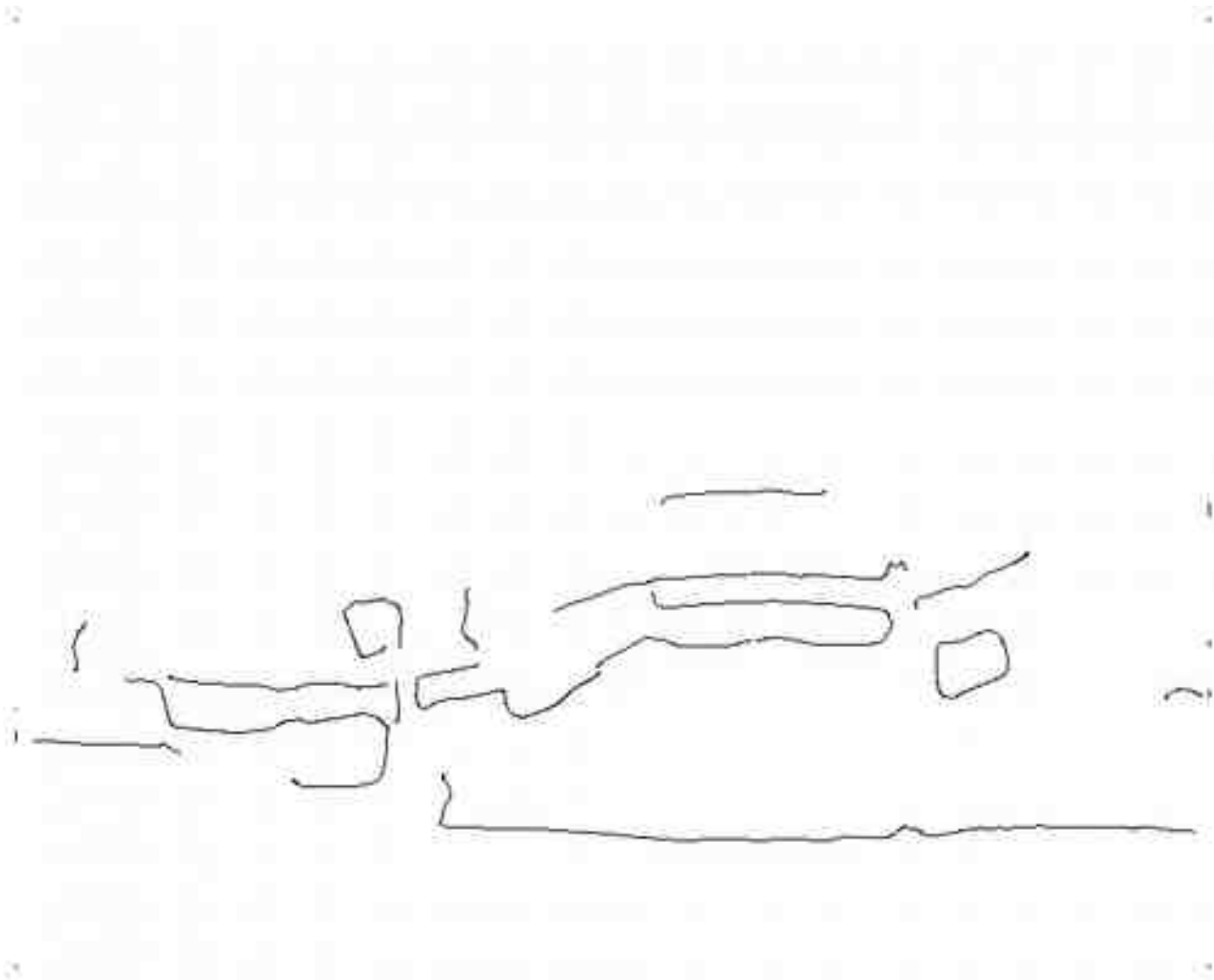
Gaussiano  $\sigma=10.0$



# Canny sobre Gaussiana $\sigma=2.0$



# Canny sobre Gaussiana $\sigma=5.0$



# Canny sobre Gaussiana $\sigma=10.0$



# Análisis multiescala

- Dada una imagen  $u_0(x)$  definida en un dominio  $\Omega$ .
- Generar una secuencia de imágenes simplificadas  $u_\lambda(x)$  cada vez más groseras conforme aumenta  $\lambda$ .
- Los detalles y bordes permanecen si su “escala” es mayor que  $\lambda$

# Propiedades que debe cumplir

- Fidelidad:  $u_\lambda(x) \rightarrow u_0(x)$  cuando  $\lambda \rightarrow 0$
- Causalidad:  $s_\lambda(u_0)$  solo depende de  $s_{\lambda^1}(u_0)$  si  $\lambda > \lambda^1$
- Invariancia euclidiana: Si  $A$  es una isometría:  $s_\lambda(u_0 \circ A) = s_\lambda(u_0) \circ A$
- Causalidad fuerte (para detección de bordes):  $K_\lambda \subset K_{\lambda^1}$  si  $\lambda > \lambda^1$



# Ecuación del calor

**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.

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

$$I_0(x, y) = I(x, y, 0)$$

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

# Ecuación del calor

- Permite resolver el problema de suavizado gaussiano con herramientas conocidas para la resolución de ecuaciones en derivadas parciales.

# Y los criterios?

- En general los modelos lineales cumplen los criterios 1, 2 y 3 del análisis multiescala pero fallan en el 4to criterio: la causalidad fuerte.
- Basta mirar que a diferentes escalas los bordes “se mueven”.
- Esto es cierto incluso para un filtro “óptimo” en el sentido de Canny.

# Imagen original



# Mediana de radio 2



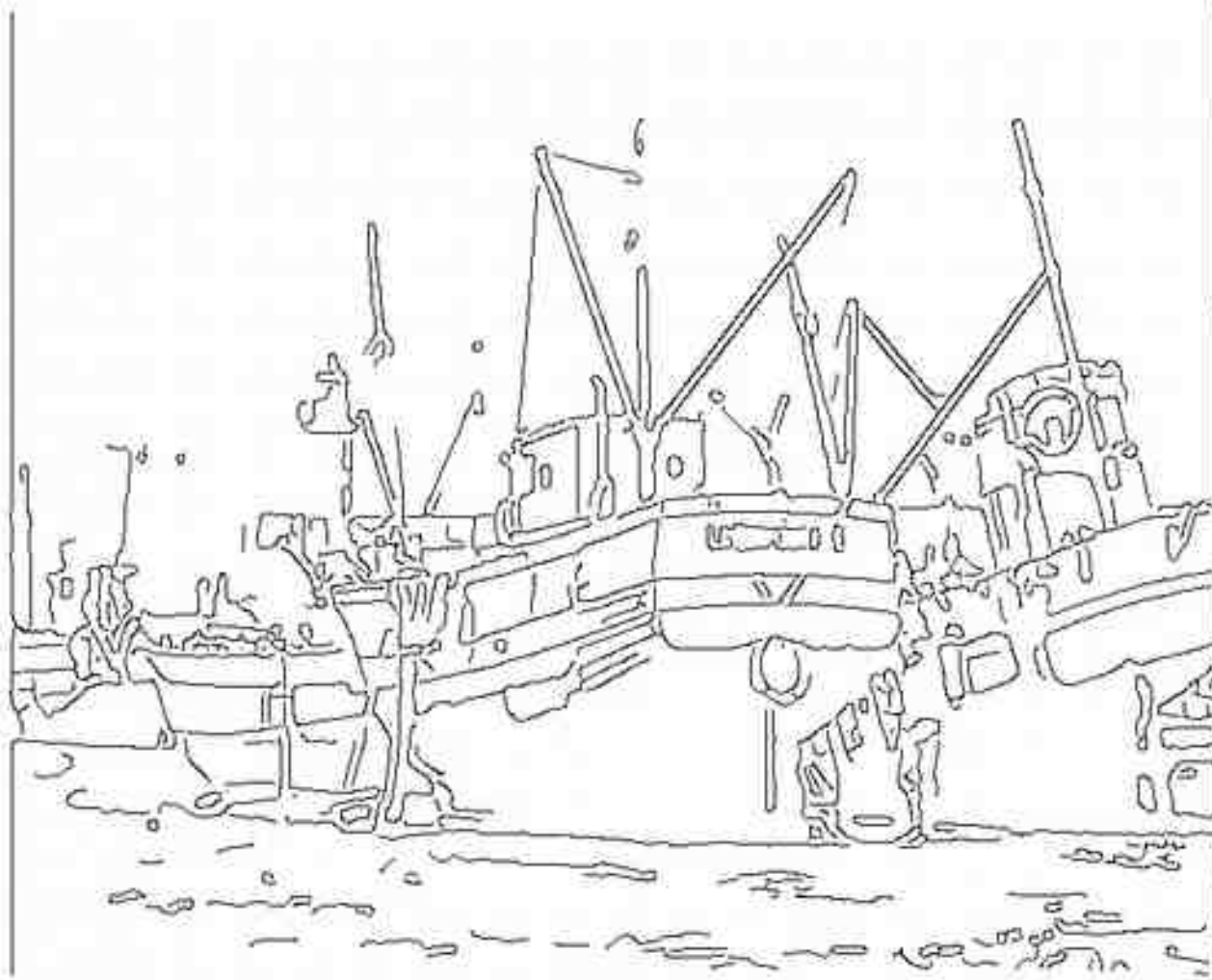
# Mediana de radio 5



# Mediana de radio 10

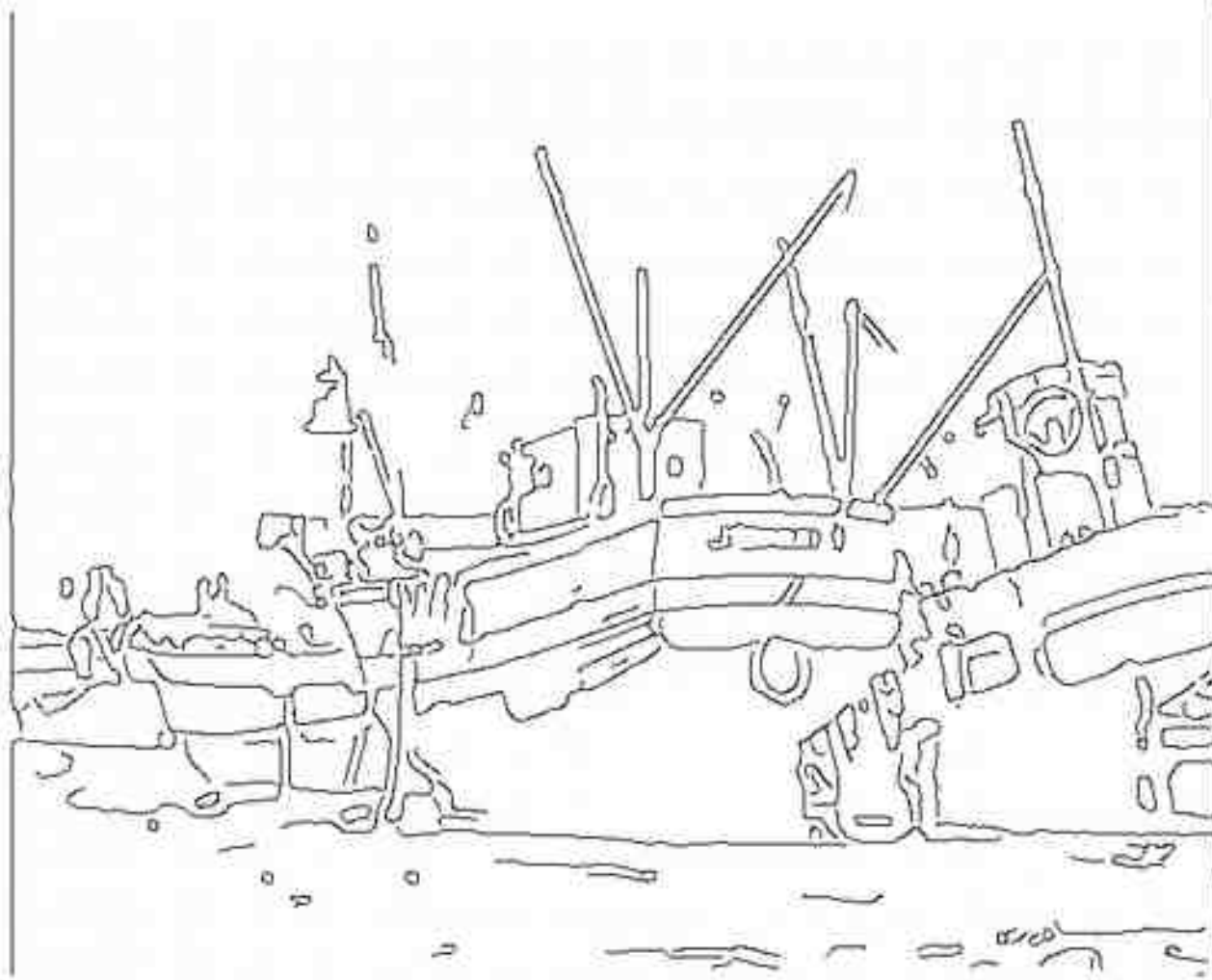


# Canny sobre imagen original

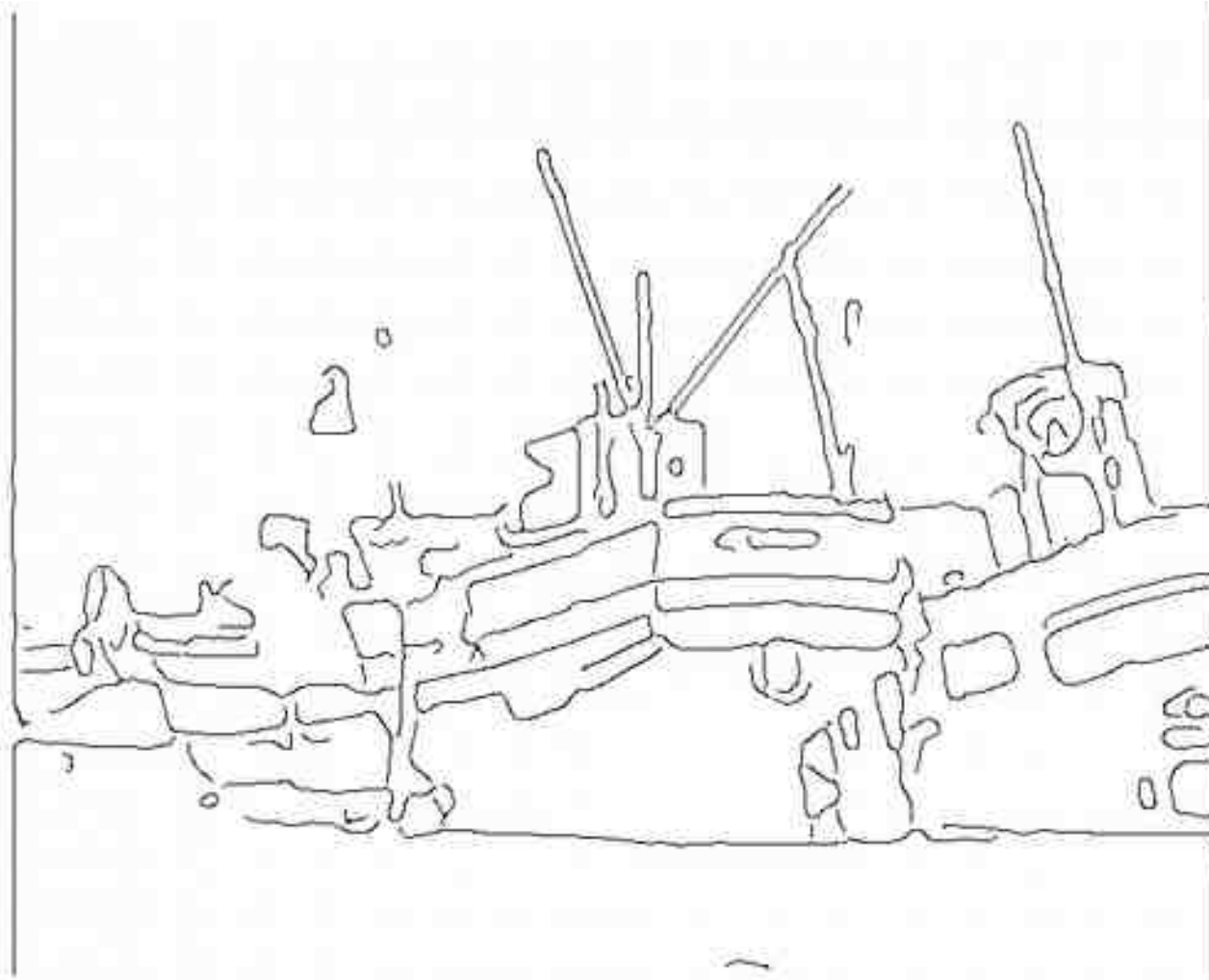




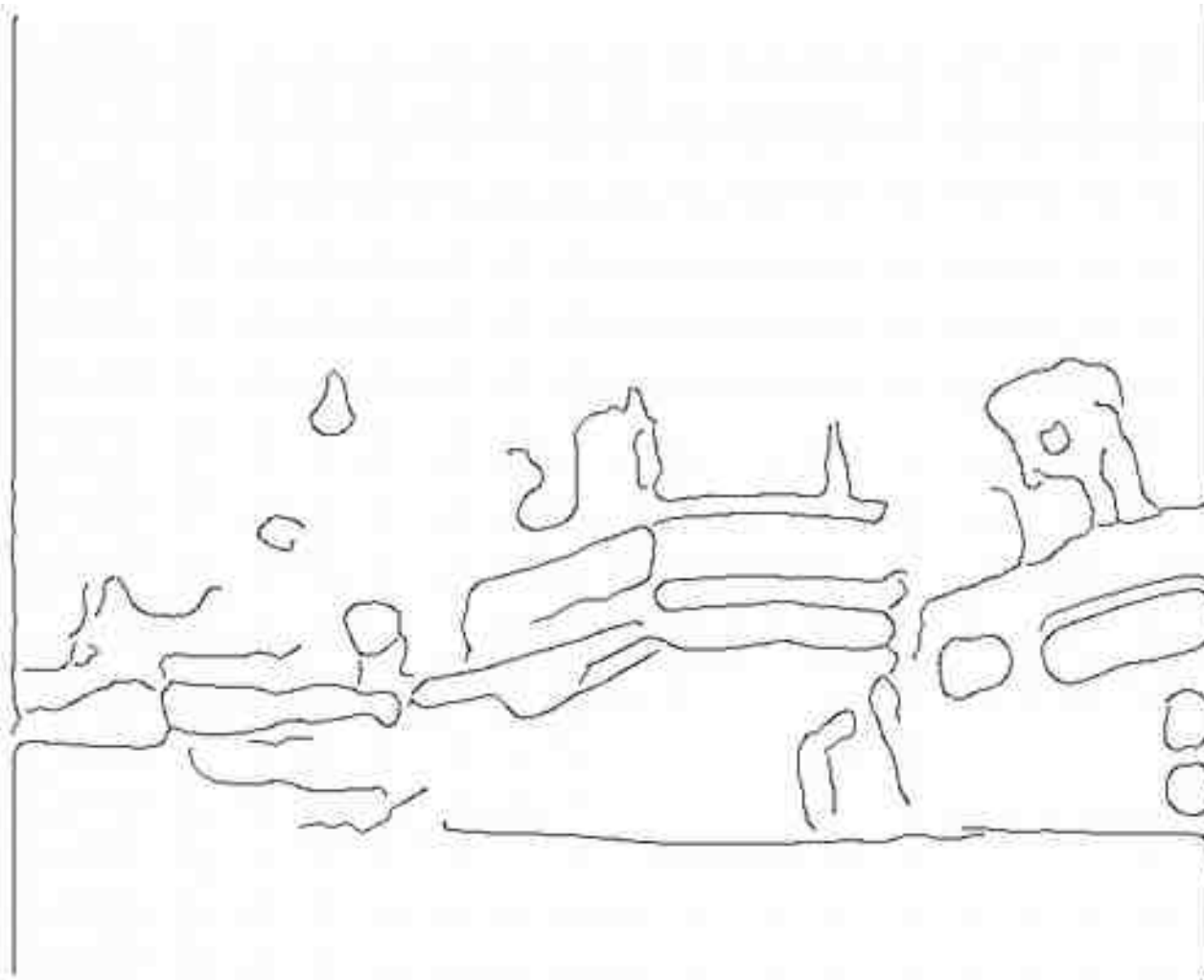
# Canny sobre mediana de $r=2$



# Canny sobre mediana de $r=5$



# Canny sobre mediana de $r=10$



# Difusión anisotrópica

- Nuevo enfoque para la eliminación de ruido y por tanto para la detección de fronteras:  
Encontrar “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: Comportamiento distinto si estoy en una región uniforme o en un borde:

$c = 1$  dentro de la región

$c = 0$  en los bordes

# Estimación de bordes

- **Problema:** No conozco la posición de los bordes a cada escala. Necesito estimar.
- Necesito un descriptor de la presencia de un borde.
- Elección razonable: módulo del gradiente

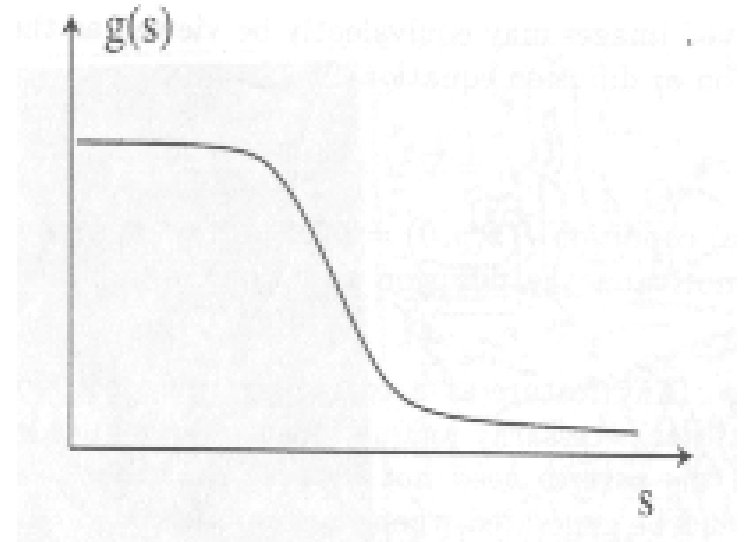
$$C(x, y, t) = g(|\nabla I(x, y, t)|)$$

# Elección del coeficiente

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

$$s \rightarrow \infty \quad s \rightarrow 0$$

$$g(s) \rightarrow 0 \quad g(s) \rightarrow 1$$

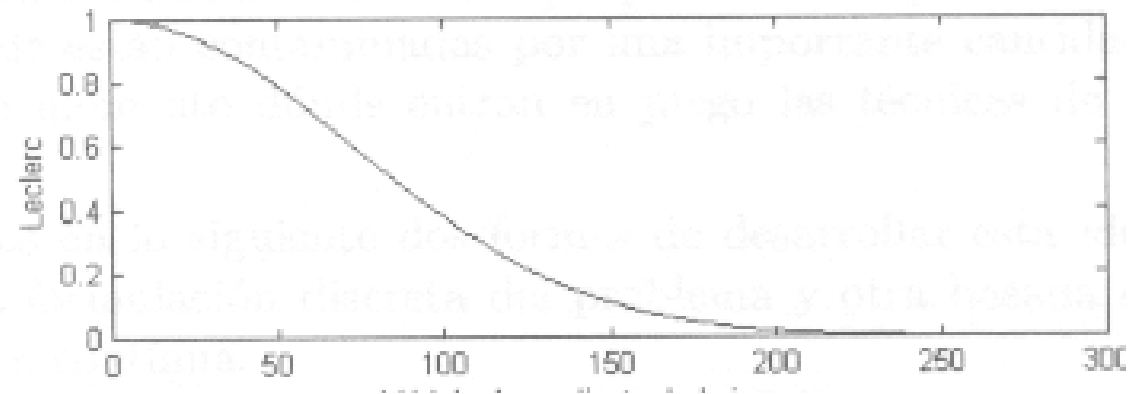
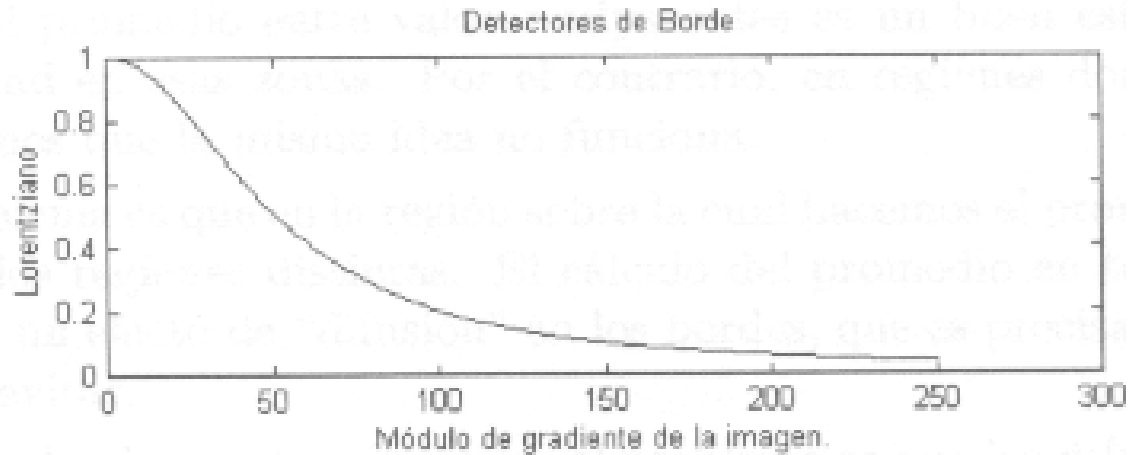




# Detectores de bordes

- Detector de Leclerc:  $g(|\nabla I|) = e^{-\frac{|\nabla I|^2}{k^2}}$ 
  - Privilegia bordes con alto contraste...
- Detector Lorentziano:  $g(|\nabla I|) = \frac{1}{1 + \frac{|\nabla I|^2}{k^2}}$ 
  - Privilegia grandes regiones sobre las pequeñas...

# Detectores de bordes



# Y los parámetros?

- La función  $g$  es una especie de umbral que antes estaba en el parámetro de qué tan fuerte es el módulo del gradiente...
- La escala está dada por el tiempo que deajo evolucionar la difusión.

# Imagen original



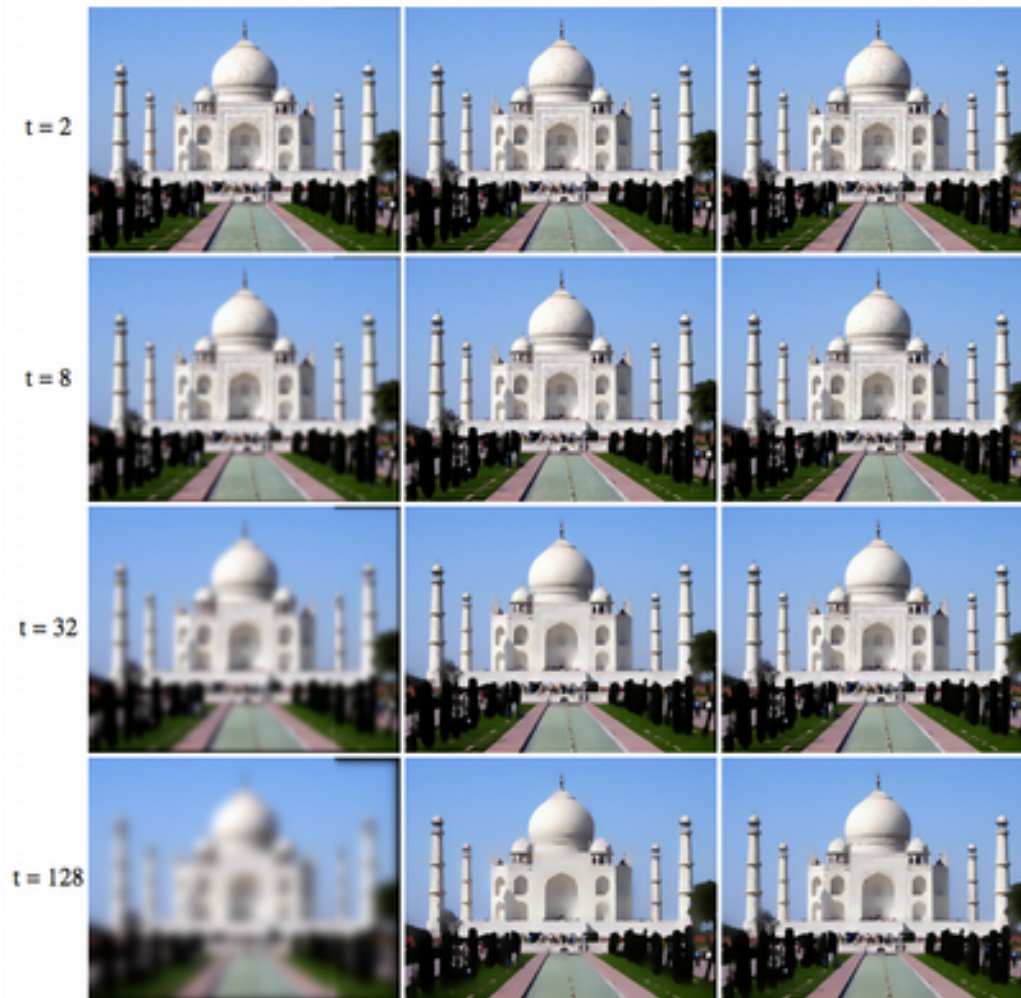
# Diffusión Anisotrópica $t=20$



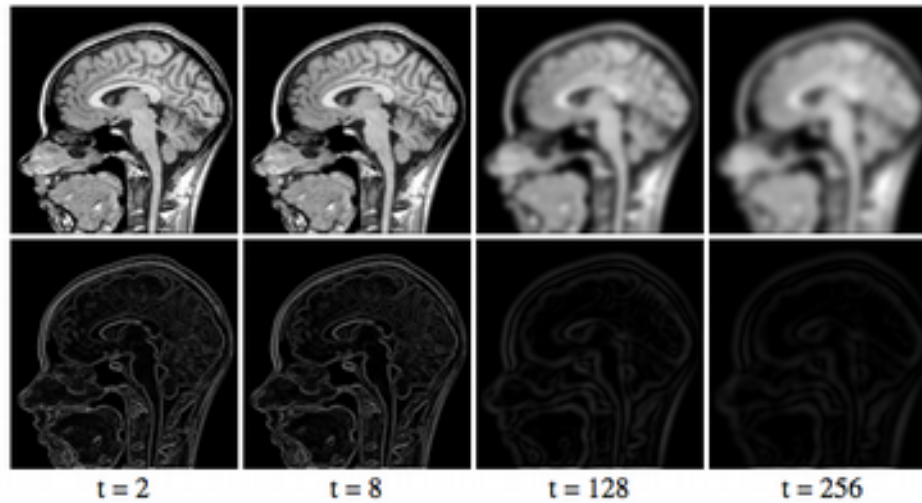
# Diffusión Anisotrópica $t=40$



# Diffusion isotrópica vs anisotrópica

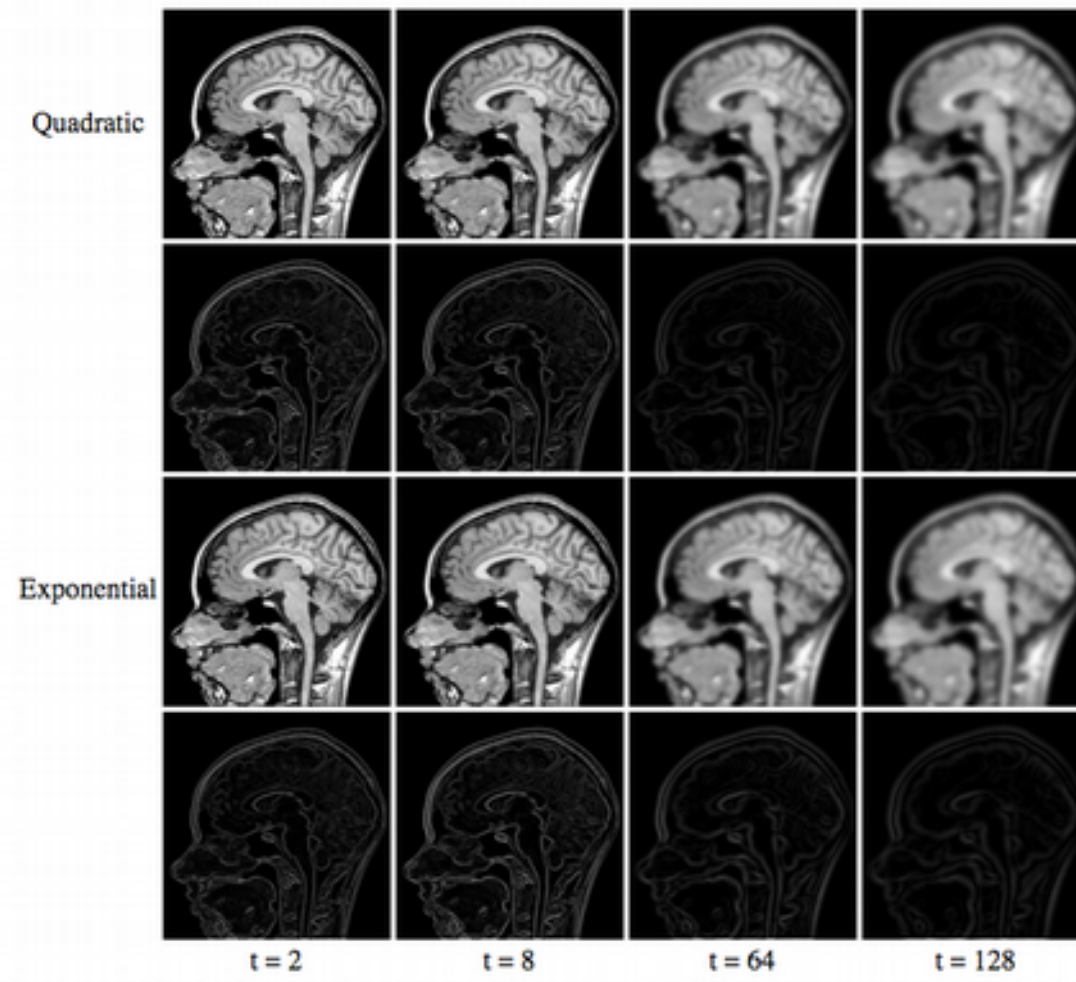


# Diffusion isotrópica vs anisotrópica





# Diffusion anisotrópica



# Discretización para la ecuación de difusión anisotrópica

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

$$D_N I_{ij} = I_{i-1,j} - I_{i,j} \quad c_{N_{ij}} = g\left(\left|D_N I_{ij}^t\right|\right)$$

# Información global/local

- En ocasiones la información debe ser detectada utilizando un operador global: toda la imagen aporta a la detección.
- La Transformada Hough es un ejemplo de operador global.

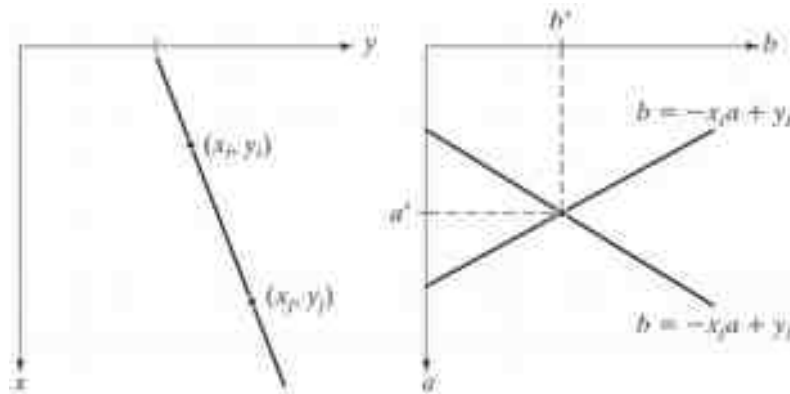
# Transformada Hough.

- Obtener el campo de gradientes de la imagen.
- Crear un espacio de acumulación en función de los parámetros de la función que se busca.
- Los máximos en el espacio de acumulación señalan la existencia de los objetos buscados.

# Transformada Hough. Líneas.

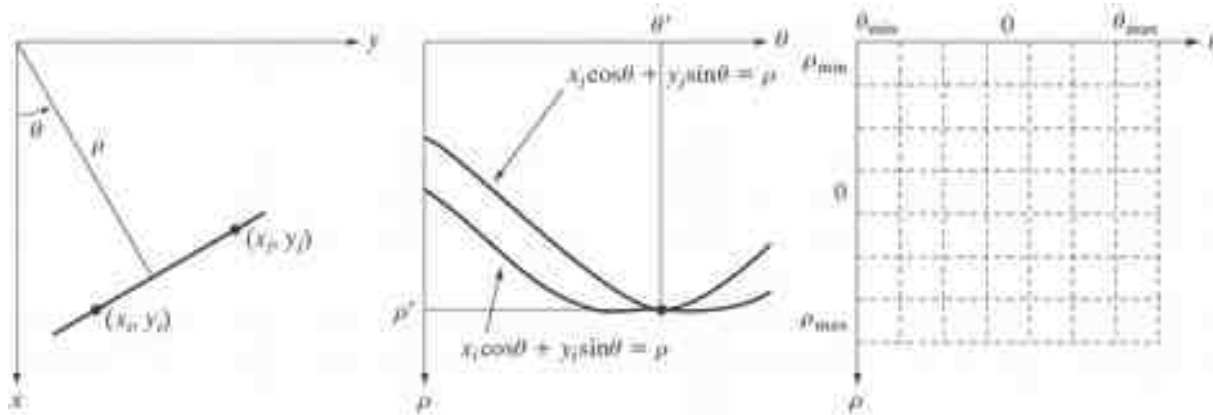
- Una línea es definida como:  $\rho = x \cos(\theta) + y \sin(\theta)$
- La transformada Hough de esa línea es un punto en el plano  $(\rho, \theta)$
- Discretizamos el espacio y para cada punto de la imagen calculamos su representación en el plano  $(\rho, \theta)$
- Los máximos locales en  $(\rho, \theta)$  son líneas.

# Transformada Hough. Líneas.



(a) (b)  
**FIGURE 10.31**  
(a)  $xy$ -plane.  
(b) Parameter space.

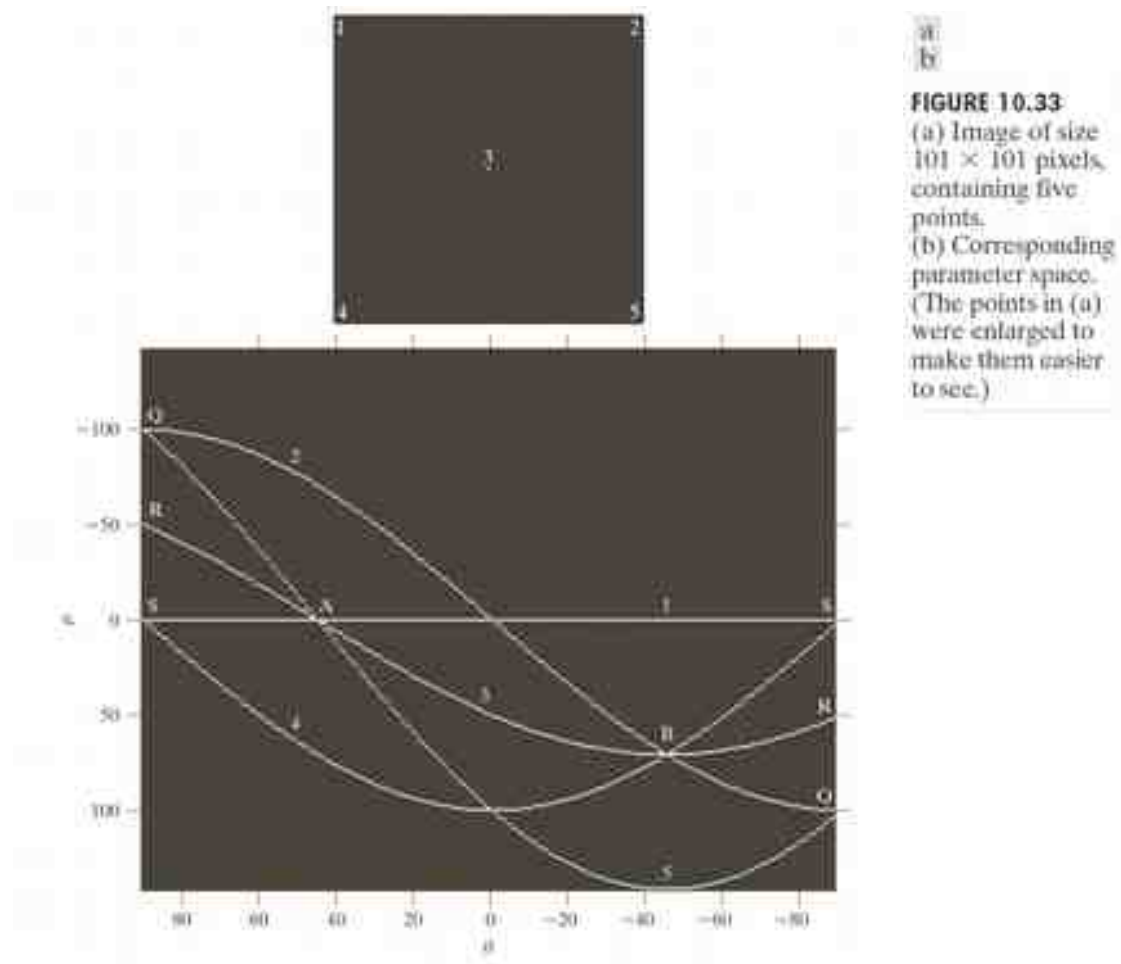
# Transformada Hough. Líneas.



a b c

**FIGURE 10.32** (a)  $(\rho, \theta)$  parameterization of line in the  $xy$ -plane. (b) Sinusoidal curves in the  $\rho\theta$ -plane; the point of intersection  $(\rho', \theta')$  corresponds to the line passing through points  $(x_1, y_1)$  and  $(x_2, y_2)$  in the  $xy$ -plane. (c) Division of the  $\rho\theta$ -plane into accumulator cells.

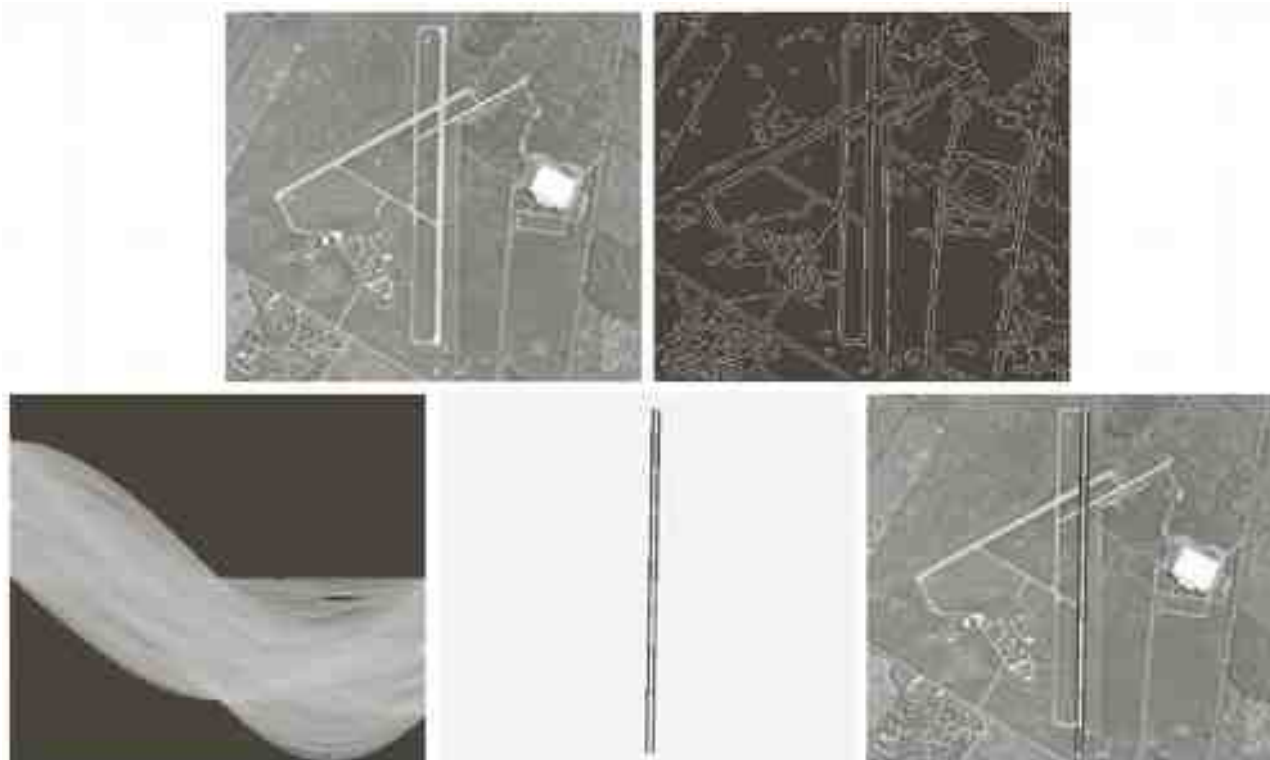
# Transformada Hough. Líneas.



**FIGURE 10.33**  
(a) Image of size  $101 \times 101$  pixels, containing five points.  
(b) Corresponding parameter space. (The points in (a) were enlarged to make them easier to see.)



# Transformada Hough. Líneas.



a b  
c d e

**FIGURE 10.34** (a) A  $502 \times 564$  aerial image of an airport. (b) Edge image obtained using Canny's algorithm. (c) Hough parameter space (the boxes highlight the points associated with long vertical lines). (d) Lines in the image plane corresponding to the points highlighted by the boxes. (e) Lines superimposed on the original image.

# Transformada de Hough

- Se puede generalizar y buscar otras formas: círculos, elipses, etc.
  - Círculos:  $(x-a)^2+(y-b)^2=r^2$
  - El espacio de parámetros será de 3 dimensiones:  $(a, b, r)$ .
- Problemas:
  - Cómo definimos el umbral de significatividad?
  - Vemos la forma buscada... aunque no exista...

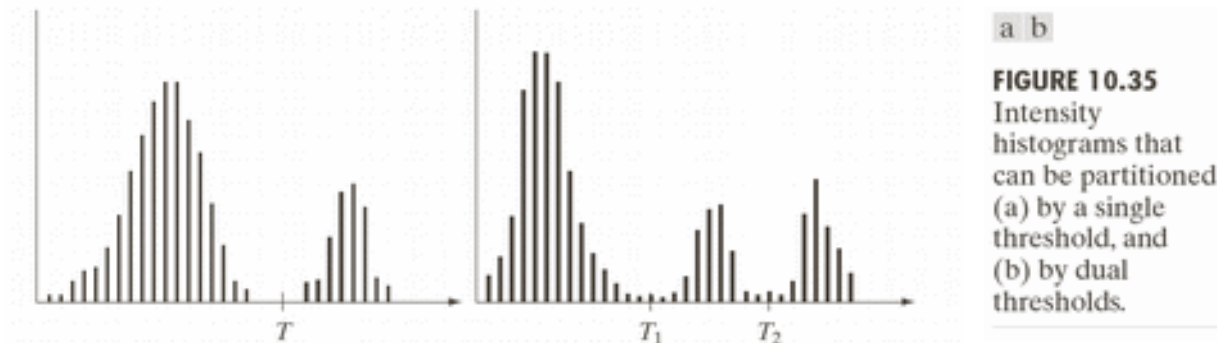
# Umbrales

- Se trata de un método para detectar regiones, es decir continuidad.
- Se busca un valor de umbral  $T$  que permita dividir los elementos de imagen en 2 grupos:

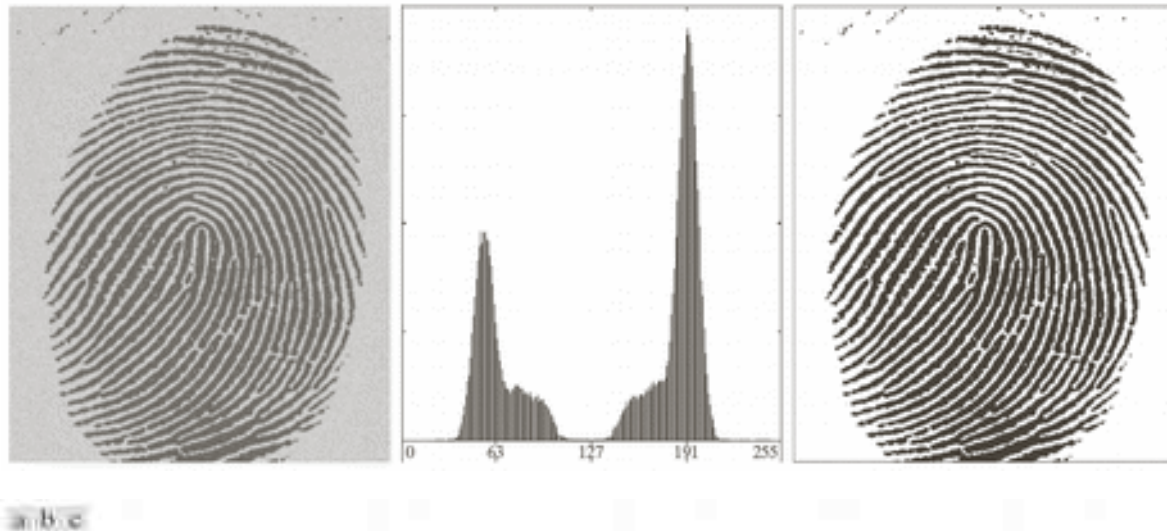
$$g(x, y) = \begin{cases} 1 & \text{if } f(x, y) > T \\ 0 & \text{if } f(x, y) \leq T \end{cases}$$

# Deteccción automática de T

- El histograma permite estudiar algunas características de la imagen.
- Histograma multi-modal:



# Un ejemplo



**FIGURE 10.38** (a) Noisy fingerprint. (b) Histogram. (c) Segmented result using a global threshold (the border was added for clarity). (Original courtesy of the National Institute of Standards and Technology.)

# Cálculo de T óptimo

- Supongamos una imagen con 2 zonas de brillo. Se trata de estimar la función de densidad de probabilidad  $p(z)$  que es la mezcla de 2 densidades de probabilidad. Una para los pixeles claros y otra para los oscuros.

# Umbral óptimo

$$p(z) = P_1 p_1(z) + P_2 p_2(z)$$

$$p(z) = \frac{P_1}{\sqrt{2\pi}\sigma_1} e^{-\frac{(z-\mu_1)^2}{2\sigma_1^2}} + \frac{P_2}{\sqrt{2\pi}\sigma_2} e^{-\frac{(z-\mu_2)^2}{2\sigma_2^2}}$$

$$P_1 + P_2 = 1$$

$\mu_1, \mu_2$  : Valores medios de nivel de brillo

$\sigma_1, \sigma_2$  : Desviación standard respecto a la media.

$P_1, P_2$  : Probabilidades a priori de los dos niveles.

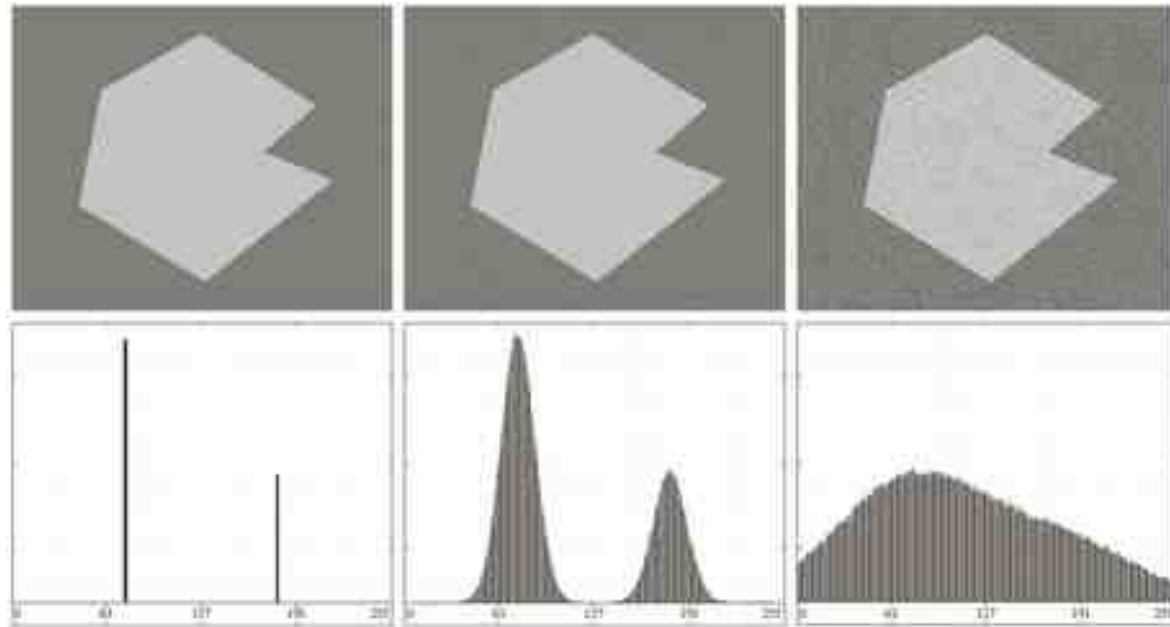
# Umbral óptimo

- Si suponemos distribución gaussiana e iguales varianzas se puede obtener una expresión del umbral óptimo de la forma:

$$T = \frac{\mu_1 + \mu_2}{2} + \frac{\sigma^2}{\mu_1 - \mu_2} \ln\left(\frac{p_2}{p_1}\right)$$



# Umbrales y ruido....



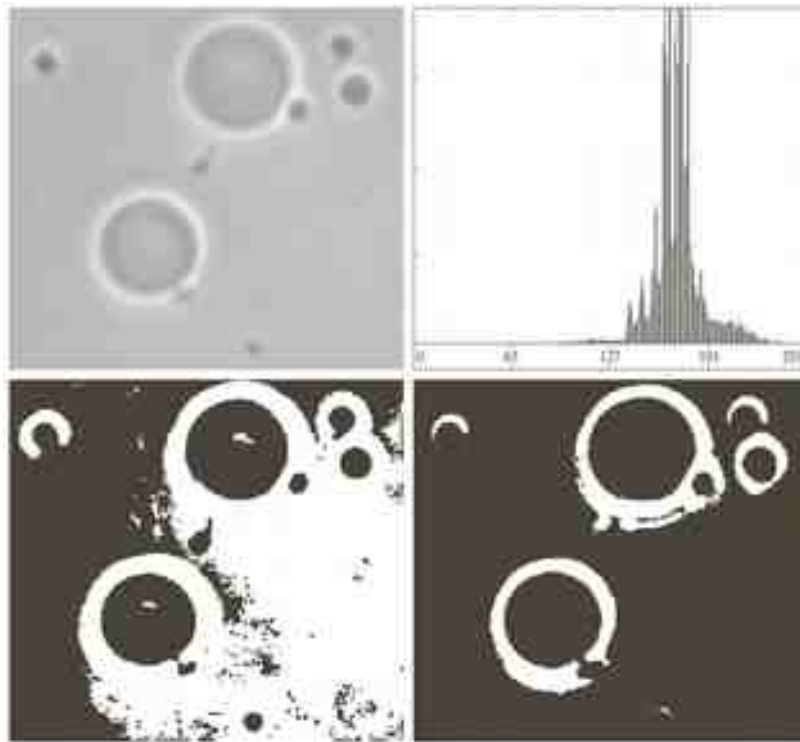
a b c  
d e f

**FIGURE 10.36** (a) Noiseless 8-bit image. (b) Image with additive Gaussian noise of mean 0 and standard deviation of 10 intensity levels. (c) Image with additive Gaussian noise of mean 0 and standard deviation of 50 intensity levels. (d)–(f) Corresponding histograms.

# Método de Otsu

- Estima el umbral óptimo  $k$  que maximiza la varianza interclase.
  - Calcula histograma normalizado.  $L$  niveles de gris.  $p_i, i=0,1,\dots,L-1$
  - Calcula las sumas y medias acumuladas.  $p_1(k)=\sum_{i=0}^k p_i$   $m(k)=\sum_{i=0}^k i p_i$
  - Calcula la media y varianza global  $m_G=\sum_{i=0}^{L-1} i p_i$   $\sigma_G^2=\sum_{i=0}^{L-1} (i-m_G)^2 p_i$
  - Calcula la varianza interclase para toda  $k$   $\sigma_B^2(k)=\frac{(m_G P_1(k)-m(k))^2}{P_1(k)(1-P_1(k))}$
  - El umbral de Otsu (óptimo) es el que maximiza la varianza interclase.  $\sigma_B^2(\hat{k})=\max_{0 \leq k \leq (L-1)} \sigma_B^2(k)$
  - Medida de separabilidad entre clases  $\eta(k)=\frac{\sigma_B^2(k)}{\sigma_G^2}$

# Otsu



a b  
c d

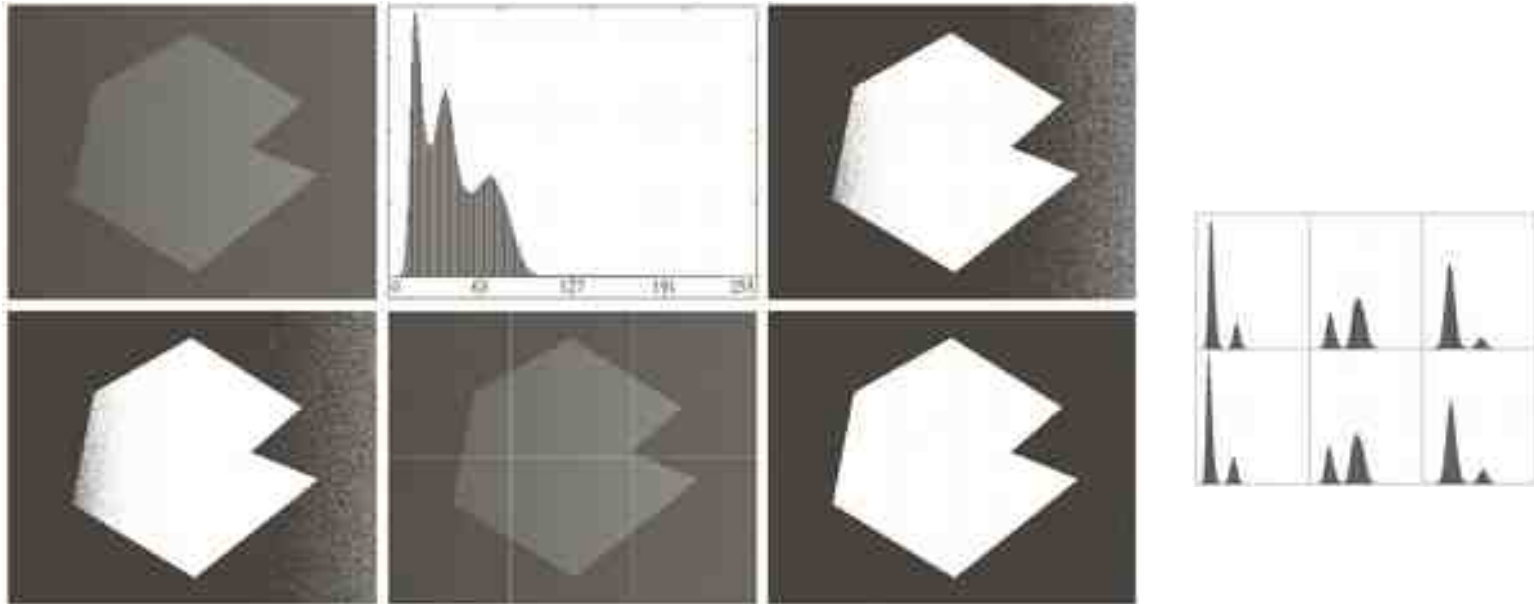
**FIGURE 10.39**

(a) Original image.  
(b) Histogram (high peaks were clipped to highlight details in the lower values).  
(c) Segmentation result using the basic global algorithm from Section 10.3.2.  
(d) Result obtained using Otsu's method. (Original image courtesy of Professor Daniel A. Hammer, the University of Pennsylvania.)

# Iluminación no uniforme

- Puede producir efectos indeseables en una segmentación por umbrales dado el carácter global de la misma.
- Soluciones:
  - ir a un operador más local
  - estimar la iluminación no uniforme y compensarla.

# Iluminación no uniforme



a b c  
d e f

**FIGURE 10.46** (a) Noisy, shaded image and (b) its histogram. (c) Segmentation of (a) using the iterative global algorithm from Section 10.3.2. (d) Result obtained using Otsu's method. (e) Image subdivided into six subimages. (f) Result of applying Otsu's method to each subimage individually.

# Iluminación no uniforme



**FIGURE 10.49** (a) Text image corrupted by spot shading. (b) Result of global thresholding using Otsu's method. (c) Result of local thresholding using moving averages.

# Segmentación de regiones

- Dividir la escena en  $n$  subregiones conexas tal que:

$$\bigcup_{i=1}^n R_i = R$$

$$R_i \cap R_j = \emptyset \text{ para todo } i \text{ y } j, i \neq j.$$

$$P(R_i) = \text{VERDADERO para } i = 1, 2, \dots, n$$

$$P(R_i \cup R_j) = \text{FALSO para } i \neq j$$

- Las últimas 2 condiciones indican que la región cumple una propiedad y que los pixels de 2 regiones son distintas en el sentido de esa propiedad, que define la segmentación.

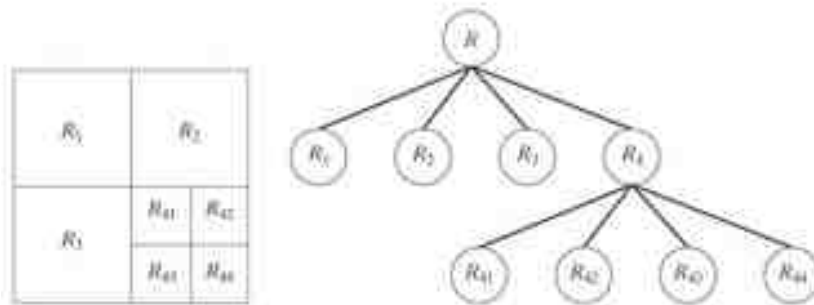
# Regiones

- Una opción: Seleccionar puntos semilla, que definen las regiones.
- Agregar píxeles a dichas regiones iterativamente mientras se cumplan las condiciones.
- Otra opción (Split and merge): Dividir y fusionar iterativamente



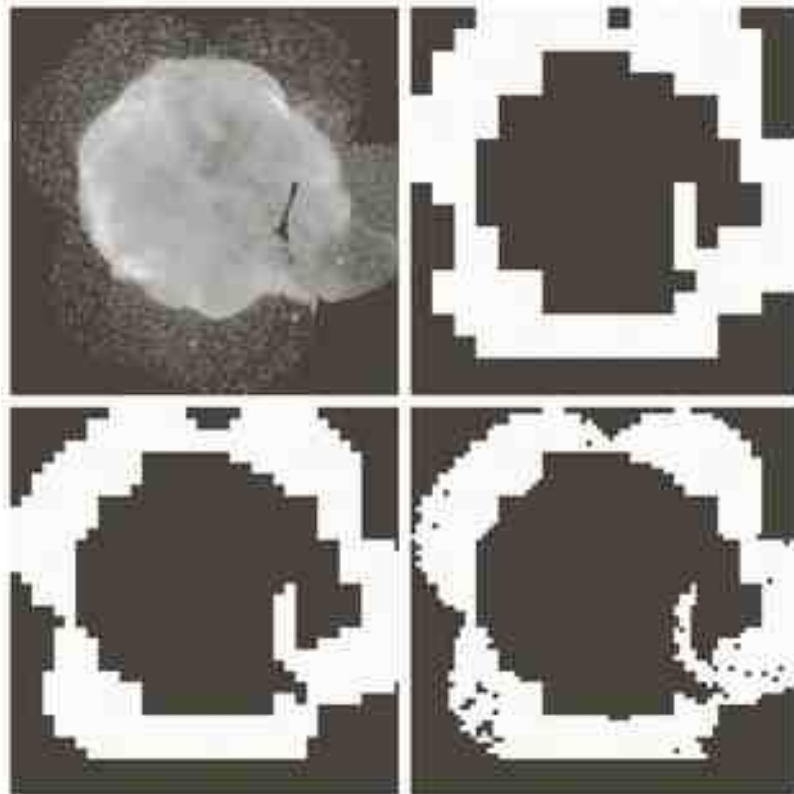
# Split and merge

- Dividir en cuadrados cada region donde  $P(R_i) = False$
- Cuando no quede nada para dividir, unir regiones adyacentes si  $P(R_j \cup R_k) = True$
- Parar cuando no sea posible más uniones.



a/b  
**FIGURE 10.52**  
(a) Partitioned image.  
(b) Corresponding quadtree.  $R$  represents the entire image region.

# Quad tree



a b  
c d

**FIGURE 10.53**  
(a) Image of the Cygnus Loop supernova, taken in the X-ray band by NASA's Hubble Telescope. (b)-(d) Results of limiting the smallest allowed quadregion to sizes of  $32 \times 32$ ,  $16 \times 16$ , and  $8 \times 8$  pixels, respectively. (Original image courtesy of NASA.)

# Geodesic Active Regions

- Versión variacional: el algoritmo GAR
  - Particionar  $I$  tal que las PDF entre regiones sean distantes y dentro de las regiones lo más uniformes posible....
  - <ftp://ftp-sop.inria.fr/odyssee/Team/Rachid.Deriche/Lectures/Master-Stic-IGMMV/level-set.pdf>