



## Segmentación

*Que es la segmentación de imágenes*

**Métodos de segmentación clásicos**

**Métodos de segmentación basados en Deep Learning**



*ugr*

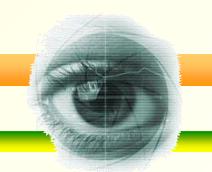
Universidad  
de Granada





# Que es la segmentación de imágenes

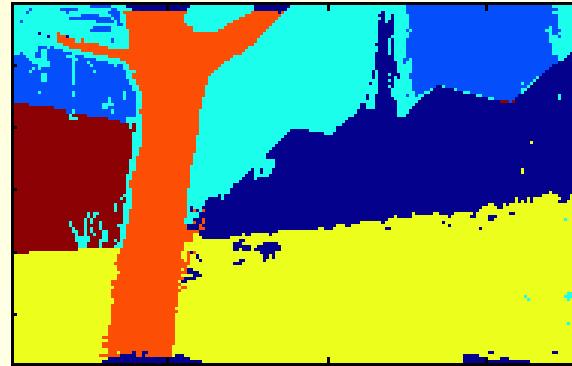
## Niveles en el análisis de imágenes



**Segmentación:**división de la imagen en regiones para simplificar el análisis de la imagen



*Original*



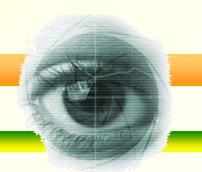
*División en regiones*

**Región:**representa objetos o partes de objetos, formado por un conjunto de pixeles. La segmentación de imágenes traslada la unidad de análisis del pixel a la unidad de análisis de región.



# Que es la segmentación de imágenes

## Niveles en el análisis de imágenes



### Niveles en el análisis de imágenes:

**Clasificación:** clasificar la imagen en un clase como personas, animales, paisaje.

Paisajes de exterior

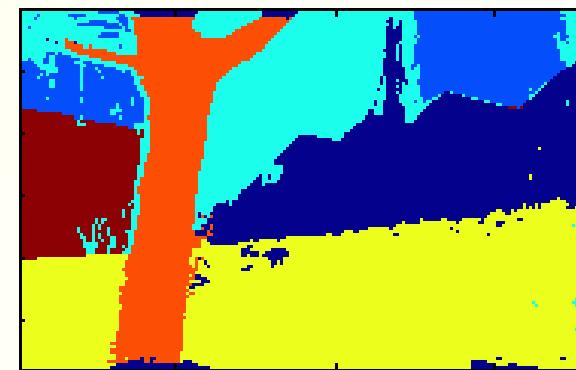


### Detección de objetos:

detectar los objetos en una imagen y crear una anotación dibujando un rectángulo para identificar el objeto.



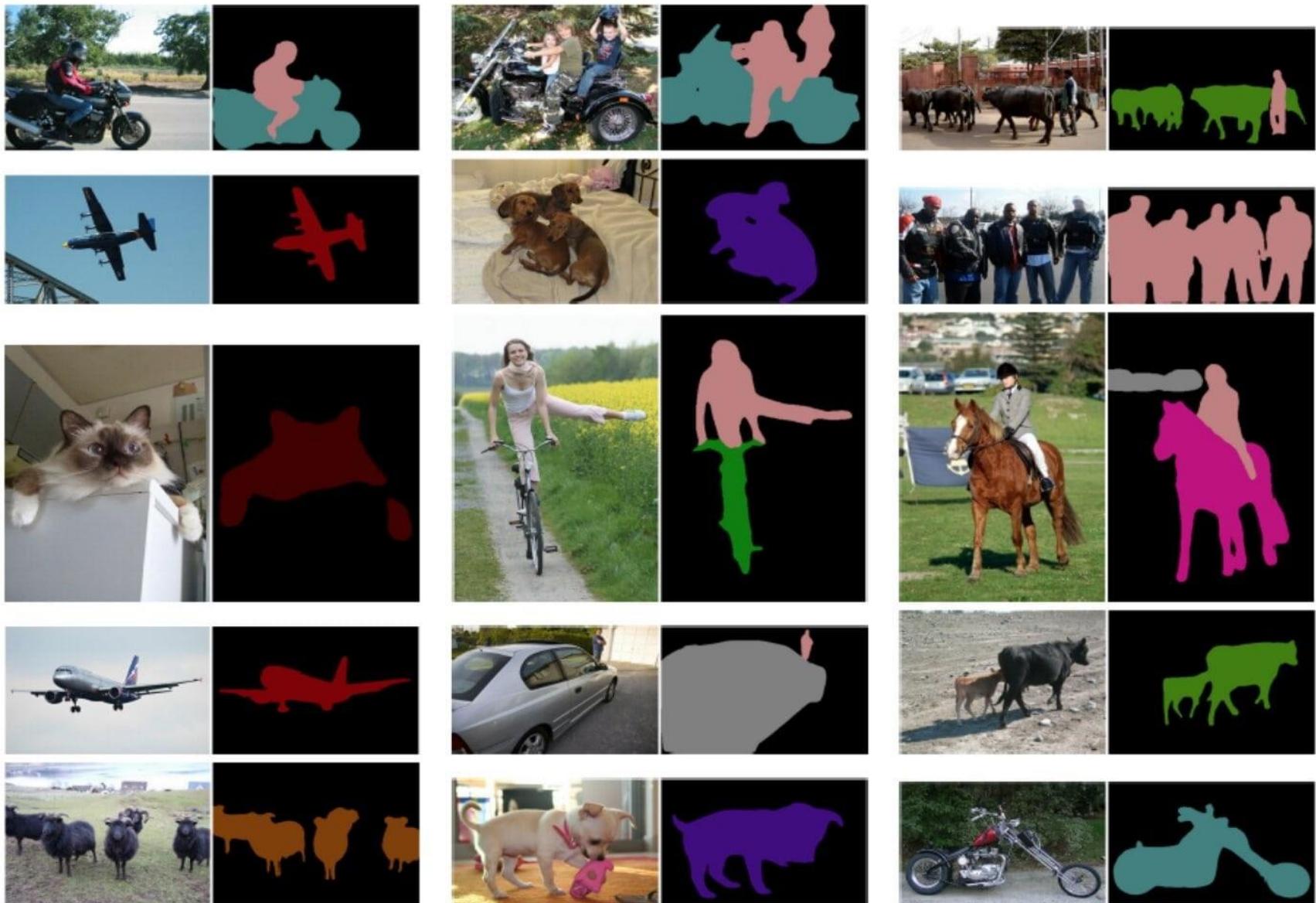
**Segmentación:** identificar partes o regiones de la imagen y entender a que objeto pertenece.





# Que es la segmentación de imágenes

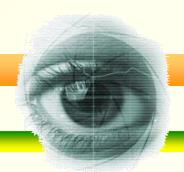
## Ejemplos de segmentación





# Que es la segmentación de imágenes

## Ejemplos de aplicaciones

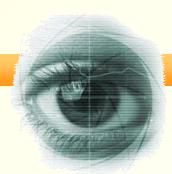


**Análisis de imágenes médicas:** extracción del límite tumoral, medidas del volumen de tejidos.

**Automoción:** establecer la superficie viable, detección de peatones, detección de la señales, etc.

**Vigilancia con video:** seguridad, cuidado a la tercera edad, etc.

**Realidad aumentada:** ayuda a personas con discapacidades, Aplicaciones militares.



## Segmentación

***Que es la segmentación de imágenes***

***Métodos de segmentación clásicos***

***Métodos de segmentación basados en Deep Learning***



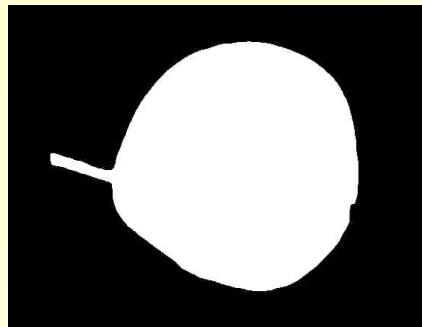
*ugr*

Universidad  
de Granada

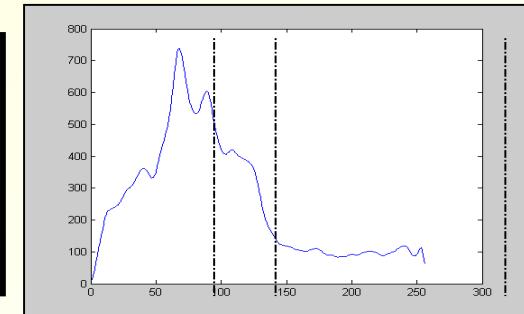




### Umbralización.



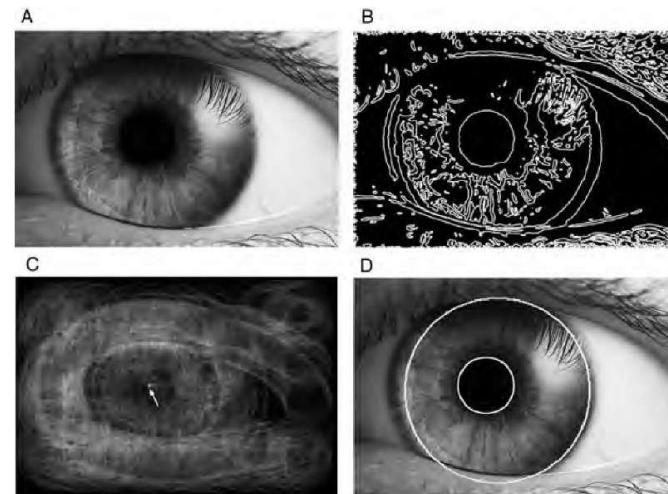
### Basados en el histograma



### Basados en el crecimiento de regiones



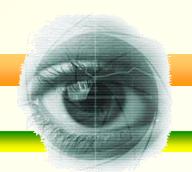
### Basados en aristas: Transformada de Hough



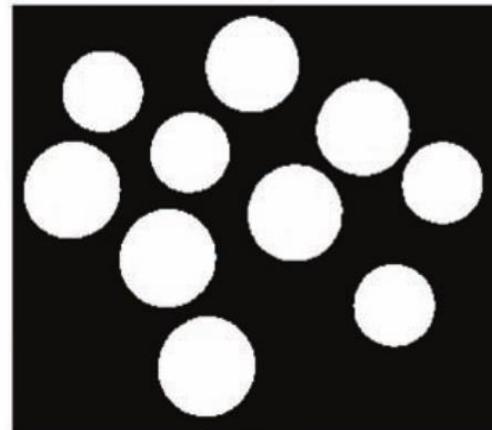


# Métodos de Segmentación Clásicos

## Umbralización



$I(i,j)$



$O(i,j)$

Con la **umbralización** creamos imágenes binarias a partir de imágenes de niveles de gris.

La **umbralización** más trivial produce dos regiones: región de fondo, región con objetos en primer plano.

**Predicado para distinguir entre las dos regiones**

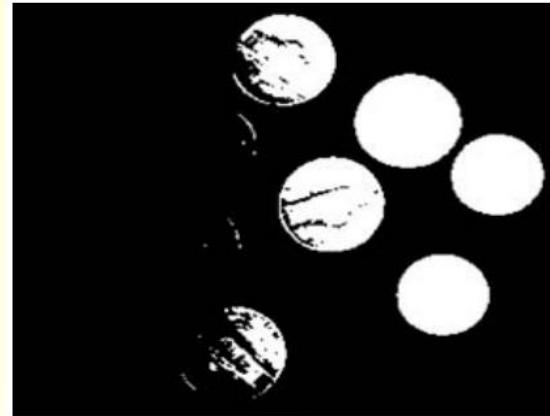
$$O(i,j) = \begin{cases} 1 & \text{si } I(i,j) > T \\ 0 & \text{en otro caso} \end{cases}$$



La estrategia para definir  $T$  determina el tipo de umbralización

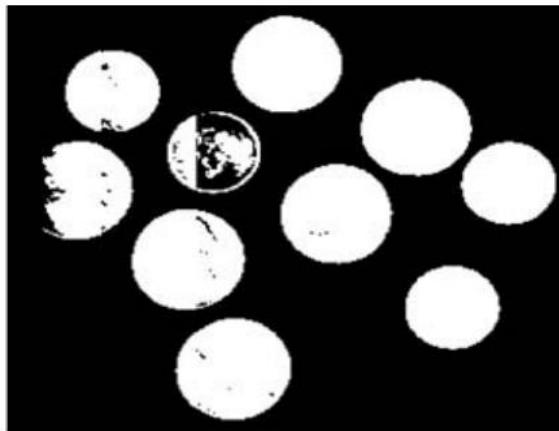
Umbralización **global**:  $T$  depende solo de los valores de nivel de gris de la imagen  $I$

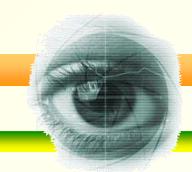
$$O(i,j) = \begin{cases} 1 & \text{si } I(i,j) > T \\ 0 & \text{en otro caso} \end{cases}$$



Umbralización **dinámica**

o **adaptativa**:  $T$  depende de  $I$  y puede que de alguna propiedad local del punto  $(i,j)$  además de las coordenadas espaciales de  $(i,j)$





- Si tenemos información a priori de que los valores de intensidad de todos los píxeles de los objetos están en el **rango** entre los valores  $T_1$  y  $T_2$ , podemos realizar la siguiente operación de umbralización:

$$F_T[i,j] = \begin{cases} 1 & \text{si } T_1 \leq F[i,j] \leq T_2 \\ 0 & \text{en otro caso} \end{cases}$$

- Si los valores de gris de todos los píxeles de los objetos **no están en un intervalo particular**, pero son diferentes de los valores del fondo, podemos hacer lo siguiente:

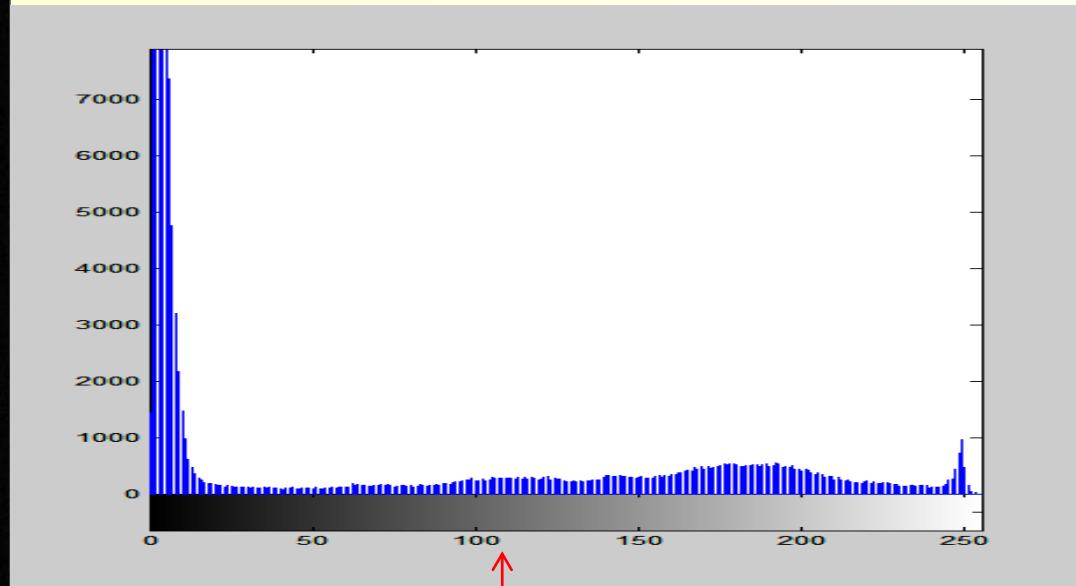
$$F_T[i,j] = \begin{cases} 1 & \text{si } F[i,j] \in Z \\ 0 & \text{en otro caso} \end{cases}$$

Z es el conjunto de niveles de gris de los píxeles de los objetos.



# Métodos de Segmentación Clásicos

## Umbralización. Ejemplo



$T=80$       objeto:  $80 < f(m,n) < 255$

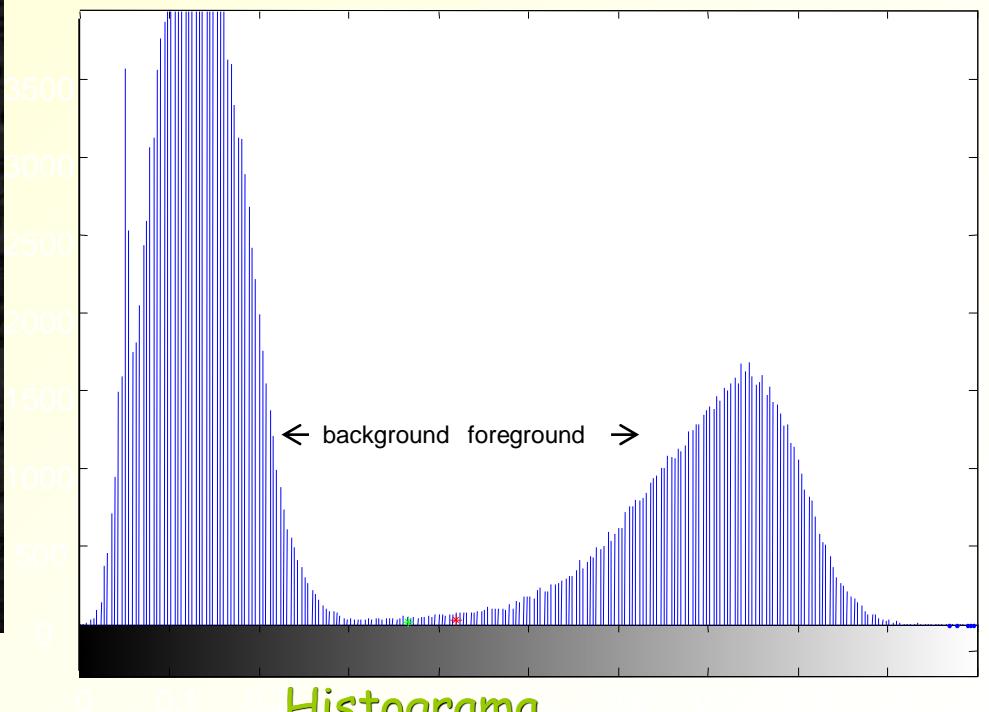


# Métodos de Segmentación Clásicos

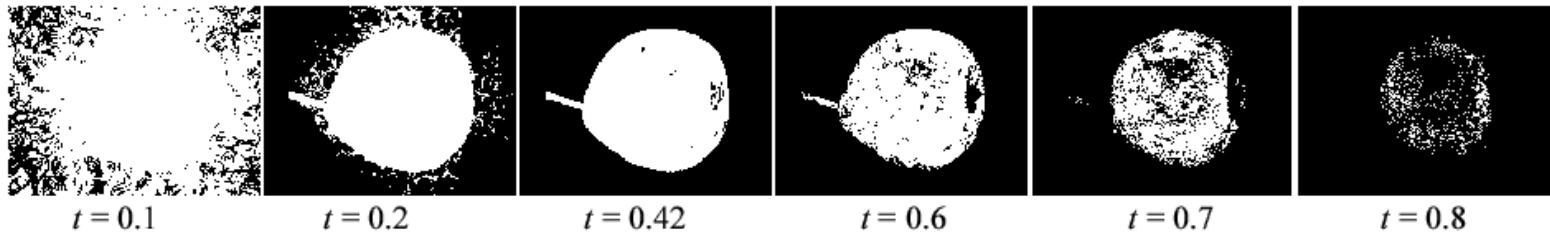
## Umbralización. Ejemplo



Imagen de alto contraste



Elección de distintos umbrales





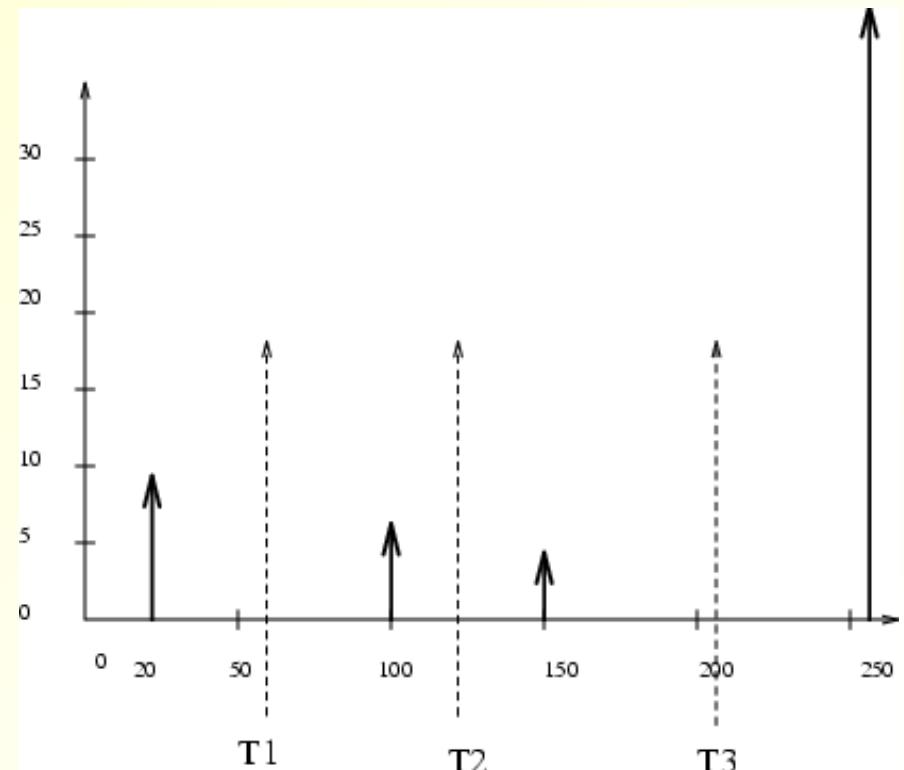
### Ejemplo numérico con más de dos regiones

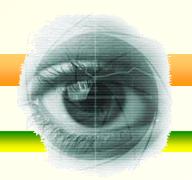
255	235	255	255	255	255	255	20
255	255	255	100	100	255	20	20
255	255	255	100	100	255	20	20
255	255	255	100	100	255	20	20
255	255	255	255	255	255	20	20
255	255	255	255	255	255	255	255
150	150	255	255	255	255	255	255
150	150	255	255	255	255	255	255

$$B_1(x, y) = \begin{cases} 1 & \text{si } 0 < f(x, y) < T_1 \\ 0 & \text{en otro caso} \end{cases}$$

$$B_2(x, y) = \begin{cases} 1 & \text{si } T_1 < f(x, y) < T_2 \\ 0 & \text{en otro caso} \end{cases}$$

$$B_3(x, y) = \begin{cases} 1 & \text{si } T_2 < f(x, y) < T_3 \\ 0 & \text{en otro caso} \end{cases}$$





**Método de Otsu:** Es un método de segmentación basado en la umbralización global. Genera dos regiones una región clara y otra oscura.

**Objetivo:** Encontrar un umbral  $T$  que minimice la varianza ponderada dentro de las clases. Que equivale a maximizar la varianza entre clases.

### Elementos:

$T_0$ : región con los niveles de gris  $\{0, 1, \dots, t\}$  (región oscura)

$T_1$ : región con los niveles de gris  $\{t+1, t+2, \dots, l\}$  (región clara)

$P(i)$ : el histograma de probabilidades, para  $i=0, 1, \dots, l$

$w_f(t)$ ,  $\mu_f(t)$ , and  $\sigma^2_f(t)$  son los pesos, media y varianza de la región  $T_1$ , respectivamente.

$w_b(t)$ ,  $\mu_b(t)$ , and  $\sigma^2_b(t)$  son los pesos, media y varianza de la región  $T_0$ , respectivamente.

$\sigma^2_w(t)$ : la suma ponderada de las dos varianzas, da lugar a la varianza en clases

$$\sigma^2_w(t) = w_b(t) \sigma^2_b(t) + w_f(t) \sigma^2_f(t)$$

$$w_b(t) = \sum_{i=0}^t P(i)$$

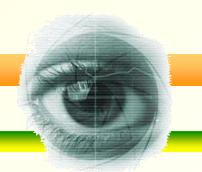
$$\mu_b(t) = \frac{\sum_{i=0}^t i P(i)}{w_b(t)}$$

$$\sigma^2_b(t) = \frac{\sum_{i=0}^t (i - \mu_b(t))^2 P(i)}{w_b(t)}$$

$$w_f(t) = \sum_{i=t+1}^l P(i)$$

$$\mu_f(t) = \frac{\sum_{i=t+1}^l i P(i)}{w_f(t)}$$

$$\sigma^2_f(t) = \frac{\sum_{i=t+1}^l (i - \mu_f(t))^2 P(i)}{w_f(t)}$$



### Método de Otsu.

**Acción:** Escoger el valor de  $t$  que minimice  $\sigma_w^2(t)$ :

**Mejoras de eficiencia para buscar el mejor  $T$ :**

**La varianza entre clases  $\sigma_B^2$  se define como (que debemos maximizar)  $\sigma_B^2 = \sigma^2 - \sigma_w^2(t)$  desarrollando tenemos que:**

$$\begin{aligned}\sigma_B^2 &= w_b(t) w_f(t)(\mu_b(t) - \mu_f(t))^2 \\ &= w_b(t) (1 - w_b(t))(\mu_b(t) - \mu_f(t))^2\end{aligned}$$

**Aplicamos los siguientes pasos:**

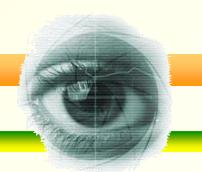
**PASO1 (Inicialización):**  $w_b(0)=P(0)$  y  $\mu_b(0)=0$  obtener  $\sigma_B^2(0)$

**PASO2 (recursión): Para todo  $t=1, \dots, l$**

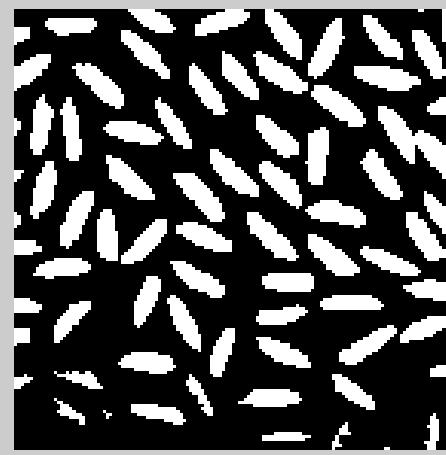
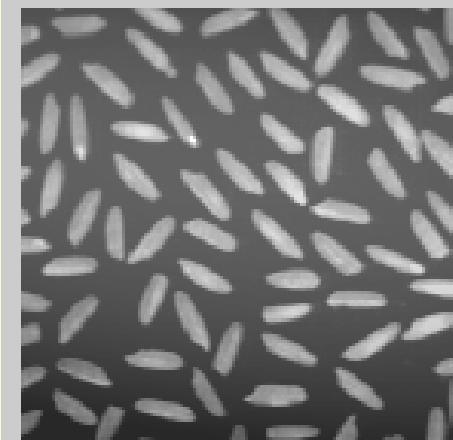
**2.1**  $w_b(t+1)=w_b(t)+P(t+1)$  y  $\mu_b(t) = \frac{w_b(t)\mu_b(t)+(t+1)P(t+1)}{w_b(t+1)}$

**2.1 Obtener  $\sigma_B^2(t+1)$**

**PASO3 (selección de  $t$ ): escoger el  $t_{best} = \operatorname{argmax}_{t=0 \dots l} (\sigma_B^2(t))$**



### Método de Otsu. Ejemplo

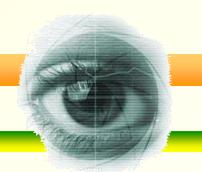


ponents or broken connection paths. There is no point past the level of detail required to identify those components.

Segmentation of nontrivial images is one of the most difficult tasks in computer vision and image processing. Segmentation accuracy determines the effectiveness of computerized analysis procedures. For this reason, considerable attention must be taken to improve the probability of rugged segmentation. In applications such as industrial inspection applications, at least some information about the environment is possible at times. The experienced computer vision designer invariably pays considerable attention to such issues.

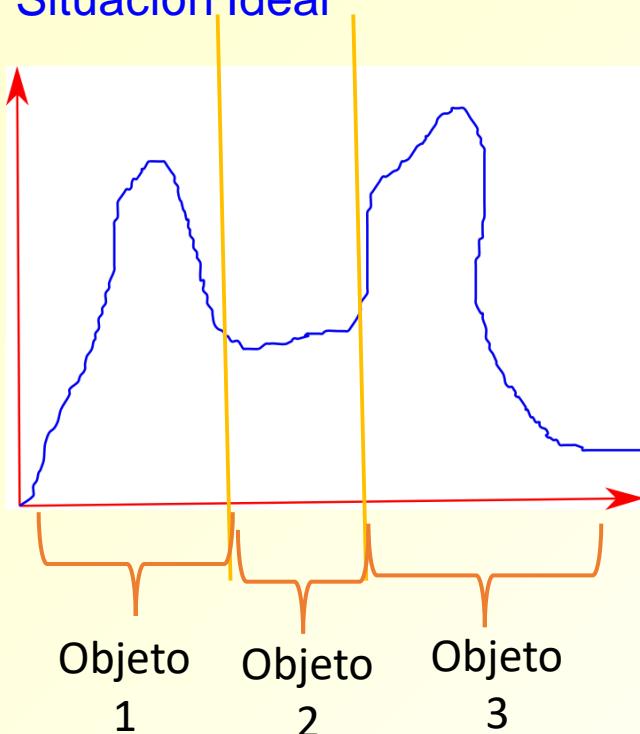
ponents or broken connection paths. There is no point past the level of detail required to identify those components.

Segmentation of nontrivial images is one of the most difficult tasks in computer vision and image processing. Segmentation accuracy determines the effectiveness of computerized analysis procedures. For this reason, considerable attention must be taken to improve the probability of rugged segmentation. In applications such as industrial inspection applications, at least some information about the environment is possible at times. The experienced computer vision designer invariably pays considerable attention to such issues.

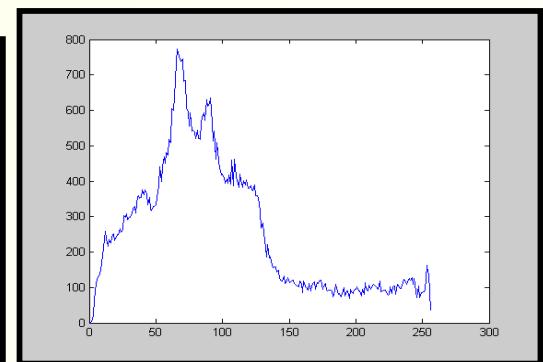


**Objetivo:** Usando la información del histograma establecer una partición del rango de valores de la imagen que permitan obtener las diferentes regiones de la imagen

Situación Ideal



Problema: **Ruido**





# Métodos de Segmentación Clásicos

## Basados en el histograma. Efectos del Ruido



Imagen  
original



Histograma

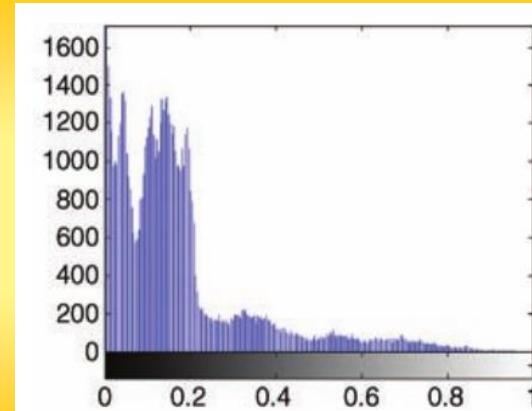
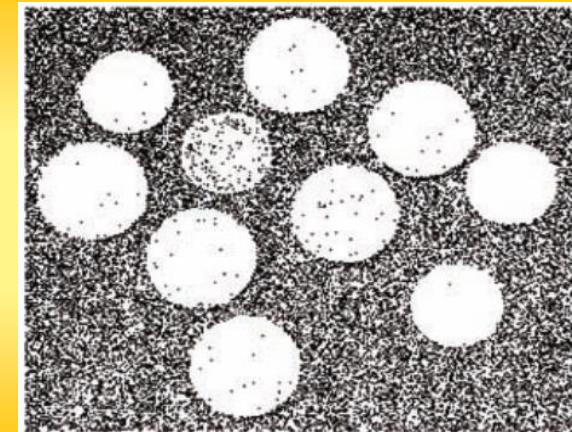
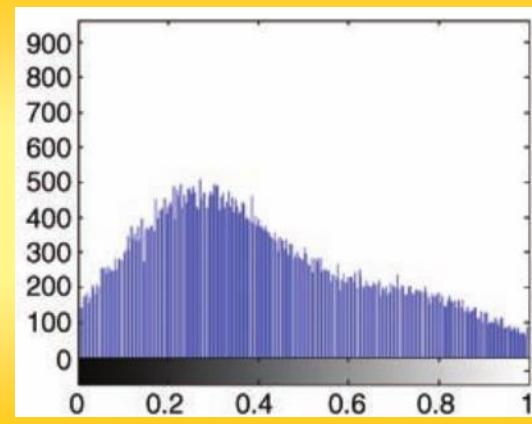
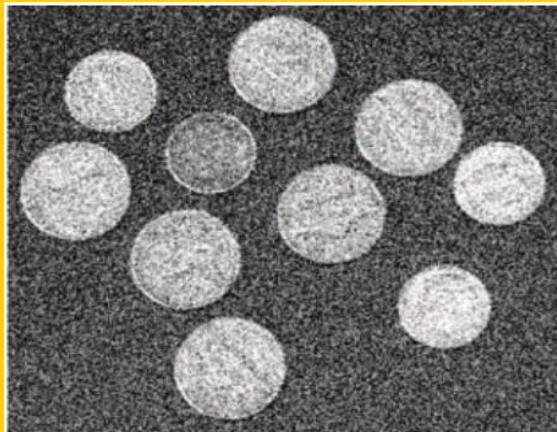
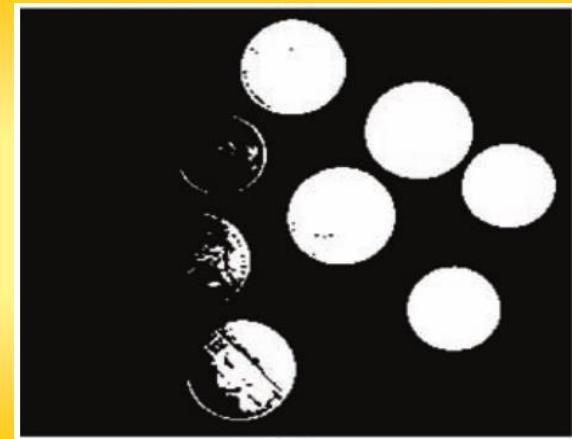
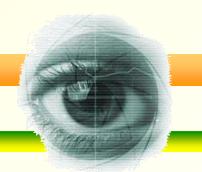


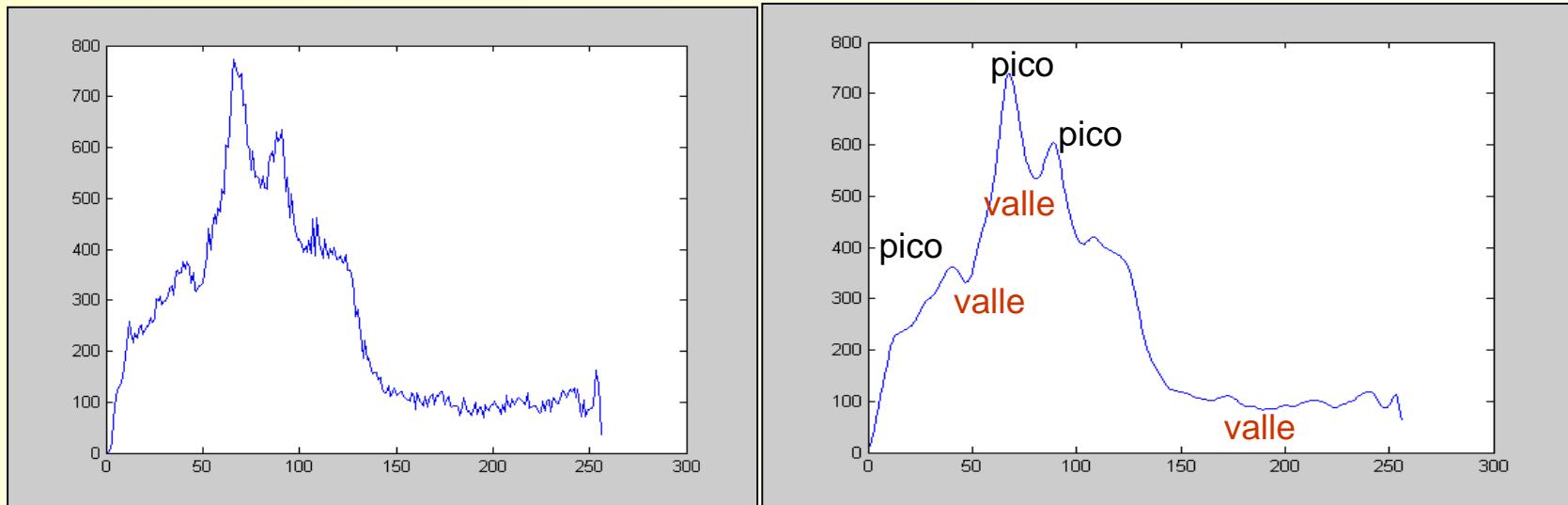
Imagen  
umbralizada





**Paso 1:** Obtener el histograma y alisar el ruido usando un filtro 1d (p.e gausiana 1d)

**Paso 2:** Detectar Picos (máximos locales) y Valles (mínimos locales) en el histograma



**Paso 3:** Segmentar la imagen en distintas imágenes binarias usando como umbrales los valles.

**Paso 4:** Aplicar un algoritmo de componentes conexas a cada imagen binaria para encontrar las regiones conectadas.



# Métodos de Segmentación Clásicos

## Basados en el histograma

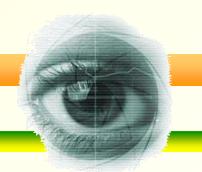
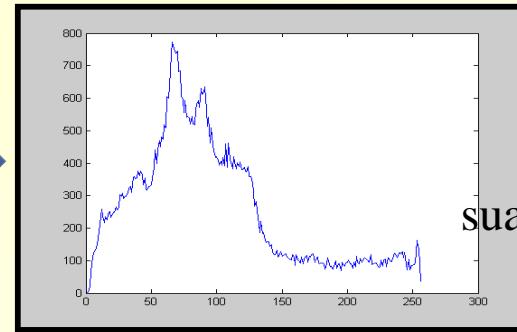
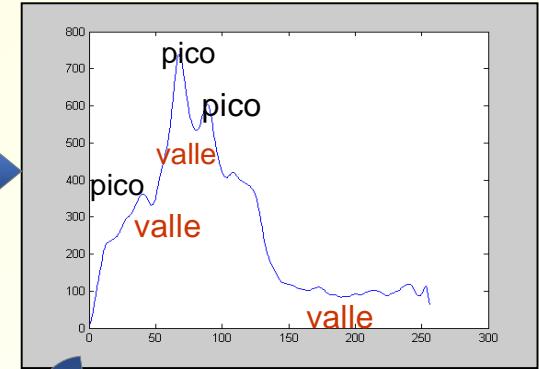


Imagen de entrada

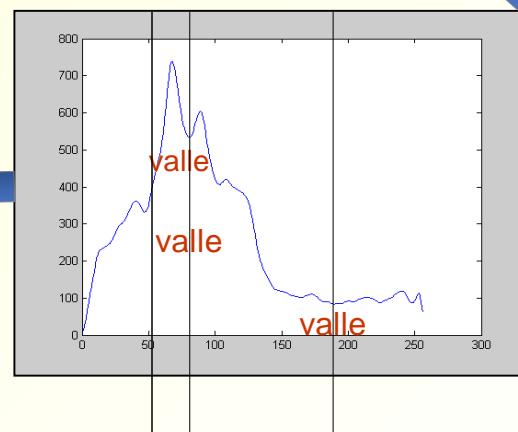


suavizar



Valles: $v_i$

Picos: $p_i$



$T_1$   $T_2$   $T_3$

Aplicar un algoritmo de componentes conexas a cada imagen binaria para encontrar las regiones conectadas.

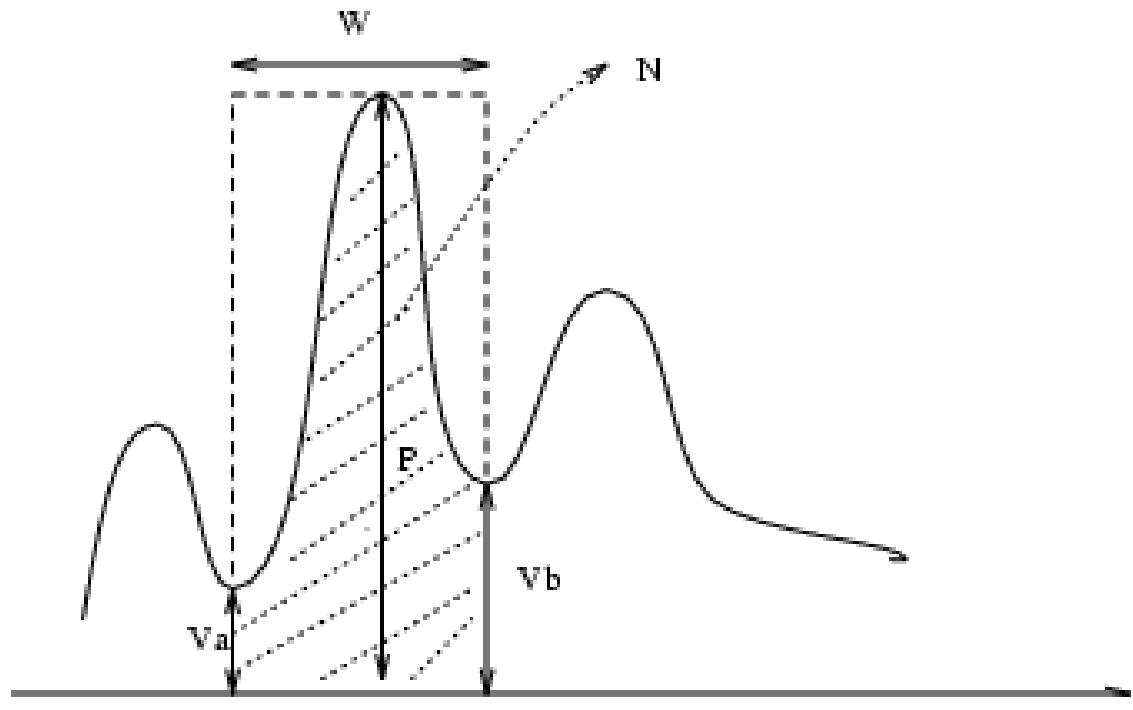


# Métodos de Segmentación Clásicos

## Basados en el histograma



### Test para encontrar picos adecuados

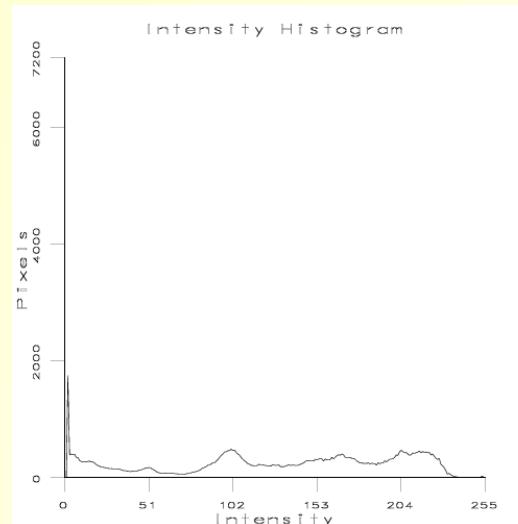
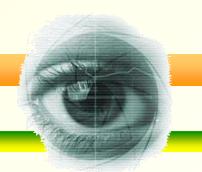


$$\text{Peakiness} = \left(1 - \frac{(V_a + V_b)}{2P}\right) \cdot \left(1 - \frac{N}{(W.P)}\right)$$

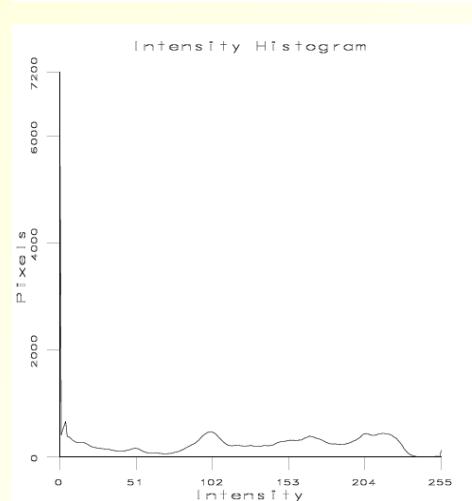


# Métodos de Segmentación Clásicos

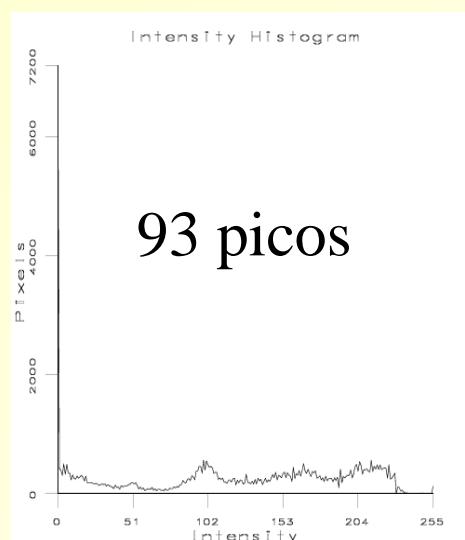
## Basados en el histograma. Ejemplo



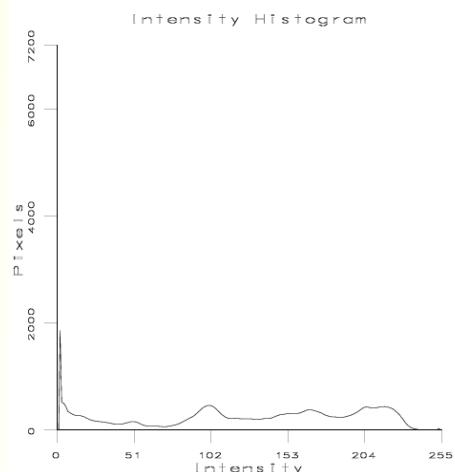
**Histograma alisado**  
Tamaño de la máscara : 5  
Filtro media Aplicado :1 vez.  
**54 picos**  
Con Peakness:18 picos



**93 picos**



**Histograma alisado**  
**21 picos (2 veces)**  
Tras el filtrado de picos: 7

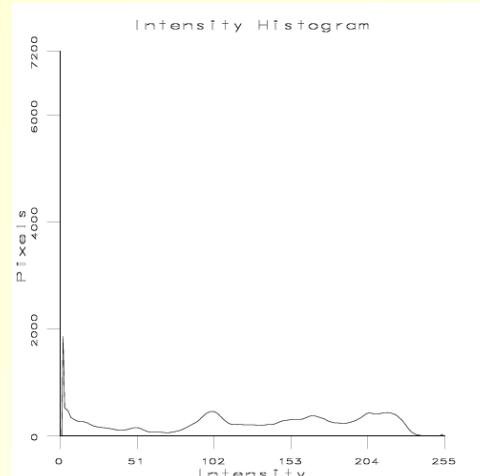


**Histograma alisado**  
**11 picos (3 veces)**  
Tras el filtrado de picos: 4



# Métodos de Segmentación Clásicos

## Basados en el histograma. Ejemplo



Histograma alisado  
11 picos (3 veces)  
Tras el filtrado de  
picos: 4



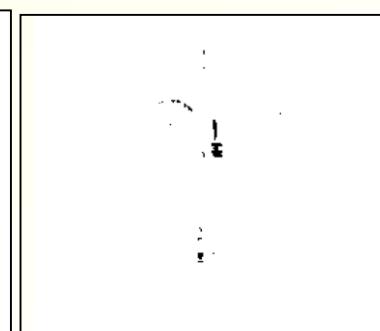
(0,40)



(40, 116)



(116,243)



(243,255)



# Métodos de Segmentación Clásicos

## Basados en el histograma

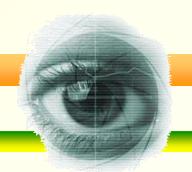
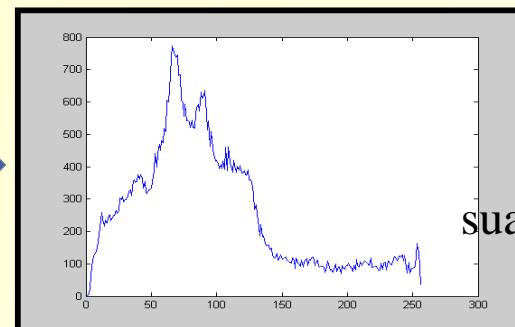
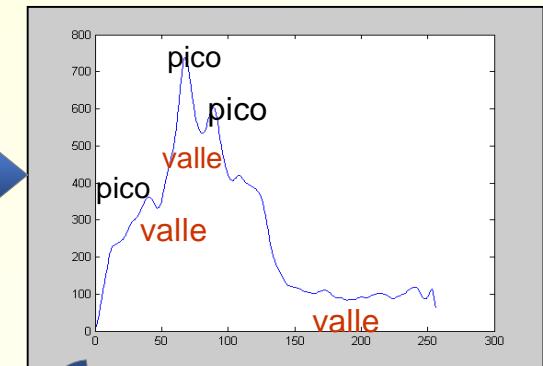


Imagen de entrada

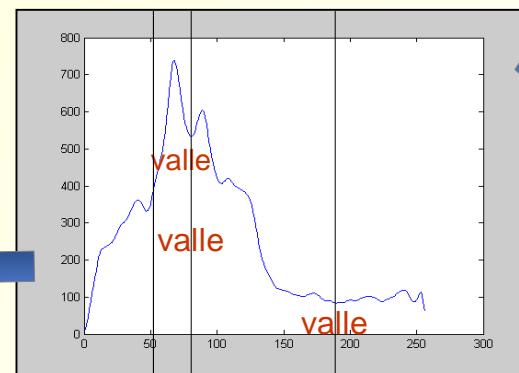


Basados en el histograma

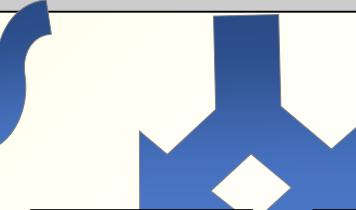
suavizar



Aplicar un algoritmo de componentes conexas a cada imagen binaria para encontrar las regiones conectadas.



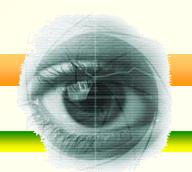
$T_1 \quad T_2 \quad T_3$



Etiquetar los píxeles dentro del rango de un umbral con la misma etiqueta, p.ej.,  $a, b, c \dots$  o  $1, 2, 3$



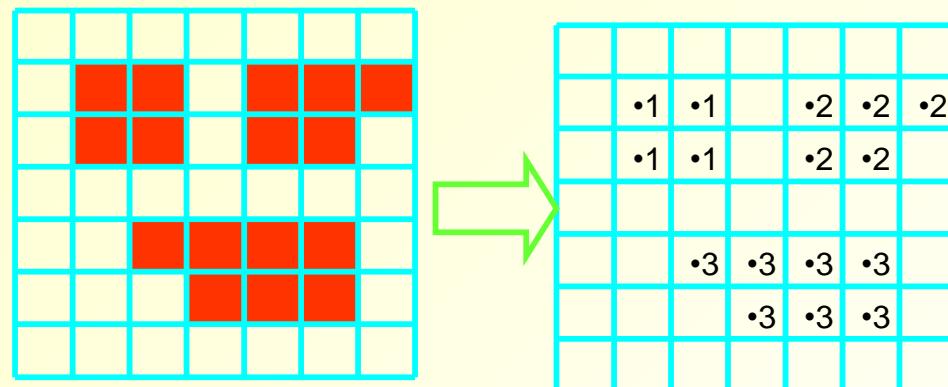
Aplicar Componentes conexas.



### Componentes Conexas. Etiquetado.

**Objetivo:** Distinguir los objetos diferentes

Se asigna una única etiqueta a todos los píxeles de la misma componente.



**Paso 1:** Recorrer la imagen para encontrar un pixel (rojo) no etiquetado y asignarle una nueva etiqueta L.

**Paso 2:** Asignar recursivamente una etiqueta L a todos sus vecinos (rojos).

**Paso 3:** Parar si no hay más píxeles (rojos) no etiquetados.

**Paso 4:** Ir al paso 1.



# Métodos de Segmentación Clásicos

## Basados en el histograma



**Umbralización automática:** Se divide el histograma usando un único umbral global

Algoritmo para obtener T automáticamente:

**Paso 1:** Seleccionar una estimación inicial de T

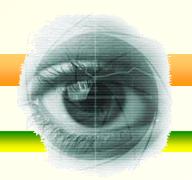
**Paso 2:** Segmentar la imagen usando T. Se generan dos grupos de píxeles G1 y G2

- 1) 2.1. Calcular la media de los valores de gris  $m_1$  y  $m_2$  en G1 y G2
- 2) 2.2. Calcular el nuevo umbral  $T = (m_1 + m_2)/2$
- 3) 2.3. Repetir los pasos 2.1 al 2.3 hasta que la diferencia de T en sucesivas iteraciones sea menor que un umbral  $T_o$

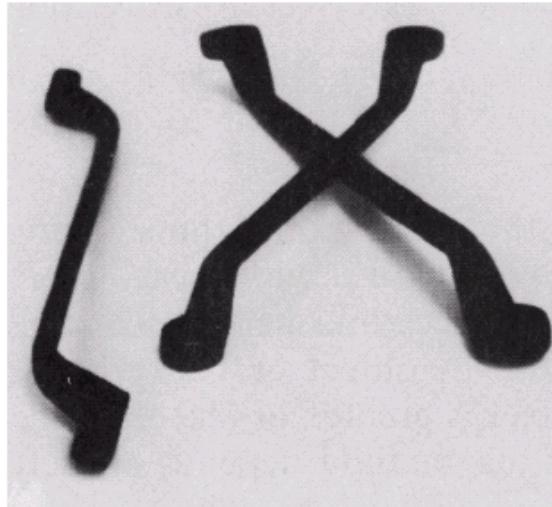
**Consideraciones:** El valor inicial de T podría ser el nivel gris medio de la imagen.

**Problemas:**

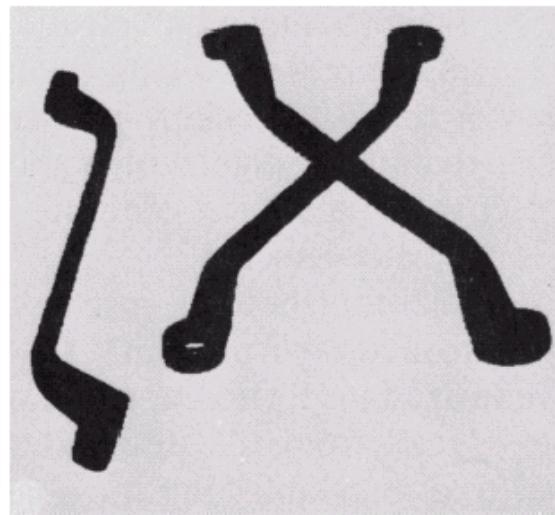
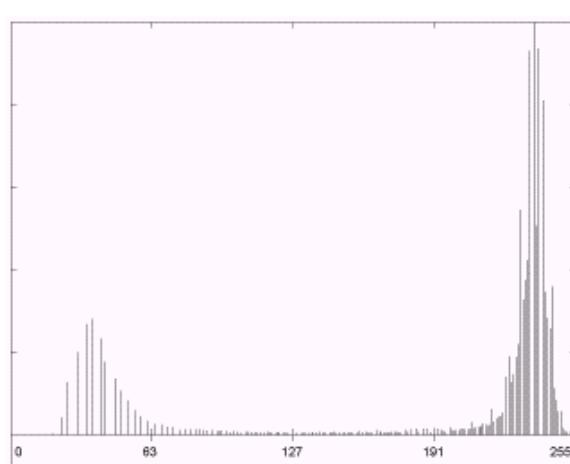
- Iluminación deficiente.
- Histograma unimodal, sin distinción entre objeto y fondo.



### Umbralización automática. Ejemplo

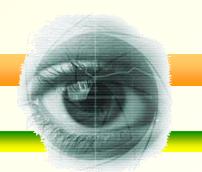


(a)Imagen original.

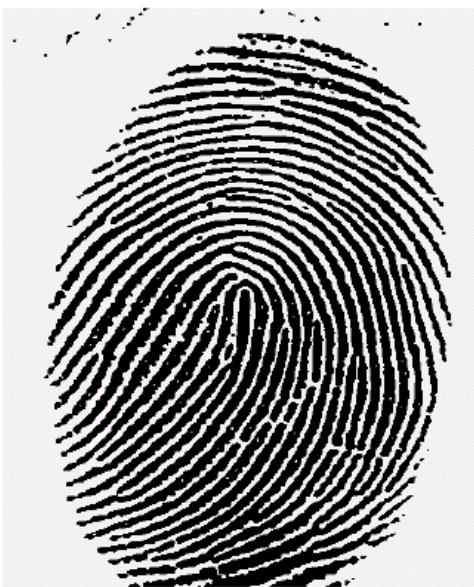
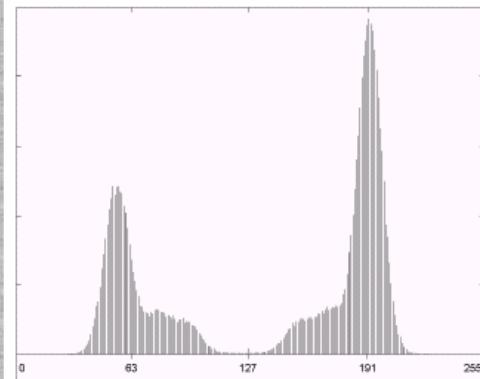


(b)Histograma.

(c)Resultado de una umbralización automática con  $T$  siendo la mitad entre los niveles de gris máximo y mínimo.



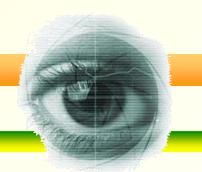
### Umbralización automática. Ejemplo



(a)Imagen original.

(b)Histograma.

(c)Resultado de la segmentación automática.



### Propiedades.

- Las regiones obtenidas en la partición,  $R_1, R_2, \dots, R_k$ , deben ser disjuntas

$$R_i \cap R_j = \emptyset, i \neq j$$

- Su unión debe ser la imagen completa,

$$\bigcup_{i=1}^k R_i = I$$

- Cada región  $R_i$  tiene que ser **conexa**, es decir, todos sus píxeles conectados
- Se debe verificar que:

$$P(R_i) = \text{VERDADERO} \quad \text{y} \quad P(R_i \cup R_j) = \text{FALSO},$$

para **regiones adyacentes** cualesquiera,  $R_i$  y  $R_j$ , siendo  $P$  el predicado que nos proporciona el **test de homogeneidad** de la región.

**Ejemplo de Predicado:** “la máxima diferencia de niveles de gris entre dos píxeles no debe ser mayor que un umbral”.



# Métodos de Segmentación Clásicos

## Basados en Regiones



### Segmentación basada en el crecimiento de regiones

**Se elige aleatoriamente un conjunto de píxeles, que se llaman **semillas**. Cada semilla determina una región.**

**Mecanismo de Crecimiento:** Para cada píxel **(x,y)** se añade a la region  $R_i$  si están en el entorno de la región y cumple la **regla de homogeneidad**:

$$P(R_i \cup (r,s)) = \begin{cases} \text{VERDAD si } |f(r,s) - m_i| \leq T_i(r,s) \\ \text{FALSO en otro caso} \end{cases}$$

$$T_i(r,s) = \left( 1 - \frac{s_i^{\star}}{m_i^{\star}} \right) T$$

6	7	7	6	5
7	7	8	6	5
5	5	6	7	6
0	1	2	0	1
1	0	0	2	0

6	7	7	6	5
7	7	8	6	5
5	5	6	7	6
0	1	2	0	1
1	0	0	2	0

Primera iteración

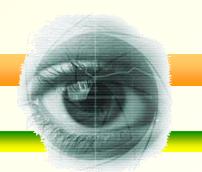
6	7	7	6	5
7	7	8	6	5
5	5	6	7	6
0	1	2	0	1
1	0	0	2	0

Última iteración



# Métodos de Segmentación Clásicos

## Basados en Regiones



### Segmentación basada en el crecimiento de regiones

#### Consideraciones

Cuando se elige más de una semilla por región es necesario **unir o fusionar** algunas de las regiones obtenidas. Está claro que las regiones a unir deben de ser adyacentes y tener propiedades estadísticas similares, como pueden ser la intensidad (nivel de gris) media de la región y su desviación típica. Por ejemplo, unas reglas de unión pueden ser:

$$|m_i - m_j| < ks_i$$

Y

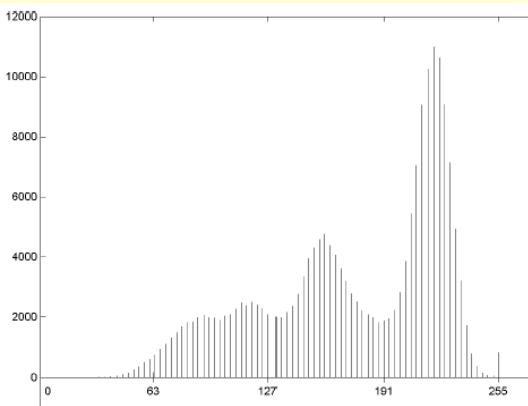
$$|m_i - m_j| < ks_j$$



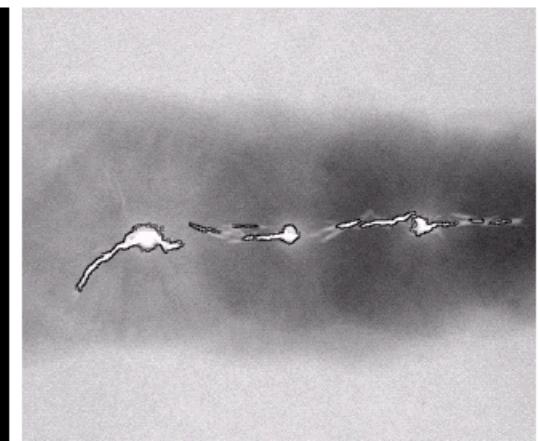
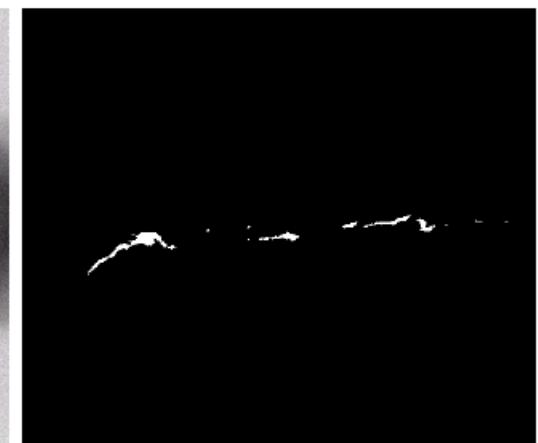
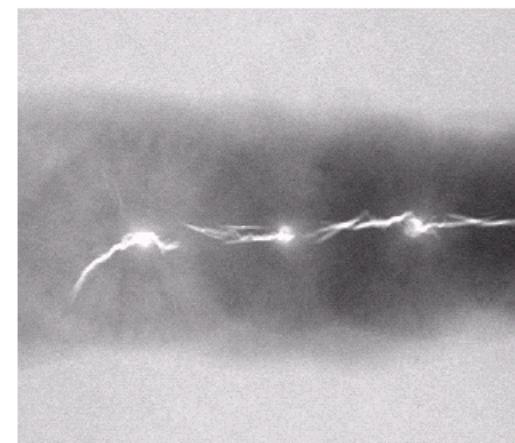
# Métodos de Segmentación Clásicos Basados en Regiones



## Segmentación basada en el crecimiento de regiones. Ejemplo

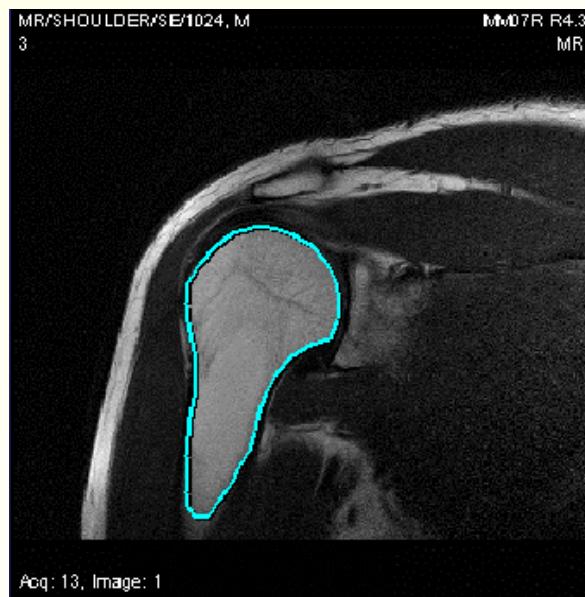
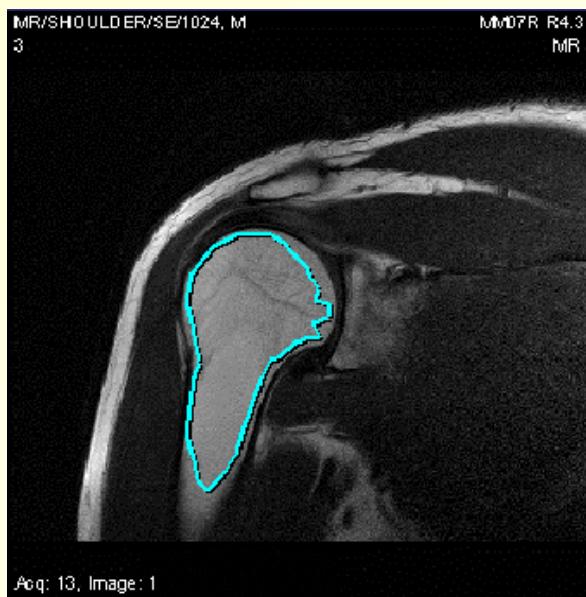
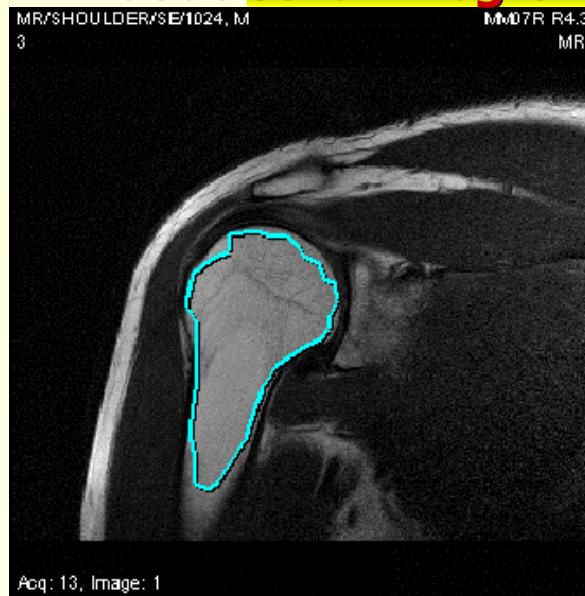
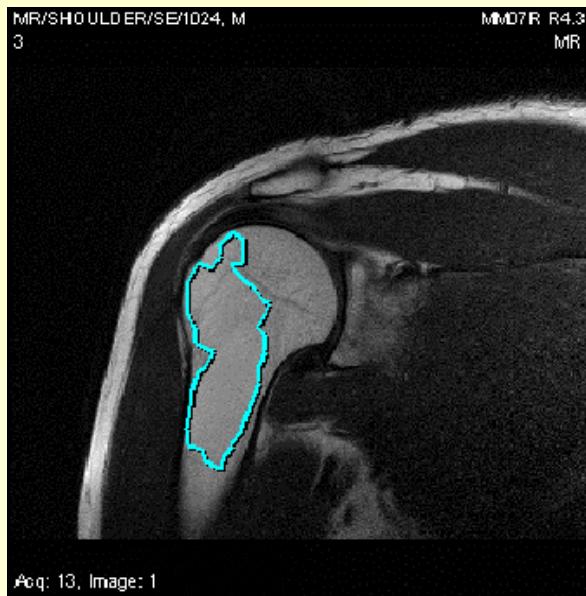


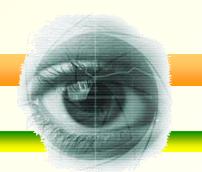
- (a)Imagen original y su histograma.
- (b)Puntos semilla
- (c)Resultado de crecimiento de regiones.
- (d)Bordes de las regiones segmentadas (en negro).



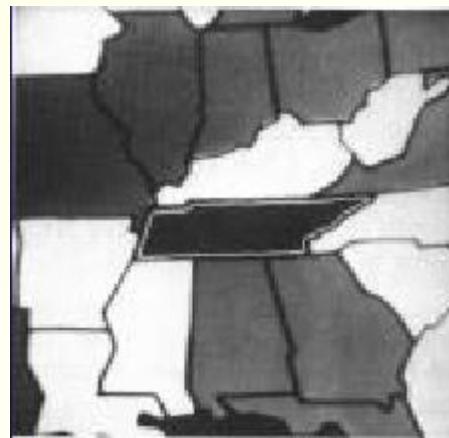
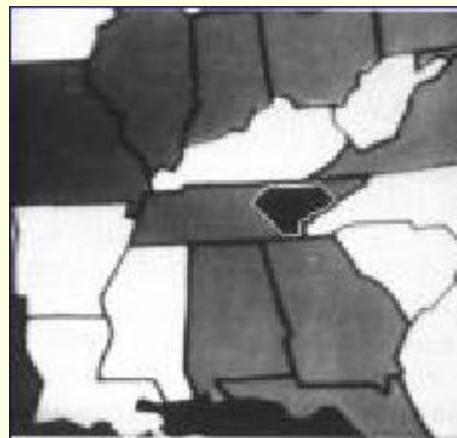
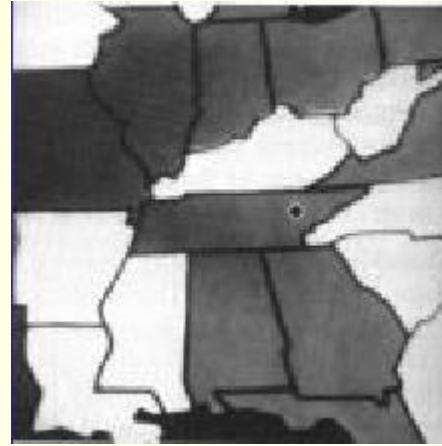
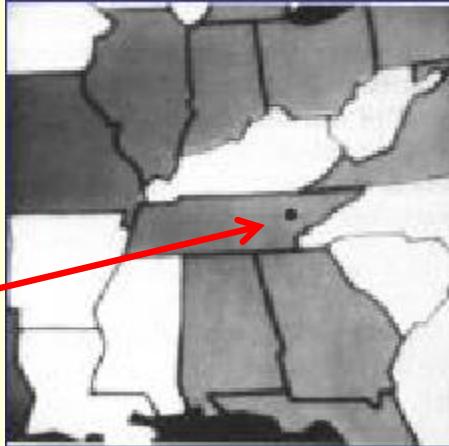


# Métodos de Segmentación Clásicos Basados en Regiones





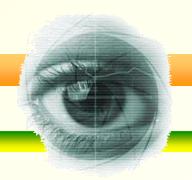
### Segmentación basada en el crecimiento de regiones. Ejemplo





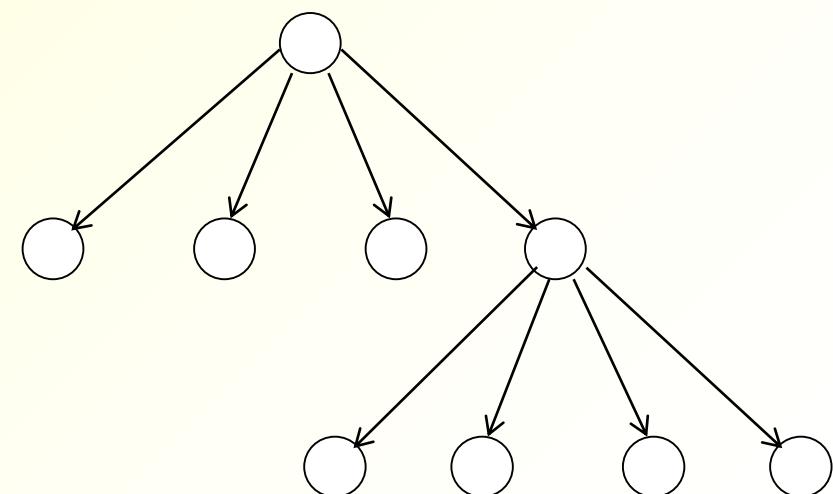
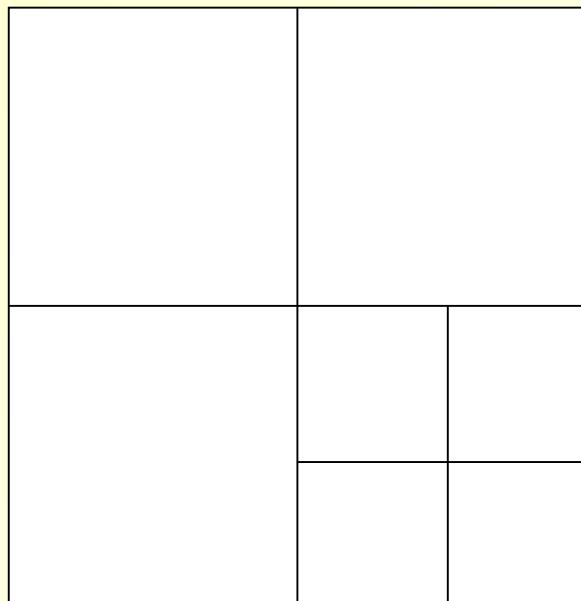
# Métodos de Segmentación Clásicos

## Basados en Regiones

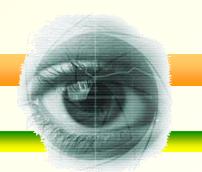


### Segmentación basada en división y fusión

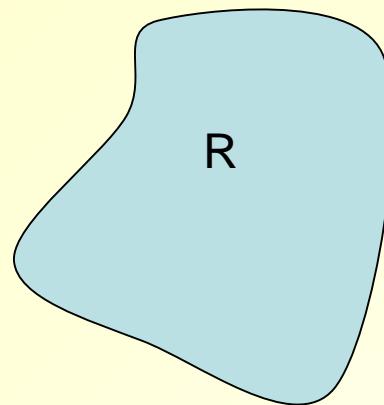
Los procedimientos de crecimiento de regiones parten de un conjunto de píxeles semilla. Un método alternativo se obtiene actuando al revés, es decir, partiendo de la imagen completa se van realizando **sucesivas divisiones** de la misma. Cada subimagen obtenida que no sea homogénea se vuelve a dividir y así sucesivamente.



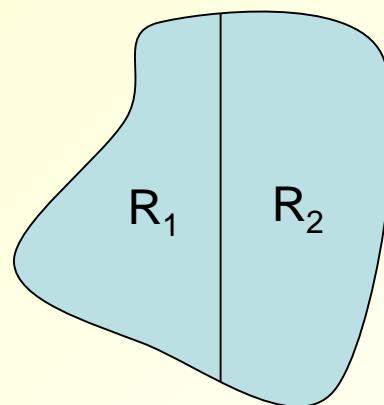
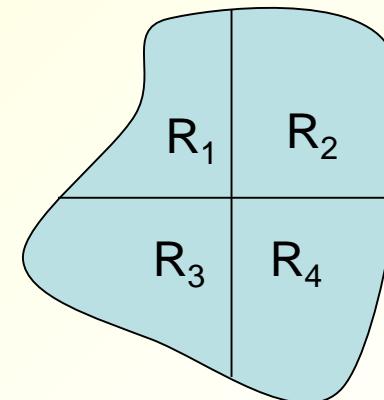
**Árbol cuaternario**



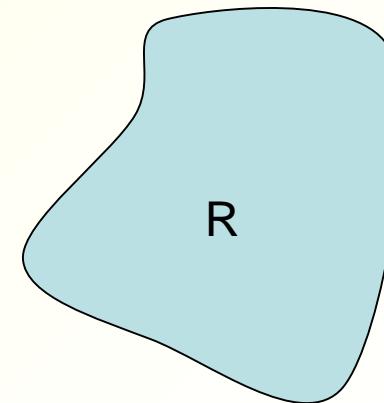
### Segmentación basada en división y fusión



$P(R)$

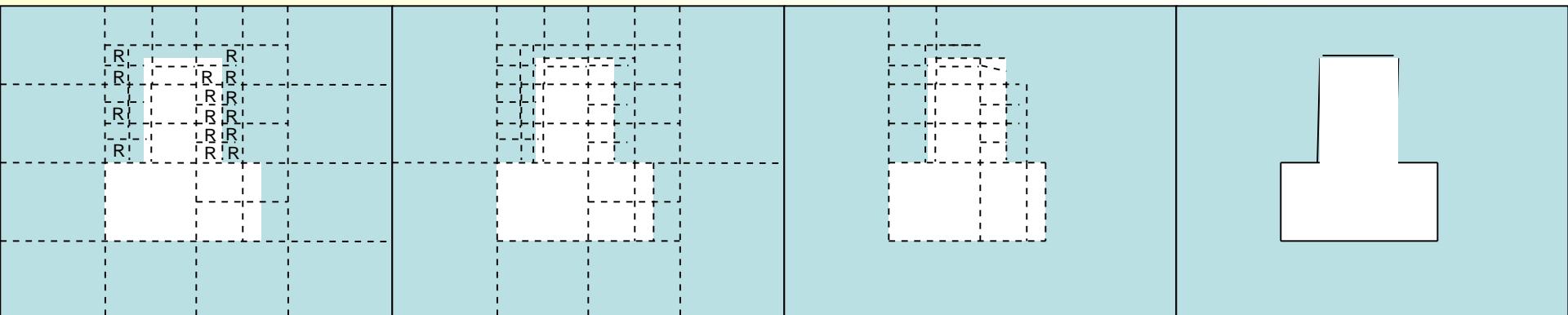
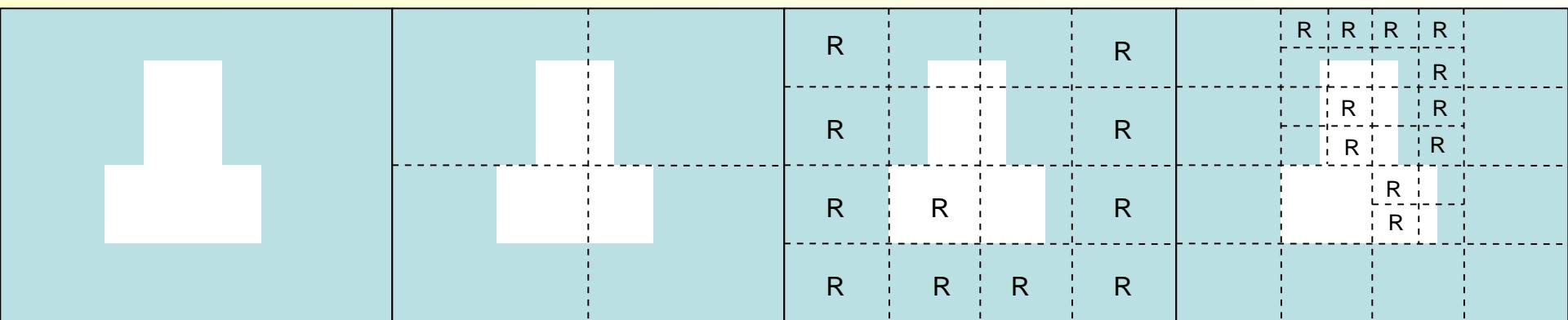


$P(R_1 \cup R_2)$





### Segmentación basada en división y fusión. Ejemplo





# Métodos de Segmentación Clásicos

## Basados en Regiones

*Segmentación basada en división y fusión.*

*Ejemplo*

1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0
1	1	8	8	8	4	1	0
1	1	6	6	6	3	1	0
1	1	5	6	6	3	1	0
1	1	5	6	6	2	1	0
1	1	1	1	1	1	0	0

Imagen inicial



# Métodos de Segmentación Clásicos

## Basados en Regiones

*Segmentación basada en división y fusión.*

*Ejemplo*

1	1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0	
1	1	8	8	8	4	1	0	
1	1	6	6	6	3	1	0	
1	1	5	6	6	3	1	0	
1	1	5	6	6	2	1	0	
1	1	1	1	1	1	1	0	0

Primera división



Segmentación basada en división y fusión.

Ejemplo

1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0
1	1	8	8	8	4	1	0
1	1	6	6	6	3	1	0
1	1	5	6	6	3	1	0
1	1	5	6	6	2	1	0
1	1	1	1	1	1	0	0

Segunda división

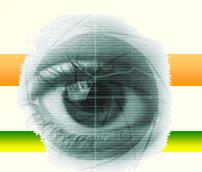


Segmentación basada en división y fusión.

Ejemplo

1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0
1	1	8	8	8	4	1	0
1	1	6	6	6	3	1	0
1	1	5	6	6	3	1	0
1	1	5	6	6	2	1	0
1	1	1	1	1	1	0	0

Tercera división



Segmentación basada en división y fusión.

### Ejemplo

1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0
1	1	8	8	8	4	1	0
1	1	6	6	6	3	1	0
1	1	5	6	6	3	1	0
1	1	5	6	6	2	1	0
1	1	1	1	1	1	0	0

Fusión



Segmentación basada en división y fusión.

### Ejemplo

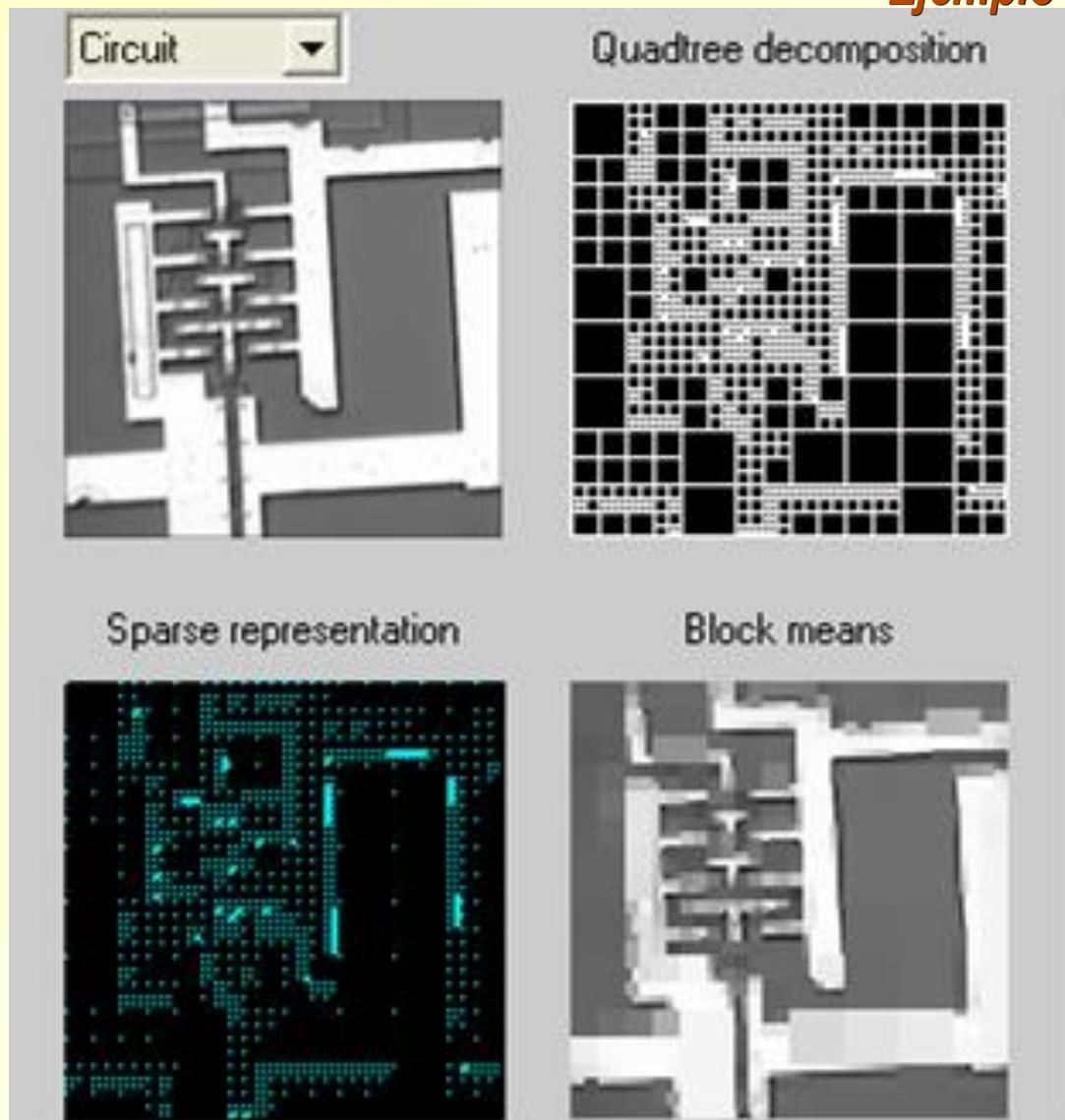
1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0
1	1	8	8	8	4	1	0
1	1	6	6	6	3	1	0
1	1	5	6	6	3	1	0
1	1	5	6	6	2	1	0
1	1	1	1	1	1	0	0

•Resultado final



*Segmentación basada en división y fusión.*

*Ejemplo*



Ejemplo de Segmentación  
basada en **división y fusión**  
de regiones

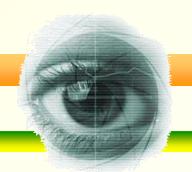


# Métodos de Segmentación Clásicos Basados en Regiones



Rice	Quadtree decomposition
Sparse representation	Block means

Segmentación basada en **división** y **fusión** de regiones



### Consideraciones

- El **algoritmo split-and-merge** es una forma muy simple de encontrar una segmentación de una imagen que proporciona homogeneidad dentro de las regiones.
- Sin embargo, comenzar el algoritmo desde la **imagen completa** e ir dividiendo en **cuadrantes** jerárquicamente, puede conducir a resultados no deseados.
- A menudo es más apropiado identificar **regiones candidatas** en la imagen y usarlas como unidades de partida en el algoritmo split-and-merge.

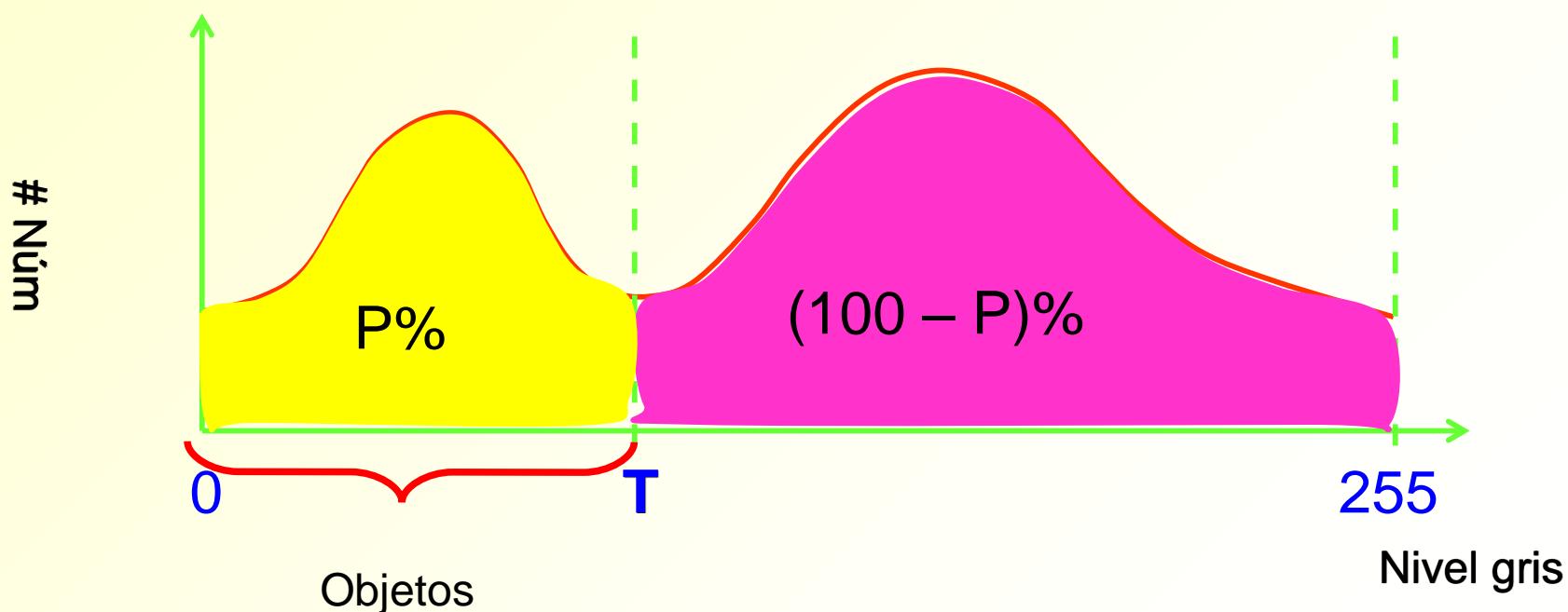


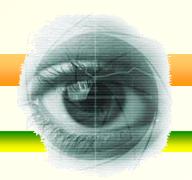
### Proporción de objetos en la imagen. Método P-Til (Percentil)

#### Información a Priori:

Estimación de la proporción de la imagen que ocupa los objetos que queremos segmentar.

Se basa en percentiles que se deducen del histograma



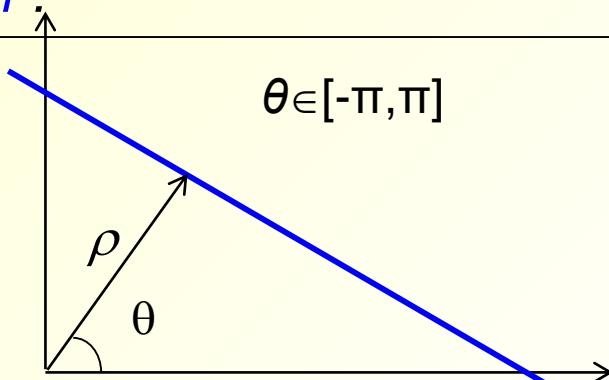


### Transformada de Hough (basados en modelos)

Las técnicas basadas en la transformación de Hough se pueden utilizar con éxito en segmentación de objetos solapados o parcialmente ocluidos.

#### DETECCIÓN DE RECTAS:

Sea  $n$  el número de puntos de aristas detectados. Se determinan las  $n(n-1)/2$  rectas (a lo sumo) que pasan por dos o más puntos y se comprueban los puntos que verifican la ecuación de cada recta. El número de operaciones elementales que se requiere para ello es de orden  $n^3$ .



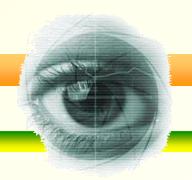
Ecuación Polar  
 $\rho = x \cos \theta + y \sin \theta$

Ecuación Paramétrica  
 $y = ax + b$



# Métodos de Segmentación Clásicos

## Basados en Aristas



### Transformada de Hough. Detección de Líneas

Una línea recta puede describirse mediante la ecuación paramétrica:

$$y = ax + b$$

Para un valor dado ( $x, y$ ) determinamos los posibles valores de  $a$  y  $b$  reescribiendo la ecuación anterior como:

$$b = -xa + y$$

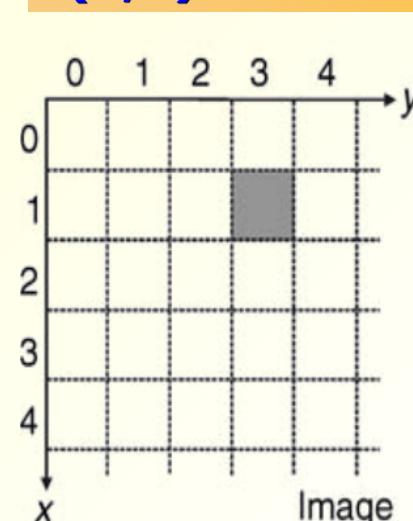
que representa la ecuación de una línea recta en el espacio  $a-b$  de salida.

**Espacio de Entrada :**

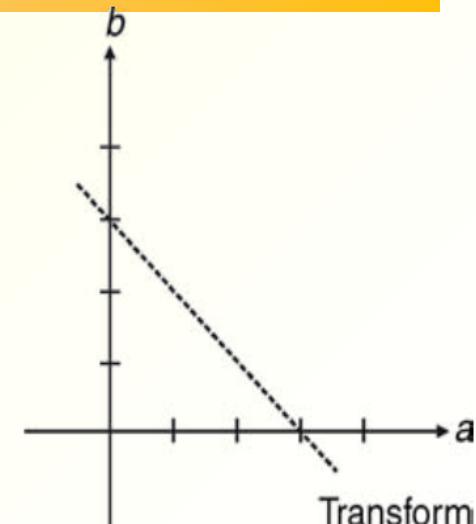
**Los píxeles frontera (x,y)**

**Espacio de Salida :**

**(a,b). Parámetros de la recta**



$$3 = a + b \quad \longleftrightarrow \quad b = -a + 3$$





# Métodos de Segmentación Clásicos

## Basados en Aristas



### Transformada de Hough. Detección de Líneas

Una recta  $y=ax+b$  queda identificada por los parámetros  $(a,b)$ , de forma que en el espacio de parámetros la recta se representa por un punto

**Espacio de Entrada : Un punto  $(x,y)$**

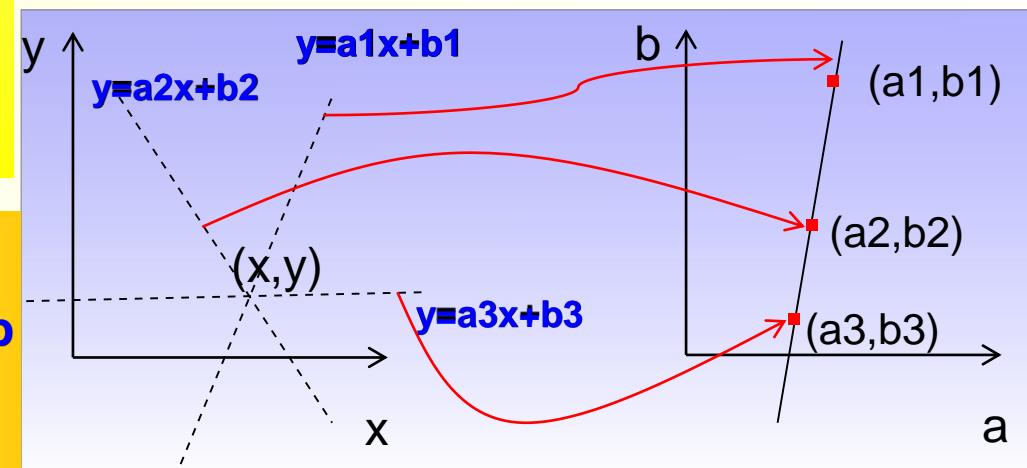
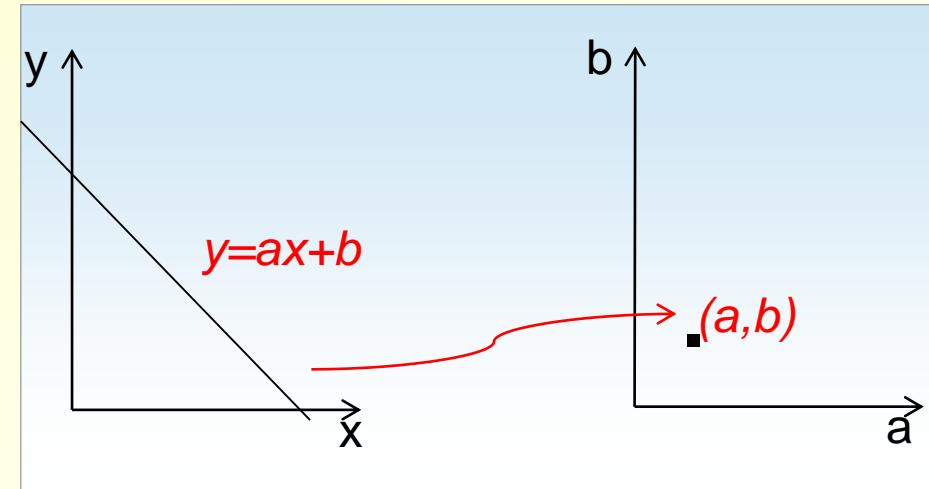


**Espacio de Salida: Una recta**  
 $b=-ax+y$

**Espacio de Entrada : Un recta**  
 $y=ax+b$



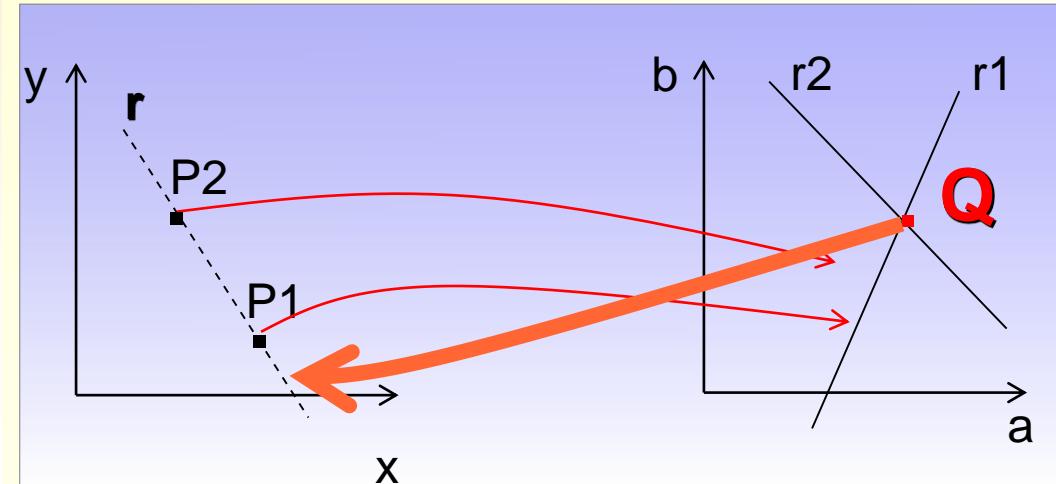
**Espacio de Salida: Un punto  $(a,b)$**





### Transformada de Hough. Detección de Líneas

$P_1 = (x_1, y_1)$  define una recta  $r_1$  en el Espacio Transformado  
 $P_2 = (x_2, y_2)$  define otra recta  $r_2$  en el Espacio Transformado  
La intersección de  $r_1$  y  $r_2$  en  $Q$  define la recta  $r$  en el espacio de entrada.  $r$  pasa por  $P_1$  y por  $P_2$ .

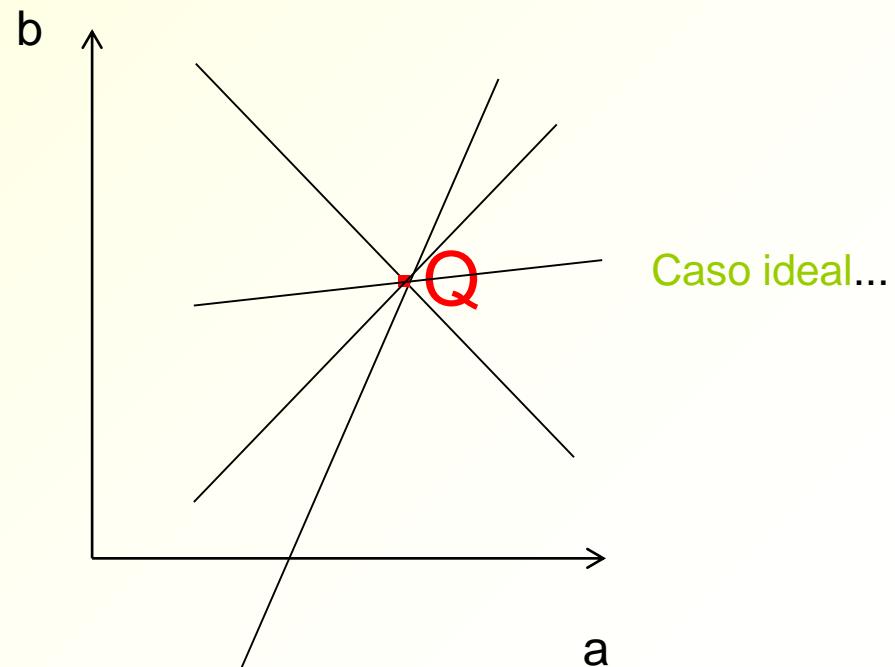
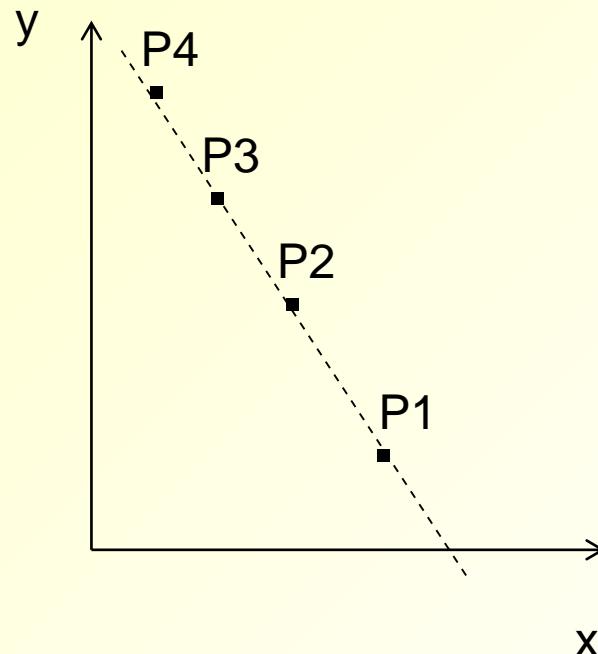


Dos puntos alineados en una recta  $r$ , corresponden en el espacio de salida a dos rectas cuya intersección es un punto  $Q=(a,b)$  que nos da los parámetros de definición de la recta  $r$  en el espacio de entrada



### Transformada de Hough. Detección de Líneas

En el caso ideal, una recta en el espacio de partida que venga definida por n puntos  $P_1, \dots, P_N$  viene identificada en el espacio de parámetros por N rectas (cada una correspondiente a un punto  $P_i$ ) que se cortan en un punto  $Q=(a,b)$  que determina los parámetros que definen la recta que une los  $P_i$ .





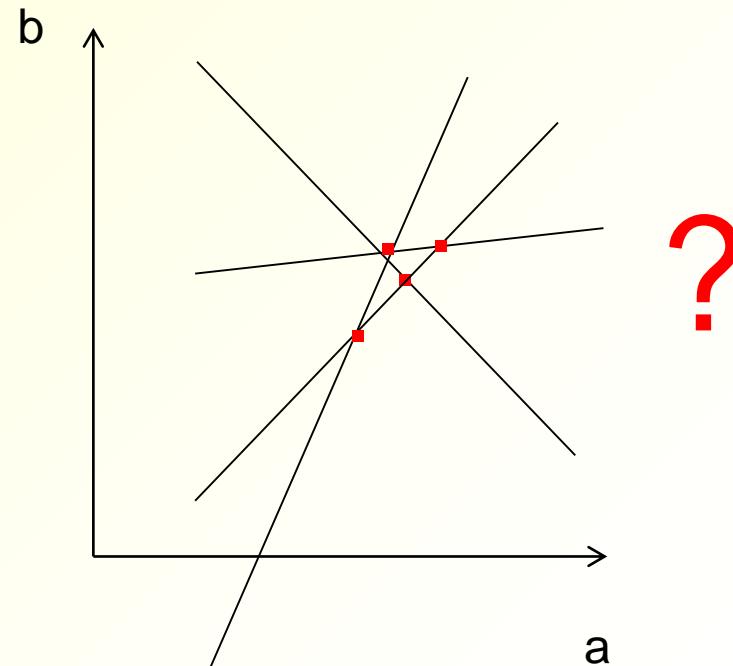
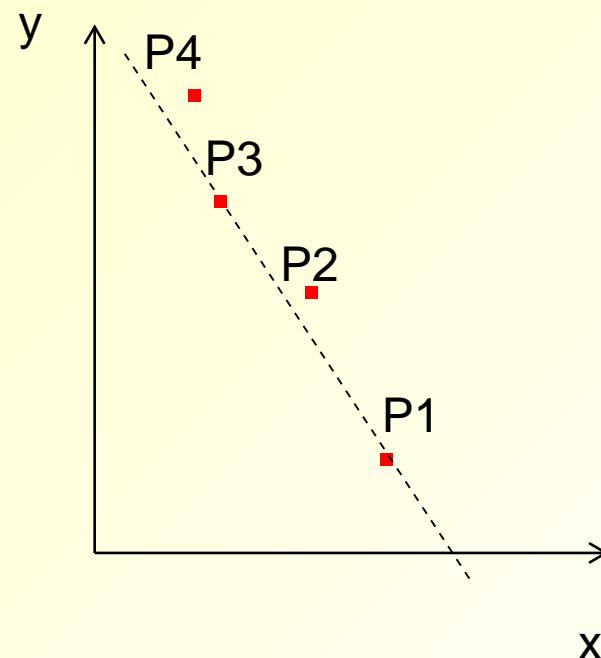
# Métodos de Segmentación Clásicos

## Basados en Aristas



### Transformada de Hough. Detección de Líneas

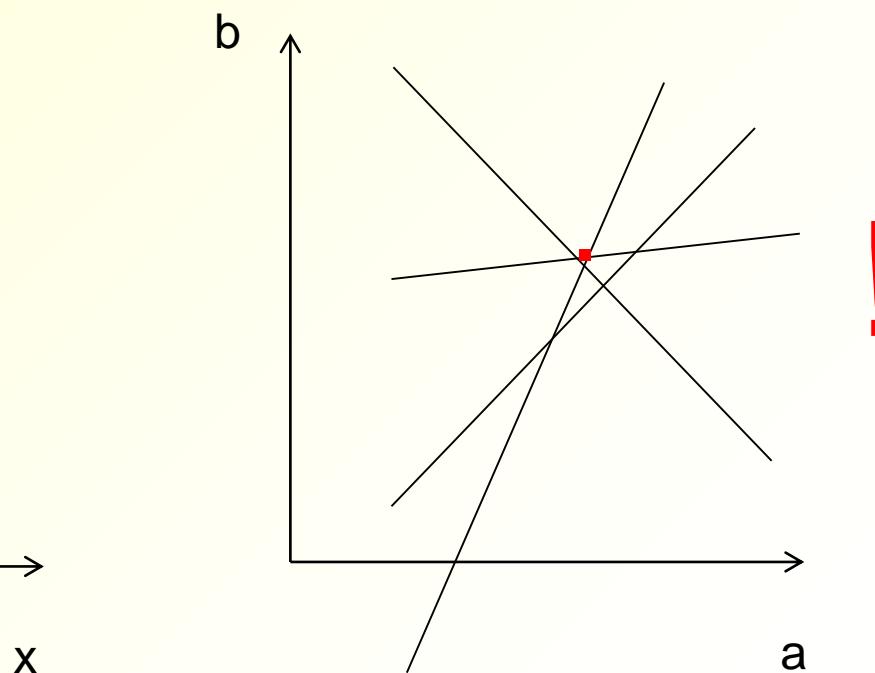
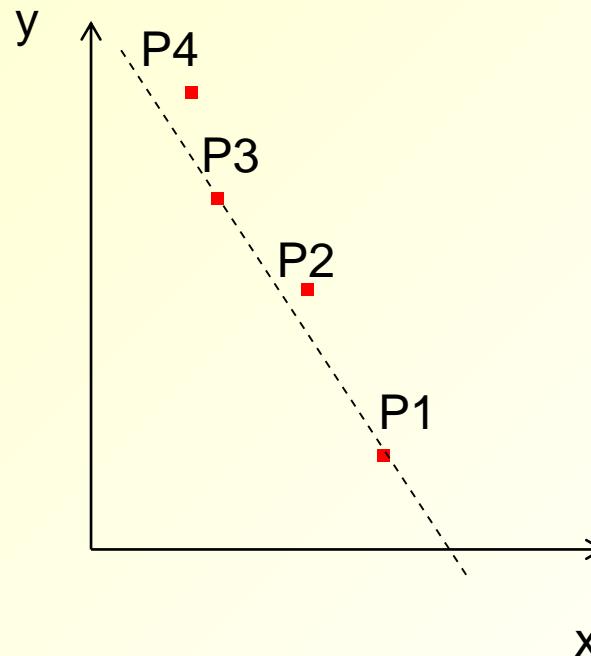
En un caso real, los puntos  $P_i$  no están perfectamente alineados de forma que la intersección en el espacio de parámetros no tiene por qué ser única.





### Transformada de Hough. Detección de Líneas

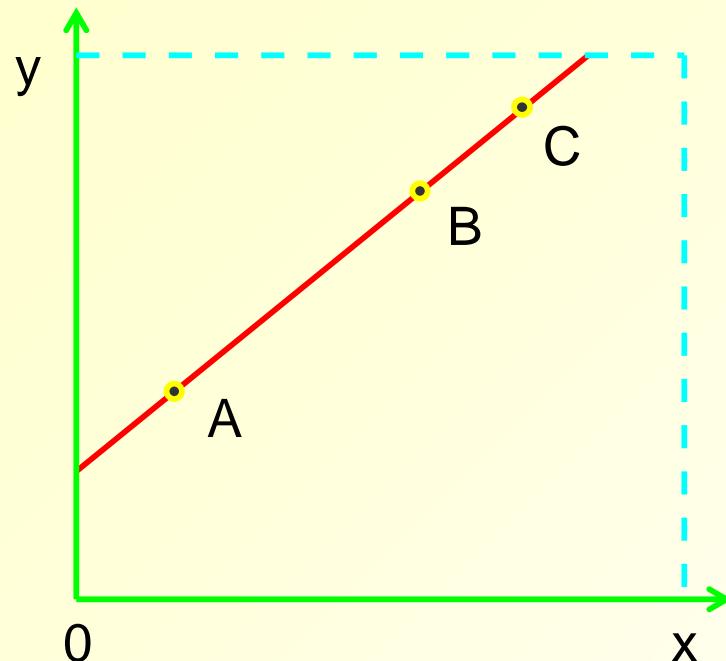
Si se tienen suficientes puntos en el espacio de partida, el problema de detectar rectas puede trasladarse a un problema de detectar picos en el espacio de parámetros (puntos en que concurren un número maximal de rectas)



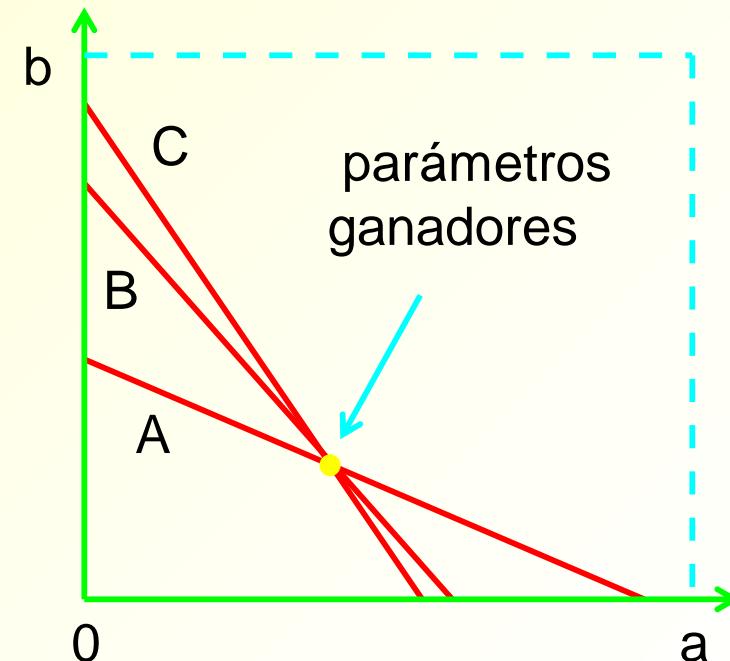


### Transformada de Hough. Detección de Líneas

**Ejemplo:** Cada uno de los 3 puntos A, B, y C sobre una línea recta en el espacio de entrada son transformados en líneas rectas en el espacio de salida.



Espacio de  
entrada



Espacio de  
salida

Los parámetros de su punto de corte (que podrían ser los ganadores en la votación) son los parámetros de la línea recta buscada en el espacio de entrada.



### Transformada de Hough. Detección de Líneas

#### Algoritmo de detección de líneas

##### Inicialización:

- .Cuantificar  $a$  y  $b$  en valores discretos "cubos"
- .De forma que  $a \in \{a_0, a_1, \dots, a_n\}$  , y  $b \in \{b_0, b_1, \dots, b_m\}$  ,
- . $H(a,b)$ : es la matriz acumuladora inicializada a cero.

##### Votación:

- .Para cada punto frontera  $(i,j)$ 
  - Para cada  $a \in \{a_0, a_1, \dots, a_n\}$  , y  $b \in \{b_0, b_1, \dots, b_m\}$
  - Si  $b = -aj + i$  entonces
    - $H(a,b) = H(a,b) + 1$

##### Maximos de $H$ :

Los máximos locales de  $H$  corresponden a las líneas más relevantes de la imagen de partida.



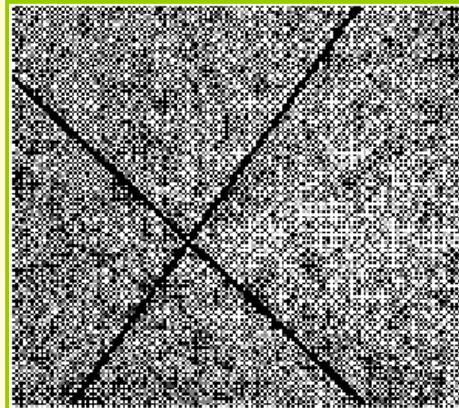
# Métodos de Segmentación Clásicos

## Basados en Aristas

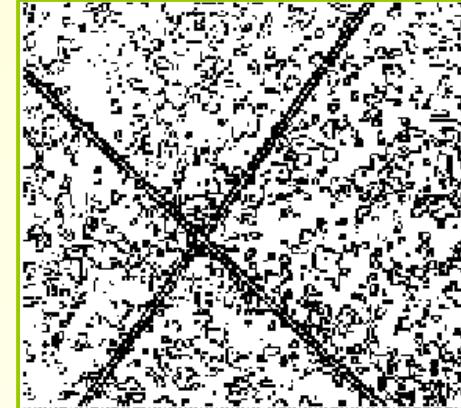
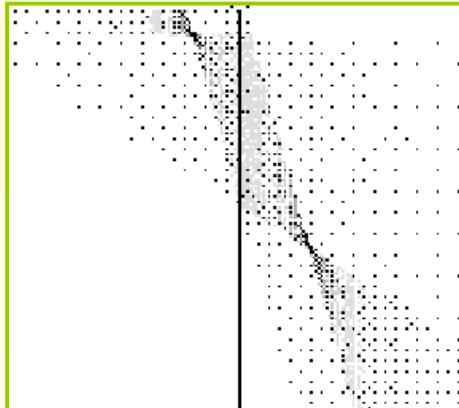


### Transformada de Hough. Detección de Líneas

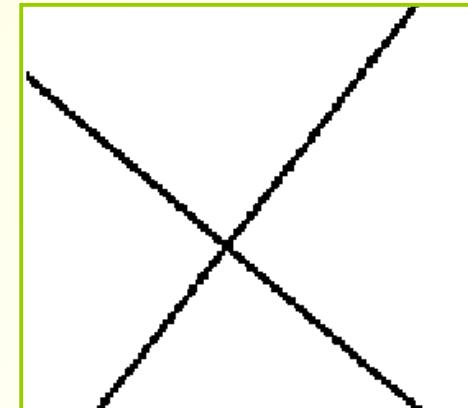
Imagen



Array  
Acumulador



Bordes



Resultado



# Métodos de Segmentación Clásicos

## Basados en Aristas

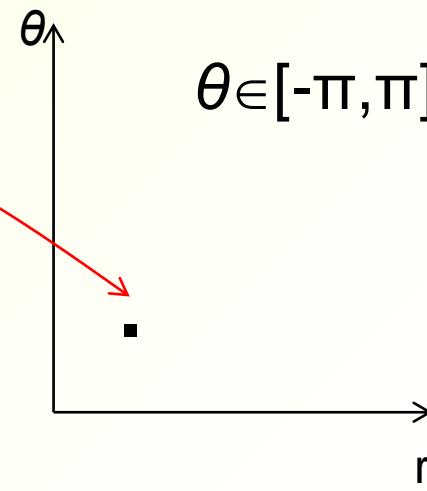
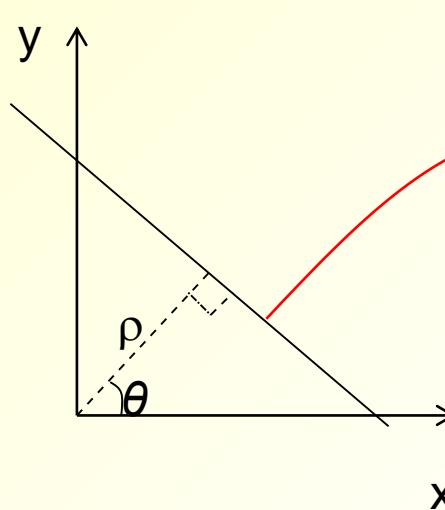


### Transformada de Hough. Detección de Líneas

Problema: ¿Qué pasa con las líneas verticales y la representación  $y=ax+b$ ? Las líneas verticales tiene una pendiente infinita.

Solución: Usar la representación normal de las líneas dada por  $(\rho, \theta)$ .  
 $\rho$ : la distancia perpendicular de la línea al origen  
 $\theta$ : ángulo entre la línea perpendicular y el eje horizontal.  
Y además  $\rho = x \cos \theta + y \sin \theta$ . Y el rango

$$0 \leq \rho \leq \sqrt{row^2 + col^2}, -\pi \leq \theta \leq \pi$$

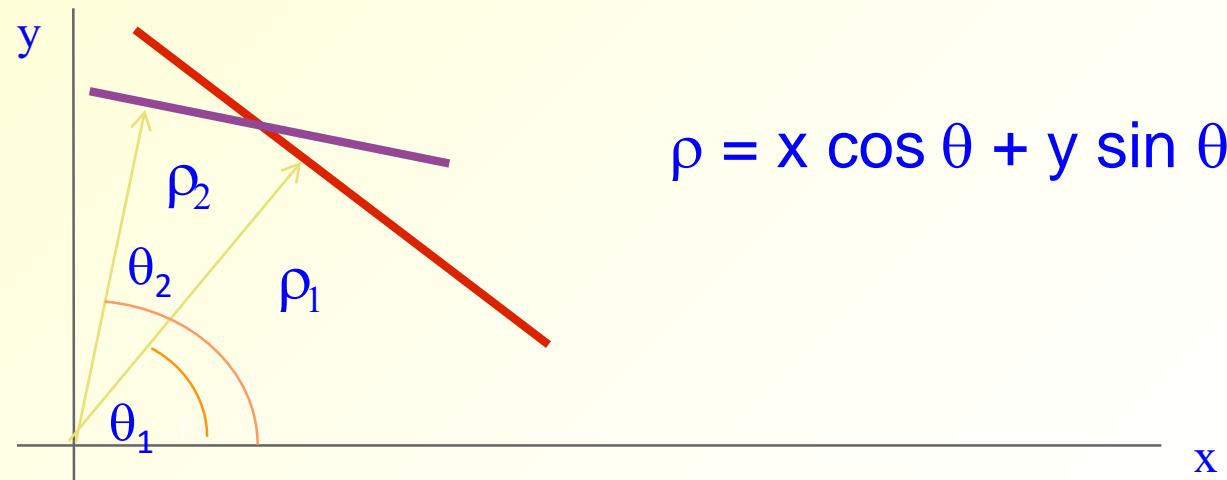




### Transformada de Hough. Detección de Líneas

Como las líneas verticales tienen pendientes infinitas es difícil cuantificar a. Para tener esto en cuenta, se usará una parameterización alternativa de una recta:

### Representación en coordenadas polares





# Métodos de Segmentación Clásicos

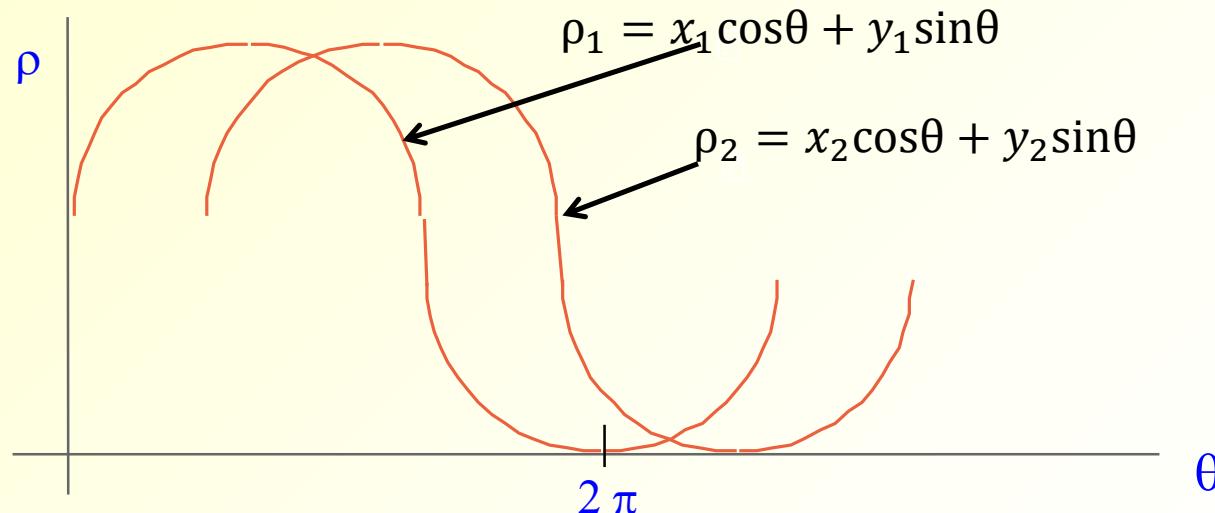
## Basados en Aristas



### Transformada de Hough. Detección de Líneas

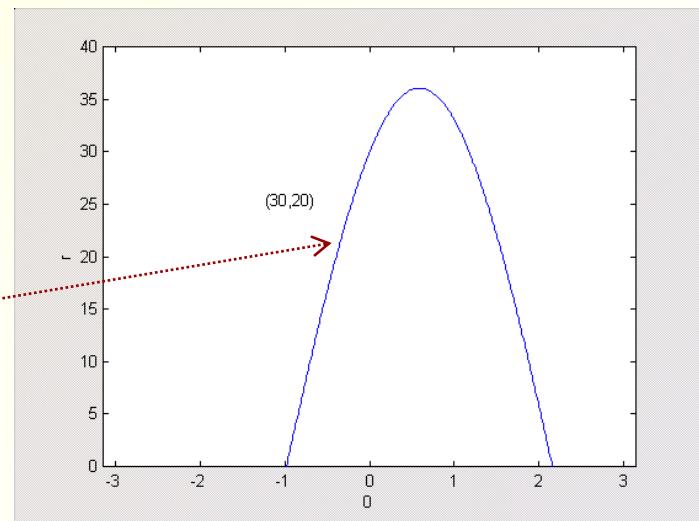
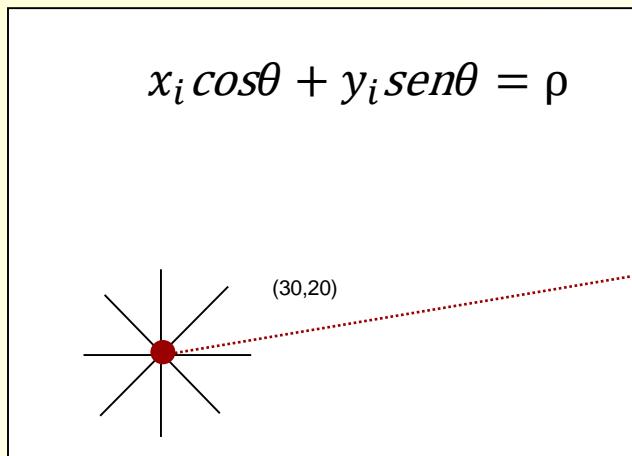
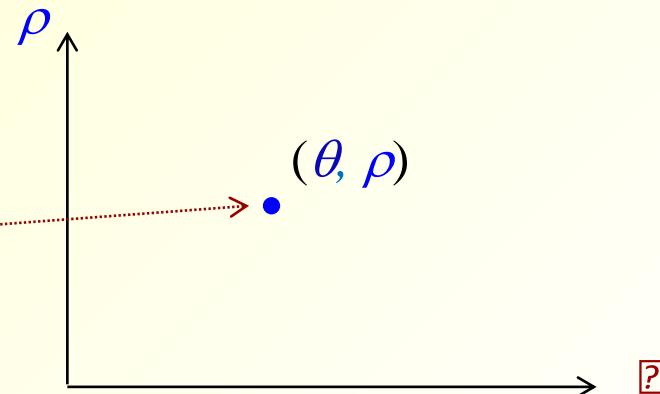
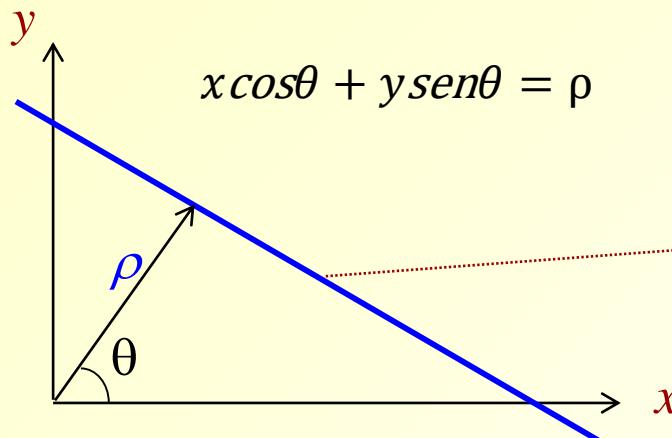
Una Curva en el espacio  $(\theta, \rho)$  es ahora una sinusoida

Pero el algoritmo sigue siendo válido.





### Transformada de Hough. Detección de Líneas



$$\{ (\theta, \rho) : 30 \cos \theta + 20 \sin \theta = \rho, \theta \in [-\pi, \pi] \text{ y } \rho \geq 0 \}$$

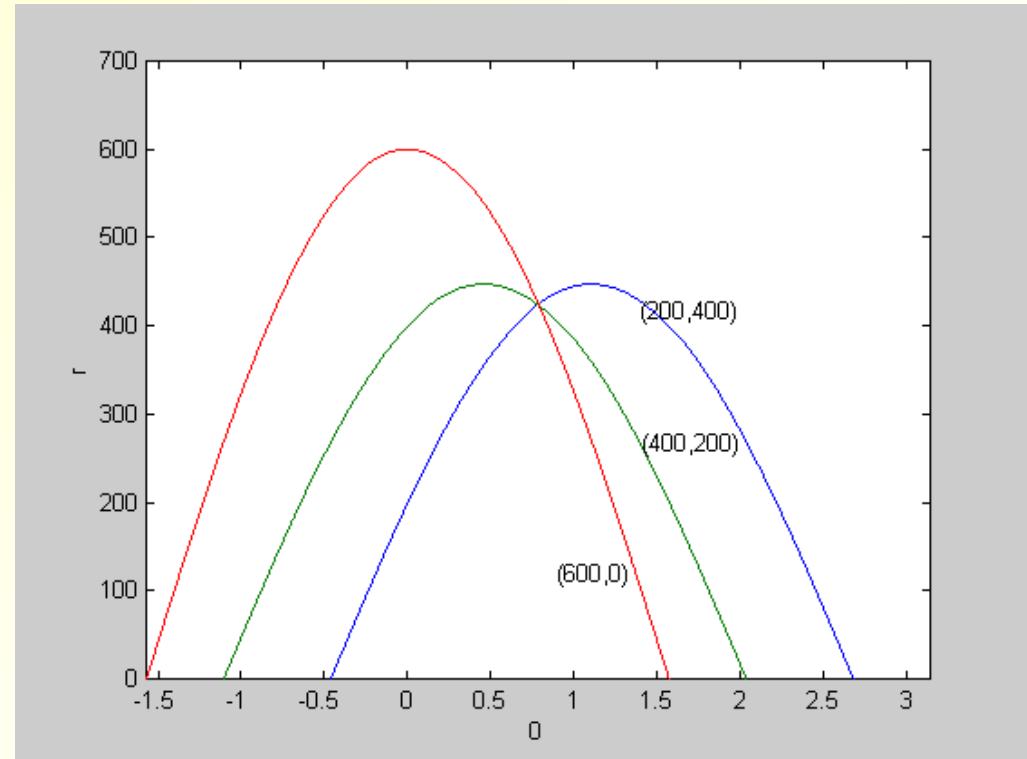
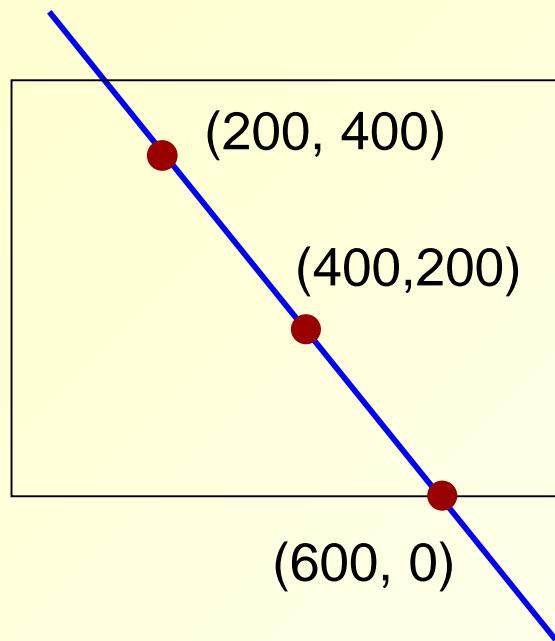


# Métodos de Segmentación Clásicos

## Basados en Aristas



### Transformada de Hough. Detección de Líneas



$$x_i \cos\theta + y_i \sin\theta = \rho, i = 1, 2, \dots, n$$



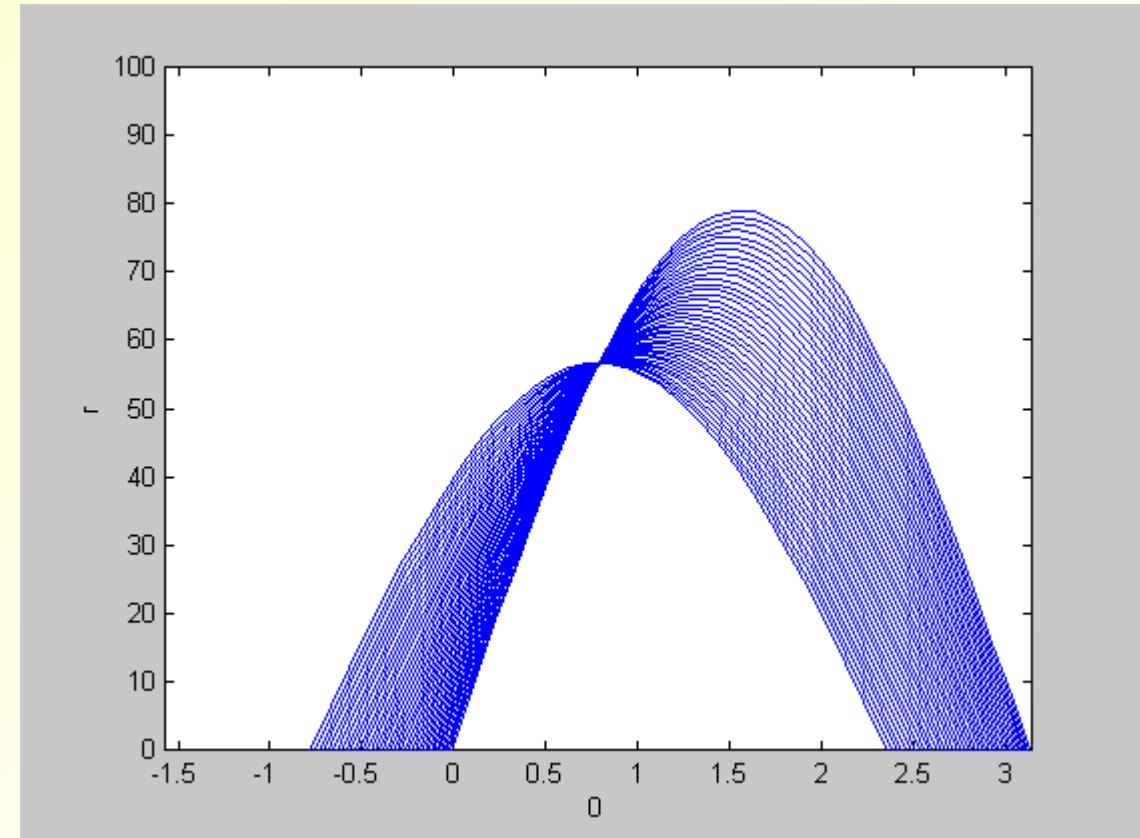
### Transformada de Hough. Detección de Líneas

1 . Puntos de la recta

$$y = -x + 80$$

$$x \cos(\pi/4) + y \sin(\pi/4)$$

$$= 56,5685$$



Puede observarse como todas las curvas sinusoidales se cortan en el punto  $(\pi/4, 56,5685)$

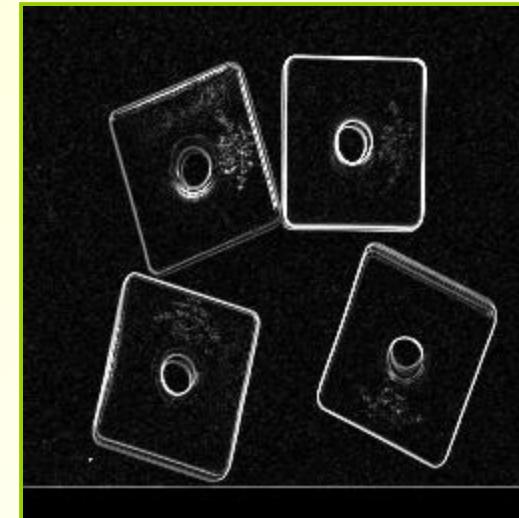


### Transformada de Hough. Detección de Líneas

Imagen



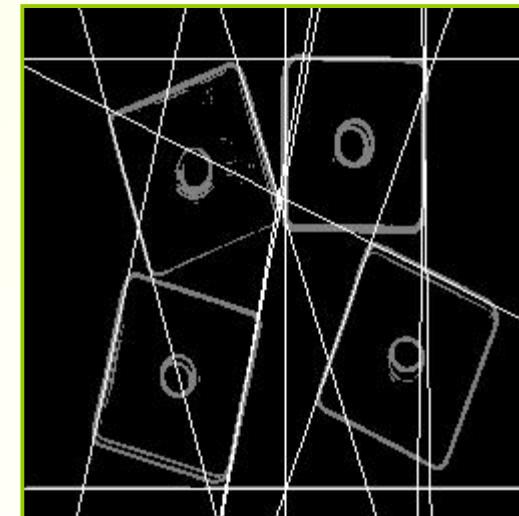
Bordes



Array  
Acumulador



Resultado





# Transformada de Hough

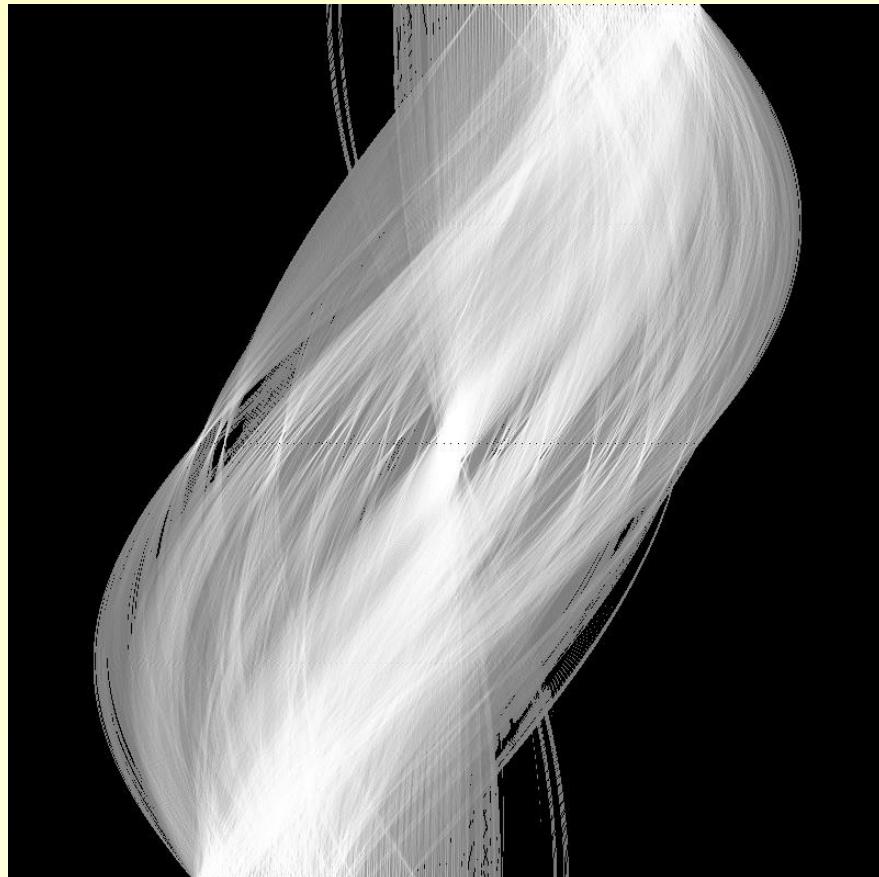
Otro ejemplo



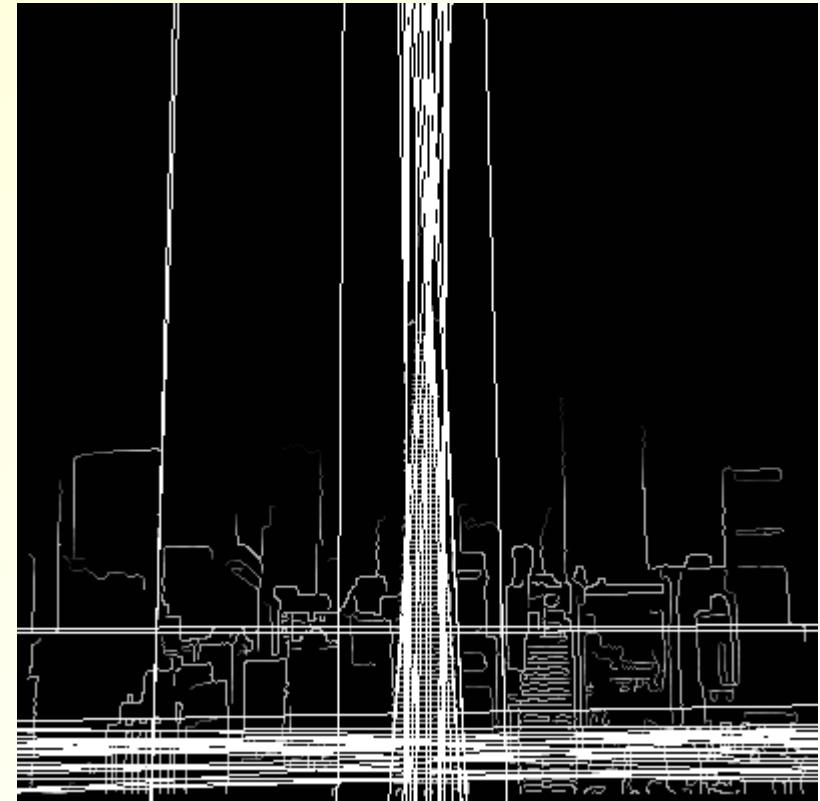
Imagen original



Imagen de fronteras



Array acumulador



Resultado de la detección

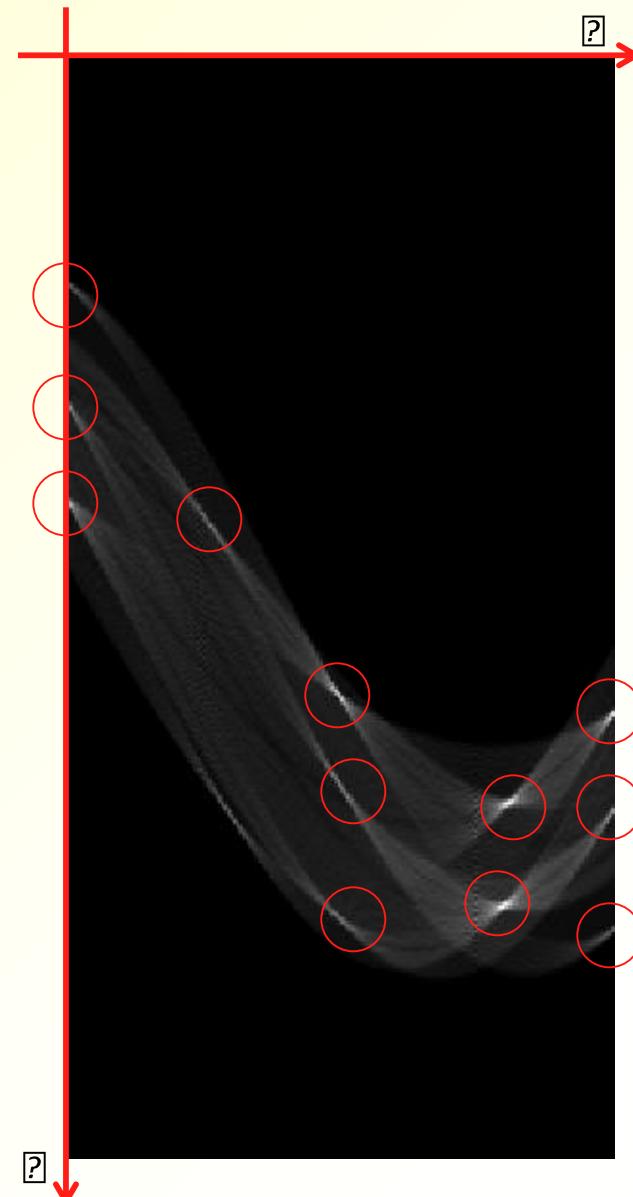
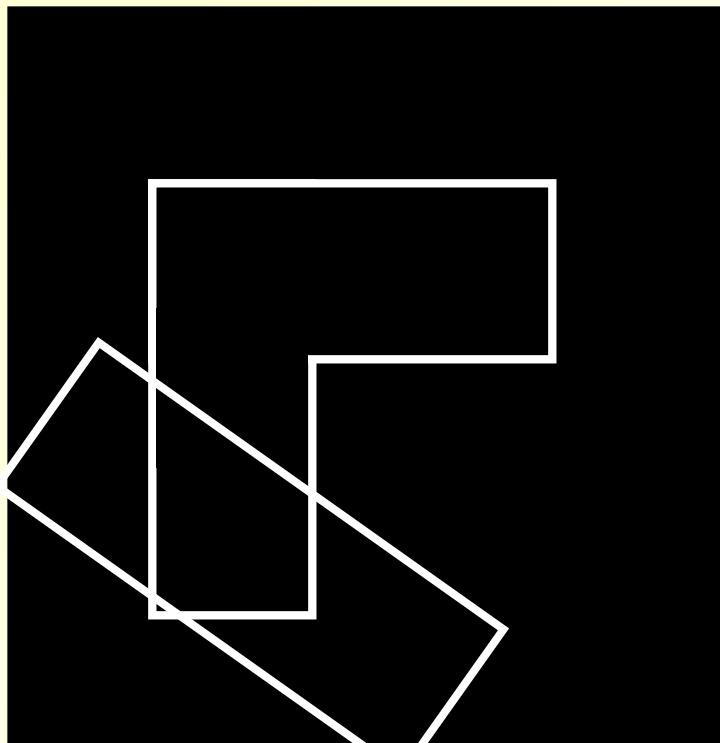


# Métodos de Segmentación Clásicos

## Basados en Aristas



### Transformada de Hough. Detección de Líneas



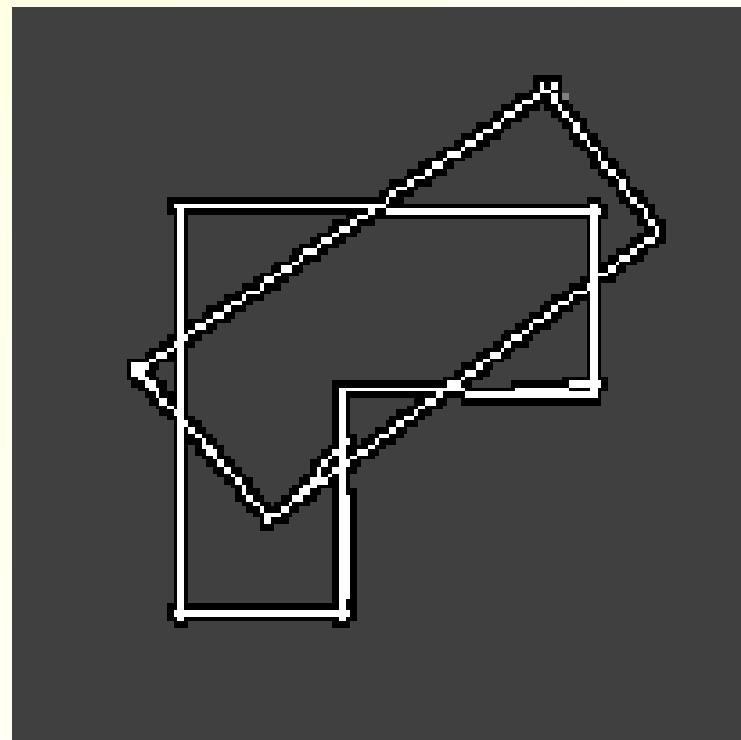
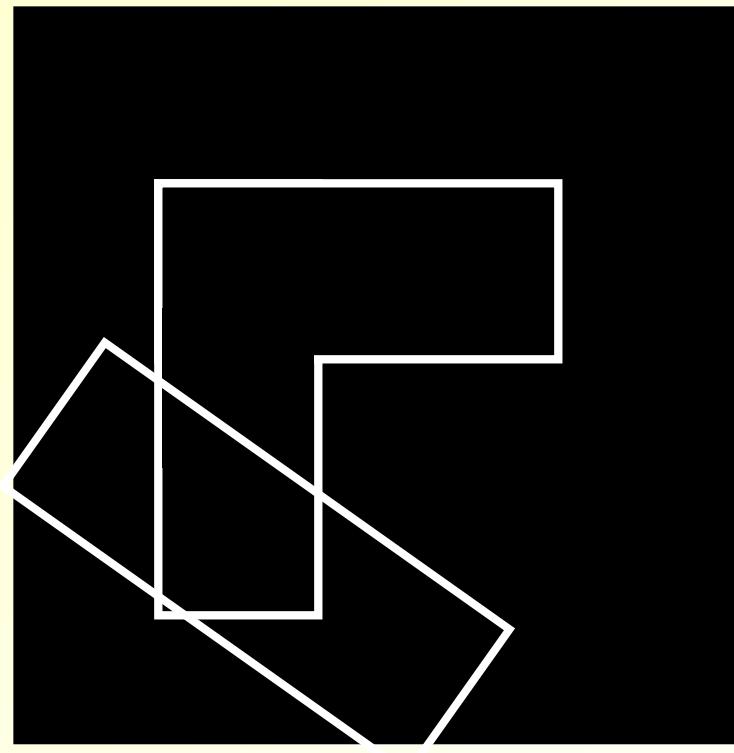


# Métodos de Segmentación Clásicos

## Basados en Aristas



### Transformada de Hough. Detección de Líneas





### Transformada de Hough.

La técnica de Hough se puede aplicar a cualquier curva que se pueda parametrizar:

$$f(x,a) = 0$$

vector de parámetros  
(ejes en el espacio de Hough)

El éxito depende del granulo de la cuantificación de los parámetros:

- .Demasiado gruesa: máximos empujados a estar juntos.
- .Demasiado fina: picos menos definidos.

La viabilidad de la aplicación de la técnica de Hough depende también del número de parámetros de la curva. Así que la posibilidad de aplicarla se restringe a curvas con pocos parámetros.



# Métodos de Segmentación Clásicos

## Basados en Aristas



### Transformada de Hough. Ejemplo para encontrar un círculo

Encontrar el centro de un círculo con radio conocido  $r$  dada una imagen de bordes sin información de la dirección del gradiente (solo localización de los bordes):

#### Planteamiento:

- .Parámetros de un círculo: Centro  $(a,b)$  y Radio  $r$
- .Ecuación del Círculo:  $f(x,y,r) = (x - a)^2 + (y - b)^2 - r^2 = 0$

#### Objetivo

- .Dados los puntos frontera de la imagen. ¿Dónde se encuentra el centro del círculo?.

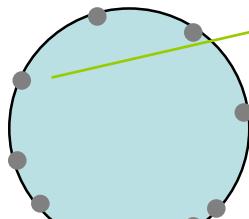
#### Como se define el array acumulador

- .Un array 3d:  $H(a,b,r)$ .
- Mecanismo a seguir: Incrementa  $H(a,b,r)$  cuando se analiza un punto
- Finalmente encontrar la máximos de  $H$ .



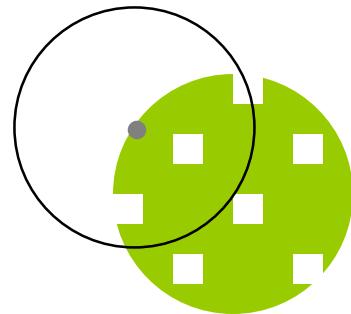
# Encontrando un círculo

Imagen

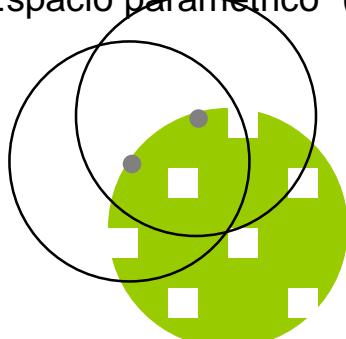


fijado (i,j)  
 $(i-a)^2 + (j-b)^2 - r^2 = 0$

Espacio paramétrico (a,b)

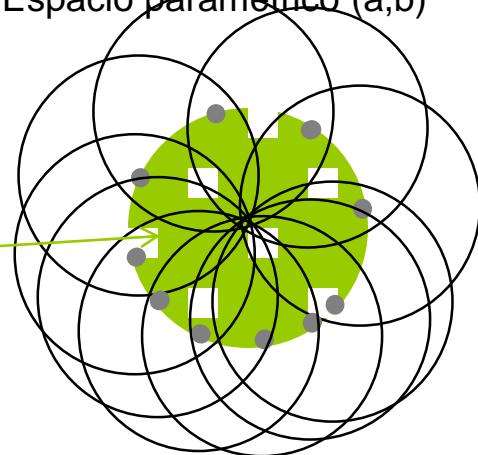


Espacio paramétrico (a,b)



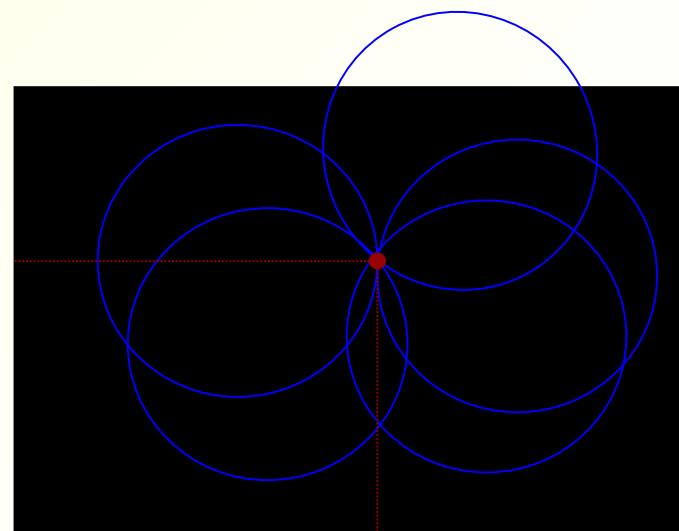
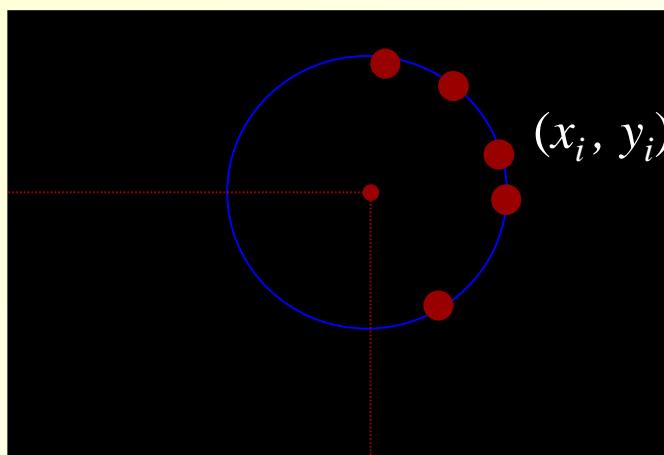
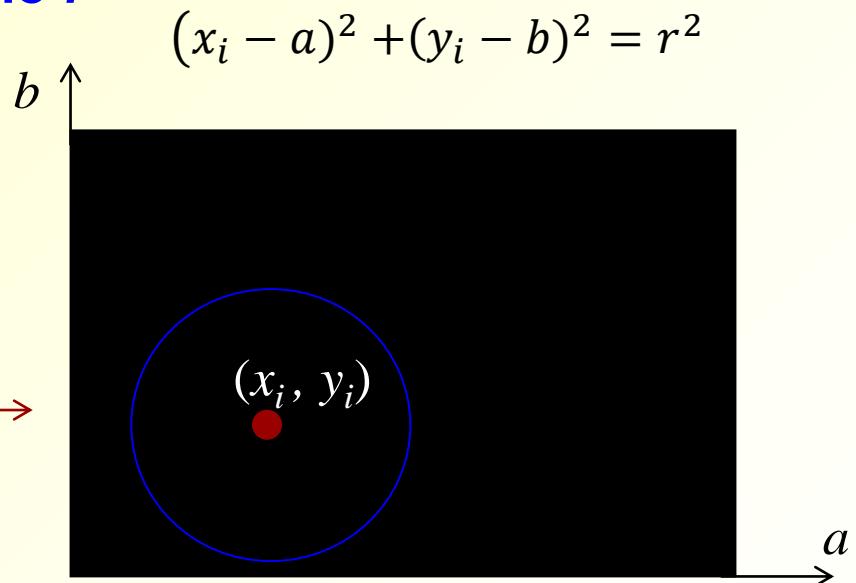
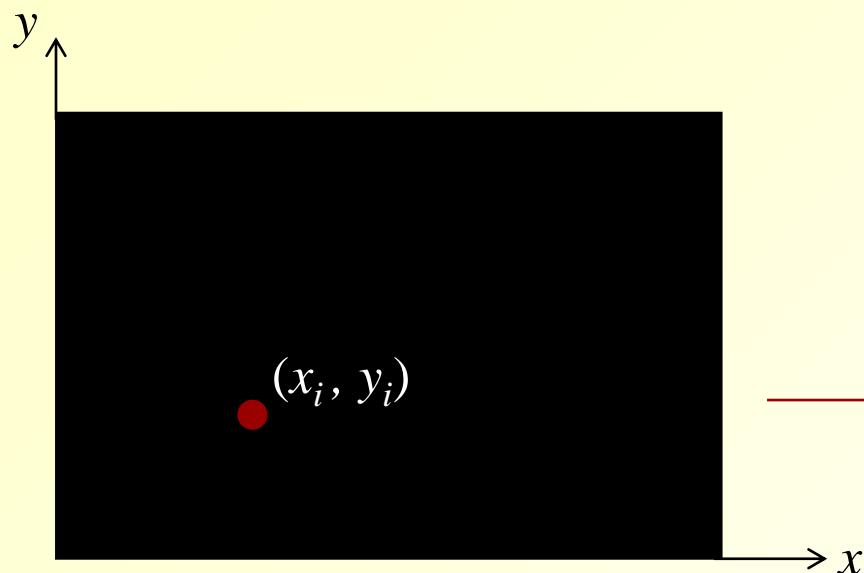
Centro del círculo  
(muchos votos!)

Espacio paramétrico (a,b)



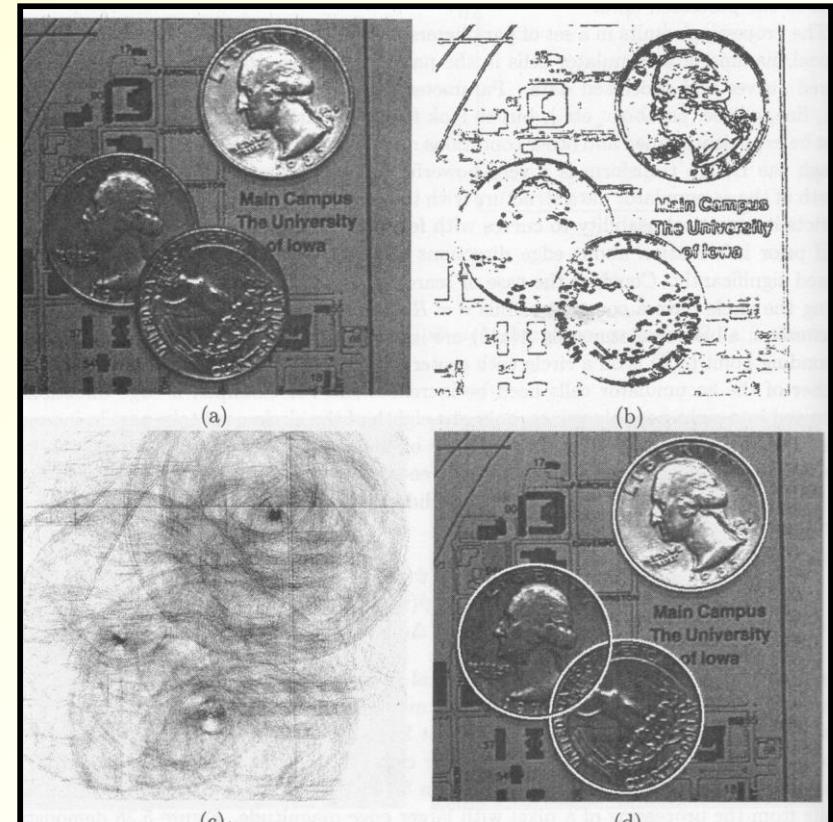
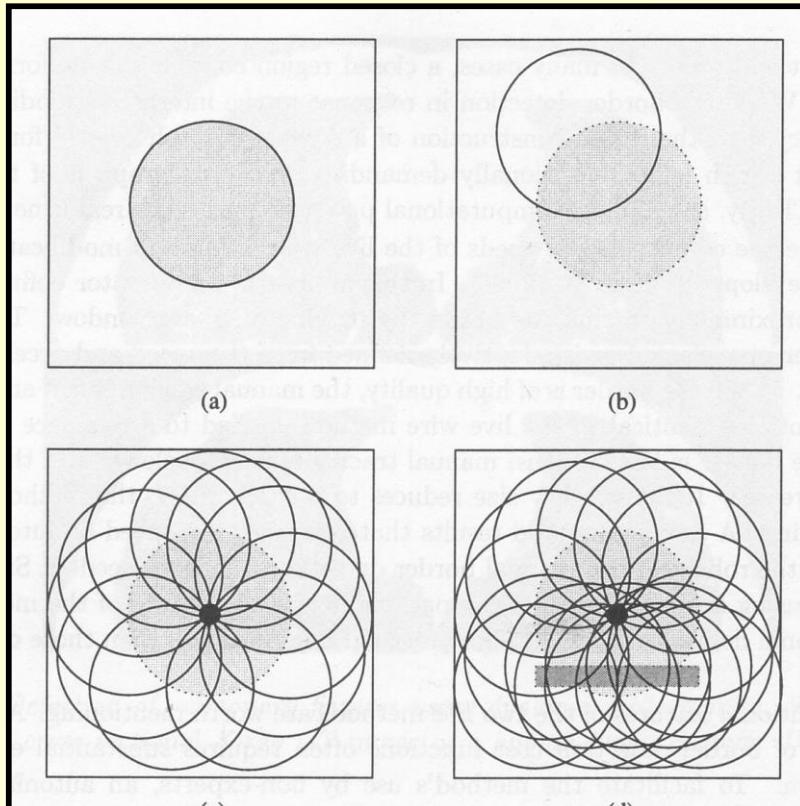
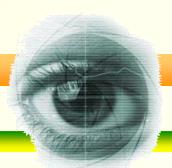


## DETECCIÓN DE CÍRCULOS de radio $r$





## Ejemplo





## ALGORITMO GENERICO:

1. **Cuantificar** el espacio de los parámetros.

1. **Iniciar** a cero los acumuladores  $A(\theta_1, \dots, \theta_p)$  obtenidos para los diferentes valores del vector paramétrico  $(\theta_1, \dots, \theta_p)$

2. Para cada píxel  $(x_i, y_i)$  de la imagen de bordes locales y para todos los valores de los parámetros especificados **incrementar** los acumuladores  $A(\theta_1, \dots, \theta_p)$  en una unidad,

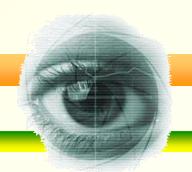
$$A(\theta_1, \theta_2, \dots, \theta_p) \leftarrow A(\theta_1, \theta_2, \dots, \theta_p) + 1$$

siempre que  $f(x_i, y_i, \theta_1, \dots, \theta_p) = 0$ ,

4. **Determinar los máximos locales** en la tabla de acumuladores y aplicarles el operador umbral. Aquellos acumuladores que superen el valor umbral nos dan los parámetros de las curvas detectadas que están presentes en la imagen original.



## *Generalización de la transformada*



Lecturas recomendadas:

- Ballard, D. H. Generalizing the Hough Transform to Detect Arbitrary Shapes, Pattern Recognition 13:111-122, 1981.
- Illingworth, J. and J. Kittler, Survey of the Hough Transform, Computer Vision, Graphics, and Image Processing, 44(1):87-116, 1988



## Segmentación

***Que es la segmentación de imágenes***

***Métodos de segmentación clásicos***

***Métodos de segmentación basados en Deep Learning***



*ugr*

Universidad  
de Granada

