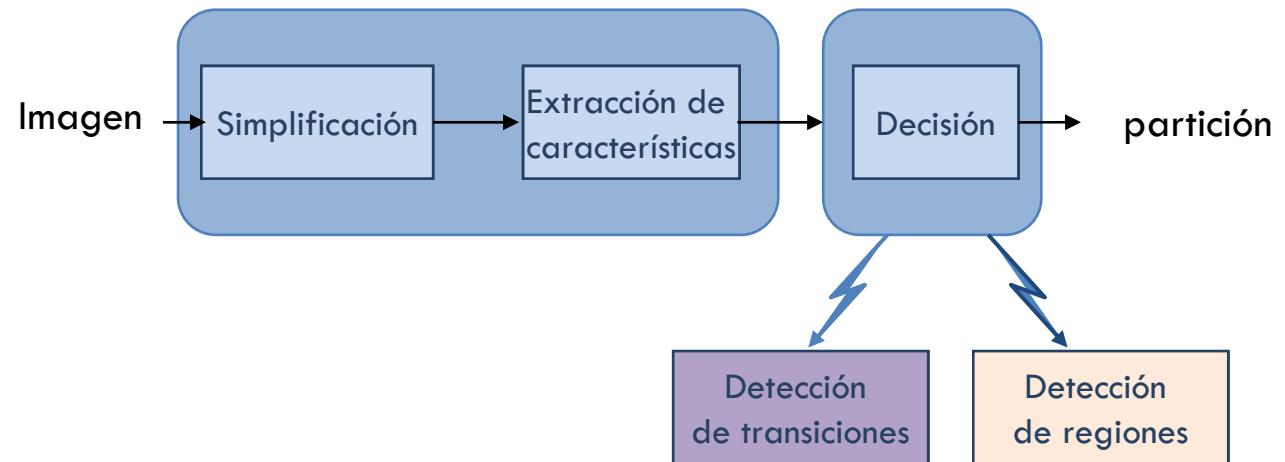


# Decisión: transiciones o regiones

55

Segmentación: 3 pasos



## □ 2. Decisión por detección de regiones

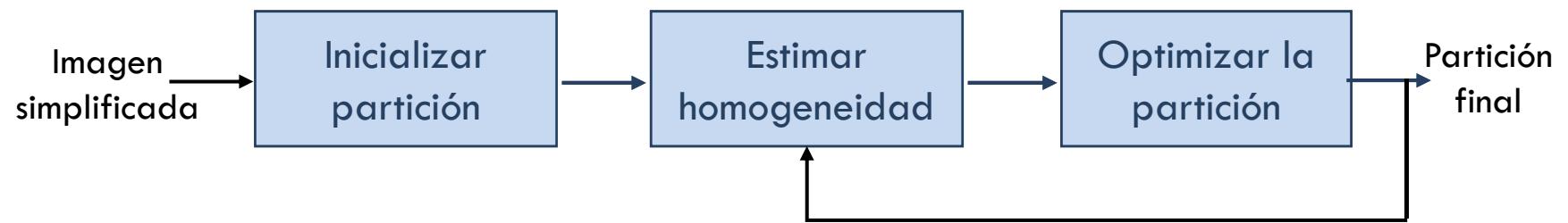
- Dos tipos de algoritmos:
  - 2.1 Segmentación en el espacio de características
    - Estudio y segmentación del espacio de características, Clasificación, Etiquetado
  - 2.2 Segmentación directa de la imagen
    - Fusión de regiones
    - Crecimiento de regiones



# Algoritmos de segmentación directa

56

Los algoritmos de segmentación directa de la imagen suelen ser algoritmos de decisión **iterativa**.



## Estrategia:

- Inicializar una partición
- Definir un criterio de homogeneidad para las regiones
- Optimizar la partición: fusionar o separar regiones, modificando la partición según el criterio de homogeneidad



# Algoritmos de segmentación directa (II)

57

- El algoritmo es inicializado seleccionando una primera estimación de la partición. La primera estimación
  - Puede contener una única región: toda la imagen es una región
  - Puede contener muchas regiones: ej. asociar cada pixel a una región diferente.
  - No tiene por qué ser una partición: puede contener un conjunto de componentes que marquen la parte más importante de las regiones (marcadores), y puede haber un gran numero de píxeles aún no asignados a ninguna región.
- El algoritmo construye la partición final mediante un algoritmo de optimización: en cada paso se van modificando los contornos de las regiones hasta cumplir una condición de terminación.
- Estudiaremos los siguientes algoritmos de decisión iterativa
  - Split and merge
  - Fusión de regiones
  - Watershed



## □ Split and merge

- Partición inicial: 1 región (toda la imagen)
- Criterio de homogeneidad: usualmente la varianza de la región

## □ Algoritmo básico:

1. Separar (split) en cuatro cuadrantes cada región  $R$  para la cual  $C(R_i)$  es falso



2. Fusionar (merge) todo par de regiones  $R_i$  y  $R_j$  para los que  $C(R_i \cup R_j)$  es verdadero

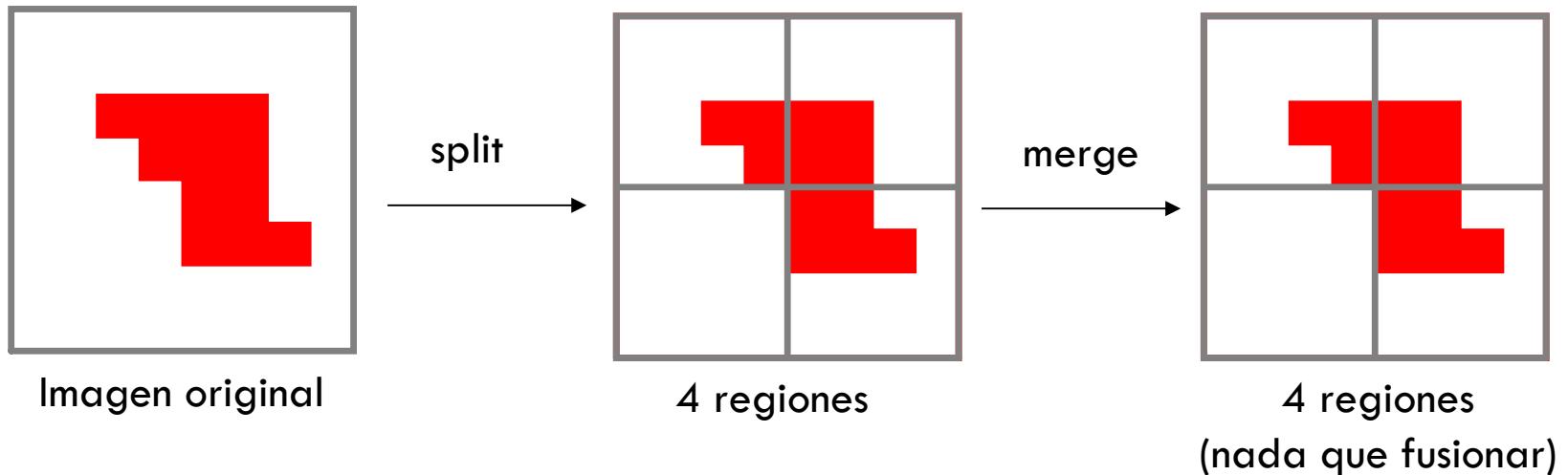


3. Repetir los pasos 1 y 2 hasta que no sea posible realizar más separaciones ni fusiones

# Ejemplo

59

## Iteración 1



**Paso Split** → separar en 4 cada región no uniforme

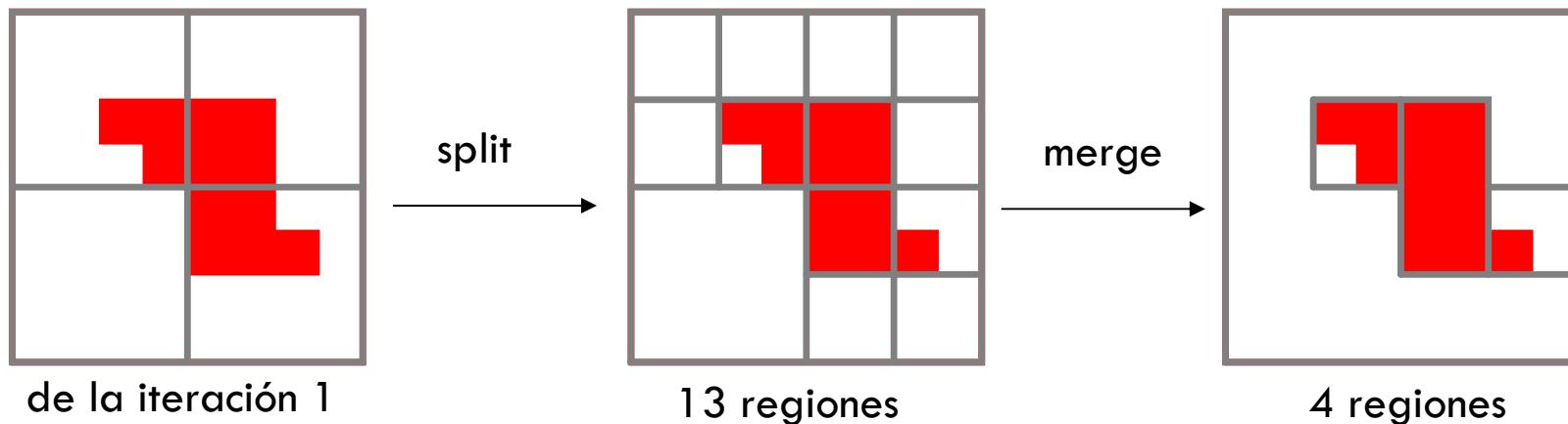
**Paso Merge** → fusionar todas las regiones adyacentes uniformes



# Ejemplo (II)

60

## Iteración 2



**Paso Split** → separar en 4 cada región no uniforme

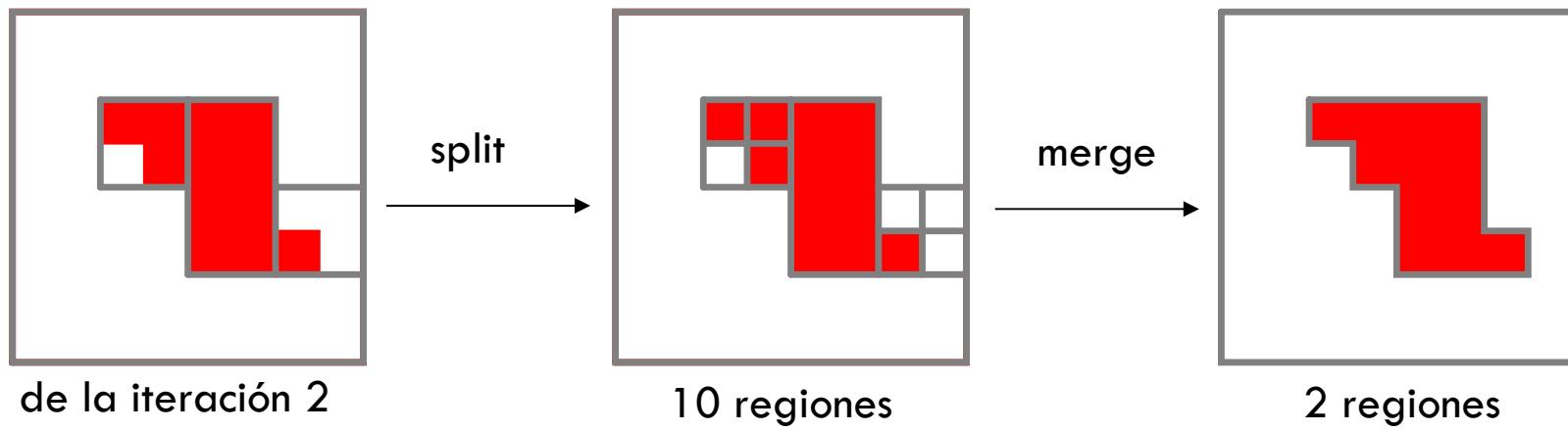
**Paso Merge** → fusionar todas las regiones adyacentes uniformes



# Ejemplo (III)

61

## Iteración 3



**Paso Split** → separar en 4 cada región no uniforme

**Paso Merge** → fusionar todas las regiones adyacentes uniformes

**Resultado final de la  
segmentación**



# Split & merge: ventajas e inconvenientes

62



Imagen original

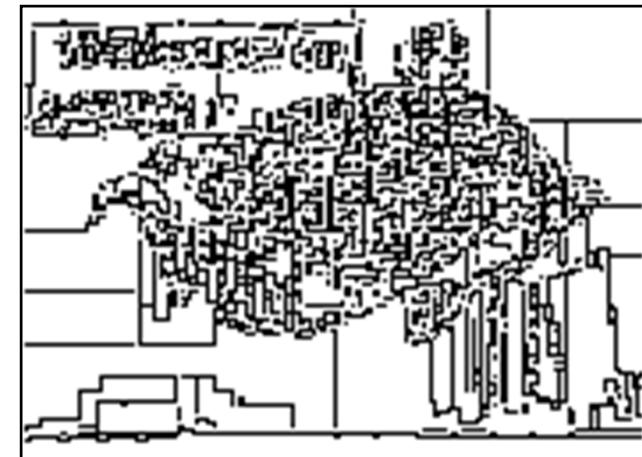
**Decisión por segmentación directa:  
algoritmo split and merge**

**Ventajas:**

- simple
- sin partición inicial específica
- visión global de la imagen

**Inconvenientes:**

- división puramente geométrica
- resultado con contornos poco naturales



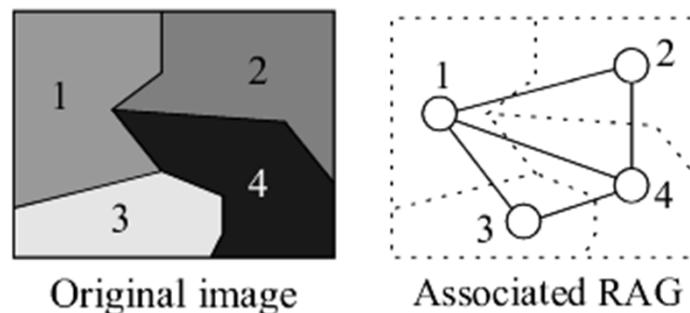
**Resultado final, representando  
cada región con su valor medio**

## □ Fusión de regiones:

- ▣ Partición inicial: cada pixel es una región (o sobresegmentación)
- ▣ Criterio de homogeneidad: medida de similitud entre cada par de regiones vecinas (usualmente basada en color / contornos)

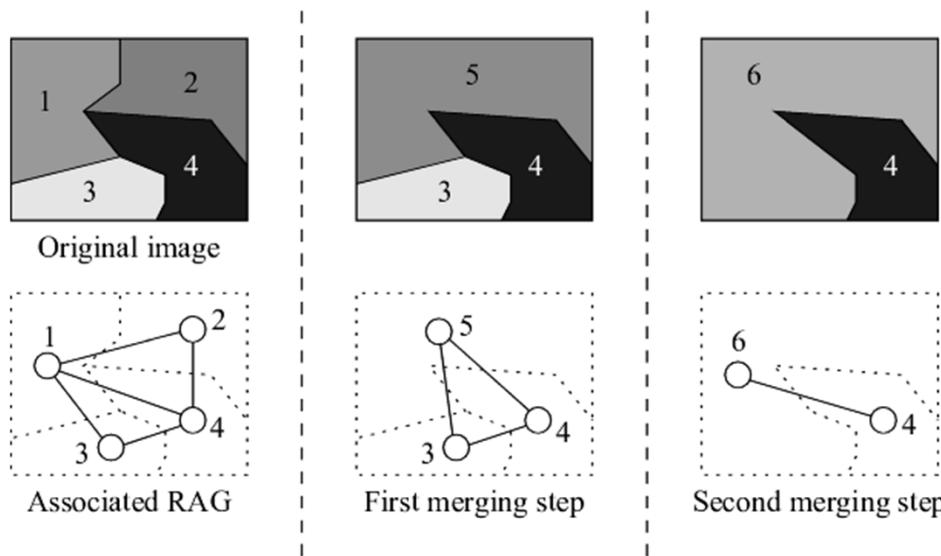
## □ Algoritmo de fusión de regiones:

- ▣ Trabaja sobre el RAG o **grafo de regiones adyacentes**
  - Nodos representan regiones
  - Arcos unen pares de regiones vecinas (tienen asociada una medida de similitud entre las regiones)



## □ Algoritmo iterativo de fusión de regiones:

1. Definir una partición inicial y su RAG (nodos: regiones, arcos: medida de similitud entre regiones)
2. Encontrar el par de regiones vecinas más similares
3. Fusionarlas
4. Actualizar los valores de los arcos
5. Repetir los pasos 2 a 4 hasta cumplir un criterio de terminación



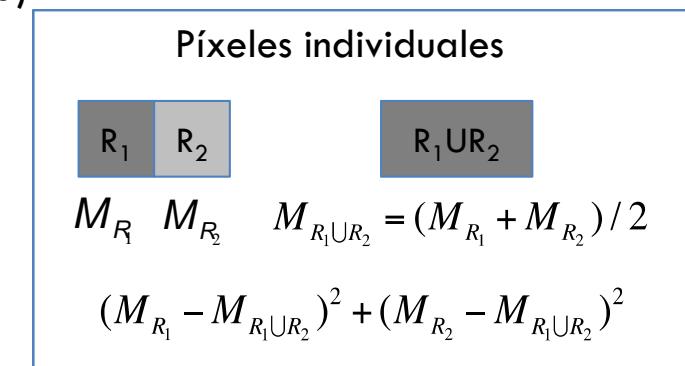
## Partición inicial

1. Cada pixel es una región individual o
2. Sobre-segmentación

## Medidas de similitud:

1. **Basada en color:** diferencia de color entre los modelos de las regiones, teniendo en cuenta su tamaño (aprox. del error cuadrático medio)

$$C_{color}(R_1, R_2) = N_{R1} \|M_{R1} - M_{R1 \cup R2}\|_2^2 + N_{R2} \|M_{R2} - M_{R1 \cup R2}\|_2^2$$



donde  $M_R$ , el modelo de cada región  $R$ , es el promedio de los valores de los píxeles en la región, y  $N_{R_i}$  es el número de píxeles de la región  $R_i$

# Fusión de regiones: medidas de similitud

66

## Medidas de similitud:

2. **Basada en color y contornos:** el término de contornos mide cómo se incrementa el perímetro al fusionar dos regiones

$$C(R_1, R_2) = \alpha C_{color}(R_1, R_2) + (1 - \alpha) C_{cont}(R_1, R_2)$$
$$C_{cont}(R_1, R_2) \approx \text{longitud del contorno común}$$

## Criterios de terminación:

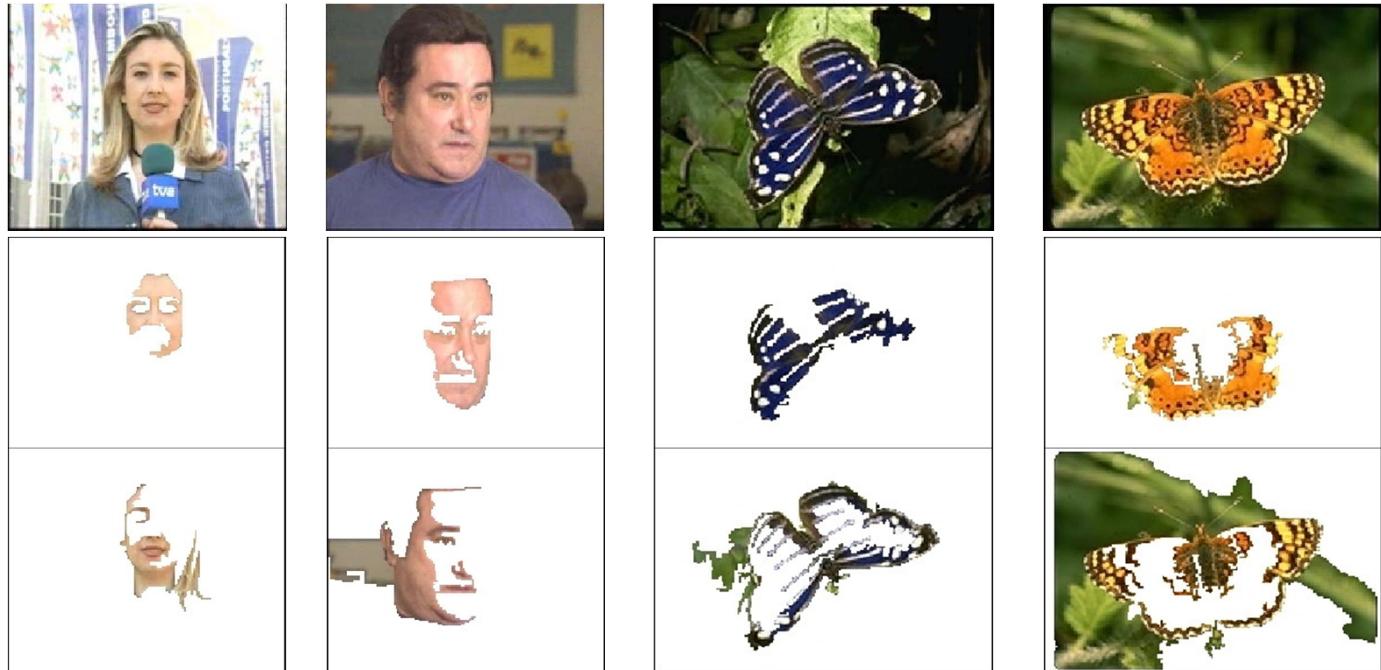
1. **Número final de regiones**
2. **PSNR** de la imagen que resulta de representar cada región con su modelo



# Fusión de regiones: criterio de fusión

67

**Criterio 1  
(sólo color)**



**Los objetos aparecen separados en dos regiones porque una de las partes es más similar en color a una región del fondo que a la otra parte del mismo objeto**

**Criterio 2  
(color y contornos)**



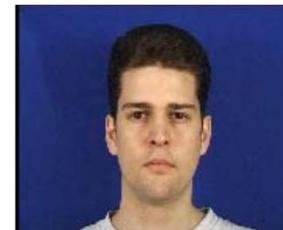
**Los objetos aparecen como una única región (se tiende a minimizar los contornos de las regiones, para regiones de color similar)**



# Fusión de regiones: criterio de terminación

68

Número de  
regiones  
(50)



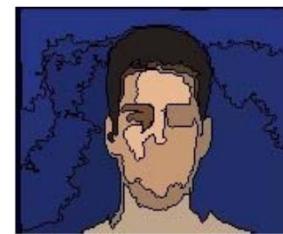
22,00 dB [50 regiones] 29,50 dB [50 regiones]

22,13 dB [50 regiones] 32,29 dB [50 regiones]

PSNR  
(26dB)



[26 dB] 273 regiones



[26 dB] 13 regiones



[26 dB] 347 regiones



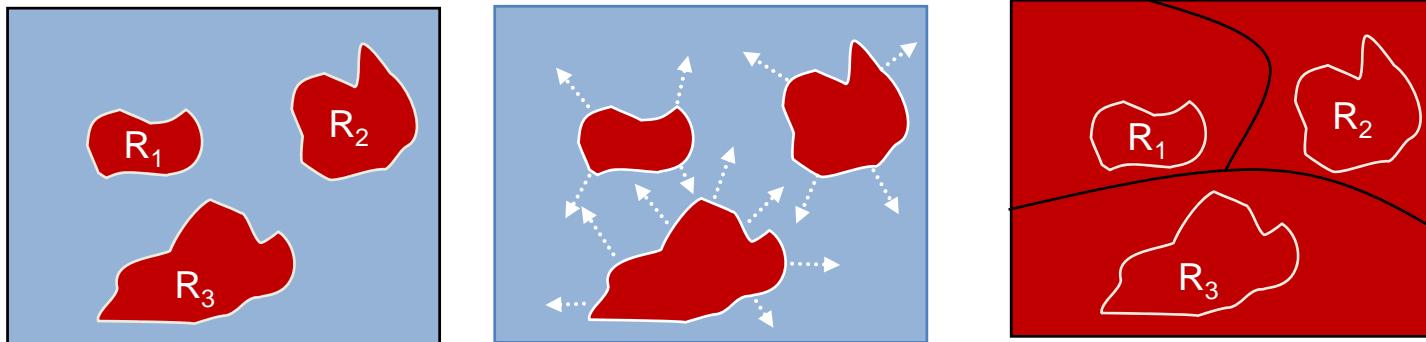
[26 dB] 7 regiones

## □ Crecimiento de regiones

- Partición inicial: segmentación parcial, definición de una zona de incertidumbre que se elimina progresivamente

$$\bigcup_i R_i \neq I$$
$$I - \bigcup_i R_i \quad \text{zona de incertidumbre}$$

- Criterio de homogeneidad: usualmente la varianza de la región

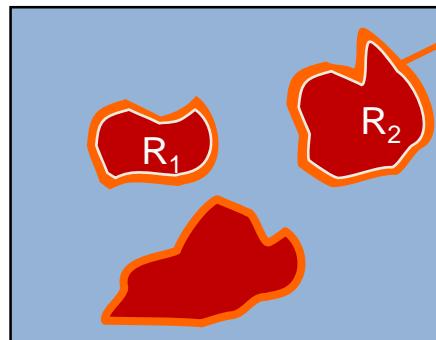


### Regiones iniciales o “marcadores”:

- pertenecerán al interior de las regiones finales
- son zonas “seguras”
- dependen de la aplicación

# Eliminación de la zona de incertidumbre

70



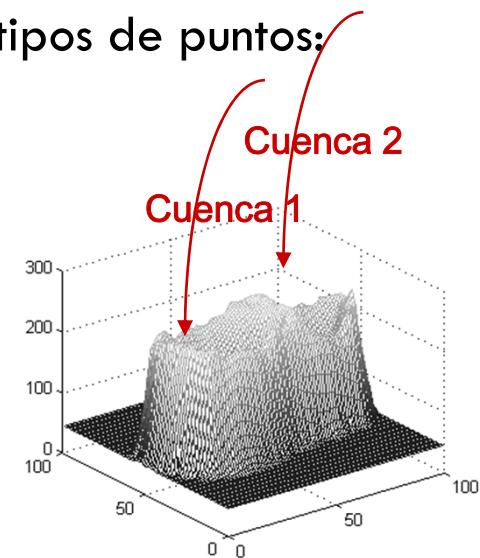
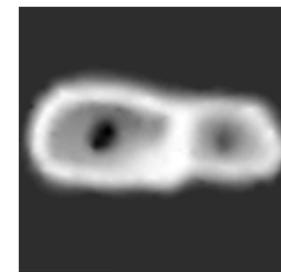
Píxeles de la zona de incertidumbre que están conectados a las regiones

Se analiza cada pixel del contorno exterior de cada una de las regiones seguras, y se agrega a alguna de las regiones existentes.

- El pixel se agrega a la región para la cual se minimice el criterio de homogeneidad.
  - **Qué criterio elegir?**
    - Por ejemplo, agregar el pixel a la región con nivel de gris o color más parecido.
    - Por ejemplo, agregar el pixel a la región para la cual el aumento de longitud del contorno al agregar el pixel es menor.
- ⇒ **Técnica de fusión entre las regiones y los píxeles de la zona de incertidumbre**

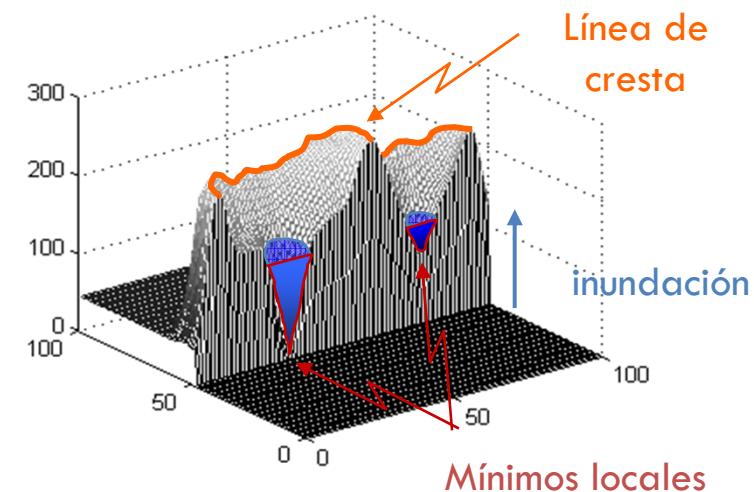
## Caso particular de crecimiento de regiones: Watershed

- Watershed es una técnica morfológica de crecimiento de regiones
- Vamos a estudiarla para imágenes en niveles de gris. Existe una extensión a imágenes en color
- El concepto de watershed se basa en visualizar una imagen **como un relieve topográfico (3D)**: dos coordenadas espaciales vs. niveles de gris (la altura).
- En esta interpretación 'topográfica', consideramos tres tipos de puntos:
  - puntos que corresponden a mínimos locales
  - puntos en los que, si se coloca una gota de agua, esta cae con certeza en un único mínimo.
  - puntos en los que el agua caería con igual probabilidad en más de uno de estos mínimos.



Consideramos tres tipos de puntos:

1. puntos que corresponden a mínimos locales
2. puntos en los que, si se coloca una gota de agua, esta cae con certeza en un único mínimo.
3. puntos en los que el agua caería con igual probabilidad en más de uno de estos mínimos.



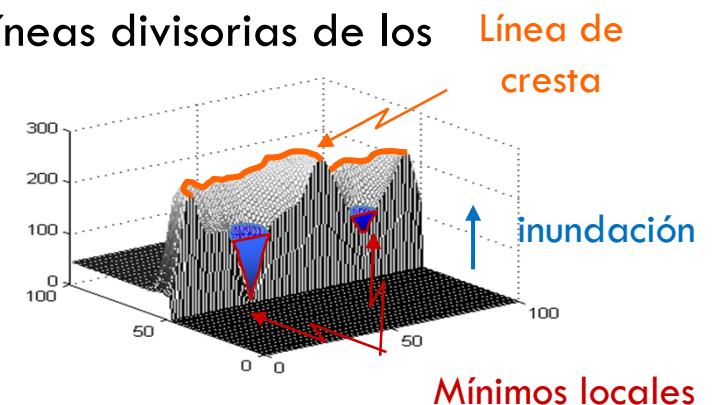
- Para un mínimo local particular, el conjunto de puntos que satisface (2) se llama **catchment basin** o **watershed** o **cuenca** de este mínimo.
- Los puntos que satisfacen la condición (3) forman líneas de cresta en la superficie topográfica y son llamados **líneas divisorias** o **líneas de watershed** o **líneas de cresta**.
- El principal objetivo de los algoritmos de segmentación basados en estos conceptos es hallar las líneas de cresta.

- **Objetivo:** encontrar las líneas de cresta (lugar donde 2 (o más) cuencas entran en contacto)

- **Idea básica:**

- Supongamos que se hace un pequeño agujero en cada mínimo local, y que todo el relieve topográfico es inundado desde abajo, dejando que el agua entre a velocidad constante.
- El agua va subiendo e inundando las cuencas. Cuando el agua de dos cuencas está a punto de juntarse, se construye un **dique (dam)** (de 1 pixel de ancho) para evitar la fusión.
- La inundación continúa, y llega a un punto en que solo se ve la parte de arriba de los diques por arriba de la línea de agua.
- Los bordes de los diques corresponden a las líneas divisorias de los watersheds.

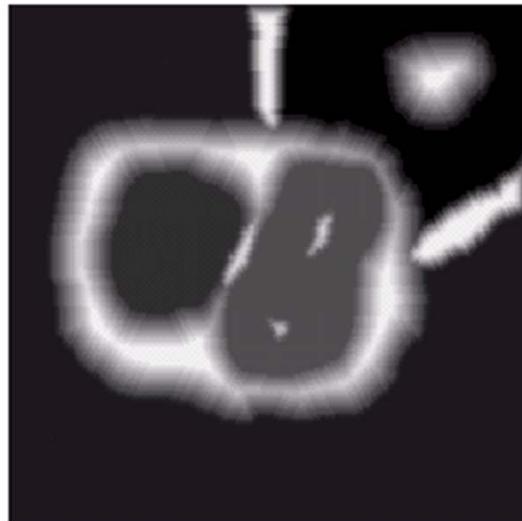
- **Las líneas de watershed forman un camino conexo, dando por lo tanto bordes continuos entre las regiones.**



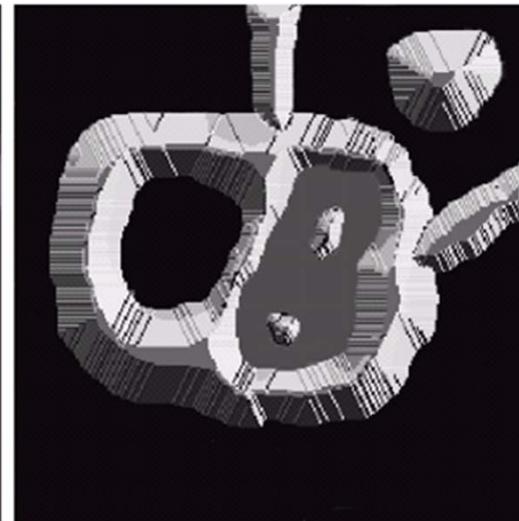
# Watershed. Ejemplo

74

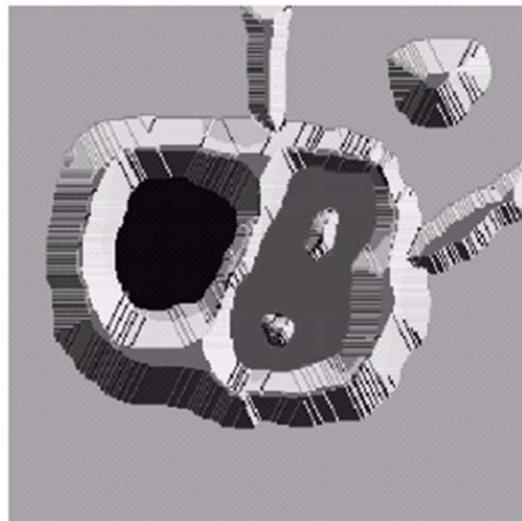
Imagen  
original



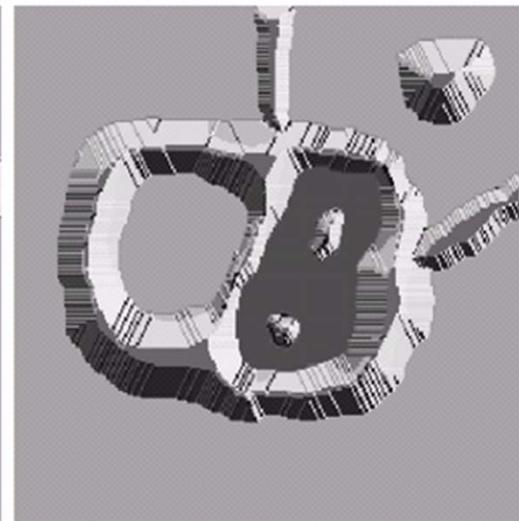
Vista  
topográfica



Inundación  
parcial  
(fondo)



Inundación  
parcial (fondo  
y una cuenca)



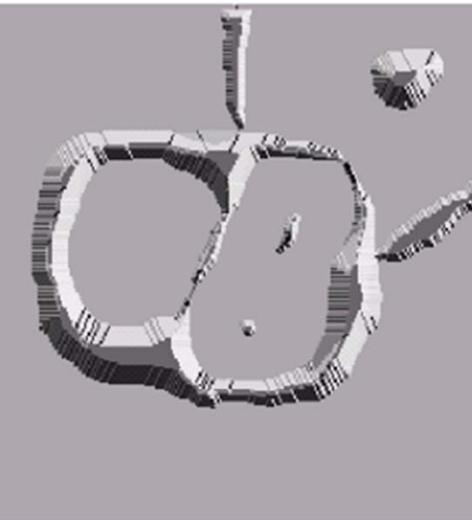
# Watershed. Ejemplo (II)

75

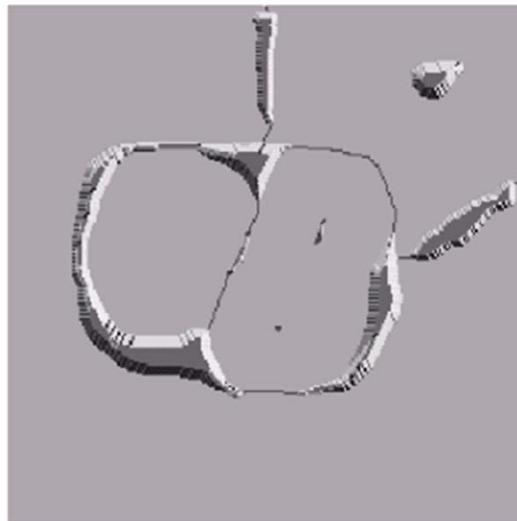
Inundación  
parcial



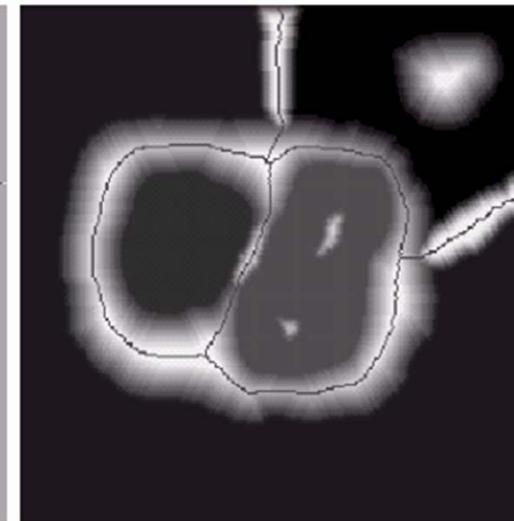
Comienzan a  
fusionarse dos  
cuencas



Diques más  
largos



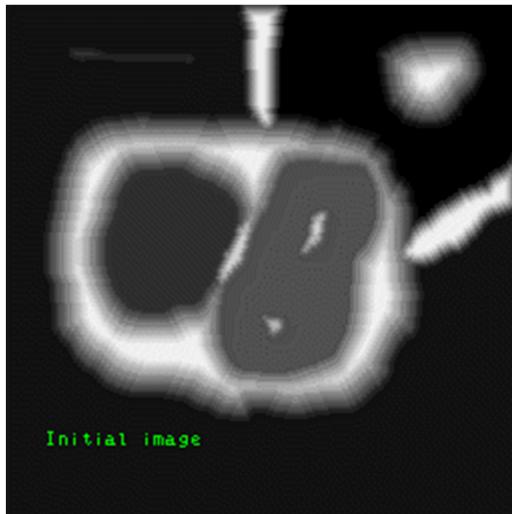
Líneas finales  
de watershed



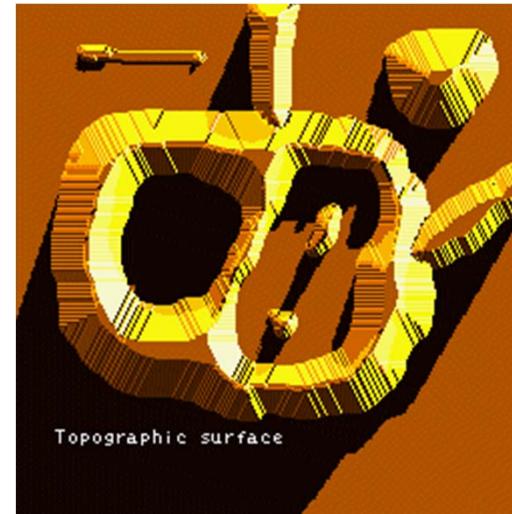
# Watershed. Ejemplo (III)

76

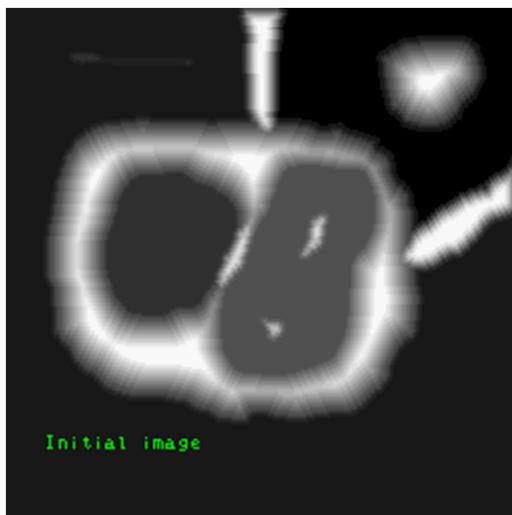
Imagen  
original



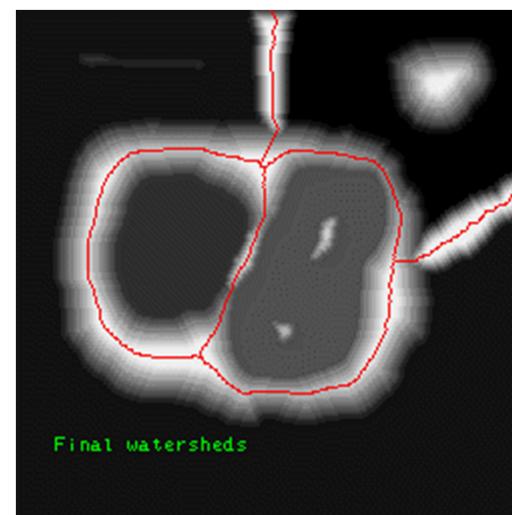
Vista  
topográfica



Inundación



Líneas finales  
de watershed



<http://cmm.ensmp.fr/~beucher/wtshed.html>

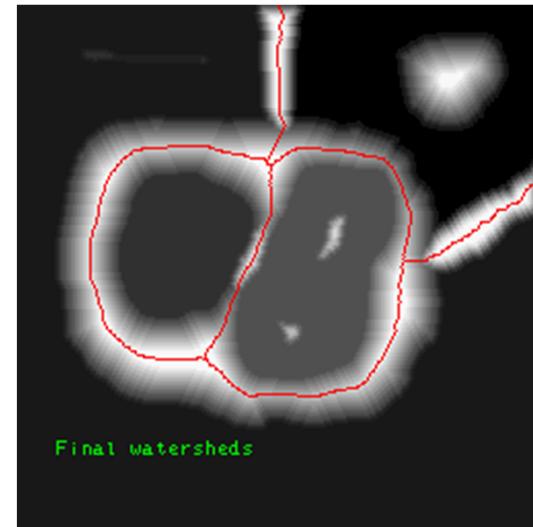
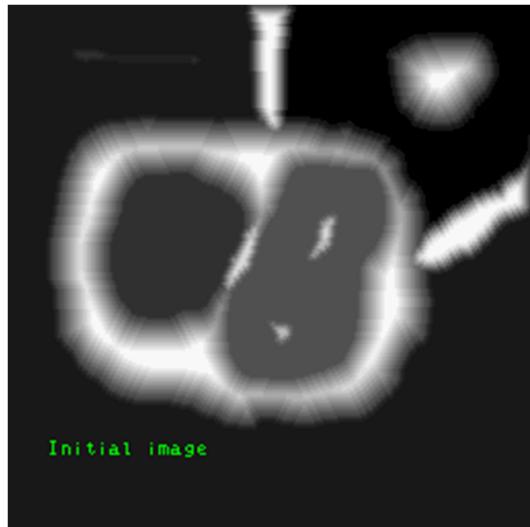


# Algoritmo de watershed

77

**Partición inicial:** las regiones iniciales son los mínimos locales de la imagen

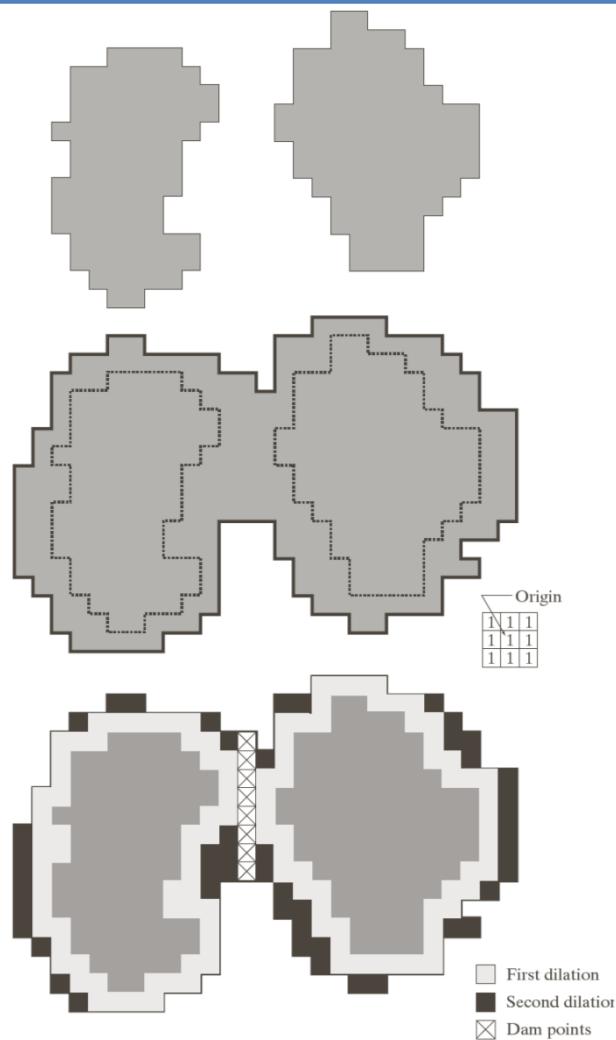
**Criterio de crecimiento:** el pixel con nivel de gris más próximo al nivel de la inundación es asignado a su cuenca



Los diques finales son las líneas de cresta, los contornos de las regiones de la partición final

# Algoritmo de watershed

78



**FIGURE 10.55** (a) Two partially flooded catchment basins at stage  $n - 1$  of flooding.  
(b) Flooding at stage  $n$ , showing that water has spilled between basins. (c) Structuring element used for dilation. (d) Result of dilation and dam construction.



# Segmentación con watershed

79

En una imagen en niveles de gris, los contornos de los objetos son las líneas de cresta del gradiente:

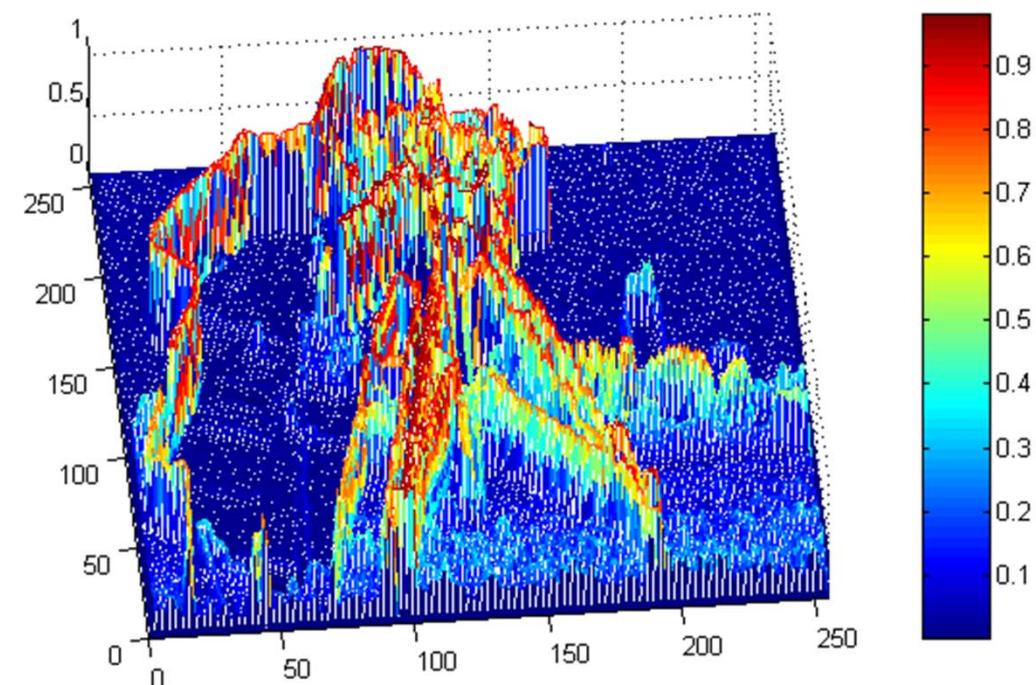
⇒ Para segmentar una imagen, el algoritmo de watershed se **aplica al gradiente de la imagen, no a la imagen original**



Imagen original



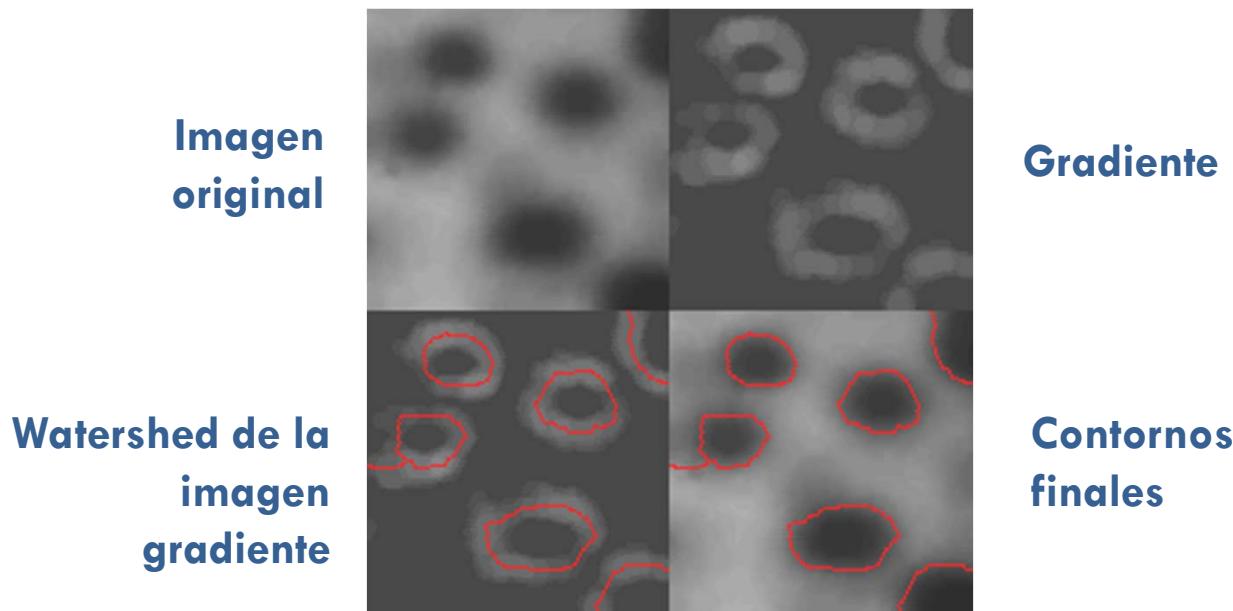
Gradiente



# Segmentación con watershed (II)

80

Si aplicamos esta transformación a la imagen gradiente, las cuencas corresponderán, en teoría, a las regiones homogéneas en nivel de gris de la imagen



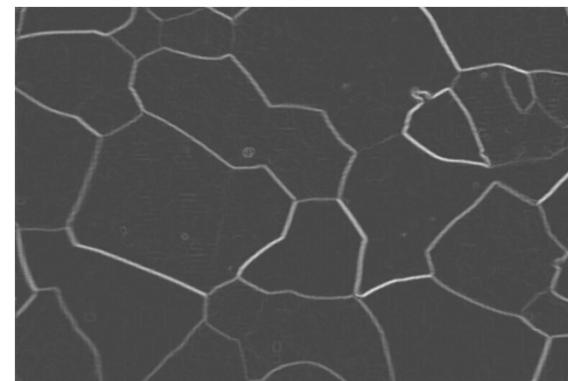
# Sobresegmentación

81

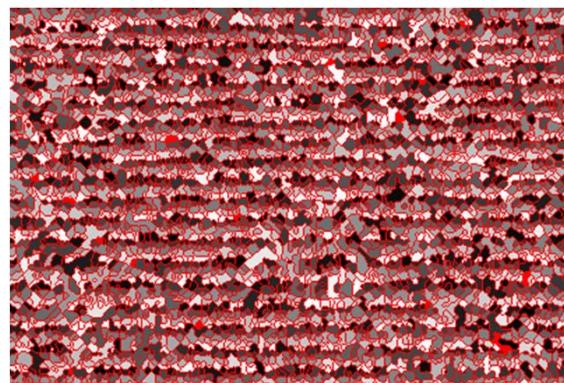
**Problema:** La aplicación directa del algoritmo lleva a la **sobresegmentación** (demasiadas regiones) debido al ruido y a la presencia de un gran número de mínimos locales



Original



Gradiente



⇒ sobre-segmentación

Imagen segmentada con watershed



## Tres posibles soluciones a la sobresegmentación

- **Simplificar el gradiente, por ejemplo binarizándolo.**
  - El problema es la selección del umbral: un umbral demasiado bajo producirá menos regiones, un umbral alto demasiadas regiones
- **Simplificar la imagen y luego calcular el gradiente.**
- **Utilizar marcadores:**
  - Marcadores internos, asociados con objetos de interés
  - Marcadores externos, asociados con el fondo.

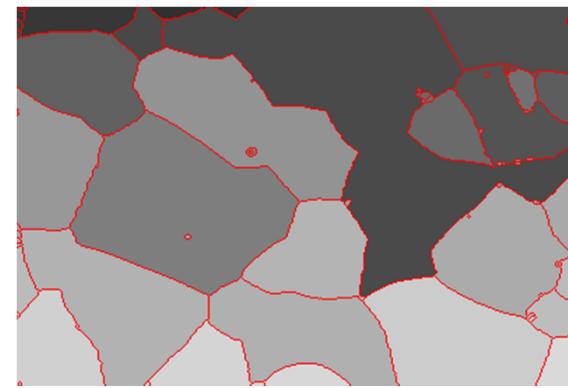
# Solución 1: Simplificación del gradiente

83

Eliminación de los mínimos superfluos por binarización



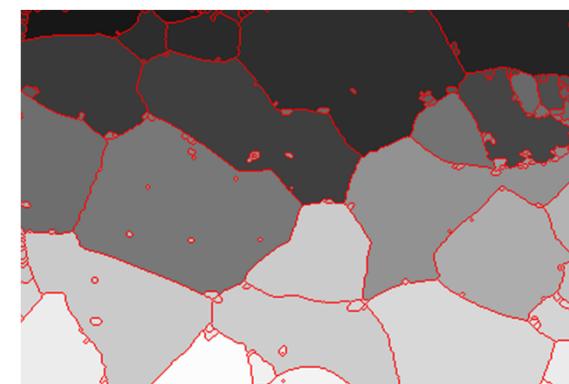
Original



Watershed con gradiente binarizado (1)



sobre-segmentación



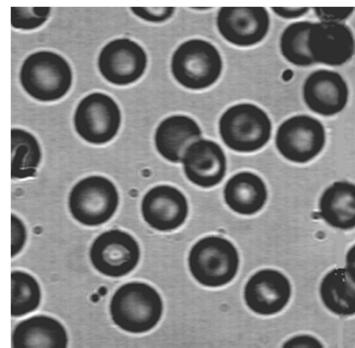
Watershed con gradiente binarizado (2)



# Solución 2: Simplificación de la imagen

84

Eliminación de los mínimos superfluos por simplificación de la imagen antes de calcular el gradiente



Original

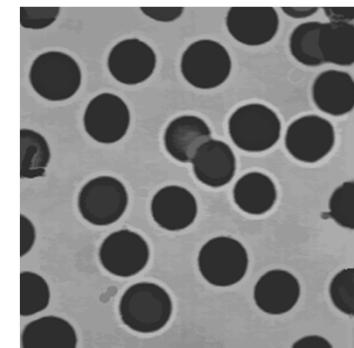
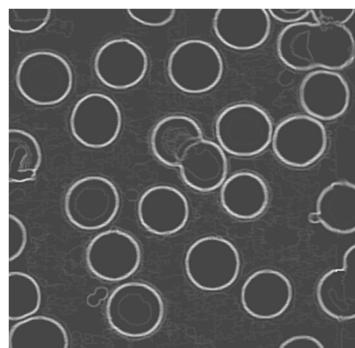
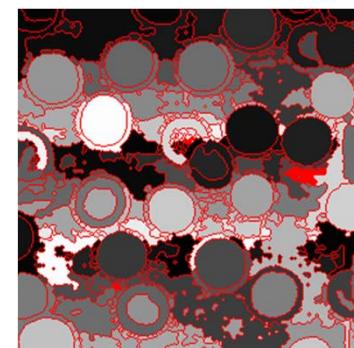


Imagen simplificada  
reconstrucción (erosión 25x25)  
reconstrucción dual (dilatación 10x10)



Gradiente



Watershed



# Solución 3: Watershed con marcadores

85

Se consiguen muy buenos resultados inundando la superficie topográfica a partir de un conjunto de marcadores previamente definidos: sólo se inunda a partir de los mínimos indicados por los marcadores



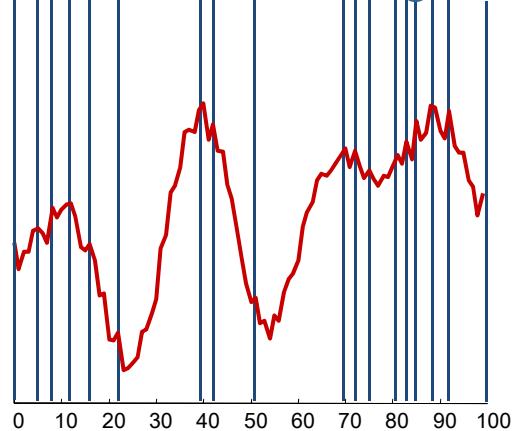
De esta manera se consigue evitar el problema de la sobresegmentación

# Solución 3: Watershed con marcadores (II)

86

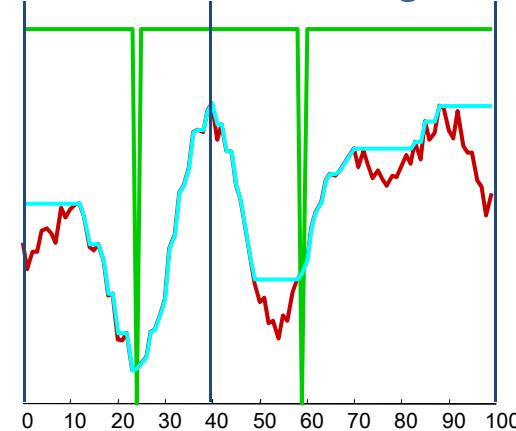
Simplificación del gradiente con marcadores :  
se limitan los mínimos mediante una reconstrucción dual de la señal  
(gradiente) a partir de los marcadores elegidos

Contornos entre las regiones



Watershed sin marcador

Contornos entre las regiones



Marcadores para los mínimos y  
reconstrucción dual



# Uso de marcadores. Ejemplo

87

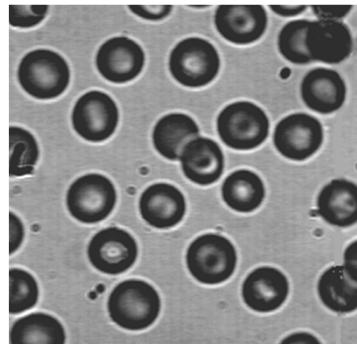
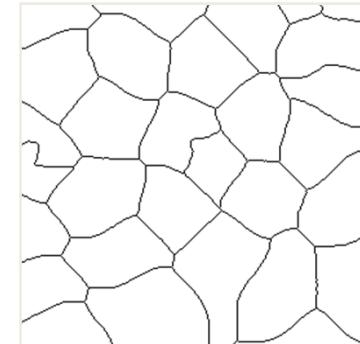


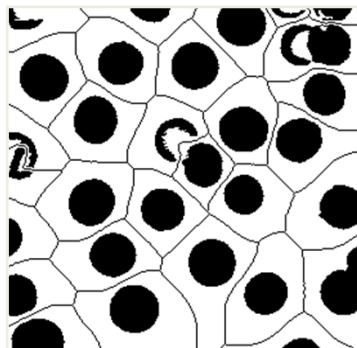
Imagen original



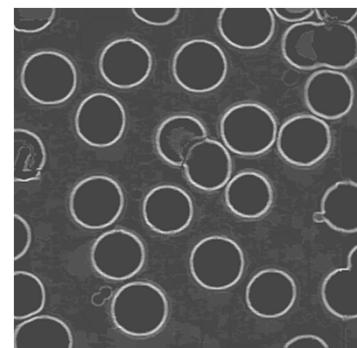
Marcador de los objetos:  
Binarización de la imagen  
original



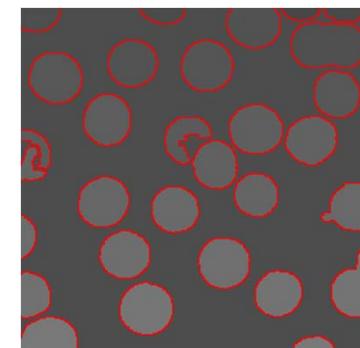
Marcador del fondo:  
watershed de los marcadores  
de los objetos



Marcadores (objetos y fondo)

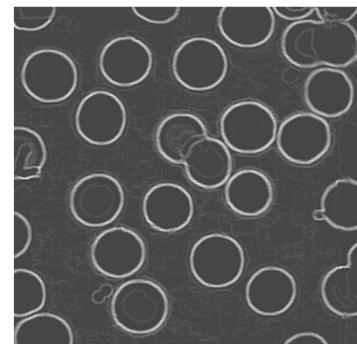


Gradiente simplificado:  
limitación de los mínimos  
por reconstrucción (reconstrucción dual del  
gradiente a partir de los marcadores)

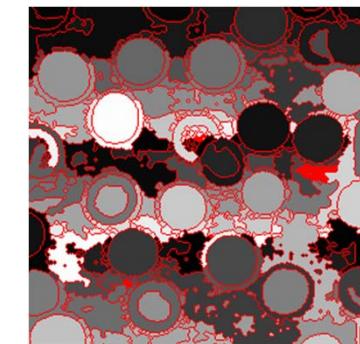


Segmentación  
final

## Simplificación de la imagen

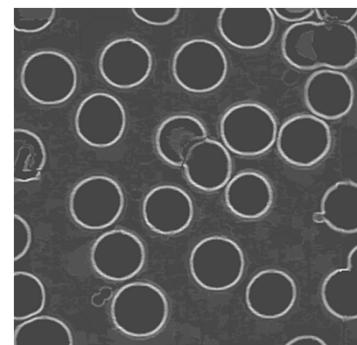


Gradiente

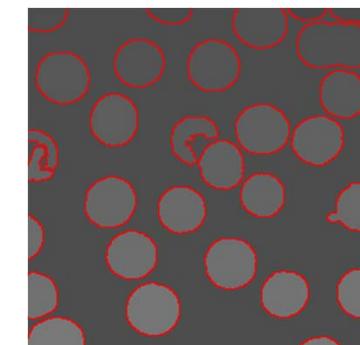


Watershed

## Uso de marcadores



Gradiente simplificado:  
limitación de los mínimos  
por reconstrucción (reconstrucción dual del  
gradiente a partir de los marcadores)



Segmentación  
final

# Uso de marcadores. Ejemplo

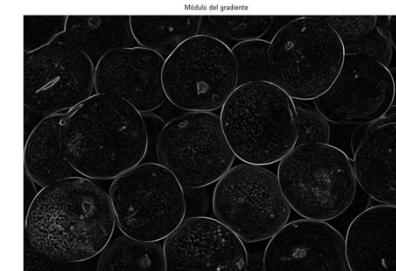
89



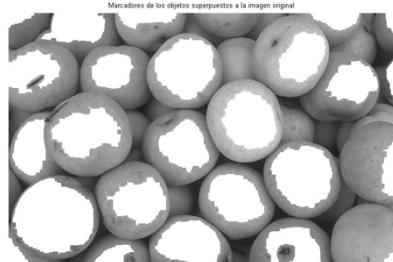
Imagen original



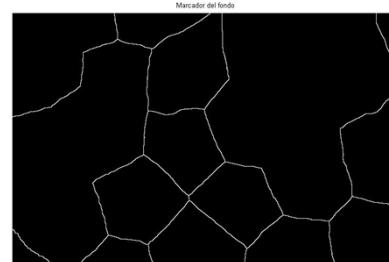
Imagen niveles de gris



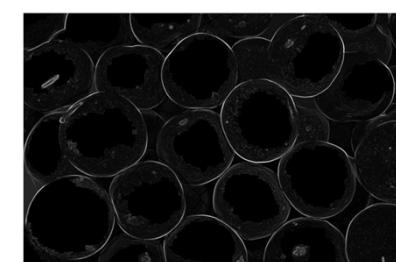
Gradiente



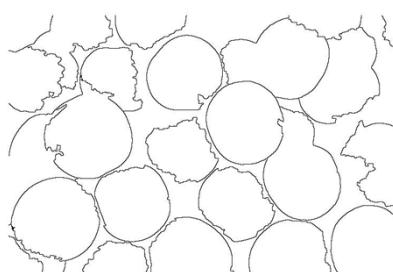
Marcadores de objetos



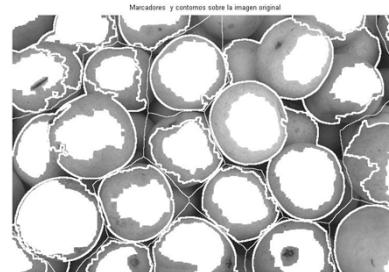
Marcadores del fondo



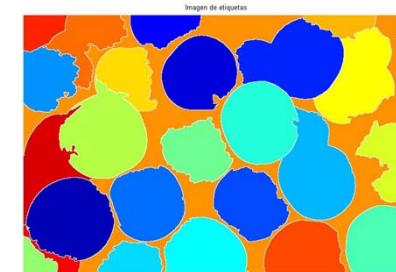
Gradiente con mínimos simplificados



Watershed: contornos finales



Contornos + marcadores + imagen



Partición final



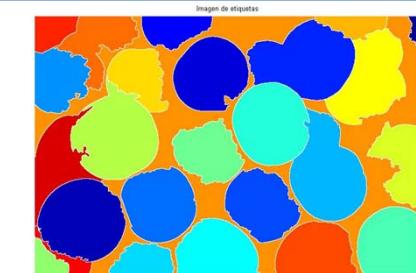
# Comparación con otras técnicas

90

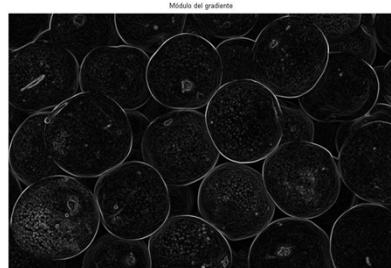


Original

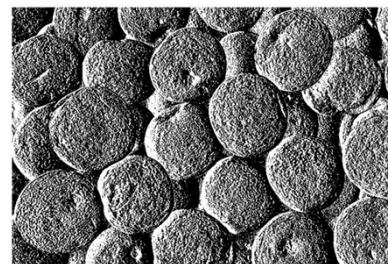
Watershed



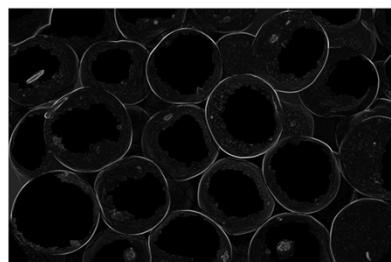
## x Detección de transiciones



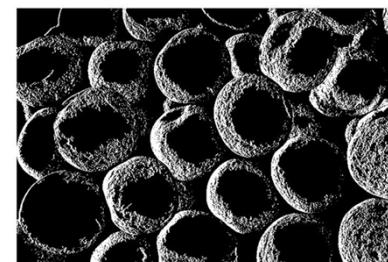
Gradiente



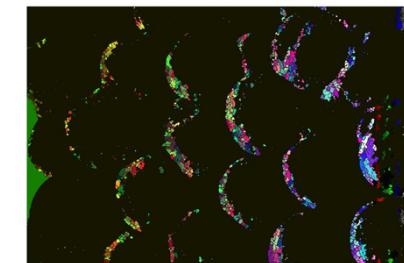
Gradiente  
binarizado



Gradiente  
simplificado



Gradiente  
binarizado



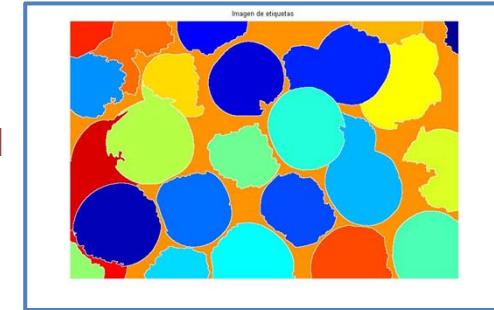
Partición final

# Comparación con otras técnicas

91



Original



Watershed

## x Segmentación del espacio de características

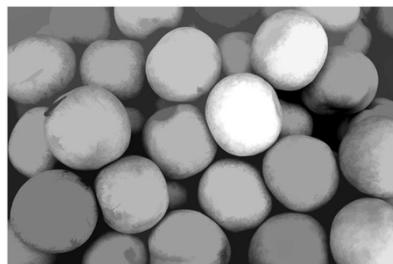
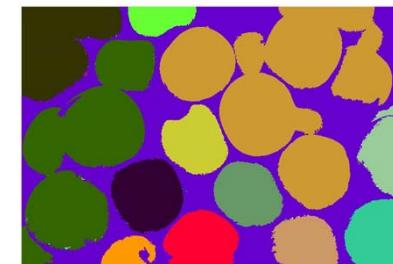


Imagen simplificada



Binarización (2clases)



Partición final

## x Algoritmo de fusión de regiones



Partición final: 30 regiones



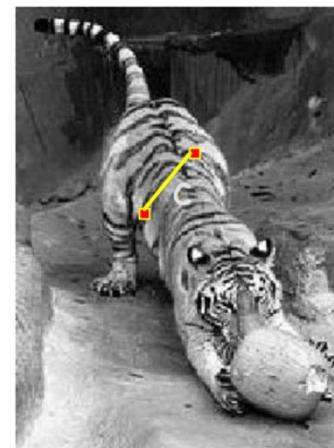
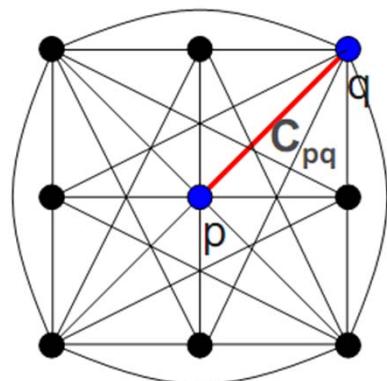
Partición final: 50 regiones



# Segmentación con Graph Cut

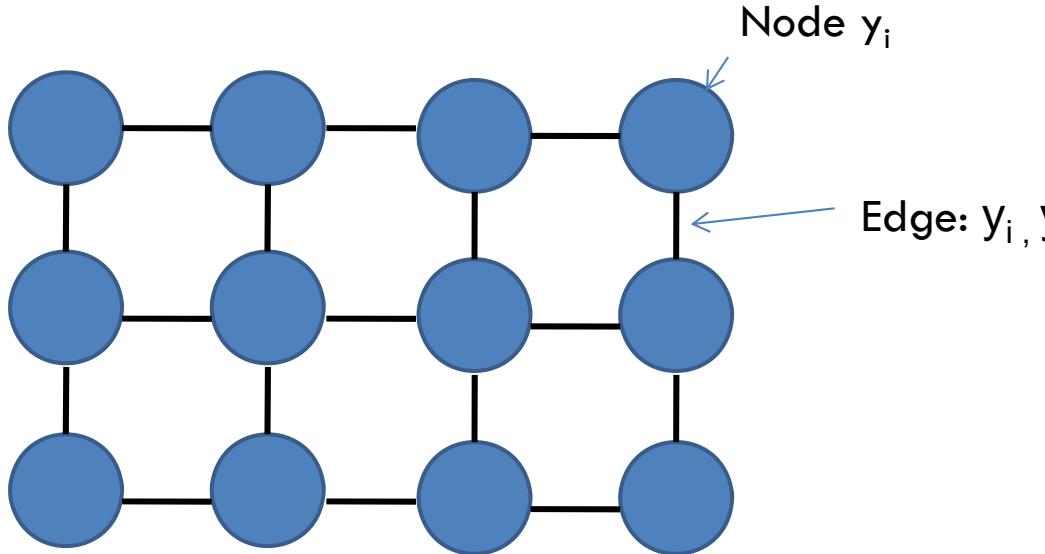
92

- Enfoque de optimización global basado en funciones de energía
  - Construye una función de energía heurística que tiene en cuenta tanto la información local como global
- Representación grafo
  - $G=(V, E, w)$ , V nodos (pixels), E arcos (unen cada par de nodos), w peso ( $w_{ij}$ , función de similitud entre nodos i y j)



# Funciones de energía para Graph cut

93



$$Energy(\mathbf{y}; data) = \sum_i \psi_1(y_i; data) + \sum_{i, j \in edges} \psi_2(y_i, y_j; data)$$

Coste de asignar una  
etiqueta a un pixel

Coste de asignar un par de  
etiquetas a dos píxeles conectados

# Ejemplo segmentación objeto – fondo

94

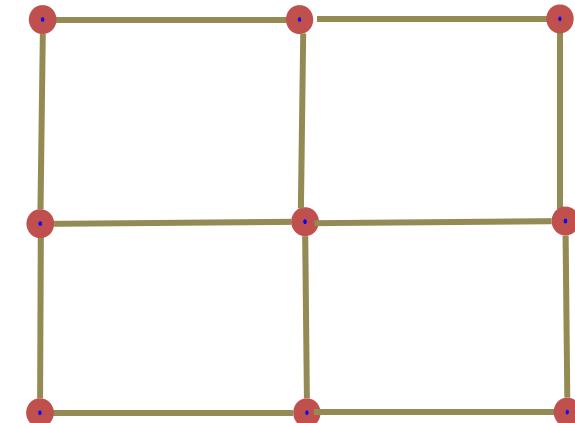
$$E(x) = \sum_i c_i x_i + \sum_{i,j} c_{ij} x_i (1-x_j)$$

$E: \{0,1\}^n \rightarrow \mathbb{R}$   
0 → fg  
1 → bg

n = number of pixels



Image (D)



$x_i$

$x_j$

# Ejemplo segmentación objeto – fondo

95

$$E(x) = \sum_i c_i x_i + \sum_{i,j} c_{ij} x_i (1-x_j)$$

$$\begin{aligned} E: \{0,1\}^n &\rightarrow \mathbb{R} \\ 0 &\rightarrow \text{fg} \\ 1 &\rightarrow \text{bg} \end{aligned}$$

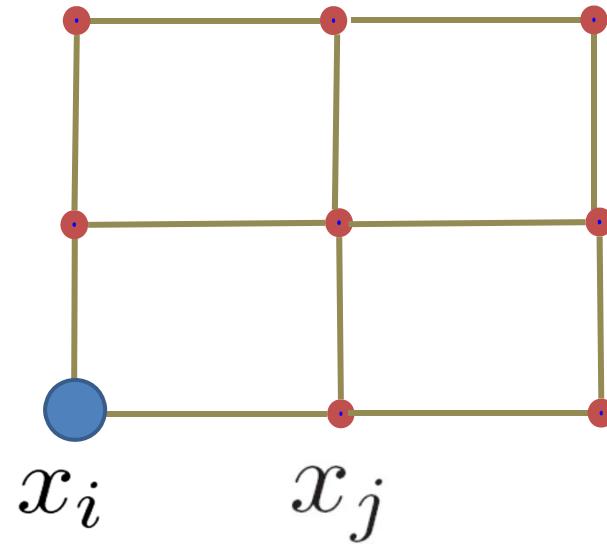
$n = \text{number of pixels}$



**Unary Cost ( $c_i$ )**

Dark (negative)

Bright (positive)



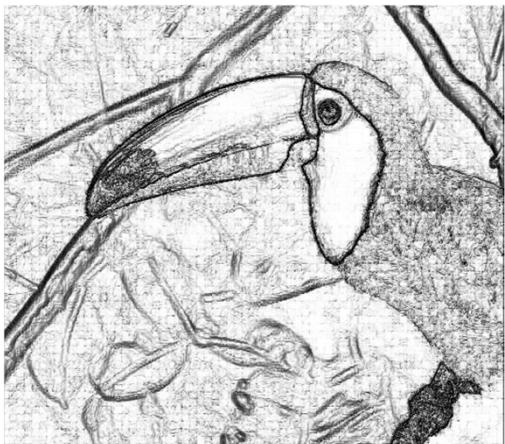
# Ejemplo segmentación objeto – fondo

96

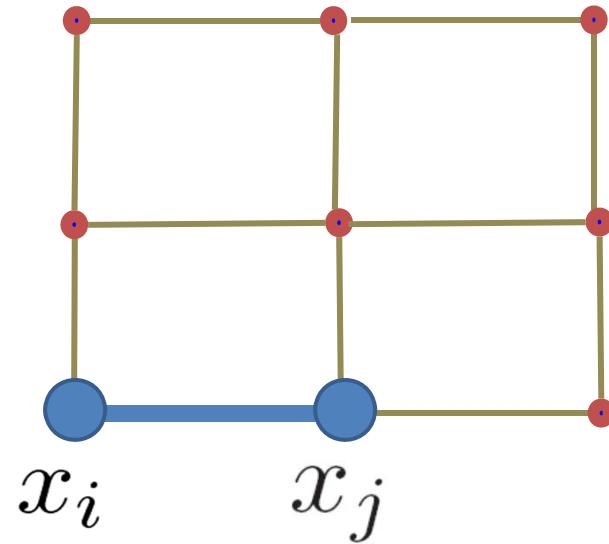
$$E(x) = \sum_i c_i x_i + \sum_{i,j} c_{ij} x_i (1-x_j)$$

$E: \{0,1\}^n \rightarrow \mathbb{R}$   
0 → fg  
1 → bg

$n = \text{number of pixels}$



**Discontinuity  
Cost ( $c_{ij}$ )**



# Ejemplo segmentación objeto – fondo

97

$$E(x) = \sum_i c_i x_i + \sum_{i,j} c_{ij} x_i (1-x_j)$$

$E: \{0,1\}^n \rightarrow \mathbb{R}$   
0 → fg  
1 → bg

n = number of pixels



Mínimo global ( $x^*$ )

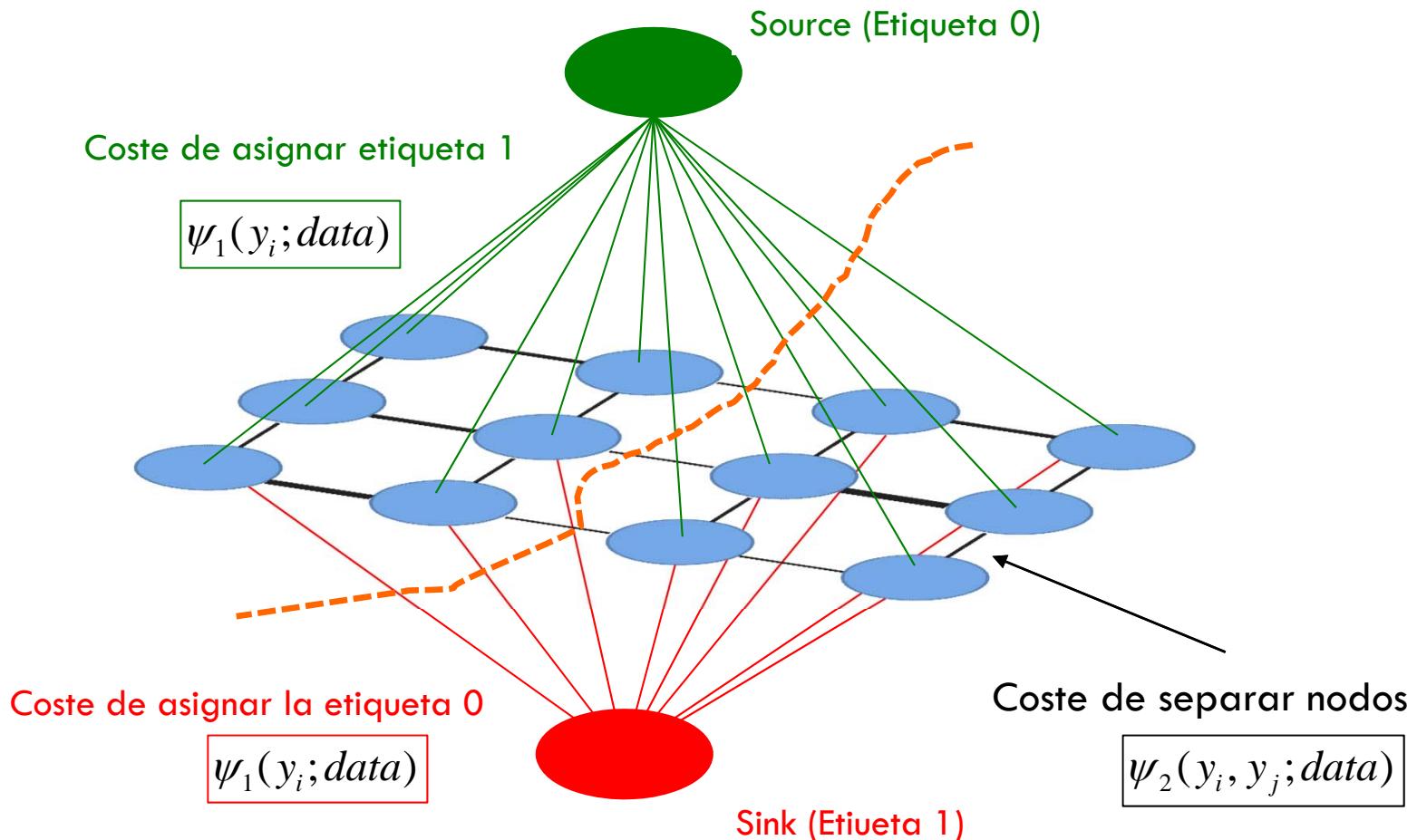
$$x^* = \arg \min_x E(x)$$

Cómo minimizar  $E(x)$ ?

# Solución con Graph Cut

98

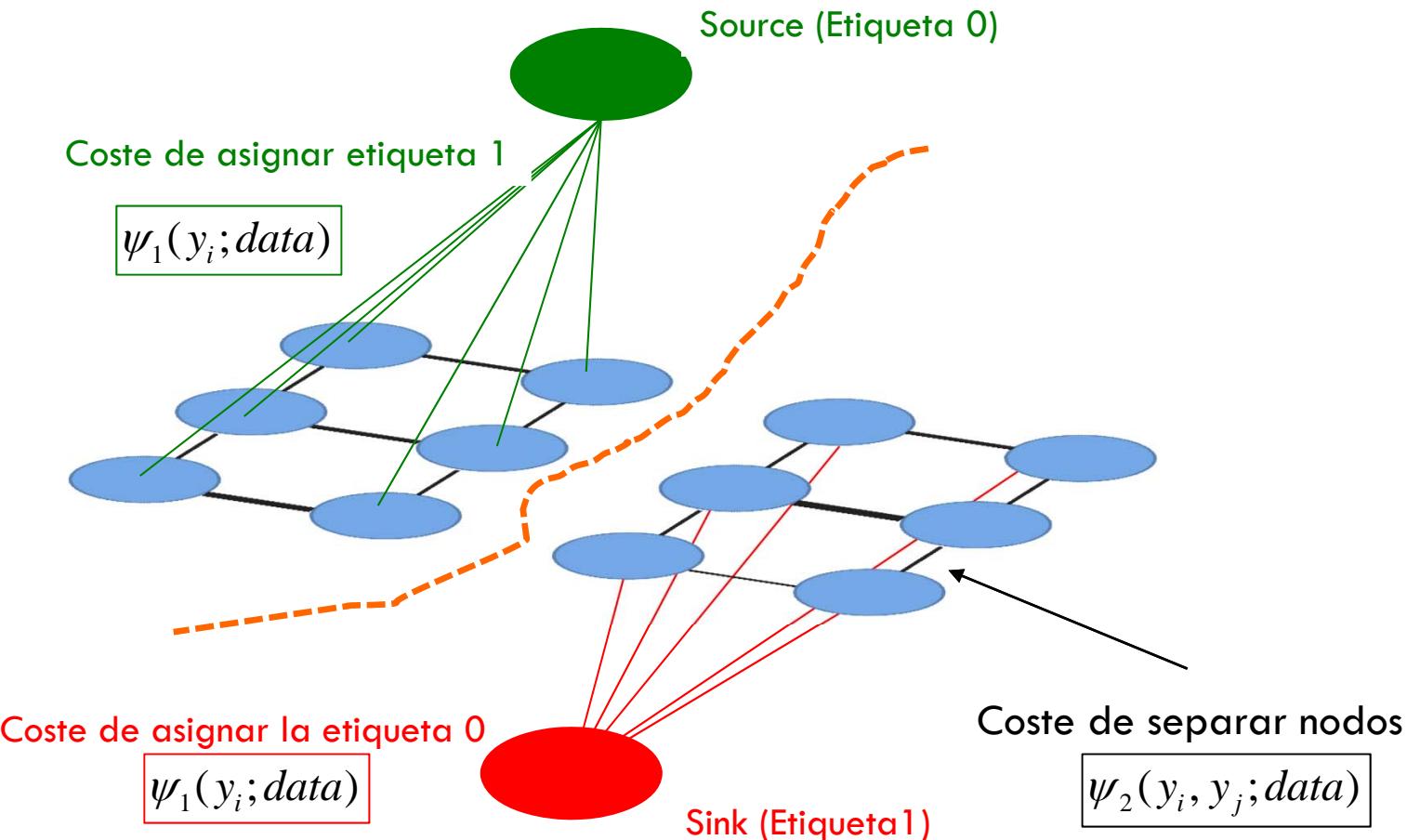
$$Energy(\mathbf{y}; data) = \sum_i \psi_1(y_i; data) + \sum_{i, j \in edges} \psi_2(y_i, y_j; data)$$



# Solución con Graph Cut

99

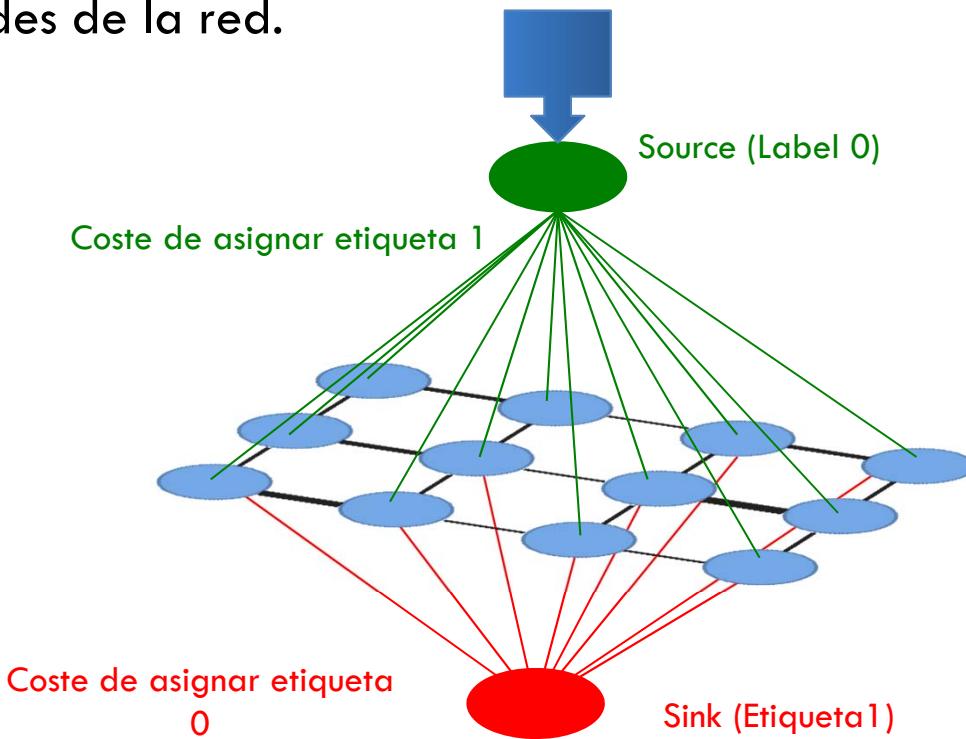
$$Energy(\mathbf{y}; data) = \sum_i \psi_1(y_i; data) + \sum_{i, j \in edges} \psi_2(y_i, y_j; data)$$



# Max-flow / Min-cut

100

- El grafo se interpreta como una red donde un flujo puede atravesar la red desde la fuente (source) hasta el sumidero (sink). Los pesos del grafo representan las capacidades de la red.

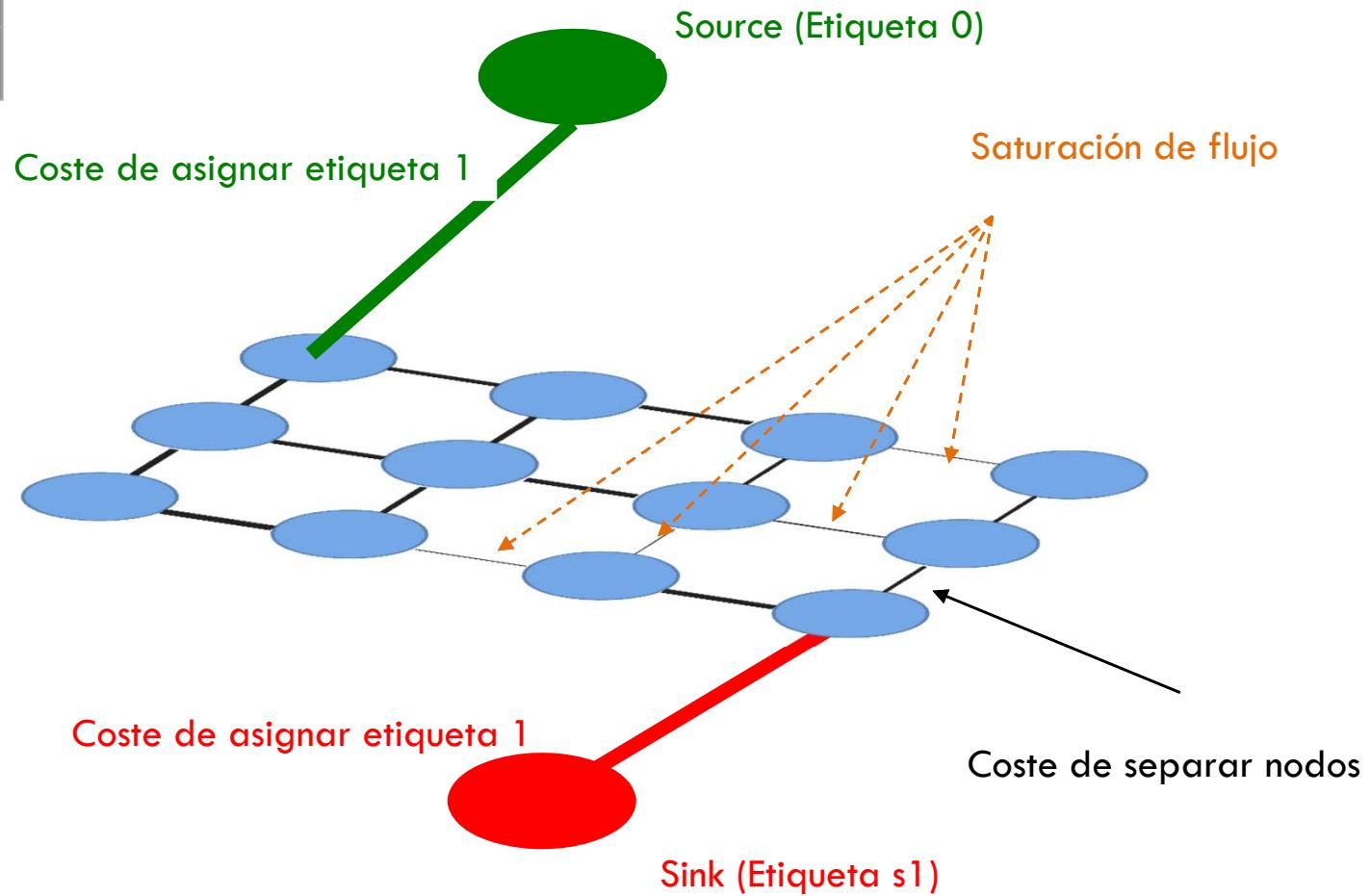
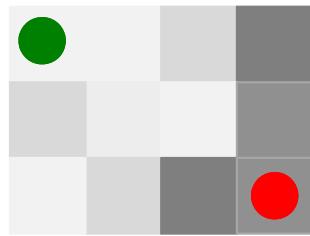


- Para cualquier red con un único nodo origen (source) y un único nodo destino (sink), el máximo flujo posible desde la fuente hasta el sumidero es igual al valor del corte mínimo entre todos los cortes posibles



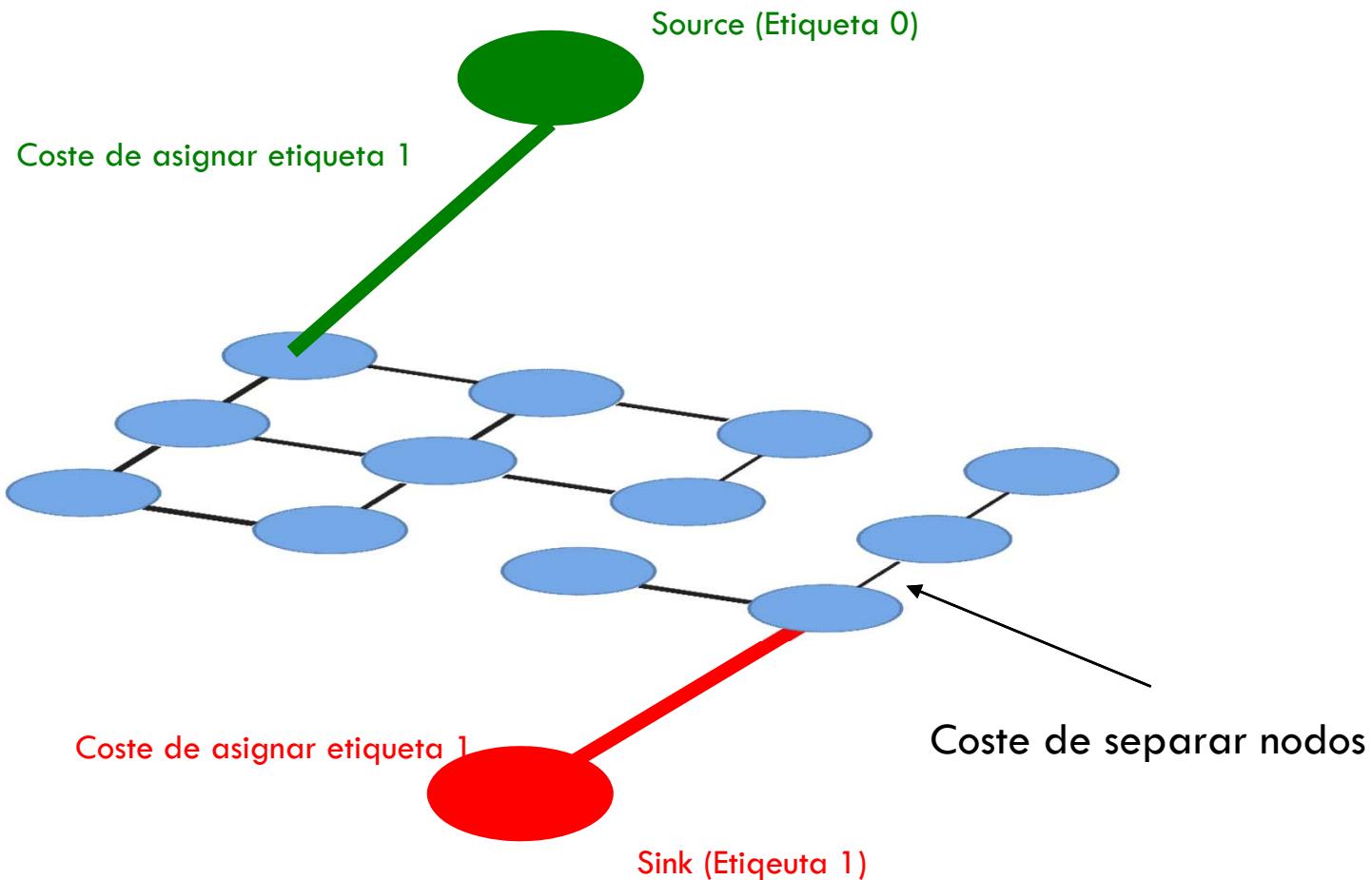
# Max-flow / Min-cut

101



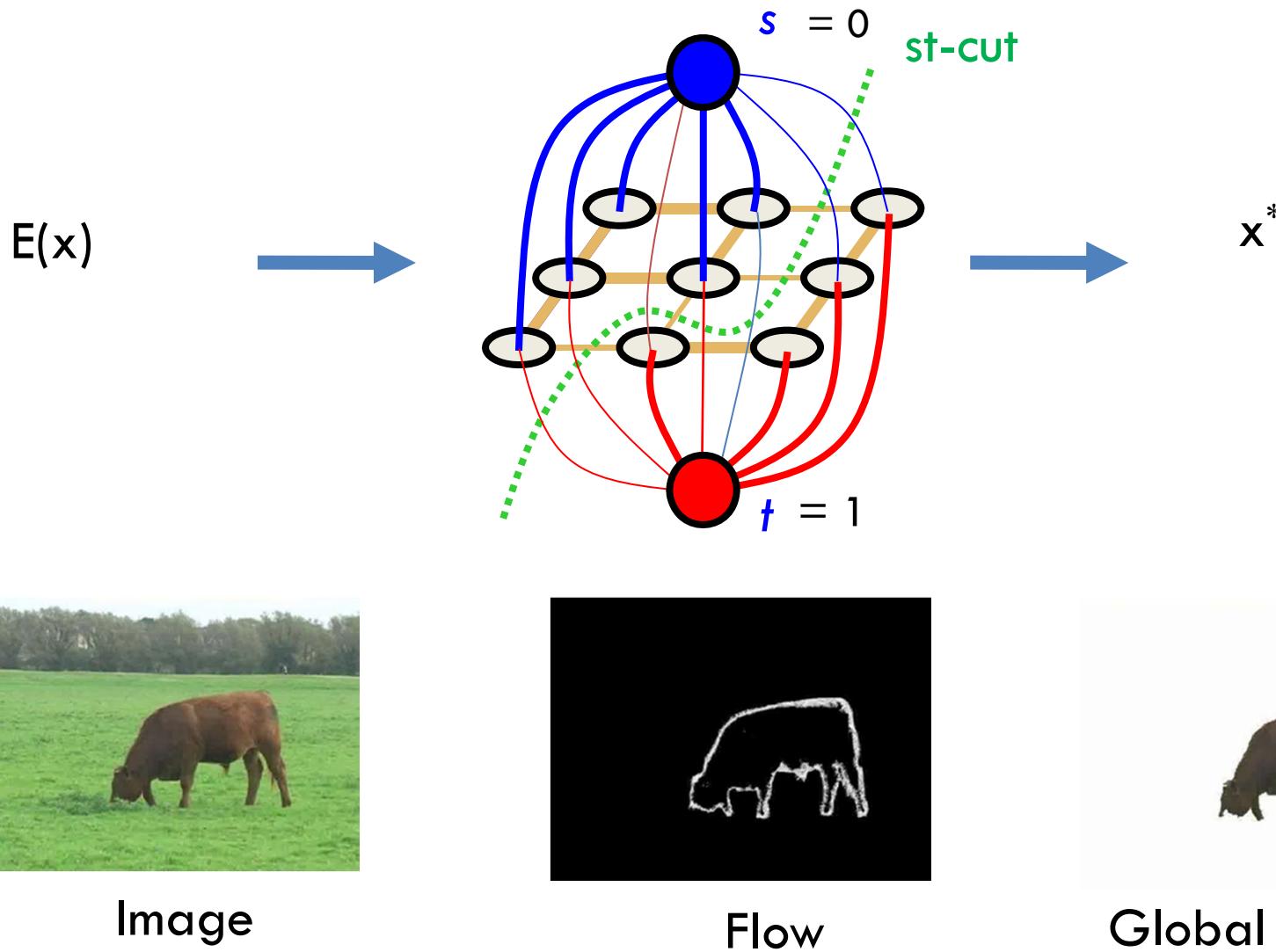
# Max-flow / Min-cut

102



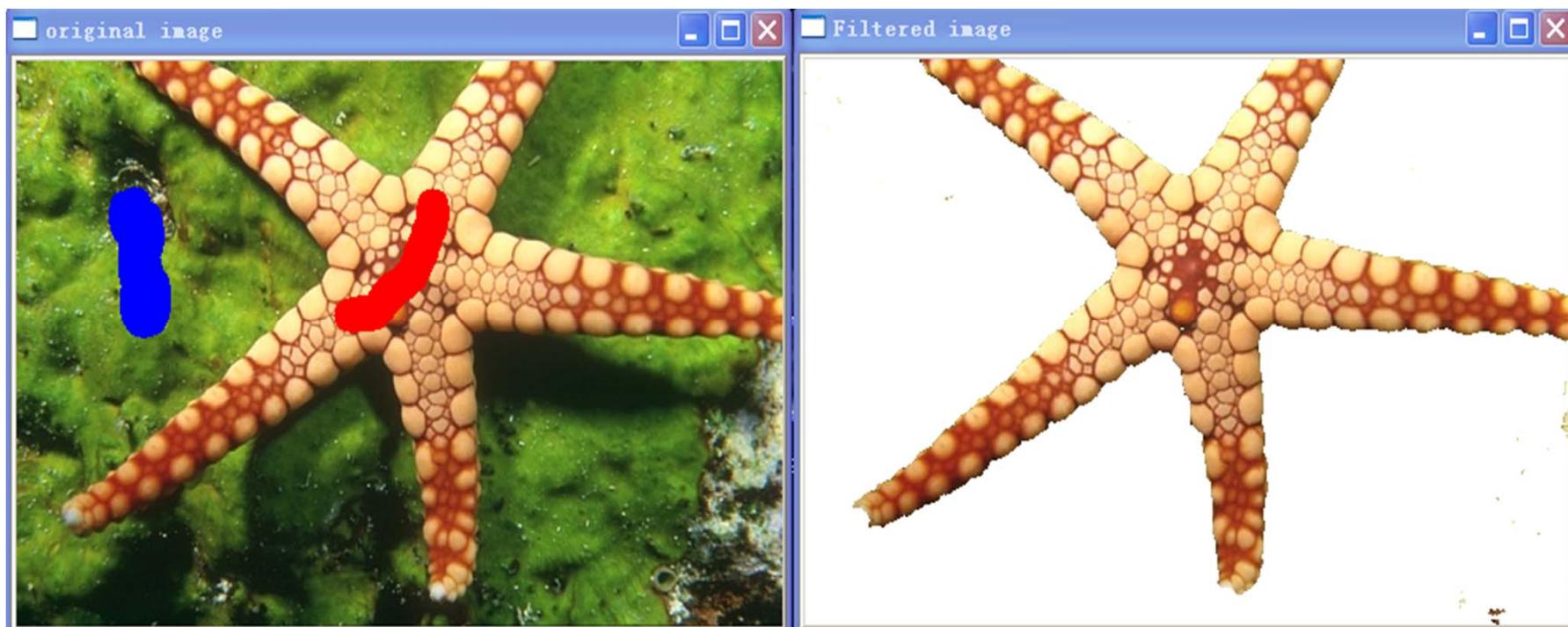
# Ejemplo

103



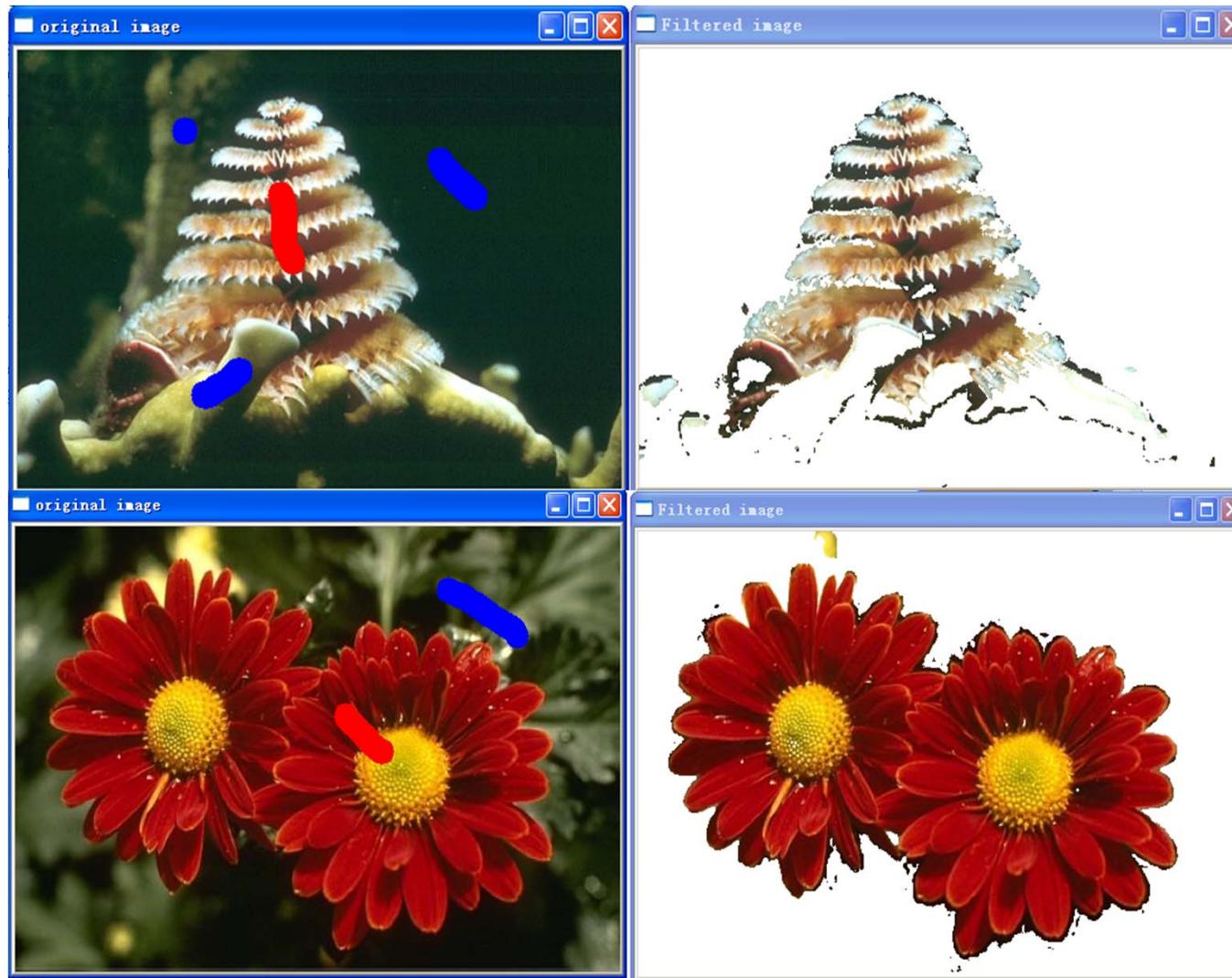
# Segmentación con marcadores y Graph Cut

104



# Ejemplos

105



# Segmentación directa: ventajas y desventajas

106

- **Decisión por segmentación directa:**
- **Ventajas:**
  - contornos precisos
  - controlan la conectividad de las regiones
- **Inconvenientes:**
  - Split and merge
    - División geométrica: resultados poco naturales
  - Fusión de regiones
    - requiere definición de
      - medida de similitud
      - criterio de terminación
    - el resultado depende de estos criterios, puede haber fugas, sobre o subsegmentación
  - Watershed:
    - requieren una segmentación parcial
      - marcadores
      - definición de una zona segura
    - el resultado depende de los marcadores iniciales
  - Graph cut
    - Caso binario
    - Requiere definición de función de energía a minimizar



# Segmentación: resumen

107

