

# Watershed

Texto e imagens: Livro Gonzalez &  
Woods

# Considerações

- Segmentação por Watershed incorpora muitos dos conceitos das outras abordagens
- Muitas vezes produz resultados mais estáveis
- Permite incorporar restrições baseadas em conhecimento.

# Apresentação

- Watershed (bacias hidrográficas)
- Visualizar a imagem em três dimensões



Imagem original  
(intensidades)



Visão “3D”

# Interpretação topográfica

- Existem 3 tipos de pontos
  - Pontos de mínimo local
  - Pontos em que, caso se despeje uma gota de água, esta se moveria até um local de mínimo local
  - Pontos em que gota d'água possui a mesma probabilidade de cair em mais de um mínimo local

# Interpretação topográfica

- Existem 3 tipos de pontos
  - Pontos de mínimo local +
  - Pontos em que, caso se despeje uma gota de água, esta se moveria até um local de mínimo local -> *catchment basin, watershed, bacia hidrográfica*
  - Pontos em que gota d'água possui a mesma probabilidade de cair em mais de um mínimo local -> *Linhas de divisão ou linhas de watershed*

# Watershed

- O principal objetivo desse tipo de segmentação é encontrar as linhas de *watershed*

# Watershed

- Como funciona: suponha que em cada mínimo local tenha um ponto que seja uma fonte de água.
- A água sobe inundando as regiões, de baixo para cima.
- A água sobe em uma taxa uniforme
- As bordas da imagem podem ser consideradas como sendo uma barragem mais alta que a maior montanha.

# Watershed

- Quando a água está prestes a se fundir nas diversas watersheds, uma barragem é construída para impedir a fusão
- A inundação chegará a um ponto em que somente apenas os topos das barragens serão visíveis acima da linha d'água
- Essas linhas são **as Linhas de divisão ou linhas de Watershed**
- Portanto, são as fronteiras conectadas que são extraídas em um algoritmo de segmentação Watershed.



# Exemplo



# Exemplo



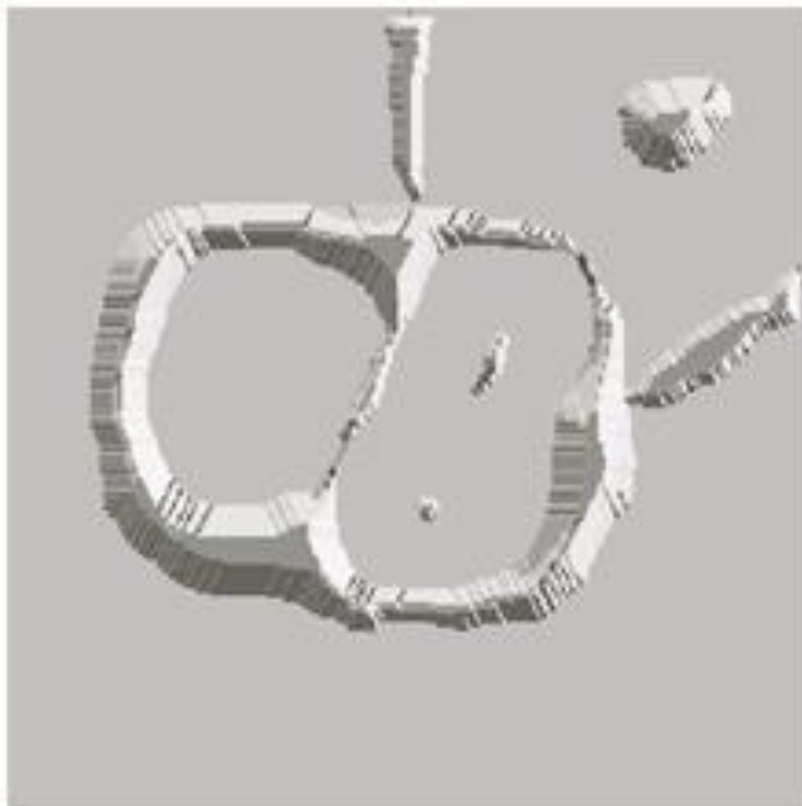
# Exemplo



# Exemplo



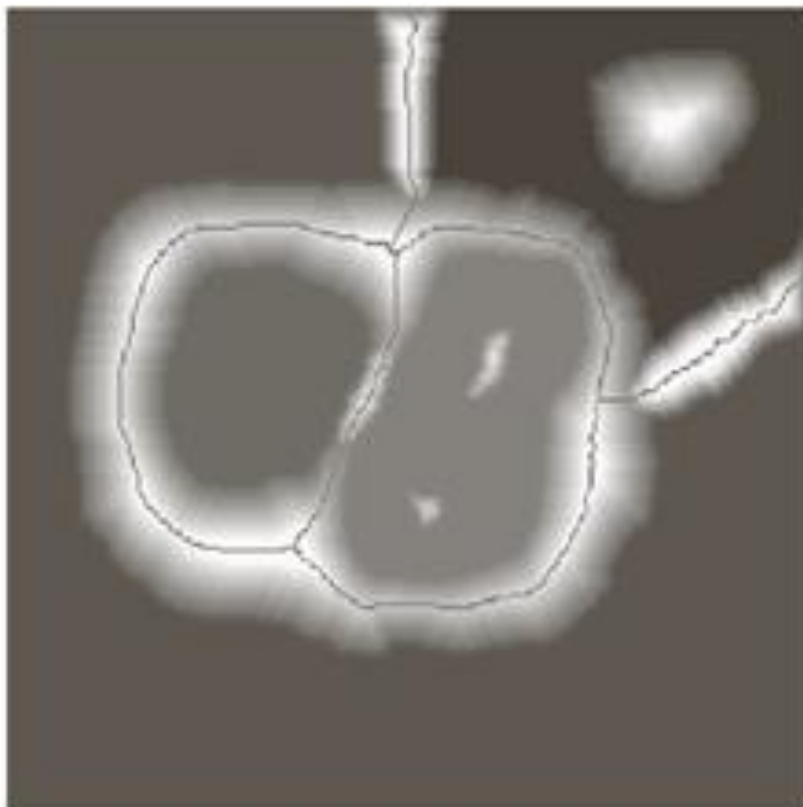
# Exemplo



# Exemplo



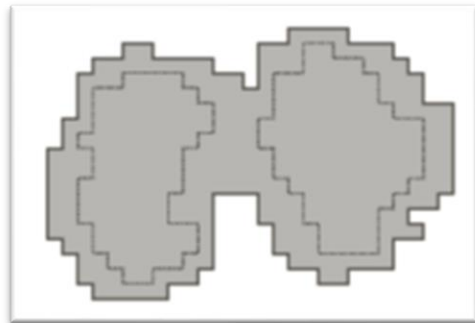
# Exemplo



# Construção de barragens



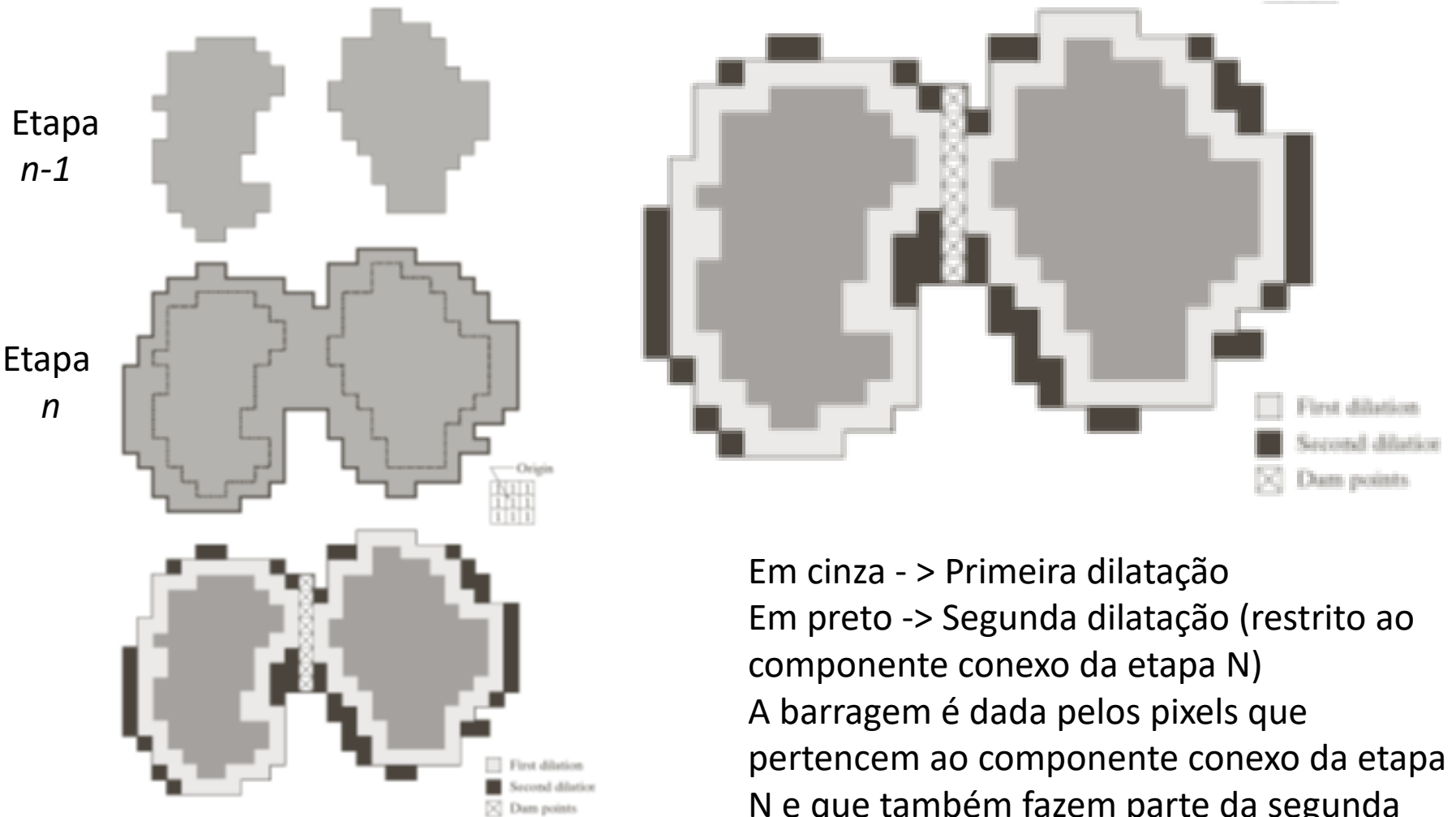
Duas *watersheds*  
(antes de grudar)  
Etapa  $n-1$



Duas *watersheds*  
Juntas  
Etapa  $n$



# Construção de barragens



Em cinza -> Primeira dilatação

Em preto -> Segunda dilatação (restrito ao componente conexo da etapa N)

A barragem é dada pelos pixels que pertencem ao componente conexo da etapa N e que também fazem parte da segunda dilatação

# Algoritmo Watershed

- Seja  $M_1, M_2, \dots, M_R$  os conjuntos que determinam as coordenadas dos mínimos regionais de uma imagem  $g(x, y)$
- Seja  $C(M_i)$  o conjunto dos pontos na *watershed* associados com o mínimo regional  $M_i$  (lembre que esse é um componente conexo)
- Seja  $min$  o valor mínimo de  $g(x, y)$
- Seja  $max$  o valor máximo de  $g(x, y)$

# Algoritmo Watershed

- Digamos que  $T[n]$  representa o conjunto de coordenadas  $(s, t)$  para os quais  $g(s, t) < n$
- $T[n] = \{(s, t) | g(s, t) < n\}$
- $T[n]$  é o conjunto de coordenadas de  $g(s, t)$  situados abaixo do plano  $g(s, t) = n$
- A topografia será inundada em incrementos inteiros, de  $n = \min + 1$  para  $n = \max + 1$

$M_i$  - coordenadas dos mínimos

$C(M_i)$  – conjunto de pontos associados a  $M_i$

$T[n]$  – Threshold de valor  $n$

$C_n(M_i) = C(M_i) \cap T[n]$  - Parte Binária associada a  $M_i$

# Algoritmo Watershed

- Seja  $C_n(M_i)$  denota o conjunto de coordenadas dos pontos na watershed associados ao mínimo  $M_i$  que são inundados na etapa  $n$
- $C_n(M_i) = C(M_i) \cap T[n]$ 
  - Isolamos a parte da imagem binária em  $T[n]$  que está associada ao mínimo regional  $M_i$

$M_i$  - coordenadas dos mínimos

$C(M_i)$  – conjunto de pontos associados a  $M_i$

$T[n]$  – Threshold de valor  $n$

$C_n(M_i) = C(M_i) \cap T[n]$  - Parte Binária associada a  $M_i$

# Algoritmo Watershed

- Seja  $C[n]$  a união das watersheds inundadas na etapa  $n$ :
- $C[n] = \bigcup_{i=1}^R C_n(M_i)$ 
  - Isolamos a parte da imagem binária em  $T[n]$  que está associada ao mínimo regional  $M_i$
- Então  $C[\max + 1]$  é a união de todas as watersheds
  - $C[\max + 1] = \bigcup_{i=1}^R C(M_i)$

$M_i$  - coordenadas dos mínimos  
 $C(M_i)$  - conjunto de pontos associados a  $M_i$

$T[n]$  - Threshold de valor  $n$

$C_n(M_i) = C(M_i) \cap T[n]$  - Parte Binária associada a  $M_i$

# Algoritmo Watershed

- Os elementos tanto em  $C_n(M_i)$  quanto em  $T[n]$  nunca são substituídos durante a execução do algoritmo.
- O número de elementos desses conjuntos aumenta ou permanece igual enquanto  $n$  aumenta

$$C_n \subset T[n] \quad (\subset - \text{subconjunto})$$

$$C[n-1] \subset C[n]$$

$\rightarrow C[n-1] \subset T[n] \rightarrow$  cada componente conectado de  $C[n-1]$  está contido em exatamente um componente conectado de  $T[n]$

# Algoritmo Watershed

- Inicialização do algoritmo
- $C[\min + 1] = T[\min + 1]$
- Obtendo  $C[n]$  a partir de  $C[n - 1]$
- Seja  $Q$  o conjunto de componentes conexos em  $T[n]$ . Então, para cada componente conectado  $q \in Q[n]$  existem três possibilidades

# Algoritmo Watershed

- Seja  $Q$  o conjunto de componentes conexos em  $T[n]$ . Então, para cada componente conectado  $q \in Q[n]$  existem três possibilidades
  - $q \cap C[n - 1]$  é vazia
  - $q \cap C[n - 1]$  contém um componente conectado em  $C[n - 1]$
  - $q \cap C[n - 1]$  contém mais de um componente conectado em  $C[n - 1]$

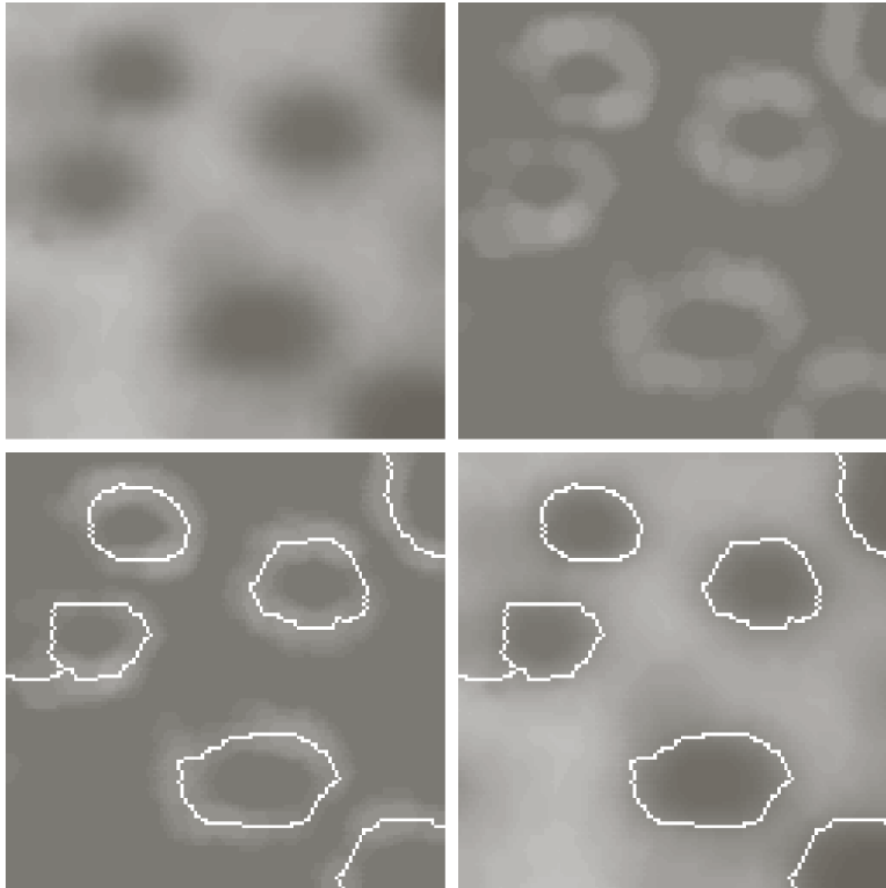


# Algoritmo Watershed

- $q \cap C[n - 1]$  é vazia
  - Um novo mínimo é encontrado.
  - Incorpore  $q$  a  $C[n - 1]$  para formar  $C[n]$
- $q \cap C[n - 1]$  contém um componente conectado em  $C[n - 1]$ 
  - $q$  fica dentro de um watershed
  - Incorpore  $q$  a  $C[n - 1]$  para formar  $C[n]$

# Algoritmo Watershed

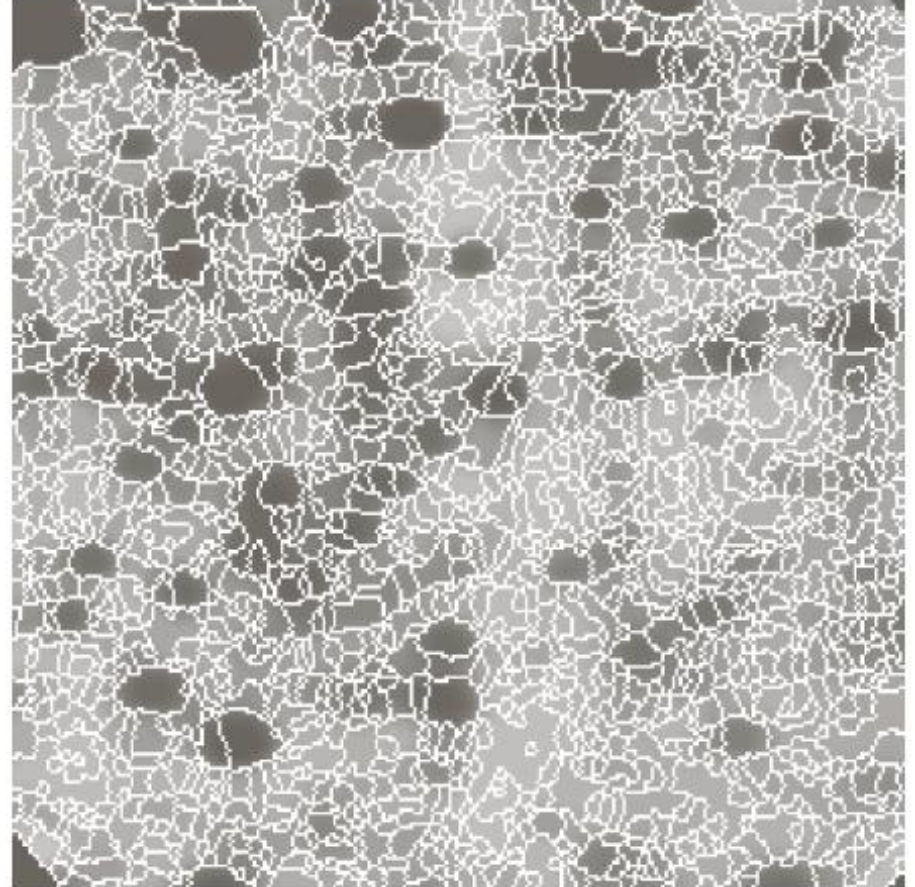
- $q \cap C[n - 1]$  contém mais de um componente conectado em  $C[n - 1]$ 
  - Duas cristas de watershed se encontrariam e iriam se misturar
  - Barragens devem ser construídas
  - Uma barragem de um pixel de espessura pode ser construída usando a dilatação  $q \cap C[n - 1]$  com um elemento estruturante 3x3 de 1s e restringindo a dilatação a  $q$ .



a	b
c	d

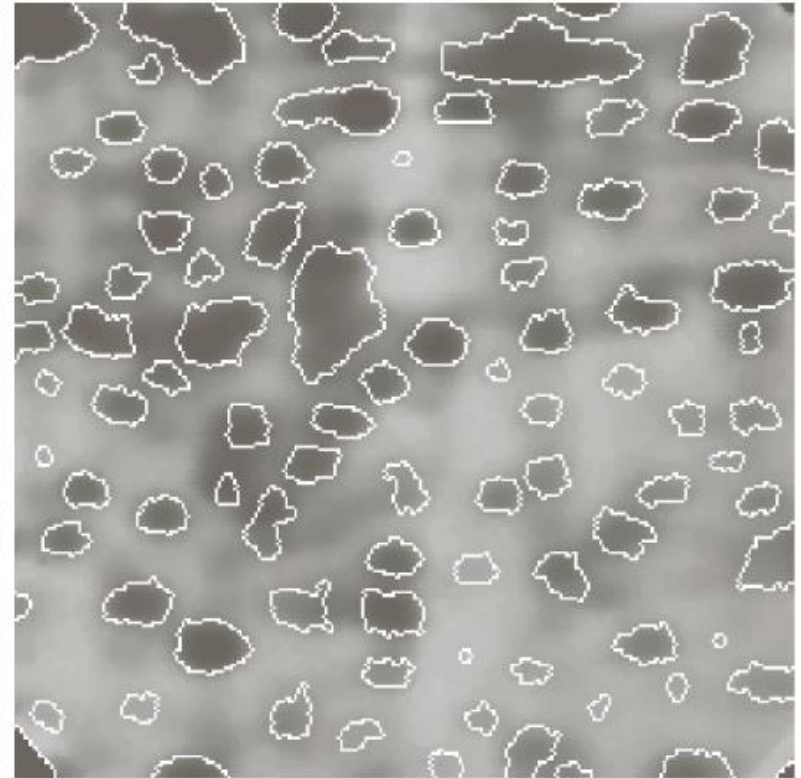
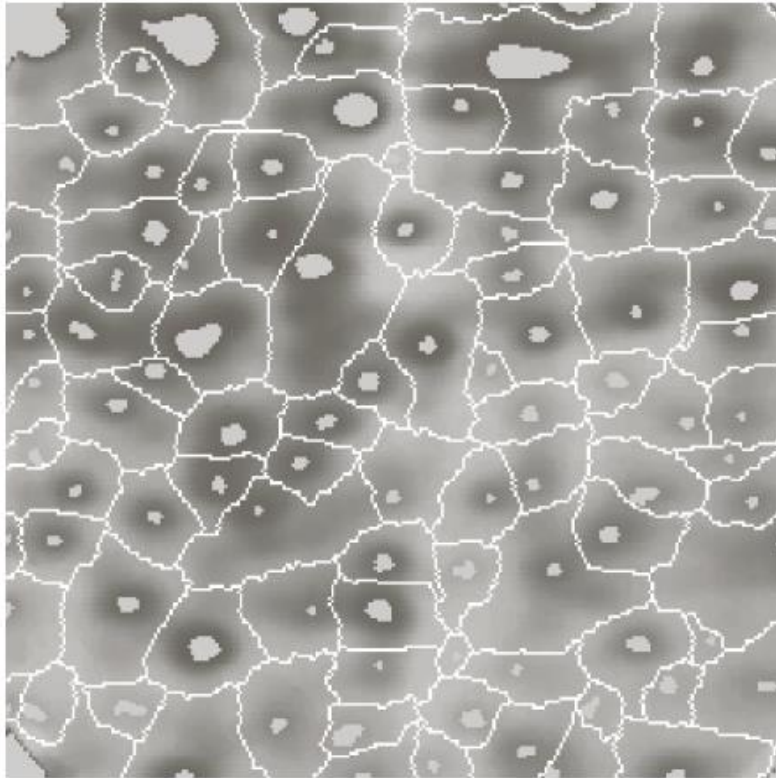
**FIGURE 10.56**

(a) Image of blobs.  
(b) Image gradient.  
(c) Watershed lines.  
(d) Watershed lines  
superimposed on  
original image.  
(Courtesy of Dr.  
S. Beucher,  
CMM/Ecole des  
Mines de Paris.)



a b

**FIGURE 10.57**  
(a) Electrophoresis image. (b) Result of applying the watershed segmentation algorithm to the gradient image. Oversegmentation is evident. (Courtesy of Dr. S. Beucher, CMM/Ecole des Mines de Paris.)



Segmentação feita dentro de cada região

a b

**FIGURE 10.58** (a) Image showing internal markers (light gray regions) and external markers (watershed lines). (b) Result of segmentation. Note the improvement over Fig. 10.47(b). (Courtesy of Dr. S. Beucher, CMM/Ecole des Mines de Paris.)