


Processamento de Imagens

Segmentação


Segmentação

- Segmentação  Análise de Imagem
 - “Divisão da imagem em partes fortemente relacionadas aos objetos e áreas do mundo real contidos na imagem”
- Completa: regiões disjuntas associadas a objetos distintos contidos na imagem
 - “high-level processing” + domain knowledge
- Parcial: regiões não necessariamente correspondem a objetos distintos
 - “low-level processing”

Segmentação Completa

- Pode requerer cooperação de estratégias de processamento de mais alto nível, que usam conhecimento específico do problema
- As vezes, apenas processamento de baixo nível é suficiente para ter uma solução
 - Independente de contexto
 - Não existe modelo para o objeto
 - Não existe conhecimento da segmentação esperada
 - Objetos contrastados em fundo mais escuro
 - identificação de células, caracteres, etc.

Segmentação Parcial

- Divisão da imagem em regiões homogêneas segundo uma propriedade
 - brilho, cor, textura
 - Segmentação de uma imagem aérea
- Mais viável  muito difícil obter segmentação completa de uma cena sempre
- Pode haver sobreposição de objetos

Classificação de Métodos de Segmentação

- Propriedades básicas: de descontinuidade e similaridade
- Conhecimento global (*histogram-based, limiarização*)
- Baseada em bordas (*edge-based segmentation*)
 - Descontinuidade entre regiões
- Baseada em Regiões (*region-based*)
 - Similaridade entre regiões

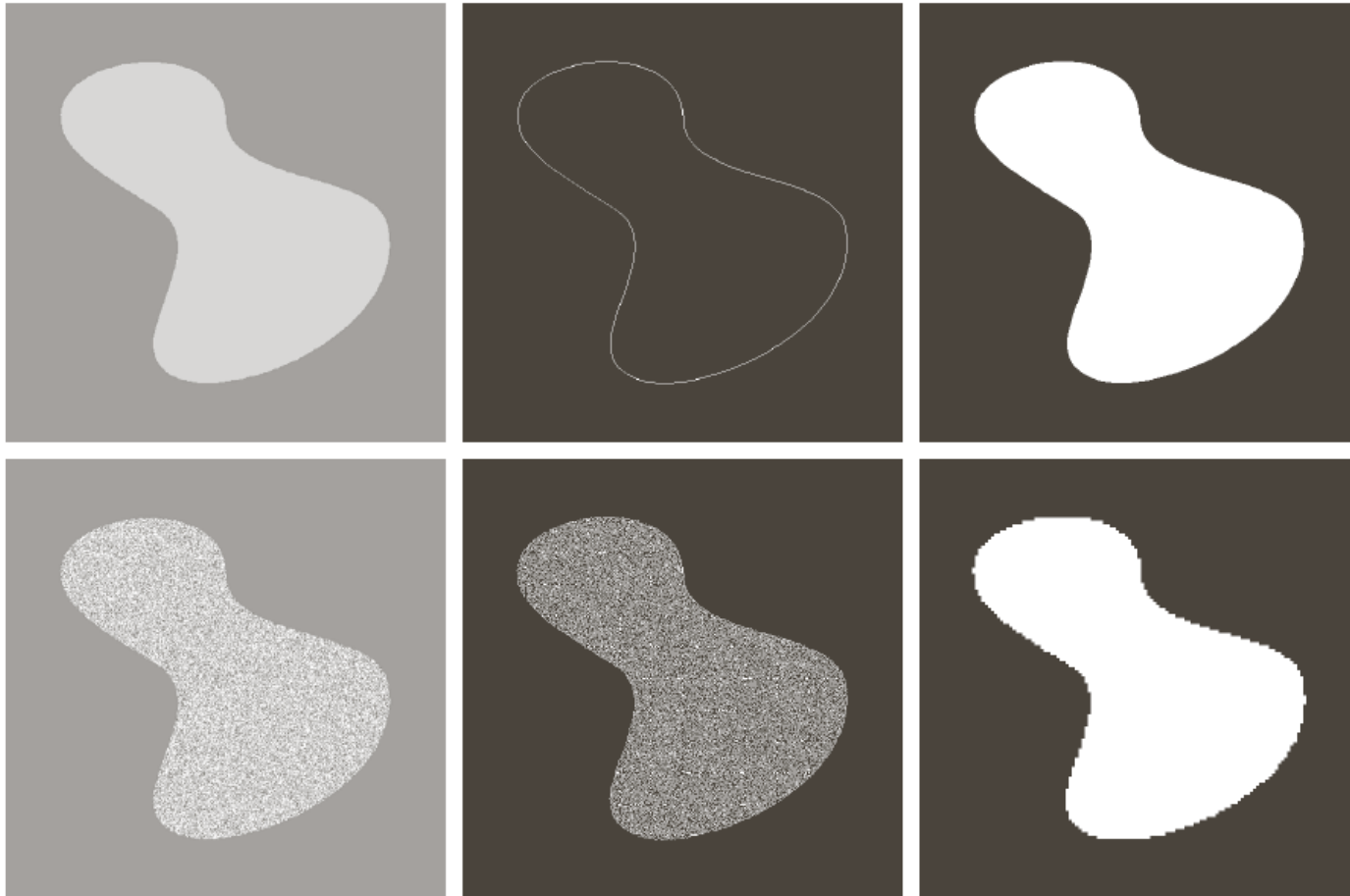


FIGURE 10.1 (a) Image containing a region of constant intensity. (b) Image showing the boundary of the inner region, obtained from intensity discontinuities. (c) Result of segmenting the image into two regions. (d) Image containing a textured region. (e) Result of edge computations. Note the large number of small edges that are connected to the original boundary, making it difficult to find a unique boundary using only edge information. (f) Result of segmentation based on region properties.

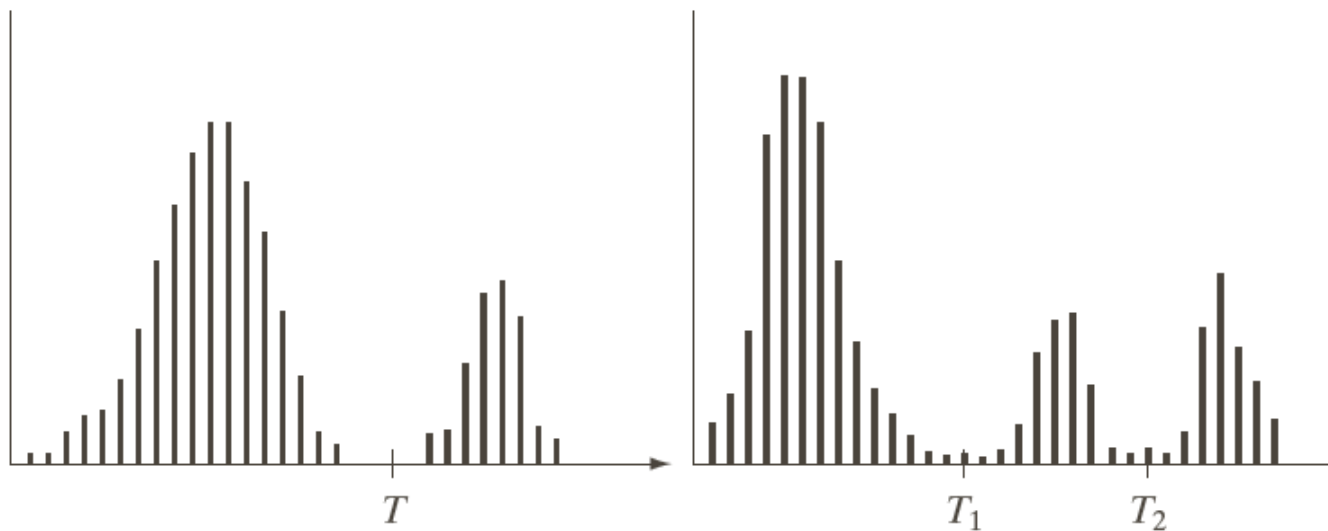
a	b	c
d	e	f

“Métodos” p/ segmentação

- Algoritmos Genéticos
- Redes Neurais
- Métodos Estocásticos (MRF)
- Morfologia Matemática
- Segmentação por Textura
- Segmentação por Cor

Limiarização (*Thresholding*)

- Estratégia simples de implementar e eficiente
- Usa a intensidade do pixel como separador



a b

FIGURE 10.35
Intensity histograms that can be partitioned (a) by a single threshold, and (b) by dual thresholds.

Limiarização (*Thresholding*)

- Dada uma imagem R , e um conjunto de regiões disjuntas R_i

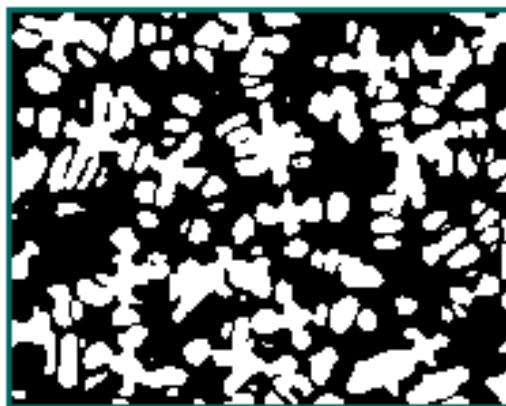
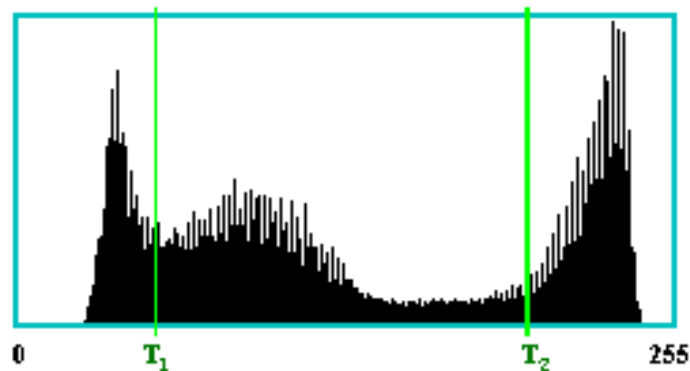
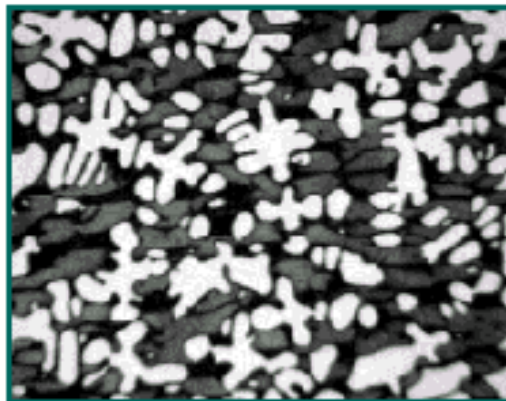
$$R = \bigcup_{i=1}^S R_i \quad R_i \cap R_j = \emptyset \quad i \neq j$$

- Limiarização (produz imagem binária g)

$$\begin{aligned} g(i, j) &= 1 && \text{for } f(i, j) \geq T \\ &= 0 && \text{for } f(i, j) < T \end{aligned}$$

- O sucesso da limiarização de intensidade está diretamente relacionado à largura e profundidade do(s) vale(s) que separa(m) os modos do histograma

Exemplos de *Thresholding*



Fase clara ($t > T_2$)



Fase média ($T_1 < t < T_2$)

problema !



Fase escura ($t < T_1$)



Thresholding

- Objetos que não se tocam
- Nível distinto do fundo (*background*)
- Determinação do limiar correto (*threshold*) é crucial
 - interativamente
 - automaticamente
 - global (menos provável): único limiar p/ imagem
$$T = T(f)$$
 - variável (dinâmica, adaptativa): limiar varia “ao longo” da imagem
$$T = T(f, f_c), \text{ } f_c \text{ é a região da imagem } f \text{ à qual } T \text{ é aplicado}$$

Tipos de Limiarização

- Por banda

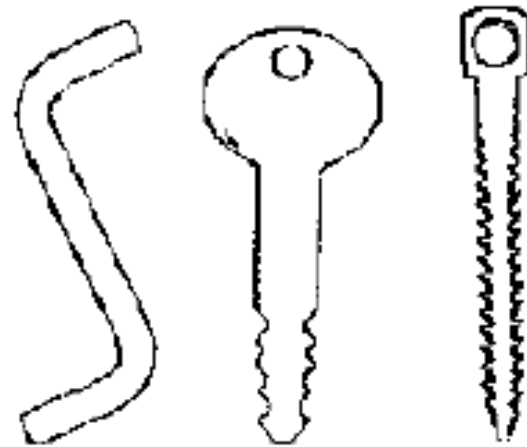
$$\begin{aligned} g(i, j) &= 1 && \text{for } f(i, j) \in D \\ &= 0 && \text{otherwise} \end{aligned}$$

D : conjunto de níveis de cinza

- tal definição de limiares também pode funcionar como um detector de bordas em imagens de objetos escuros sobre um fundo claro



Original image.



Border regions detected.

Tipos de Limiarização

- Multi-bandas
$$\begin{aligned}g(i, j) &= 1 && \text{for } f(i, j) \in D_1 \\ &= 2 && \text{for } f(i, j) \in D_2 \\ &= 3 && \text{for } f(i, j) \in D_3 \\ &= 4 && \text{for } f(i, j) \in D_4 \\ &\dots \\ &= n && \text{for } f(i, j) \in D_n \\ &= 0 && \text{otherwise}\end{aligned}$$

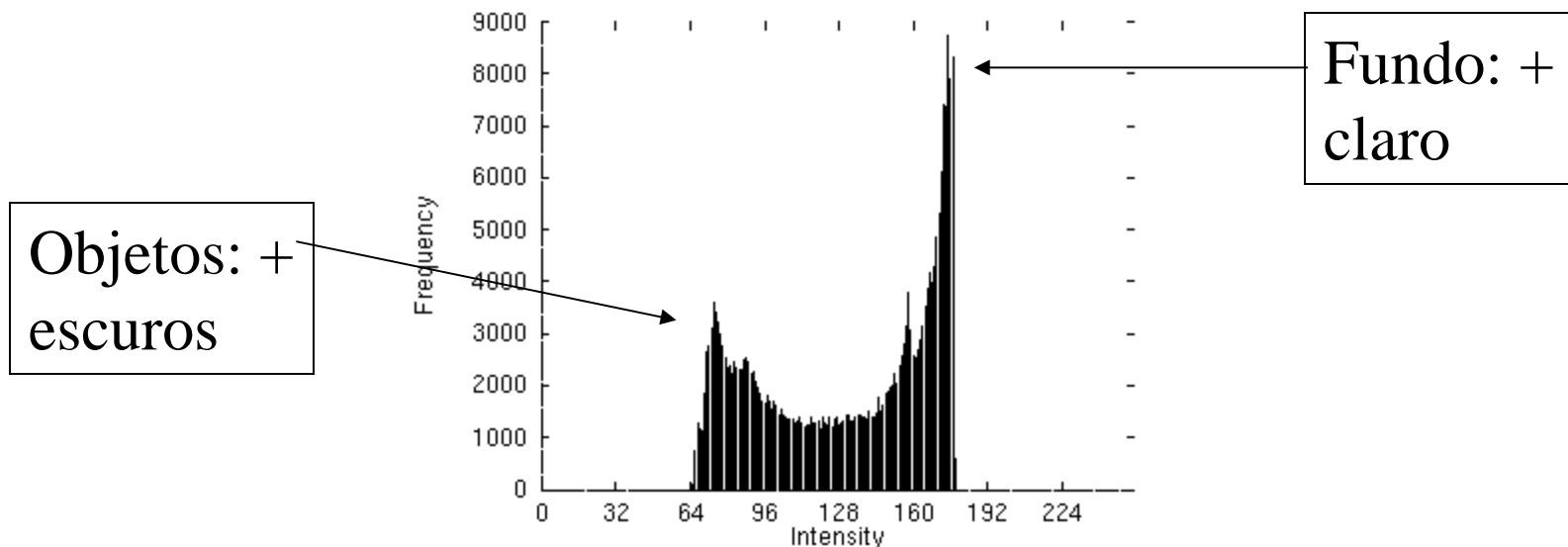
D : conjunto de níveis de cinza

Produz imagem com vários níveis de cinza, ao invés de binária

p.ex., como na imagem das células mostrada anteriormente

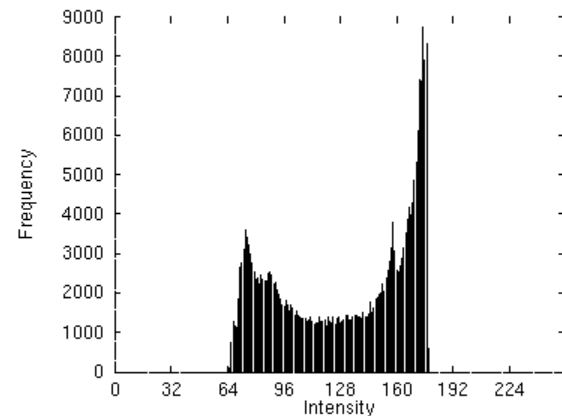
Determinação do limiar (T)

- Tem-se informação “*a priori*”? ➡ + simples
 - p.ex., caracteres em uma folha papel: fundo claro, caracteres cobrem 1/p da área da folha
- Análise do formato do histograma
 - bimodal: objetos de mesmo NC + fundo

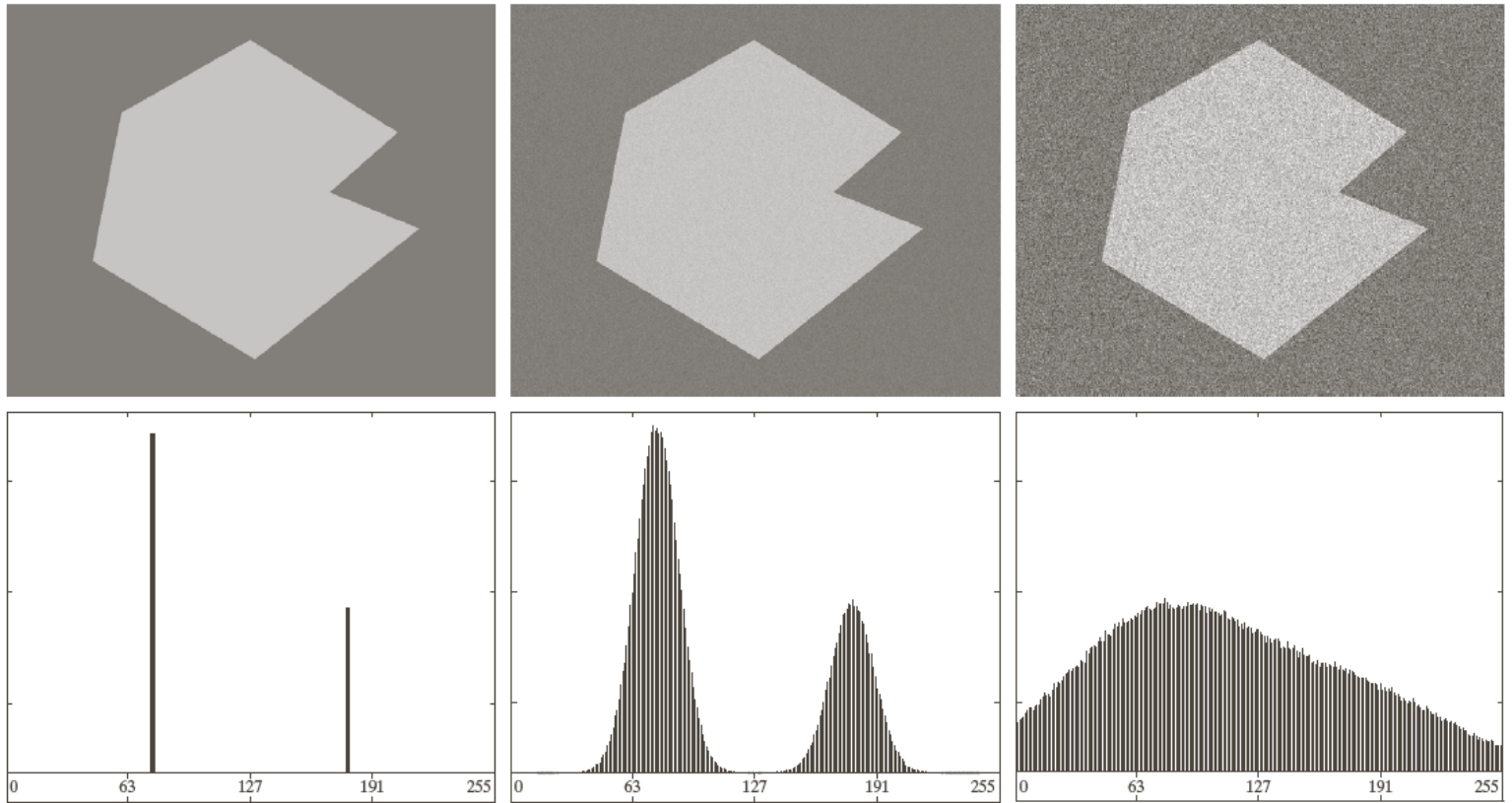


Determinação do limiar (T)

- Histograma bimodal: 2 picos que caracterizam objetos e fundo
 - Tons de cinza intermediários: não predominantes... (possivelmente bordas)
- Limiar candidato: o tom de cinza entre esses 2 picos com menor valor no histograma



Efeito do ruído



a	b	c
d	e	f

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

Efeito da iluminação e refletância

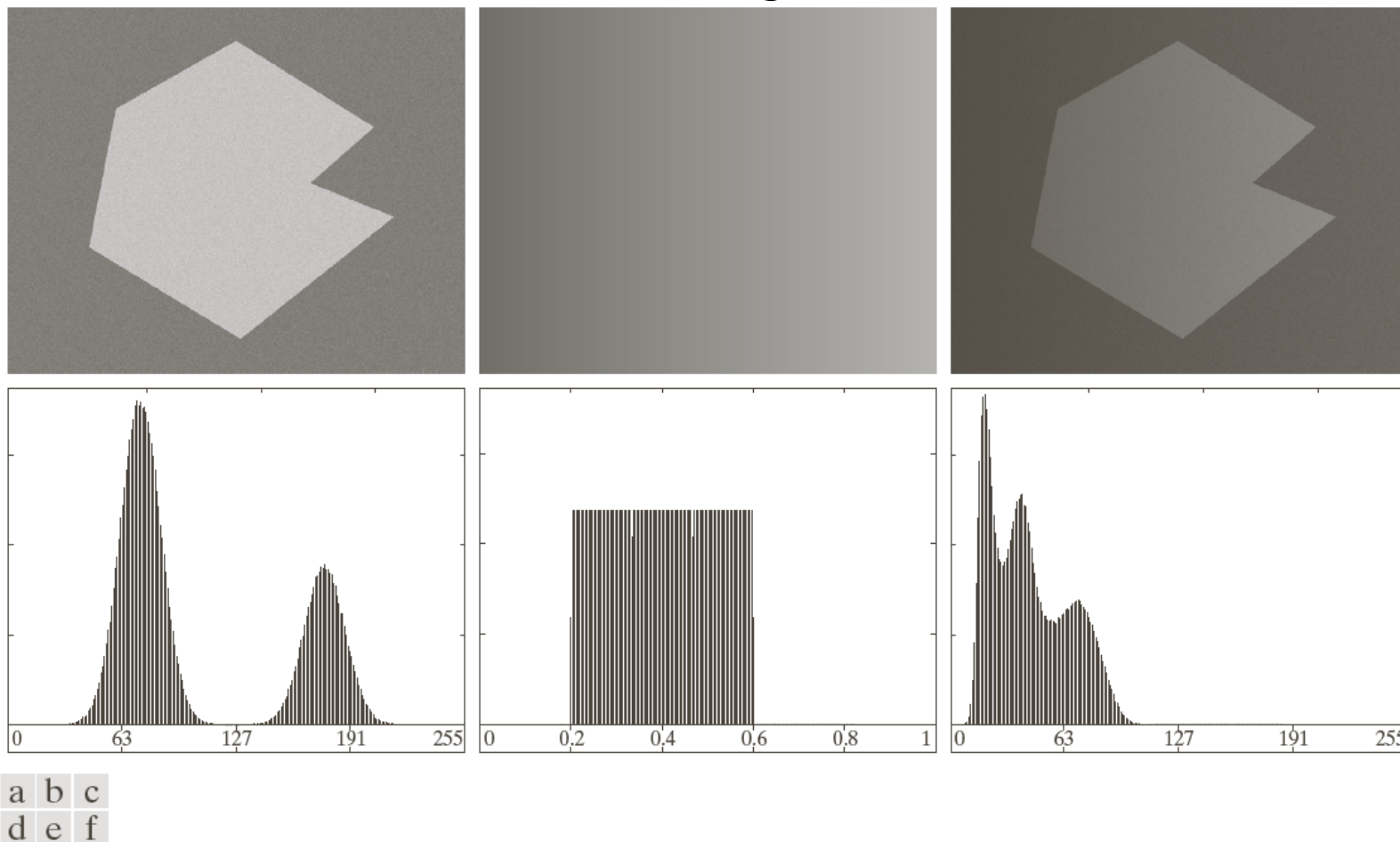


FIGURE 10.37 (a) Noisy image. (b) Intensity ramp in the range $[0.2, 0.6]$. (c) Product of (a) and (b). (d)–(f) Corresponding histograms.

Limiarização global simples

- Pode ser utilizada quando as distribuições de intensidade entre o fundo e os objetos são suficientemente diferentes
 - 'vale' no histograma
- Um único limiar para toda a imagem (global)
- Em muitas aplicações, no entanto, mesmo quando a abordagem global mostra-se adequada, a variabilidade das imagens é alta e o limiar muda, i.e., precisa ser determinado para cada imagem

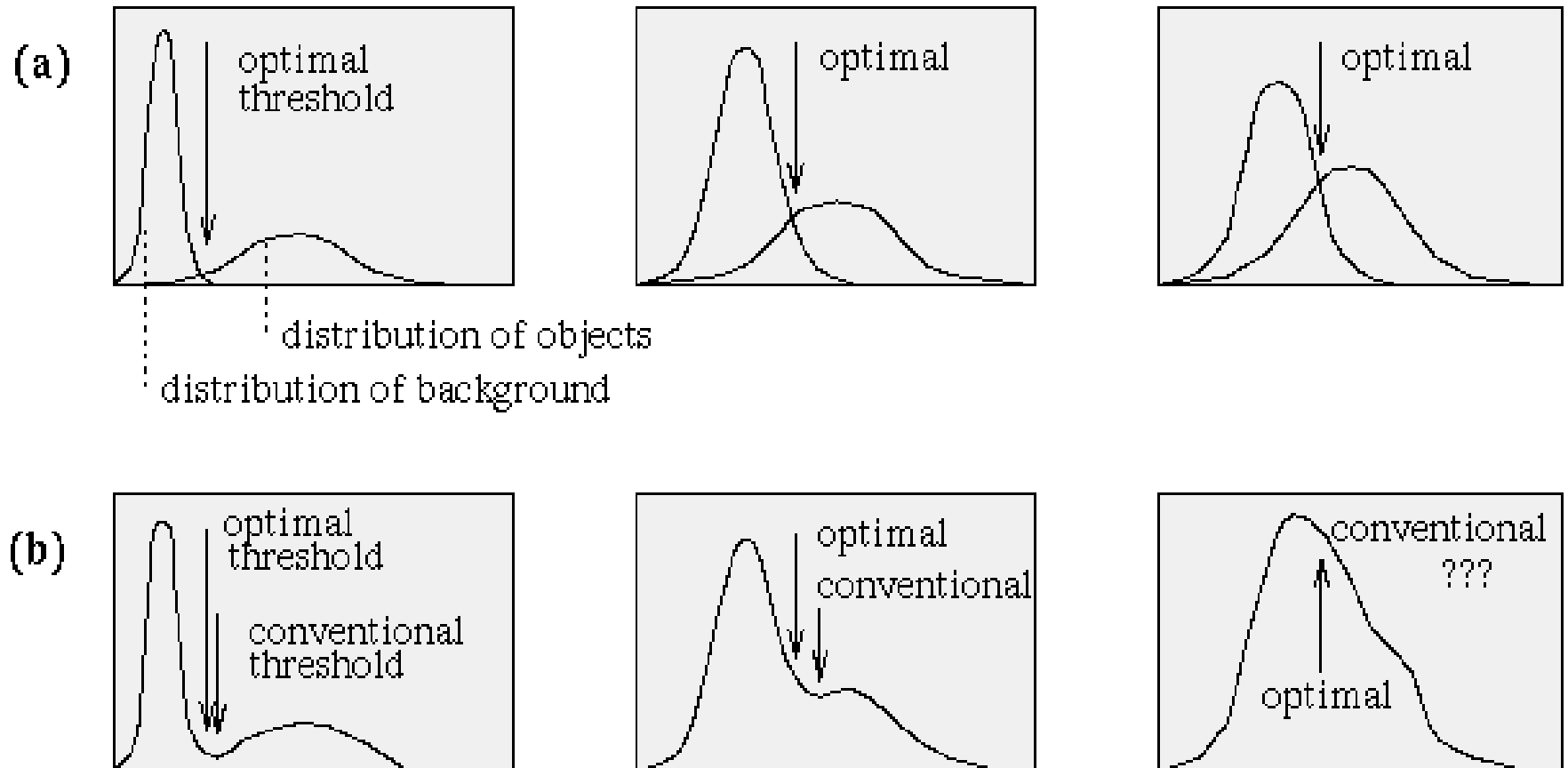
Algoritmo Limiarização Global Simples

1. Estimar um limiar global inicial, T
2. Segmentar a imagem usando a equação tradicional de limiarização. Dois grupos se formarão: G_1 com pixels $> T$ e G_2 com pixels $\leq T$
3. Calcular a intensidade média dos pixels em cada grupo, m_1 e m_2
4. Calcular novo $T = (m_1 + m_2)/2$
5. Repetir as etapas 2 a 4, até que a diferença entre interações sucessivas de T seja inferior a um ΔT

Limiar Ótimo

- Limiarização como um problema teórico de decisão estatística
 - Objetivo: minimizar o erro médio incorrido na atribuição dos pixels a 2 (ou +) grupos (classes)
- Limiar ótimo: nível de cinza mais próximo ao ponto de mínimo entre dois pontos de máximo de duas (ou +) distribuições normais

Optimal Thresholding



Método de Otsu (1979)

- Idéia bastante simples: encontrar o limiar que minimiza a variância ponderada *intra* classes (*within-class variance*)
 - Isso equivale a maximizar a variância *entre* classes (*between-class variance*)
- Tudo se baseia em cálculos no vetor 1D do histograma da imagem

Método de Otsu

- Assume
 - Histograma (e imagem) bimodal
 - Não explora coerência espacial ou qualquer tipo de ‘estrutura’ dos objetos da imagem
 - Iluminação uniforme (comportamento bimodal resulta tão somente das diferenças entre os objetos da imagem)

Método de Otsu (+ custoso)

- A variância ponderada intra classes é a soma ponderada das variâncias de cada classe

$$\sigma_w^2(t) = q_1(t)\sigma_1^2(t) + q_2(t)\sigma_2^2(t)$$

- As probabilidades das classes são dadas por

$$q_1(t) = \sum_{i=1}^t P(i)$$

$$q_2(t) = \sum_{i=t+1}^I P(i)$$

Método de Otsu (+ custoso)

e as médias das classes são dadas por

$$\mu_1(t) = \frac{1}{q_1(t)} \sum_{i=1}^t iP(i) \quad \mu_2(t) = \frac{1}{q_2(t)} \sum_{i=t+1}^l iP(i)$$

Método de Otsu (+ custoso)

As variâncias individuais das classes

$$\sigma_1^2(t) = \sum_{i=1}^t [i - \mu_1(t)]^2 \frac{P(i)}{q_1(t)}$$

$$\sigma_2^2(t) = \sum_{i=t+1}^I [i - \mu_2(t)]^2 \frac{P(i)}{q_2(t)}$$

Pronto. Precisamos apenas percorrer toda a faixa de valores de T $[1, 256]$ e escolher o valor que minimiza $\sigma_w^2(t)$

– Isso é um pouco custoso...

Otsu (variâncias)

- Ao subtrairmos a variância intra classes da variância total do histograma (ambas as classes), obtemos a variância entre classes (between-class variance)

$$\begin{aligned}\sigma^2_{\text{Between}}(T) &= \sigma^2 - \sigma^2_{\text{Within}}(T) \\ &= q_1(T) [\mu_1(T) - \mu]^2 + q_2(T) [\mu_2(T) - \mu]^2\end{aligned}$$

- Note que a variância entre classes é apenas a variância ponderada da média dos clusters sobre a média global

Otsu

- Sejam L os níveis de cinza de uma imagem $M \times N$
- n_i é o nro de pixels com intensidade i
- $p_i = n_i / MN$

$$\sum_{i=0}^{L-1} p_i = 1$$

- Suponha um limiar k , para separar duas classes C_1 e C_2
- C_1 com pixels no intervalo $[0, k]$
- C_2 com pixels no intervalo $[k, L-1]$

Otsu

- É fácil vermos que a probabilidade $p_1(k)$, de que um pixel pertença à classe C_1 é:

$$P_1(k) = \sum_{i=0}^k p_i$$

e analogamente,

$$P_2(k) = \sum_{i=k+1}^{L-1} p_i = 1 - P_1(k)$$

Otsu

- Sabemos ainda que a intensidade média dos pixels de C_1

$$m_1(k) = \sum_{i=0}^k iP(i/C_1)$$

$$\dot{=} \sum_{i=0}^k iP(C_1/i) P(i) / P(C_1)$$

$$\dot{=} \frac{1}{p_1(k)} \sum_{i=0}^k ip_i$$

e

$$m_2(k) = \frac{1}{p_2(k)} \sum_{i=k+1}^{L-1} ip_i$$

Otsu

- A média acumulada (intensidade média) até o nível k é dada por

$$m(k) = \sum_{i=0}^k ip_i$$

e a intensidade média de toda a imagem (global) é

$$m_G = \sum_{i=0}^{L-1} ip_i$$

Por substituição direta das fórmulas anteriores, temos*

$$p_1 m_1 + p_2 m_2 = m_G \qquad p_1 + p_2 = 1$$

* Por clareza, foram omitidos os termos “k”

Otsu

- Para avaliar a qualidade do limiar no nível k , usamos métrica normalizada

$$\eta = \frac{\sigma_B^2}{\sigma_G^2}$$

onde

$$\sigma_G^2 = \sum_{i=0}^{L-1} (i - m_G)^2 p_i$$

e a variância entre classes (*between-class variance*)

$$\sigma_B^2 = P_1 (m_1 - m_G)^2 + P_2 (m_2 - m_G)^2$$

Otsu

- A expressão anterior pode ser escrita por

$$\begin{aligned}\sigma_B^2 &= P_1 P_2 (m_1 - m_2)^2 \\ &= \frac{(m_G P_1 - m)^2}{P_1 (1 - P_1)}\end{aligned}$$

- Quanto mais distantes entre si forem m_1 e m_2 , maior a variância. Sendo a variância global uma constante, entendemos que η também é uma medida de separabilidade e maximizá-la equivale a maximizar a variância entre classes. Portanto, basta computarmos para todo k e escolhermos o valor máximo!

$$\eta(k) = \frac{\sigma_B^2(k)}{\sigma_G^2}$$

Imagem original
com iluminação
irregular

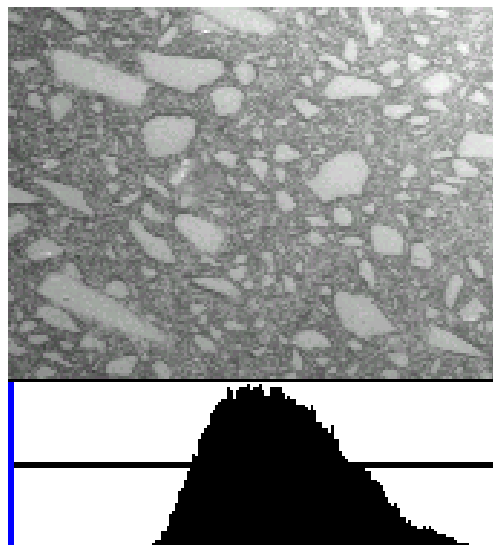


Imagem corrigida
com filtro passa-
altas, 49x49, 4
iterações

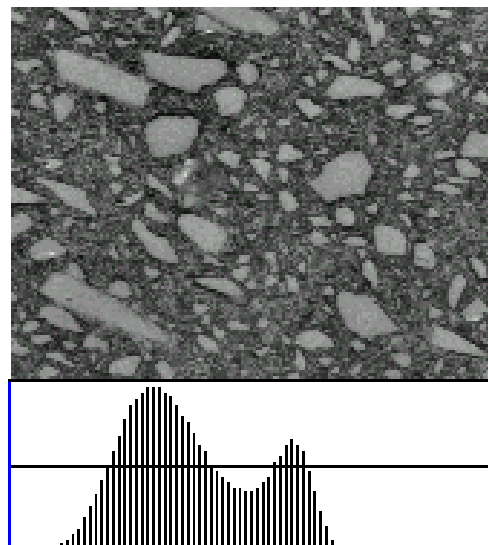


Imagem
segmentada com
limiar no mínimo

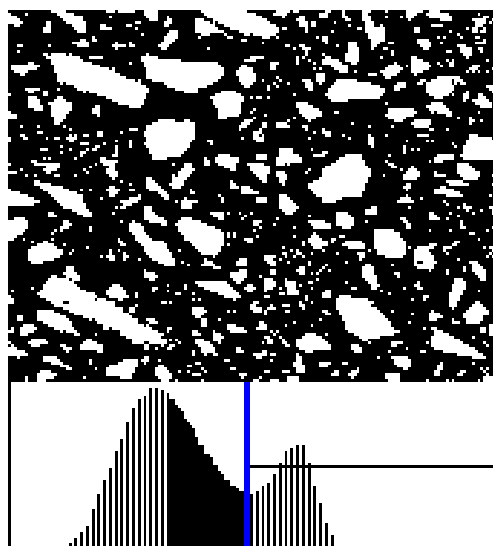
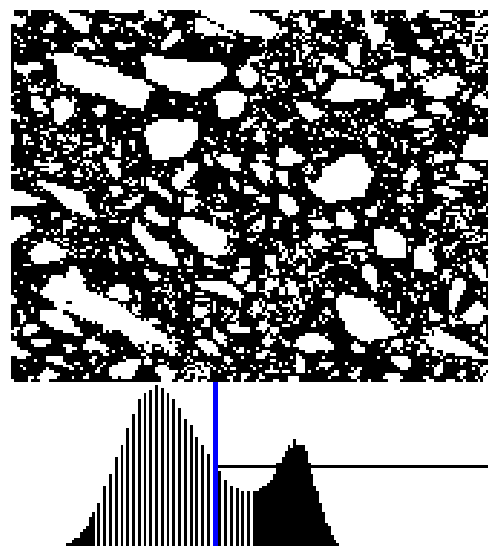
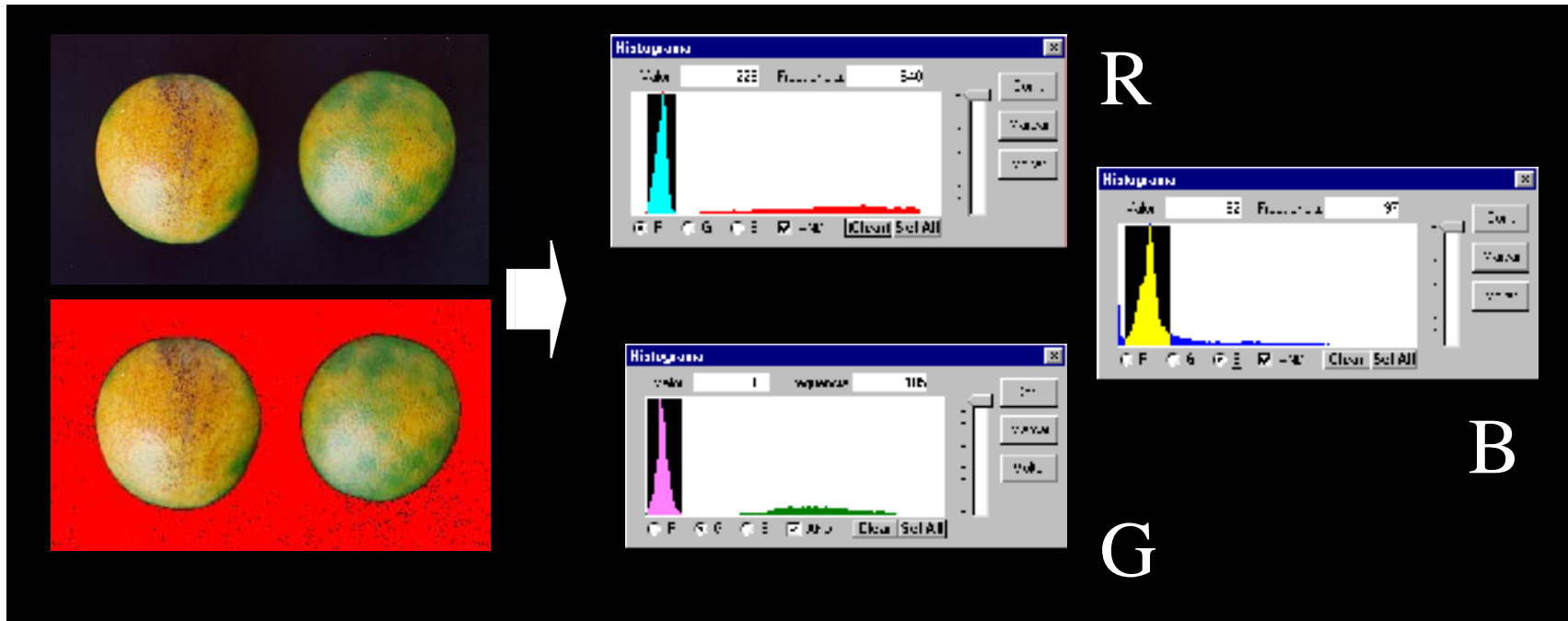


Imagem
segmentada pelo
método de Otsu



Limiarização Multi-Espectral



Segmentação por Bordas

- Depende de arestas das imagens localizados pelos operadores de bordas
 - arestas: descontinuidades no nível de cinza, cor, textura, etc.
- Muito susceptíveis ao ruído (presença de arestas onde não há bordas ou vice-versa)

Segmentação por Bordas

- Não basta apenas identificar as bordas e construir um contorno a partir delas
- É preciso identificar o objeto a partir desses contornos
- Objeto: região delimitada por um contorno

Seg. Bordas: limiarização

- Operação de detecção de bordas
- Aplicar *Thresholding* sobre a imagem de arestas (*edge image*)
 - Valor de mínimo ou média do histograma
- Ligar os segmentos de bordas em um contorno (*edge linking*)
 - Os pontos preservados pelo *thresholding* são ligados aos seus vizinhos, segundo uma distância máxima

Seg. Bordas: limiarização

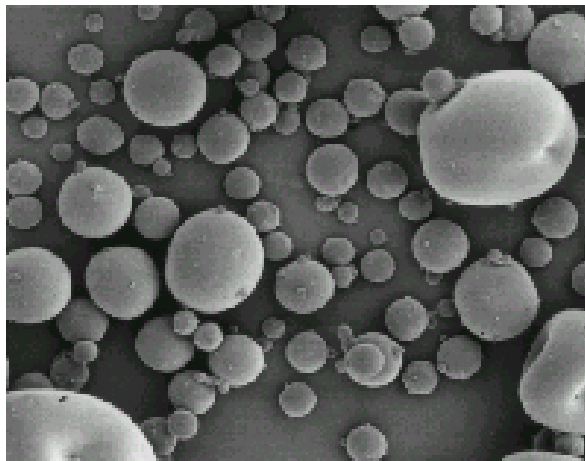
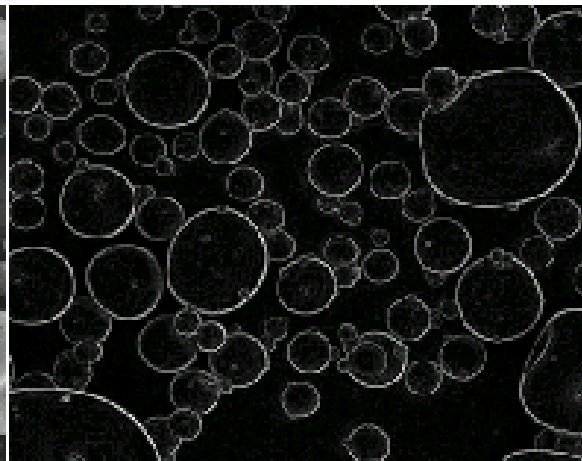


Imagem original



Sobel



Sobel limiarizado

Supressão de arestas

- Uma imagem de arestas (*edge image*) apresenta borda ‘espessa’, mas a borda deve ser composta por um pixel simples
 - Discretizar as direções de acordo com vizinhança 8-conectada
 - Para cada pixel com magnitude de aresta não-zero, inspecione os dois pixels adjacentes (segundo a direção da aresta)
 - Se a magnitude de um deles for maior que a do pixel corrente, marque-o para remoção
 - Percorra novamente a imagem e zere os pixels marcados para remoção

Seg. por Rastreamento de Bordas

- *Border Tracing* (ou *tracking*)
- Depende do grau de conectividade utilizado
 - 4-conectado 8-conectado
- A borda vai sendo composta aplicando-se a cada pixel um operador que define a direção do gradiente (que indica a posição do próximo pixel da borda)
- Objetivo: encontrar uma “*inner region border*”

Seg. Bordas: transformada de Hough

- Objetos têm tamanho e forma conhecidos
 - localizar áreas em circuitos impressos
- Podemos usar máscaras de convolução
 - Problemas: *zoom*, rotação, distorções, etc.
- Transformada de Hough: projetada inicialmente para identificar linhas. Pode ser generalizada para localização de círculos e outras curvas

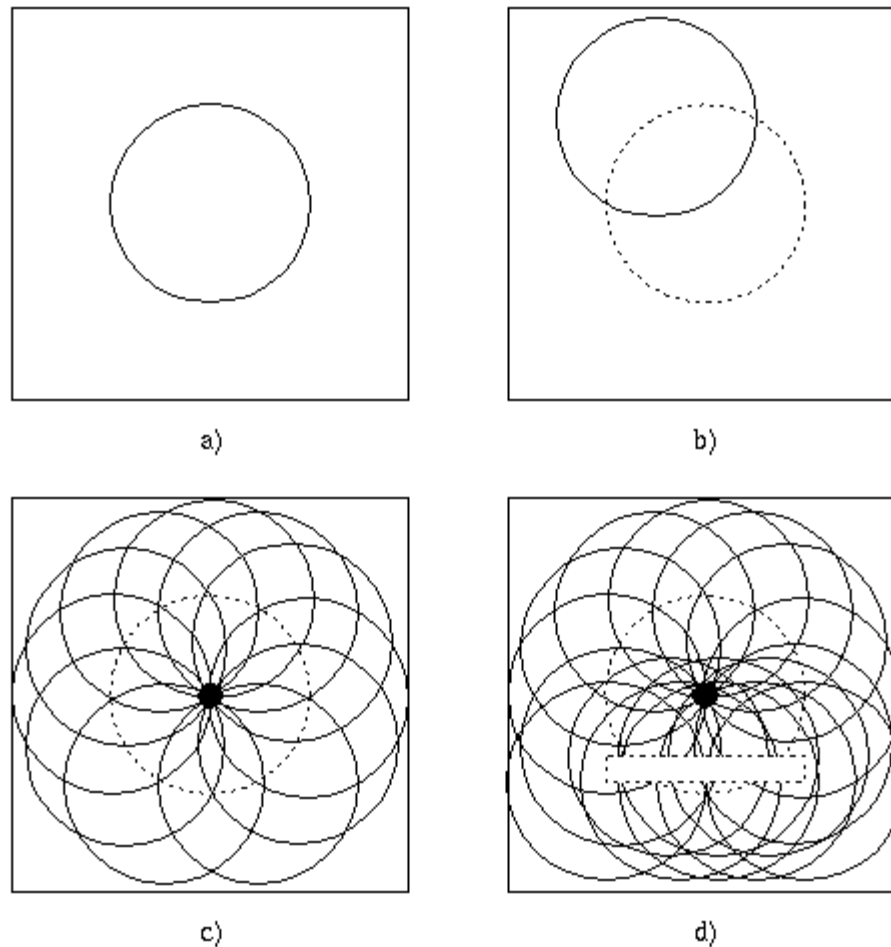
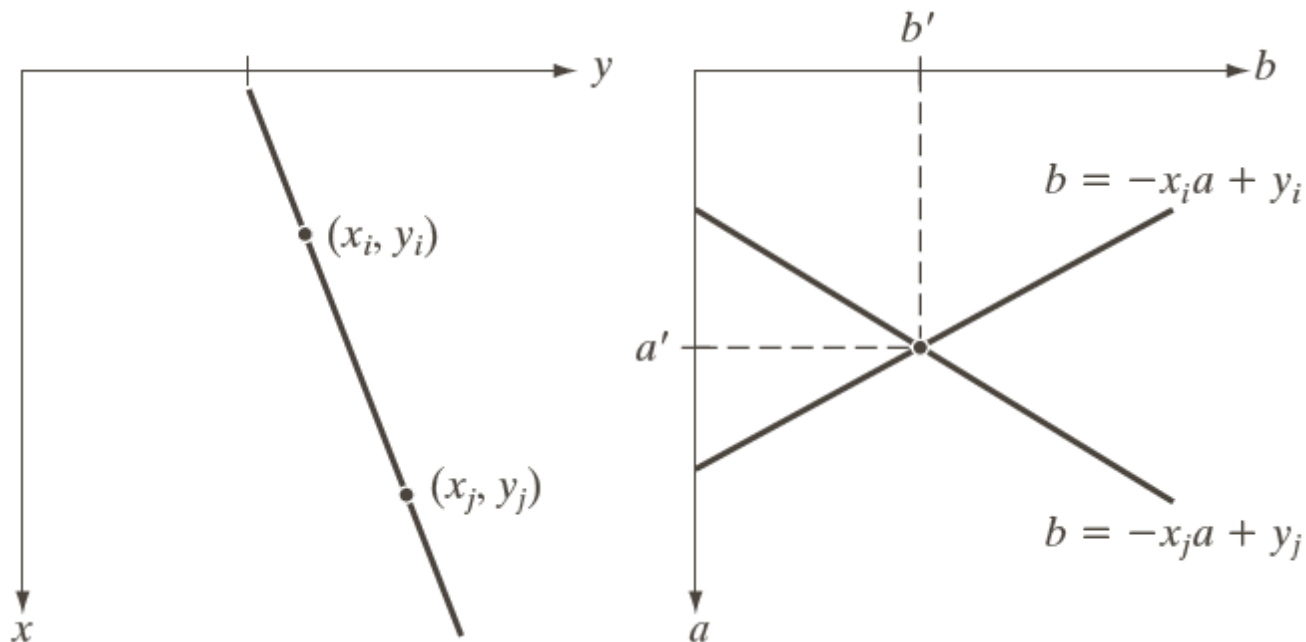


Figure 5.29 *Hough transform - example of circle detection. (a) Original image of a dark circle (known radius r) on a bright background, (b) for each dark pixel, a potential circle-center locus is defined by a circle with radius r and center at that pixel, (c) the frequency with which image pixels occur in the circle-center loci is determined; the highest-frequency pixel represents the center of the circle (marked by \bullet), (d) the Hough transform correctly detects the circle (marked by \bullet) in the presence of incomplete circle information and overlapping structures (see Figure 5.34 for a real-life example).*

Transformada de Hough

- Ideia subjacente: tendo n pontos em uma imagem, queremos encontrar subconjuntos desses pontos que pertencem ao elemento de interesse...
- Suponha que queremos encontrar linhas retas...
- Eq. da reta: $y_i = ax_i + b$

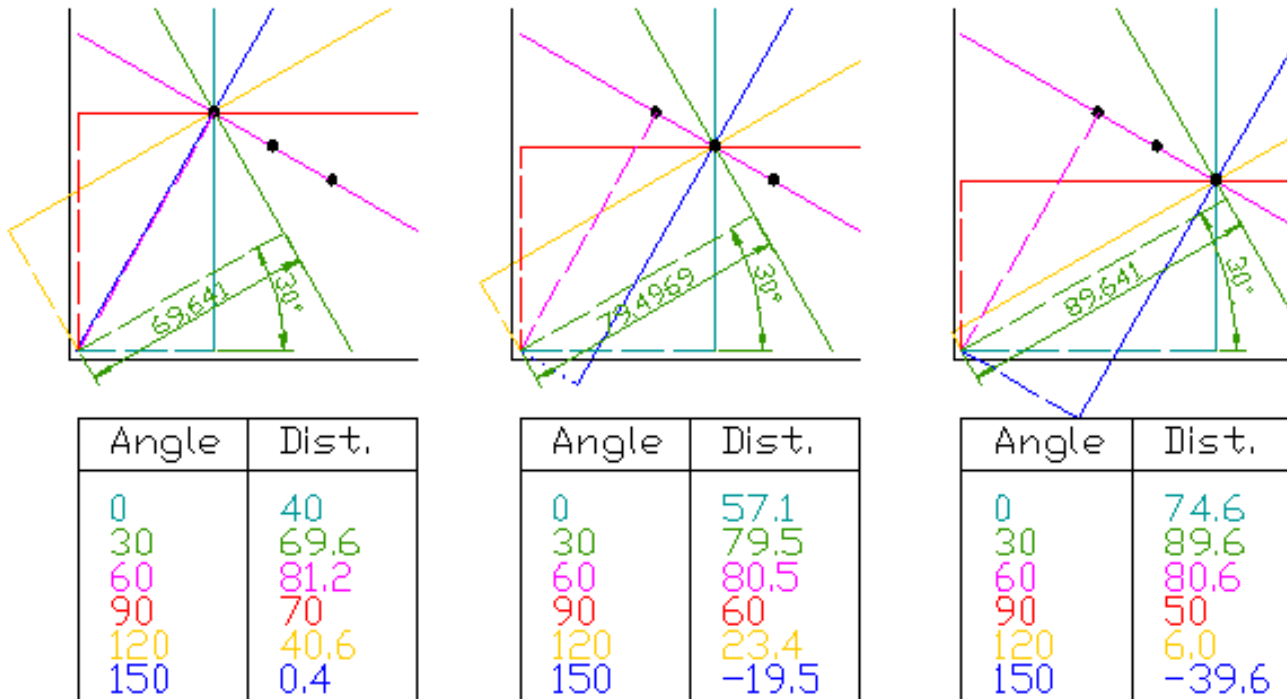
Transformada de Hough



a b

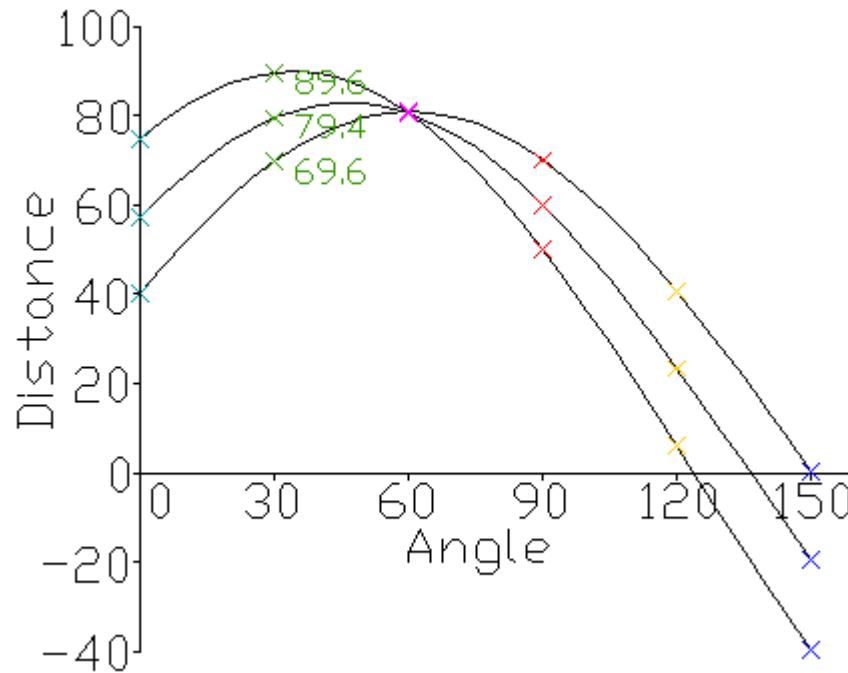
FIGURE 10.31
(a) xy -plane.
(b) Parameter space.

Transformada Hough



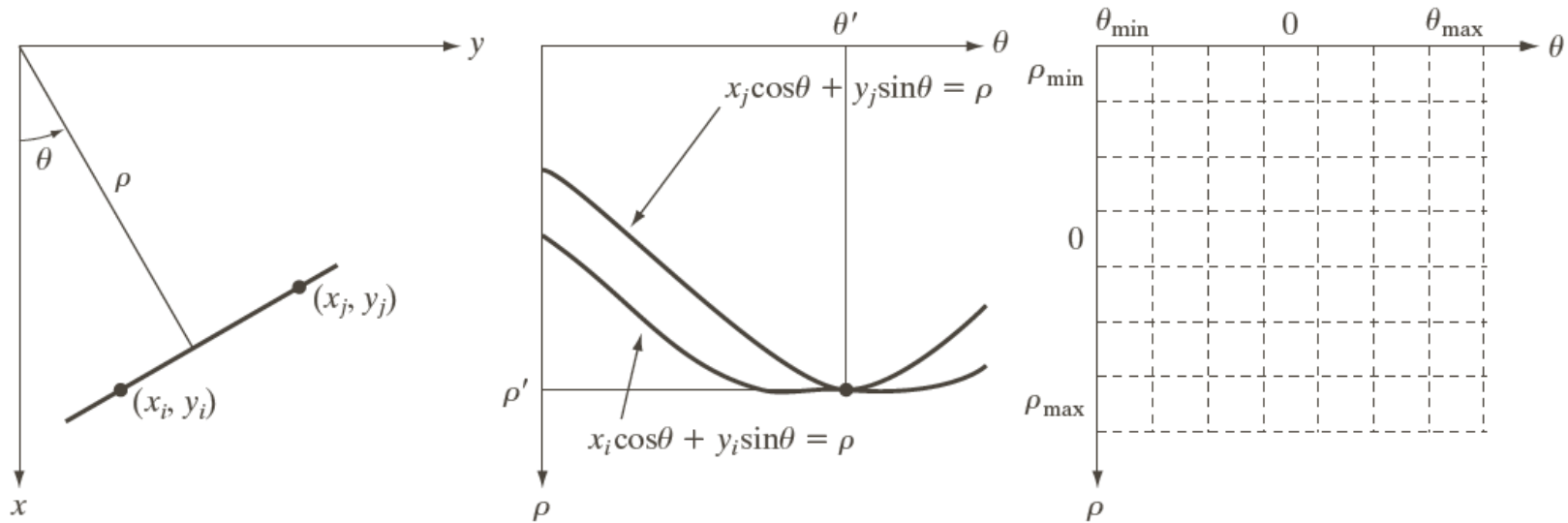
- Para cada ponto, retas são traçadas para certos ângulos (linha cheia)
- A linha pontilhada é perpendicular a cada reta e serve para computar o ângulo e o raio (coordenadas polares)
- Faça isso para os três pontos

Transformada de Hough



- Este é o espaço de Hough para os três pontos !!!
- Note que o maior acumulador é aquele para a linha rosa e tem valor 3

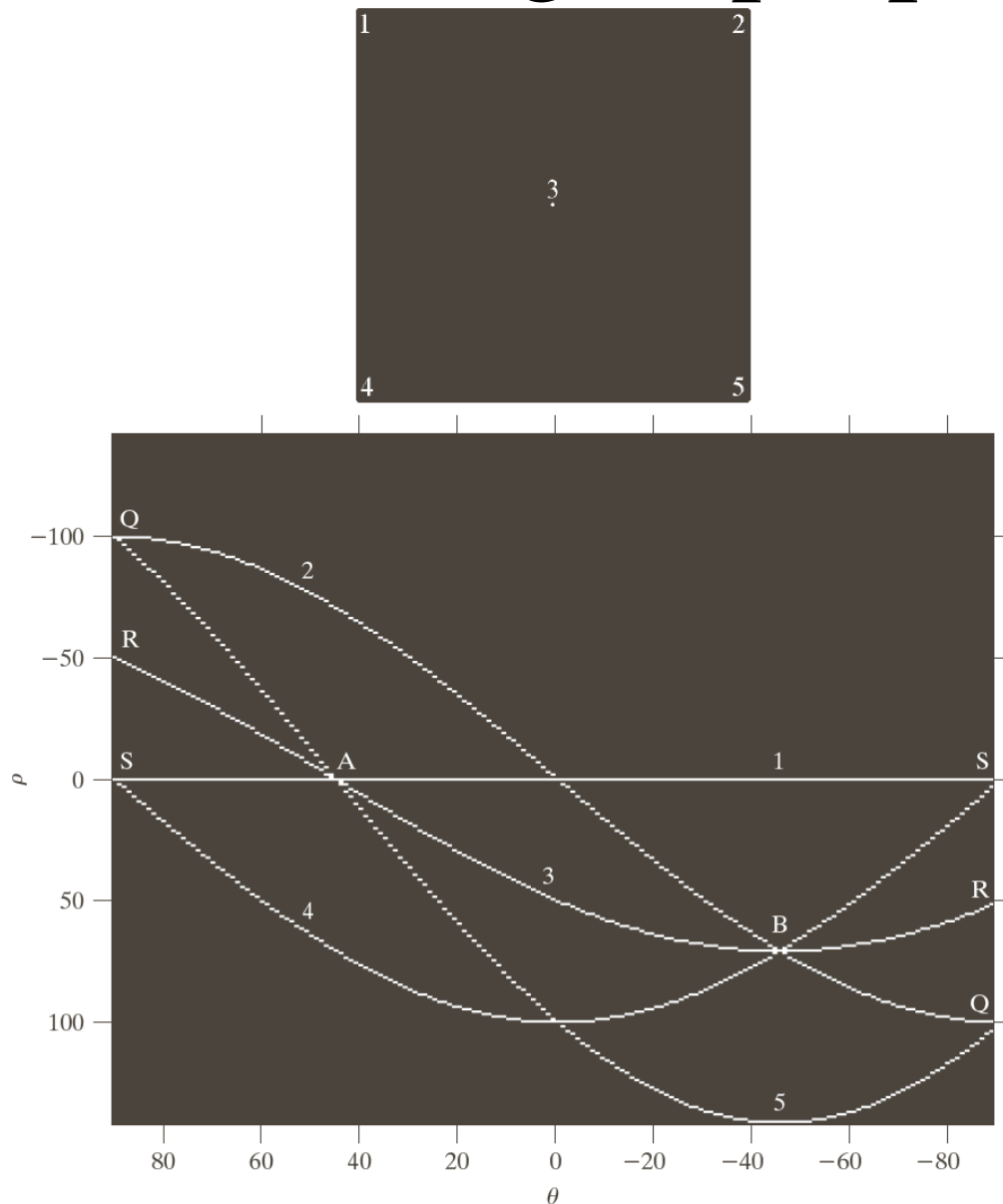
Espaço de Hough



a b c

FIGURE 10.32 (a) (ρ, θ) parameterization of line in the xy -plane. (b) Sinusoidal curves in the $\rho\theta$ -plane; the point of intersection (ρ', θ') corresponds to the line passing through points (x_i, y_i) and (x_j, y_j) in the xy -plane. (c) Division of the $\rho\theta$ -plane into accumulator cells.

Transf. Hough - propiedades



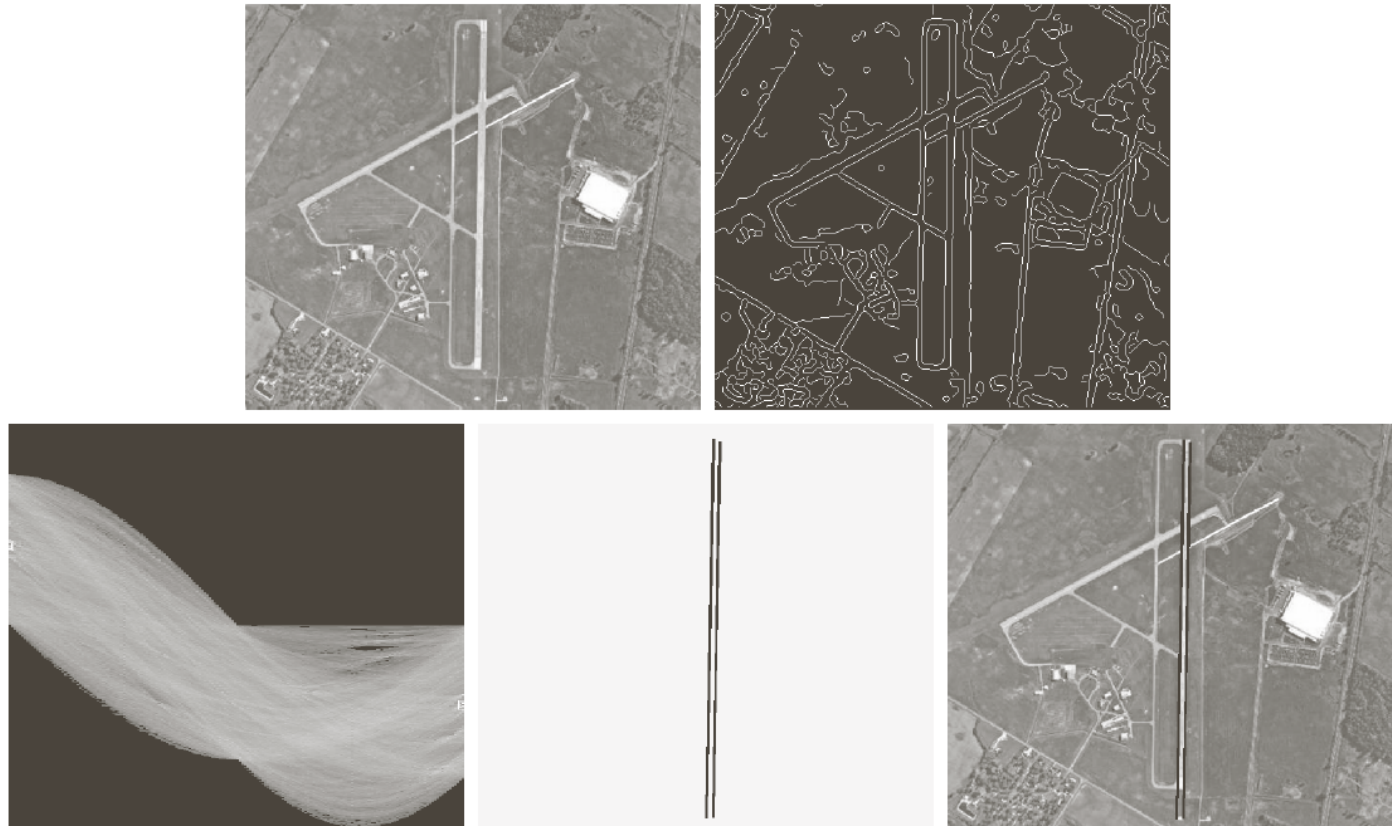
a
b

FIGURE 10.33

(a) Image of size 101×101 pixels, containing five points.

(b) Corresponding parameter space. (The points in (a) were enlarged to make them easier to see.)

Transformada de Hough



a b
c d e

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

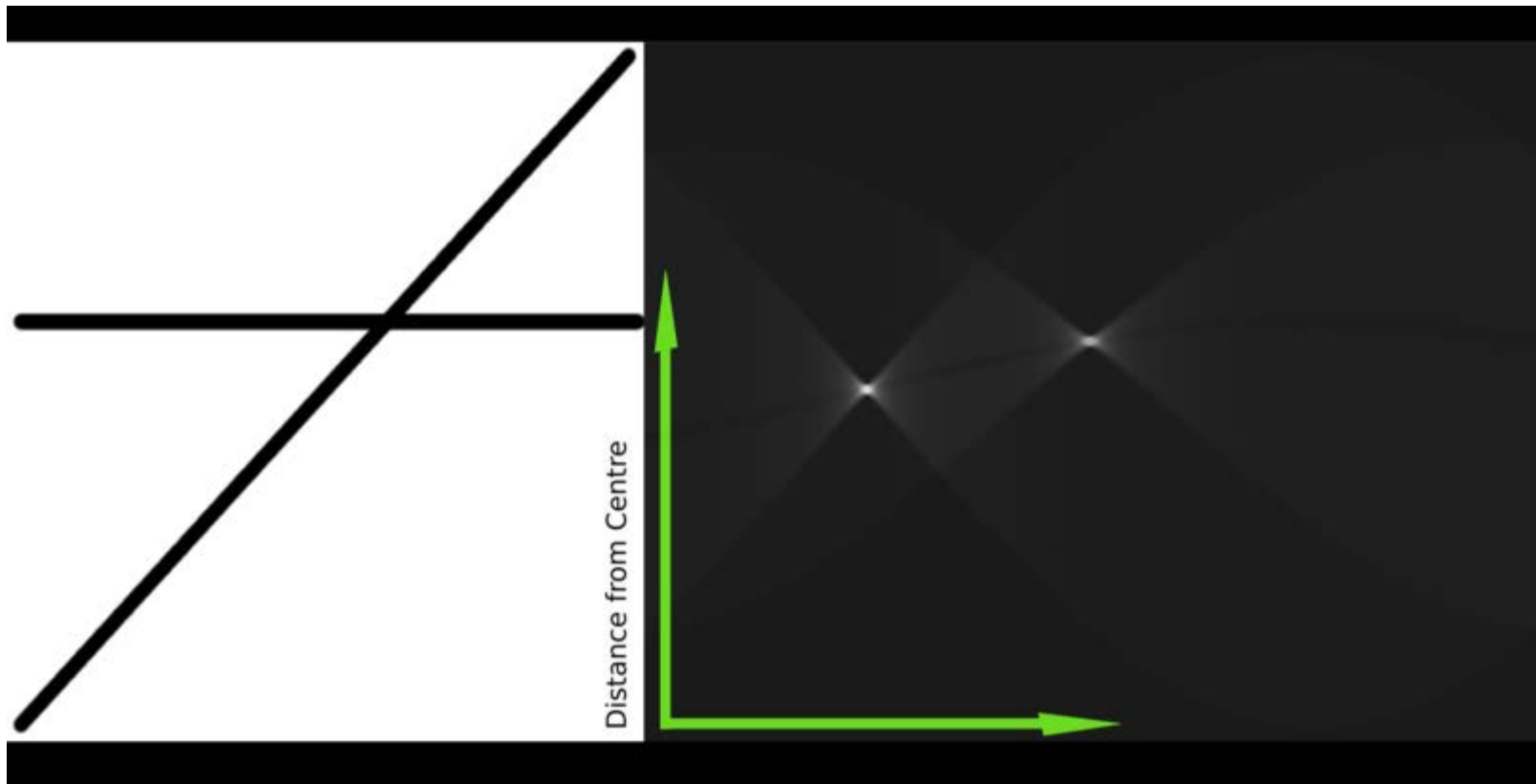
Hough – questões de implementação

- Usando eq. da reta em coordenadas polares
 - $x \cos(\theta) + y \sin(\theta) = \rho$
 - Reta horizontal: $\theta = 0$ e $\rho =$ interseção positiva de x
 - Reta Vertical:
 - $\theta = 90$ e $\rho =$ interseção positiva y
 - $\theta = -90$ e $\rho =$ interseção negativa y
 - Cada curva senoidal no plano de parâmetros representa a família de retas que passam por um ponto (x_k, y_k) no plano xy
 - O ponto de interseção (ρ, θ) corresponde à reta que passa por (x_i, y_i) e (x_j, y_j)

Hough – questões de implementação

- O “espaço de Hough” é uma matriz do tipo
 - $(\rho_{\min}, \rho_{\max})$ e $(\theta_{\min}, \theta_{\max})$
 - $-90^\circ < \theta < +90^\circ$
 - $-D < \rho < +D$, onde D é a distância máxima entre os cantos opostos de uma imagem
 - O eixo θ varia entre $\pm 90^\circ$
 - O eixo ρ varia entre $\pm \sqrt{2} \max(\text{image.h}, \text{image.w})$
 - Os valores da matriz são inicializados com zero
 - Para cada valor (x_i, y_i) em xy , fazemos θ igual a cada valor da subdivisão permitido no eixo θ e calculamos ρ pela equação polar... Arredondamos o valor de ρ e fazemos $A[p, q] = A[p, q] + 1$

Transformada Hough



- Imagem real com duas linhas se cruzando e o espaço de Hough com pontos mais claros indicando as coordenadas polares exatamente destas duas linhas !

Processamento de Imagens

Segmentação por regiões

Segmentação Baseada em Região

- Métodos anteriores: achar bordas entre regiões
- Outra abordagem: encontrar as regiões diretamente
 - Difícil detectar bordas na presença de ruído
- Conceito fundamental: *homogeneidade*
 - principal critério de segmentação

Inúmeros Métodos

Clusterização

Baseados em histograma

Detecção de bordas

Crescimento de regiões

Level Set

Particionamento de grafos

Watershed

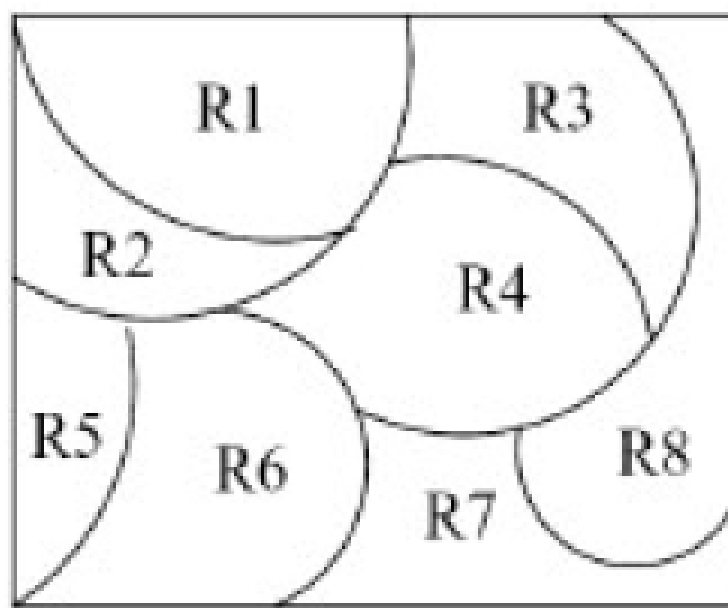
Baseados em modelos

Multi-escala

Redes neurais

Segmentação por Região

Idéia: particionar uma imagem em sub regiões homogêneas $R_1, R_2, R_3, \dots R_n$



Homogeneidade

Cr terios de homogeneidade

N vel de cinza

Textura

Cor

Forma (*shape*)

Modelo (sem ntico): por exemplo um modelo de Markov (livro Sonka et al. 8.6)

Propriedades

$$R_1 \cup R_2 \cup \dots \cup R_n = R$$

R_i é uma região conexa, $i = 1, 2, \dots, n$

$R_i \cap R_j = \text{vazio}$ para todo i e j , $i \neq j$

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

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

($P(R_i)$ é um predicado lógico definido sobre todos os pontos do conjunto R_i)

$P(R_i)$: critério binário de avaliação de homogeneidade. Por exemplo: intensidades dos pixels em R_i no intervalo $[100, 140]$

Métodos de Crescimento de Regiões

Agregação de pixels

Fusão (*merging*)

Divisão (*splitting*)

Divisão e Fusão (*split-and-merge*)

Agregação de Pixels

Conjunto de sementes (*seeds*)

Anexar a cada semente os vizinhos

homogêneos (nível cinza, textura, cor, etc.)

Qual critério de anexação?

Valor absoluto da diferença entre a semente e o pixel vizinho < *threshold*, por exemplo

Conectividade (4-conectado ou 8-conectado). Ou seja, a região final é sempre conectada...

Agregação de pixels

Procedimento começa com um conjunto de pixels (sementes) sendo que novos pixels vizinhos com propriedades similares são adicionados. O objetivo é gerar regiões conectadas e uniformes a partir de cada semente

Um pixel é adicionado a uma região se:

- Ele não tiver sido designado para outra região
- Ele for vizinho desta região (i.e., conectado a um pixel dela)
- A nova região criada pela adição do pixel continuar uniforme

Algoritmo

Seja f uma imagem, e R_1, R_2, \dots, R_n um conjunto de regiões, cada qual com um único pixel semente de nível de cinza $seed_i$.

Repita

for ($i = 1 \dots n$)

for (cada pixel p na borda de R_i)

for (todos os vizinhos de p)

 Seja x, y a coordenada do vizinho

 Se vizinho não rotulado e $|f(x, y) - seed_i| \leq D$

 Adicione vizinho a R_i

Até que mais nenhum pixel seja designado para regiões

Para $D = 3$, $n = 2$, N_8

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

Imagem original

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

Primeira iteração

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

Final

Fusão de regiões (*Region Merging*)

Considere de início

Imagem original (*raw image*) em que cada pixel representa uma região

Algoritmo

- 1- Defina um método inicial capaz de segmentar a imagem em pequenas regiões, segundo um critério $P(R_i)$.
- 2 - Defina um critério de agregação de 2 regiões adjacentes.
- 3 - Mescle todas as regiões adjacentes que satisfaçam o critério. Quando não for mais possível mesclar, pare.

Exemplo

Critérios

1. As sementes são todos os pixels com nível de cinza 255
2. O valor absoluto da diferença de nível de cinza entre qualquer pixel e a semente deve ser menor do que 65
3. O pixel deve ser 8-conectado com pelo menos um pixel da região

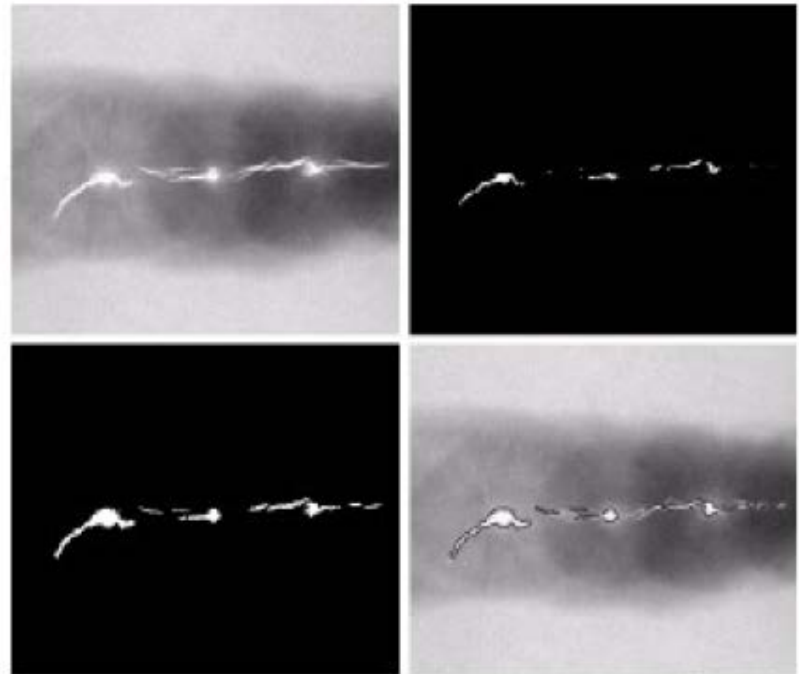


Imagem Raio-X usada na inspeção de solda; pontos brancos indicam quebras e porosidade

Sementes e Homogeneidade

Como escolhemos a(s) semente(s), na prática

Depende da natureza do problema

Se alvos precisam ser detectados, por exemplo, em imagens de infravermelho, escolha os pixels mais claros (objetos mais quentes)

Sem um conhecimento *a priori*, calcule o histograma e escolha os valores de níveis de cinza dos picos mais fortes

Como escolhemos o critério de similaridade (predicado)?

Depende da natureza aplicação

Se imagens coloridas estão disponíveis, pode-se usar a informação da cor

Outras informações importantes: textura, forma e tamanho das regiões

Divisão - *Region Splitting*

Oposto de *region merging*

Imagem representa uma única região

não satisfaz critério $P(R_i)$

objetivo: separar a imagem em regiões que
satisfazam o critério $P(R_1 \cup R_2) = \text{False}$

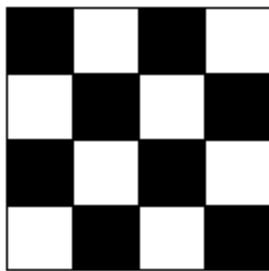
Split / Merge: s/ dualidade (segmentação \neq)

Exemplo: um tabuleiro de xadrez

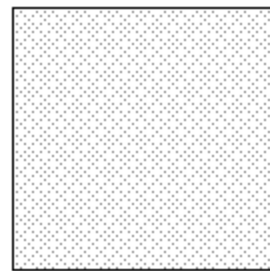
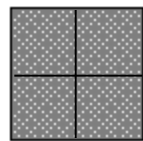
Divisão - *Region Splitting*

Split / Merge: s/ dualidade (segmentação \neq)

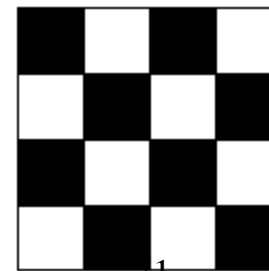
Cr terio de homogeneidade:
vari ncia da m dia dos NC
nos quadrantes atual e de
n vel mais baixo da
pir mide.



(a)



(b)



(c)

Divisão e Fusão (split-and-merge)

Combinação de ambos os métodos

Representação hierárquica das imagens: estrutura de dados *quadtree*

Regiões têm formato quadrado

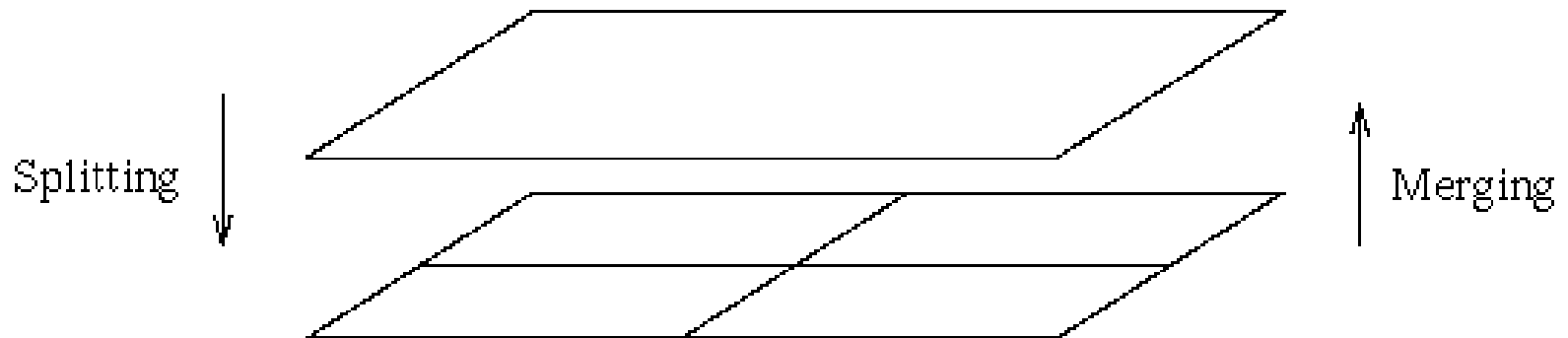
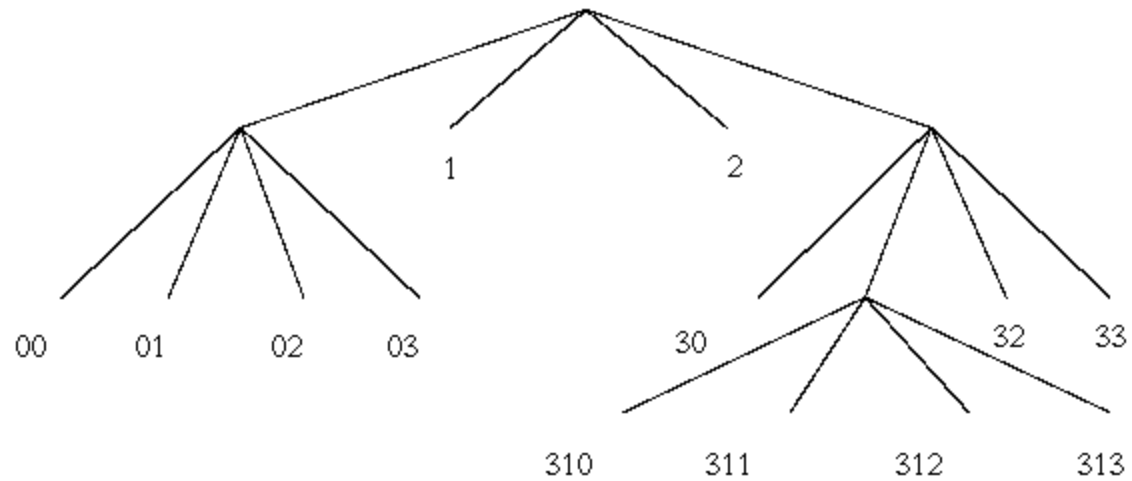
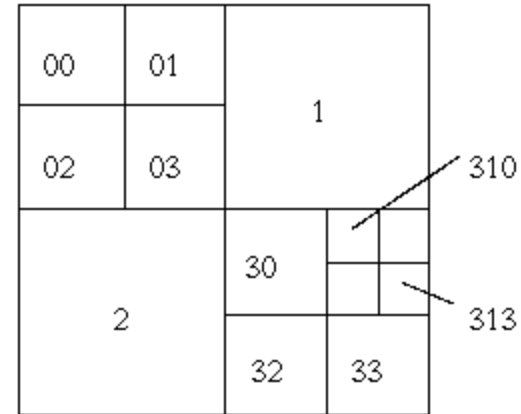
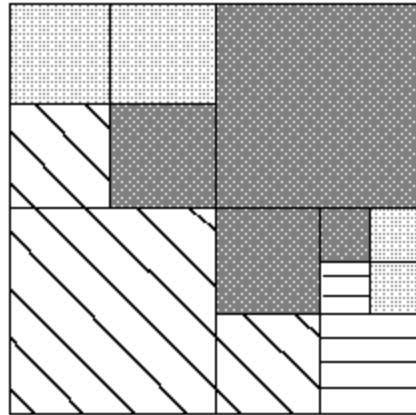


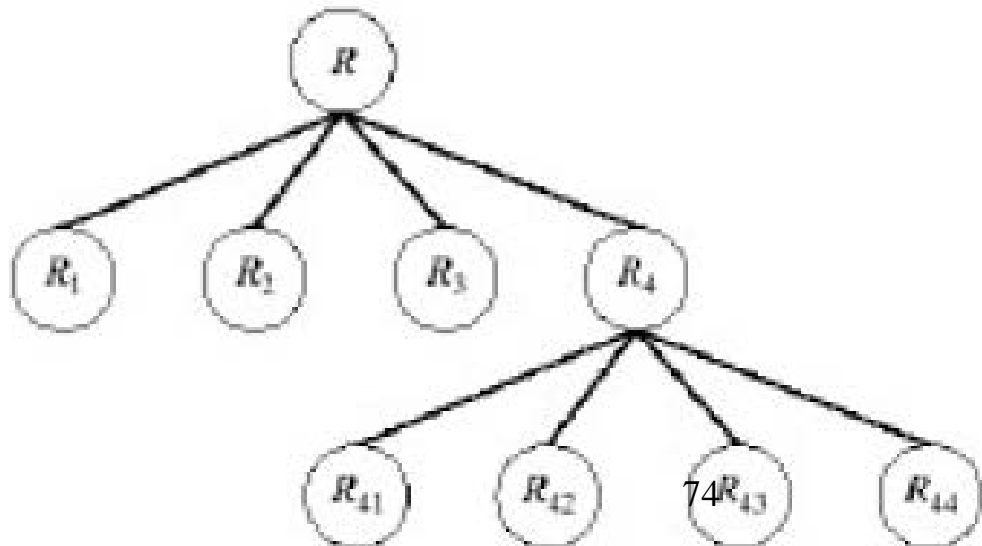
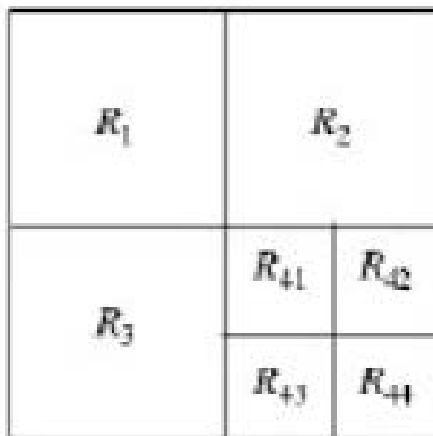
Figure 5.44 *Split-and-merge in a hierarchical data structure.*

Estrutura *Quadtree*

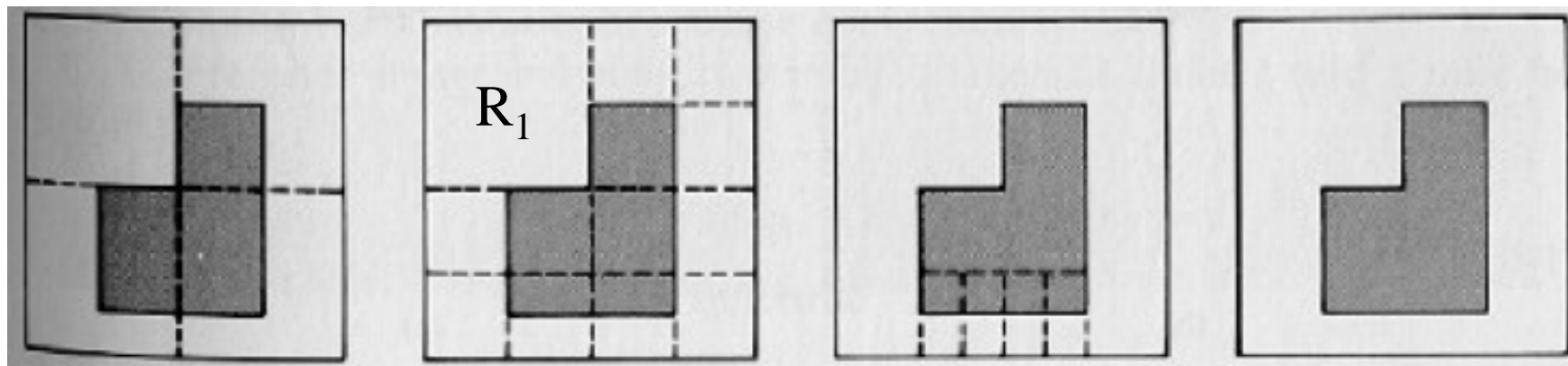


Algoritmo

- Divida a imagem em um conjunto inicial de regiões
- Realize a subdivisão e/ou fusão destas regiões novamente até que as regiões resultantes satisfaçam determinadas propriedades (critério de homogeneidade)
- Procedimento de subdivisão
 1. Para qualquer região R_i , se $P(R_i) = \text{Falso}$, divida R_i em quatro quadrantes
 2. Agregue quaisquer regiões adjacentes R_j e R_k tal que $P(R_j \cup R_k) = \text{Verdadeiro}$
 3. Pare quando nenhuma divisão ou fusão adicional for possível



Exemplo 1



$P(R) = \text{Falso}$

$R_1 = \text{ok}$

$R = \text{imagem inteira}$

Exemplo 2



a) Imagem Original



b) Split/Merge



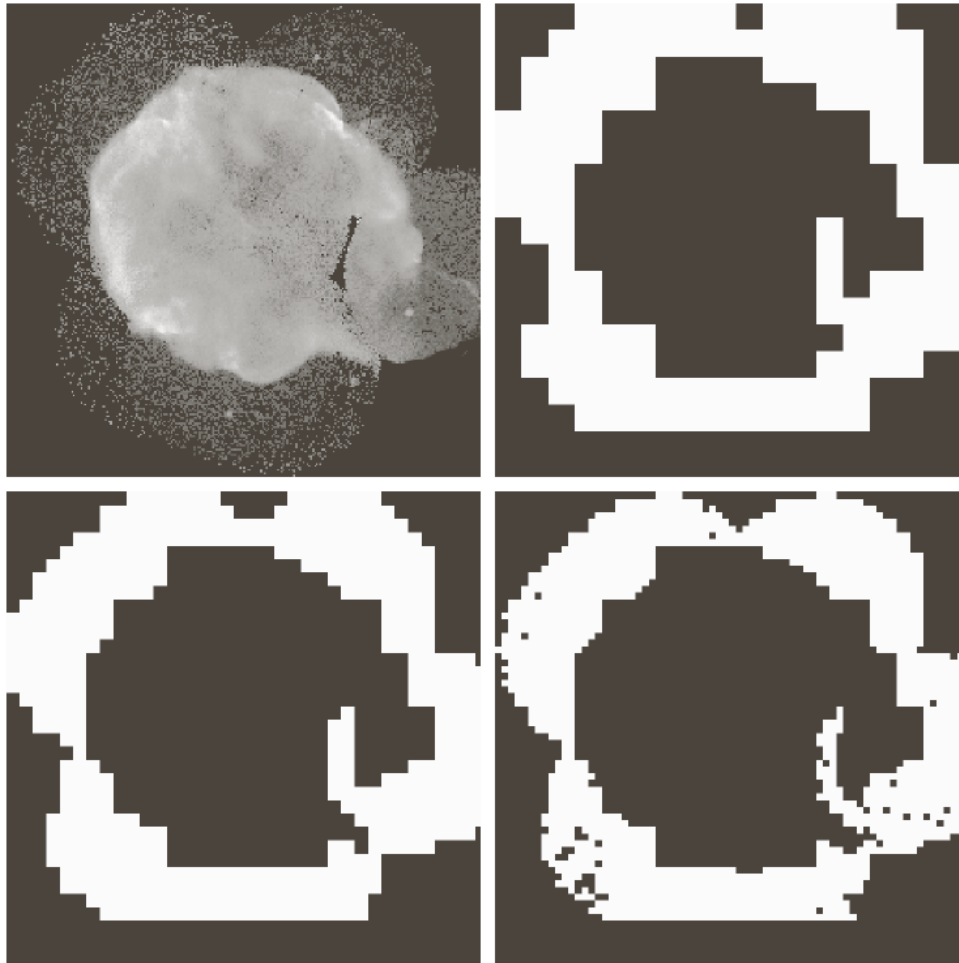
c) Limiarização de (b)

Predicado usado: $P(R_i) = \textit{Verdadeiro se } |z_i - \mu_i| \leq 2\sigma_i$
para 80% dos pixels em R_i

(μ_i, σ_i são a média e o desvio padrão dos pixels em R_i)

Se $P(R_i)$, cada nível de cinza é ajustado para μ_i

Exemplo 3



a	b
c	d

FIGURE 10.53

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

Bibliografia

- Sonka, Cap. 5
- Gonzalez & Woods, Cap. 10