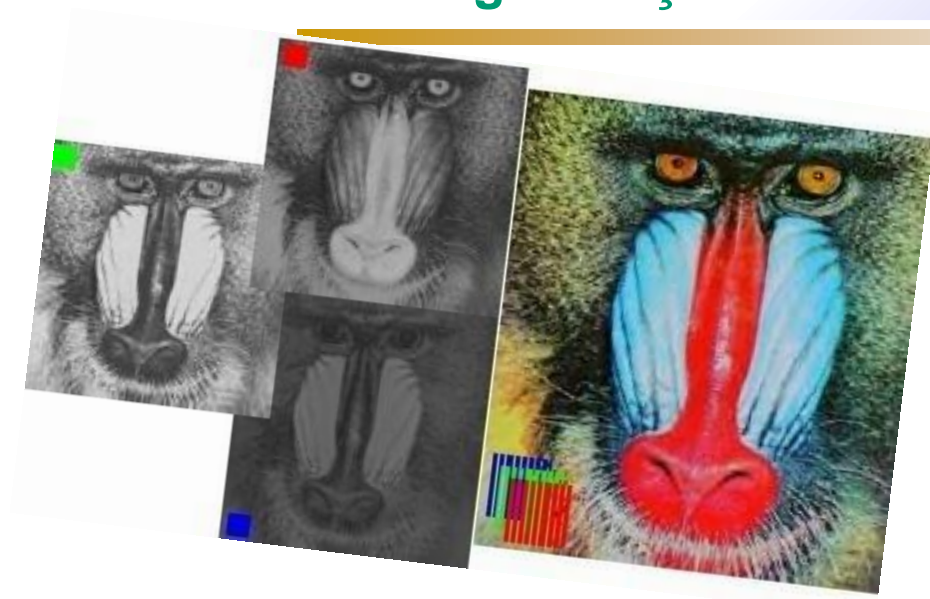


Prof. Dr. Leandro Alves Neves

Pós-graduação em Ciência da Computação



Aula 09

Processamento de Imagens
Digitais

Sumário

- **Segmentação de Imagens**
 - **Fundamentos**

- **Detecção por Descontinuidades**
 - **Segmentação de Pontos e Retas**
 - **Bordas (Operadores de Gradiente)**
 - Roberts, Prewitt, Sobel, Kirsch, Robinson, Frei-Chen e Método de Canny

- **Limiarização**
 - **limiar global, múltiplos limiares e Limiar local**

- **Detecção por Similaridades**
 - **Crescimento de regiões**
 - **Divisão de regiões**
 - **Divisão e fusão de regiões**

Segmentação de Imagens

- Processo que **visa particionar o conjunto** de dados de entrada
 - **Objetivo:** obter **conteúdo semântico relevante**
 - **Justificativa:** **facilitar a interpretação dos dados** contidos em imagens digitais
 - **Metodologias:**
 - Técnicas utilizam-se das **propriedades geométricas e topológicas** dos objetos

Segmentação de Imagens

■ Processo de Segmentação

□ Tarefa difícil

■ Dependente da extração de características dos objetos

- Problemas adicionais em imagens ruidosas

□ Explora propriedades dos níveis de cinza

■ Técnicas baseadas em Descontinuidades:

- Consideram alterações abruptas nos níveis de cinza
 - Bordas

■ Técnicas baseadas em Similaridades:

- Agrupam pontos da imagem com valores similares (regiões)

Segmentação de Imagens

■ Detecção de Descontinuidades

- Tipos básicos de descontinuidades: **pontos, segmentos de retas, junções e bordas**
- Processo adotado: **Convolução**

$W =$

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

Figura: Máscara de 3×3 pixels.

- A resposta R da máscara posicionada sobre um ponto da imagem é dada por:

$$R = w_1z_1 + w_2z_2 + \dots + w_9z_9 = \sum_{i=1}^9 w_i z_i$$

- z_i : nível de cinza associado com o coeficiente w_i

Segmentação de Imagens

■ Detecção de Pontos

- Uso da máscara $\longrightarrow h =$
 - Diferenças ponderadas entre:
 - Valores do Ponto central e de seus vizinhos

-1	-1	-1
-1	8	-1
-1	-1	-1

- **Ponto detectado** na região central **se** $|R| > T$
 - T é um limiar não-negativo
 - R é o resultado obtido da convolução
- **Região homogênea (Máscara é nula, $R=0$):**
 - Pixels pertencentes à região possuem a mesma intensidade

Segmentação de Imagens

■ Detecção de Retas

□ Resultado obtido a partir da aplicação de máscaras

- Imagem com intensidade de **fundo constante** \Rightarrow **máscara indica reta detectada**

$$h_1 = \begin{bmatrix} -1 & -1 & -1 \\ 2 & 2 & 2 \\ -1 & -1 & -1 \end{bmatrix}$$

Retas \Rightarrow horizontais

$$h_2 = \begin{bmatrix} -1 & -1 & 2 \\ -1 & 2 & -1 \\ 2 & -1 & -1 \end{bmatrix}$$

45°

$$h_3 = \begin{bmatrix} -1 & 2 & -1 \\ -1 & 2 & -1 \\ -1 & 2 & -1 \end{bmatrix}$$

verticais

$$h_4 = \begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix}$$

135°

Segmentação de Imagens

■ Tipos de Segmentação

□ Bordas

- Roberts, Prewitt, Sobel, **Kirsch, Robinson, Frei-Chen**, Laplaciano, Laplaciano do Gaussiano, e Método de Canny

□ Limiarização

- Limiar local, limiar global, múltiplos limiares

□ Regiões

- Crescimento de regiões
- Divisão de regiões
- Divisão e união de regiões ou divisão e fusão de regiões (Split-merge)
- Outros

Segmentação de Imagens

■ Detecção de Bordas

□ Borda

- Limite ou fronteira entre **duas regiões** \Rightarrow propriedades relativamente distintas de nível de cinza

□ Princípio:

- As regiões de **fundo são consideradas homogêneas**

■ Transição entre duas regiões

- Descontinuidade de níveis de cinza

□ Uso de um **operador local diferencial**

Segmentação de Imagens

- Uso de derivadas
- Lembrando que:
 - Derivada parcial de primeira ordem pode ser indicada como: $\frac{\partial f}{\partial x}$
 - Dado um ponto em f , obter a diferença em relação ao próximo ponto
- Na forma discreta, temos que $\frac{\partial f}{\partial x} = f(x+1, y) - f(x, y)$.

Segmentação de Imagens

- Derivada parcial de **segunda ordem**: $\frac{\partial^2 f}{\partial x^2}$
 - **Diferença do próximo somada a diferença do anterior**
- Na forma discreta, temos que:

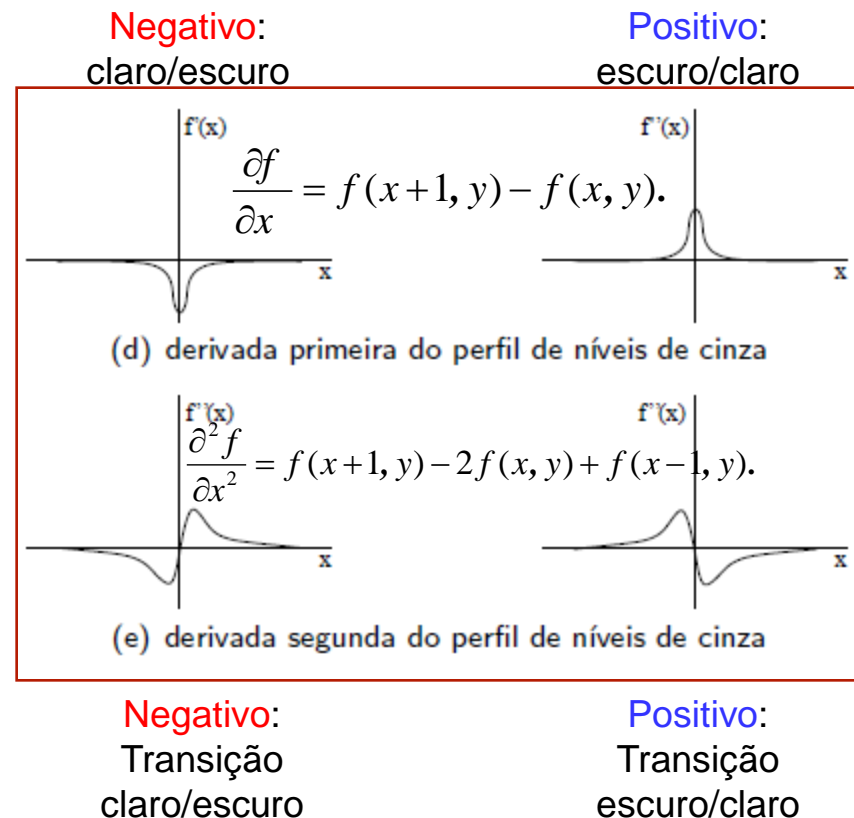
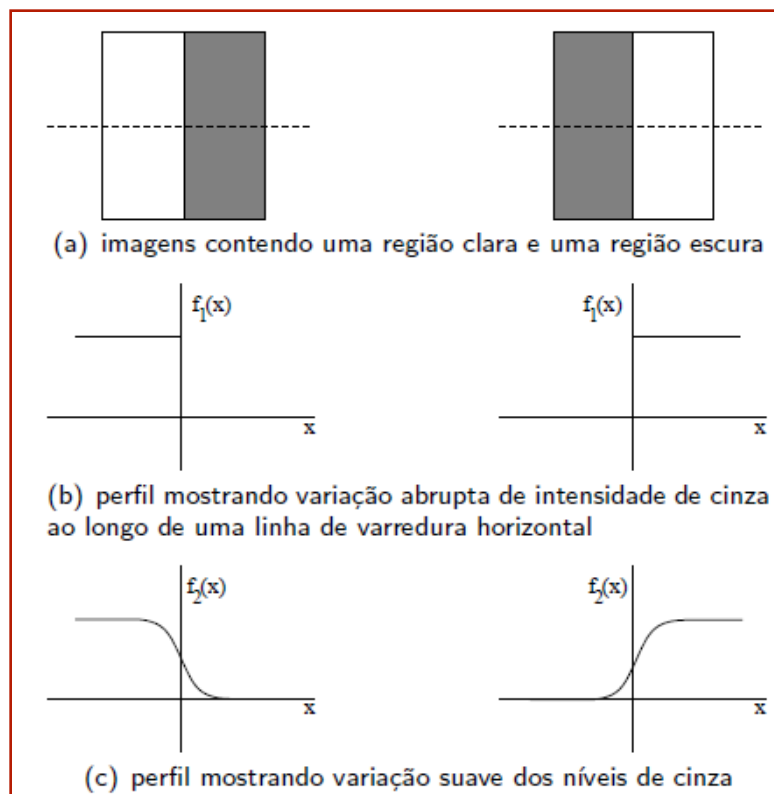
$$\frac{\partial^2 f}{\partial x^2} = f(x+1, y) - f(x, y) + f(x-1, y) - f(x, y).$$

- Logo, $\frac{\partial^2 f}{\partial x^2} = f(x+1, y) - 2f(x, y) + f(x-1, y).$

Segmentação de Imagens

■ Detecção de Bordas

□ Etapas de um **operador local diferencial**



□ **Detecção de qualquer ponto de borda:**

- Derivada Primeira: Magnitude do ponto
- Derivada Segunda: Operador Laplaciano

Operadores de Gradiente

Segmentação de Imagens

- **Detecção de bordas: operador local diferencial**
 - Identificar **mudanças locais significativas**
 - Aplicar Derivadas
- Imagem depende de duas coordenadas espaciais
 - **Logo, Bordas** podem ser expressas por **derivadas parciais**.
- **Gradiente** (∇f): operador comumente utilizado
 - Vetor cuja **direção indica** regiões em que os níveis de cinza sofreram **maior variação (Bordas)**

Segmentação de Imagens

- **Gradiente** na forma matricial: $\nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$
- **Exemplo de interpretação:** *variação rápida* de $f(x, y)$ em uma *direção* e *lenta na outra direção* ➡ pode indicar a presença de *borda*.

Segmentação de Imagens

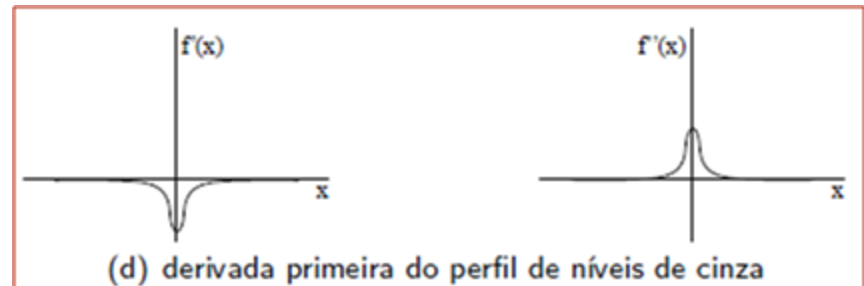
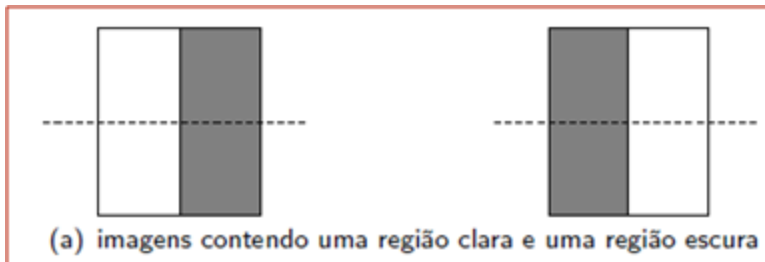
- **Magnitude** do vetor gradiente (∇f)

$$\nabla f = \text{mag}(\nabla f) = \sqrt{G_x^2 + G_y^2} = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$$

- Indica a **maior taxa de variação** de $f(x, y)$ por unidade de distância na direção de ∇f
- **Magnitude do gradiente (exemplo):** aproximada pelo uso dos **valores absolutos**:
 - $\nabla f \approx |G_x| + |G_y|$
 - ou, valor máximo entre os gradientes: $\nabla f \approx \max(|G_x|, |G_y|)$

Segmentação de Imagens

- **Magnitude** do vetor gradiente (∇f)
 - Mudança em intensidade **detectada pela diferença entre os valores de pixels adjacentes**
- **Bordas Verticais:** Diferença horizontal entre os pontos



- **Bordas Horizontais:** Diferença vertical entre os pontos

Segmentação de Imagens

■ Dada uma região de imagem (3x3)

$f(x-1, y-1)$	$f(x, y-1)$	$f(x+1, y-1)$
$f(x-1, y)$	$f(x, y)$	$f(x+1, y)$
$f(x-1, y+1)$	$f(x, y+1)$	$f(x+1, y+1)$

Figura : Região da imagem formada por 3×3 pixels.

- **Aproximações** para obter a magnitude do gradiente no ponto $f(x,y)$

$$\nabla f = \text{mag}(\nabla f) = \sqrt{G_x^2 + G_y^2} = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$$

Segmentação de Imagens

■ Dada uma região de imagem (3x3)

$f(x-1, y-1)$	$f(x, y-1)$	$f(x+1, y-1)$
$f(x-1, y)$	$f(x, y) \leftarrow \rightarrow f(x+1, y)$	$f(x+1, y)$
$f(x-1, y+1)$	$f(x, y+1)$	$f(x+1, y+1)$

Primeira Derivada

Figura : Região da imagem formada por 3×3 pixels.

- **Aproximações**, calcular as diferenças nas direções x e y :

$$\nabla f = \sqrt{[f(x, y) - f(x+1, y)]^2 + [f(x, y) - f(x, y+1)]^2}$$

Segmentação de Imagens

■ Dada uma região de imagem (3x3)

$f(x-1, y-1)$	$f(x, y-1)$	$f(x+1, y-1)$
$f(x-1, y)$	$f(x, y)$	$f(x+1, y)$
$f(x-1, y+1)$	$f(x, y+1)$	$f(x+1, y+1)$

Figura : Região da imagem formada por 3×3 pixels.

- **Aproximações**, calcular as diferenças cruzadas nas direções x e y :

$$\nabla f = \sqrt{[f(x, y) - f(x+1, y+1)]^2 + [f(x, y+1) - f(x+1, y)]^2}$$

Segmentação de Imagens

■ Dada uma região de imagem (3x3)

$f(x-1, y-1)$	$f(x, y-1)$	$f(x+1, y-1)$
$f(x-1, y)$	$f(x, y)$	$f(x+1, y)$
$f(x-1, y+1)$	$f(x, y+1)$	$f(x+1, y+1)$

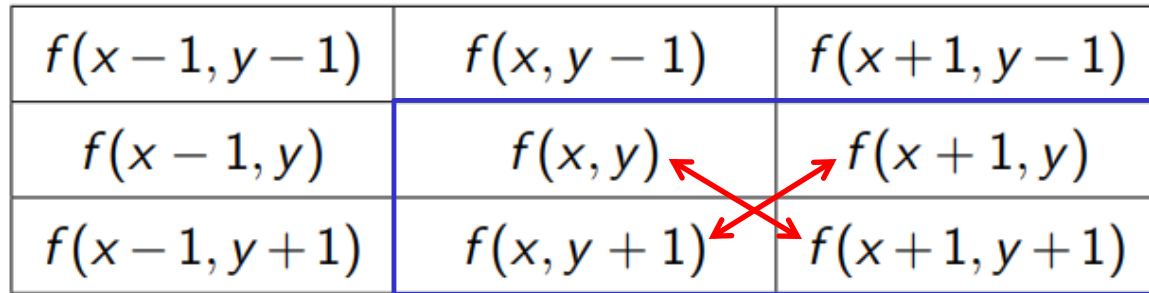


Figura : Região da imagem formada por 3×3 pixels.

- **Aproximações**, calcular os **valores absolutos**:

$$\nabla f \approx |f(x, y) - f(x+1, y+1)| + |f(x, y+1) - f(x+1, y)|.$$

Segmentação de Imagens

■ Dada uma região de imagem (3x3)

$f(x-1, y-1)$	$f(x, y-1)$	$f(x+1, y-1)$
$f(x-1, y)$	$f(x, y)$	$f(x+1, y)$
$f(x-1, y+1)$	$f(x, y+1)$	$f(x+1, y+1)$

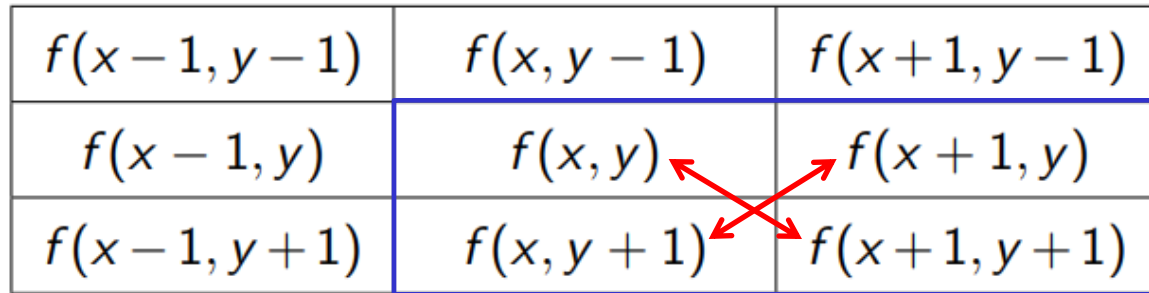


Figura : Região da imagem formada por 3×3 pixels.

- **Magnitude do gradiente:** pode ser aproximada pelo uso das **diferenças cruzadas** (**Vizinhança 2x2**):

- $\nabla f \approx |G_x| + |G_y|,$

Gradiente de Roberts ou Operadores de Roberts

$$\nabla f \approx |f(x, y) - f(x+1, y+1)| + |f(x, y+1) - f(x+1, y)|.$$

Segmentação de Imagens

- **Operador de Roberts (vizinhança 2x2)**
 - Calcula-se o gradiente das diferenças cruzadas e soma os resultados
 - As máscaras são:

$$G_x = \begin{array}{|c|c|} \hline 1 & 0 \\ \hline 0 & -1 \\ \hline \end{array}$$

$$G_y = \begin{array}{|c|c|} \hline 0 & -1 \\ \hline 1 & 0 \\ \hline \end{array}$$

Segmentação de Imagens

- ❑ Magnitude do gradiente: Outras aproximações

Segmentação de Imagens

■ Operador de Prewitt (vizinhança 3x3)

- Diferença entre a primeira e terceira colunas (\approx derivada direção x)
- Diferença entre a terceira e primeira linhas (\approx derivada direção y)

$$\nabla f \approx |G_x| + |G_y|,$$

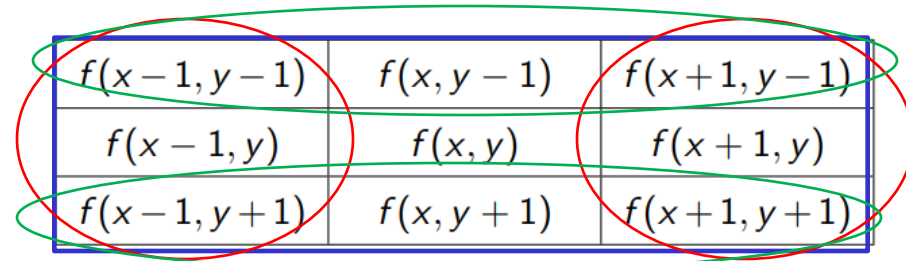
$$\nabla f \approx [f(x+1, y-1) + f(x+1, y) + f(x+1, y+1)] -$$

$$[f(x-1, y-1) + f(x-1, y) + f(x-1, y+1)]$$

+

$$[f(x-1, y+1) + f(x, y+1) + f(x+1, y+1)] -$$

$$[f(x-1, y-1) + f(x, y-1) + f(x+1, y-1)]$$



$f(x-1, y-1)$	$f(x, y-1)$	$f(x+1, y-1)$
$f(x-1, y)$	$f(x, y)$	$f(x+1, y)$
$f(x-1, y+1)$	$f(x, y+1)$	$f(x+1, y+1)$

Figura : Região da imagem formada por 3×3 pixels.

- As máscaras são:

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$

$$G_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

Segmentação de Imagens

■ Operador de Sobel (vizinhança 3x3)

- Diferença de valores ponderados, pesos maiores para realçar vizinhança 4 (Baseado em Prewitt):

$$\nabla f \approx |G_x| + |G_y|,$$

$$\nabla f \approx [f(x+1, y-1) + 2f(x+1, y) + f(x+1, y+1)] -$$

$$[f(x-1, y-1) + 2f(x-1, y) + f(x-1, y+1)] +$$

$$[f(x-1, y+1) + 2f(x, y+1) + f(x+1, y+1)] -$$

$$[f(x-1, y-1) + 2f(x, y-1) + f(x+1, y-1)]$$

$f(x-1, y-1)$	$f(x, y-1)$	$f(x+1, y-1)$
$f(x-1, y)$	$f(x, y)$	$f(x+1, y)$
$f(x-1, y+1)$	$f(x, y+1)$	$f(x+1, y+1)$

Figura : Região da imagem formada por 3×3 pixels.

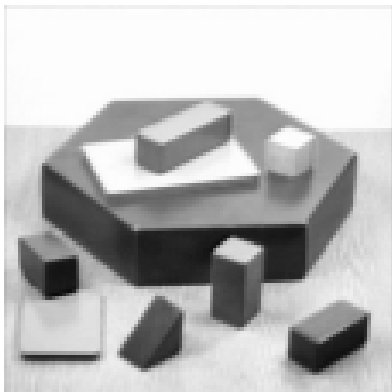
- As máscaras são:

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

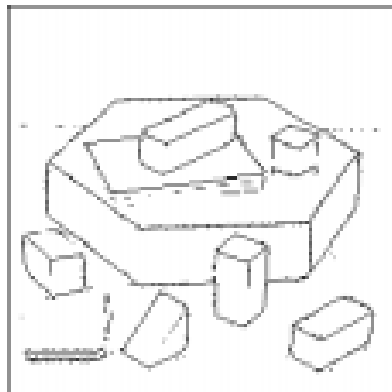
$$G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

Segmentação de Imagens

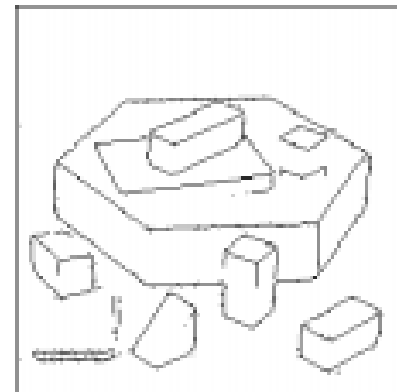
■ Exemplos



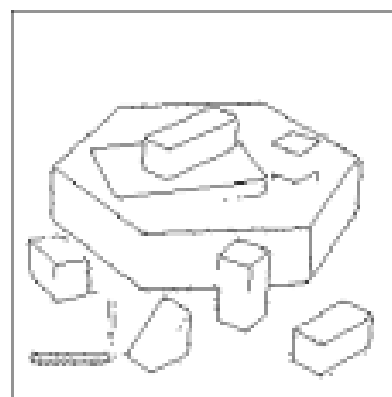
(a) imagem original



(b) Roberts



(c) Prewitt



(d) Sobel

Segmentação de Imagens

- ❑ **Operador de Kirsch (1971):** Utiliza-se dos princípios de Sobel e Prewitt:

- Oito máscaras de convolução orientadas em 45°.

5	-3	-3	-3	-3	-3	-3	-3	-3	-3	-3	-3
5	0	-3	5	0	-3	-3	0	-3	-3	0	5
5	-3	-3	5	5	-3	5	5	5	-3	5	5
-3	-3	5	-3	5	5	5	5	5	5	5	-3
-3	0	5	-3	0	5	-3	0	-3	5	0	-3
-3	-3	5	-3	-3	-3	-3	-3	-3	-3	-3	-3

- ❑ Para cada pixel da imagem:
 - Aplicar cada máscara ➡ **manter o valor máximo**
 - ❑ Logo, o **cálculo da magnitude** do gradiente **não determina valores separados** para G_x e G_y (Sobel e Prewitt)

O gradiente de cada pixel é obtido pela maior resposta do conjunto de oito máscaras

Segmentação de Imagens

- ❑ **Operador de Robinson (1977):** Utiliza-se dos princípios de **Kirsch**, também com oito máscaras

1	0	-1	0	-1	-2	-1	-2	-1	0
2	0	-2	1	0	-1	0	0	0	-1
1	0	-1	2	1	0	1	2	1	0
-1	0	1	0	1	2	1	2	1	0
-2	0	2	-1	0	1	0	0	0	1
-1	0	1	-2	-1	0	-1	-2	-1	-2

- ❑ Para **cada pixel da imagem**:
 - Aplicar cada máscara ➡ **manter o valor máximo**

O gradiente de cada pixel é obtido pela maior resposta do conjunto de oito máscaras

Segmentação de Imagens

❑ Operador de Frei-Chen (1977):

$M_1 =$	<table><tr><td>1</td><td>$\sqrt{2}$</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>-1</td><td>$-\sqrt{2}$</td><td>-1</td></tr></table>	1	$\sqrt{2}$	1	0	0	0	-1	$-\sqrt{2}$	-1	$M_2 =$	<table><tr><td>1</td><td>0</td><td>-1</td></tr><tr><td>$\sqrt{2}$</td><td>0</td><td>$-\sqrt{2}$</td></tr><tr><td>1</td><td>0</td><td>-1</td></tr></table>	1	0	-1	$\sqrt{2}$	0	$-\sqrt{2}$	1	0	-1	$M_3 =$	<table><tr><td>0</td><td>-1</td><td>$\sqrt{2}$</td></tr><tr><td>1</td><td>0</td><td>-1</td></tr><tr><td>$-\sqrt{2}$</td><td>1</td><td>0</td></tr></table>	0	-1	$\sqrt{2}$	1	0	-1	$-\sqrt{2}$	1	0
1	$\sqrt{2}$	1																														
0	0	0																														
-1	$-\sqrt{2}$	-1																														
1	0	-1																														
$\sqrt{2}$	0	$-\sqrt{2}$																														
1	0	-1																														
0	-1	$\sqrt{2}$																														
1	0	-1																														
$-\sqrt{2}$	1	0																														
$M_4 =$	<table><tr><td>$\sqrt{2}$</td><td>-1</td><td>0</td></tr><tr><td>-1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>$-\sqrt{2}$</td></tr></table>	$\sqrt{2}$	-1	0	-1	0	1	0	1	$-\sqrt{2}$	$M_5 =$	<table><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>-1</td><td>0</td><td>-1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr></table>	0	1	0	-1	0	-1	0	1	0	$M_6 =$	<table><tr><td>-1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>0</td><td>-1</td></tr></table>	-1	0	1	0	0	0	1	0	-1
$\sqrt{2}$	-1	0																														
-1	0	1																														
0	1	$-\sqrt{2}$																														
0	1	0																														
-1	0	-1																														
0	1	0																														
-1	0	1																														
0	0	0																														
1	0	-1																														
$M_7 =$	<table><tr><td>1</td><td>-2</td><td>1</td></tr><tr><td>-2</td><td>4</td><td>-2</td></tr><tr><td>1</td><td>-2</td><td>1</td></tr></table>	1	-2	1	-2	4	-2	1	-2	1	$M_8 =$	<table><tr><td>-2</td><td>1</td><td>-2</td></tr><tr><td>1</td><td>4</td><td>1</td></tr><tr><td>-2</td><td>1</td><td>-2</td></tr></table>	-2	1	-2	1	4	1	-2	1	-2	$M_9 =$	<table><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	1	1	1	1	1	1	1	1	1
1	-2	1																														
-2	4	-2																														
1	-2	1																														
-2	1	-2																														
1	4	1																														
-2	1	-2																														
1	1	1																														
1	1	1																														
1	1	1																														

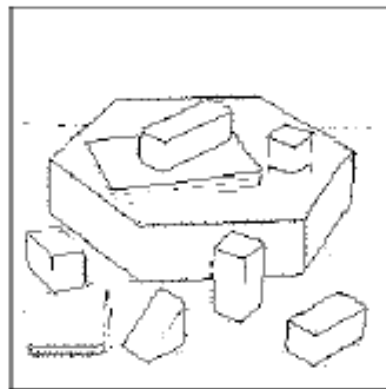
- M_1 a M_4 : detectar bordas
- M_5 a M_8 : detectar retas
- M_9 : média dos pixels na região de 3×3 pixels

Segmentação de Imagens

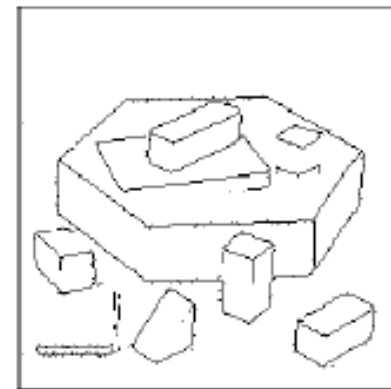
■ Detecção de Bordas: Operadores locais diferenciais



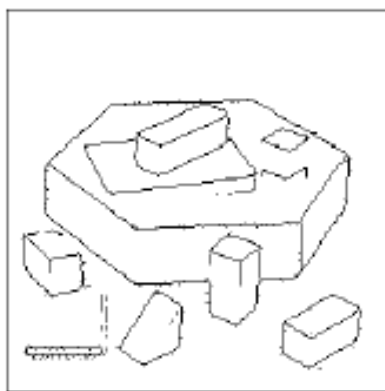
(a) imagem original



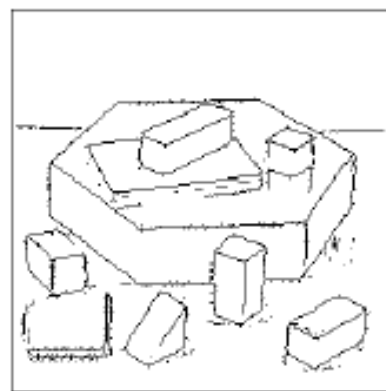
(b) Roberts



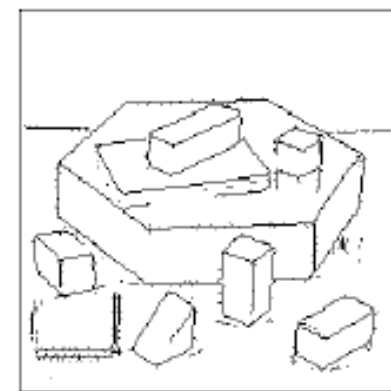
(c) Prewitt



(d) Sobel



(e) Kirsch



(f) Frei-Chen

Segmentação de Imagens

- **Detecção de Bordas: Operador de Canny**
- **Reconhecido como um método completo**
 - Princípios
 - Ruídos não devem gerar bordas
 - Bordas detectadas: mais próximas das bordas reais
 - Cada borda real deve ser representada por um único ponto

Segmentação de Imagens

- **Detecção de Bordas: Operador de Canny**
- **Algoritmo pode ser descrito em cinco fases:**
 1. Filtragem Gaussiana
 - Objetivo é a redução de ruídos
 2. Detecção da intensidade de gradientes
 - Detectar a magnitude e a direção das bordas
 - Aplicar operadores de Sobel ou Prewitt, por exemplo

Segmentação de Imagens

- **Detecção de Bordas: Operador de Canny**
- **Algoritmo pode ser descrito em cinco fases:**
 - 3. Supressão de pixels não máximos: **Afinamento das bordas**
 - Comparações de pixels da borda com seus vizinhos para identificar a magnitude do gradiente
 - Exemplo: bordas verticais
 - Pixel da borda é preservado se sua magnitude for maior do que as magnitudes dos vizinhos da esquerda e da direita

Segmentação de Imagens

- **Detecção de Bordas: Operador de Canny**

- **Algoritmo pode ser descrito em cinco fases:**

- 4. Duplo limiar (**ponto crítico**)

- Definição de limiares com base em uma estimativa da **relação sinal-ruído**
 - Pixel da borda é comparado com dois limiares, L_1 e L_2 ($L_1 < L_2$)
 - **Borda fraca (pixel rejeitado):** pixel menor que L_1
 - **Borda forte (pixel aceito):** pixel maior que L_2
 - **Reavaliar:** Aplicar etapa cinco (rastreamento)

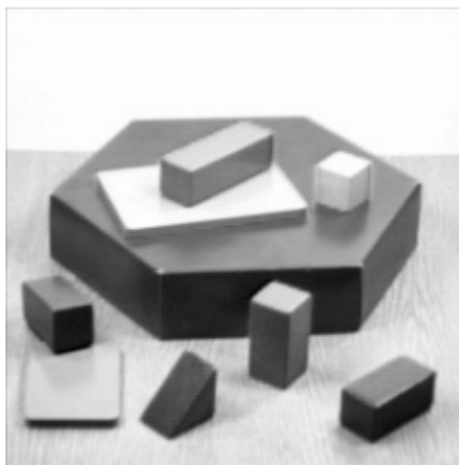
Segmentação de Imagens

- **Detecção de Bordas: Operador de Canny**
- **Algoritmo pode ser descrito em cinco fases:**
 - 5. Rastreamento de bordas
 - Pixel de borda no intervalo: $L_1 < \text{pixel} < L_2$
 - Pixel aceito se o mesmo define um percurso
 - Um pixel conectado a um pixel de borda é considerado como pertencente à borda se a magnitude de seu gradiente estiver acima de L_1

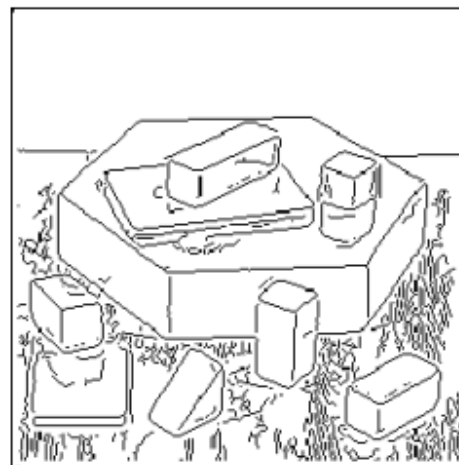
Segmentação de Imagens

■ Detecção de Bordas: Operador de Canny

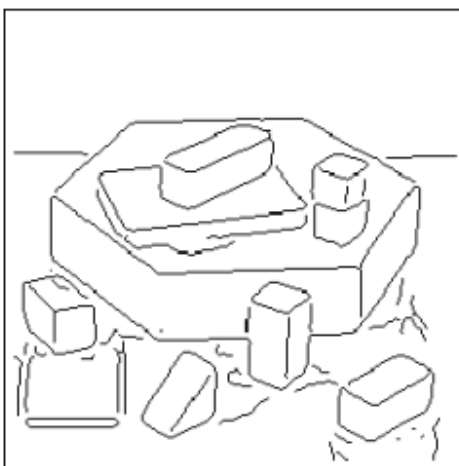
■ Exemplo



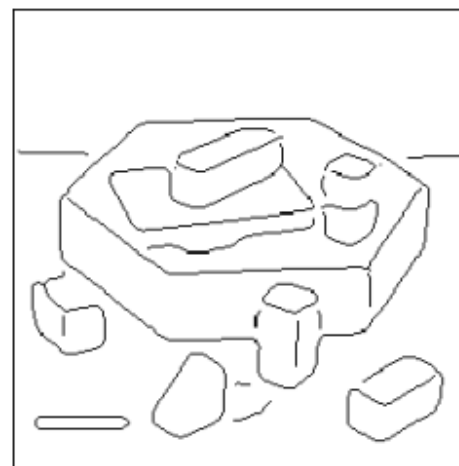
(a) imagem original



(b) $\sigma = 0.5$



(c) $\sigma = 1.0$

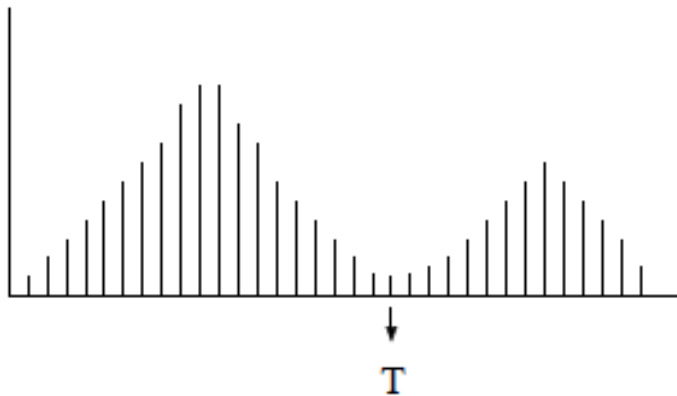


(d) $\sigma = 2.0$

Segmentação de Imagens

■ Limiarização

- Classificação dos pixels de uma imagem de acordo com a especificação de um ou mais limiares
- Dado o histograma de uma imagem $f(x,y)$

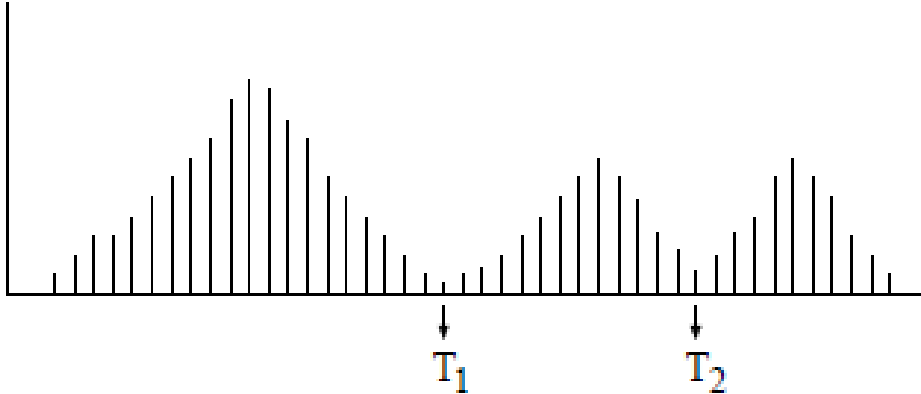


$$g(x,y) = \begin{cases} 0, & \text{se } f(x,y) \leq T \\ 1, & \text{se } f(x,y) > T \end{cases}$$

- Extrair os objetos do fundo
 - Seleção de um limiar T que separe os dois grupos, fornecendo $g(x,y)$

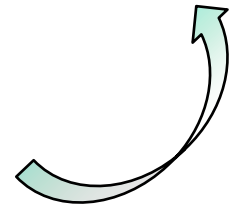
Segmentação de Imagens

■ Limiarização (Caso mais geral)



$$g(x,y) = \begin{cases} l_1, & \text{se } f(x,y) \leq T_1 \\ l_2, & \text{se } T_1 < f(x,y) \leq T_2 \\ l_3, & \text{se } f(x,y) > T_2 \end{cases}$$

- Extrair os objetos do fundo
 - Seleção de limiares para separar os grupos de interesse



Segmentação de Imagens

■ Limiarização: Ilustração

- Selecionar um valor T apropriado é uma tarefa difícil



(a) imagem original



(b) $T = 108$



(c) $T = 179$



(d) $T = 213$

Segmentação de Imagens

■ Tipos de Limiarização

□ Global:

- Um **único valor de limiar** para segmentar toda a imagem
 - **Em geral, não é adequada**
 - **Imagens podem conter variações nos níveis de cinza dos objetos e do fundo**

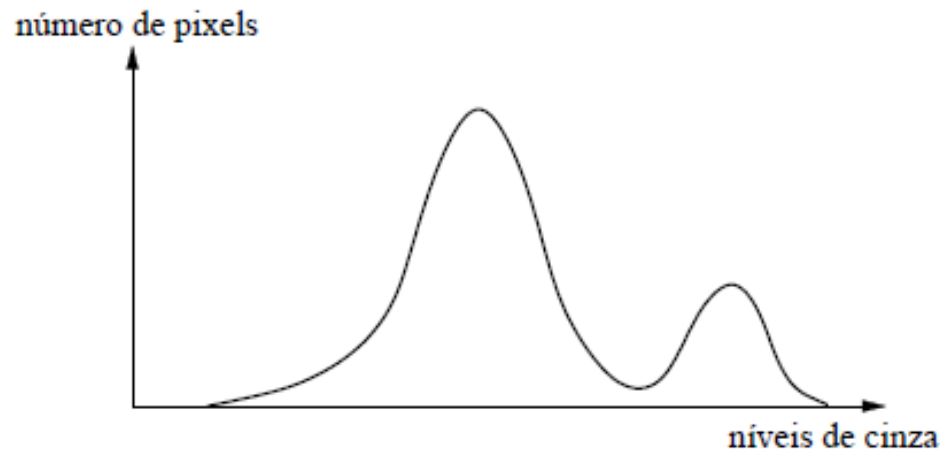
□ Local:

- **Valores de limiares podem variar sobre a imagem**
- Visa considerar as **características locais**

Segmentação de Imagens

■ Limiarização Global

- Dado um histograma bimodal



- **Desafio:**

- Encontrar **regiões de máximos** (picos)
- Selecionar **região de vale** (entre picos)

Segmentação de Imagens

■ Limiarização Global Ótima: Otsu (1979)

□ Princípio (**Variância** \implies é uma **medida de contraste**)

- **maximizar** a **variância** **entre as classes** (grupos)
- **minimizar** a **variância interna** das classes (grupos)

□ Premissas, as intensidades de pixels

- De uma **mesma classe** devem ser **similares**
- De **classes diferentes** devem ser **diferentes**

□ Cálculos realizados a partir do **histograma de uma imagem**

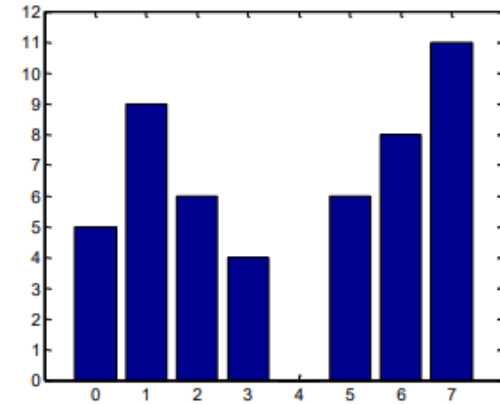
Objetivo: encontrar um valor de limiar k que divida uma imagem em dois grupos de pixels

Segmentação de Imagens

■ Limiarização Global Ótima: Otsu (1979)

- Dado um **limiar k** : $0 \leq k \leq L-1$
 - L (profundidade): indica os níveis de cinza: 0, 1, 2, 3..., $L-1$
 - Exemplo: $L = 8$

- Separar duas classes
 - $C1$: pixels entre $[0, k]$
 - $C2$: pixels entre $[k+1, L-1]$



- Porcentagens de pixels na classe $C1$: $P_1 = \sum_{i=0}^k p(i)$
- Porcentagens de pixels na classe $C2$: $P_2 = \sum_{i=k+1}^{L-1} p(i) = 1 - P_1$

$p(i)$: quantidade de pixels i dividida pelo total de pixels

Segmentação de Imagens

■ Limiarização Global Ótima: Otsu (1979)

□ Dado um limiar k : $0 \leq k \leq L-1$, calcular:

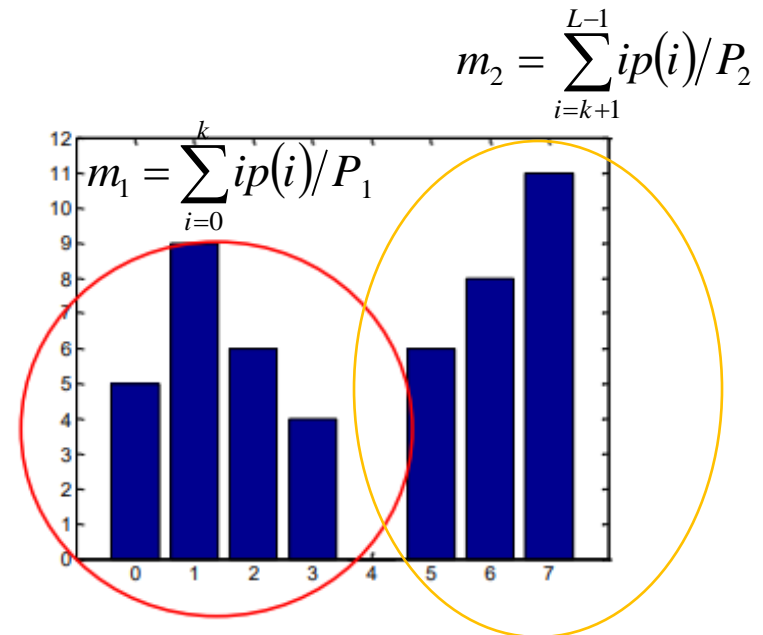
- Média das intensidades (m_1) dos pixels que pertencem a classe C1, pixels $[0,k]$

$$\square m_1 = \sum_{i=0}^k ip(i)/P_1$$

- Média das intensidades (m_2) dos pixels que pertencem a classe C2, pixels $[k+1, L-1]$

$$\square m_2 = \sum_{i=k+1}^{L-1} ip(i)/P_2$$

- Média total da imagem (m_t): $m_t = P_1m_1 + P_2m_2$



Segmentação de Imagens

■ Limiarização Global Ótima: Otsu (1979)

□ Dado um limiar k :

■ $0 \leq k \leq L-1$

□ Calcular a Variância (entre as classes):

■
$$\sigma^2 = P_1(m_1 - m_t)^2 + P_2(m_2 - m_t)^2$$

Quanto maior a distância entre m_1 e m_2 , maior será o valor da variância entre classes

■ O resultado depende do valor k (Limiar sob análise)

Avaliar a expressão para todo inteiro k

Selecionar o valor de k que maximiza a expressão

Segmentação de Imagens

■ Limiarização Global Ótima: Otsu (1979)

□ Resumo

1. Calcular o histograma normalizado da imagem de entrada
 - $p_i, i=1,2,3,\dots,L-1$
2. Calcular $P1([0,k])$ e $P2([k+1,L-1])$, $k = 0,1,2 \dots L-1$
3. Calcular médias das intensidades m_1 e m_2 , $k = 0,1,2,\dots,L-1$
4. Calcular média geral m_t das intensidades para a imagem
5. Calcular a variância entre classes σ^2 , $k = 0,1,2,\dots,L-1$
6. Obter o limiar Otsu **k** e avaliar a qualidade
7. **Repetir etapas de 2 a 6**

Bom desempenho em imagens com maior variância de intensidade

Desvantagem: método assume que o histograma da imagem seja bimodal

Segmentação de Imagens

■ Limiarização Global Ótima: Otsu (1979)

```
import matplotlib.pyplot as plt
from skimage import data
from skimage import filters
from skimage import exposure

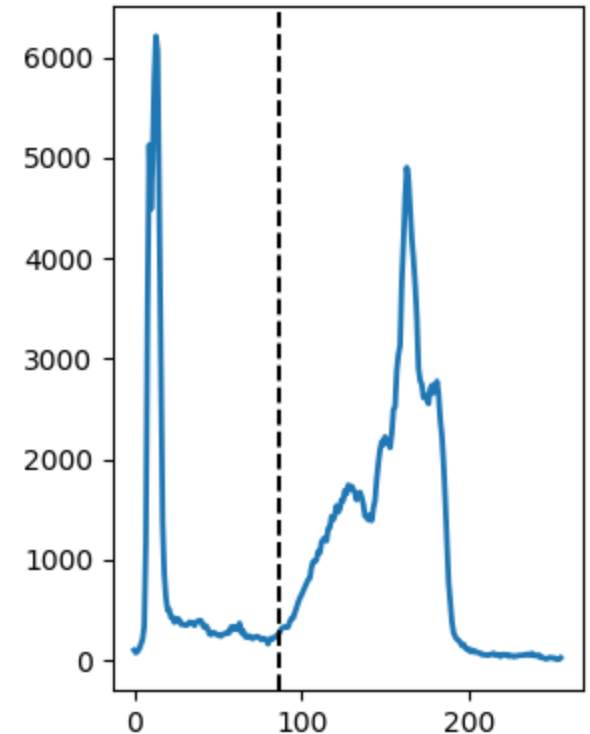
camera = data.camera()
val = filters.threshold_otsu(camera)
hist, bins_center = exposure.histogram(camera)

plt.figure(figsize=(9, 4))
plt.subplot(131)
plt.imshow(camera, cmap='gray', interpolation='nearest')
plt.axis('off')
plt.subplot(132)
plt.imshow(camera < val, cmap='gray', interpolation='nearest')
plt.axis('off')
plt.subplot(133)
plt.plot(bins_center, hist, lw=2)
plt.axvline(val, color='k', ls='--')
plt.tight_layout()
plt.show()
```

Código python disponível em (Acesso 04/2023): http://scipy-lectures.org/packages/scikit-image/auto_examples/plot_threshold.html

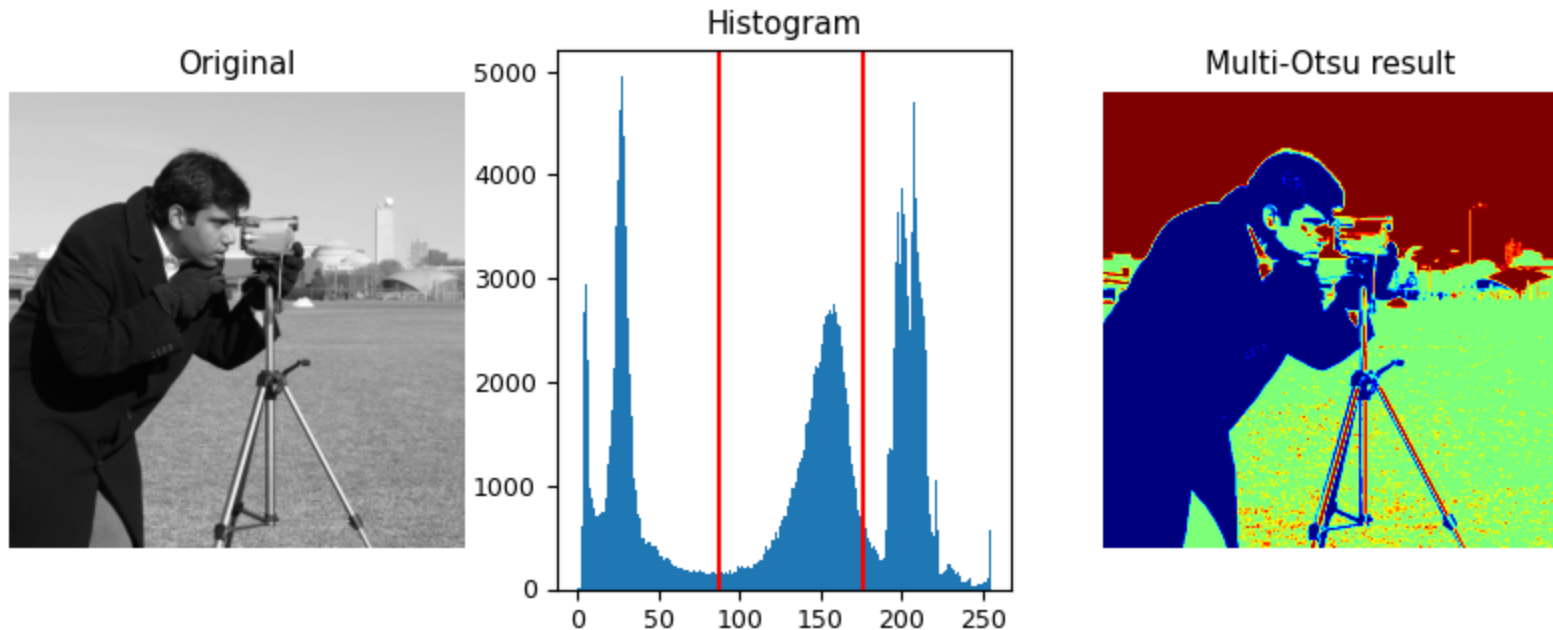
Segmentação de Imagens

■ Limiarização Global Ótima: Otsu (1979)



Segmentação de Imagens

■ Limiarização Global Ótima: Multi-Otsu (2001)



https://scikit-image.org/docs/dev/auto_examples/segmentation/plot_multiotsu.html

Segmentação de Imagens

■ Limiarização Global Ótima: Otsu (1979)

□ Outros enfoques:

- Pun (1980)

- Kapur et al. (1985)

Maximização da entropia da imagem

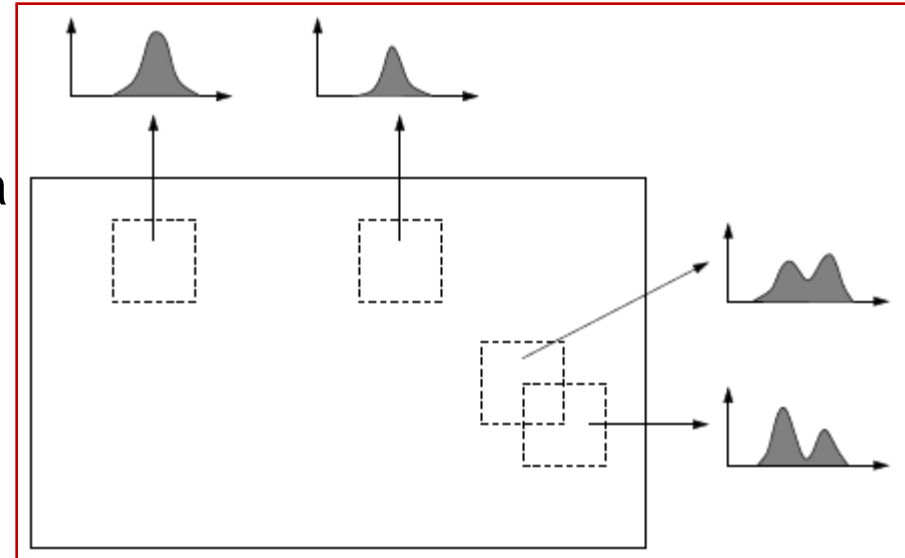
Segmentação de Imagens

■ Limiarização Local

- Analisar as intensidades dos pixels em uma região da imagem para determinar limiares locais
- **Medidas estatísticas** simples para **calcular um limiar local**:

- $T = média_v(p)$
- $T = mediana_v(p)$
- $T = \frac{\min_v(p) + \max_v(p)}{2},$

- média dos valores mínimo e máximo
- v é uma vizinhança local ao ponto p



Ponto crítico: tamanho da janela

Segmentação de Imagens

■ Limiarização Local

- Método de Bernsen (1986): $T = \frac{\min_v(p) + \max_v(p)}{2}$
- Método de Niblack (1986): $T = \mu_v(p) + k\sigma_v(p)$
 - Limiar em um pixel (x, y) é baseado na **média local (μ)** e no **desvio padrão (σ)** de uma vizinhança v de tamanho $n \times n$
 - **k**: valor é **ajustado conforme o tipo de imagem**
 - Ajuda na **supressão de ruído** e preservação de detalhes
- Sauvola e Pietaksinen (2000) $\Rightarrow T = \mu_v(p) \left[1 + k \left(\frac{\sigma_v(p)}{R} - 1 \right) \right]$
 - Baseado no método de Niblack
 - Sugerem **k=0.5** e **R=128**

Segmentação de Imagens

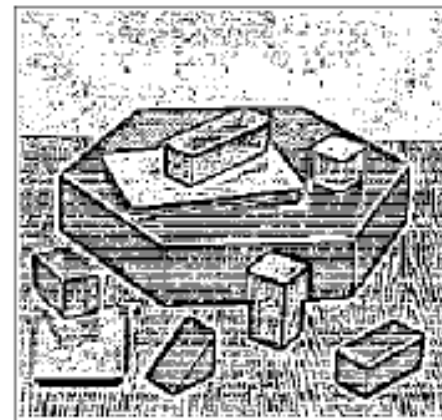
- Limiarização:
 - Exemplos



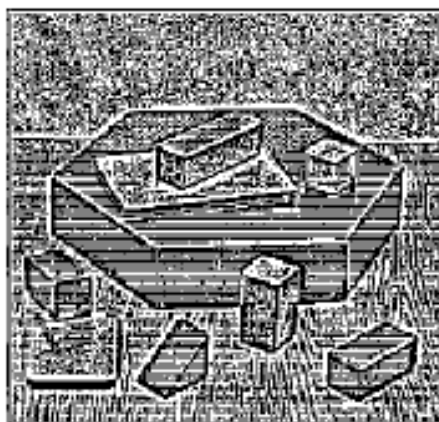
Imagem original



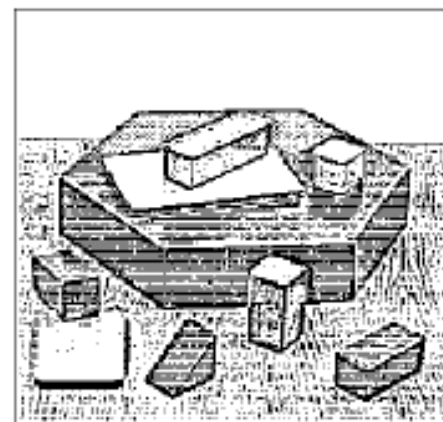
(a) global ($T = 179$)



(b) Bernsen ($n = 9$)



(c) Niblack ($n = 9$, $k = 0.01$)



(d) Sauvola e Pietaksinen ($n = 9$, $k = 0.5$, $R = 128$)

Segmentação de Imagens

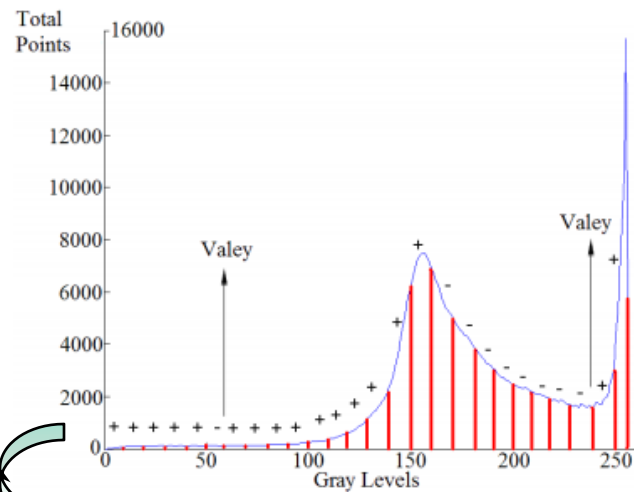
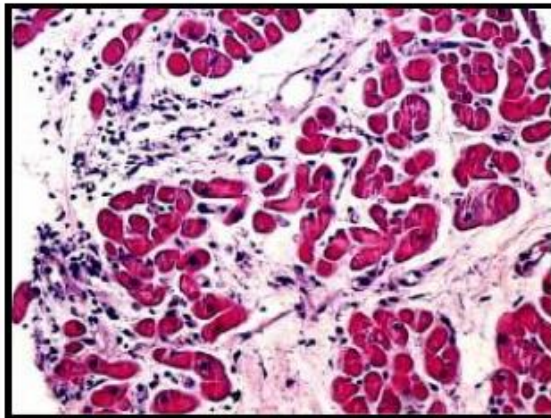


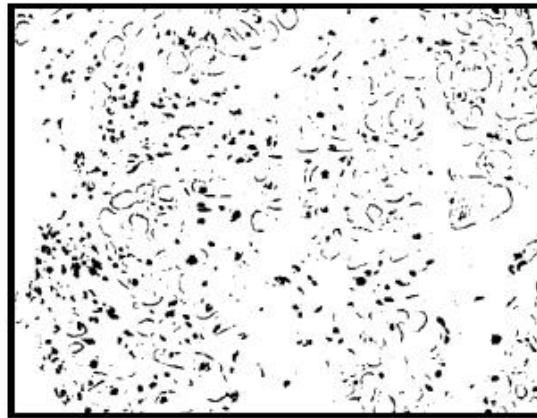
Figure 2. Example of histogram divided into 26 classes with indication of two relevant valleys to the determination of thresholds – one at level 50-59 and another at level 230-239.

■ Limiarização local

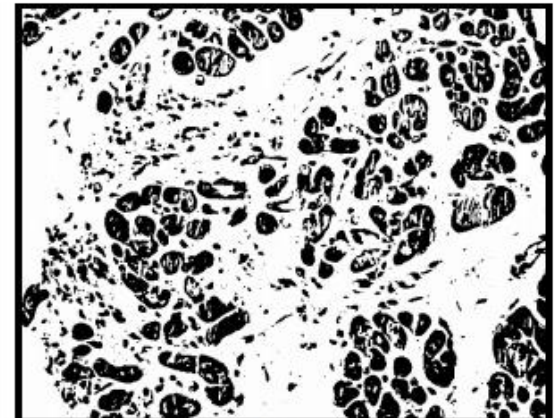
Neves, L. A., et al. "Maximum entropy, fractal dimension and lacunarity in quantification of cellular rejection in myocardial biopsy of patients submitted to heart transplantation." *Journal of Physics: Conference Series*. Vol. 285. No. 1. IOP Publishing, 2011.



(a)



(b)



(c)

Figure 4. (a) Endomyocardial biopsy classified as 2R used to illustrate visual differences obtained with different segmentation methods; (b) cell nuclei segmented using proposed method and (c) using Otsu's method [7].

Segmentação de Imagens

■ Técnicas baseadas em Similaridades:

- Agrupam pontos da imagem **com valores similares, regiões**
- **Segmentação de regiões**
 - Detectam regiões diretamente nas imagens
 - **Pontos com propriedades similares são agrupados** para formar uma região.
 - Propriedades similares
 - Intensidade de cinza, cor, informação semântica ou textura

Segmentação de Imagens

■ Segmentação de regiões

□ Seja **R a região definida** pela **própria imagem de entrada**

■ **Particionar R em n regiões** R_1, R_2, \dots, R_n , tal que:

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

(b) R_i é uma função conexa, $i = 1, 2, \dots, n$

(c) $R_i \cap R_j = \emptyset$ para todo i e j , $i \neq j$

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

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

- a) Cada pixel deve pertencer a uma região da imagem
- b) Pixels de R devem satisfazer critérios de conectividade
- c) As regiões devem ser disjuntas
- d) Pixels segmentados de R devem atender ao mesmo critério de similaridade (intensidade, ex.)
- e) Regiões adjacentes R_i e R_j são diferentes em relação ao predicado

■ $P(R_i)$: **predicado lógico** (**métrica/medida para analisar a similaridade e agrupar pixels**) sobre os pontos do conjunto R_i

■ \emptyset : conjunto vazio

Segmentação de Imagens

■ Segmentação de regiões

□ A partir dos critérios indicados, os principais tipos são:

- Crescimento de regiões
- Divisão de regiões
- Divisão e fusão de regiões

Segmentação de Imagens

■ Segmentação por crescimento de regiões

□ Conjunto inicial de pixels, denominados sementes

- Definidos de maneira aleatória, determinística ou indicados pelo usuário

□ Agregar pixels com propriedades similares em regiões

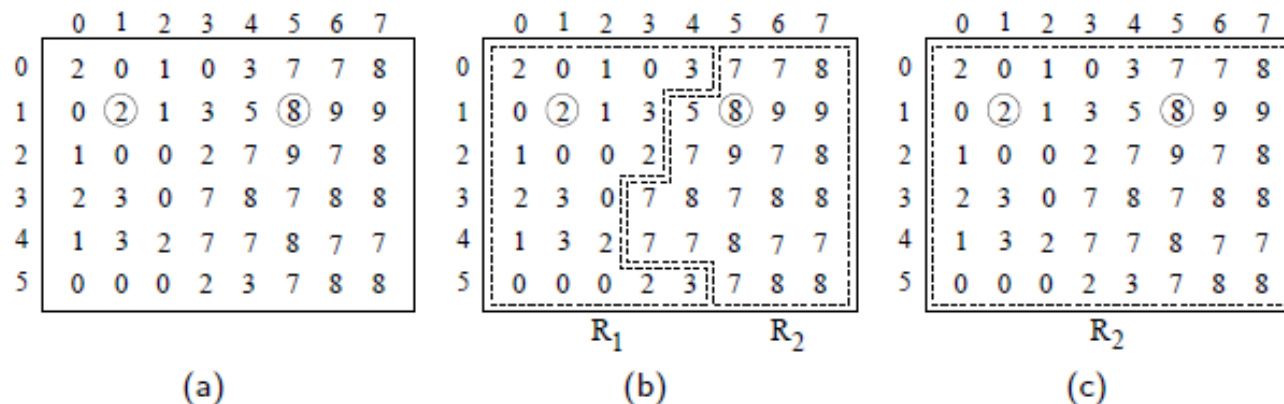


Figura: Exemplo de crescimento de regiões. (a) imagem original; (b) segmentação utilizando uma diferença absoluta menor que 4 entre os níveis de cinza; (c) segmentação utilizando uma diferença absoluta menor que 8.

Segmentação de Imagens

■ Segmentação por crescimento de regiões

- Considerando os pontos sementes com coordenadas (1, 1) e (5, 1)
 - Resultado é uma **segmentação com no máximo duas regiões (R_1 e R_2)**
- O **predicado** P é aplicado para agregar um pixel
 - Verificar se a diferença absoluta entre os níveis de cinza é menor que um limiar T

$$P(R) = \begin{cases} \text{VERDADEIRO}, & \text{se } |f(x, y) - f(r, s)| \leq T \\ \text{FALSO}, & \text{caso contrário} \end{cases}$$

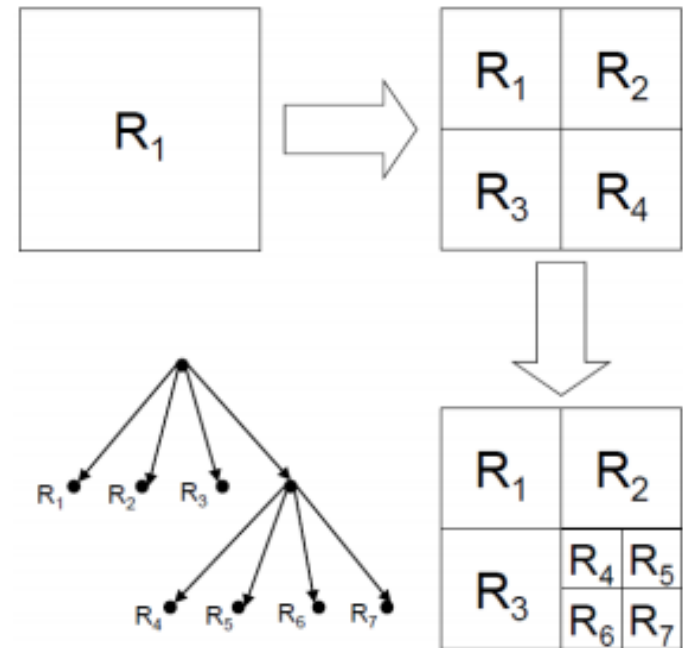
- **Critério: vizinhança-8**

Dificuldades: escolher as sementes que representem as regiões de interesse e as propriedades apropriadas para agregar os pixels

Segmentação de Imagens

■ Segmentação por divisão de regiões

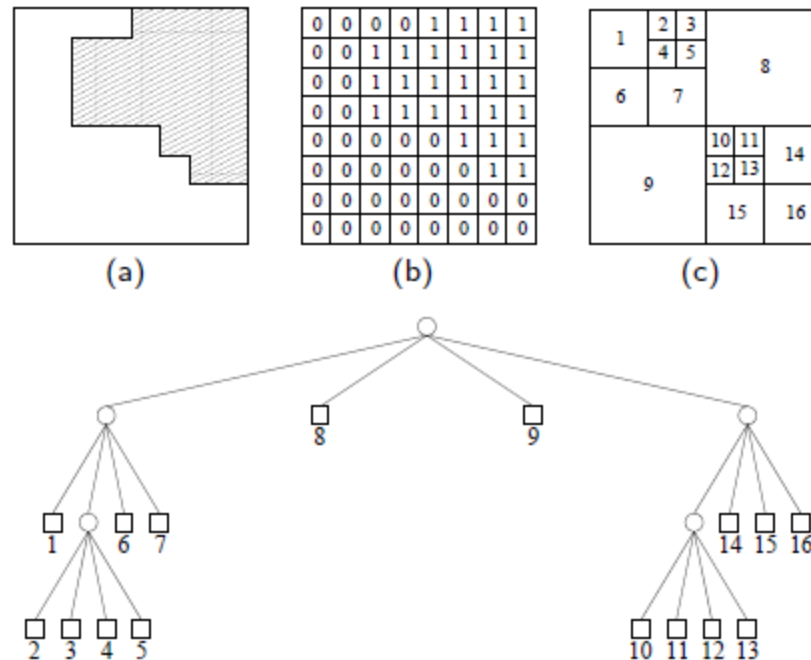
- Entrada: Imagem completa
- Definir critério de divisão
- Divisão iterativa em sub-regiões
- Parada quando falhar o critério de divisão



Segmentação de Imagens

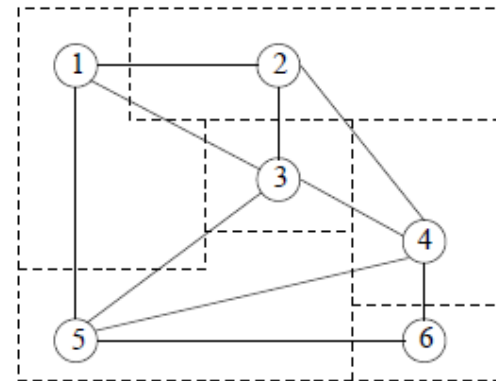
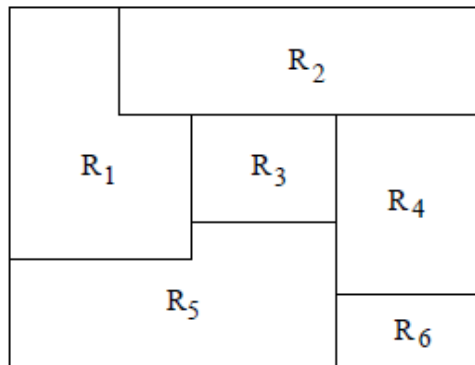
■ Segmentação por divisão de regiões

- ❑ **Problemas:** **partição final** eventualmente pode conter regiões adjacentes apresentando propriedades similares



Segmentação de Imagens

- **Segmentação por divisão e fusão de regiões**
 - Considere uma imagem segmentada em seis regiões



grafo de adjacência de regiões

- A **relação de adjacência** entre as regiões pode ser **representada** por um **grafo de adjacência**
- As **regiões** são **representadas** por um conjunto de **nós** $V = \{V_1, V_2, \dots, V_m\}$, tal que a região R_i na imagem e suas propriedades são associadas ao nó V_i
- Uma **aresta** E_{ij} entre os **nós** V_i e V_j **representa a adjacência** entre as regiões R_i e R_j
- Duas **regiões** R_i e R_j são **adjacentes** se **existir um pixel na região R_i que seja adjacente a um pixel na região R_j** por meio de vizinhança-4 ou vizinhança-8.

Segmentação de Imagens

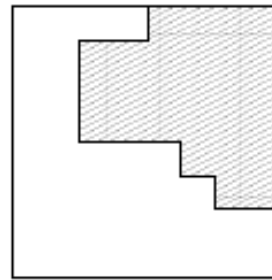
■ Segmentação por divisão e fusão de regiões

- Método útil para segmentação de imagens complexas

Algoritmo Divisão e união de regiões

- 1: Dividir em sub-regiões distintas qualquer região R_i em que $P(R_i) = \text{FALSO}$.
 - 2: Unir quaisquer regiões adjacentes R_i e R_j tal que $P(R_i \cup R_j) = \text{VERDADEIRO}$.
 - 3: Parar quando nenhuma divisão ou nenhuma fusão for mais possível.
-

- Processo Recursivo



(a)

0	0	0	0	1	1	1	1
0	0	1	1	1	1	1	1
0	0	1	1	1	1	1	1
0	0	1	1	1	1	1	1
0	0	0	0	0	1	1	1
0	0	0	0	0	0	1	1
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

(b)



(c)

**Vantagem: usar a mesma quadtree para a divisão e para a união.
Usar representação via grafo.**

Segmentação de Imagens

- **Exemplo:** *Simple Linear Iterative Clustering* (SLIC)
- Usa K-means
- Usa o conceito de *superpixel* e um espaço 5D (R,G,B,x,y)
- Requer (número de aglomerados e parâmetro de compacidade para definir a semelhança e refinamento das regiões, além de outros parâmetros)

Acesso em 04/2023: <https://scikit-image.org/docs/dev/api/skimage.segmentation.html#skimage.segmentation.slic>

Segmentação de Imagens

■ **Exemplo:** *Simple Linear Iterative Clustering* (SLIC)

```
# Importing required boundaries
from skimage.segmentation import slic, mark_boundaries
from skimage.data import retina

# Setting the plot figure as 15, 15
plt.figure(figsize=(15, 15))

# Sample Image of scikit-image package
retina = retina()
```

Segmentação de Imagens

■ **Exemplo:** *Simple Linear Iterative Clustering* (SLIC)

```
# Applying SLIC segmentation
# for the edges to be drawn over
retina_segments = slic(retina,
                        n_segments=20,
                        compactness=1)

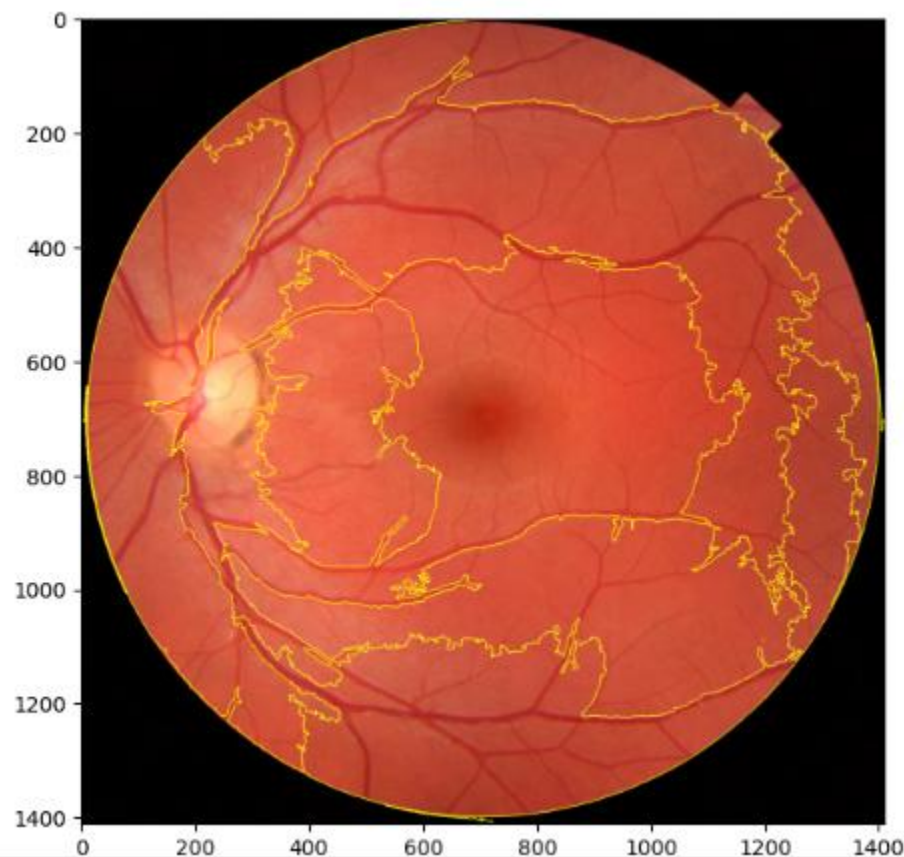
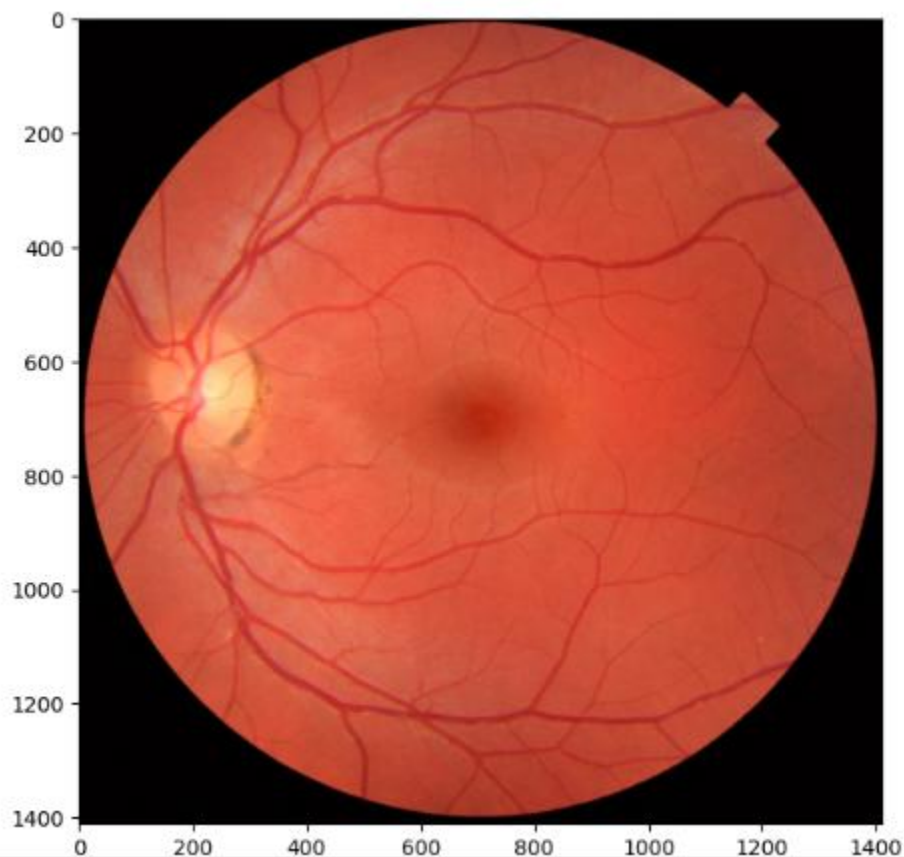
plt.subplot(1, 2, 1)

# Plotting the original image
plt.imshow(retina)

# Detecting boundaries for labels
plt.subplot(1, 2, 2)

# Plotting the output of marked_boundaries
# function i.e. the image with segmented boundaries
plt.imshow(mark_boundaries(retina, retina_segments))
```

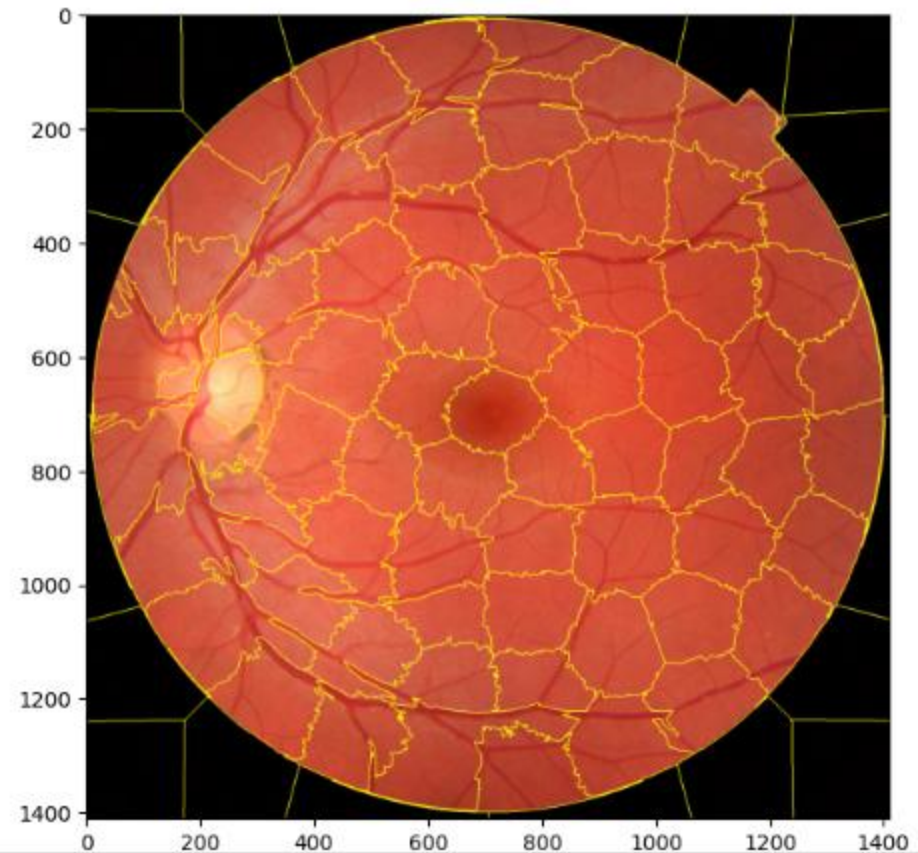
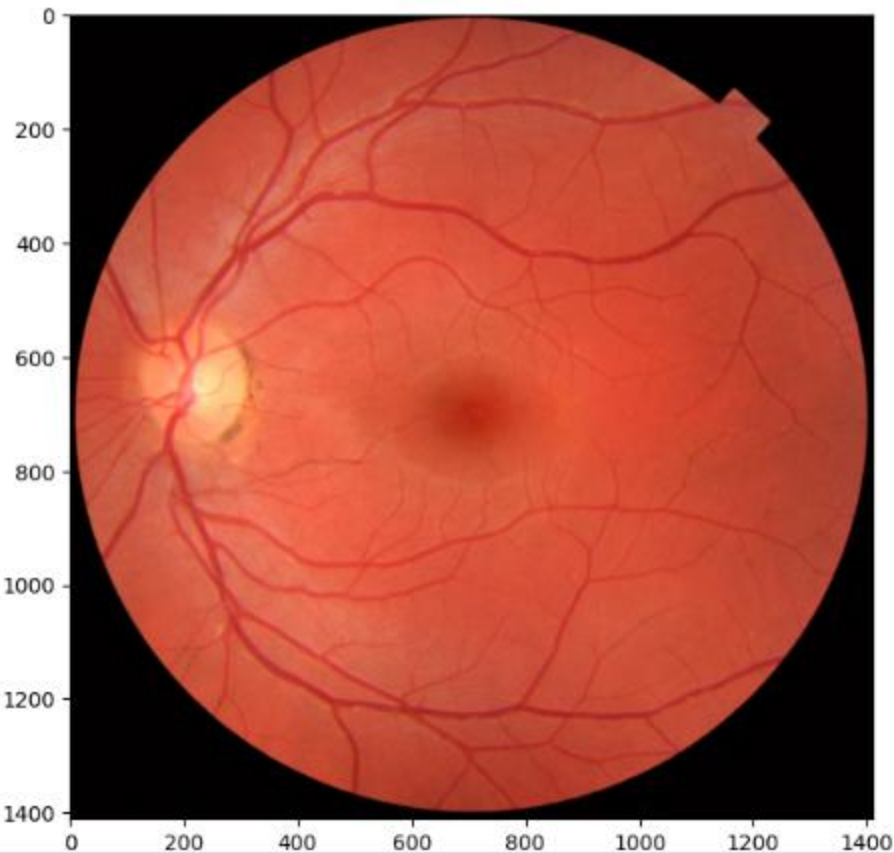
Segmentação de Imagens



n_segments=20 e compactness=1)

Segmentação de Imagens

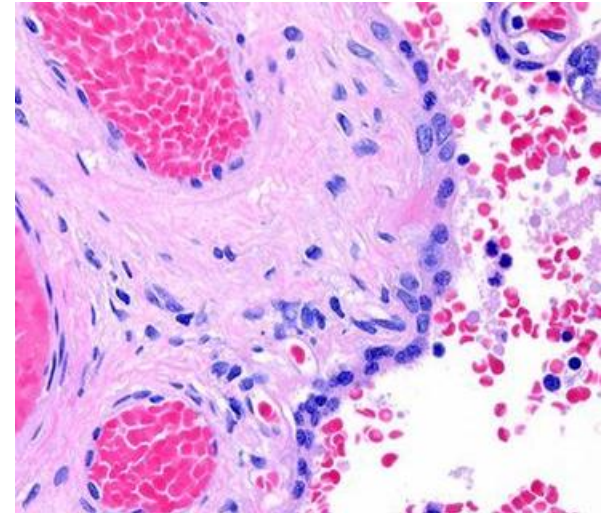
■ Exemplo: *Simple Linear Iterative Clustering* (SLIC)



n_segments=100 e compactness=10)

Exercícios

1. Exercícios listados na página 200 do livro *Análise de Imagens Digitais: Princípios, Algoritmos e Aplicações* (Hélio Pedrini), tópico 5.5 – Problemas. Total de 15 exercícios.
2. Dada a imagem à direita, desenvolva um método capaz de segmentar as regiões “circulares” em azul/violeta. Descreva cada etapa utilizada para caracterizar o método. Em seguida, considere algumas regiões de controle e calcule as taxas de acerto e erro do método. O programa deve fornecer como saída uma imagem com as regiões circulares segmentadas e as taxas obtidas.



Referências

1. Pedrini, H., Schwartz, W. R. Análise de Imagens Digitais: Princípios Algoritmos e Aplicações. São Paulo: Thomson Learning, 2008.
Leitura: Capítulo 5
2. González, R. C., Woods, R. E. Processamento de Imagens Digitais. São Paulo: Edgard Blücher Ltda, 2000.
Leitura: Capítulo 10, tópicos 10.1 a 10.4
3. Backes, A. R., Sá Junior, J. J. De M. Introdução à Visão Computacional Usando MatLab. Rio de Janeiro: Alta Books, 2016.
Leitura: Capítulo 7, tópicos 7.1 e 7.2

