



Universidade Estadual de Londrina
Departamento de Computação



Programa de Mestrado em
Computação
Módulo 10 - Processamento
Digital de Imagens- Segmentação
de Imagens

Autor: Prof. Dr. Alan Salvany Felinto
email: alan@uel.br
(2017)



Segmentação de Imagens



- Segmentação de imagens: Separa parte(s) da imagem.
- Segmentação de pontos, linhas e bordas
- Segmentação por limiar, crescimento de regiões, divisão e fusão de regiões.
- A segmentação pode ser feita tanto em imagens quanto em vídeo (imagens que variam com o tempo).



Segmentação de Imagens

Considere o processo de convolução de uma máscara com uma imagem:

$$R = w_1 z_1 + w_2 z_2 + \dots + w_9 z_9$$

$$= \sum_{i=1}^9 w_i z_i$$

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

Onde: w é a máscara e z são os pixels da imagem

a) Detectar pontos:

$$|R| > T$$

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

b) Detectar linhas:

-1	-1	-1
2	2	2
-1	-1	-1

Horizontal

R1

-1	-1	2
-1	2	-1
2	-1	-1

+45°

R2

-1	2	-1
-1	2	-1
-1	2	-1

Vertical

R3

2	-1	-1
-1	2	-1
-1	-1	2

-45°

R4

A máscara que obtiver a maior resposta indica que o ponto pertence a uma linha na direção desta máscara.



Segmentação de Imagens

Escuro

Claro

Modelo

Borda

Primeira
derivada

A magnitude da
primeira derivada é
utilizada para
detectar bordas

Segunda
derivada

O cruzamento por
zero da segunda
derivada também é
utilizada para
detectar bordas

c) Detectar bordas:

- Positiva se a borda é uma transição de escuro para claro.
- Negativa se a borda é uma transição de claro para escuro.

Caso a derivada primeira seja positiva:

- valor positivo da segunda derivada representa parte escura.
- valor negativo da segunda derivada representa parte clara.

Caso a derivada primeira seja negativa o que acontece com a segunda derivada?



Detectar Bordas

Operador gradiente:

$$\nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

$$|\nabla f| = \text{mag}(\nabla f) = [G_x^2 + G_y^2]^{1/2}$$

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

$$\alpha(x, y) = \tan^{-1} \left(\frac{G_y}{G_x} \right)$$

Imagem

G_x

G_y

(a)

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

(b)

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

(c)

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

$$G_x = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)$$

$$G_y = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)$$

Ângulo de direção da borda em relação ao eixo x

Laplaciano:

0	-1	0
-1	4	-1
0	-1	0

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

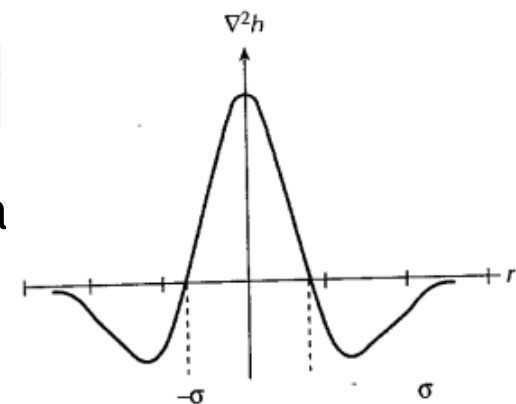
Gaussiana

$$h(x, y) = \exp \left(-\frac{x^2 + y^2}{2\sigma^2} \right)$$

Derivada Segunda da gaussiana

$$\nabla^2 h = \left(\frac{r^2 - \sigma^2}{\sigma^4} \right) \exp \left(-\frac{r^2}{2\sigma^2} \right)$$

$$r^2 = x^2 + y^2$$



- Cruzamento com zero detecta bordas
- Borra (retira ruído) quanto maior o sigma mais borra

$$\nabla^2 f = 4z_5 - (z_2 + z_4 + z_6 + z_8)$$



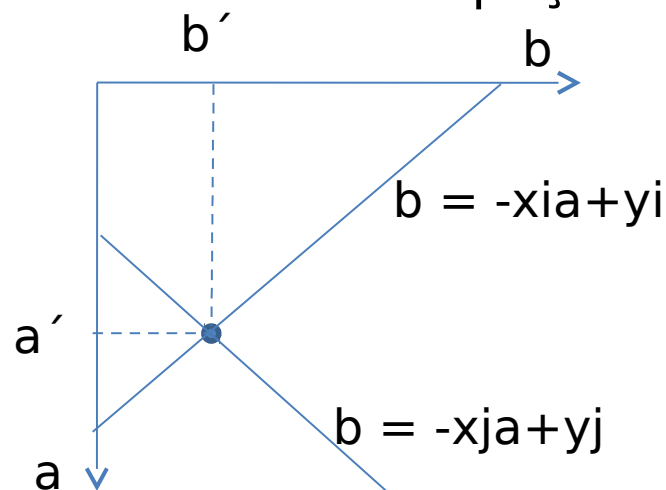
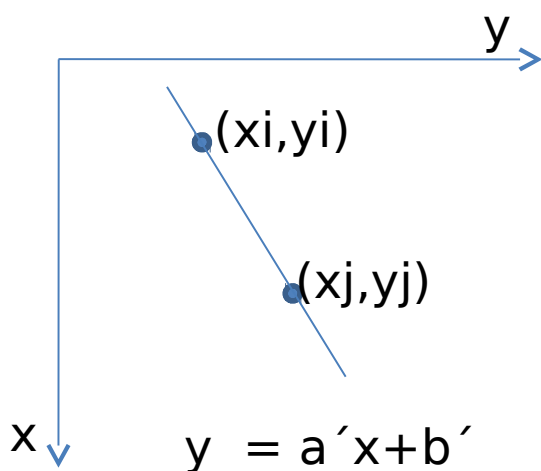
Pontos alinhados

- Considera a relação global entre os pixels.
- Como calcula os subconjuntos de n pontos alinhados ?
(separa os pontos alinhados).
- Como achar todos os pontos alinhados de uma imagem ?
- A cada dois pontos calcula-se a equação da reta e verifique se existem pontos com a mesma equação da reta (Esta abordagem é proibitiva devido ao custo computacional envolvidos).
- Solução da Transformada de Hough.



Transformada de Hough

- Técnica de Hough [1062]
- Dado um ponto $p(x_i, y_i)$
- $y_i = ax_i + b$, Infinitas linhas passam por (x_i, y_i) que satisfazem a equação.
- $b = -x_i a + y_i$, espaço de parâmetro (a, b) . Fixa (x_i, y_i) e forneça “a” para produzir uma linha neste espaço.



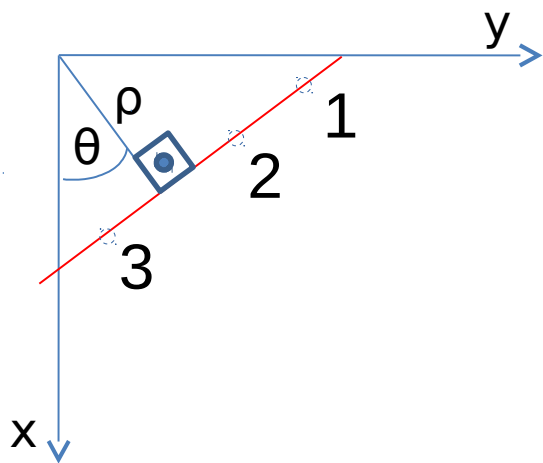
Considerando que $a = \text{deltay}/\text{deltax}$ temos que o problema com as retas próximas a vertical onde deltax tendem a zero e “a” tende ao infinito.



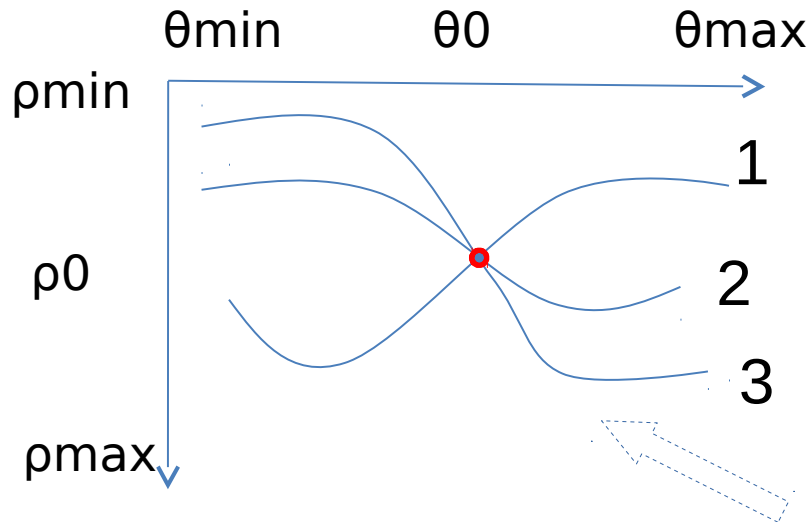
Transformada de Hough

Solução :

$$x.\cos(\theta) + y.\sin(\theta) = \rho$$



Fixando x, y e variando θ e ρ



Matriz acumuladora

Obs: Linha na vertical: $\theta = 0$, $x = \rho$



Transforma de Hough

Algoritmo 1:

- Detectar as bordas.
- Definir a matriz acumuladora MA:
 - Definir e quantizar: a_{min} , a_{max} , b_{min} e b_{max} .
 - $MA[a_{min} .. a_{max}][b_{min} .. b_{max}]$
- Com todos os píxels (x,y) , obtidos pela filtragem passa alta, faça:
 For($a = a_{min}$; $a \leq a_{max}$; $a++$) {
 $b = -xa + y$;
 $MA[a][b]++$
 }

Pesquisar as maiores contagens em $MA[a][b]$

Observe que se $MA[a][b] = k$

então temos k pontos colineares sobre a linha $y = ax + b$



Transformada de Hough

Algoritmo 2:

- Detectar bordas
 - Definir e quantizar (p_{min} , p_{max} , θ_{min} e θ_{max})
 - $MA[\rho][\theta]$ = matriz acumuladora.
 - Com todos os píxels de borda faça:
 For($\theta = \theta_{min}$; $\theta \leq \theta_{max}$; $\theta++$) {
 $\rho = x \cdot \cos(\theta) + y \cdot \sin(\theta)$;
 $MA[\rho][\theta]++$
 }
• Pesquisar as maiores contagens em $MA[\rho][\theta]$.
- Obs: se $MA[\rho][\theta] = k$
então temos k pontos sobre a linha $\rho = x \cdot \cos(\theta) + y \cdot \sin(\theta)$;



Segmentação de Imagens



Figura 6.1. Etapas de um sistema de reconhecimento de padrões
Livro: Conci, A; Azevedo, E; Leta, F.R.
Computação gráfica- volume 2, editora campus

Segmentação pelo contorno do objeto:
Filtragens no domínio do espaço e/ou da frequência

Segmentação por região:
É baseada na caracterização de um grupo de pixels conectados.



Segmentação por Limiar

Limiar ou “threshold”

Imagem original

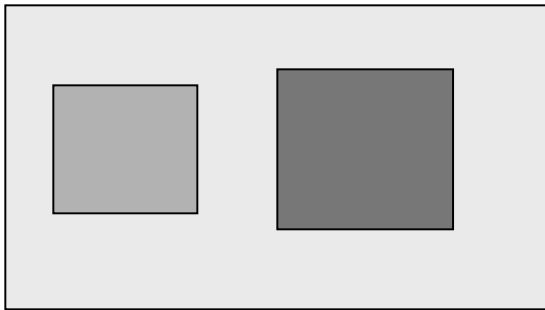
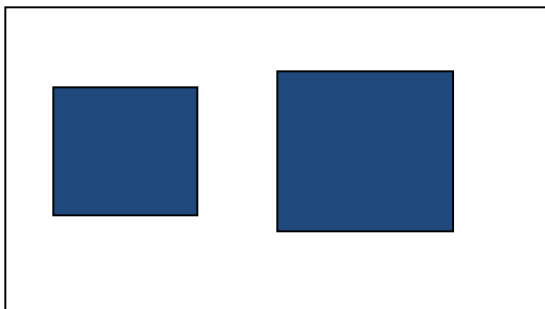
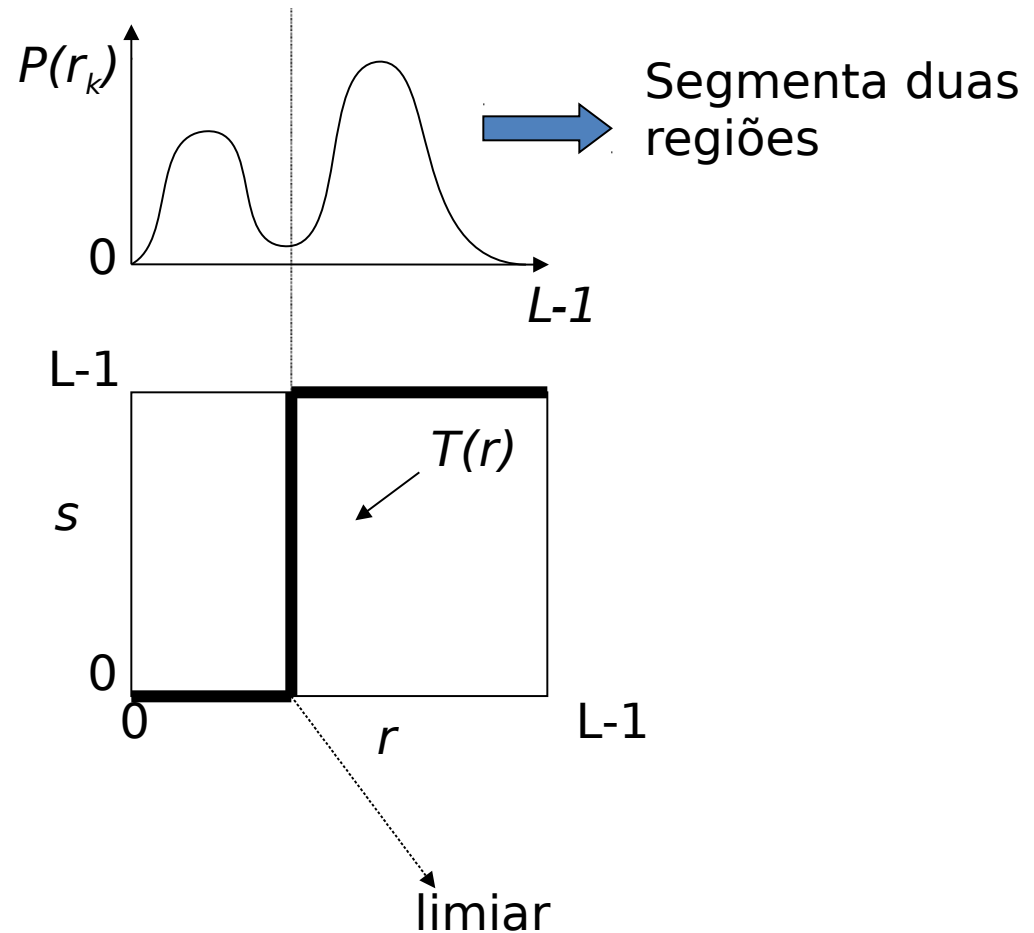


Imagem resultante



Histograma





Segmentação por Limiar

Limiar ou “threshold”

Imagem original

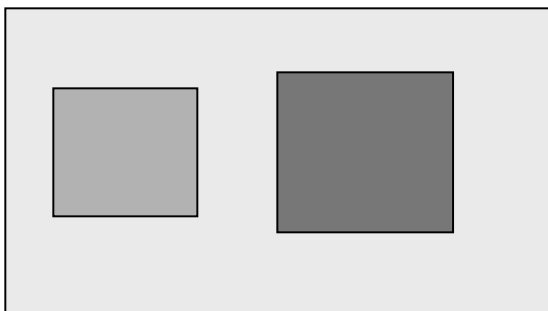
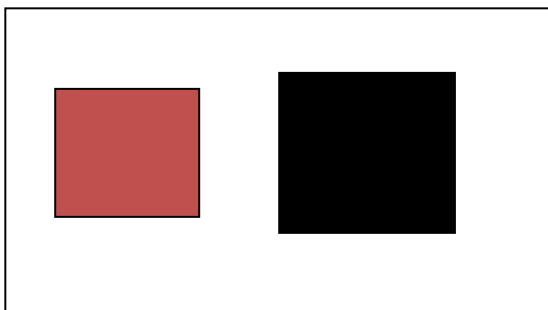
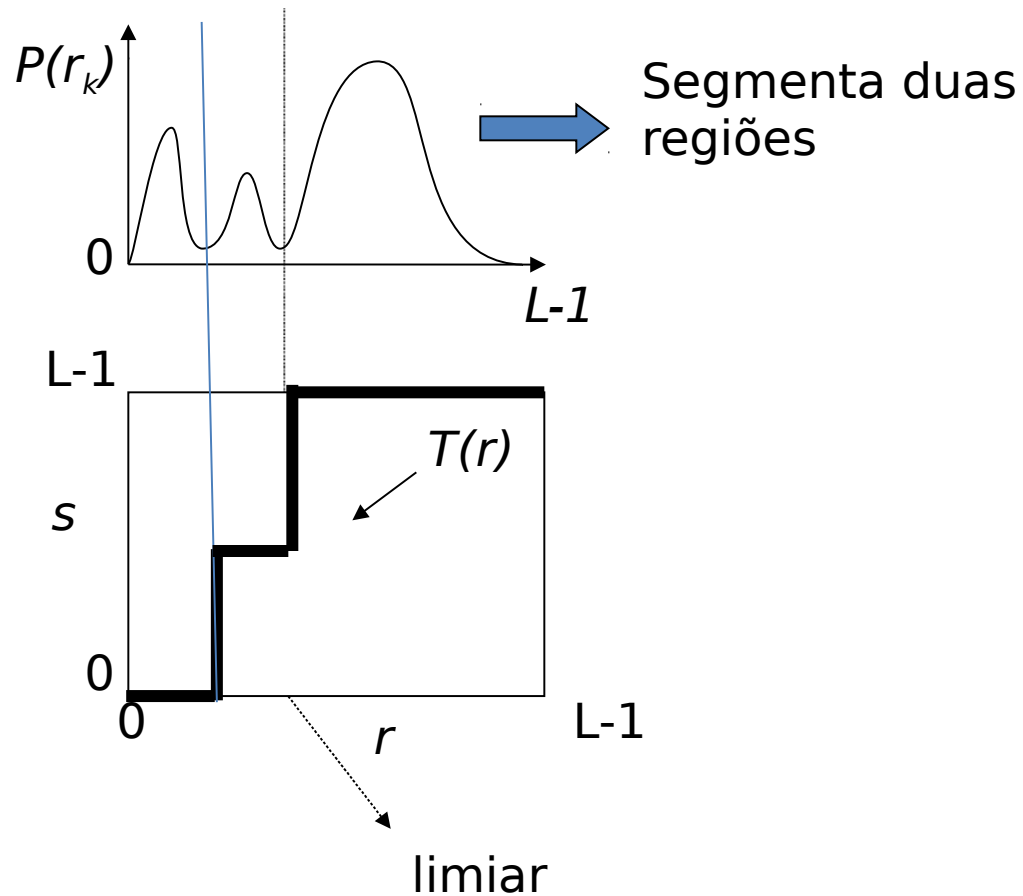


Imagem resultante



Histograma



Obs: Árvore de decisão – limiar automático

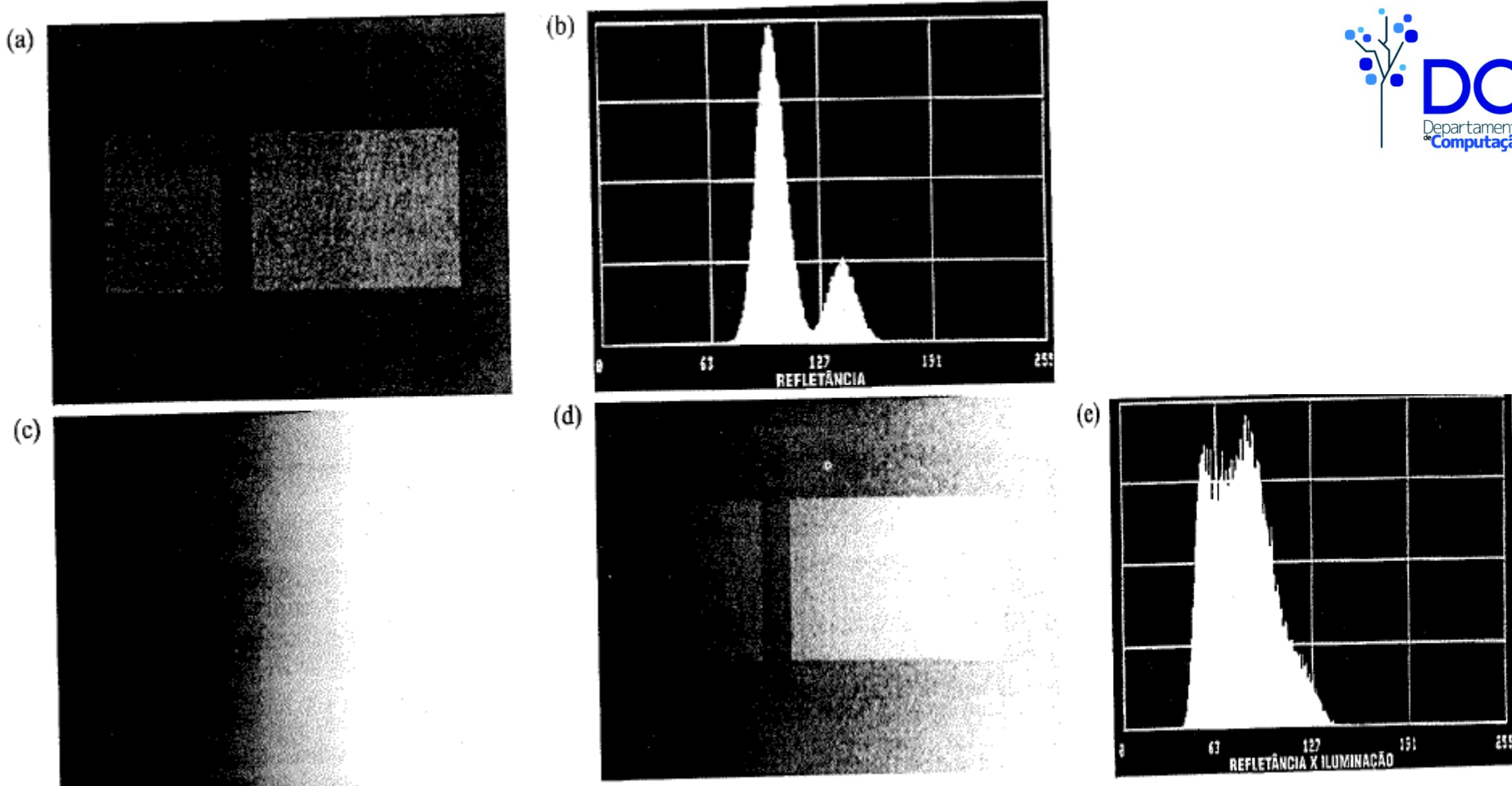


Figura 7.26 (Gonzalez & Woods - Livro texto da disciplina)

- a) Função de refletância $r(x,y)$ b) Histograma de $r(x,y)$ c) função de iluminação $g(x,y) = K_i(x,y)$
- d) $f(x,y) = i(x,y) \cdot r(x,y)$ (imagem) e) Histograma de $f(x,y)$ de "d"

$f(x,y)/g(x,y) = r(x,y)/k \rightarrow$ isso facilita a segmentação



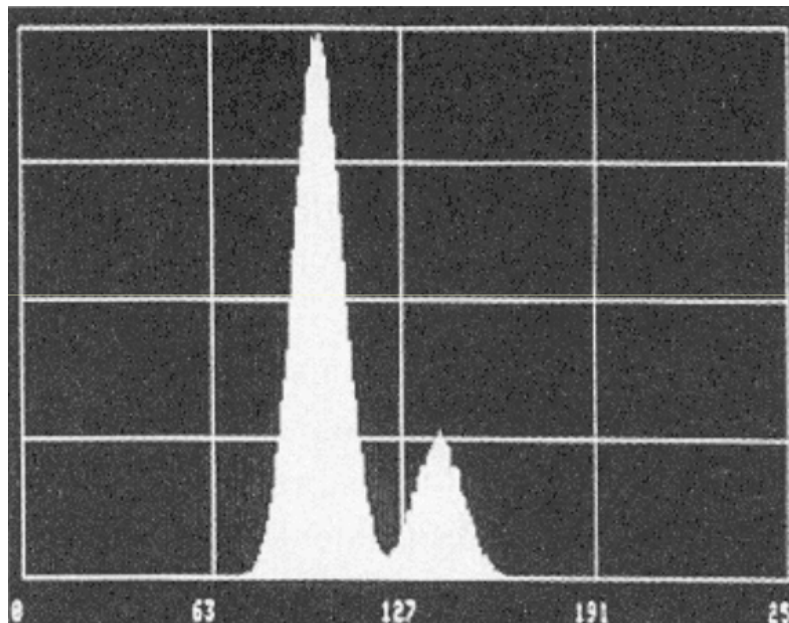
Otsu

Útil para classificar 2 classes (pode também ser usada para classificar mais que duas classes)

Utiliza o Histograma

Maximiza a variância entre duas classes (média de cada classe afastada)

Minimiza a variância de cada classe (agrupa os dados de cada classe)



Histograma Bimodal



Considere: Uma imagem MN

$$p_i = n_i / MN$$

Probabilidade de encontrar o nível de cinza i na imagem
(p é o histograma)

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

Onde L é igual a 256

Considere um limiar K que separe o histograma em 2 Classes:

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

Probabilidade do pixel pertencer a classe 1

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

Probabilidade do pixel pertencer a classe 2

Média da classe 1

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

$P(C_1/i) = 1$, probabilidade de C_1 , dado i .

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

Bayes

$$= \frac{1}{P_1(k)} \sum_{i=0}^k i p_i$$



Otsu

Média da classe 2

$$m_2(k) = \sum_{i=k+1}^{L-1} iP(i/C_2)$$

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

Média até o Nível de cinza K:

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

Média Global

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

Métrica é maximizar a equação:

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

Variância entre classes

Variância global (constante)

Verdades:

$$P_1 m_1 + P_2 m_2 = m_G$$

$$P_1 + P_2 = 1$$

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

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

$$\sigma_B^2(k) = \frac{[m_G P_1(k) - m(k)]^2}{P_1(k)[1 - P_1(k)]}$$

$$\sigma_B^2(k*) = \max_{0 \leq k \leq L-1} \sigma_B^2(k)$$

Resumo:

1. Calcular o histograma

p_i

2. Calcular

$P_1(k)$

para $k=0,1,\dots,L-1$

3. Calcular

$m(k)$

4. Calcular

m_G

5. calcular

$\sigma_B^2(k*)$

6. caso de empate use a média



Canny

Princípios:

- Baixa taxa de erro: detectar somente as bordas (suprimir os ruídos)
- Boa localização: a Borda detectada deve estar na posição real da borda
- Resposta Mínima: A borda deve ser da espessura de um pixel (suprimir ruídos)

Algoritmo:

- Redução de ruído: Utilize um filtro gaussiano.

$$K = \frac{1}{159} \begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix}$$

- Gradiente da Imagem: Aplica-se o filtro Sobel
Calcule a Magnetude e a direção de cada píxel

$$\nabla f = \text{mag}(\nabla f) = [G_x^2 + G_y^2]^{1/2}$$

$$\alpha(x, y) = \tan^{-1} \left(\frac{G_y}{G_x} \right)$$

Gx

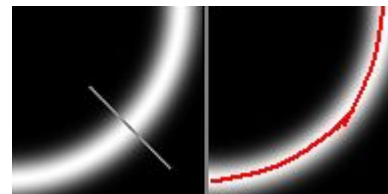
-1	-2	-1
0	0	0
1	2	1

Gy

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

- Afinar a borda:

Maior magnetude ao longo da direção normal da borda.

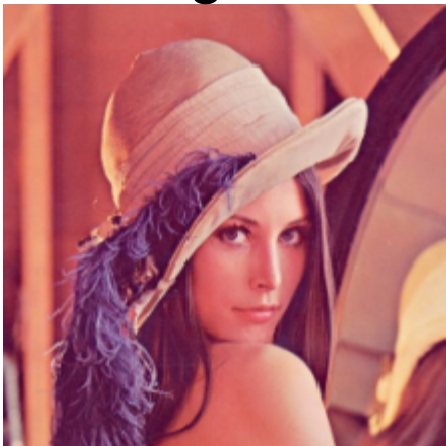


- Histerese: Utiliza 2 limiares com a finalidade de preencher pequenas lacunas
Se o gradiente do píxel for maior que o **limiar superior** então é aceito como borda.
Se o gradiente do píxel for menor que o **limiar inferior** então é rejeitado como borda.
Se o gradiente do píxel estiver entre os dois limiares E Se o píxel estiver conectado a um píxel cujo gradiente seja maior que o limiar superior então é aceito como borda



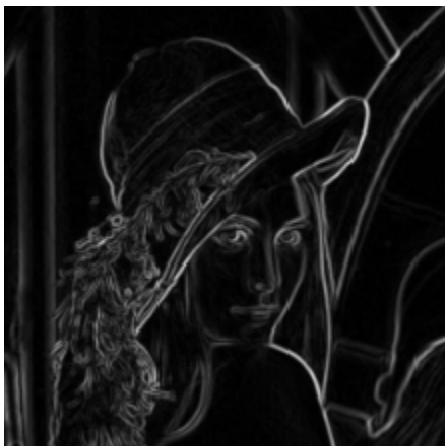
Algoritmo:
Original

Redução de Ruído

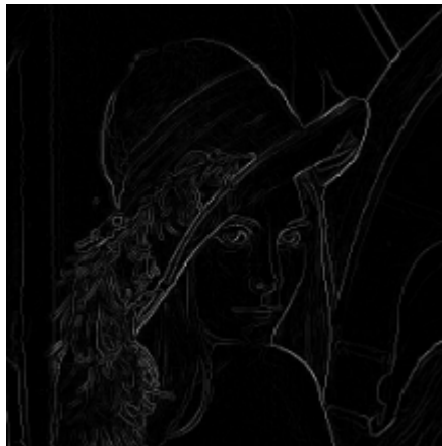


Canny

Gradiente da Imagem



Afina a borda



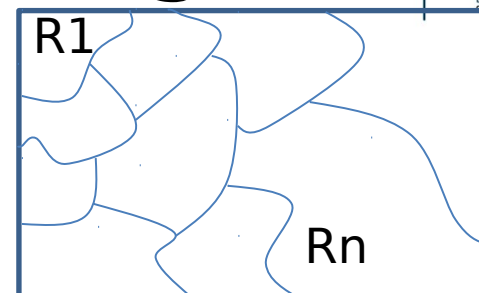
Histerese



- Canny, J., A Computational Approach to Edge Detection, IEEE Trans. Pattern Analysis and Machine Intelligence, 8:679-714, 1986.
- http://docs.opencv.org/2.4/doc/tutorials/imgproc/imgtrans/canny_detector/canny_detector.html

Segmentação Orientada a regiões:

- Fornece informações espaciais.
- Regiões possuem características próprias.



Dividir a imagem em regiões segundo um critério pré-estabelecido.

Algoritmo

(Exemplo de crescimento de regiões por agregação de pixels)

Plante n sementes em uma imagem,

cada semente irá crescer e formar uma região R .

Considere m_i = média da cor (níveis de cinza de uma região R).

Repita para ($i = 1 \dots n$)

Para (cada pixel P na borda de R_i)

Para (os pixels vizinhos não rotulados de P)

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

Inclua o vizinho a R_i , atualize m_i

Até que não haja mais inclusões de pixels em regiões.



Segmentação de Regiões

Formulação Básica:

- $R_1 \cup R_2 \cup \dots \cup R_n = R$, onde R é a região da imagem a ser segmentada
- R_i é uma região conexa, $i = 1, 2, 3, \dots, n$
- $R_i \cap R_j = \text{vazio}$ para todo i e j , i diferente de j .
- $P(R_i)$ é um predicado lógico sobre os pontos do conjunto R_i .
- $P(R_i)$ é a propriedade dos pixels da região R_i (média entre pixels, intervalo da cor de pixels, diferença entre pixels)
- $P(R_i) = \text{Verdadeiro}$ para $i = 1, 2, \dots, n$.
- $P(R_i \cup R_j) = \text{FALSO}$ para i diferente de j

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

Ex: os pixels das regiões R_1 e R_2 possuem uma diferença absoluta de menos de 3 entre os níveis de cinza.



K-Média

- 1) Plantar as K sementes (centroides) na imagem: A localização das sementes influenciam no resultado do algoritmo.
- 2) Para cada ponto da imagem classificar no cluster que obtiver a menor distância euclidiana (RGB do ponto com o RGB da semente).
- 3) Calcular a nova semente (nova média da característica do cluster) para cada um dos K clusters.
- 4) Repetir até que não altere os valores das sementes, ou escolher um número fixo de interação, ou que não haja mudança de um pixel de um cluster para outro cluster.

Obs:

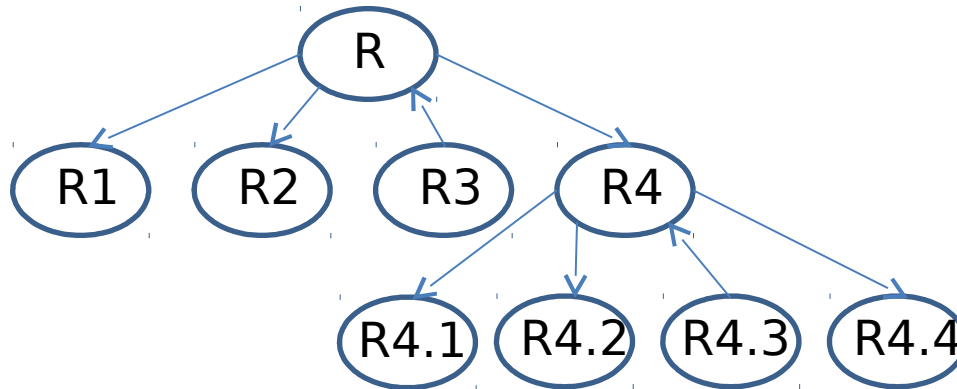
Três características para cada semente: R, G, B



Divisão e fusão de regiões

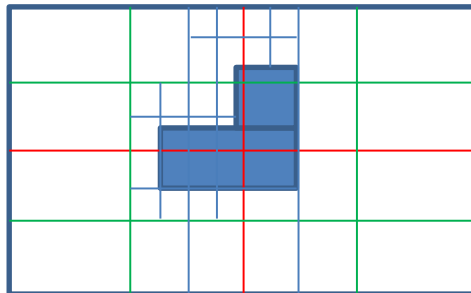
Quadtree

R1	R2	
R3	R4.1R4.2	
	R4.3R4.4	



Resumo

1. Se $P(R_i) = \text{falso}$ então divida por 4 a região R_i
2. Se R_j e R_k são regiões adjacentes e $P(R_j \cup R_k) = \text{verdadeiro}$ então junte as regiões R_j e R_k em uma única região.
3. Pare quando não for mais possível dividir ou fundir regiões.





Movimentação e Segmentação

Técnica Espacial

- Seja $f(x, y, t_i)$ e $f(x, y, t_j)$ dois quadros de imagens tomados nos instantes t_i e t_j .
- Considere que não há mudanças nas características do ambiente durante as aquisições das imagens e que existam objetos em movimentos.
- Seja D_{ij} diferenças entre as duas imagens.
- $D_{ij}(x, y) = 1$ se $|f(x, y, t_i) - f(x, y, t_j)| > T$
0 caso contrário.
- Onde θ é um limiar (calibrar o valor de θ para não pegar pequenas variações).
- Utilizando D_{ij} temos que os valores 1 estão relacionados aos objetos em movimentos e os valores 0 estão relacionados com os objetos estacionários.
- E o ruído o que fazer? (ignorar pequenos agrupamentos de pixels).
- Mudança de iluminação (Normalizar a imagem pela média das intensidades dos pixels)



Diferenças Acumulativas

- Considere uma sequência de imagens.
- $F(x,y,t_1), f(x,y,t_2), \dots, f(x,y,t_n)$
- Existem 3 tipos de diferenças acumuladas:
 - Absoluta - AADI - $D_{ij}(x,y) = 1$ se $|f(x,y,t_i) - f(x,y,t_j)| > T$
0 caso contrário.
 - Positiva - PADI - $D_{ij}(x,y) = 1$ se $f(x,y,t_i) - f(x,y,t_j) > T$
0 caso contrário.
 - Negativa - NADI - $D_{ij}(x,y) = 1$ se $f(x,y,t_i) - f(x,y,t_j) < -T$
0 caso contrário.
- $F(x,y,t_i) - f(x,y,t_j)$ = negativo ou positivo.
 - menor - maior = negativo
 - maior - menor = positivo.



Exercícios

Desenvolvam programas em OpenCV que:

- 1) Implemente um exemplo que utilize o detector de bordas Canny em um processamento de vídeo.
- 2) Implemente um exemplo que utilize o OTSU para binarizar as imagens contidas em um vídeo.
- 5) Apresente uma segmentação de imagens coloridas utilizando o algoritmo k-means com 3, 4, 5 e 6 clusters.



Bibliografia:

