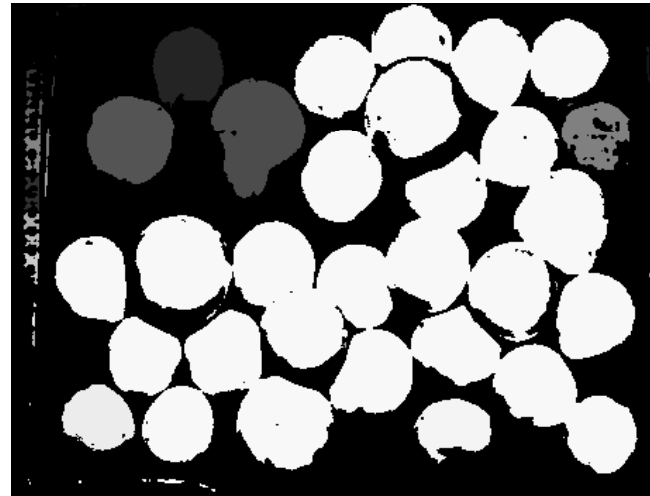


# Segmentação

- **O que é Segmentar?**
  - O processo de isolar objectos de interesse do resto da cena.
  - O processo de subdividir uma imagem em regiões disjuntas. Cada região é homogénea relativamente a uma característica. As regiões de interesse normalmente correspondem a objectos diferentes na cena.
  - Operação essencial em Visão por Computador, mas com um elevado grau de dificuldade.
  - A eficácia da segmentação é determinante para a eficácia das subsequentes operações de análise.

# Exemplo

Seleccção de cogumelos



# Exemplo

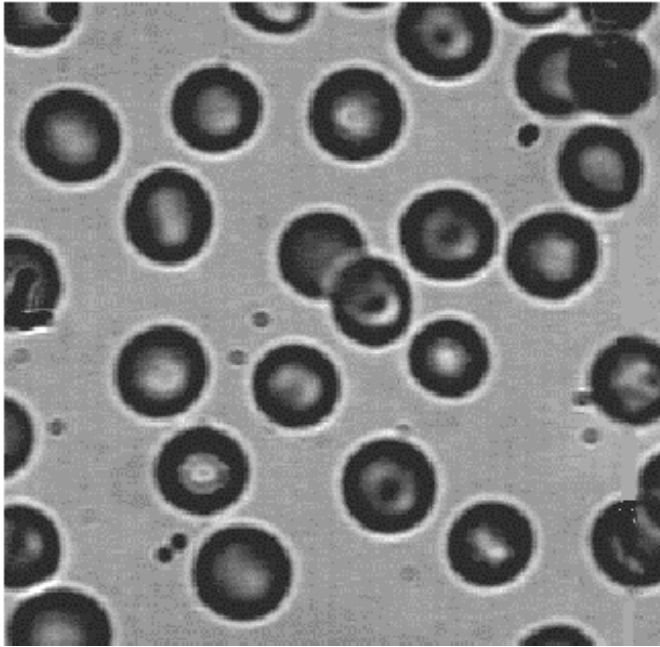


Imagem de entrada

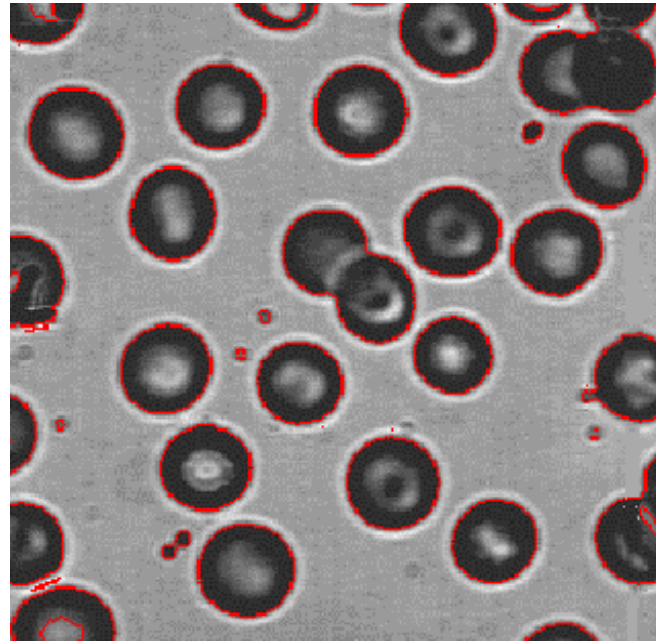


Imagem segmentada

# Segmentação

- **Qual a utilidade da Segmentação?**
  - Extrai informação
  - Realça detalhes
  - Permite efectuar medições
  - Fornece orientação
  - Permite comparação

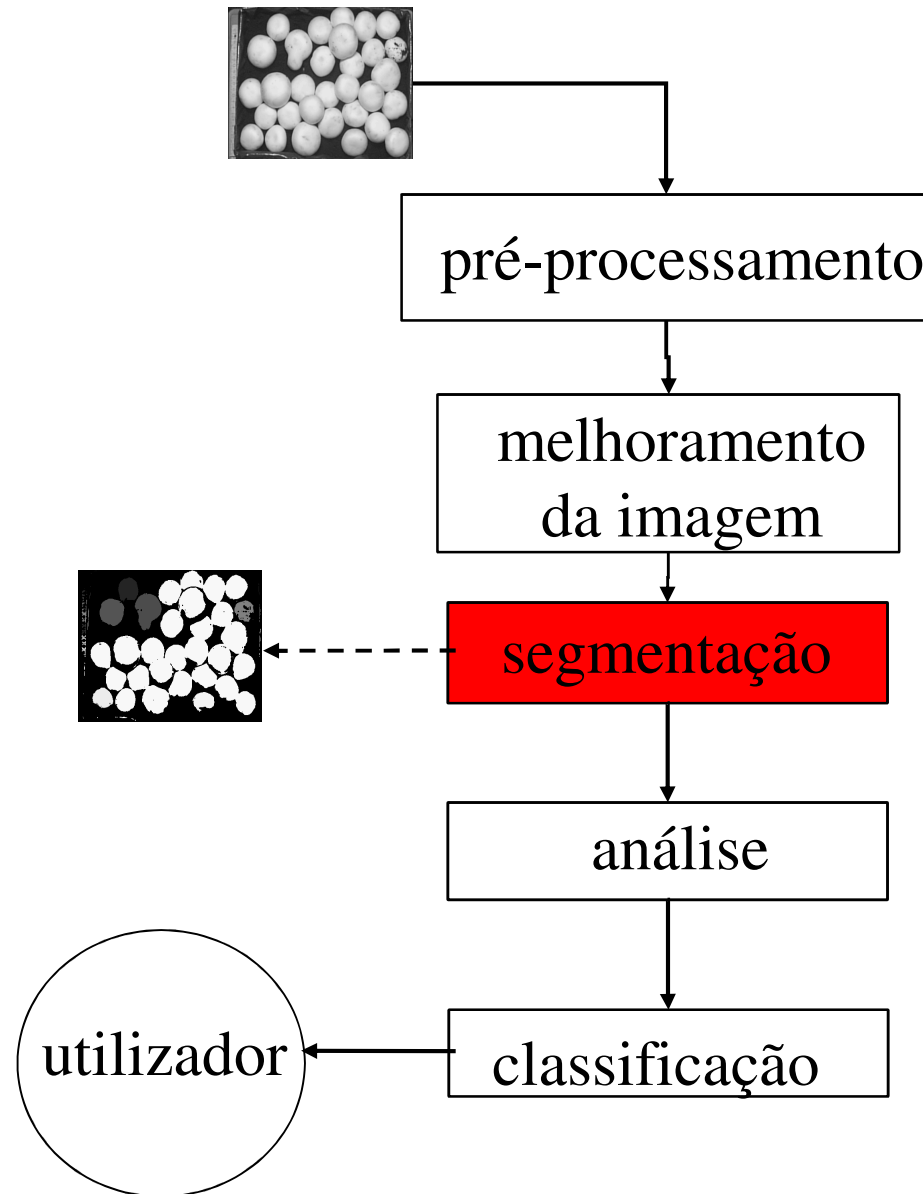
# Segmentação

- **Dificuldades na Segmentação?**
  - Mapeamento não uniforme
  - Não homogeneidade
  - Modelos de objectos inadequados
  - Artefactos espontâneos
  - Ruído

# Segmentação

## Objectivo da Segmentação:

- Dividir a imagem em regiões de interesse
- Etapa essencial para a interpretação da imagem e a sua compreensão



# Definição

R representa toda a imagem.

Segmentar é subdividir a região R em sub-regiões  $R_1, \dots, R_k$  tal que:

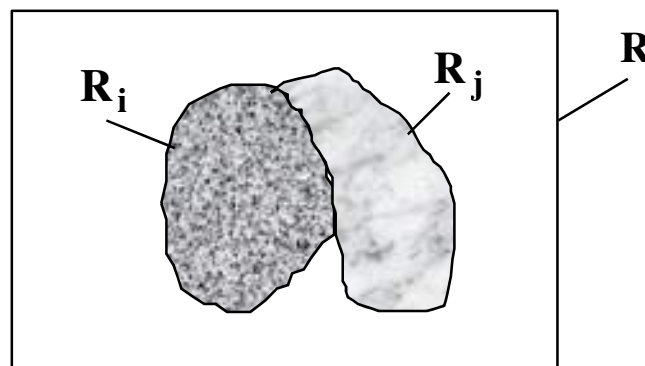
$$\bigcup_{i=1}^k R_i = R$$

$$R_i \cap R_j = \emptyset, \quad i \neq j$$

$$R_i \text{ é conexo } \forall i$$

$$P(R_i) = \text{TRUE}, \quad \forall i$$

$$P(R_i \cup R_j) = \text{FALSE} \quad i \neq j \text{ e } R_i, R_j \text{ adjacentes}$$

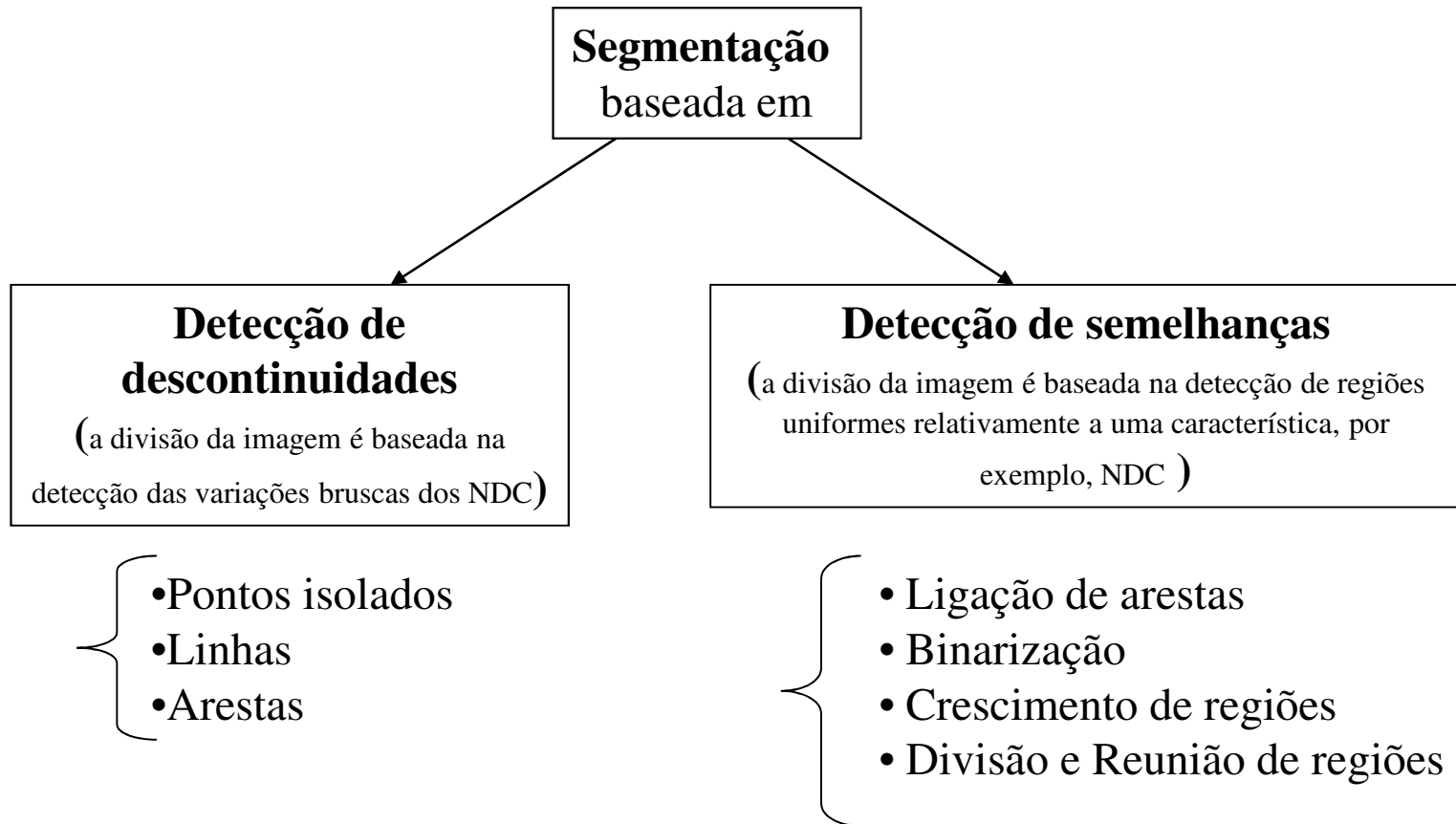


# Abordagens à Segmentação

	Paralelo	Sequencial	
Semelhanças	<b>Binarização</b>	<b>Crescimento de regiões</b>	<b>Propriedades do Pixel</b> <ul style="list-style-type: none"><li>• NDC</li></ul>
Descontinuidades	<b>Detector de arestas</b>	<b>Ligação de arestas</b>	<b>Propriedades Locais</b> <ul style="list-style-type: none"><li>• Média/variância</li><li>• Textura</li><li>• Gradiente</li></ul>



# Abordagens à Segmentação



# Binarização

- Operação de classificação realizada sobre uma imagem em NDC que distingue os pixels com NDC elevado dos pixels com um NDC baixo;
- Normalmente, a imagem resultante é binária (preto e branco).

$$I_{out}(x, y) = \begin{cases} 0, & I_{in}(x, y) < T \\ 255, & I_{in}(x, y) \geq T \end{cases}$$

**Exemplo:**

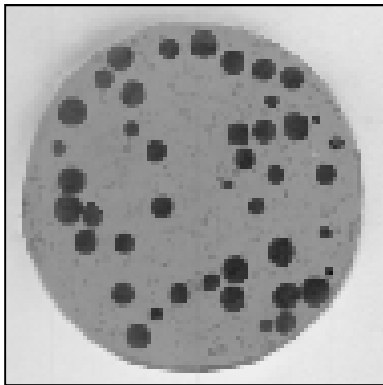


Imagem de entrada

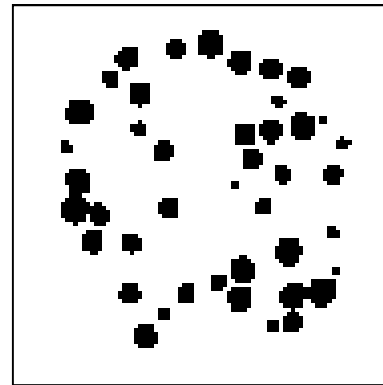


Imagem binarizada

# **Seleccção do nível binarização**

## **Manual**

- O operador selecciona interactivamente o nível de binarização para cada situação;
- Subjectividade;
- Eficaz em imagens simples;
- Exige a presença do operador.

## **Automática**

- A binarização é efectuada com base num algoritmo que analisa as características da imagem e define qual o nível de binarização adequado para cada situação;
- Existem várias abordagens possíveis;
- Problema permanente em PDI.

## **Semi-automática**

- A binarização é efectuada com base num algoritmo que analisa as características da imagem cabendo ao utilizador validar o resultado obtido e efectuar as correcções necessárias;
- Muito utilizada.

# Binarização

## Global

- O nível de binarização depende apenas do NDC do pixel  $(x,y)$ .
- O valor de  $T$  é fixo para toda a imagem. Normalmente baseado no histograma da imagem.

$$T = T[I_{in}(x,y)];$$

## Local

- O nível de binarização depende das propriedades do píxel e da sua vizinhança.
- A imagem é dividida em regiões e para cada uma dela é calculado o nível de binarização.

$$T = T[I_{in}(x,y), V(x,y)];$$

## Dinâmica ou adaptativa

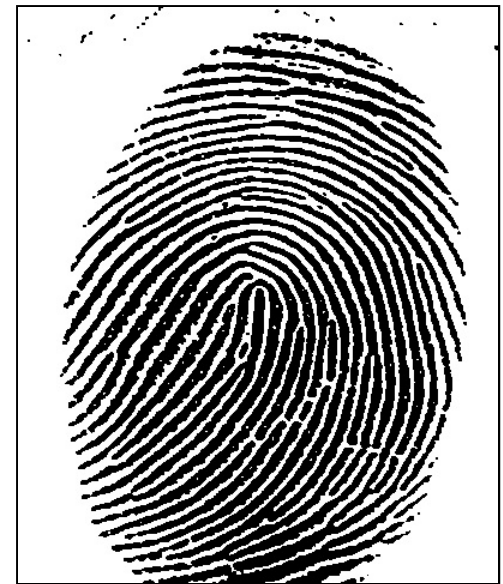
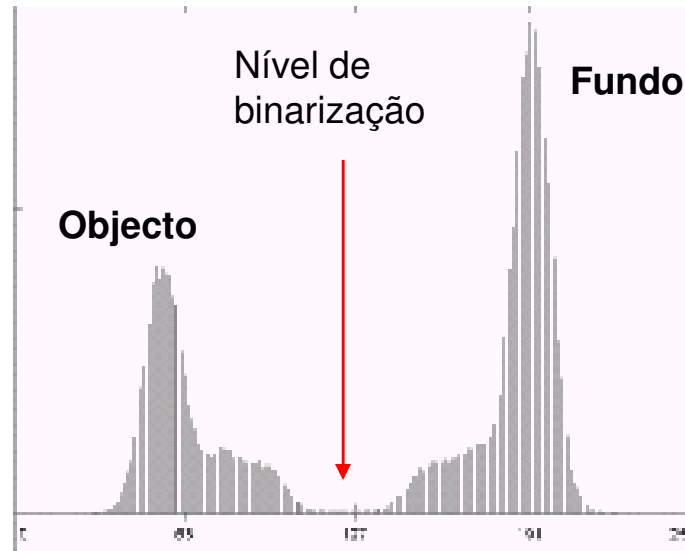
- O nível de binarização depende das propriedades do píxel e da sua vizinhança e ainda da sua localização espacial dada pelas coordenadas  $(x,y)$ .

$$T = T[I_{in}(x,y), V(x,y), x, y].$$

# Binarização global simples

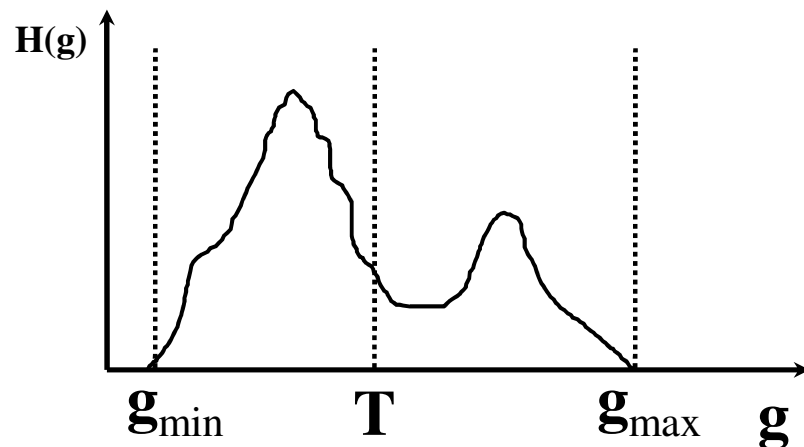
Baseado na inspecção visual do histograma

Histograma bimodal



# Binarização global simples

Seleccção do nível de binarização automaticamente (I)



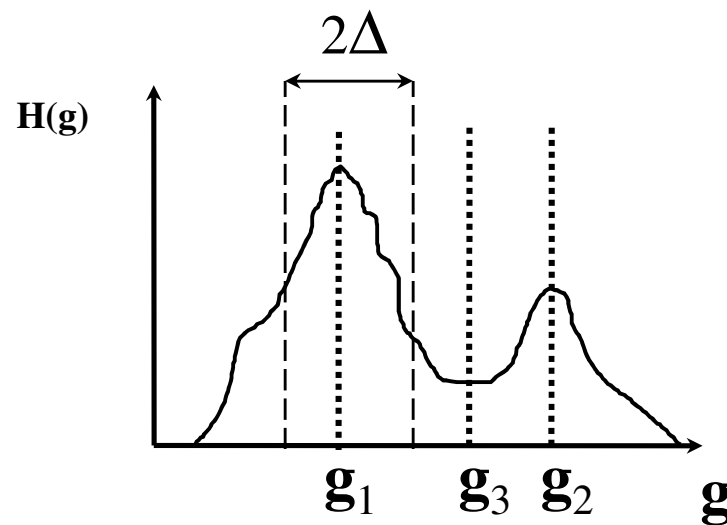
1.  $t_0 = \frac{g_{\max} - g_{\min}}{2}, \quad i = 0$
2.  $\bar{m}_1 = \frac{\sum_{g=g_{\min}}^{t_i} g.H(g)}{\sum_{g=g_{\min}}^{t_i} H(g)}, \quad \bar{m}_2 = \frac{\sum_{t_i}^{g=g_{\max}} g.H(g)}{\sum_{t_i}^{g=g_{\max}} H(g)}$
3.  $t_{i+1} = \frac{m_1 + m_2}{2}$
4. if  $|t_{i+1} - t_i| > \varepsilon$  then  $i = i + 1$  and goto 2.
5.  $T = t_i$
6. stop

Aplique este método à imagem da impressão digital com  $t_0=0$

# Binarização global simples

Seleção do nível de binarização automaticamente (II)

Detecção de Pico/Vale



$$g_1 : H(g_1) = \max \{H(g)\}, \forall g$$

$$g_2 : H(g_2) = \max \{H(g)\}, \forall g \notin [g_1 - \Delta; g_1 + \Delta]$$

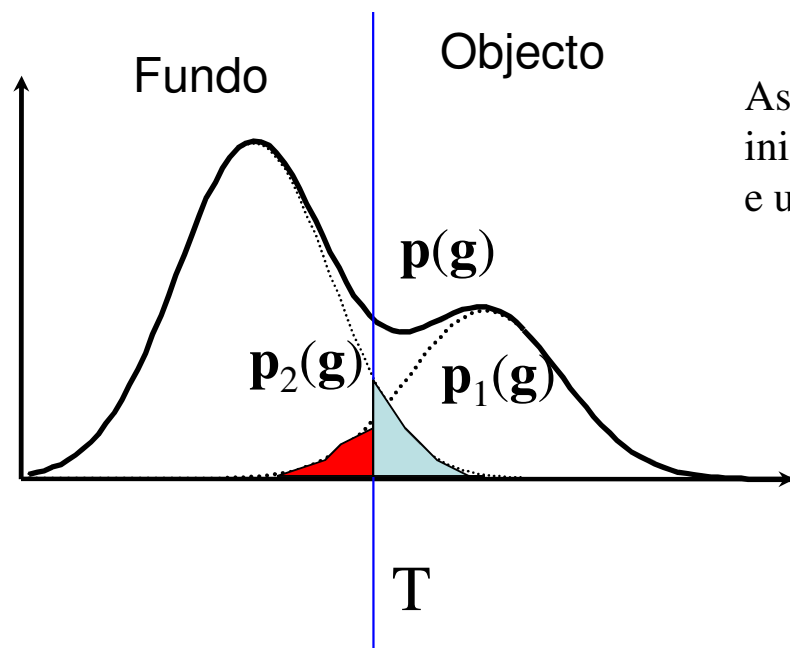
$$g_3 : H(g_3) = \min \{H(g)\}, g \in [g_1; g_2]$$

$$\text{If } \frac{H(g_2)}{H(g_3)} > m \text{ then } T = g_3$$

Aplique este método à imagem da impressão digital

# Binarização global simples

Seleção automática do nível óptimo de binarização (III)



Assume dois objectos com uma probabilidade inicial  $P_1, P_2$  e uma distribuição  $p_1(g), p_2(g)$

$$p(g) = P_1 p_1(g) + P_2 p_2(g)$$

$$P(\text{erro}) = P_1 \int_{-\infty}^T p_1(g) dg + P_2 \int_T^{+\infty} p_2(g) dg$$

$$\text{óptima} \quad \frac{dP(\text{erro})}{dT} = 0 \Rightarrow P_1 p_1(T) = P_2 p_2(T)$$

Exemplo para uma distribuição Gaussiana com variâncias iguais:

$$T = \frac{\mu_1 + \mu_2}{2} + \frac{\sigma^2}{\mu_1 - \mu_2} \ln \left( \frac{P_2}{P_1} \right)$$

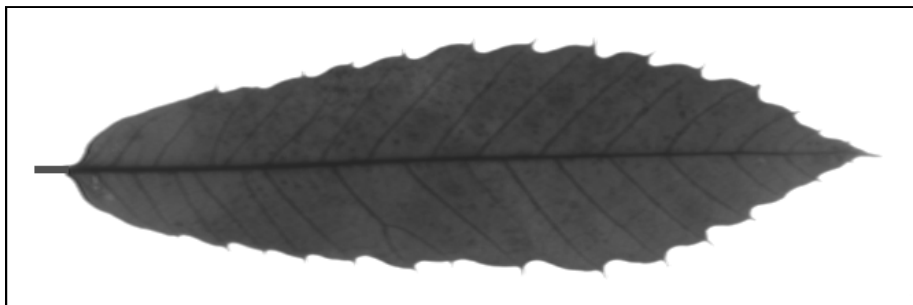


# Binarização global simples

Seleção automática do nível binarização com base no histograma acumulado (IV)

Este método baseia-se no conhecimento da percentagem de área ocupada pelo objecto na imagem. O nível de binarização é escolhido com base no histograma acumulado, procurando qual o nível de cinzento para o qual se verifica a relação conhecida entre a área da imagem e a área ocupada pelo objecto

Exemplo:



Problema :

Imagens adquiridas com diferentes níveis de iluminação da cena.

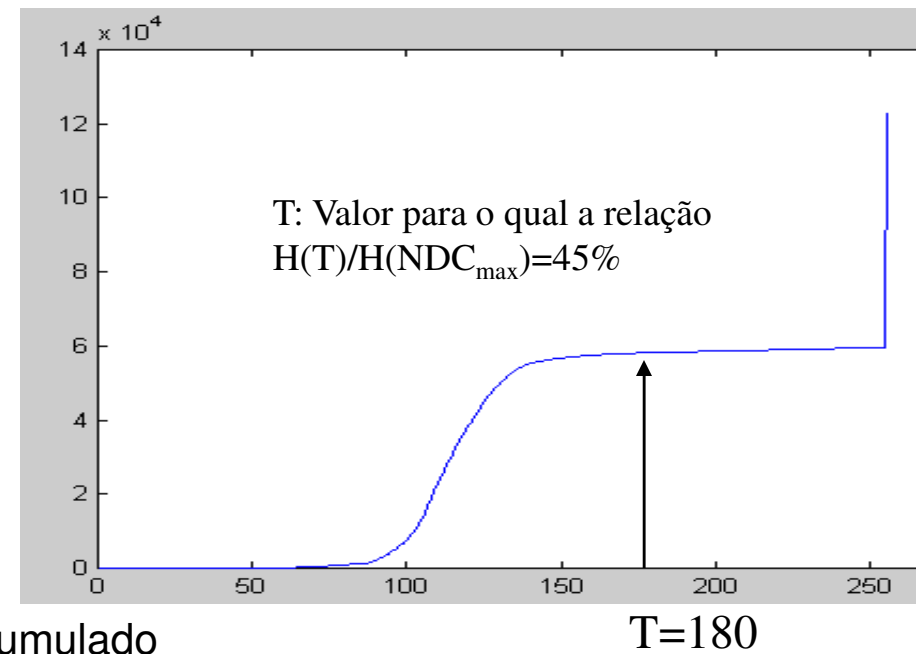
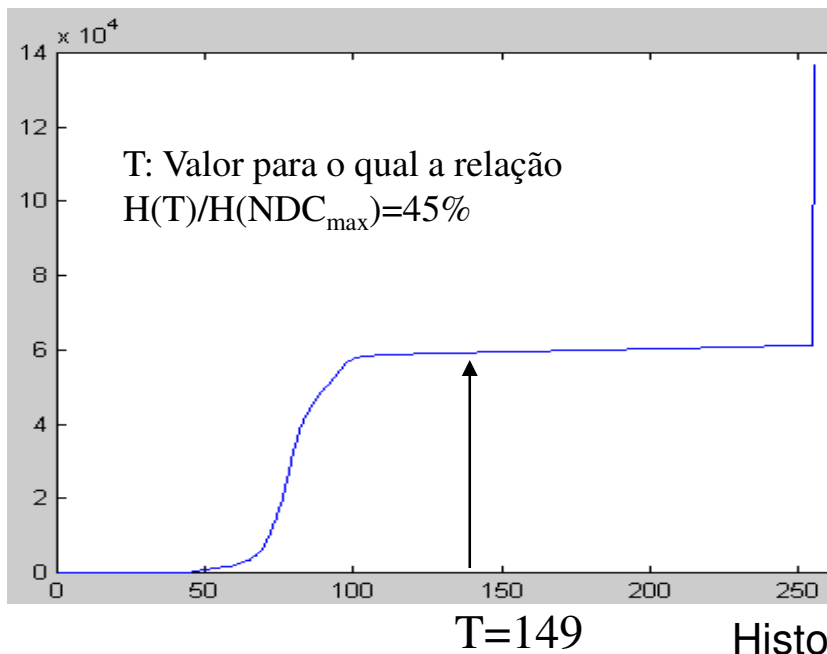
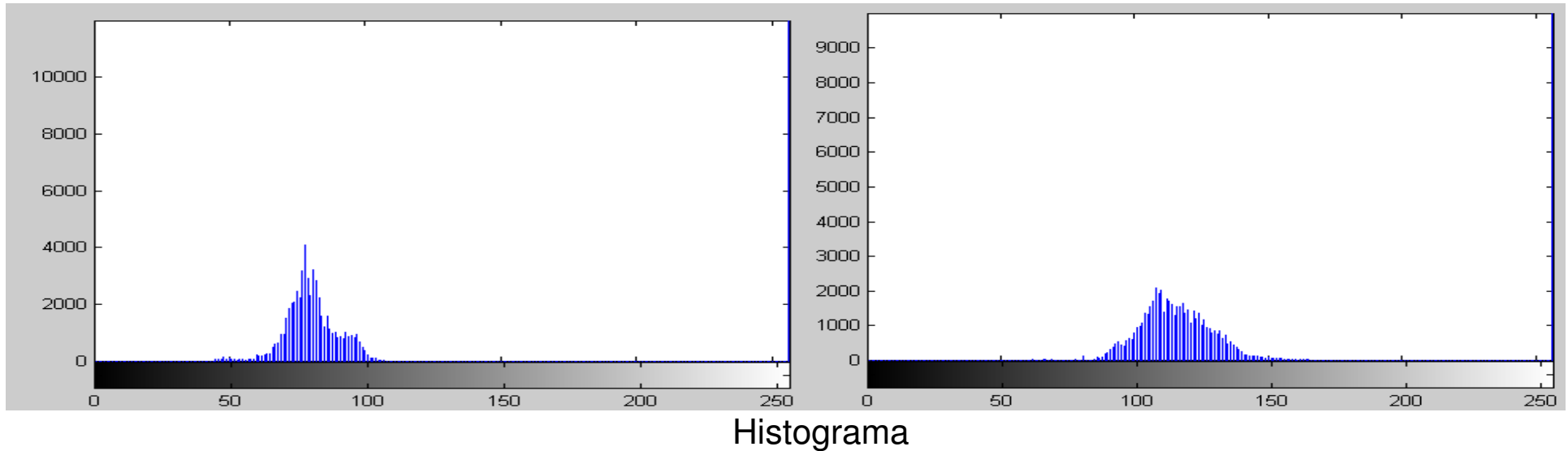
A escolha de um nível de binarização global fixo para todas as imagens não resulta.

Como encontrar o nível de binarização óptimo para cada imagem.

Na extremidade da folha aparecem NDC muito elevados (valores próximos do fundo)

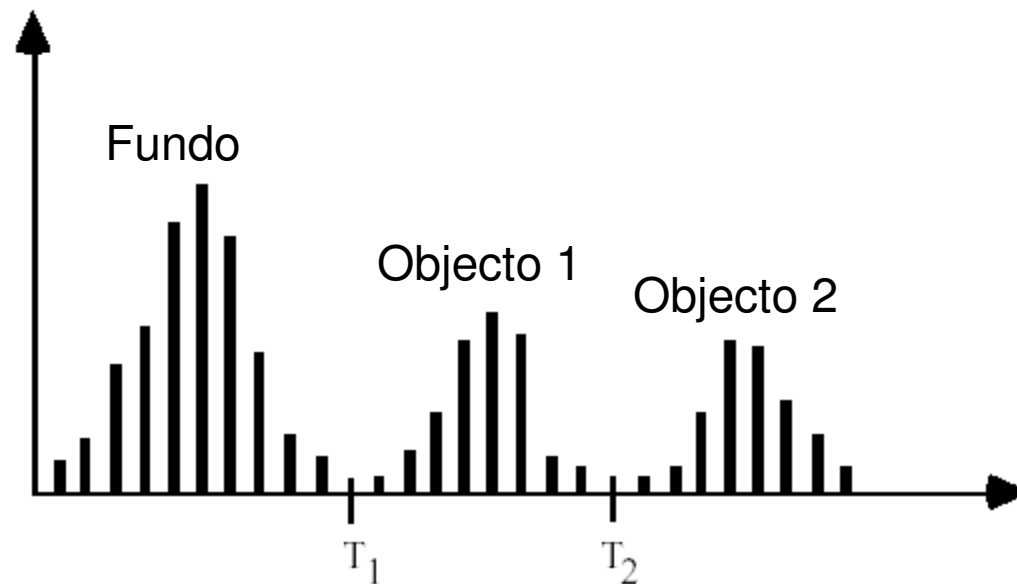
# Binarização global simples

Seleccção automática do nível binarização com base no histograma acumulado (IV)



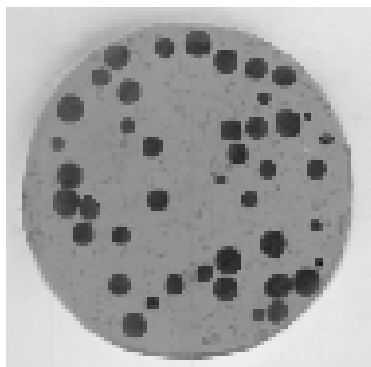
# Binarização global Multinível

- Quando se tem mais de duas regiões ( $n$  regiões) na imagem a binarização pode ser feita com  $n-1$  níveis de binarização.
- Há medida que o número de regiões aumenta, torna-se mais difícil distinguir os vários picos, dificultando a escolha dos níveis de binarização.



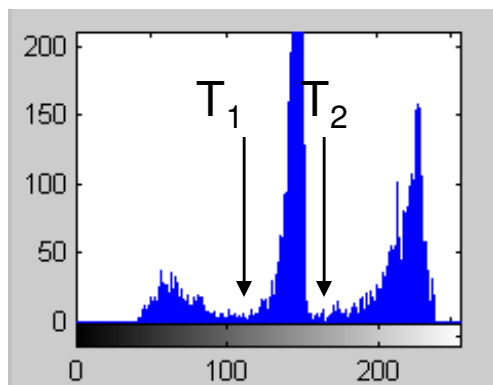
# Binarização global Multinível

- Exemplo

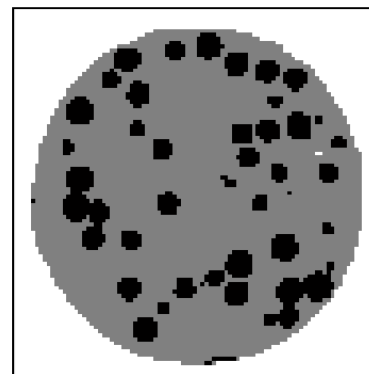


$$I_{out}(x, y) = \begin{cases} 0, & I_{in}(x, y) < T_1 \\ 128, & T_1 \leq I_{in}(x, y) \leq T_2 \\ 255, & I_{in}(x, y) > T_2 \end{cases}$$

Selecione os níveis de binarização  $T_1$  e  $T_2$  de modo a separar a imagem em três regiões

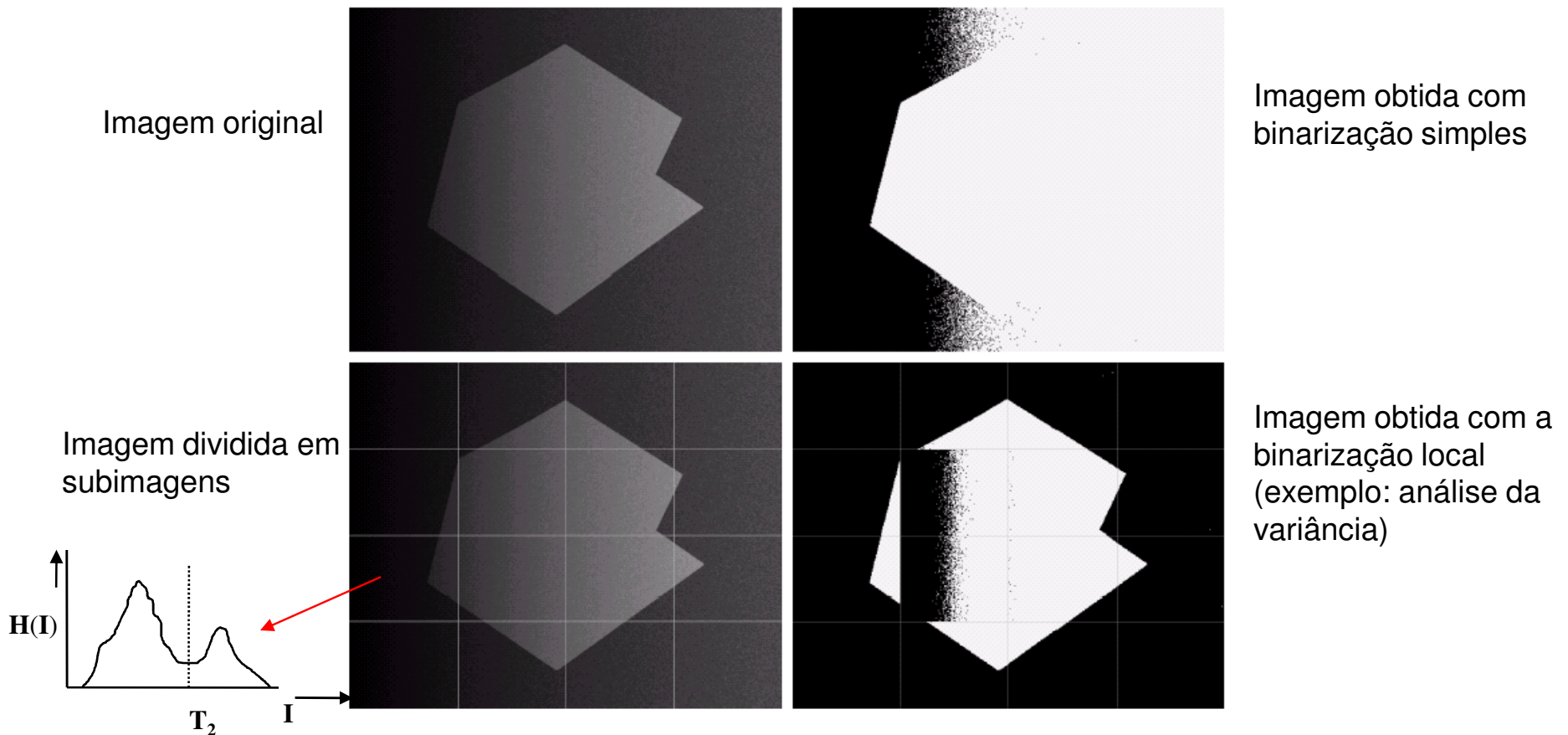


Resultado



# Binarização Local

- Divisão da imagem em várias subimagens;
- Binarização de cada subimagem com um nível de binarização próprio;
- Dificuldades: Encontrar a subdivisão óptima e estimar o nível de binarização em cada janela.

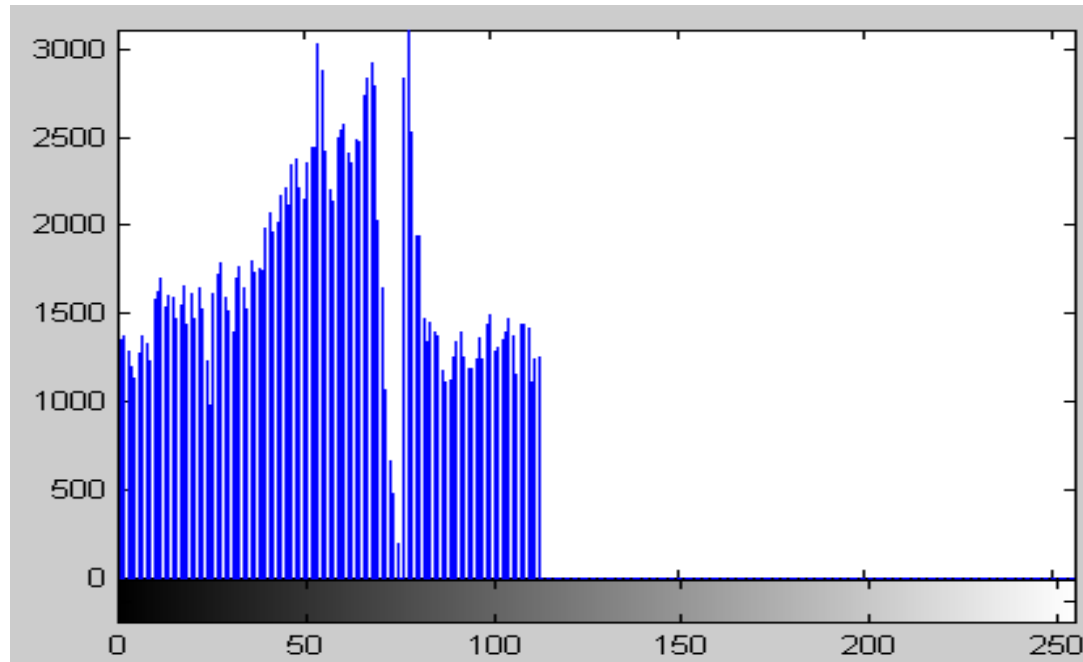


# Binarização Dinâmica ou Adaptativa

- Cada pixel da imagem é binarizado com um nível de binarização próprio que depende das características da sua vizinhança.
- Permite a binarização de uma imagem cujo histograma não possuí picos e vales distintos.

Exemplo:

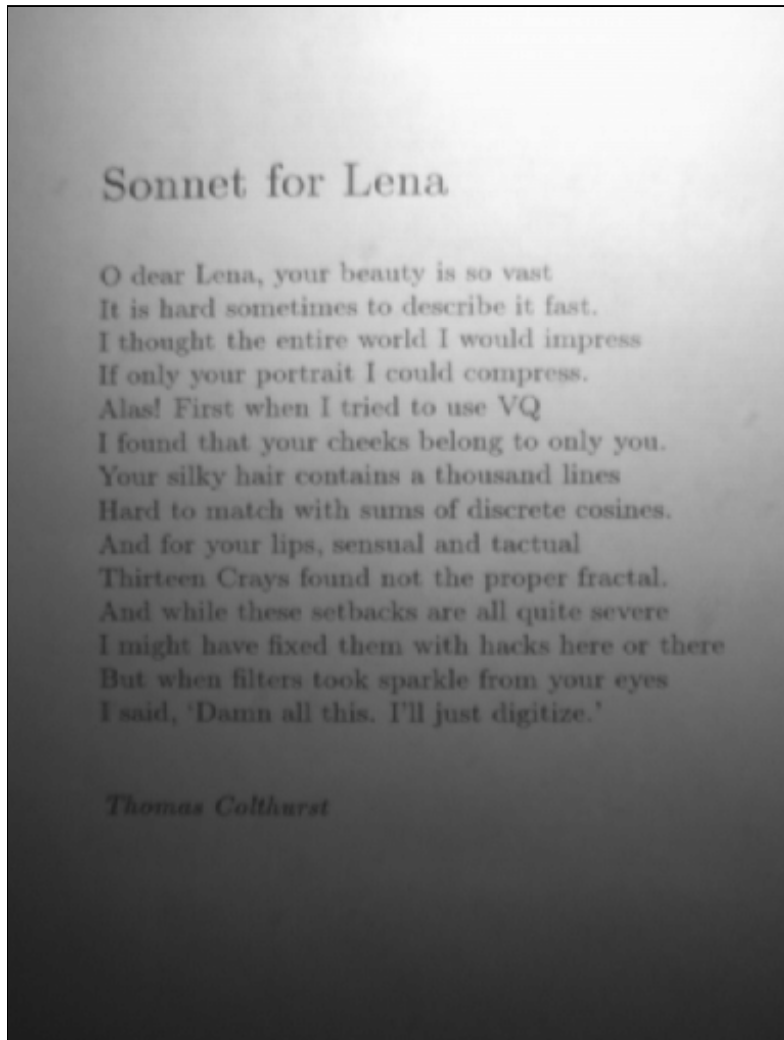
Segmentação de texto a partir de uma imagem que apresenta uma distribuição da iluminação não uniforme ( com forte gradiente de iluminação).



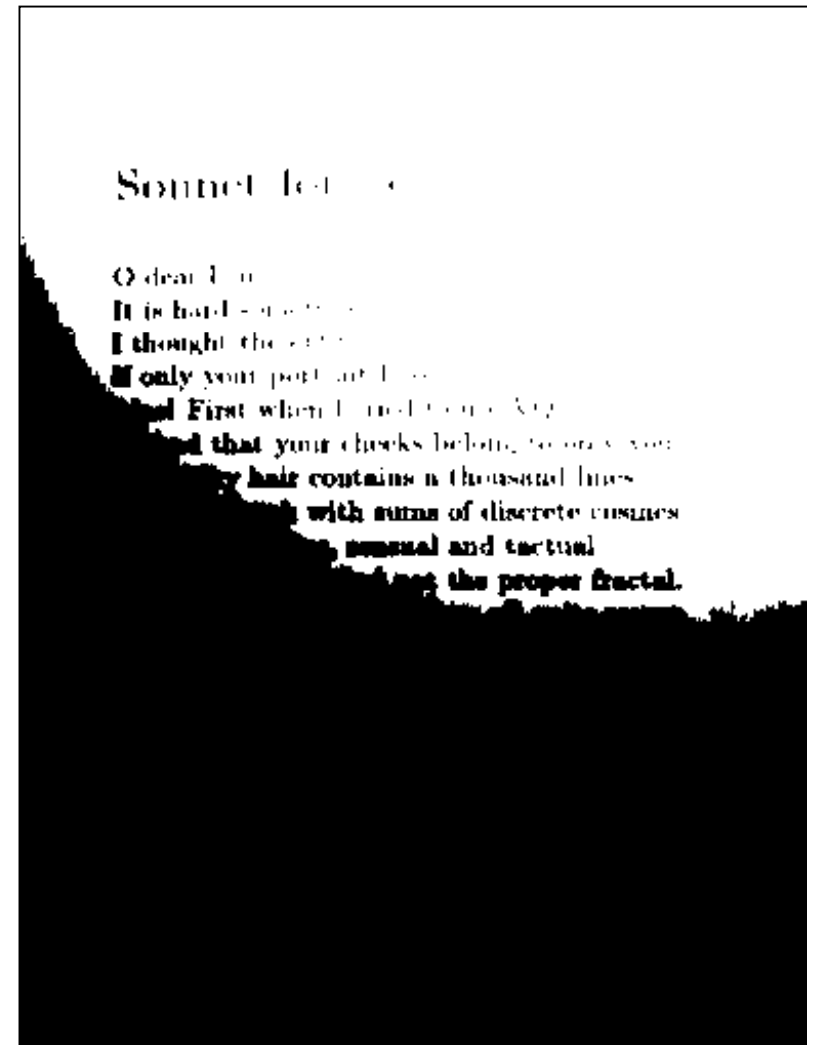
Histograma da imagem com texto

# Binarização Dinâmica ou Adaptativa

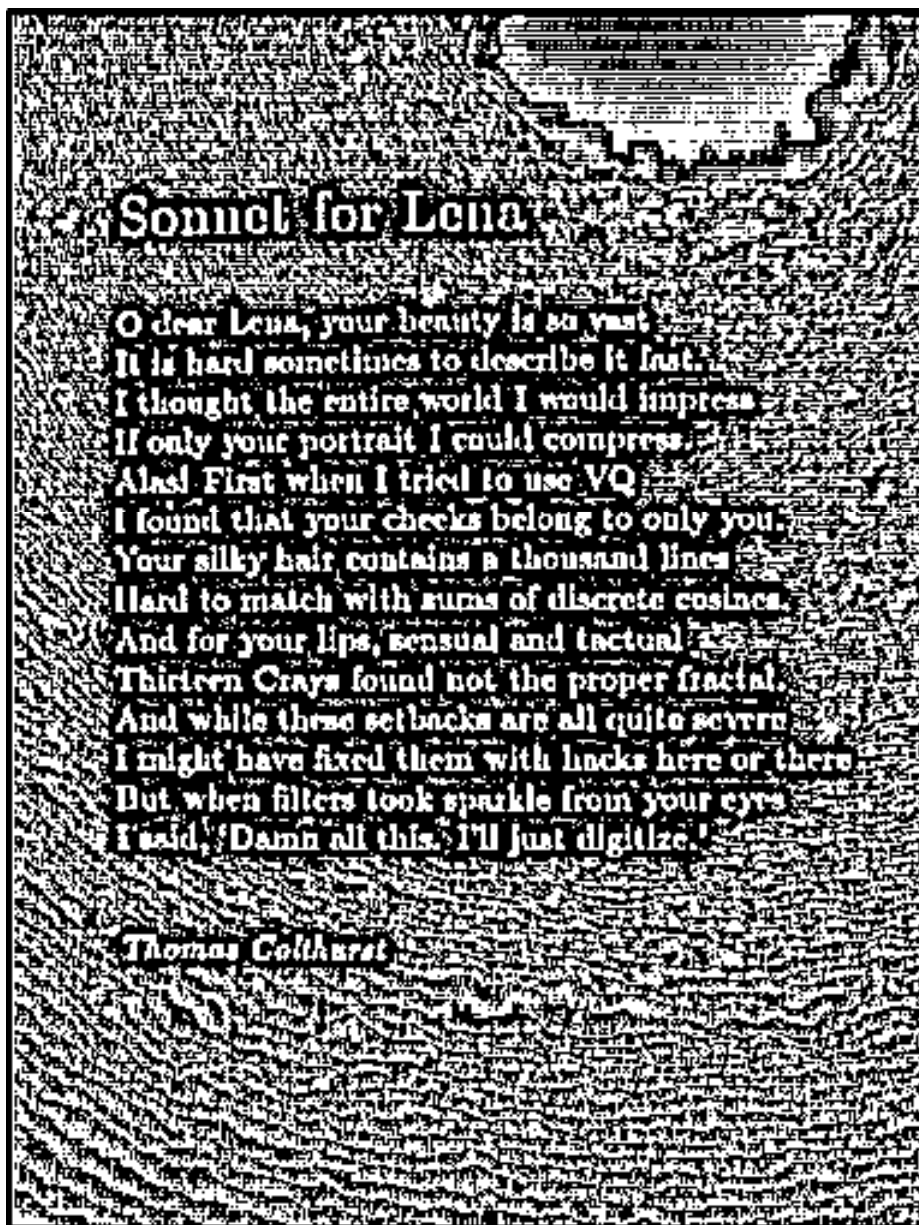
Imagem original



Resultado da binarização global



# Binarização Dinâmica ou Adaptativa



Resultado da binarização dinâmica com o nível de binarização igual ao valor médio numa vizinhança 7x7

Método eficaz na região onde existe texto, porque na vizinhança do pixel existem pixels que pertencem ao fundo e outros ao texto. O valor médio situa-se entre os valores das duas regiões e separa bem o fundo do texto. No entanto, nas margens, a média não é eficaz para valor de binarização, porque a gama de valores da vizinhança é muito estreita.



# Binarização Dinâmica ou Adaptativa

## Sonnet for Lena

O dear Lena, your beauty is so vast  
It is hard sometimes to describe it fast.  
I thought the entire world I would impress  
If only your portrait I could compress.  
Alas! First when I tried to use VQ  
I found that your cheeks belong to only you.  
Your silky hair contains a thousand lines  
Hard to match with sums of discrete cosines.  
And for your lips, sensual and tactual  
Thirteen Crays found not the proper fractal.  
And while these setbacks are all quite severe  
I might have fixed them with hacks here or there  
But when filters took sparkle from your eyes  
I said, 'Damn all this. I'll just digitize.'

*Thomas Culthurst*

Resultado da binarização dinâmica com o nível de binarização igual ao valor médio numa vizinhança 7x7 menos uma constante  $C=7$

Desta forma, os pixels localizados em regiões uniformes (margens) são atribuídos ao fundo.

# Binarização Dinâmica ou Adaptativa

## Sonnet for Lena

O dear Lena, your beauty is so vast  
It is hard sometimes to describe it fast.  
I thought the entire world I would impress  
If only your portrait I could compress.  
Alas! First when I tried to use VQ  
I found that your cheeks belong to only you.  
Your silky hair contains a thousand lines  
Hard to match with sums of discrete cosines.  
And for your lips, sensual and textual  
Thirteen Crays found not the proper fractal.  
And while these setbacks are all quite severe  
I might have fixed them with hacks here or there  
But when filters took sparkle from your eyes  
I said, 'Damn all this. I'll just digitize.'

*Thomas Calkhurst*

Resultado da binarização dinâmica com o nível de binarização igual ao valor médio numa vizinhança 75x75 menos uma constante  $C=10$

Adoptado uma janela de maiores dimensões o resultado final é pior, porque é afectado pela variação da iluminação do fundo. Além disso, o uso da janela 75x75 é computacionalmente mais pesado.

# Binarização Dinâmica ou Adaptativa

## Algoritmo

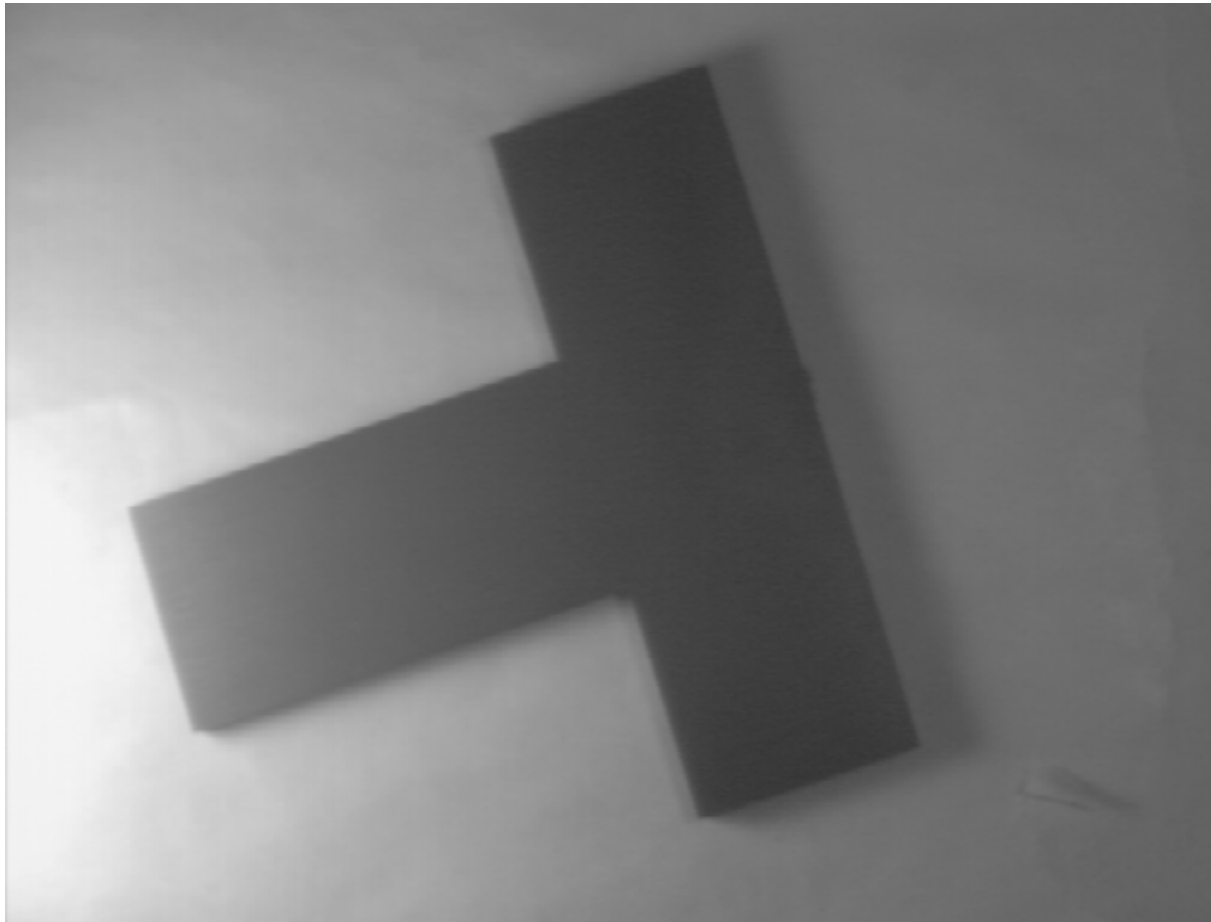
1. Convolução da imagem com um filtro de média ou com um filtro de mediana.
2. Subtração do resultado da convolução pela imagem original
3. Binarização da imagem diferença com o nível  $C$ .
4. Negativo da imagem resultado.

# Binarização Dinâmica ou Adaptativa

Experimente este método na imagem apresentada.

7x7,  $C = 4$

140x140,  $c=8$



# **Crescimento de regiões – Agregação de pixels**

- Método que agrupa pixels ou sub-regiões em regiões maiores.
  - 1) Partindo de um conjunto de pontos iniciais ('sementes').
  - 2) Crescimento das regiões através do agrupamento dos pixels vizinhos de cada pixel semente com base num critério de similaridade (ex:NDC)

# Crescimento de regiões – Agregação de pixels

## Exemplo

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

2 'sementes'

Critério de similaridade

$$|z_i - s_j| < T$$

A	A	B	B	B
A	A	B	B	B
A	A	B	B	B
A	A	B	B	B
A	A	B	B	B

T=3

A	A	A	A	A
A	A	A	A	A
A	A	A	A	A
A	A	A	A	A
A	A	A	A	A

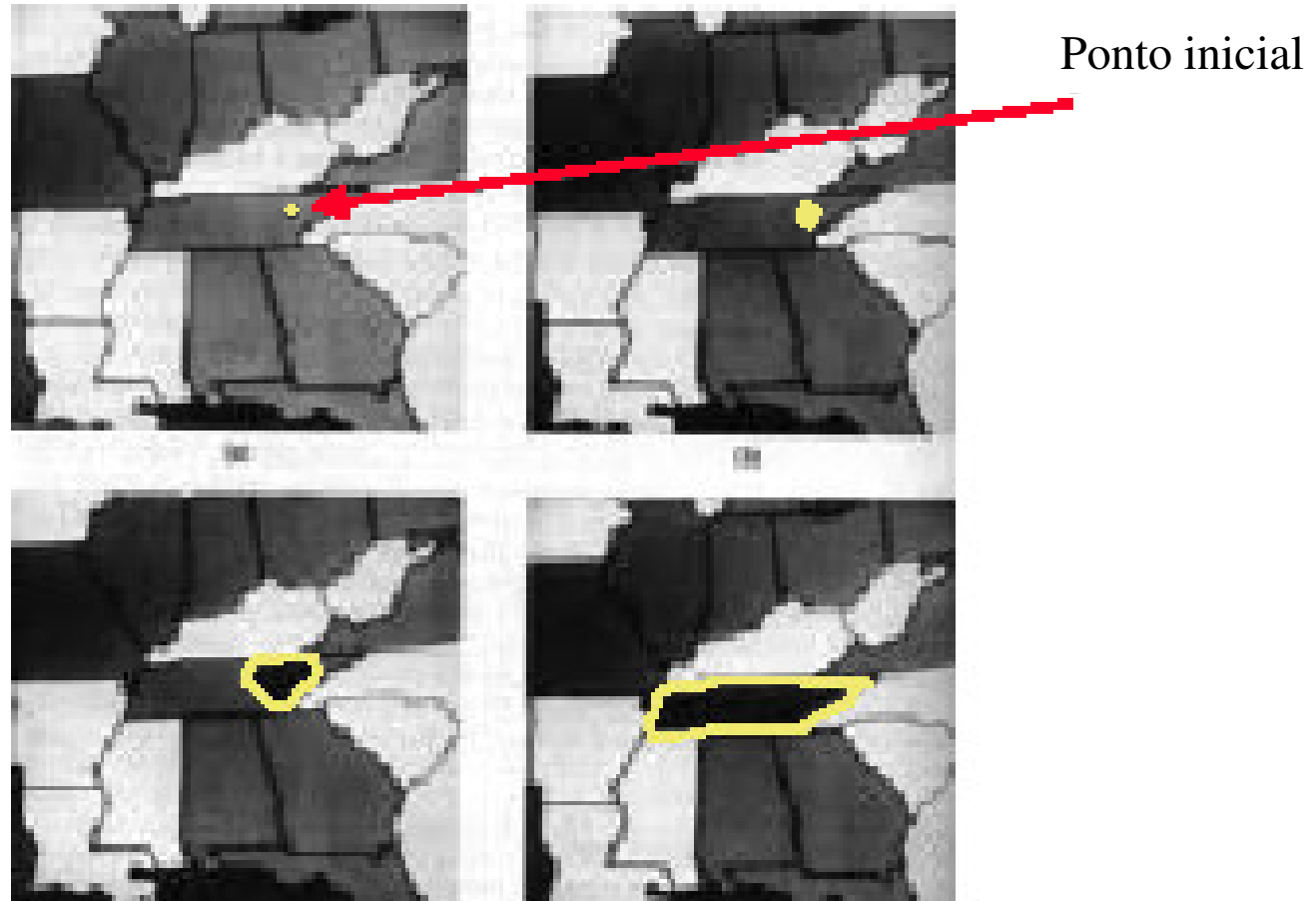
T=8

# **Crescimento de regiões – Agregação de pixels**

- **Dificuldades**
  - Escolha de pontos iniciais apropriados
  - Definição do critério de similaridade
  - Definição do  $n^o$  de regiões.
  - Criação automática de novas regiões.

# Crescimento de regiões – Agregação de píxeis

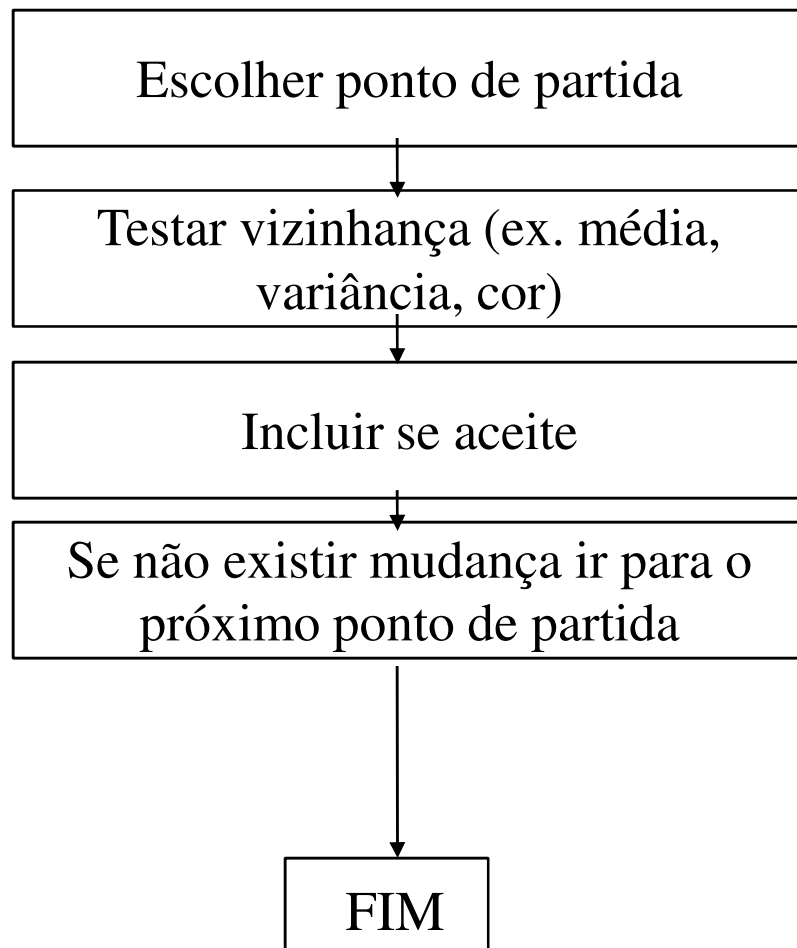
Exemplo:



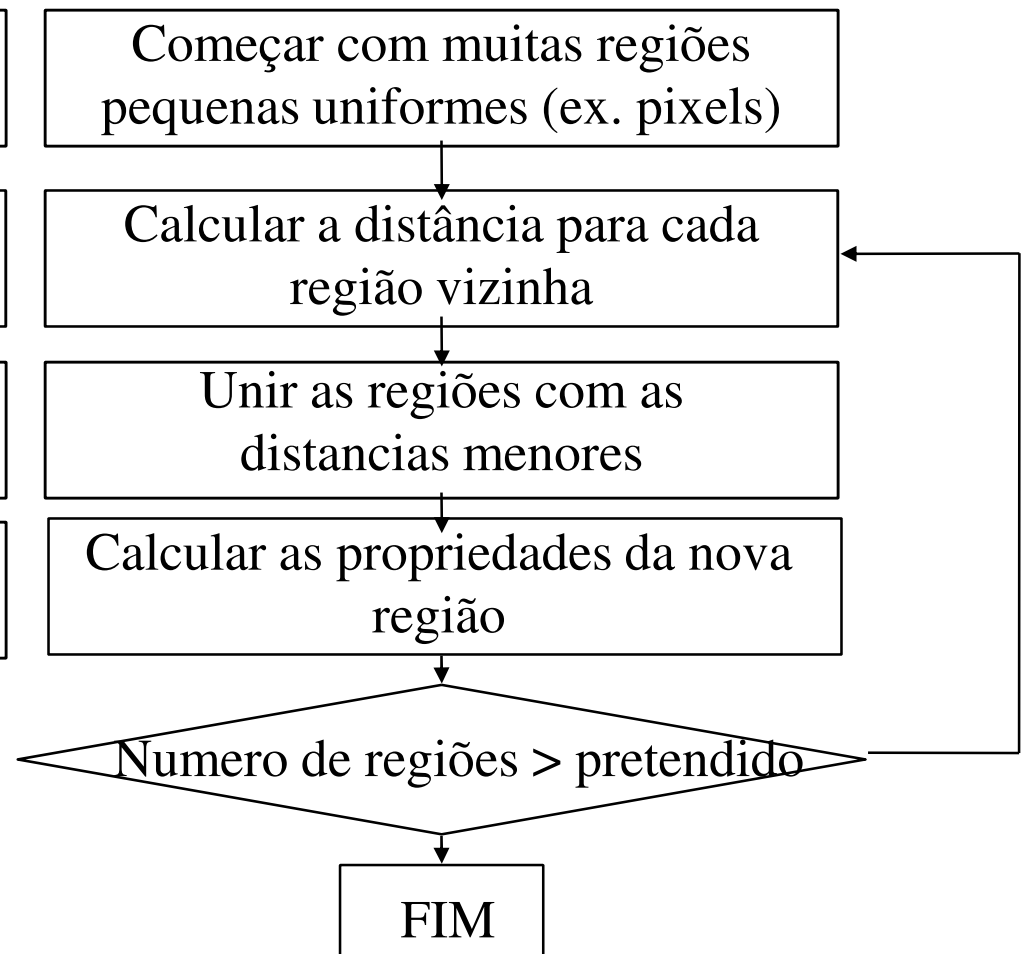


# Crescimento de regiões

Numero de regiões variável



Numero de regiões pré-definido



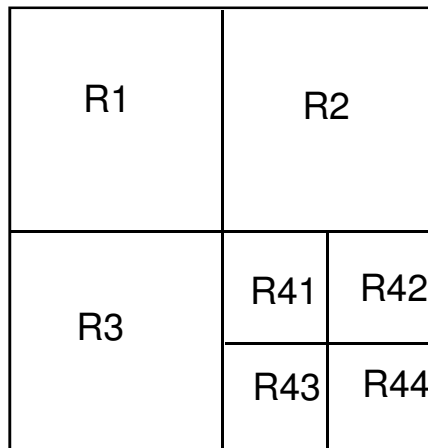
# Divisão e Reunião de regiões

- Inicialmente, subdivide-se a imagem num conjunto arbitrário de regiões disjuntas;
- Em seguida
- REUNIÃO e/ou DIVISÃO das regiões numa tentativa de satisfazer a condição  $P$  (ex: variância)

**Ex : quadtree**

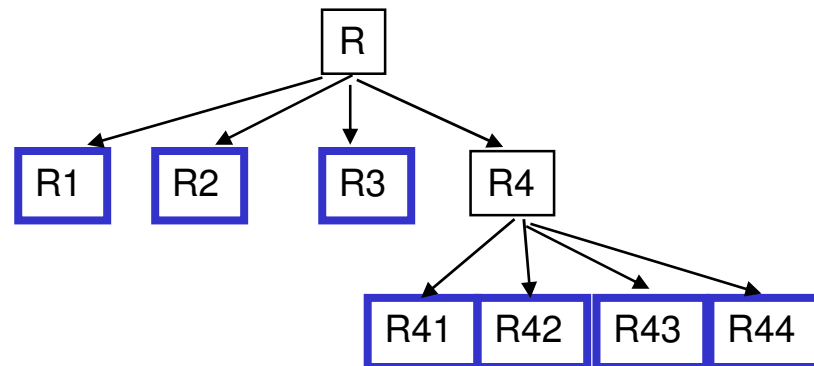
# Divisão e reunião de regiões - *Quadtree*

- Divisão da região  $R_i$  em 4 quadrantes disjuntos sempre que  $P(R_i) = \text{Falso}$   $\rightarrow$  i.e., a condição  $P$  não se verifica na região  $R_i$
- Juntar quaisquer duas regiões  $R_j$  and  $R_k$  para as quais se verifica  $P(R_j \cup R_k) = \text{Verdadeiro}$   $\rightarrow$  i.e., a condição  $P$  verifica-se na região obtida pela reunião das 2 regiões.
- Parar quando já não é possível realizar mais divisões nem reuniões



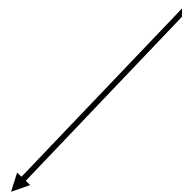
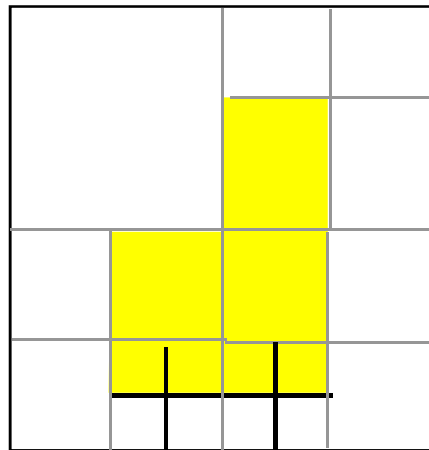
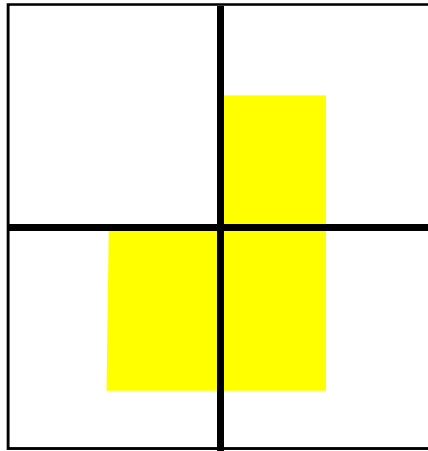
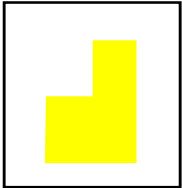
Divisão em partes

Estrutura em árvore quaternária da regiões obtidas quadtree

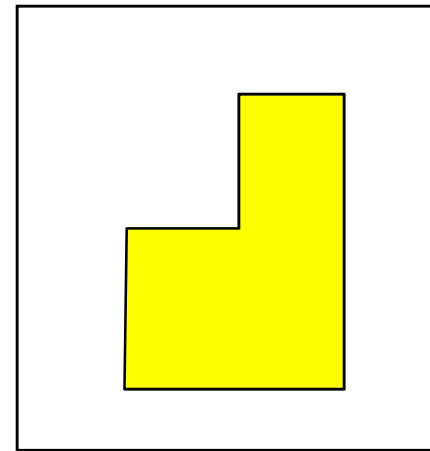
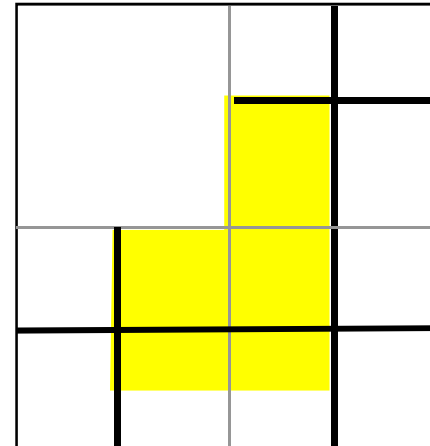


# Exemplo

Imagem inicial



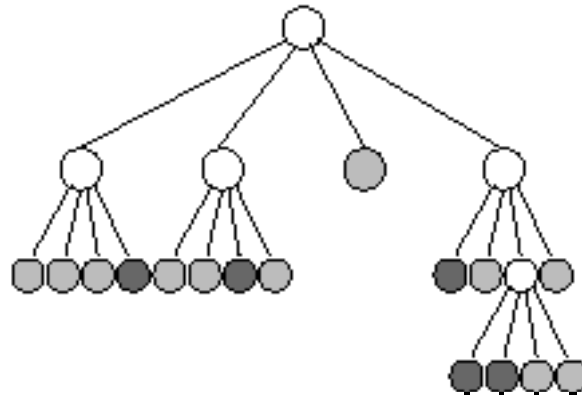
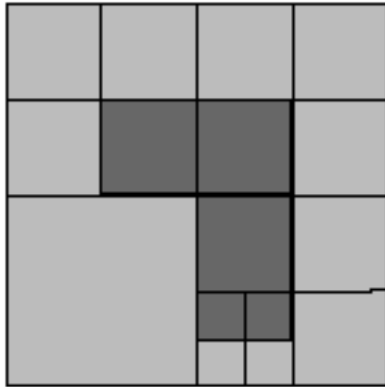
4 passos



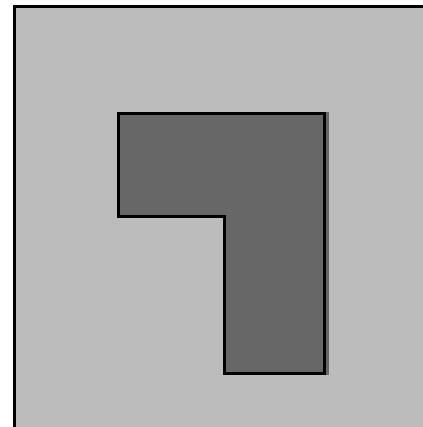
Resultado final

# Exemplo

Passo 1: Divisão



Passo2: Reunião



# Exemplo

- Definindo  $P(R_i)=\text{VERDADE}$  se pelo menos 80% dos pixels de  $R_i$  respeitam a condição:

$$|Z_j - \bar{m}_i| \leq 2\sigma_i$$

Se  $P(R_i)=\text{VERDADE}$  , todos os pixels da região  $R_i$  ficam com o valor  $\bar{m}_i$

- Dividindo e reunindo conforme o algoritmo apresentado atrás obtém o resultado apresentado.

Imagem original



Binarização por  
divisão/reunião



Binarização  
global



# Detecção de pontos isolados

- Usado para a detecção de pontos isolados na imagem que correspondam a um pico ou um vale em termos de NDC.
- O NDC de um ponto isolado é bastante diferente do valor dos seus vizinhos.
- A detecção pode ser feita com a seguinte máscara 3x3:

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

- O valor da operação local é binarizado.
- Um ponto isolado é detectado se

$$|8I_5 - (I_1 + I_2 + I_3 + I_4 + I_6 + I_7 + I_8 + I_9)| > T$$

T é um pré-determinado valor não negativo.

# Detecção de pontos isolados

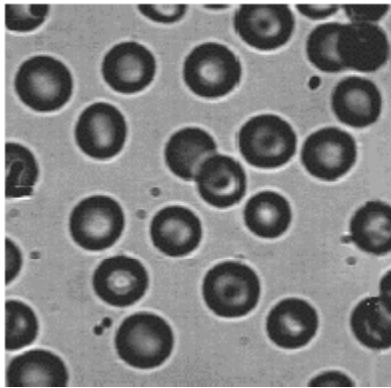


Imagem original

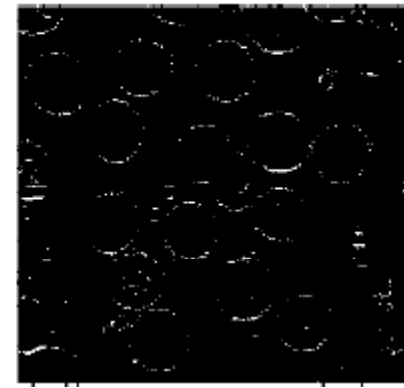
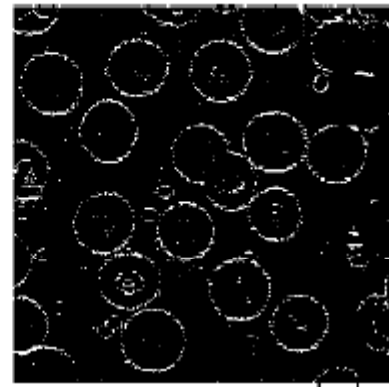
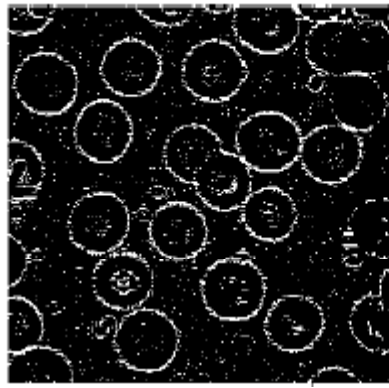


Imagem resultante da segmentação para diferentes valores de  $T$



# Detecção de linhas

- Usada na detecção de linhas na imagem.
- A detecção pode ser feita com quatro máscaras 3x3:

$$\mathcal{D}_{0^\circ} = \begin{bmatrix} -1 & -1 & -1 \\ 2 & 2 & 2 \\ -1 & -1 & -1 \end{bmatrix} \quad \mathcal{D}_{45^\circ} = \begin{bmatrix} -1 & -1 & 2 \\ -1 & 2 & -1 \\ 2 & -1 & -1 \end{bmatrix} \quad \mathcal{D}_{90^\circ} = \begin{bmatrix} -1 & 2 & -1 \\ -1 & 2 & -1 \\ -1 & 2 & -1 \end{bmatrix} \quad \mathcal{D}_{135^\circ} = \begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix}$$

Detecta linhas horizontais

Detecta linhas inclinadas a 45°

Detecta linhas verticais

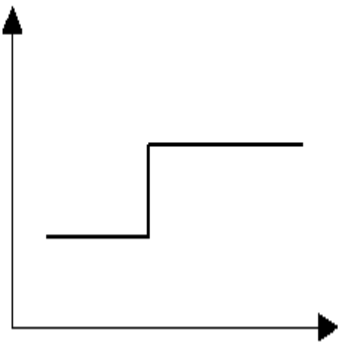
Detecta linhas inclinadas a 135°

Seja  $R_{0^\circ}$ ,  $R_{45^\circ}$ ,  $R_{90^\circ}$ ,  $R_{135^\circ}$ , respectivamente a resposta das máscaras  $\mathcal{D}_{0^\circ}$ ,  $\mathcal{D}_{45^\circ}$ ,  $\mathcal{D}_{90^\circ}$ ,  $\mathcal{D}_{135^\circ}$ .

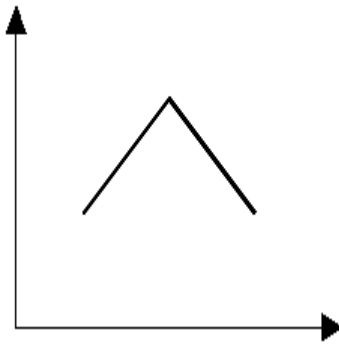
Num determinado pixel  $(m,n)$ , se  $R_{135^\circ}$  é o máximo entre as respostas  $\{R_{0^\circ}, R_{45^\circ}, R_{90^\circ}, R_{135^\circ}\}$ , diz-se que sobre o pixel é provável que passe uma recta com uma inclinação de 135°.

# Detecção de arestas

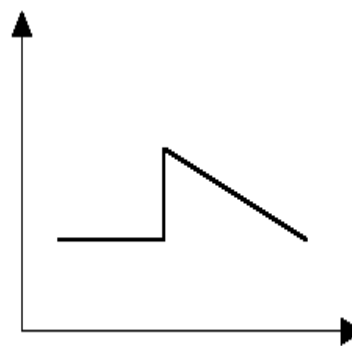
- Em imagens reais é pouco frequente a ocorrência de pontos isolados e de linhas finas.
- Na segmentação da imagem está-se mais interessado em detectar as fronteiras entre as regiões em que as propriedades dos seus NDC são distintas.
- É assumido que cada região apresenta características homogêneas de forma a que a transição entre as regiões pode ser detectada com base na descontinuidades dos NDC.
- Uma aresta numa imagem pode ser definida como uma descontinuidade ou uma variação abrupta dos NDC.



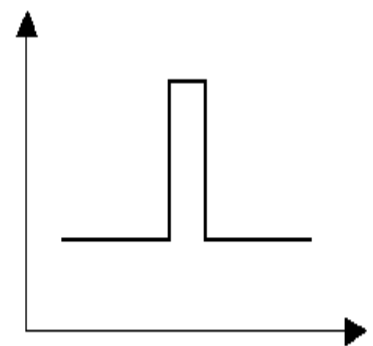
Aresta em degrau



Aresta em rampa



Combinação da aresta em rampa e degrau

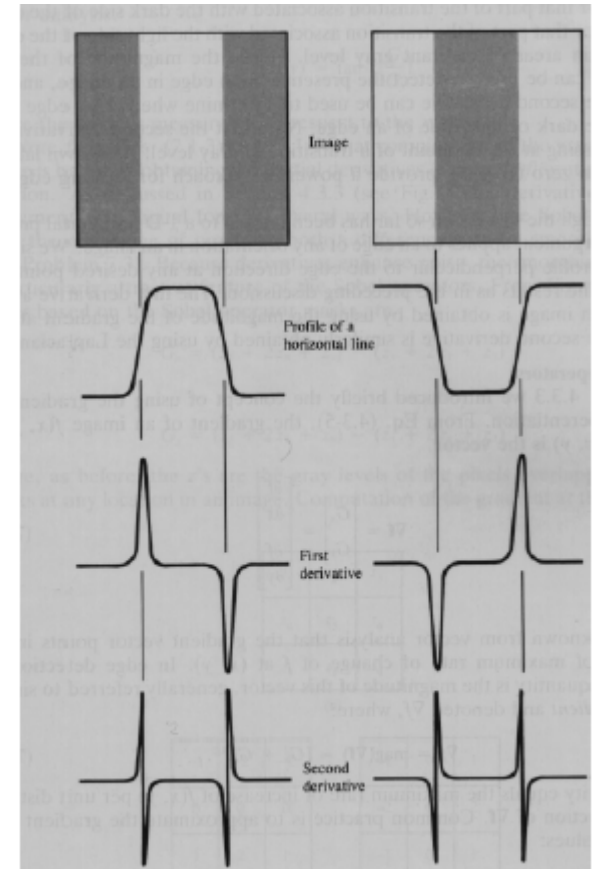


Aresta em pico

# Detecção de arestas

- Estas são arestas ideais que não ocorrem frequentemente em imagens de cenas reais. Além disso em 2D as arestas podem ter uma orientação qualquer.
- Nas arestas não correspondem a descontinuidades perfeitas. Assim, a tarefa de detectar as arestas de uma imagem é muita mais complexa do que aparenta ser.
- Uma ferramenta matemática útil para o desenvolvimento de detectores de arestas são os operadores de primeira e segunda derivada.

- A partir da imagem torna-se evidente que a magnitude da primeira derivada pode ser usada na detecção de arestas da imagem.
- O sinal da segunda derivada pode ser usada para determinar se um pixel pertence à parte mais escura ou mais clara da aresta.
- O cruzamento por zero da segunda derivada localizar a aresta na imagem.



# Detecção de arestas

## Operadores baseados na 1ª derivada - Gradiente

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

**Magnitude:** 
$$\begin{aligned} \nabla f &= \text{mag}(\nabla \mathbf{f}) \\ &= [G_x^2 + G_y^2]^{1/2} \\ &= \left[ \left( \frac{\partial f}{\partial x} \right)^2 + \left( \frac{\partial f}{\partial y} \right)^2 \right]^{1/2} \end{aligned}$$

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

# Detecção de arestas

**Aproximações  
Discretas:**

-1	0	0	-1
0	1	1	0

Roberts

-1	-1	-1	-1	0	1
0	0	0	-1	0	1
1	1	1	-1	0	1

Prewitt

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

Sobel

# Detecção de arestas - Gradiente



Imagem original



$|G_x|$



$|G_y|$



$|G_x| + |G_y|$

# Detecção de arestas

## Operadores baseados na 2ª derivada - Laplaciano

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

**Aproximação Discreta:**

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

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

---

$$\nabla^2 f = 8z_5 - (z_1 + z_2 + z_3 + z_4 + z_6 + z_7 + z_8 + z_9)$$

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

# Detecção de arestas

## Operadores baseados na 2ª derivada - Laplaciano

Na prática, na sua versão original, não é usado para a detecção de arestas, porque...

- (1) Muito sensível ao ruído.
- (2) A magnitude do operador produz saídas de “dupla aresta”.
- (3) Não permite a detecção da orientação da aresta

O seu papel na segmentação é:

- (1) A propriedade do cruzamento por zero permite localizar a aresta
- (2) Objectivo complementar de indicação se um pixel pertence à parte clara ou escura da aresta.

O Laplaciano normalmente é combinado com o filtro de suavização Gaussiano para encontrar as arestas através do cruzamento por zero. Este operador designa-se por:

**LOG** (*Laplacian of Gaussian*)

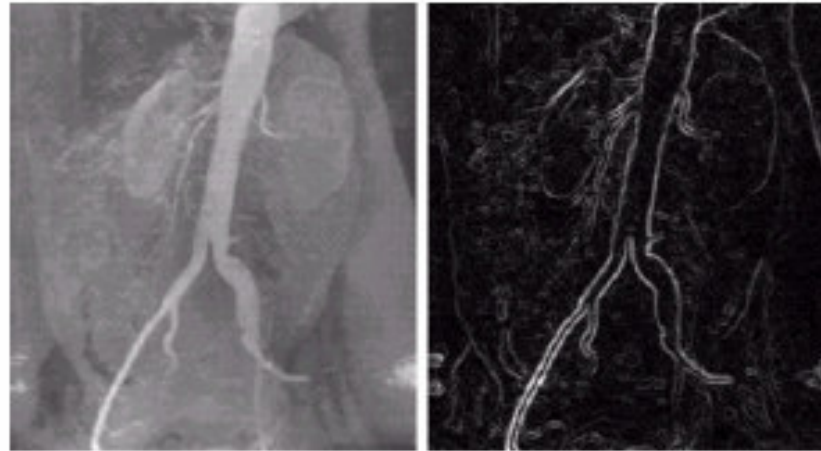
0	0	-1	0	0
0	-1	-2	-1	0
-1	-2	16	-2	-1
0	-1	-2	-1	0
0	0	-1	0	0

Máscara de um operador LOG



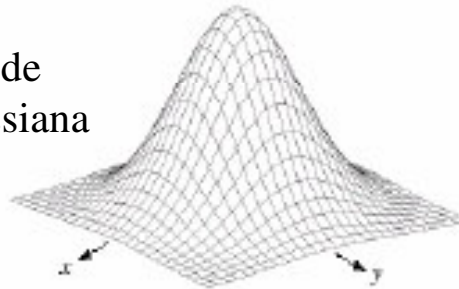
# Detecção de arestas - LOG

Imagem original



Resultado obtido com o operador Sobel (Gradiente)

Função espacial de suavização Gaussiana



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

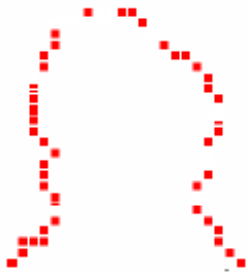
Máscara do operador Laplaciano

Resultados: do LOG (esquerda), do LOG binarizado (centro) e do Cruzamento por zero (direita)

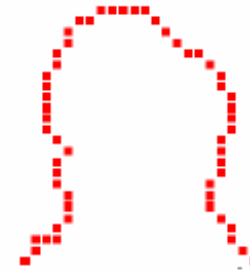


# Ligação de contornos – *Edge linking*

- A saída dos operadores que realizam a segmentação com base na detecção de descontinuidades, raramente conseguem caracterizar completamente o contorno dos objectos presentes na cena (contornos quebrados).
- Este problema é causado pelo ruído, quebras do contorno e outros efeitos indesejáveis.
- Para solucionar este problema, sobre o resultado da detecção de arestas binarizada são aplicados algoritmos realizar a ligação do contorno.
- Duas abordagens possíveis: Processamento Local e Processamento Global.



Contorno quebrado



Contorno ligado

# Ligação de contornos – Processamento Local

- Análise da vizinhança de cada pixel  $(x,y)$ , na imagem resultante após a aplicação do detector de arestas.
- Todos os pixels semelhantes são ligados, formando um contorno (boundary) de pixels.
- Podem ser usados duas propriedades para estabelecer o critério de semelhança:
  - A magnitude da resposta do gradiente;
  - A direcção do gradiente.

- O pixel com coordenadas  $(x_0,y_0)$  pertencente à vizinhança de  $(x, y)$  é semelhante na magnitude ao pixel  $(x, y)$  se:

$$|\nabla f(x, y) - \nabla f(x_0, y_0)| \leq E$$

- O pixel com coordenadas  $(x_0,y_0)$  pertencente à vizinhança de  $(x, y)$  é semelhante na direcção do ângulo do pixel  $(x, y)$  se

$$|\alpha(x, y) - \alpha(x_0, y_0)| < A$$

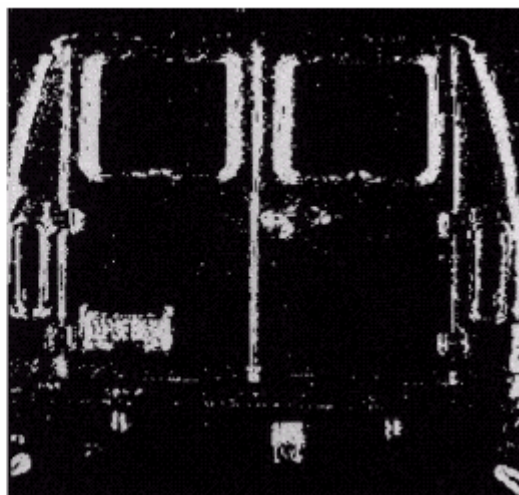
- O pixel  $(x_0,y_0)$  é ligado ao pixel  $(x, y)$  se ambas as condições se verificam.
- Método simples.
- Sensível ao ruído.
- De difícil aplicação em imagens complexas.

# Ligação de contornos – *Edge linking*

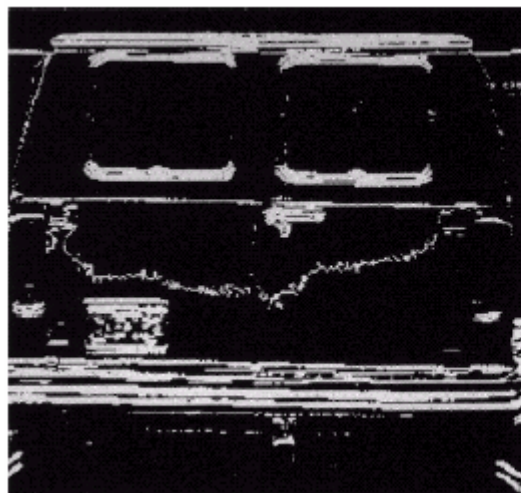
Imagem original



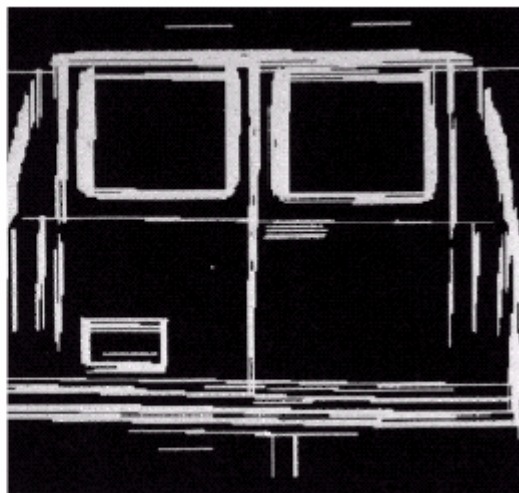
$G_x$



$G_y$



Resultado obtido após  
ligação dos contornos



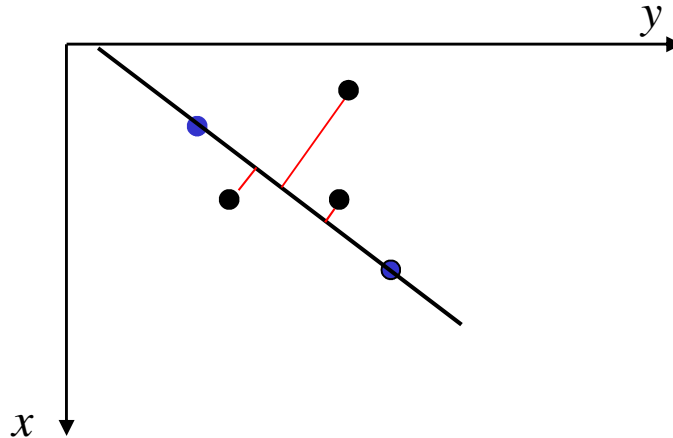
# Ligação de contornos – Processamento Global

- A outra técnica para a ligação de contornos consiste em determinar se os pixels encontrados pelo detector de arestas “assentam” sobre uma curva com uma determinada forma (recta, círculo, etc.)
- Técnica aplicável nos casos em que se tem pouco conhecimento acerca da localização do contorno, cuja forma é bem descrita por uma curva paramétrica.
- Técnica robusta ao ruído e a falhas nas arestas.

## Método da “força bruta”

- Considere-se uma imagem com  $n$  pontos candidatos a pertencerem a rectas presentes na imagem.
- O segmento de recta entre cada par de pontos é considerada.

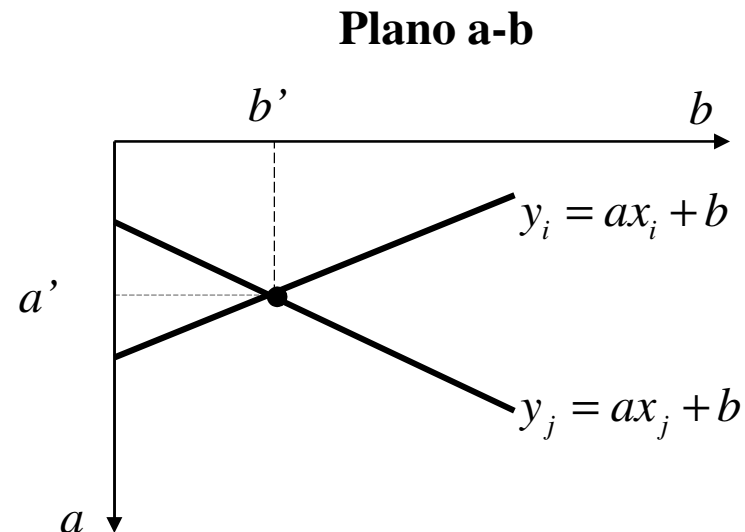
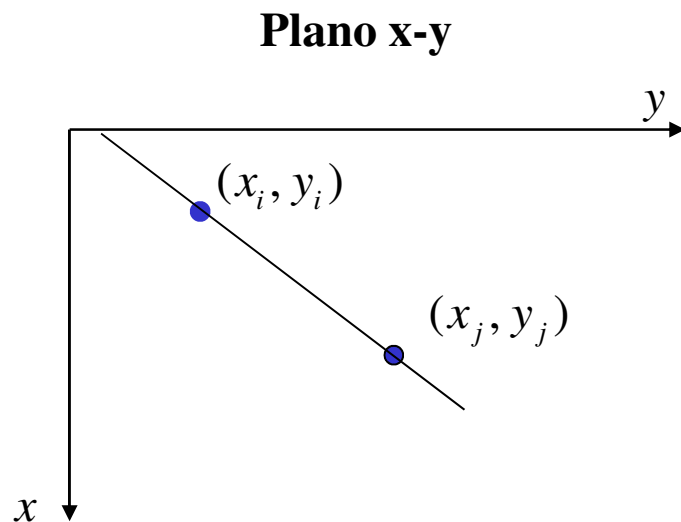
# Ligação de contornos – Processamento Global



- Calcula-se a distância entre cada ponto candidato e a recta em questão.
  - Quando o píxel está a uma curta distância da recta considera-se que ele faz parte dela.
  - Quando o píxel está a uma curta distância da recta considera-se que ele faz parte dela.
  - N° de linhas:  $n^2$
  - N° de distância a calcular:  $n^3$
- } Método Computacionalmente pesado

# Ligação de contornos – Transformada de Hough

- Método alternativo: Transformada de Hough.
- $(x_i, y_i)$  : Consideram-se todas as rectas possíveis que passam por este ponto:  $y_i = ax_i + b$
- A equação  $b = -ax_i + y_i$  representa uma linha única no plano  $a$ - $b$  que passa no ponto  $(x_i, y_i)$
- Um outro ponto  $(x_j, y_j)$  tem também uma linha única no plano  $a$ - $b$ :  $y_j = ax_j + b$
- Considere-se que as duas rectas se intersectam no ponto  $(a', b')$ , então a equação:  $y = -a'x + b'$  representa a recta do plano  $xy$  que passa sobre os dois pontos.

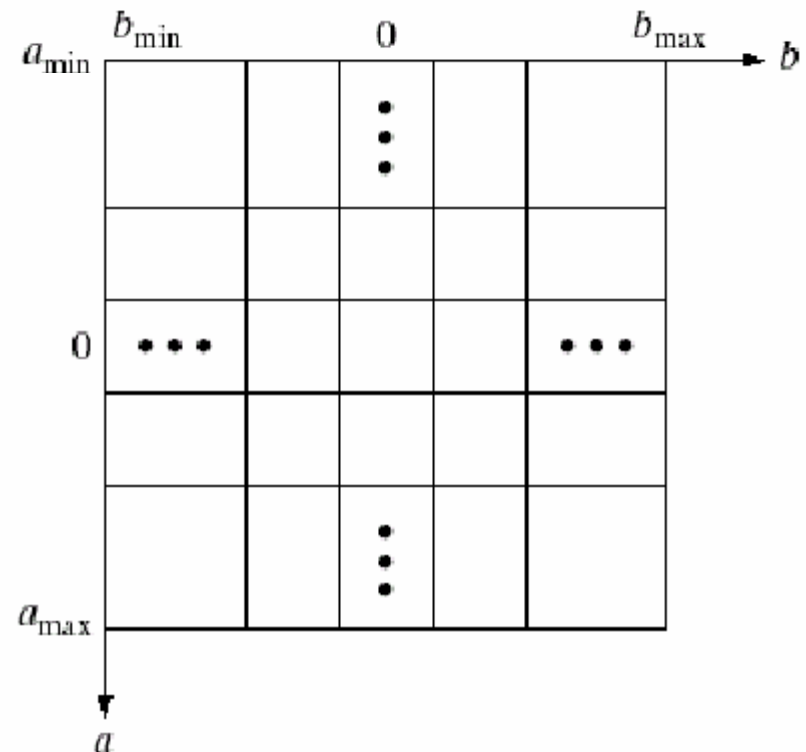


# Transformada de Hough

- O espaço paramétrico  $ab$  (espaço acumulador) é dividido num número finito de células acumuladoras.
- $(a_{\min}, a_{\max})$ : Gama de valores possíveis para os declives,  $a$ .
- $(b_{\min}, b_{\max})$ : Gama de valores possíveis para os pontos de intercessão com o eixo,  $b$ .
- $A(i,j)$ : Número de pontos na célula de coordenadas  $(i,j)$ .

## Algoritmo

1. Colocar a zero todas as células do plano acumulador
2. Para cada ponto  $(x_i, y_i)$ 
  - 2.1. Para cada valor de  $a$
  - 2.2. Calcular  $b = -ax_i + y_i$
  - 2.3. Arredondar o valor de  $b$  para o valor mais próximo existente na tabela.
  - 2.4. Incrementar a célula do espaço acumulador  $(a,b)$  com +1.



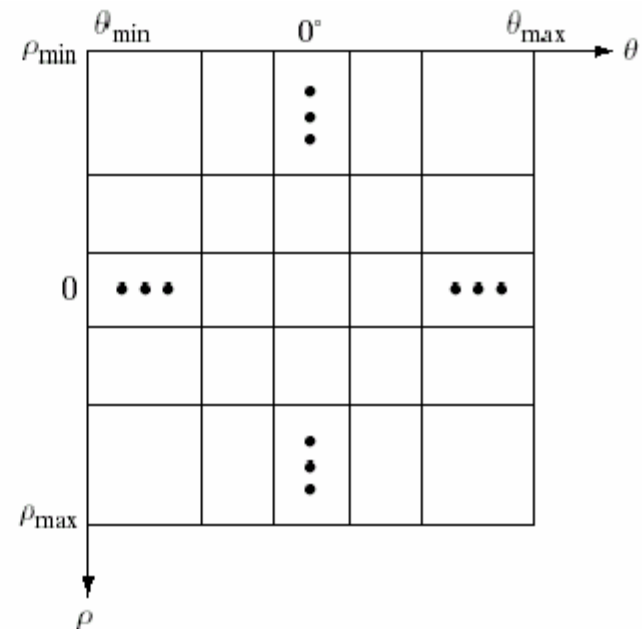
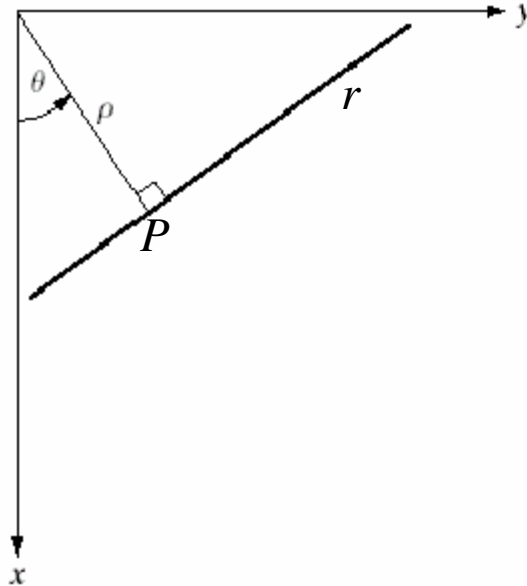


# Transformada de Hough

- Ao contrário do método de “força bruta” que precisava de  $n^3$ , cálculos este método é linear, necessitando apenas  $nk$  cálculos ( $k$  número de divisões do eixo  $a$ ).
- Problema  $-\infty \leq a \leq +\infty$  e  $-\infty \leq b \leq +\infty$
- **Solução:** representar não a recta  $r$  que passa pelo ponto  $P$ , mas antes a recta perpendicular à recta  $r$  e que passa na origem do sistema de eixos representada por:

$$(a, b) \rightarrow (\rho, \theta)$$

$$x \cos \theta + y \sin \theta$$



# Transformada de Hough

$$\rho \in [-\sqrt{2}D, \sqrt{2}D] \text{ e } \theta \in [-90^\circ, 90^\circ]$$

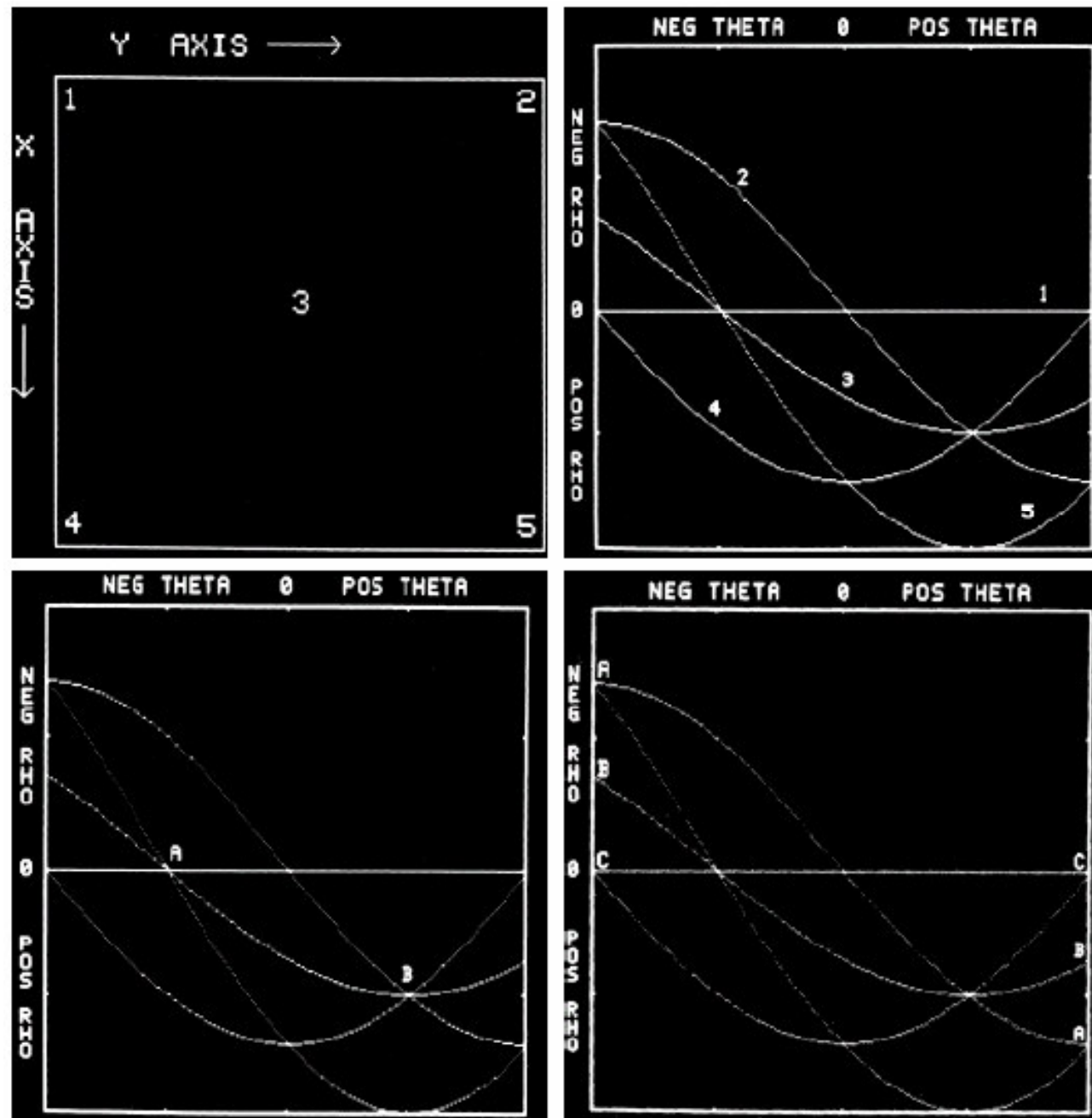
$\sqrt{2}D$  - distância da diagonal entre dois cantos da imagem

## Algoritmo

1. Colocar a zero todas as células do plano acumulador  $(\rho, \theta)$
2. Para cada ponto  $(x_i, y_i)$ 
  - 2.1. Para cada valor de  $\theta$
  - 2.2. Calcular  $\rho = x \cos \theta + y \sin \theta$
  - 2.3. Arredondar o valor de  $\rho$  para o valor mais próximo existente no plano acumulador.
  - 2.4. Incrementar a célula do espaço acumulador  $(\rho, \theta)$  com +1.

# Transformada de Hough

Ilustração da transformada de Hough



# Algoritmo para a ligação de contornos

1. Calcular  $|\nabla I|$  e isolar os pixels pertencentes às arestas através da binarização.
2. Especificar as subdivisões do plano acumulador  $(\rho, \theta)$
3. Aplicar a Transformada de Hough aos pixels das arestas
4. Procurar no plano acumulador as células com os valores mais elevados
5. Examinar a continuidade dos pixels que constituem a célula
6. Ligar os contornos se as quebras forem significativas

# Transformada de Hough para a detecção de círculos

A transformada de Hough é aplicável a qualquer função da forma  $g(\mathbf{v}, \mathbf{c})=0$ , em que  $\mathbf{v}$  é o vector de coordenadas e  $\mathbf{c}$  é o vector de coeficientes.

## Detecção de círculos

$$(x - c_1)^2 (y - c_2)^2 = c_3^2$$

3 Parâmetros  $(c_1, c_2, c_3)$

Espaço acumulador 3D,  $A(i, j, k)$

Algoritmo:

1. Incrementar  $c_1$  e  $c_2$
2. Calcular  $c_3$ .
3. Incrementar a célula do plano acumulador associada a  $(c_1, c_2, c_3)$