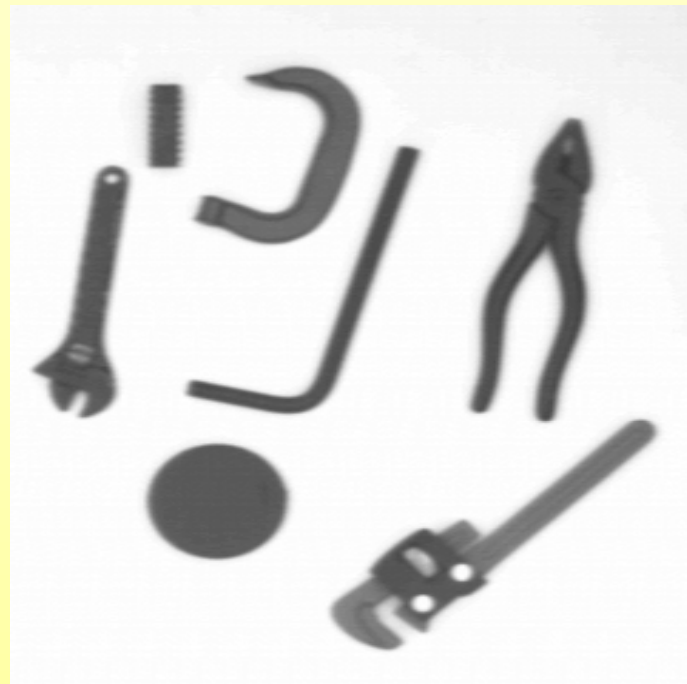


Aula 6 - Segmentação de Imagens

Parte 2



Prof. Adilson Gonzaga

Motivação



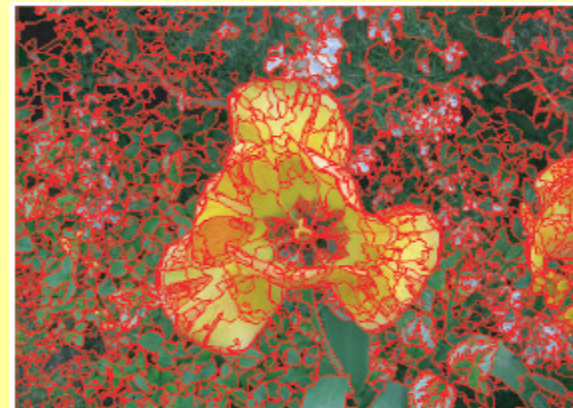
Extração do Objeto



Dificuldades



Super segmentação
over-segmentation



1) Segmentação por Limiarização (Thresholding Global):

Efeitos da escolha do Limiar



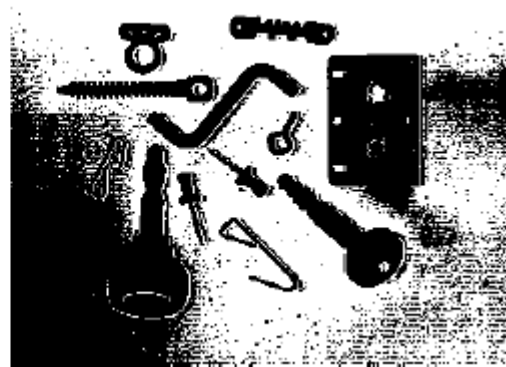
Imagem Original



Imagem segmentada por
Thresholding



Thresholding Baixo



Thresholding Alto

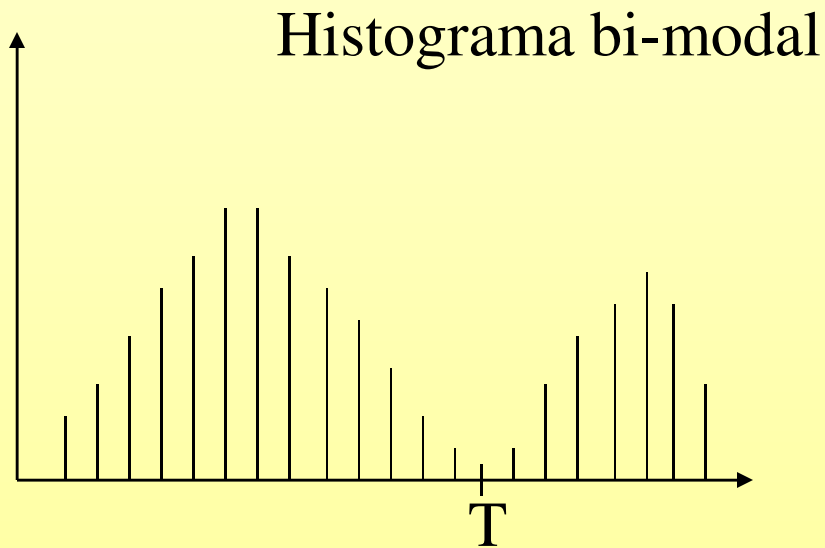
1) Segmentação por Limiarização (Thresholding Global):

Modos de se escolher o Threshold:

1. Inspeção visual do histograma
2. Tentativa e erro
3. Threshold Automático

1) Segmentação por Limiarização (Thresholding Global):

1. Inspeção visual do histograma

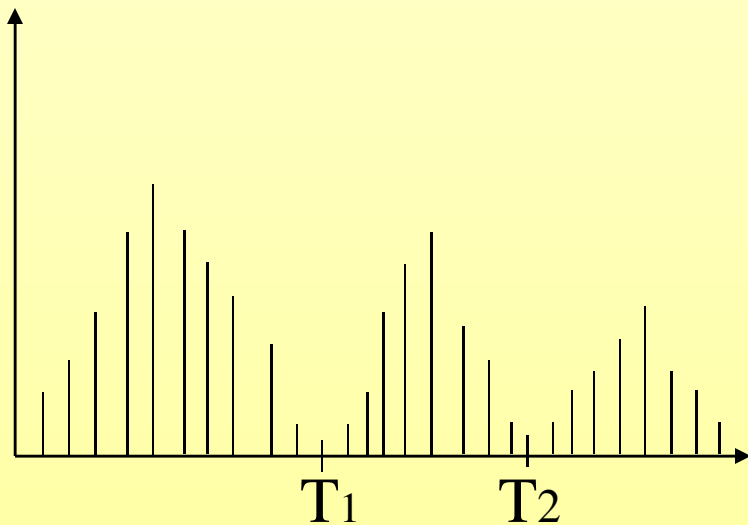


- Imagem $f(x, y)$ composta de objetos brilhantes sobre fundo escuro
- Um ponto (x, y) é parte dos objetos se $f(x, y) > T$

1) Segmentação por Limiarização (Thresholding Global):

1. Inspeção visual do histograma

Limiarização em multinível

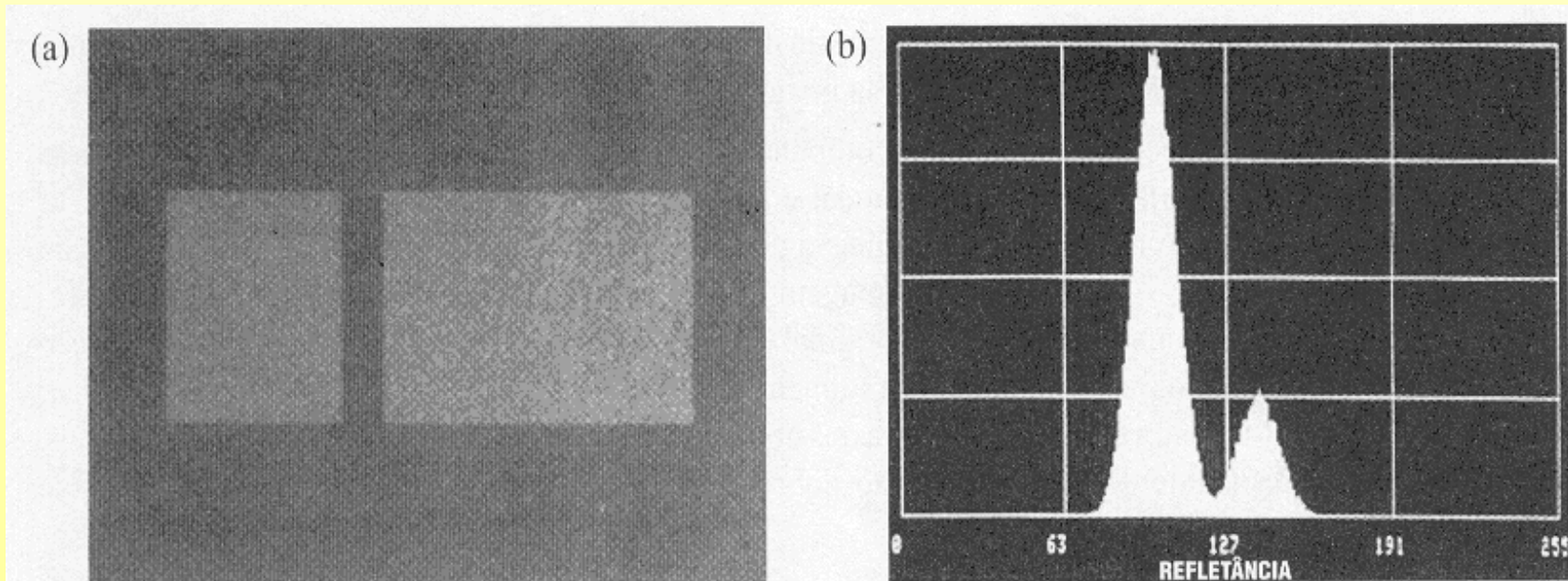


- Se $T_1 < f(x, y) \leq T_2 \rightarrow$ o ponto (x, y) pertence a uma classe de objetos.
- Se $f(x, y) > T_2 \rightarrow$ o ponto (x, y) pertence a outra classe.
- Se $f(x, y) \leq T_1 \rightarrow$ o ponto (x, y) pertence ao fundo.

- **Dificuldade:** estabelecer múltiplos T que efetivamente isolem regiões de interesse.

1) Segmentação por Limiarização (Thresholding Global):

Influência da Iluminação:

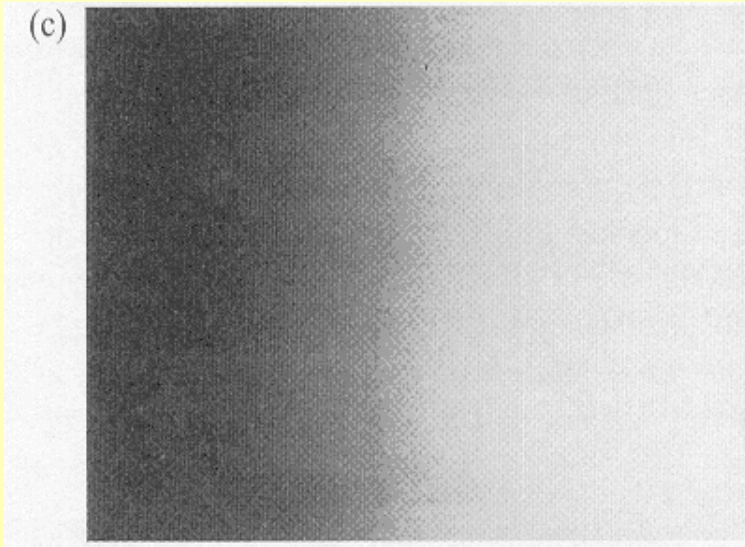


a) Reflectância $r(x,y)$ gerada por computador

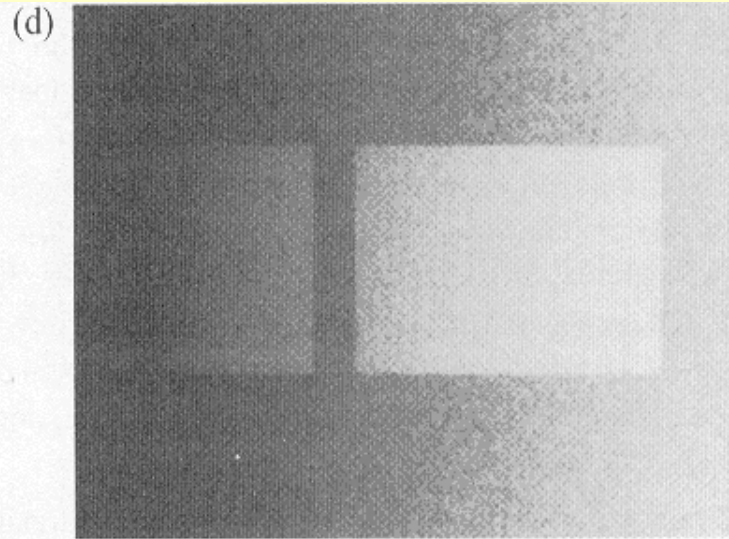
b) Histograma da reflectância (Bi-modal)

$$f(x, y) = i(x, y) \cdot r(x, y)$$

Influência da Iluminação:

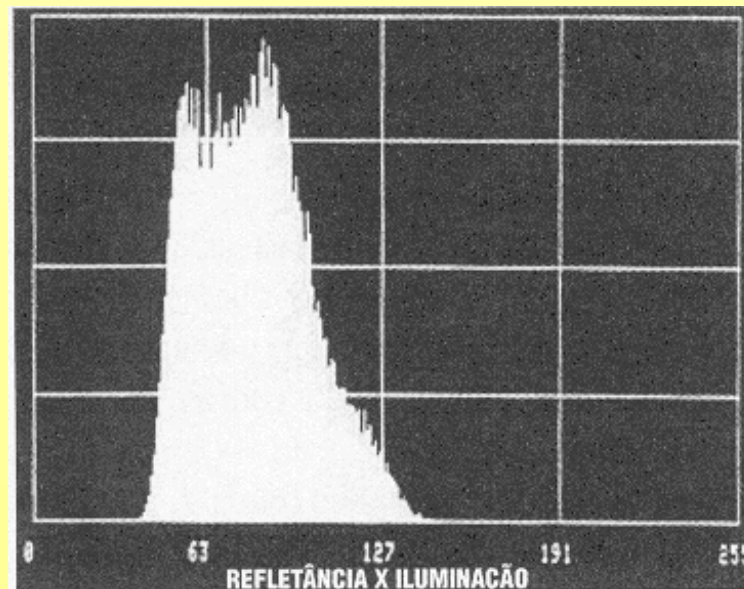


c) Função de Iluminação $i(x,y)$
gerada por computador



$$d) \quad f(x, y) = i(x, y).r(x, y)$$

Histograma da $f(x,y)$



1) Segmentação por Limiarização (Thresholding Global):

2. Tentativa e erro

Aplicado em processos interativos.

O usuário testa diferentes níveis de Threshold até produzir um resultado satisfatório de acordo com o observador.

1) Segmentação por Limiarização (Thresholding Global):

3. Threshold Automático

- Algoritmo: (Gonzalez;Woods,2002)
 1. Selecionar um valor estimado para T (Ponto intermediário entre os valores mínimos e máximos de intensidade da imagem)
 2. Segmentar a imagem usando T .
Isso produzirá dois grupos de pixels

$$G_1 \geq T$$

$$G_2 < T$$

1) Segmentação por Limiarização (Thresholding Global):

3. Computar a média das intensidades dos píxels em cada região.

$$\mu_1(G_1) \quad \mu_2(G_2)$$

4. Computar o novo valor de T:

$$T = \frac{1}{2}(\mu_1 + \mu_2)$$

5. Repetir os passo 2 a 4 até que a diferença em T em sucessivas iterações seja menor que um T_0 pré-estabelecido.

1) Segmentação por Limiarização (Thresholding Global):

Método de Otsu

1. Tratar o Histograma da Imagem como uma Função Densidade de Probabilidade Discreta:

$$p_r(r_q) = \frac{n_q}{n} \quad q = 0, 1, 2, \dots, L-1$$

Onde:

n = número total de píxels da Imagem

n_q = número de píxels com intensidade r_q

L = número total dos possíveis níveis de intensidade da Imagem

1) Segmentação por Limiarização (Thresholding Global):

Método de Otsu

2. Um valor k para o Threshold pode ser escolhido tal que:

C_0 seja a classe de Píxels com níveis entre $[0, k-1]$ e C_1 seja a classe de Píxels com níveis entre $[k, L-1]$

3. O método de Otsu escolhe k tal que maximize a variância inter-classes:

$$\sigma_B^2 = \omega_0 (\mu_0 - \mu_T)^2 + \omega_1 (\mu_1 - \mu_T)^2$$

1) Segmentação por Limiarização (Thresholding Global):

Método de Otsu

$$\sigma_B^2 = \omega_0 (\mu_0 - \mu_T)^2 + \omega_1 (\mu_1 - \mu_T)^2$$

Onde:

$$\omega_0 = \sum_{q=0}^{k-1} p_q(r_q)$$

$$\omega_1 = \sum_{q=k}^{L-1} p_q(r_q)$$

$$\mu_0 = \sum_{q=0}^{k-1} qp_q(r_q) / \omega_0$$

$$\mu_1 = \sum_{q=k}^{L-1} qp_q(r_q) / \omega_1$$

$$\mu_T = \sum_{q=0}^{L-1} qp_q(r_q)$$

O Método de Otsu pode ser chamado de Thresholding Dinâmico.

1) Segmentação por Limiarização (Thresholding Local):

- A Limiarização Global pode falhar quando a iluminação for não uniforme.
- Aplicar um Threshold Local é definir uma função $T(x,y)$ que varie o valor de T de acordo com as coordenadas (x,y) .

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

- Uma função de Threshold Local $T(x,y)$ pode ser obtida como:

$$T(x, y) = f_0(x, y) + T_0$$

- Onde $f_0(x,y)$ é a abertura morfológica de $f(x,y)$ e a constante T_0 o valor do Threshold, pelo método de Otsu, aplicado em $f_0(x,y)$.

2) Segmentação Orientada a Região:

- Bordas e Fronteiras – Segmentação baseada em descontinuidades
- Regiões – Segmentação baseada em similaridade das propriedades dos Pixels.

Crescimento de Região: (Region Growing)

- Seja R a região completa da Imagem, e R_1, R_2, \dots, R_n sub-partições de R tal que:

a) $\bigcup_{i=1}^n R_i = R$ (todo pixel deve estar em uma região)

b) R_i é uma região 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 $i \neq j$

$P(R_i) \rightarrow$ Propriedade definida para a região é verdadeira para todos os pixels da região

Ex: Intensidade igual.

Crescimento de Região por agregação de pixel.

- Agrupamento de Pixels ou grupo de pixel em regiões maiores.
- Os pixels a serem agrupados devem ter propriedades similares. (nível de cinza, textura, cor, etc...).
- Inicia-se com um conjunto de “sementes” em torno do qual as regiões crescem.

Imagem:

	1	2	3	4	5
1	0	0	5	6	7
2	1	1	5	8	7
3	0	<u>1</u>	6	<u>7</u>	7
4	2	0	7	6	6
5	0	1	5	6	5

	R2				
R1	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

T=8

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

Sementes $\rightarrow (3, 2)$ e $(3, 4)$

Propriedade (P) $\rightarrow |I(x) - I(s)| < T$
 $I(s) \rightarrow$ Intensidade da semente

Problemas com a Técnica:

1) **Seleção das sementes:** depende da natureza do problema.

Ex: em aplicações militares com imagens com infravermelho, os pontos mais quentes, logo, mais brilhantes, são de interesse.

2) **Seleção das Propriedades que estabeleçam os critérios de similaridade:** depende do tipo de dados disponíveis.

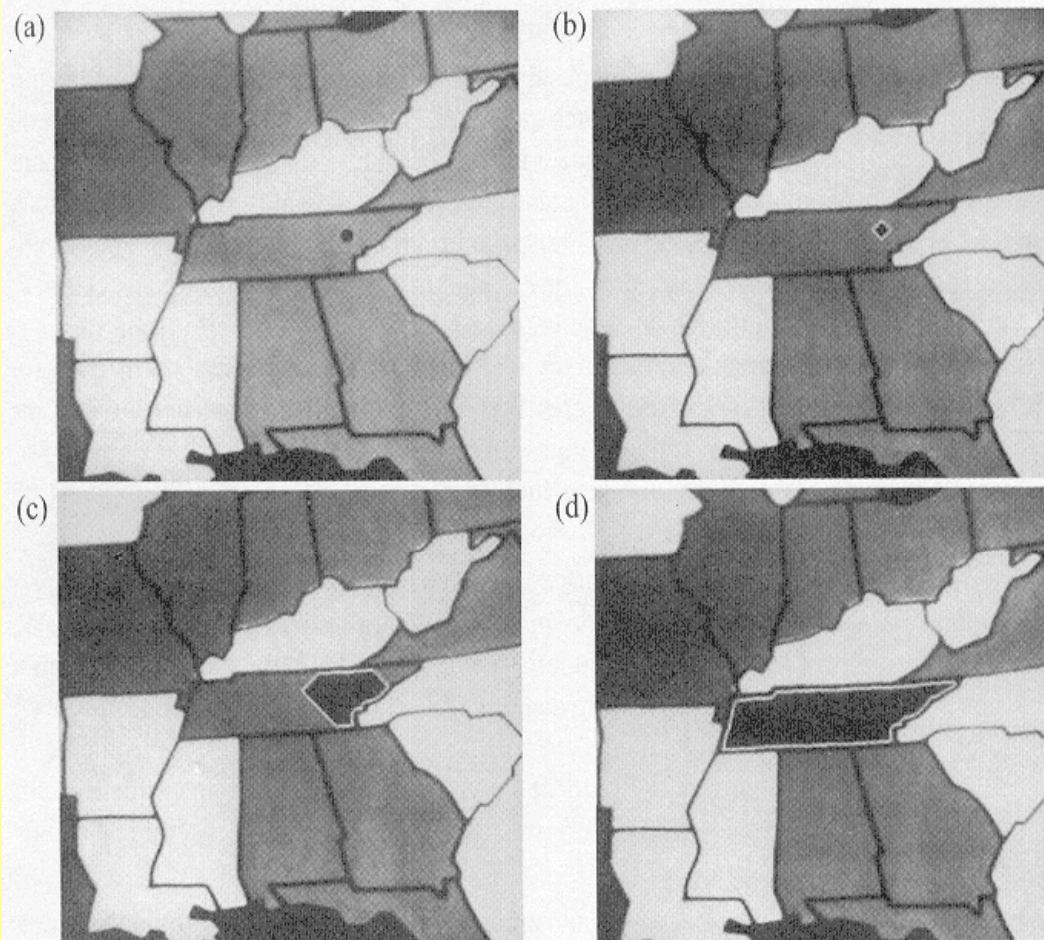
Ex: as imagens de satélite usam a informação de cor.

3) Utilização de conectividade e adjacência:

Ex: uma imagem formada por um arranjo aleatório de 3 intensidades diferentes. Se a conexão entre pixels não for levada em conta, o resultado da segmentação não terá nenhum significado.

4) **Formulação de uma regra de parada:** utilização de critérios de tamanho, semelhança entre um pixel candidato e os pixels da Região e Formato de uma dada Região.

Exemplo: Crescimento de Região



a) Imagem $[I(x,y)]$ com semente $I(s)$

Critério:

$$| I(x, y) - I(s) | \leq 10\% (255 - \min)$$

8 – conectada em cada pixel.

b) **Início do crescimento:**
Pixels com a mesma distancia "city-block" da semente.

c) Estado intermediário de crescimento.

d) Região crescida completa: o processo pára devido a borda de nível mais escuro que fura a conectividade.

Divisão e Fusão de Regiões:

“Split and Merge”

- Seja R uma Imagem e P uma característica de similaridade definida.

- Subdividir R em 4 Regiões (Quadrantes)

$$R_i / P(R_i) = \text{VERDADEIRO}$$

- Se $P(R_i) = \text{FALSO}$ subdividir a Região em sub-quadrantes.

- Fundir as Regiões adjacentes onde:

$$P(R_i \cup R_k) = \text{VERDADEIRO}$$

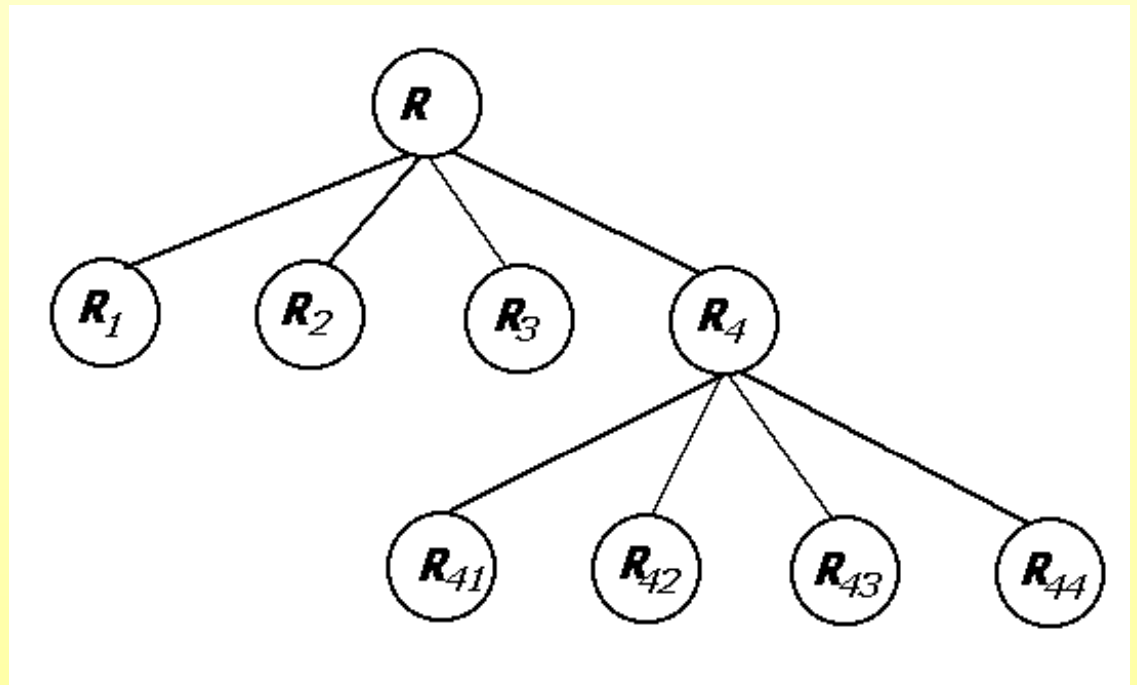
- Parar quando nenhuma divisão nem nenhuma fusão for possível

Divisão e Fusão de Regiões:

Região R

R_1	R_2	
R_3	R_{41}	R_{42}
	R_{43}	R_{44}

Dividir ("Split") uma Região R.

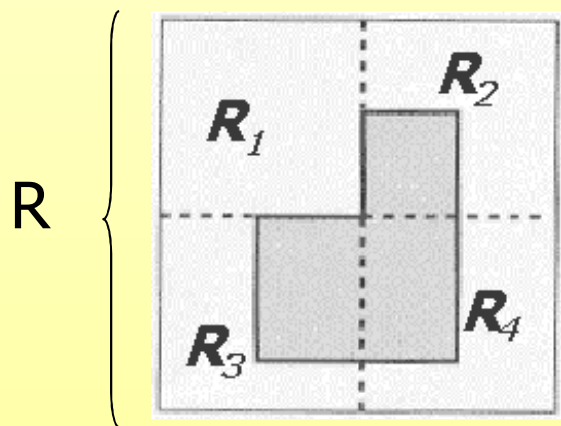


Árvore Quadrática (Quadtree) que representa a Região R.

Exemplo do Algoritmo "Split and Merge":

$P(R_i) = \text{VERDADEIRO}$ para R_i de mesma intensidade

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



Primeira Sub-divisão
"Split"

$P(R_1) = \text{VERDADEIRO}$

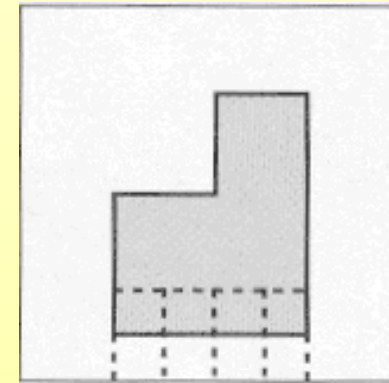
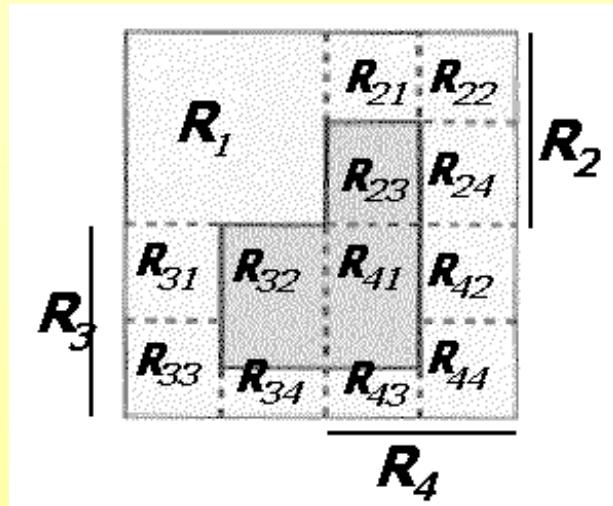
$P(R_2) = \text{FALSO}$

$P(R_3) = \text{FALSO}$

$P(R_4) = \text{FALSO}$

Exemplo do Algoritmo "Split and Merge":

Segunda sub-divisão:
"Split"



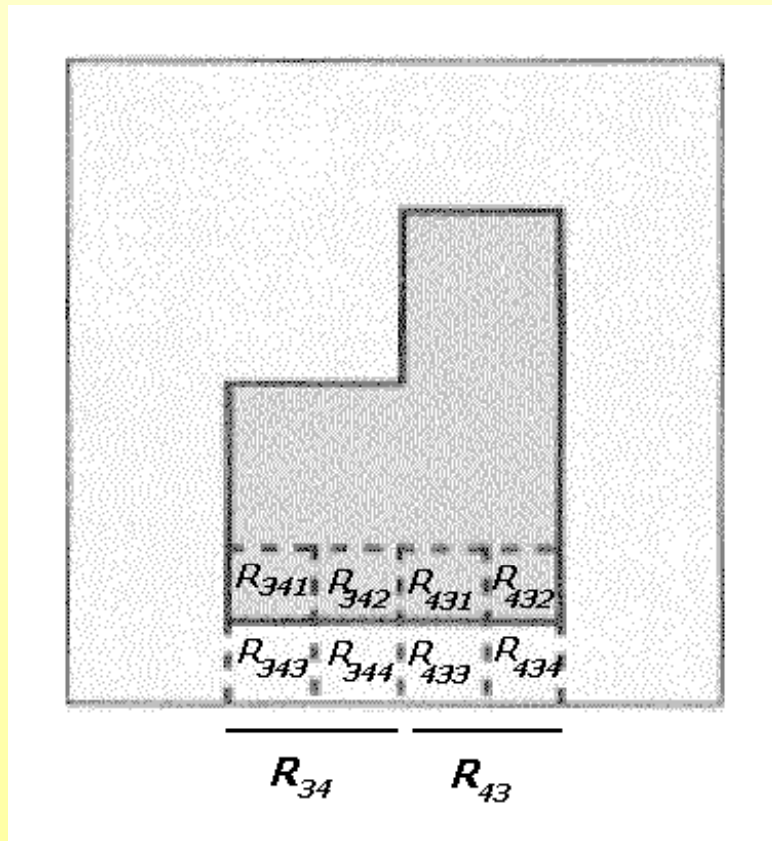
$$P(R_{34}) \text{ e } P(R_{43}) = \text{FALSO}$$

Fusão "Merge":

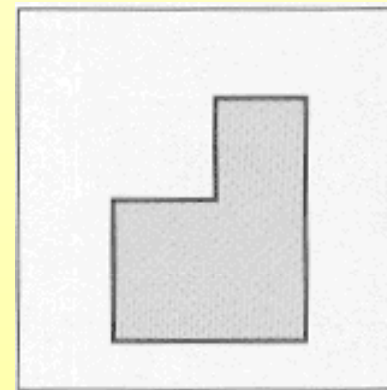
$$P(R_1 \cup R_{21} \cup R_{22} \cup R_{24} \cup R_{42} \cup R_{44} \cup R_{33} \cup R_{31}) = \text{VERDADEIRO}$$

$$P(R_{23} \cup R_{41} \cup R_{32}) = \text{VERDADEIRO}$$

"Split"



Região Segmentada.



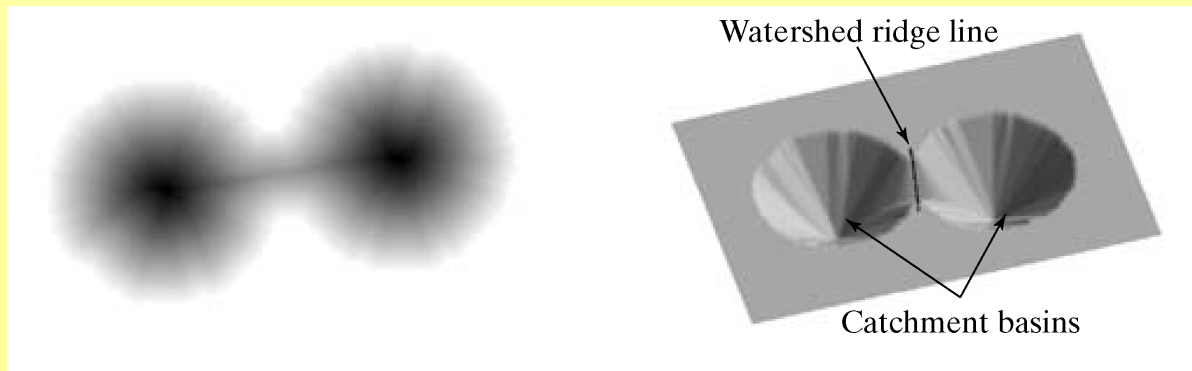
"Merge"

$$P(R_{341} \cup R_{342} \cup R_{431} \cup R_{432}) = \text{VERDADEIRO}$$

$$P(R_{343} \cup R_{344} \cup R_{433} \cup R_{434}) = \text{VERDADEIRO}$$

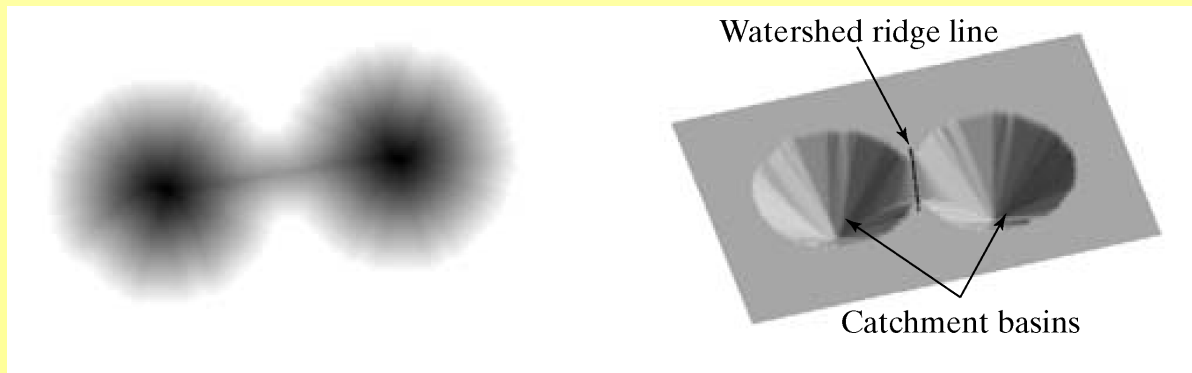
3) Transformada Watershed

- Watershed, em geografia, são as saliências que dividem as áreas inundadas por diferentes rios (Bacias Hidrográficas).
- A área que armazena água são as Bacias (Catchment Basin), formada pelas partes mais fundas.
- A Transformada Watershed aplica estas idéias nas imagens em nível de cinza para a segmentação.
- A Imagem é vista como a Topografia 3-D de uma área onde o valor da intensidade do pixel é plotado no eixo z, em cada coordenada (x,y).



3) Transformada Watershed

- A chuva que cai nesta área vai escorrer e ocupar as partes mais baixas do terreno.
- A água que cair exatamente sobre a linha divisória (Watershed) terá a mesma probabilidade de escorrer para qualquer das bacias por ela dividida.
- A Transformada Watershed segmenta as regiões considerando as áreas inundadas entre as linhas de Watershed como as regiões da imagem.

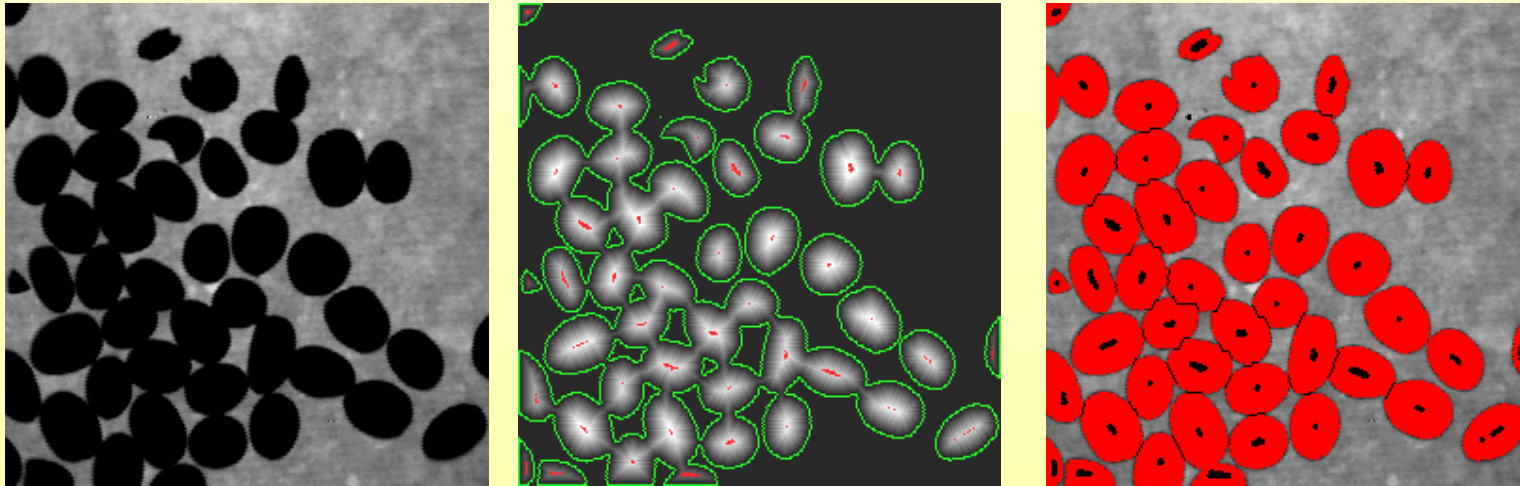


3) Transformada Watershed

Se enchermos esta superfície de um mínimo e prevenirmos a mistura das águas originárias de diferentes pontos, a imagem será separada em dois diferentes conjuntos: as Bacias de Captação(**catchment basins**) e as Linhas de Watershed (**watershed lines**).



Exemplo: Aplicação da Transformada Watershed na Segmentação de grãos de café



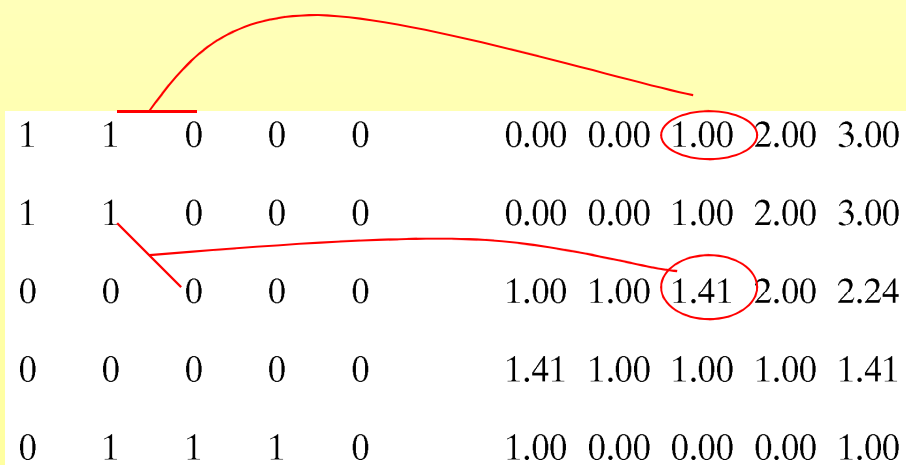
A Transformada Watershed utiliza diversas técnicas auxiliares para segmentar a imagem de maneira coerente e evitar a super-segmentação (Oversegmentation).

Segmentação por Watershed usando a Transformada da Distância.

- Transformada da Distância:

Transformada da Distância em uma Imagem Binária é a distância de cada pixel para o próximo pixel de valor não zero.

Exemplo: Transformada da Distância Euclidiana.



1	1	0	0	0	0.00	0.00	1.00	2.00	3.00
1	1	0	0	0	0.00	0.00	1.00	2.00	3.00
0	0	0	0	0	1.00	1.00	1.41	2.00	2.24
0	0	0	0	0	1.41	1.00	1.00	1.00	1.41
0	1	1	1	0	1.00	0.00	0.00	0.00	1.00

Imagem Binária

Transformada da
Distância Euclidiana

Segmentação por Watershed usando a Transformada da Distância.

Imagem Binária

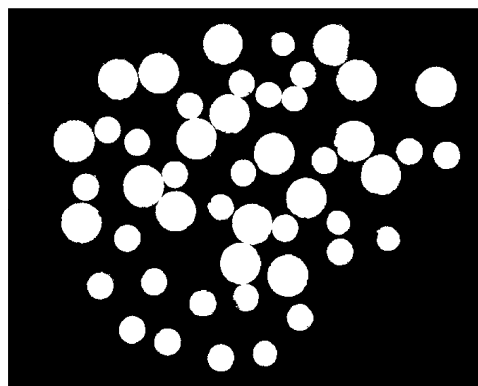
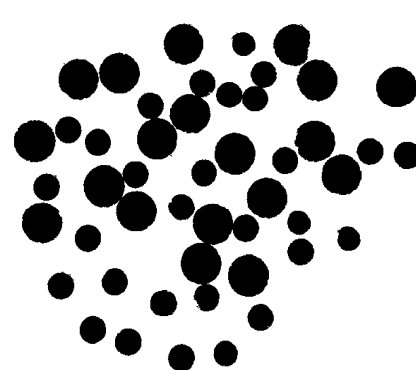
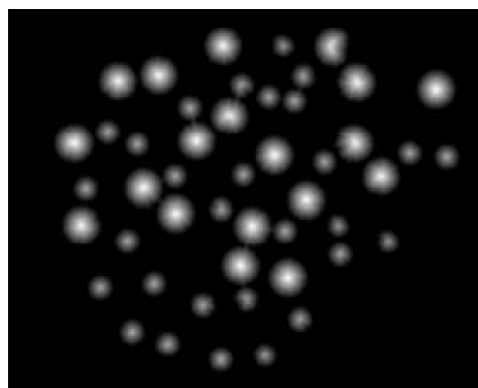


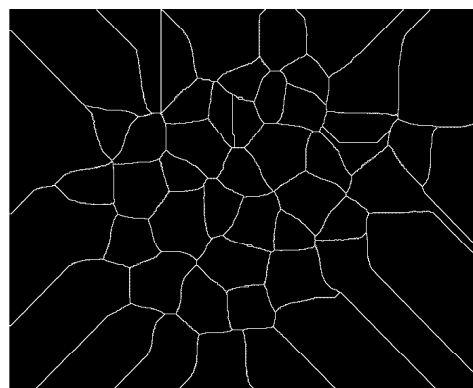
Imagem Binária Complementada



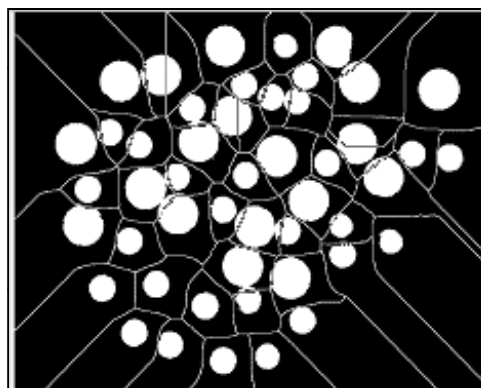
Transformada da Distância (TDE)



Linhas de Watershed do negativo da TDE



Super-Segmentação
(Oversegmentation) – A Transformada Watershed pode gerar segmentação extra que não corresponde a regiões na imagem

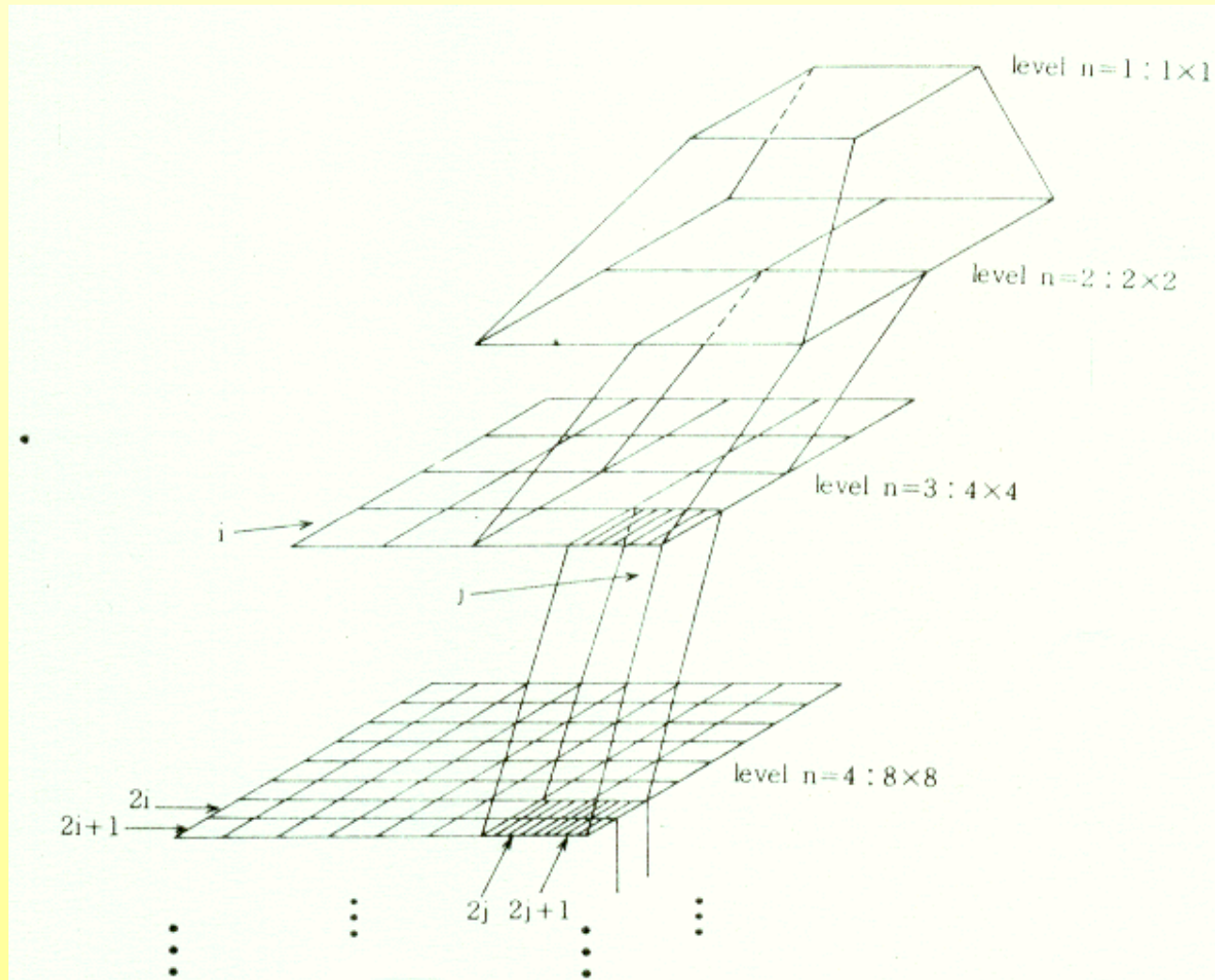


Superposição das linhas de Watershed na Imagem Original

4) Pirâmides de Resolução:

- ❑ Quando a imagem $f(x,y)$ é muito grande e quadrada ($n \times n$), uma abordagem para a detecção das bordas é assumir vários níveis de resolução organizados como uma estrutura de Pirâmide de arranjos bi-dimensionais.
- ❑ Esse procedimento imita a atenção seletiva ou percepção focalizada.
- ❑ Os arranjos são do tipo $f(i,j,n)$ com $n=1, 2, \dots, p$
- ❑ Os arranjos de mais alto nível (n menor) auxiliam a detectar as bordas mais grossas.
- ❑ Os arranjos de mais baixo nível (n maior) fornecem a posição das bordas.

Exemplo de uma Pirâmide para $n=4$:



Geração da Pirâmide:

Cada 4 píxels no arranjo de nível inferior, gera 1 píxel no arranjo de nível superior (média).

As bordas são mais facilmente detectadas no nível $n=1$.

A posição da borda detectada em $n=1$, é localizada em $n=4$.

Qualquer detector de Bordas pode ser utilizado.

5) Contornos Ativos (Snakes)

- Os algoritmos de Contornos Ativos, deformam um contorno para coincidir com características de interesse em uma imagem.
- Normalmente estas características de interesse são bordas ou fronteiras.
- O nome dado de Snakes é devido ao fato que os contornos se deformam durante o processo iterativo, como serpentes em movimento.
- Um Contorno Ativo é uma coleção V de n pontos no plano de Imagem.

$$V = \{v_1, \dots, v_n\}$$

$$v_i = (x_i, y_i) \quad i = 1, \dots, n$$

5) Contornos Ativos (Snakes)

- Os pontos, iterativamente, aproximam-se da fronteira do objeto através da solução de um problema de **Minimização de Energia**.
- Para cada ponto na vizinhança de v_i , um termo de Energia é computado:

$$E_i = \alpha E_{\text{int}}(v_i) + \beta E_{\text{ext}}(v_i)$$

onde

$E_{\text{int}}(v_i)$ é uma **Função de Energia** dependente do formato do contorno e

$E_{\text{ext}}(v_i)$ é uma **Função de Energia** dependente de propriedades da imagem
(Ex: Gradiente) próximo a v_i

α e β são constantes que providenciam a ponderação entre os dois termos de Energia.

5) Contornos Ativos (Snakes)

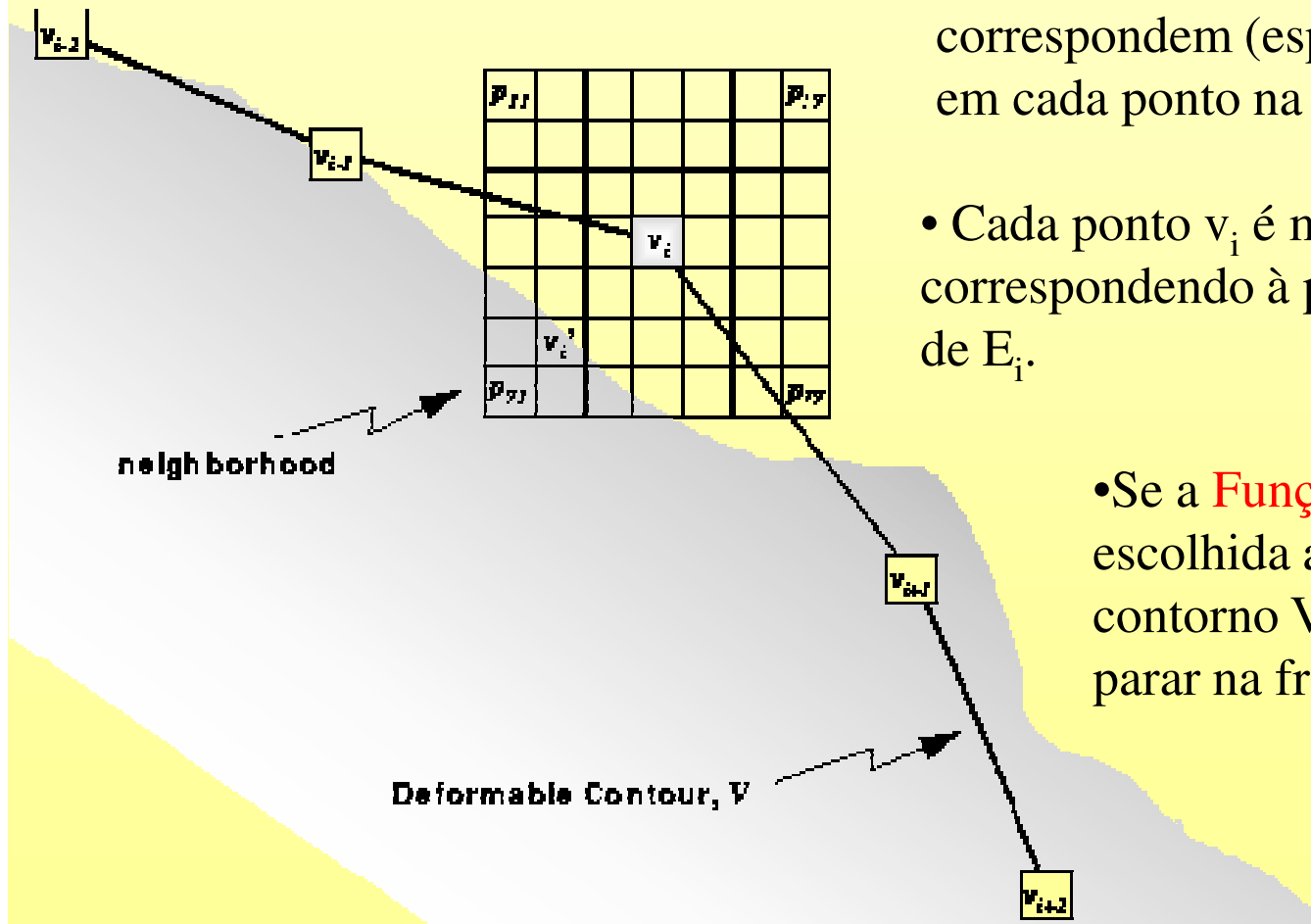
E_i , E_{int} e E_{ext} são matrizes.

- O valor no centro de cada matriz corresponde à **Energia do Contorno** no ponto v_i .

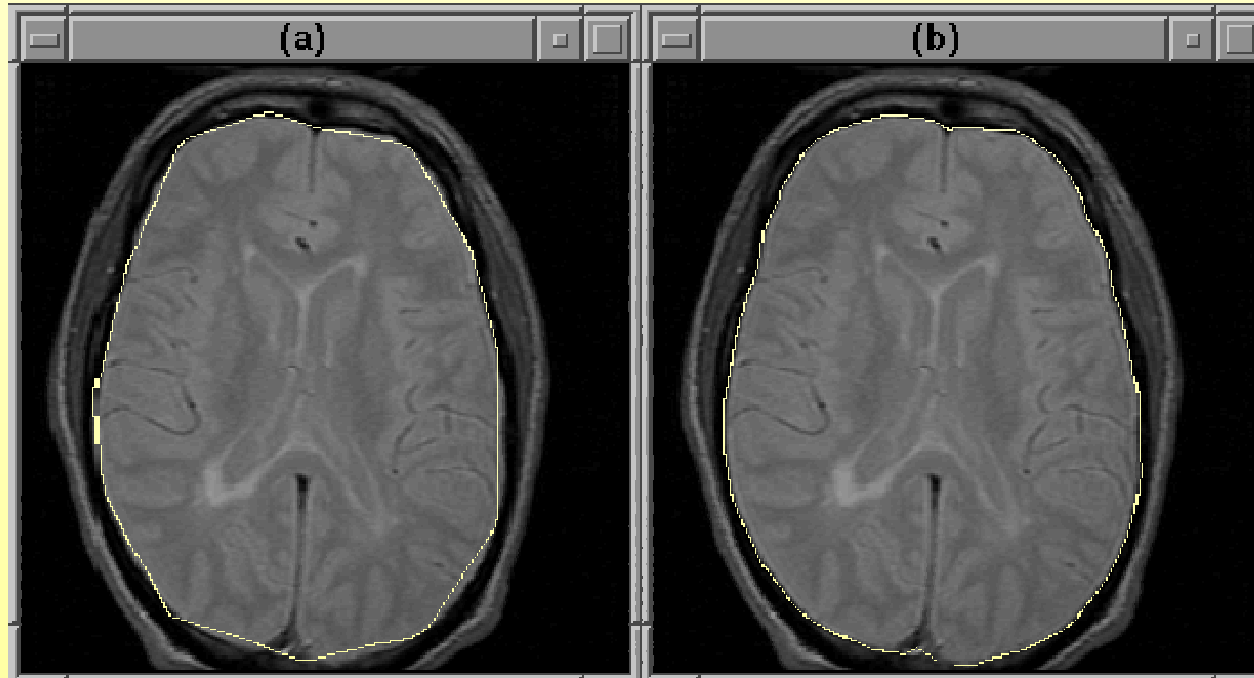
- Os outros valores nas Matrizes correspondem (especialmente) à energia em cada ponto na vizinhança de v_i .

- Cada ponto v_i é movido para o ponto v_i' , correspondendo à posição de menor valor de E_i .

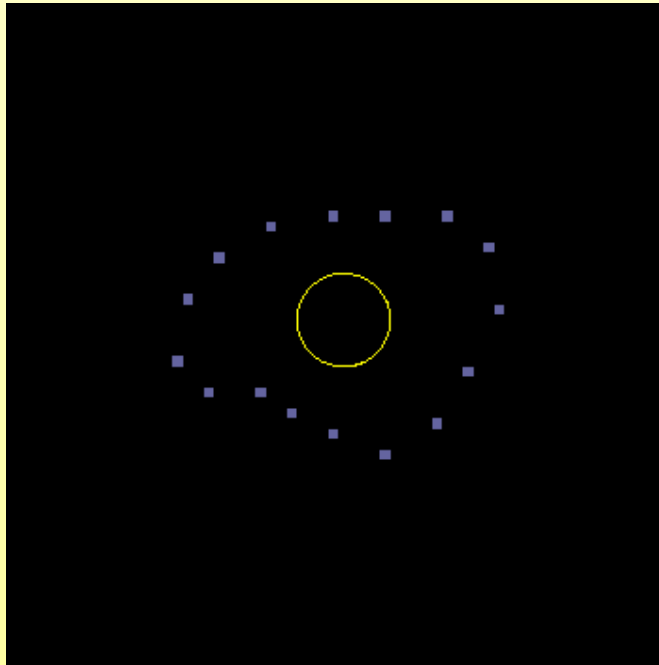
- Se a **Função de Energia** for escolhida apropriadamente, o contorno V deverá aproximar e parar na fronteira do objeto.



Exemplo: Snake utilizada para segmentar o contorno do cérebro em uma cavidade Craniana.



Exemplo: Snake segmentando um contorno.



Outras Técnicas para Segmentação de Imagens

Segmentação por Textura

Segmentação por Cor (Imagens Coloridas)

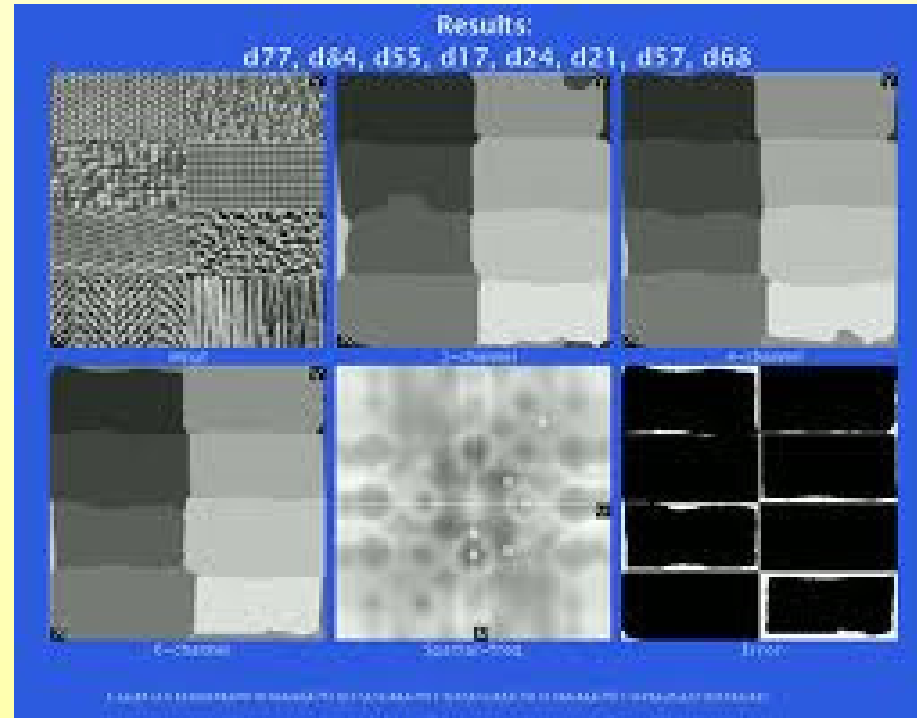
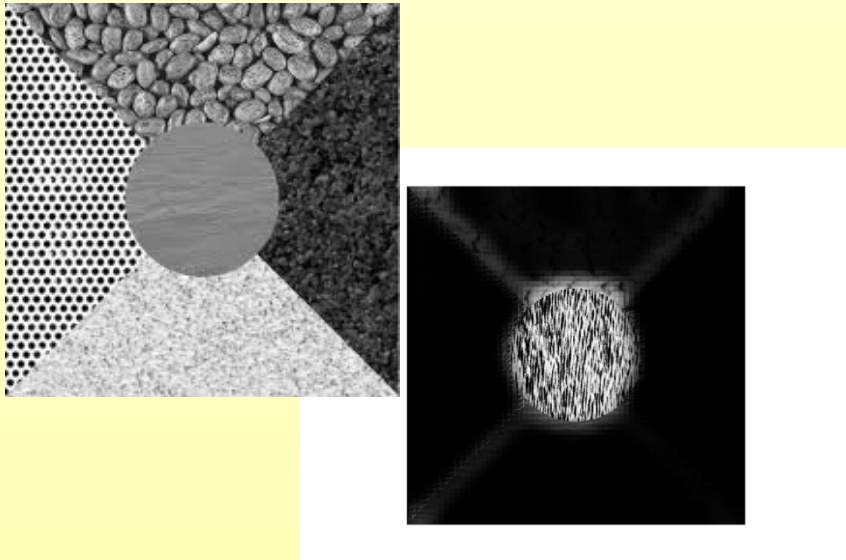
Segmentação utilizando Morfologia Matemática

Segmentação por Agrupamento (K-médias)

Segmentação por Movimento

Segmentação utilizando Redes Neurais Artificiais,
Lógica Fuzzy, etc...

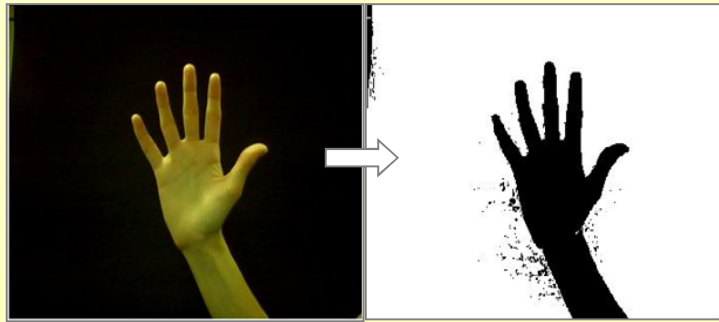
Segmentação por Textura



A Textura de regiões da imagem pode ser segmentada utilizando-se :

- Abordagem Estatística (Matriz de Co-ocorrência)
- Abordagem Estruturais (Símbolos)
- Abordagem Espectrais (Fourier, Wavelets)
- Padrões Locais (Texture Unit, LBP, Transformada Census)

Segmentação por Cor (Imagens Coloridas)



Deve ser utilizada onde a Cor exerce papel importante na identificação dos segmentos ou objetos da imagem.

Pode-se utilizar os diversos espaços de cor (sRGB, HSI, HSV, YCbCr, LUV, etc...)

Segmentação por Movimento

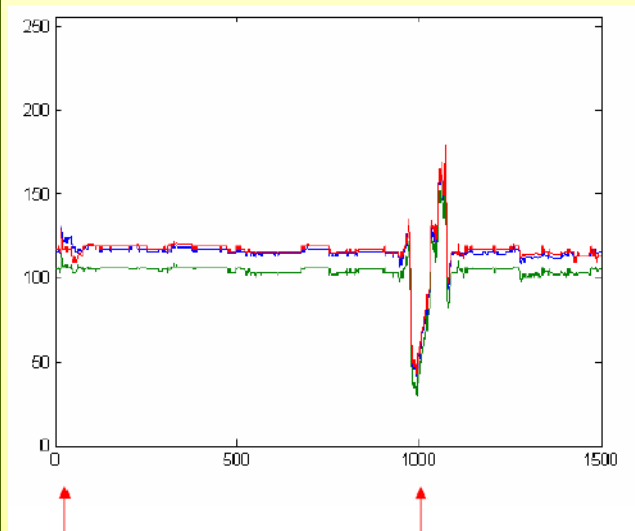
O movimento de objetos em “frames” de vídeo fornece meio para segmentação do objeto e/ou do fundo da cena.

A principal metodologia é a Subtração do Fundo (Background Subtraction):

- Fundo Simples → Obtém-se um modelo do fundo que é subtraído de cada quadro.
- Fundo Complexo → Modelo de Mistura de Gaussianas (GMM)

Segmentação por Movimento (Fundo Simples)

Curva de Evolução de um píxel (RGB)

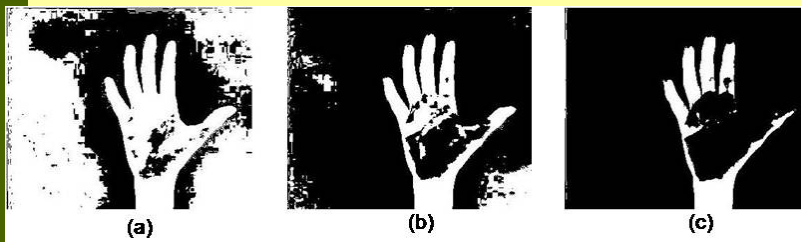


Média ou
Mediana

Ruído

$$I(x, y, t) = \bar{I}(x, y) + \omega(x, y, t)$$

$$|I(x, y, t) - \bar{I}(x, y)| > T$$

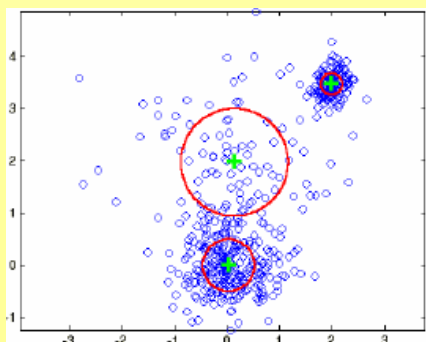


Este método é muito popular para operações de rastreamento em tempo real por ser simples e rápido. No entanto, tem dois grandes inconvenientes. Ele não pode lidar com ruídos intermitentes (por exemplo, mudanças de iluminação) e pode gerar objetos fantasmas.

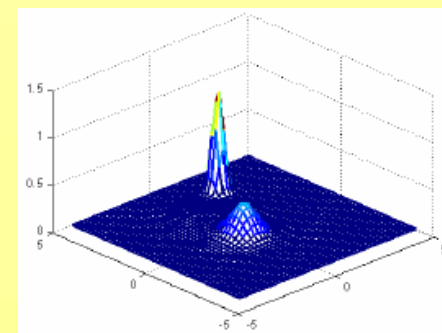
Segmentação por Movimento (Fundo Complexo)

Subtração do fundo usando Mistura de Gaussianas (GMM)

- Cada pixel é classificado baseado na distribuição de Gaussianas que o representa mais eficazmente como fundo.
- Realiza aproximações quadro a quadro para atualizar o modelo de cena de fundo.
- Segmentação ambientes externos dinâmicos, sujeitos a diferentes condições de iluminação.
- Diferentes Gaussianas supostamente representam diferentes cores.
- Espera-se que uma distribuição de Gaussianas que represente fundo tenha grande peso e baixa variância, ou seja, ocorra frequentemente e varie pouco no tempo.



Exemplo de agrupamentos de densidades criadas usando-se GMM.



Modelo de Mistura de Gaussianas (GMM)

- GMM:
 - Robusto com fundos complexos e variações graduais de iluminação.
 - Segmentação correta onde existe fundo com cores próximas a da pele.
 - Não apresenta bons resultados à variação brusca de iluminação e sombras.
 - Objeto praticamente estático, provoca a detecção desse objeto como fundo.

