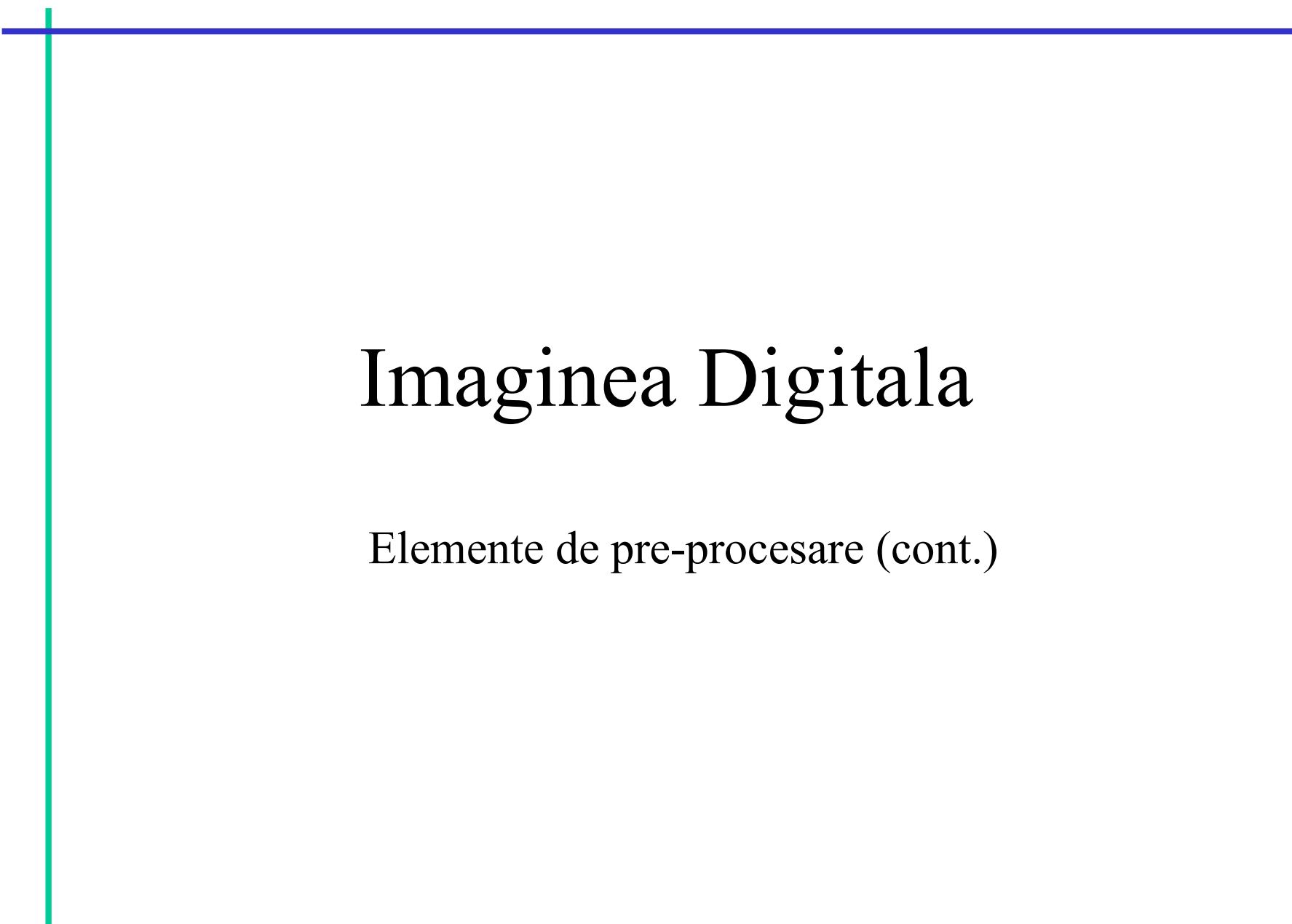




Procesarea Imaginilor si Viziune Artificiala

Curs 2

Conf. dr. ing. Remus BRAD, Universitatea “Lucian Blaga” din Sibiu



Imaginea Digitala

Elemente de pre-procesare (cont.)

Diferențierea statistică

generarea unei imagini prin împărțirea valorii fiecărui pixel la deviația sa standard $S(j,k)$:

$$G(j, k) = \frac{F(j, k)}{S(j, k)} \quad (4.5)$$

unde deviația standard estimată este:

$$S(j, k) = \frac{1}{W^2} \sum_{m=j-w}^{j+w} \sum_{n=k-w}^{k+w} [F(m, n) - M(m, n)]^2 \quad (4.6)$$

este calculată pentru fiecare pixel care se află în vecinătatea de dimensiune $W \times W$, unde $W=2w+1$.

$M(j,k)$ este valoarea medie estimată a imaginii originale în punctul (j,k) :

$$M(j, k) = \frac{1}{W^2} \sum_{m=j-w}^{j+w} \sum_{n=k-w}^{k+w} F(m, n) \quad (4.7)$$

Diferențierea statistică

Wallis a sugerat o generalizare a operatorului diferențierii statistice, în care imaginea mărită este forțată spre o formă cu momente de ordin întâi și ordin secund dorite. Operatorul este definit de relația:

$$G(j, k) = [F(j, k) - M(j, k)] \left[\frac{AS_d}{As(j, k) - S_d} \right] + [rM_d + (1 - r)M(j, k)] \quad (4.8)$$

unde M_d și S_d reprezintă media și deviația standard dorite,

A este un factor de câștig care previne valorile prea mari atunci când $S(j, k)$ este prea mic

r este un factor de proporționalitate a mediei care controlează raportul dintre contur și fundalul imaginii.

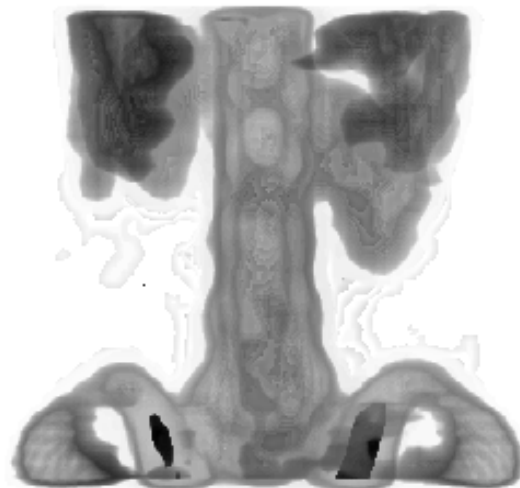
Diferențierea statistică



a) valoarea medie estimată



b) deviația standard estimată

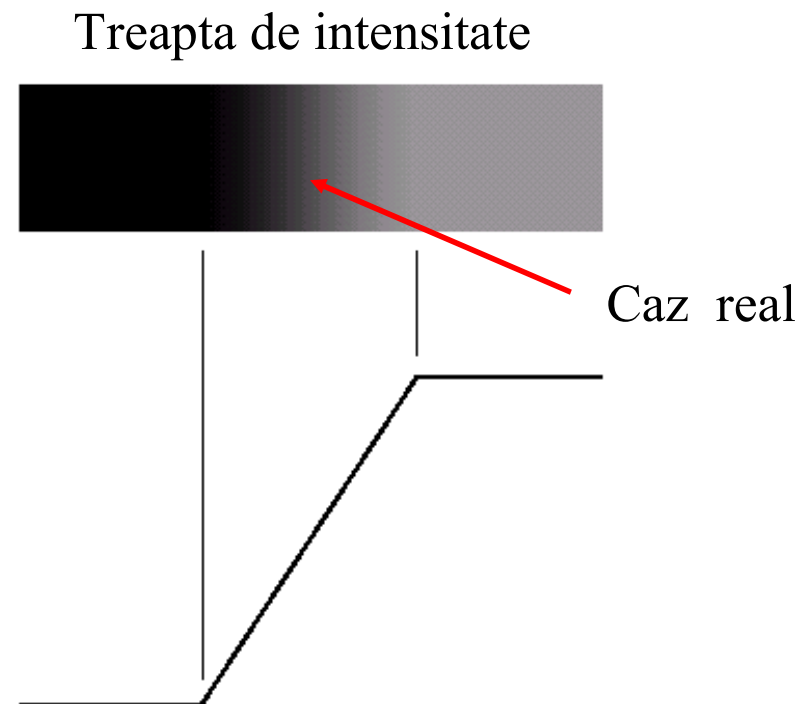


c) $r=0.1, A=4, M_d=0.5, S_d=0.33$

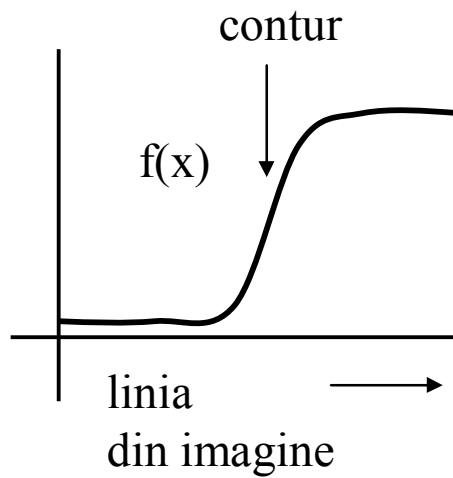


d) $r=0.8, A=4, M_d=0.5, S_d=0.33$

Contururi in imagine



Detecția de contur



Derivata poate fi exprimata:

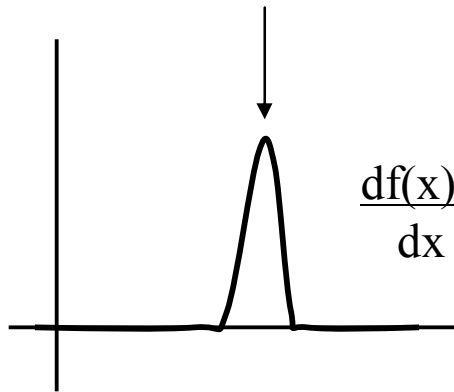
$$\frac{df(x)}{dx} \approx \frac{f(x+h) - f(x)}{h}$$

sau

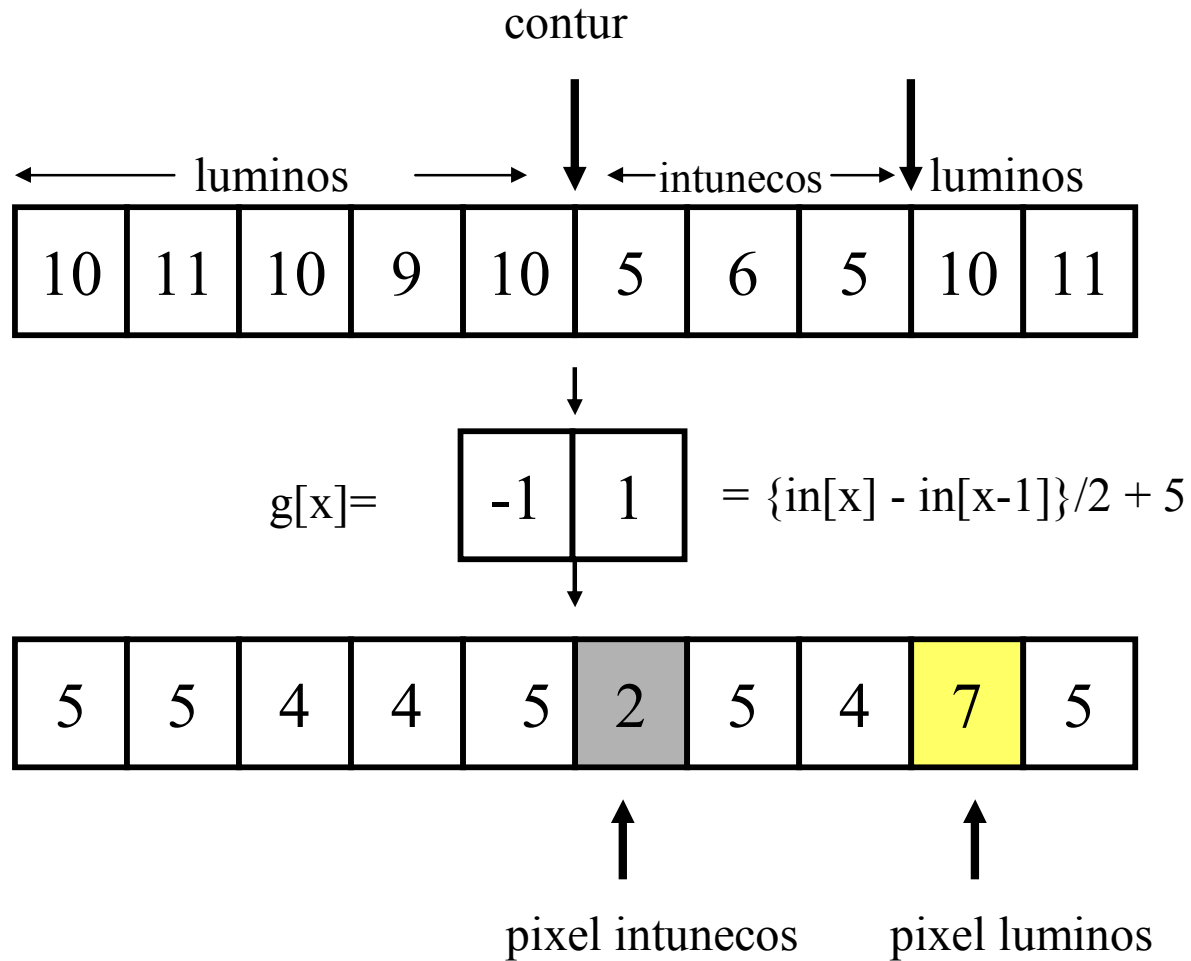
$$\text{out}[x,y] = \text{in}[x+1,y] - \text{in}[x,y]$$

sau

$$g[x,y] = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

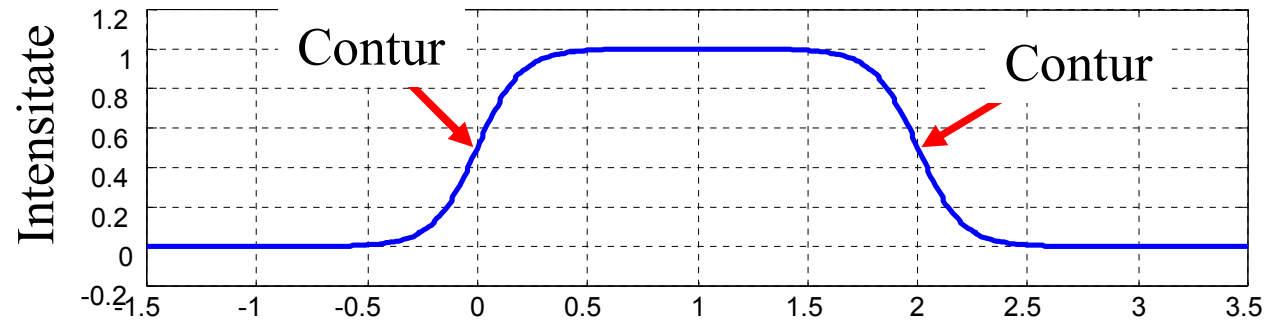


Detecția de contur

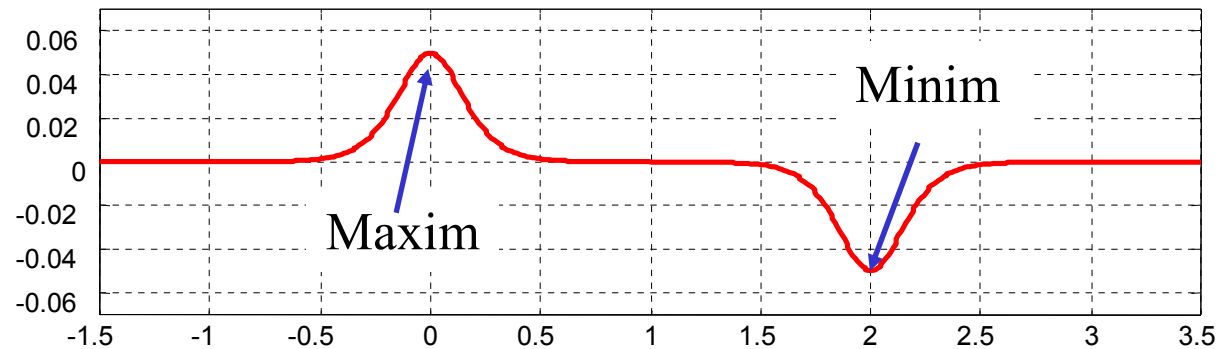


Detecția de contur

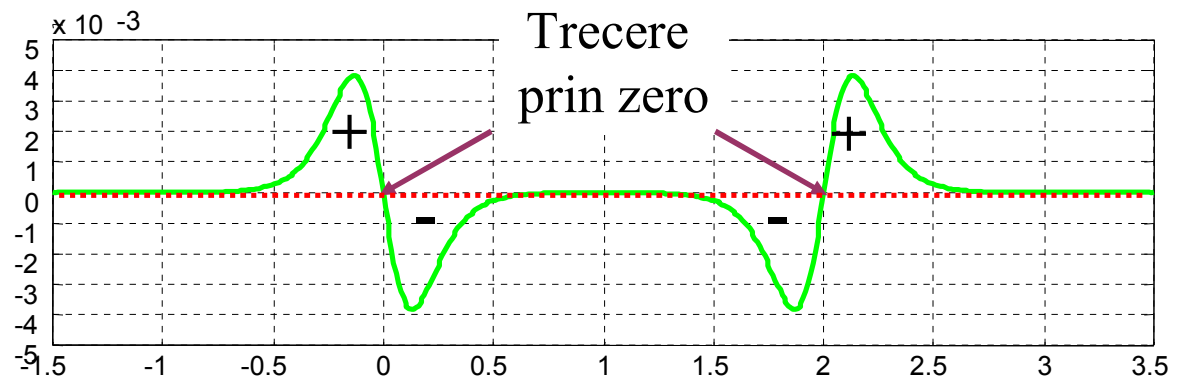
Profil de contur



Prima derivata



A doua derivata

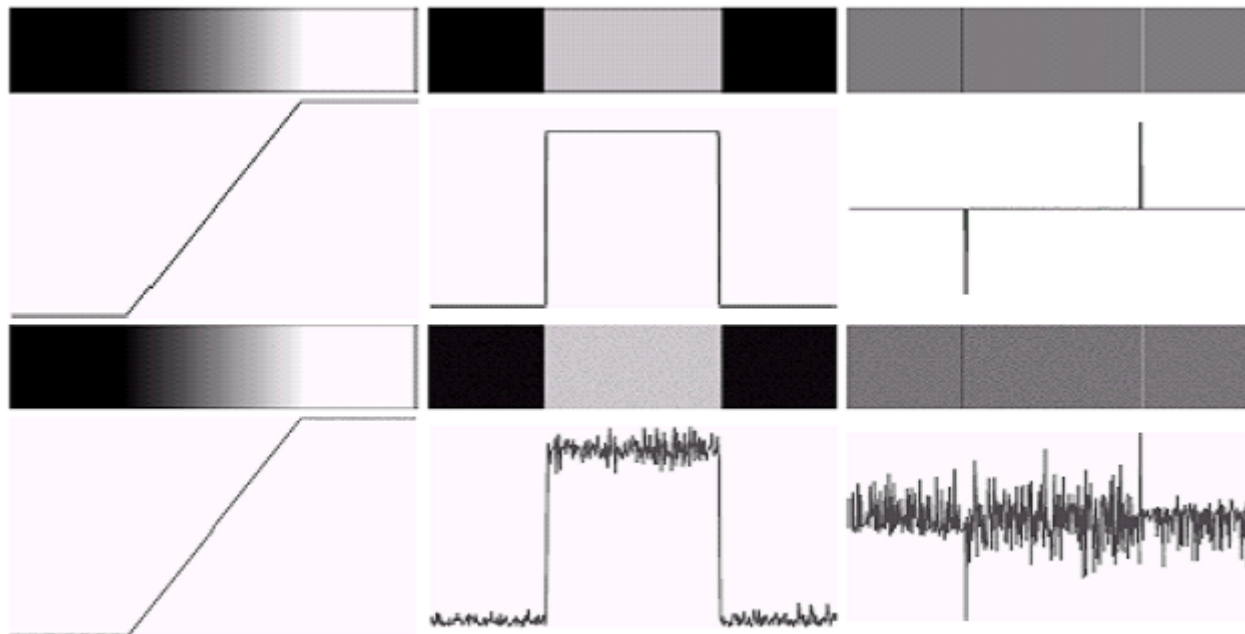


Detecția de contur

Pentru a detecta:

- maximele valorii absolute a primei derivate
- trecerile prin zero ale derivatei a doua

Operatorul derivata este foarte sensibil la zgomot !



Detecția de contur

Gradientul

$$\nabla P = [\partial P / \partial x, \partial P / \partial y]$$

cu:

P este înălțimea

∇P este vectorul

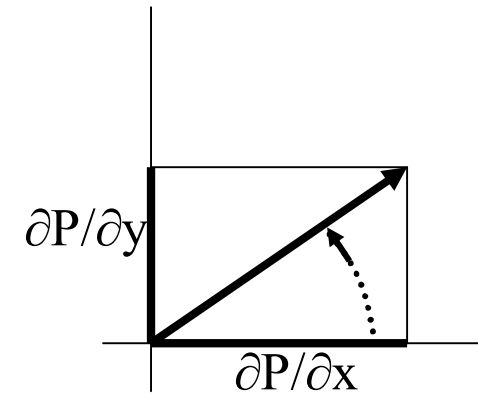
$|\nabla P|$ este panta

unde:

$\partial P / \partial x$ = derivata pe orizontală

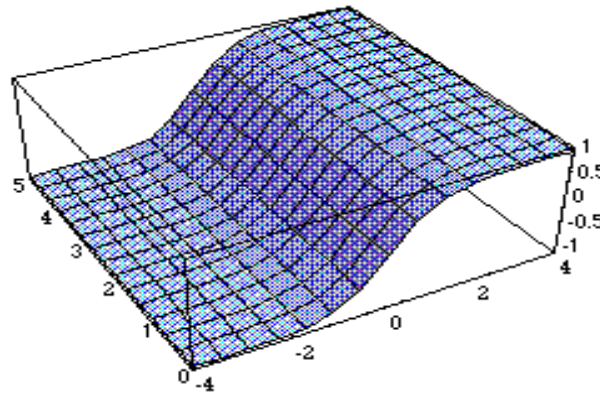
$\partial P / \partial y$ = derivata pe verticală

$|\nabla P|$ = mărimea gradientului



Detecția de contur

Un contur aflat în domeniul continuu $F(x,y)$ poate fi detectat prin calculul gradientului $G(x,y)$ de-a lungul normalei pe direcția conturului, ce formează un unghi θ cu axa orizontală.



Gradientul de-a lungul normalei la direcția conturului poate fi calculat cu ajutorul derivatelor pe două direcții ortogonale, conform ecuației:

$$G(x,y) = \frac{\partial F(x,y)}{\partial x} \cos \theta + \frac{\partial F(x,y)}{\partial y} \sin \theta \quad (5.1)$$

Detecția de contur

Prin generarea separată a gradientului pe linii $G_R(j,k)$ și a gradientului pe coloane $G_C(j,k)$, amplitudinea gradientului spațial este dată de:

$$G(j,k) = \{[G_R(j,k)]^2 + [G_C(j,k)]^2\}^{1/2} \quad (5.2)$$

iar orientarea referitoare la axa liniilor:

$$\theta(j,k) = \tan^{-1} \left\{ \frac{G_C(j,k)}{G_R(j,k)} \right\} \quad (5.4)$$

$$G_R(j,k) = F(j,k) - F(j,k-1) \text{ și } G_C(j,k) = F(j,k) - F(j+1,k) \quad (5.5)$$

Detecția de contur

Detectorul lui Roberts:

$$G(j, k) = |G_1(j, k)| + |G_2(j, k)| \quad (5.6a)$$

sau a rădăcinii pătrate:

$$G(j, k) = \{[G_1(j, k)]^2 + [G_2(j, k)]^2\}^{1/2} \quad (5.6b)$$

unde:

$$G_1(j, k) = F(j, k) - F(j + 1, k - 1) \text{ si } G_2(j, k) = F(j, k + 1) - F(j + 1, k) \quad (5.7)$$

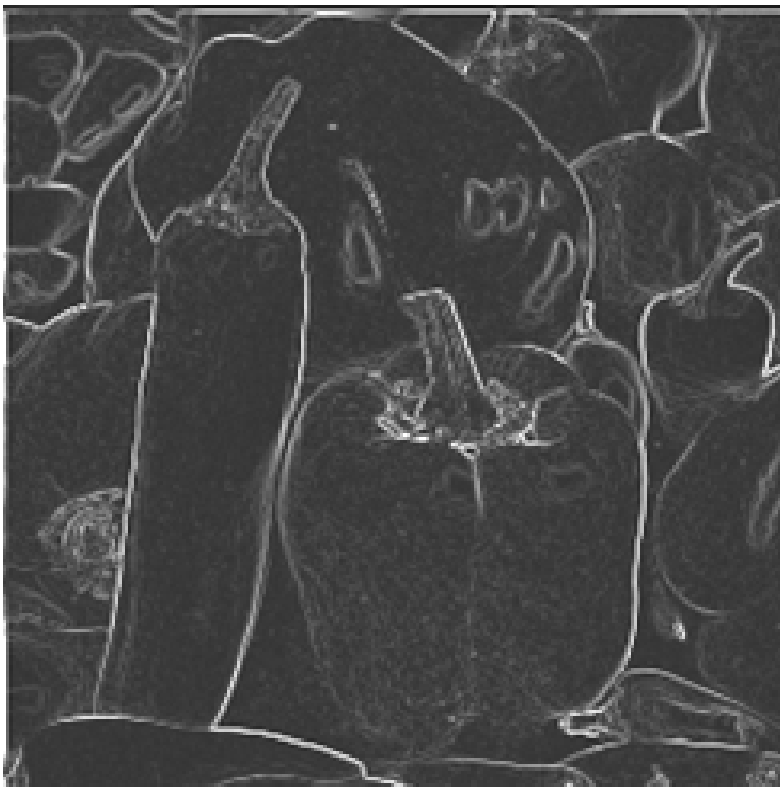
Orientarea conturului referitor la axa liniilor este:

$$\theta(j, k) = \frac{\pi}{4} \tan^{-1} \left\{ \frac{G_2(j, k)}{G_1(j, k)} \right\} \quad (5.8)$$

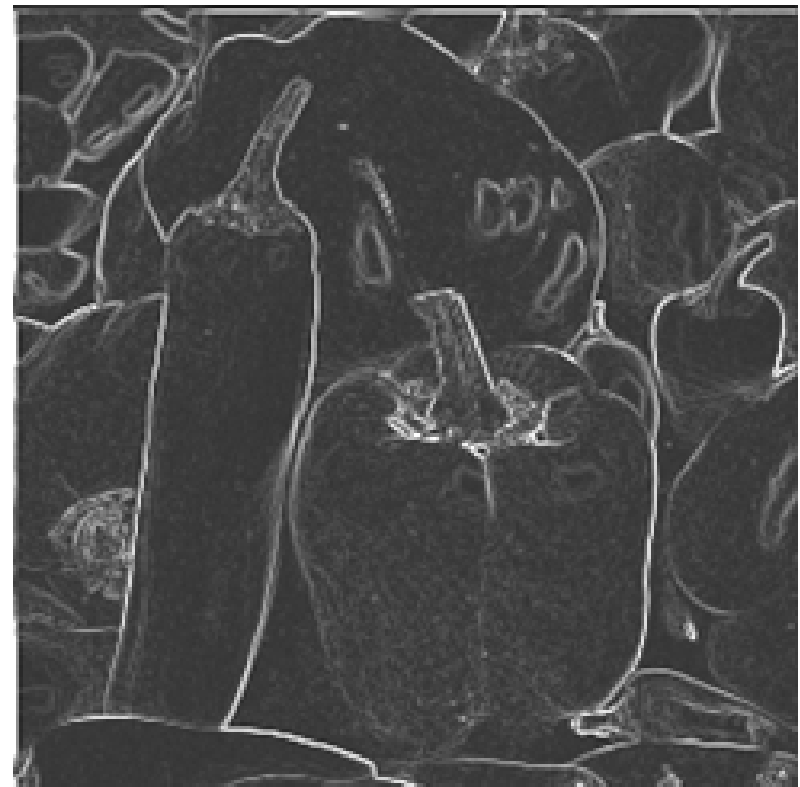
Detecția de contur

-1	0	0	-1
0	1	1	0

Roberts



a) amplitudine



b) rădăcină pătrată

Detecție de contur cu gradientul lui Roberts

Detecția de contur

Detectorul lui Prewitt

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

Prewitt



Detecția de contur

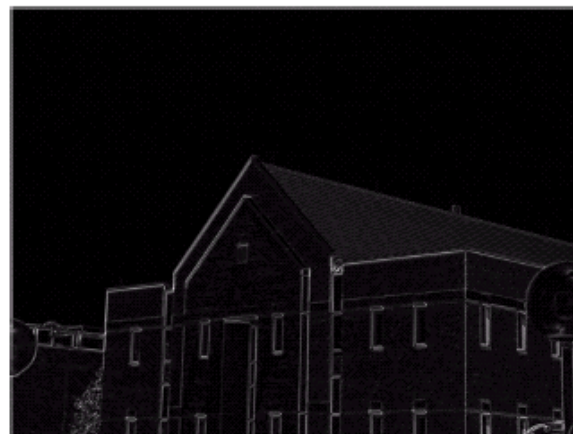
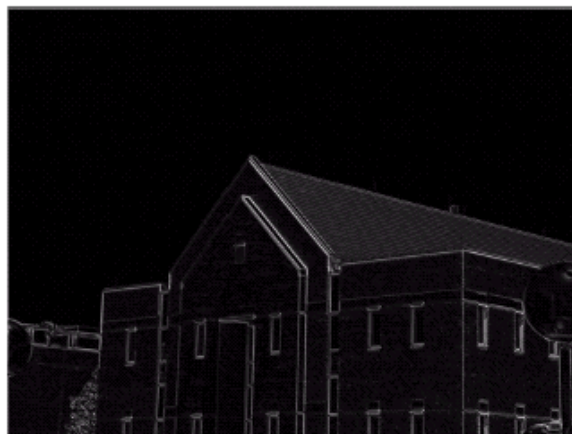
Detectorul lui Sobel

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

Sobel

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

Sobel



Detecția de contur

Generalizare:

$$G_R(j, k) = \frac{1}{K+2} [(A_2 + KA_3 + A_4) - (A_0 + KA_7 + A_6)] \quad (5.9)$$

și

$$G_C(j, k) = \frac{1}{K+2} [(A_0 + KA_1 + A_2) - (A_6 + KA_5 + A_4)] \quad (5.10)$$

Prewitt $K=1$.

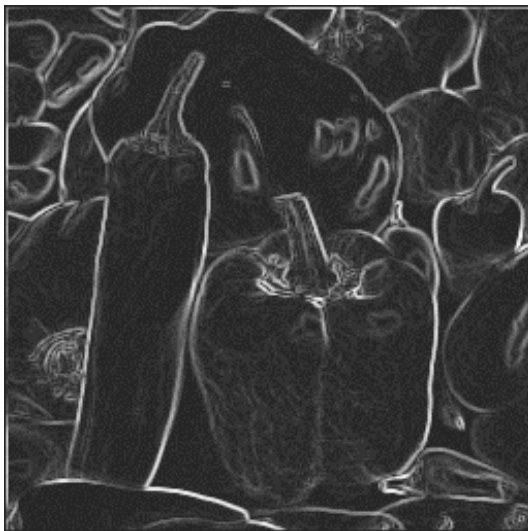
detectorul de contur al lui Sobel $K=2$.

Frei si Chen $K=\sqrt{2}$

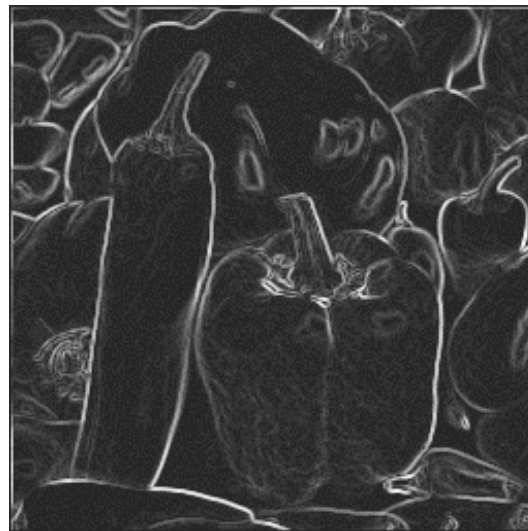
A_0	A_1	A_2
A_3		A_4
A_5	A_7	A_6

Convenția de numerotare a vecinilor unui pixel

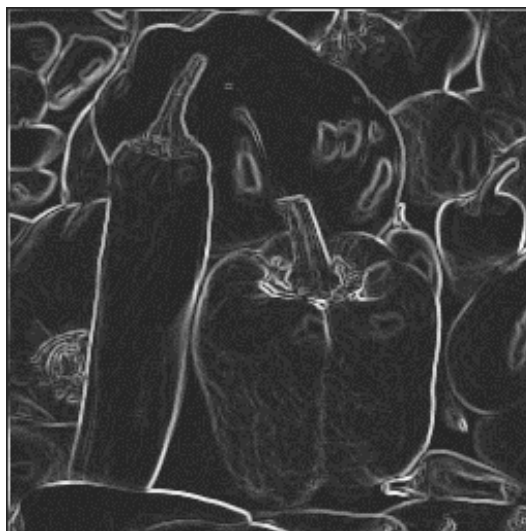
Detecția de contur



a) Prewitt



b) Sobel



c) Frei-Chen

Detecția de contur

Operatori compas (8 directii)

$$\begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & -1 \\ 0 & -1 & -1 \end{bmatrix} \quad \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} \quad \begin{bmatrix} 0 & 1 & 1 \\ -1 & 0 & 1 \\ -1 & -1 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix} \quad \begin{array}{c} \nearrow \\ \uparrow \\ \nwarrow \\ \rightarrow \\ \downarrow \\ \swarrow \\ \leftarrow \\ \searrow \end{array} \quad \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 0 & -1 & -1 \\ 1 & 0 & -1 \\ 1 & 1 & 0 \end{bmatrix} \quad \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad \begin{bmatrix} -1 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

Se alege cel de
amplitudine maxima

$$q(m, n) = \max_k \{ |g_k(m, n)| \}$$

Detecția de contur

Un operator de tip Prewitt de mărime 7x7 pentru calculul gradientului pe orizontală (boxcar), va avea forma:

$$H_R = \frac{1}{21} \begin{bmatrix} 1 & 1 & 1 & 0 & -1 & -1 & -1 \\ 1 & 1 & 1 & 0 & -1 & -1 & -1 \\ 1 & 1 & 1 & 0 & -1 & -1 & -1 \\ 1 & 1 & 1 & 0 & -1 & -1 & -1 \\ 1 & 1 & 1 & 0 & -1 & -1 & -1 \\ 1 & 1 & 1 & 0 & -1 & -1 & -1 \\ 1 & 1 & 1 & 0 & -1 & -1 & -1 \end{bmatrix} \quad (5.11)$$



Detectia de contur bazata pe filtrul Gabor

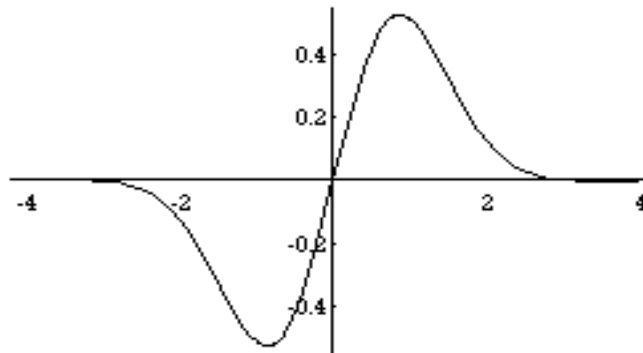
1. Introducere

2. Filtrul impar al lui Gabor

expresia generala: $s(x) = \exp\left[-\frac{1}{2}\left(\frac{x-x_0}{\sigma}\right)^2 + j\omega_x x\right]$ (1)

componenta para: $s_e(x) = \exp\left[-\frac{1}{2}\left(\frac{x}{\sigma}\right)^2\right] \cos(\omega_x x)$ (2)

componenta impara: $s_o(x) = \exp\left[-\frac{1}{2}\left(\frac{x}{\sigma}\right)^2\right] \sin(\omega_x x)$ (3)



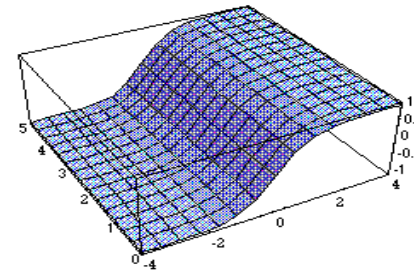
Detectia de contur bazata pe filtrul Gabor

3. Detectia modificarii de intensitate

treapta de intensitate $\Leftrightarrow$ contur

raspunsul filtrului impar: $R(x) = s_o(x) * f(x)$

derivata a doua:



$$R(x) = -\frac{x}{\sigma^2} \exp\left[-\frac{1}{2}\left(\frac{x}{\sigma}\right)^2\right] \sin(\omega_x x) + \omega_x \exp\left[-\frac{1}{2}\left(\frac{x}{\sigma}\right)^2\right] \cos(\omega_x x) \quad (4)$$

maxime locale la $x = n\pi / \omega_x$

rastringem actiunea filtrului de la $-\pi / \omega_x$ la π / ω_x

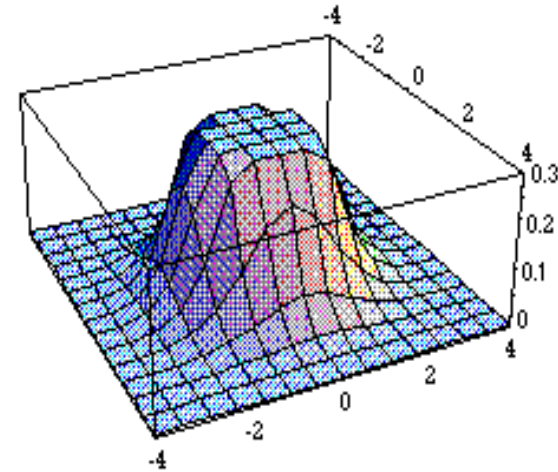
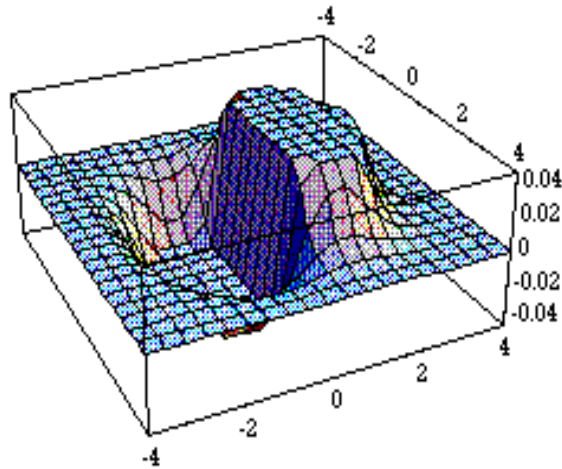
realizam un filtraj cu o functie gaussiana de ecart $2\pi\sigma$

atunci rezulta conditia $\frac{2\pi}{\omega_x} = 2\pi\sigma$ sau $\omega_x \sigma = 1$

Detectia de contur bazata pe filtrul Gabor

filtrul impar orientat cu unghiul θ :
$$s_o(x, y) = \exp\left[-\frac{1}{2}\left(\frac{x^2 + y^2}{\sigma^2}\right)\right] \sin[\omega(x \cos \theta + y \sin \theta)] \quad (4)$$

pentru $\theta = \pi/2$:
$$s_{o1}(x, y) = \exp\left[-\frac{1}{2}\left(\frac{x^2 + y^2}{\sigma^2}\right)\right] \sin(\omega y) \quad (5)$$



0	0	0	-1	-1	0	1	1	0	0	0
0	0	-1	-6	-8	0	8	6	1	0	0
0	-1	-9	-34	-48	0	48	34	9	1	0
0	-2	-30	-120	-166	0	166	120	30	2	0
0	-4	-64	-254	-352	0	352	254	64	4	0
1	-6	-82	-32	-452	0	452	326	82	6	-1
0	-4	-64	-254	-352	0	352	254	64	4	0
0	-2	-30	-120	-166	0	166	120	30	2	0
0	-1	-9	-34	-48	0	48	34	9	1	0
0	0	-1	-6	-8	0	8	6	1	0	0
0	0	0	-1	-1	0	1	1	0	0	0

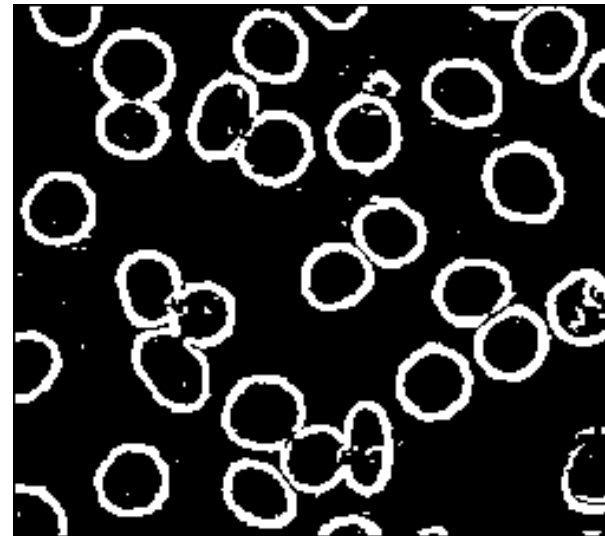
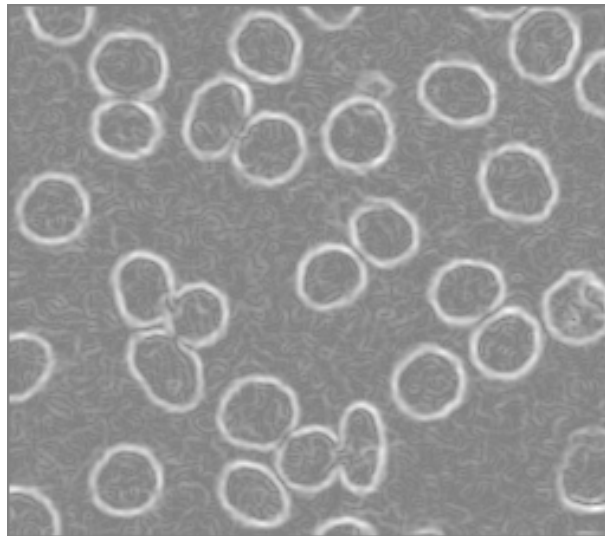
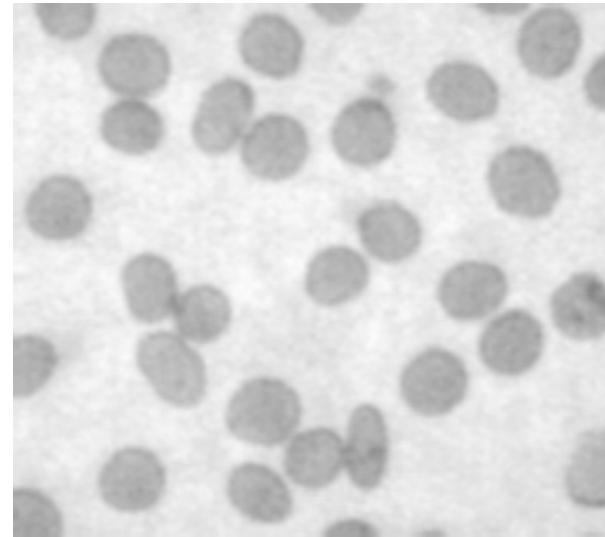
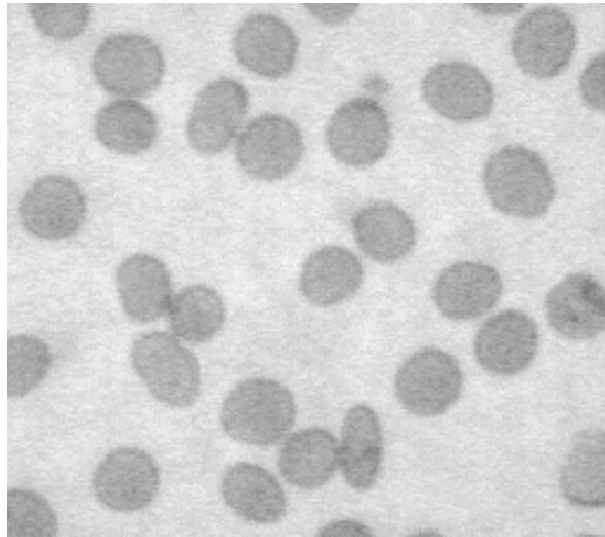
Detectia de contur bazata pe filtrul Gabor

5. Implementare

1. Netezirea imaginii printr-un filtru Gaussian de latime σ
2. Pentru fiecare pixel al imaginii
 - 2'. Determinarea printr-o metoda de gradient a directiei θ a conturului
 - 2''. Determinarea mastii de convolutie pentru un filtru Gabor orientat dupa directia θ
 - 2'''. Calculul convolutiei imaginii cu masca determinata
3. Normalizarea nivelelor de gri obtinute

Detectia de contur bazata pe filtrul Gabor

6. Rezultate



Detecția de contur - Laplacianul

Laplacian utilizat in estimarea contururilor unei imagini:

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

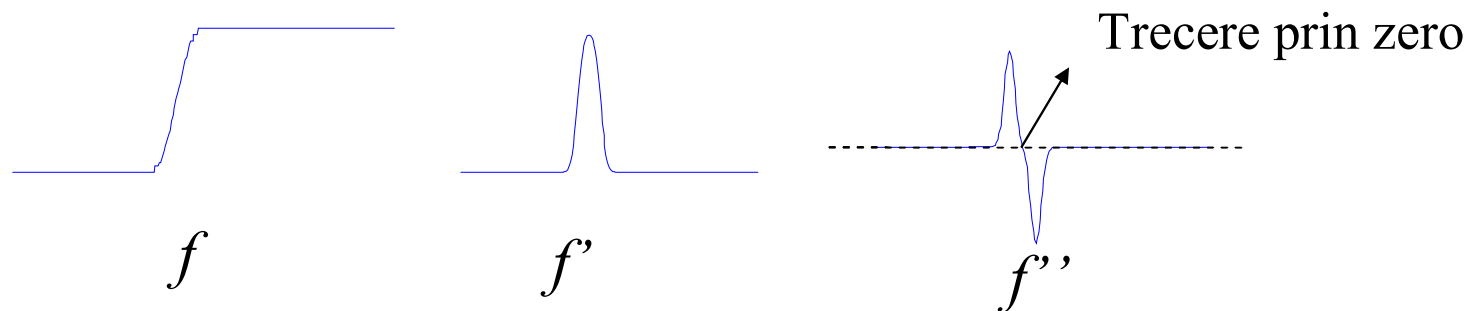
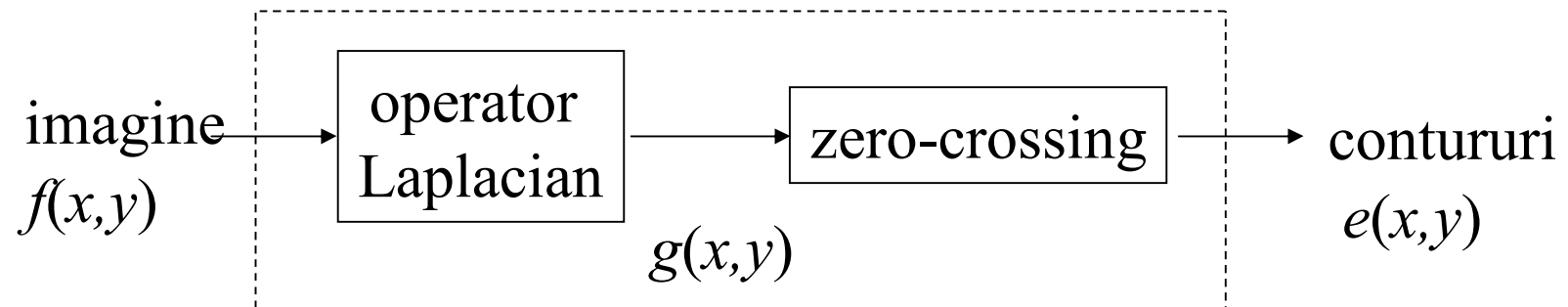
Mastile trebuie sa fie invariante la directie: simetrie radiala

Utilizare: contururi subtiri, contururi inchise

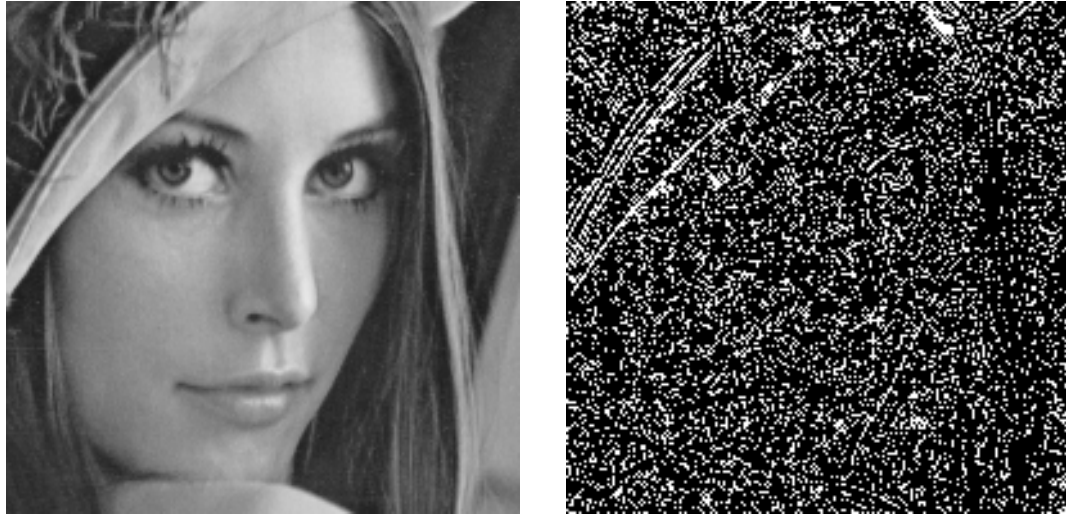
0	-1	0	-1	-1	-1
-1	4	-1	-1	8	-1
0	-1	0	-1	-1	-1

Detecția de contur - Laplacianul

Datorita derivatei a doua, operatorul este urmat de o detectie a trecherilor prin zero



Detecția de contur - Laplacianul

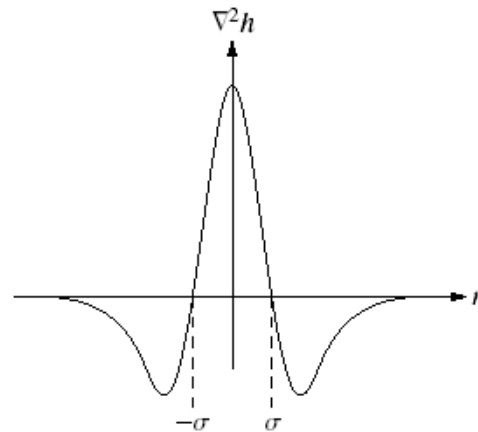
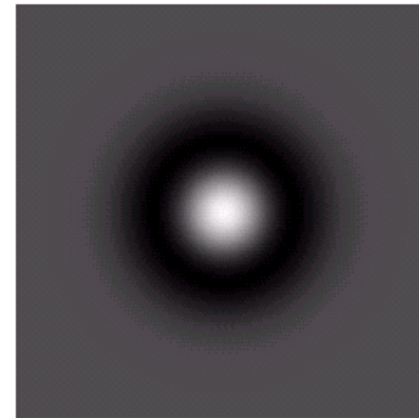
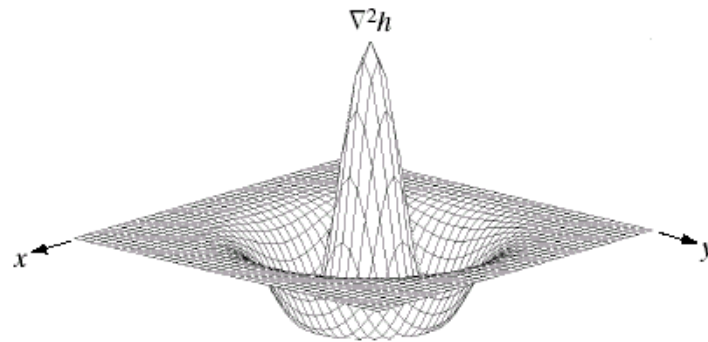


- Deoarece este foarte sensibil la zgomot, se folosește un operator compus
- o netezire Gaussiană urmată de Laplacian $\Rightarrow$ LoG (Laplacian of Gaussian)

$$\nabla^2 G(x, y) = - \left[\frac{x^2 + y^2 - \sigma^2}{\sigma^4} \right] e^{-\left(\frac{x^2 + y^2}{2\sigma^2} \right)}$$

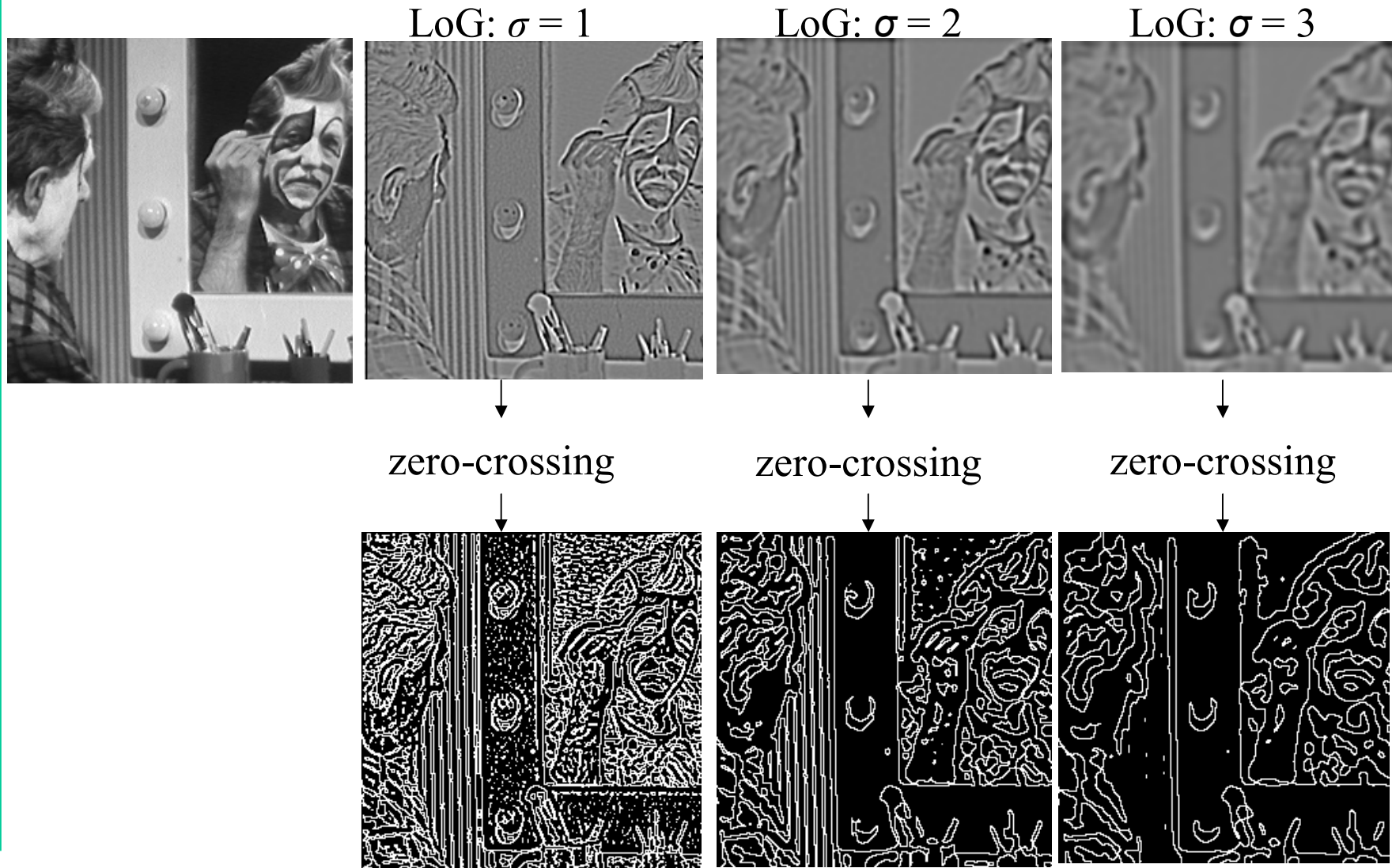
Detecția de contur - Laplacianul

Laplacian of Gaussian (LoG)

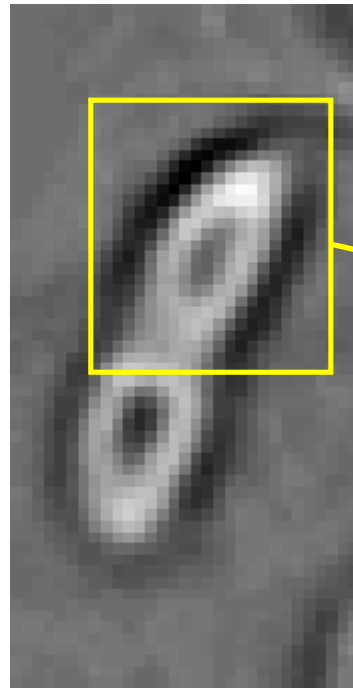
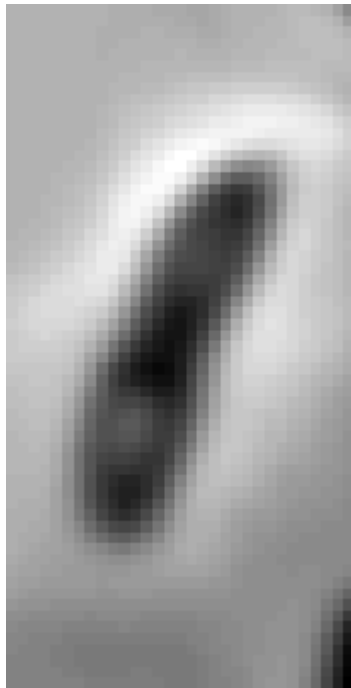


0	0	-1	0	0
0	-1	-2	-1	0
-1	-2	16	-2	-1
0	-1	-2	-1	0
0	0	-1	0	0

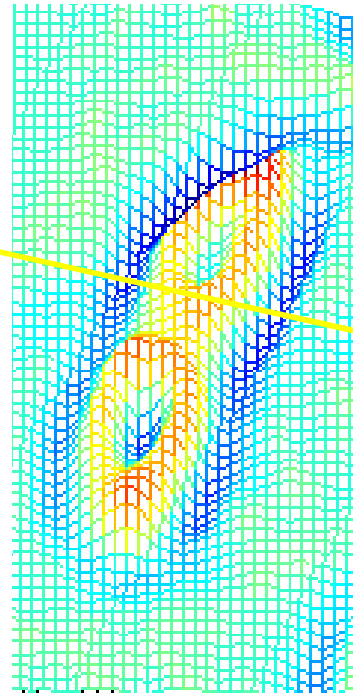
Detecția de contur - Laplacianul



Detecția de contur - Laplacianul

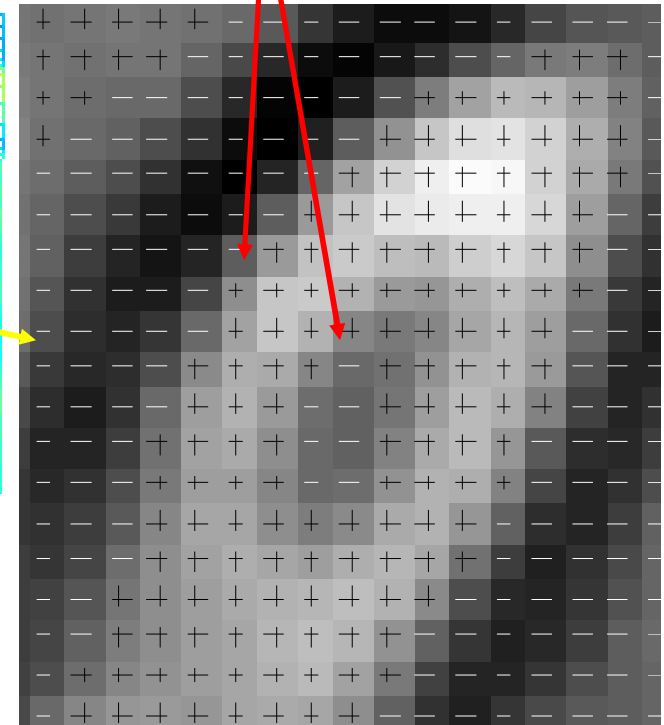


$\nabla^2 P$



$\nabla^2 P$

Trecere prin zero





Elemente de procesare de nivel inalt

Segmentarea

Tehnici de segmentare a imaginilor

Introducere

Segmentarea imaginii - a împărți imaginea în regiuni ce au o corelație puternică cu obiecte sau suprafețe din lumea reală conținute în imagine

segmentare completă - ce rezultă într-un set de regiuni disjuncte ce corespund în mod unic cu obiectele din imagine

segmentare parțială - în care regiunile nu corespund direct cu obiectele din imagine, imaginea este împărțită în regiuni disjuncte ce sunt omogene relativ la o anumită proprietate cum ar fi luminozitate, culoare, context, reflectivitate, etc.

Tehnici de segmentare de imagine pot fi clasificate în patru categorii:

1. filtrare locală și fixare de praguri (*threshold*)
2. algoritmi *snake* și balon, algoritmi *watershed*
3. tehnici de *region growing* și *split and merge*
4. metode de optimizare globală bazate pe funcții energie, bayesiene, etc.

Segmentare prin thresholding (binarizare)

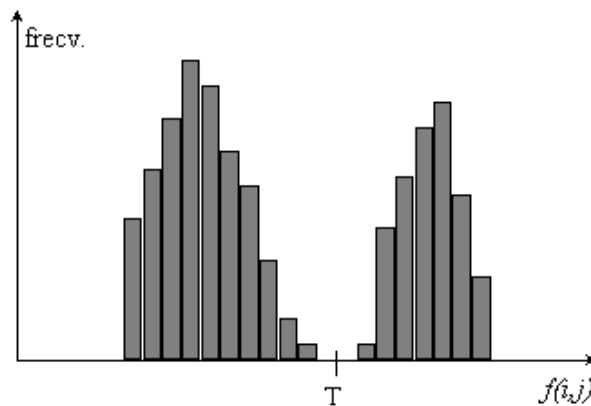
Fixarea pragului de binarizare

O segmentare completă a unei imagini I este o multime finită de regiuni $R_1, \dots, R_s$

$$I = \bigcup_{i=1}^s R_i \quad R_i \cap R_j = \emptyset \quad i \neq j \quad (2.1)$$

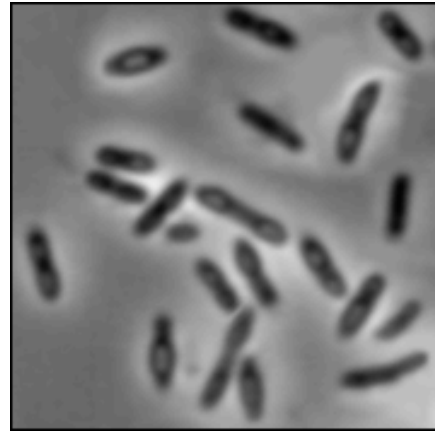
Binarizarea este operația de transformare a unei imagini de intrare f într-o imagine de ieșire binară g (segmentată) după cum urmează:

$$g(i, j) = \begin{cases} 1 & \text{pentru } f(i, j) \geq T \\ 0 & \text{pentru } f(i, j) < T \end{cases} \quad (2.2)$$

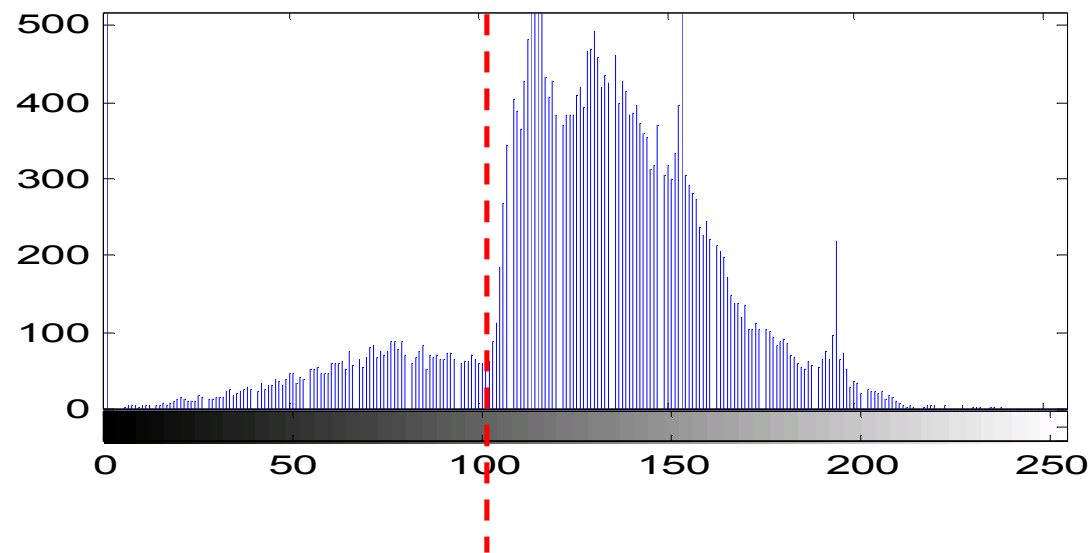


Histograma bimodală și modul de alegere a pragului.

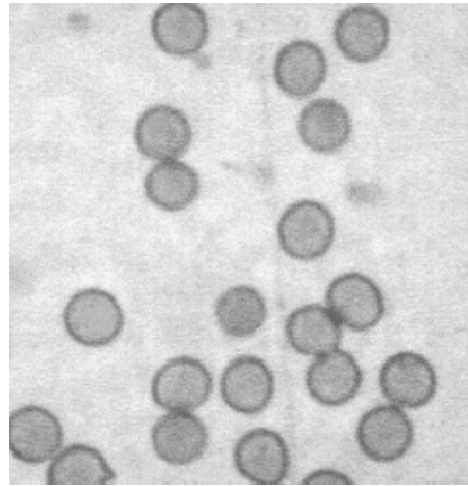
Segmentare prin thresholding (binarizare)



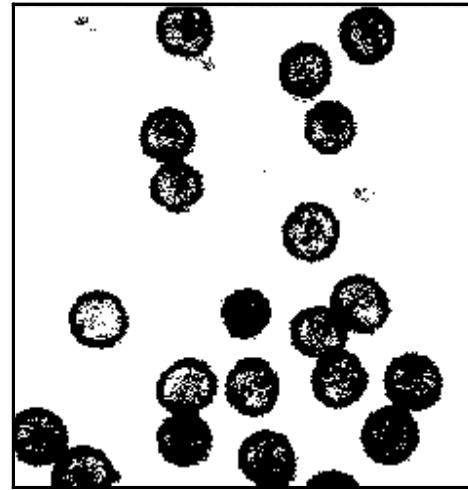
$T = 102$



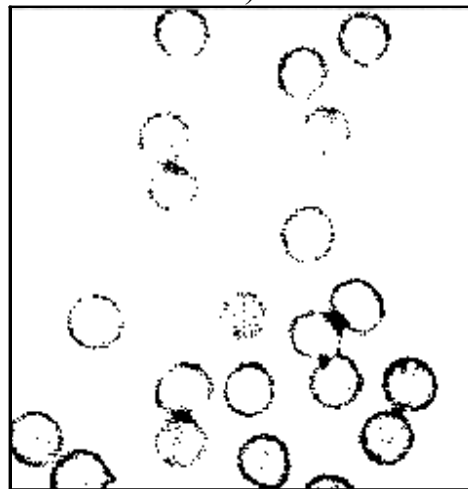
Segmentare prin thresholding (binarizare)



a)



b)



c)



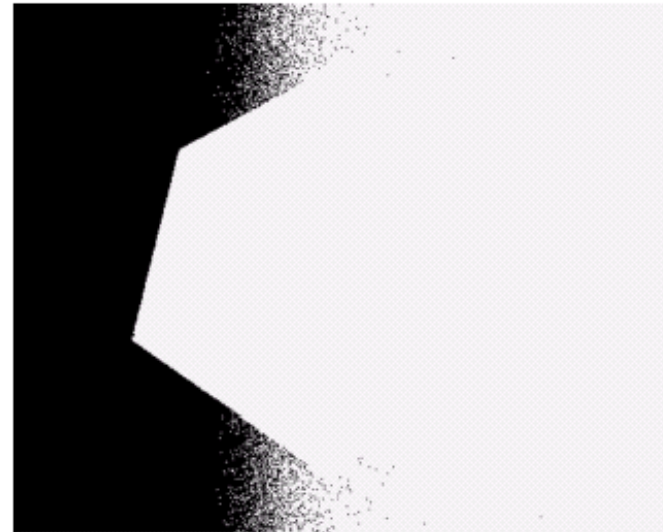
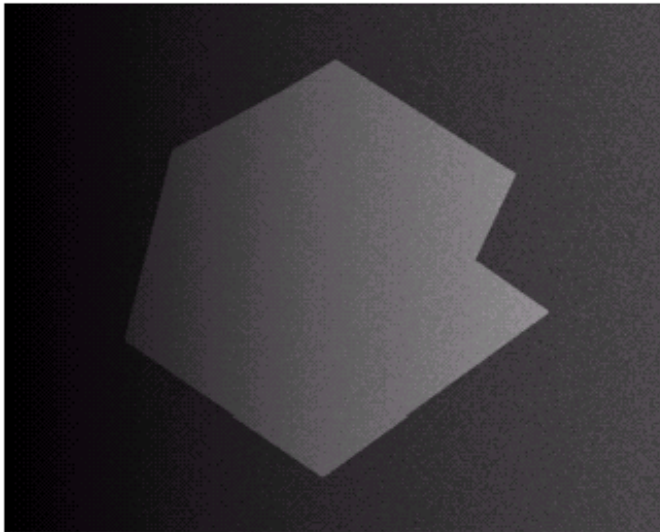
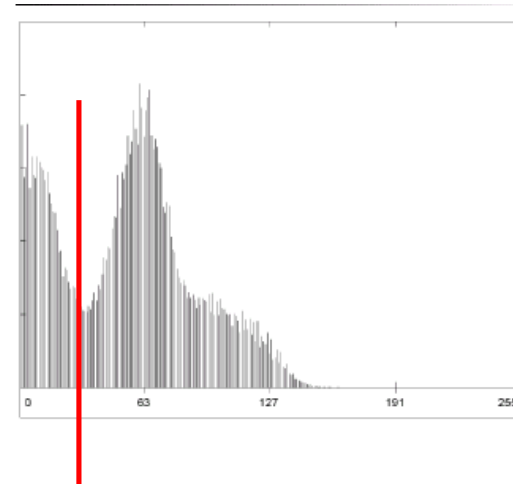
d)

Binarizarea unei imagini folosind diferite valori ale pragului. a) Imaginea originală, rezultatul binarizării cu valori de prag de b) 154, c) 180, d) 110

Segmentare prin thresholding (binarizare)

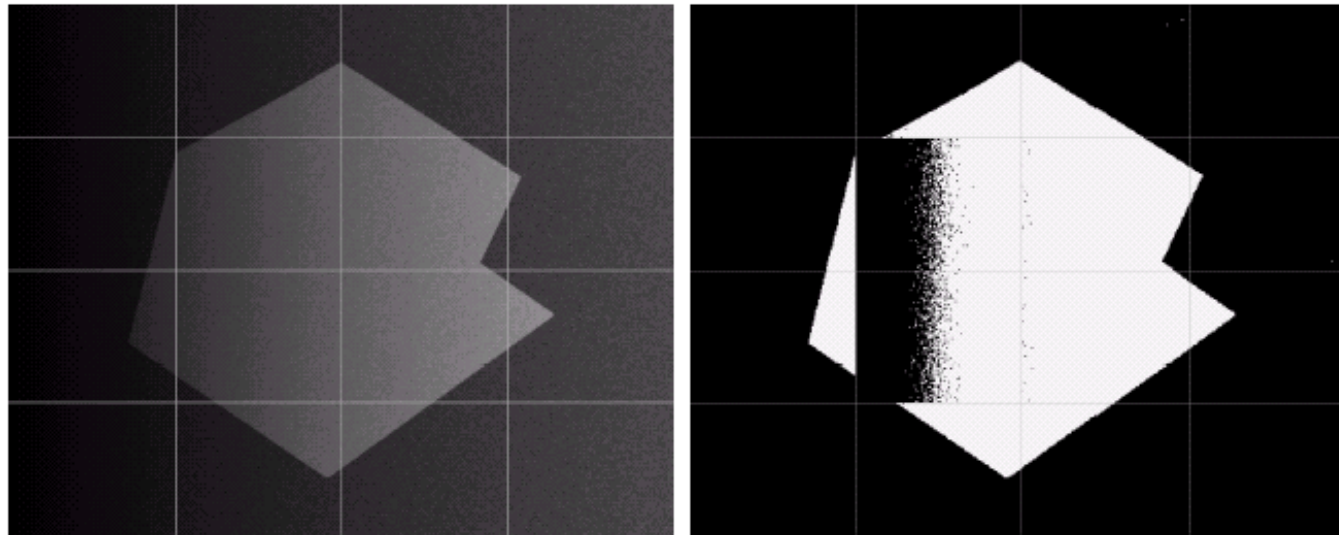
Problema !

Iluminarea neuniforma



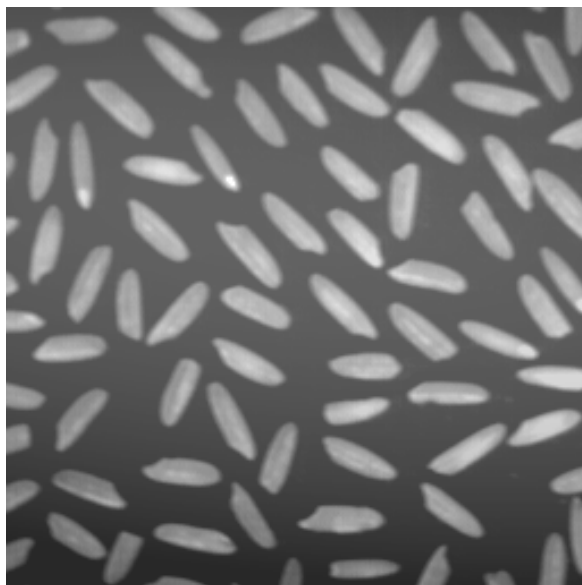
Segmentare prin thresholding (binarizare)

- Impartirea imaginii in subregiuni
- Utilizarea de praguri locale

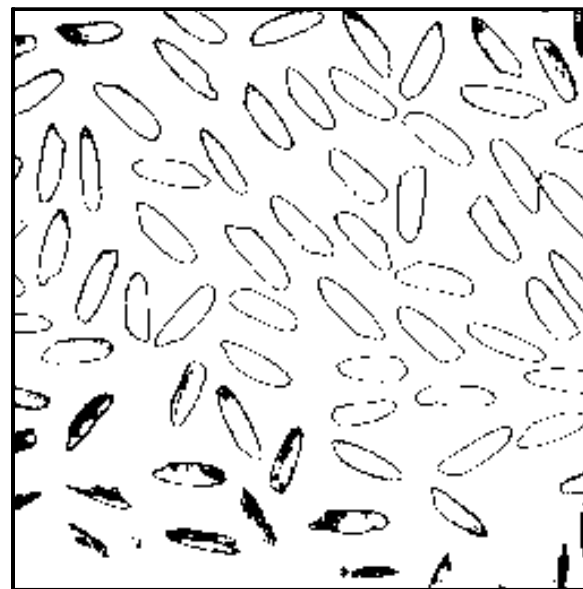


Segmentare în banda

Un prag pentru toată imaginea f este determinat de $T = T(f)$ pe când unul local este dat $T = T(f, f_c)$, unde f_c este partea din imagine pe care este determinat pragul.



a)



b)

Segmentarea în bandă; a) imaginea originală, b) rezultatul binarizării.

$$g(i, j) = \begin{cases} 1 & \text{pentru } f(i, j) \in D \\ 0 & \text{altfel} \end{cases} \quad (2.3)$$

Segmentare nesupervizata

Metoda iterativa pentru calculul pragului

Initializare: $T = \bar{x}$

REPEAT

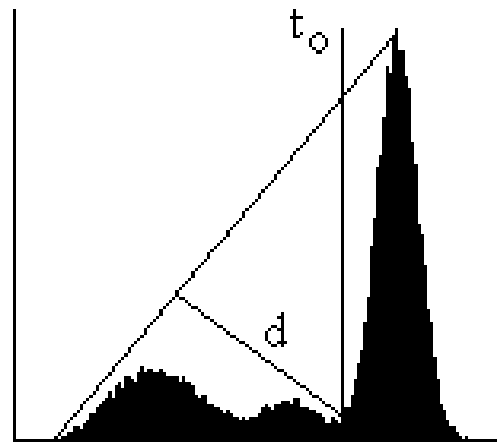
- Segmenteaza imaginea folosind T
- Calculeaza media segmentelor

$$\bar{x}_1 = \frac{1}{n_1} \sum_{x_i \leq T} x_i \quad \bar{x}_2 = \frac{1}{n_2} \sum_{x_i > T} x_i$$

- Recalculeaza pragul $T = \frac{\bar{x}_1 + \bar{x}_2}{2}$

UNTIL T nu se modifica

Segmentare nesupervizata



Algoritmul triunghiului

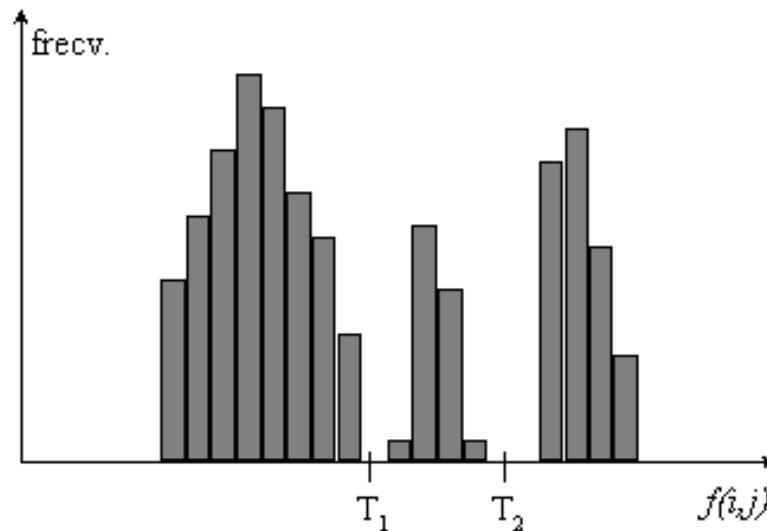
Algoritmul triunghiului. Aceasta constă în trasarea unei linii între cea mai mică valoare a nivelului de gri din imagine $b_{\min}$ și valoarea maximă a histogramei ce are loc pe nivelul $b_{\max}$. Se calculează distanța d dintre linia trasată și histograma, pentru toate nivelele dintre $b_{\min}$ și $b_{\max}$.

Praguri multiple de binarizare

segmentare *multipla* (*multilevel thresholding*),

găsirea a două praguri T_1 și T_2 astfel încât pixelul din imaginea originală va fi clasificat într-una din cele trei clase astfel:

$$g(i, j) = \begin{cases} t_1 & \text{pentru } f(i, j) \leq T_1 \\ t_2 & \text{pentru } T_1 < f(i, j) \leq T_2 \\ t_3 & \text{pentru } f(i, j) > T_2 \end{cases} \quad (2.4)$$



Histograma și modul de alegere a pragurilor multiple.

Criteriul de maximizare a entropiei

Să considerăm o imagine $f(x,y)$ cu dimensiunile $N \times N$ pixeli și având m nivele de gri. Fie $G_m \equiv \{0, 1, \dots, (m-1)\}$ mulțimea nivelelor de gri și $f_i \in G_m$ frecvențele nivelelor de gri observate în imagine. Probabilitatea nivelului de gri i în imaginea f poate fi calculată astfel :

$$p_i = \frac{f_i}{N \times N}, i \in G_m \quad (2.6)$$

Astfel se poate obține distribuția $\{ p_i \mid i \in G_m \}$. Pentru un nivel de gri s dat, dacă următoarea sumă îndeplinește inegalitatea $0 < \sum_{i=0}^{s-1} p_i < 1$, atunci următoarele două distribuții pot fi derivate din aceasta, după normalizare :

$$\begin{aligned} A &\equiv \left\{ \frac{p_0}{P(s)}, \frac{p_1}{P(s)} \dots \frac{p_{s-1}}{P(s)} \right\} \\ B &\equiv \left\{ \frac{p_s}{1 - P(s)}, \frac{p_{s+1}}{1 - P(s)} \dots \frac{p_{m-1}}{1 - P(s)} \right\} \end{aligned} \quad (2.7)$$

unde $P(s) = \sum_{i=0}^{s-1} p_i$ este probabilitatea totală până la nivelul de gri $(s-1)$.

Criteriul de maximizare a entropiei

Ideea ce stă la baza criteriului entropiei maxime este alegerea pragului astfel încât cantitatea totală de informație obținută din obiect și din fundal să fie maximizată. În cazul de față aceasta informație este măsurată pe baza entropiei asociate fiecărei din cele două distribuții A, B astfel:

$$E_A(s) = -\sum_{i=0}^{s-1} \frac{p_i}{P(s)} \ln\left(\frac{p_i}{P(s)}\right) = \ln(P(s)) + \frac{H(s)}{1 - P(s)} \quad (2.8)$$

$$E_B(s) = -\sum_{i=s}^{m-1} \frac{p_i}{1 - P(s)} \ln\left(\frac{p_i}{1 - P(s)}\right) = \ln(1 - P(s)) + \frac{H'(s)}{1 - P(s)} \quad (2.9)$$

unde $H(s) = -\sum_{i=0}^{s-1} p_i \cdot \ln(p_i)$, $H'(s) = -\sum_{i=s}^{m-1} p_i \cdot \ln(p_i)$ și $P(s) = \sum_{i=0}^{s-1} p_i$

Cum informația este măsurată de entropie, cantitatea totală de informație dată de A și B este :

$$\begin{aligned} TE(s) &= E_A(s) + E_B(s) = -\sum_{i=0}^{s-1} \left(\frac{p_i}{P(s)}\right) \ln\left(\frac{p_i}{P(s)}\right) - \sum_{i=s}^{m-1} \left(\frac{p_i}{1 - P(s)}\right) \ln\left(\frac{p_i}{1 - P(s)}\right) \\ &= \ln[P(s)(1 - P(s))] + \frac{H(s)}{P(s)} + \frac{H'(s)}{1 - P(s)} \end{aligned} \quad (2.10)$$

Criteriul de maximizare a entropiei

Criteriul maximei entropii presupune găsirea pragului s' astfel încât :

$$TE(s') = \max_{s \in G_m} TE(s) \quad (2.11)$$

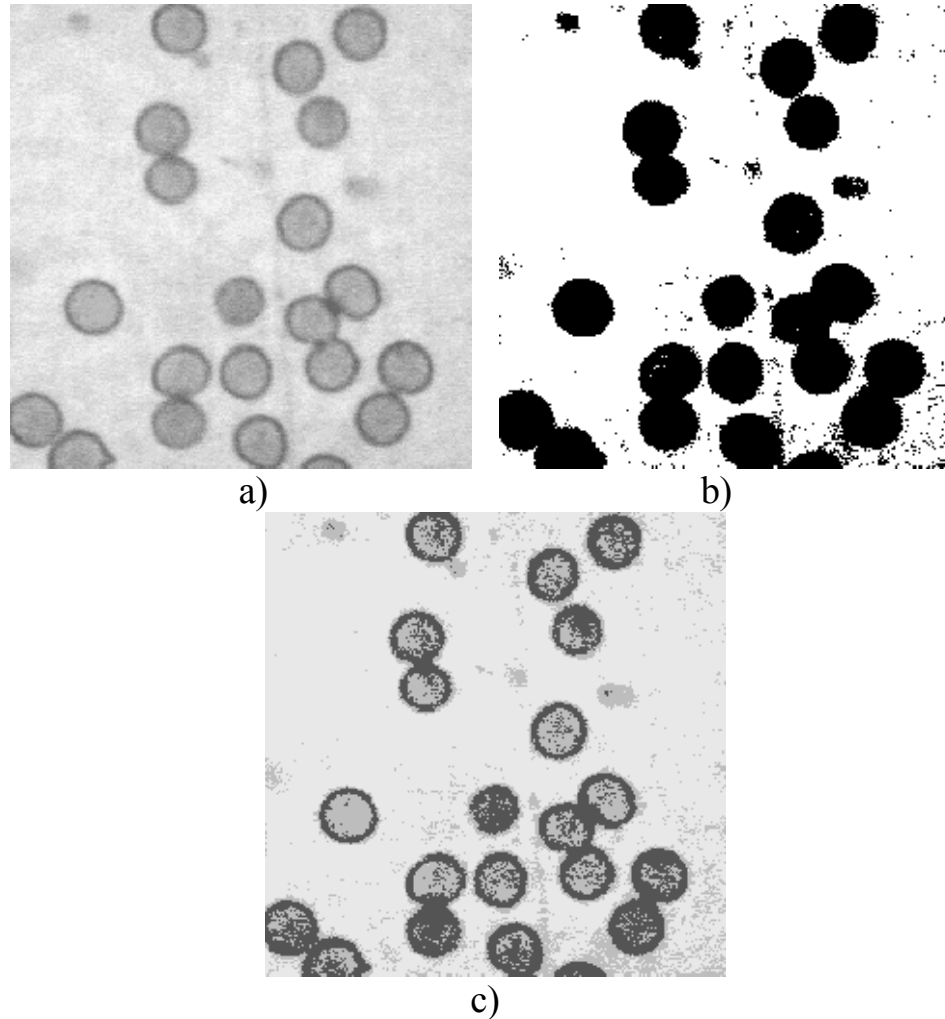
Acest algoritm poate fi extins și la segmentarea pe mai multe nivele. Dacă două sau mai multe obiecte se suprapun peste același fundal astfel încât histograma să fie multimodală, pe baza ecuațiilor de mai sus putem scrie :

$$TE(s_1, s_2, \dots, s_k) = \ln \left(\sum_{i=0}^{s_1-1} p_i \right) + \ln \left(\sum_{i=s_1}^{s_2-1} p_i \right) + \dots + \ln \left(\sum_{i=s_k}^{m-1} p_i \right) -$$
$$- \frac{\sum_{i=0}^{s_1-1} p_i \ln(p_i)}{\sum_{i=0}^{s_1-1} p_i} - \dots - \frac{\sum_{i=s_k}^{m-1} p_i \ln(p_i)}{\sum_{i=s_k}^{m-1} p_i} \quad (2.12)$$

unde $(s_1, s_2, \dots, s_k)$ este vectorul pragurilor dintre clase $(0 \leq s_1 \leq s_2 \leq \dots \leq s_k \leq m)$.

Ca și la binarizarea simplă, pragurile $s_1, s_2, \dots, s_k$ se vor alege astfel încât să maximizeze $TE(s_1, s_2, \dots, s_k)$

Criteriul de maximizare a entropiei



Binarizare prin metoda entropiei maxime; a) imaginea originală;
b) un singur prag (nivelul 203); c) două praguri (nivelele 169 și 210)

Aplicatie: Tracking Water Courses in Satellite Images

1. INTRODUCTION

2. SPOT IMAGES

XS3 images

- near infra-red, 0.79 - 0.90 μm
- ground resolution of 20x20 m per pixel and a ground dimension of 60x60 km

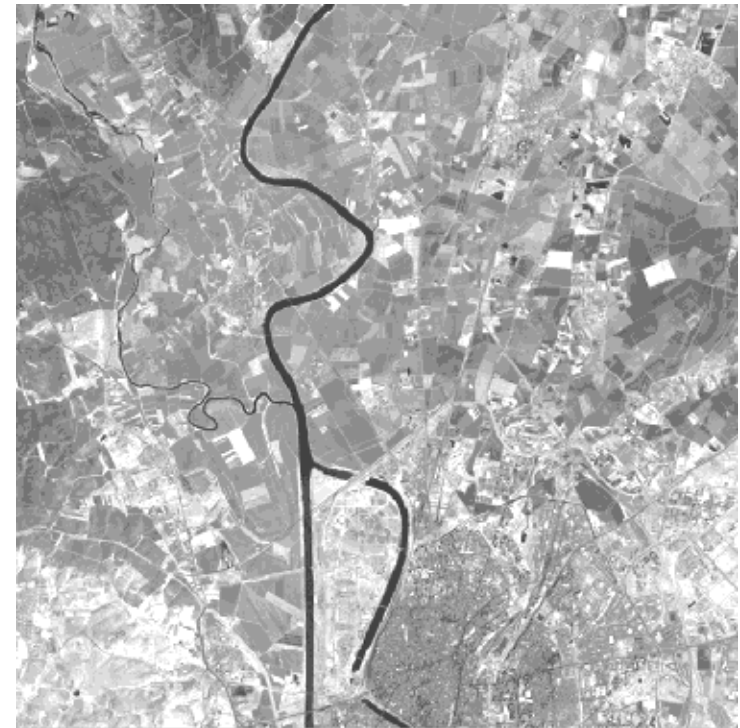


Fig.1. SPOT XS3 image of the city of Seville in Spain

Aplicatie: Tracking Water Courses in Satellite Images

3. THE FRAMEWORK

Our framework will have the following steps:

1. pre-processing

- a greyscale transform of the image if needed
- a histogram smoothing or any other smoothing filter

2. threshold

- thresholding the image, by fixing the level such as target object will be emphasised

3. recognition

- “noise” cleaning
- based on morphological properties, extract all object that match

Aplicatie: Tracking Water Courses in Satellite Images



Fig.2. Original image of Freiburg-Germany

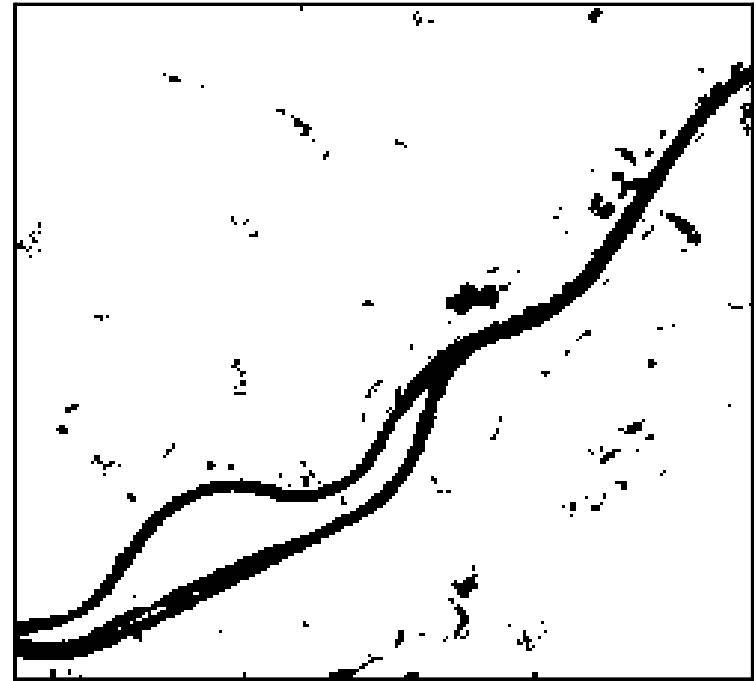


Fig.3. Output of the MEC threshold method

Aplicatie: Tracking Water Courses in Satellite Images

5. RECOGNITION

- noise cleaning by erasing all pixel groups that fit in a 5 pixel width square
- top-to-bottom scanning, searching and marking every zone of black pixels
- computes the area of the selected zone
 - if too small
 - the zone will be erased
 - else
 - a grey level will be conferred to all of its pixels
- top-to-bottom seeking, for extracting morphological measures by polygonal approximation.
 - if the ratio of the two best fit polygon diagonals are in a certain range
 - decides that a water course is present

Aplicatie: Tracking Water Courses in Satellite Images

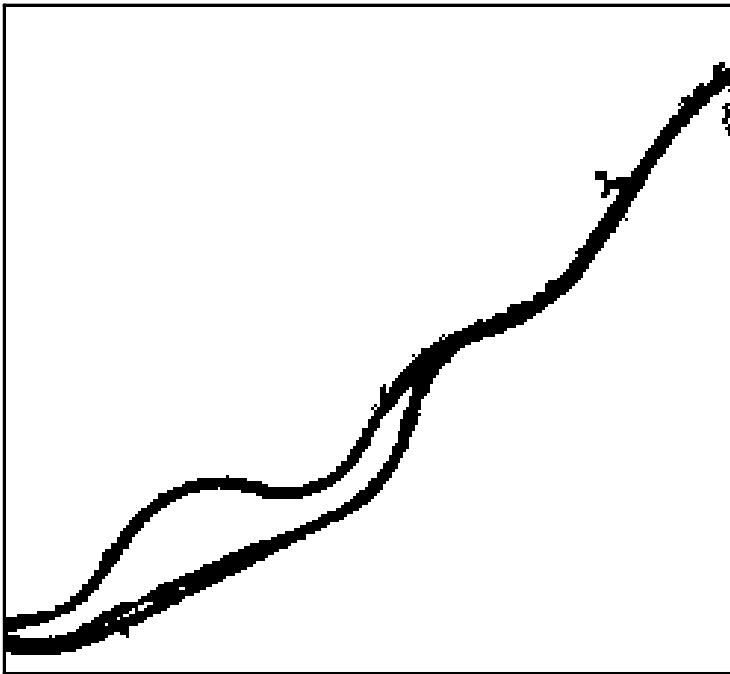


Fig.4. Output of the first recognition step

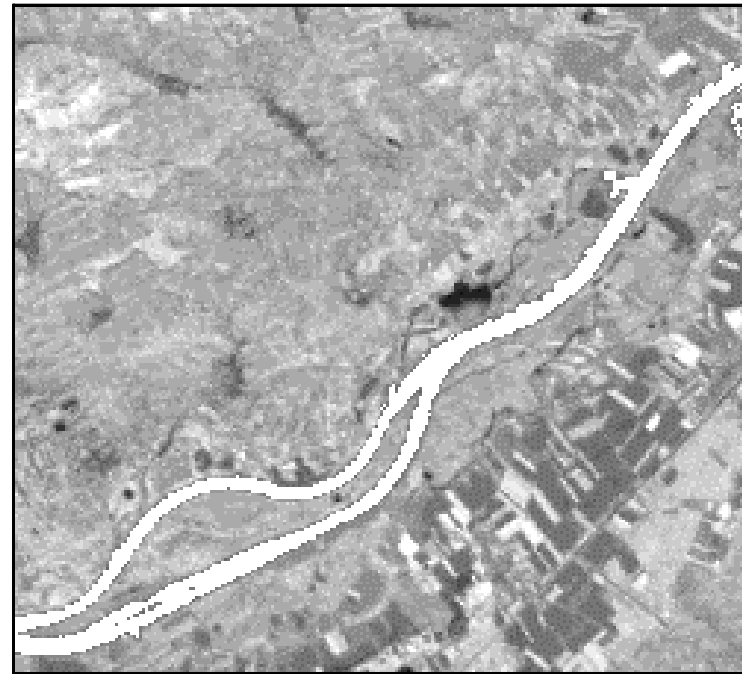


Fig.5. Output of the second recognition step

Aplicatie: Tracking Water Courses in Satellite Images

5. CONCLUSIONS

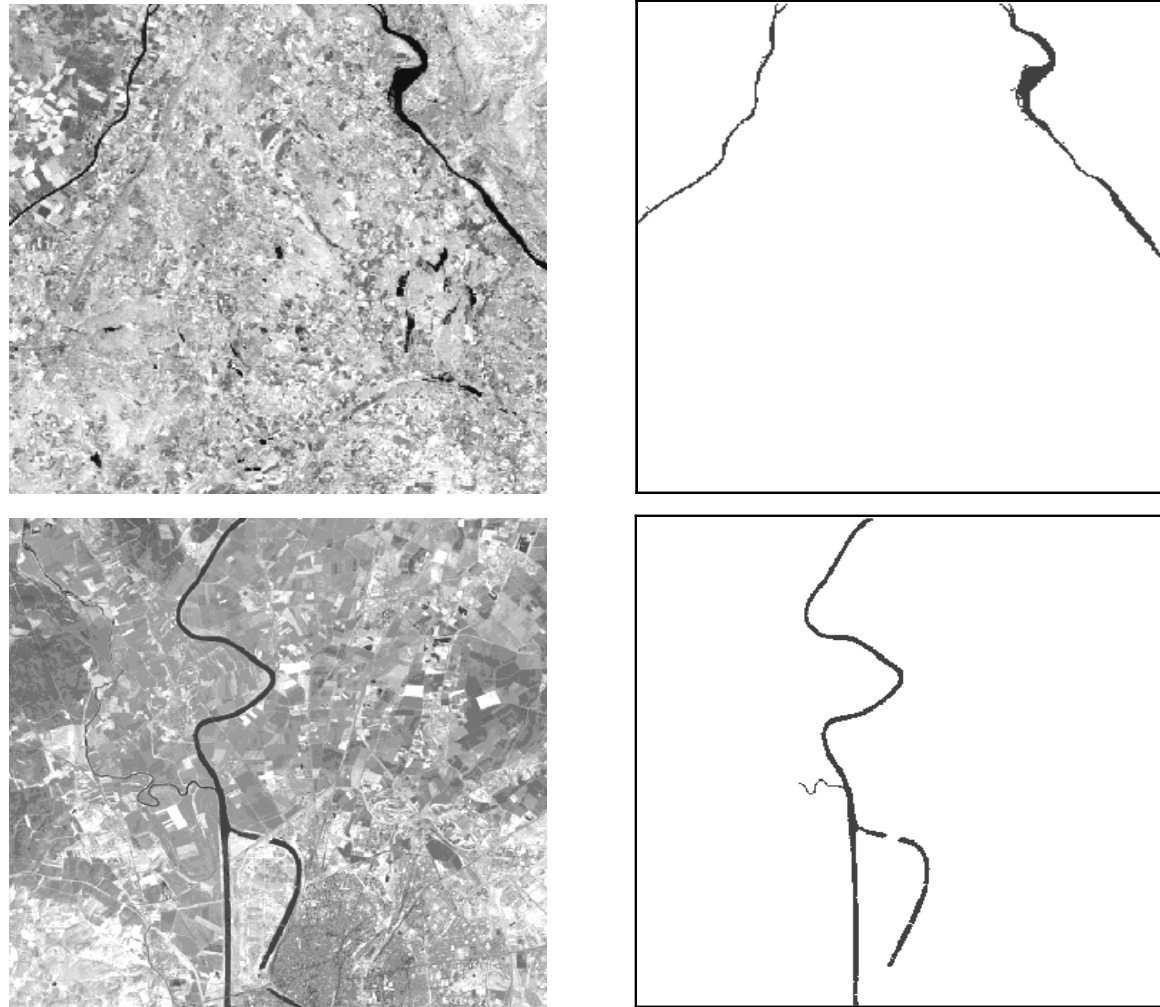


Fig.7. Original SPOT image and resulting water course mask.

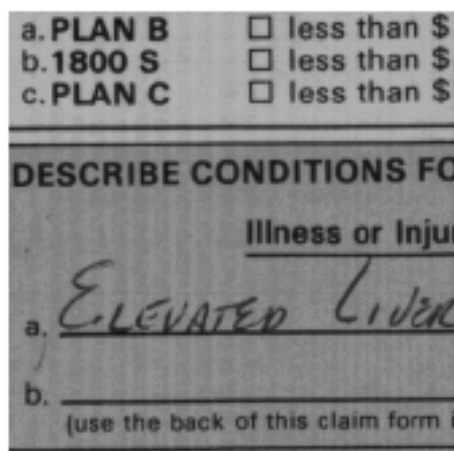
Algoritmul *Adaptive Threshold Processor* (ATP)

A fost dezvoltat de Y. Lee, J. Basile și P. Rudak și a fost brevetat de compania Eastman Kodak.

Trecerea de la un prag local la unul global se face dacă pixelul procesat se află sau nu lângă un contur. Acest lucru se determină prin compararea gradientului local cu un gradient de prag fixat G . Gradientul local netezit L se calculează prin medierea operatorului lui Sobel pe o fereastră centrată pe pixel.

Mărimea ferestrei este dependentă de zgomotul prezent în imagine. Dacă $L > G$, se presupune că pixelul se află în vecinătatea unui contur și se utilizează un prag local de binarizare. Valoarea pragului local se calculează ca media dintre valoarea maximă și minimă a nivelului de gri dintr-o a doua fereastră centrată pe pixelul procesat.

Algoritmul *Adaptive Threshold Processor* (ATP)

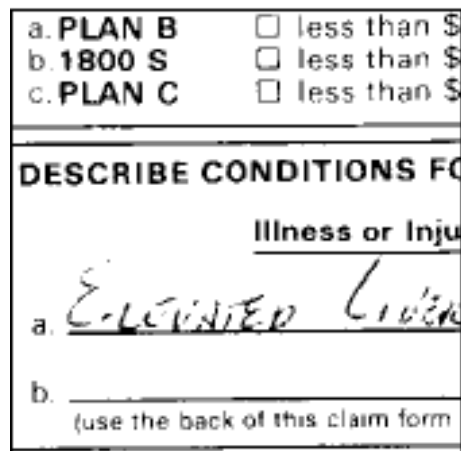


a. PLAN B ☐ less than \$
b. 1800 \$ ☐ less than \$
c. PLAN C ☐ less than \$

DESCRIBE CONDITIONS FOR
Illness or Injury

a. ELEVATED LIVER
b. _____
(use the back of this claim form)

a)

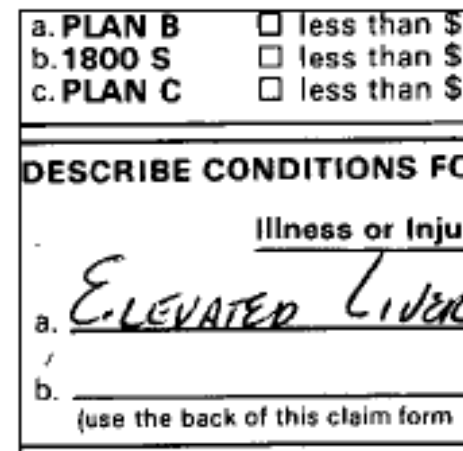


a. PLAN B ☐ less than \$
b. 1800 \$ ☐ less than \$
c. PLAN C ☐ less than \$

DESCRIBE CONDITIONS FOR
Illness or Injury

a. ELEVATED LIVER
b. _____
(use the back of this claim form)

b)

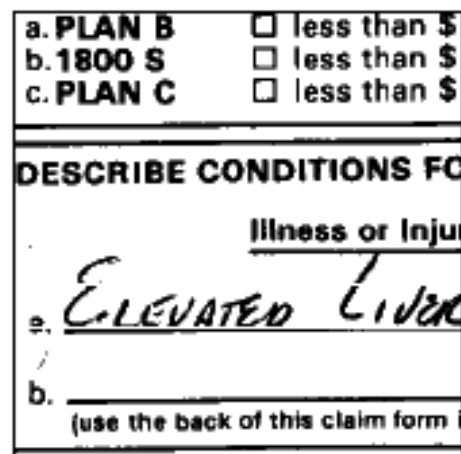


a. PLAN B ☐ less than \$
b. 1800 \$ ☐ less than \$
c. PLAN C ☐ less than \$

DESCRIBE CONDITIONS FOR
Illness or Injury

a. ELEVATED LIVER
b. _____
(use the back of this claim form)

c)

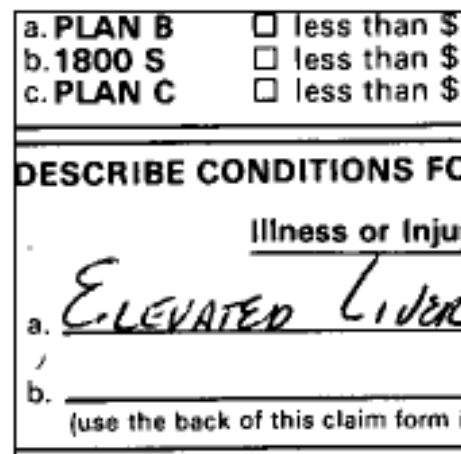


a. PLAN B ☐ less than \$
b. 1800 \$ ☐ less than \$
c. PLAN C ☐ less than \$

DESCRIBE CONDITIONS FOR
Illness or Injury

a. ELEVATED LIVER
b. _____
(use the back of this claim form)

d)



a. PLAN B ☐ less than \$
b. 1800 \$ ☐ less than \$
c. PLAN C ☐ less than \$

DESCRIBE CONDITIONS FOR
Illness or Injury

a. ELEVATED LIVER
b. _____
(use the back of this claim form)

e)

Comparație între diferite metode de segmentare, după [Savakis 98].

a) imaginea originală; segmentare cu b) prag fix, c) ATP, d) FBC, e) combinație FBC cu ATP.

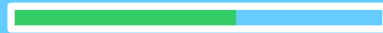
Algoritmul *Adaptive Threshold Processor* (ATP)

Metoda *Foreground and Background Clustering* (FBC) este utilizată în binarizarea imaginilor și se bazează pe atribuirea fiecărui pixel unei clase, a fundalului sau a obiectului. Clasificarea pixelilor se face cu o variantă a algoritmului *K-means* [Jain 88]

1. **Selecția regiunii.** Se împarte imaginea în subregiuni distincte și care se pot include una în cealaltă. Se alege o subregiune pentru care se va calcula pragul și o regiune ce o cuprinde pentru care se va determina cărei clase îi aparține subregiunea.
2. **Inițializarea.** Se inițializează centrul clasei fundal și a clasei obiect cu valorile calculate pentru subregiunea anterioară. Dacă nu a existat o subregiune anterioară, se dau valori diferite celor două. Pentru fiecare pixel din regiune, se iterează pașii 3 și 4:
3. **Atribuirea pixelului.** Se atribuie fiecare pixel, clasei celei mai apropiate.
4. **Calcularea centrului clasei.** După fiecare nouă atribuire, se recalculează centrul clasei.
5. **Calculul pragului.** După ce toți pixelii din regiune au fost atribuiți, se calculează valoarea de prag a subregiunii ca fiind media centrelor celor două clase.

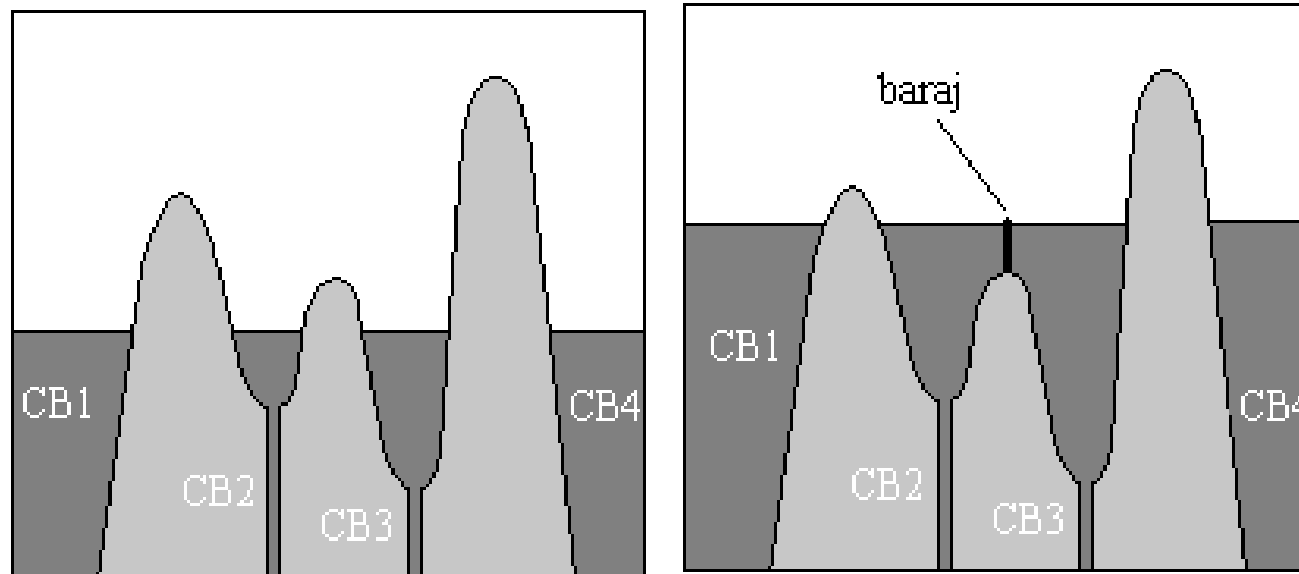
Segmentarea Watershed

Forming a Terrain...



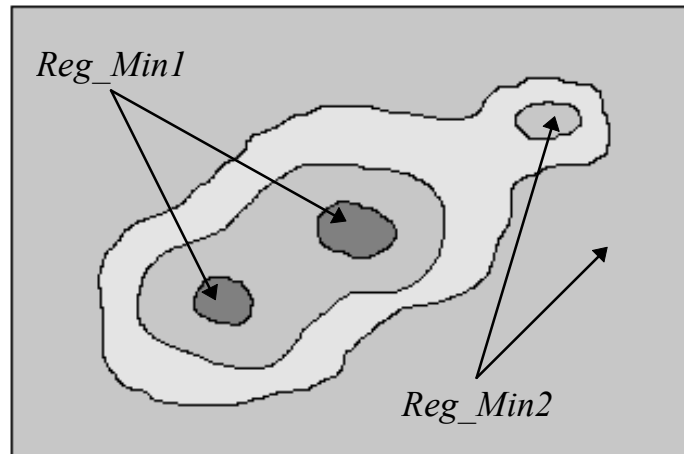
Segmentarea Watershed

A fost introdus de Lantuejoul și Beucher și consideră că o imagine este o suprafață topologică, prin utilizarea mărimii nivelului de gri ca o altitudine. Locurile în care au loc schimbări bruște ale nivelului de gri constituie în acest peisaj, locurile în care este prezent un contur. În scopul segmentării, vor fi în primul rând descoperite liniile de creastă ale imaginii gradient.

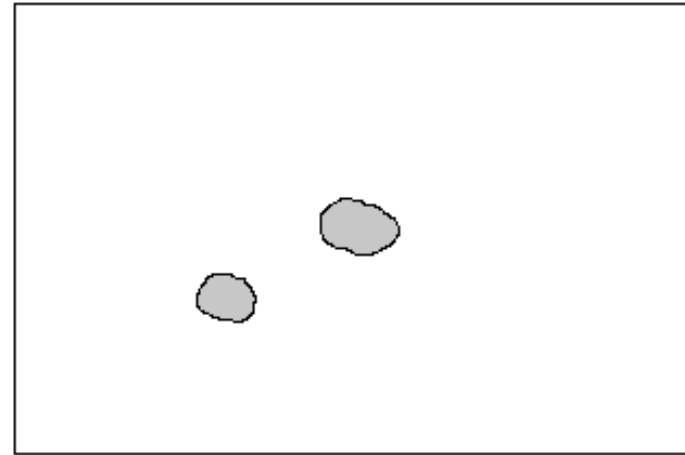


Obținerea watershed-ului. a) La momentul t , cele două bazine nu s-au întâlnit, b) la momentul $t+h$, se construiește un baraj pentru a separa bazinul CB2 de CB3.

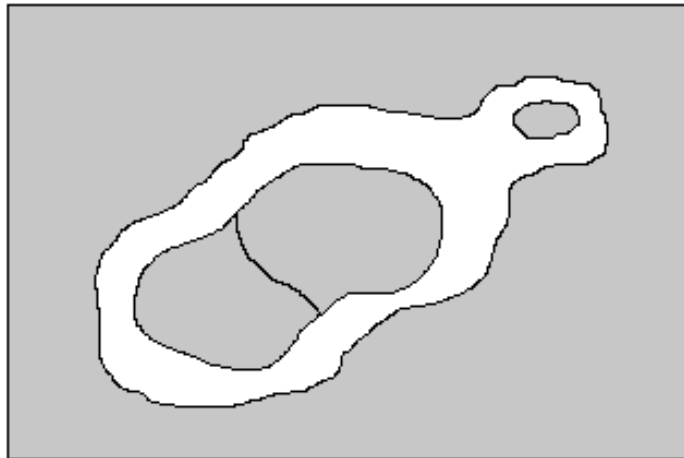
Watershed



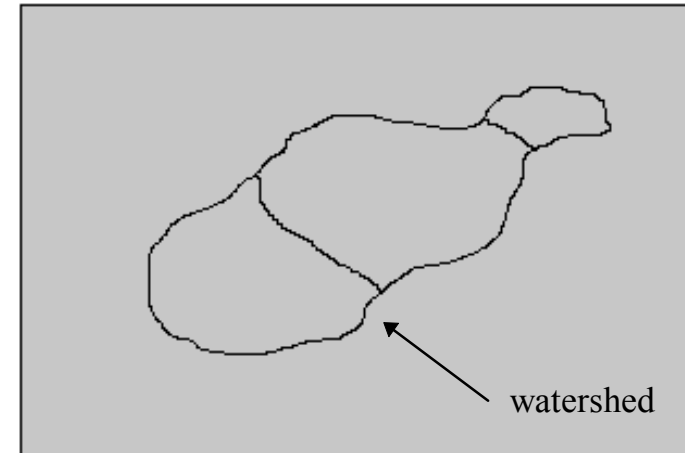
a)



b)



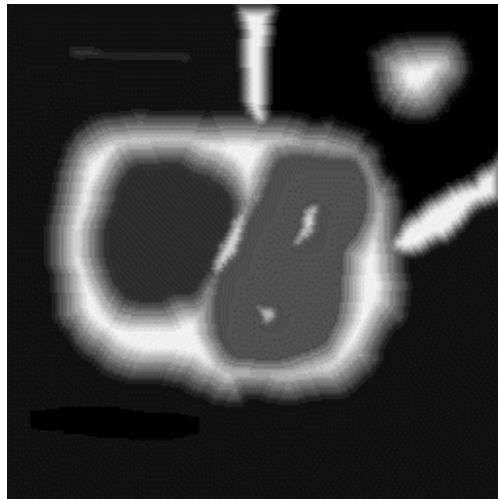
c)



d)

Ilustrarea procesului recursiv de imersie. a) Imaginea cu trei nivele de gri, b) bazinele de acumulare *X1*, c) bazinele *X2*, d) bazinele *X3*.

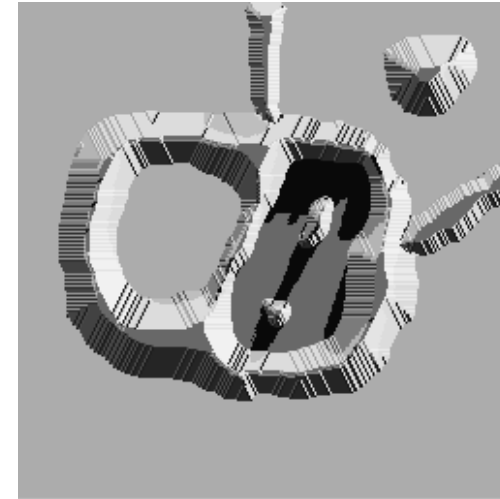
Watershed



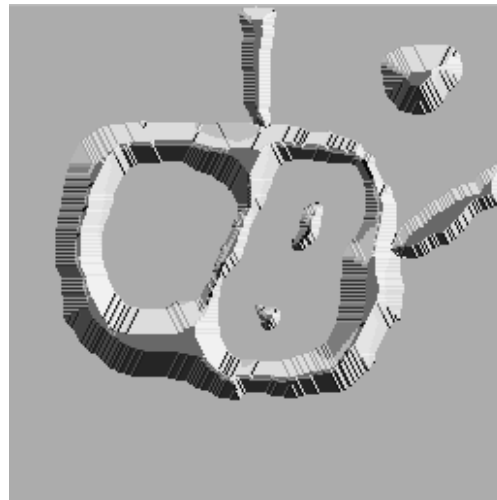
a)



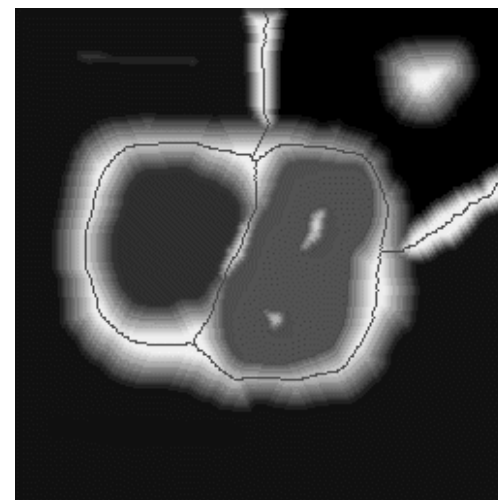
b)



c)



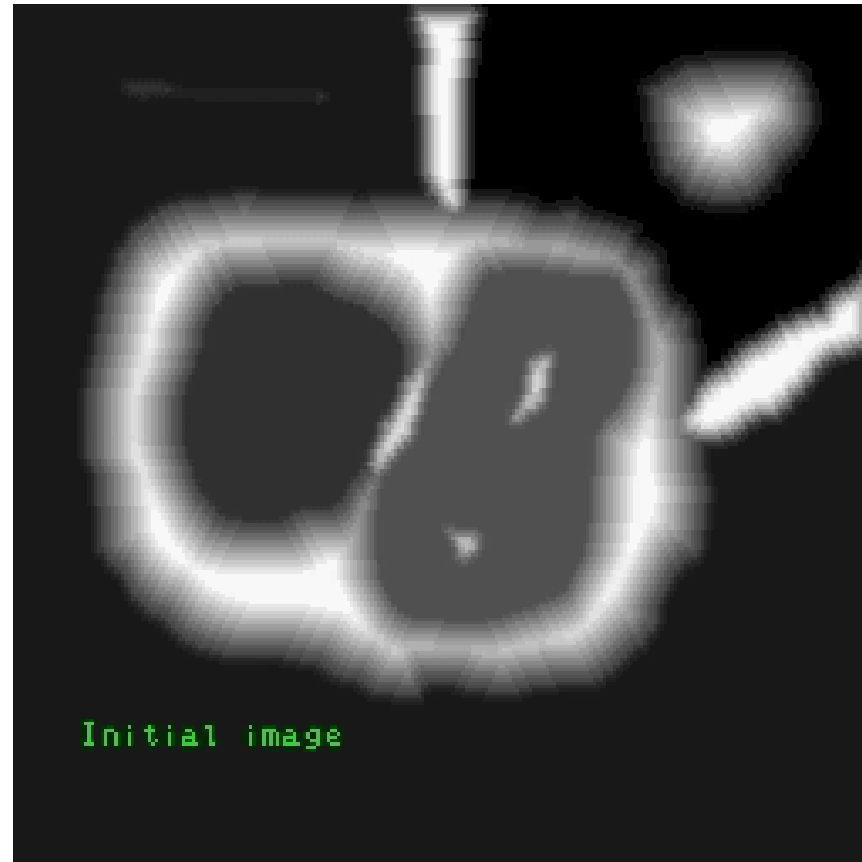
d)



e)

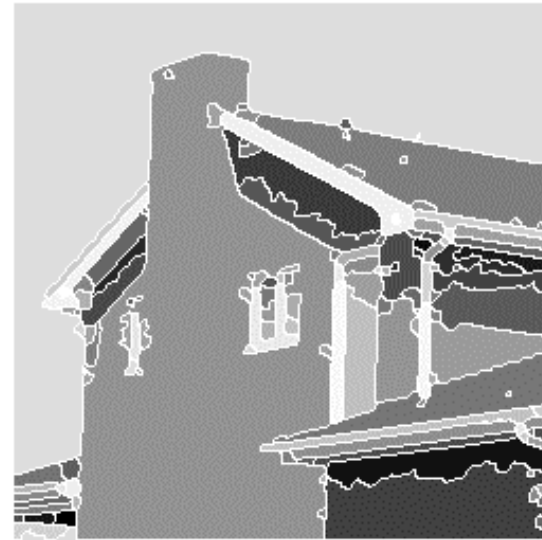
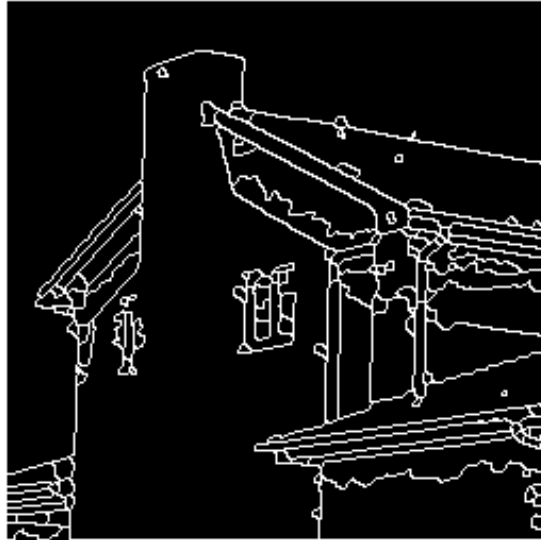
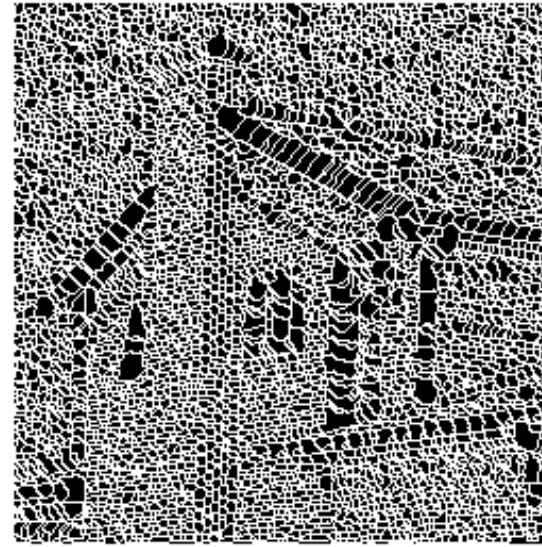
Segmentare prin metoda watershed (după [Beucher 94]); a) imaginea originală, b) suprafața topografică, c) d) imagini intermediare în procesul de inmersie, e) watershed-ul

Watershed



<http://bigwww.epfl.ch/demo/jwatershed/start.php>

Watershed



Contururi active - “snakes”

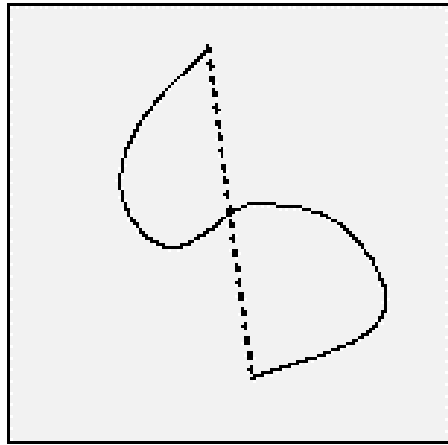
Dezvoltarea modelelor de contururi active a început cu cercetările lui Kass, Witkin și Terzopoulos.

Acestea se bazează pe paradigma potrivit căreia existența unui contur nu depinde numai de gradientul calculat într-un anumit punct, ci și de o distribuție spațială.

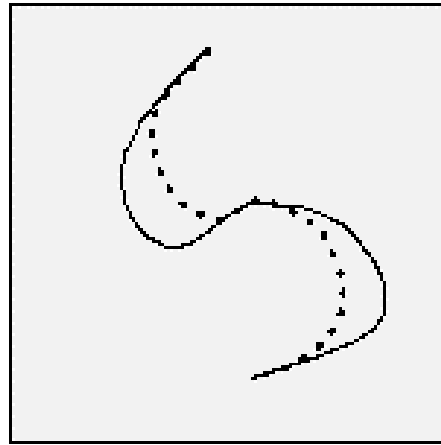
Principalul avantaj asupra altor tehnici de segmentare îl constituie faptul că această metodă integrează datele provenite din imagine, estimările inițiale și proprietățile dorite ale conturilor într-un singur proces de segmentare.

Contururile active sau “snakes” au fost definite ca fiind funcții *spline* ce minimizează o energie. Energia snake-ului depinde de forma sa și de poziția sa în imagine.

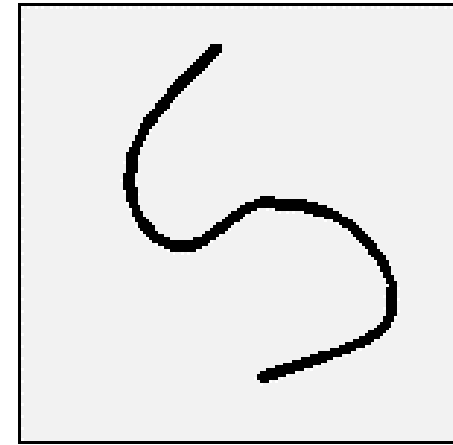
Contururi active - “snakes”



a)



b)



c)

Conturul activ - snake. a) Poziția inițială a snake-ului a fost aleasă interactiv în apropierea conturului; b), c) două momente din procesul de minimizare a energiei.

Contururi active - “snakes”

Funcția energie ce se urmărește a fi minimizată este o combinație ponderată între forțele interne și externe ce acționează asupra conturului activ.

Forțele interne provin din forma snake-ului iar forțele externe provin din imagine sau de la mecanisme de recunoaștere de nivel înalt.

Snake-ul este definit sub forma parametrică ca fiind $v(s) = (x(s), y(s))$ unde $x(s), y(s)$ sunt coordonatele de-alungul conturului și $s \in [0,1]$.

Funcția energie ce trebuie minimizată este:

$$E_{snake}^* = \int_0^1 E_{snake}(v(s))ds = \int_0^1 E_{int}(v(s)) + E_{image}(v(s)) + E_{con}(v(s))ds \quad (4.1)$$

unde E_{int} reprezintă energia internă a curbei spline datorată încovoierii, E_{image} cuantifică forțele datorate elementelor prezente în imagine și E_{con} exprimă constângerii de natură externă aplicate conturului activ. În mod obișnuit, $v(s)$ este aproximată cu o funcție spline în scopul asigurării proprietății de continuitate.

Contururi active - “snakes”

Cel de al doilea termen din integrala 4.1 provine din informația prezentă în punctele imaginii peste care snake-ul se găsește. Spre exemplu, o sumă ponderată a trei funcții diferite poate determina conturul activ să fie atras de linii, contururi și punctele terminale:

$$E_{\text{image}} = w_{\text{linie}} E_{\text{linie}} + w_{\text{contur}} E_{\text{contur}} + w_{\text{term}} E_{\text{term}} \quad (4.3)$$

Funcția energie ce are la bază liniile poate avea forma simplă:

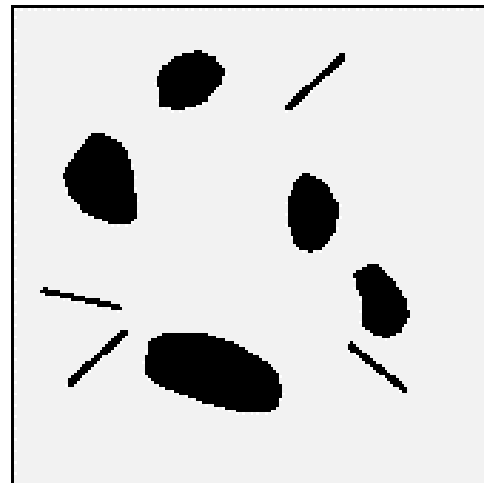
$$E_{\text{linie}} = f(x, y) \quad (4.4)$$

unde $f(x, y)$ este intensitatea nivelului de gri a pixelului de coordonate (x, y) . Semnul ponderii w_{linie} specifică dacă snake-ul este atras de linii luminoase sau mai întunecate. Funcția energie ce are la bază contururile poate avea următoarea formă:

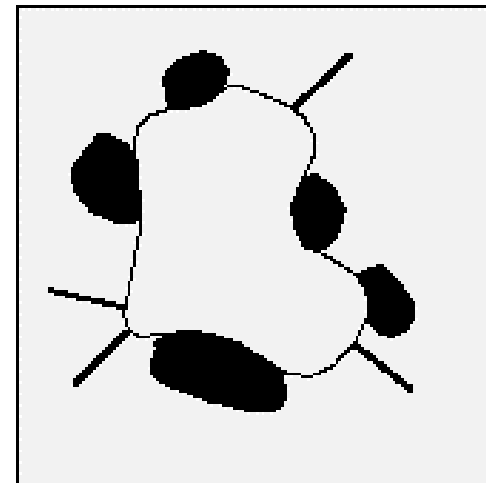
$$E_{\text{contur}} = -|\nabla f(x, y)|^2 \quad (4.5)$$

și ea va atrage snake-ul de contururile cu gradient de valoare mare și deci de punctele în care există un contur foarte pronunțat.

Contururi active - “snakes”



a)



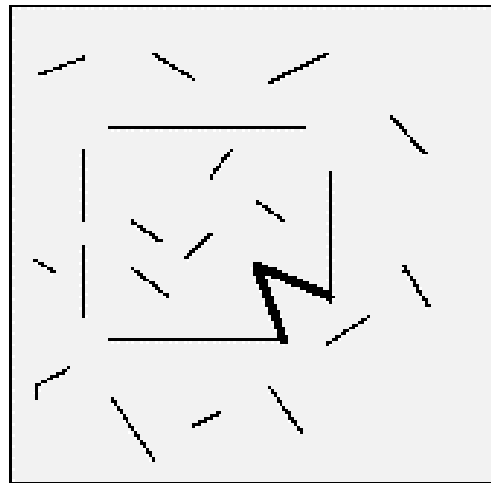
b)

Un snake atras de contururi și terminații de linii. a) Formele prezente în imagine, b) snake-ul este atras de elementele ce definesc conturul.

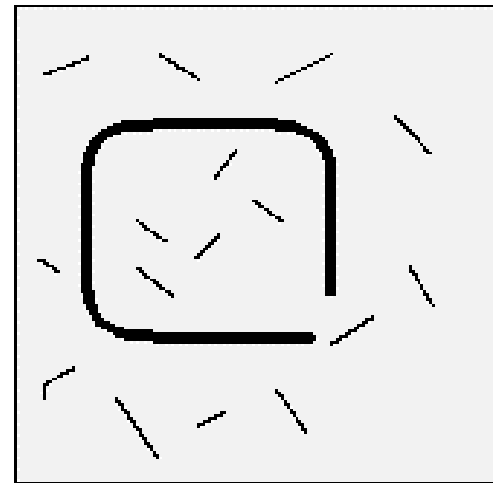
Cel de al treilea termen al integralei 4.1 provine din constrângeri externe impuse fie de utilizator fie de anumiți algoritmi de nivel mai înalt, ce pot determina snake-ul sa se apropie sau să se depărteze de anumite forme.

Dezvoltări ale metodei contururilor active

Tehnici de tip balon



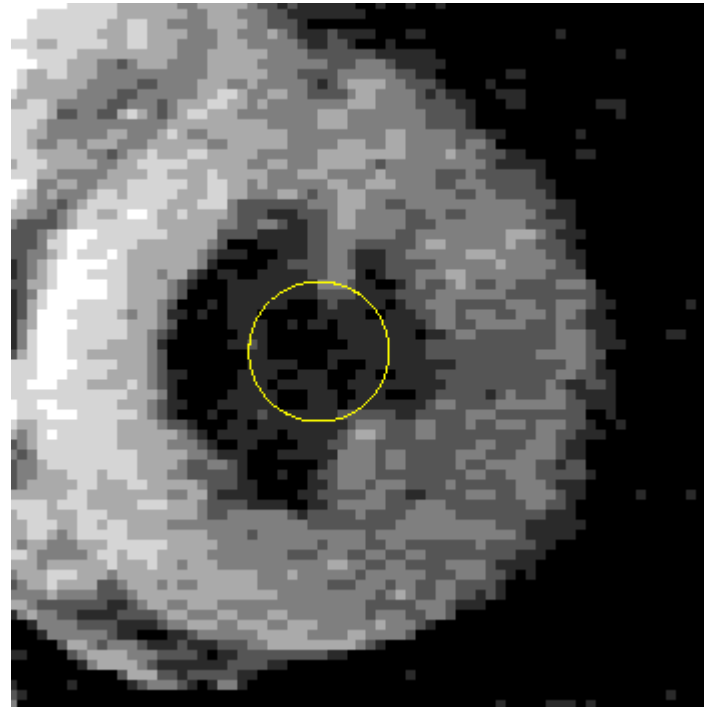
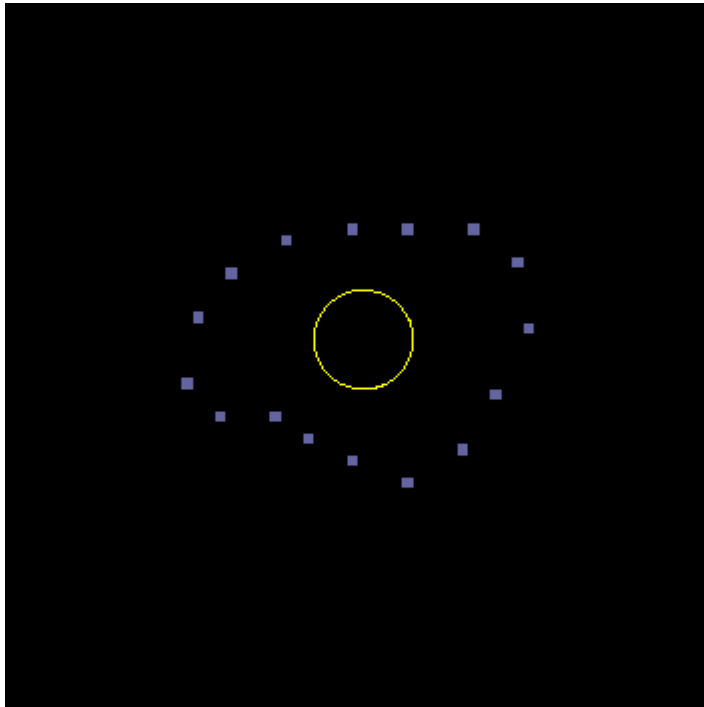
a)



b)

Model de contur activ balon. a) Conturul inițial; b) conturul final după umflare și minimizarea energiei.

Dezvoltări ale metodei contururilor active

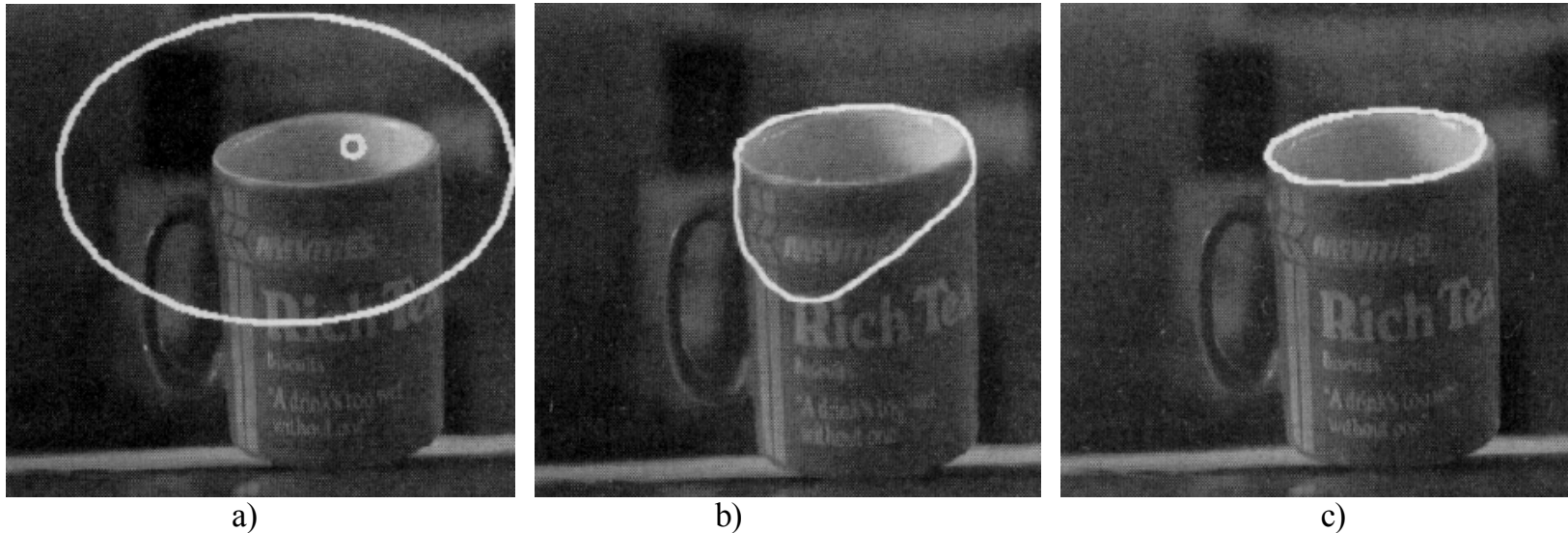


http://users.ecs.soton.ac.uk/msn/book/new_demo/Snakes/

<http://www.markschulze.net/snakes/>

Dezvoltări ale metodei contururilor active

Se propune utilizarea a două contururi, unul situat în interiorul și altul în exteriorul zonei de interes. Cele două contururi încearcă să găsească un minim local al energiei în funcție de conținutul imaginii și de poziția lor inițială.



Comparația detecție unei forme folosind doi algoritmi (după [Gunn 97]);

a) inițializarea snake-ului; b) algoritmul clasic al snake-ului; c) algoritmul cu două contururi

Segmentarea bazată pe region growing

Scopul metodelor de segmentare bazate pe contururi este găsirea unor granițe între diversele regiuni.

Omogenitatea este o proprietate importantă a regiunilor și este criteriul principal de segmentare în tehnică region growing. Acesta realizează împărțirea imaginii în zone de omogenitate.

Pentru a putea segmenta prin region growing, regiunile trebuie să îndeplinească următoarele criterii:

$$H(R_i) = TRUE \quad i = 1, 2, \dots, S \quad (5.1)$$

$$H(R_i \cup R_j) = FALSE \quad i \neq j, \quad R_i \text{ este adiacent cu } R_j \quad (5.2)$$

unde S este numărul total de regiuni din imagine și $H(R_i)$ este funcția binară de evaluare a omogenității regiunii R_i .

Criteriul de omogenitate poate fi bazat pe nivelul de gri, textura sau model (dacă se folosește informația semantică).

Segmentarea bazată pe region growing (cont.)

Cea mai naturală metodă de region growing este de a porni creșterea considerând fiecare pixel ca reprezentând o singură regiune. Deoarece astfel de regiuni nu satisfac ecuația 5.2, acestea vor fi unite în noi regiuni, atât timp cât ecuația 5.1 este satisfăcută.

Algoritmul de uniune a regiunilor

1. Se definește o metodă de segmentare pentru început a imaginii în regiuni foarte mici și care satisfac ecuația 5.1.
2. Se definește un criteriu de unire a două regiuni adiacente.
3. Se unesc toate regiunile ce satisfac acest criteriu. Dacă nici o pereche de regiuni nu poate fi unită conform ecuației 5.1, se oprește.

Criteriul de unire poate fi unul statistic și anume cel al lui Fisher. Să presupunem că există două regiuni adiacente R_1 și R_2 , ce sunt caracterizate de următoarele mărimi: valoarea medie a nivelului de gri μ_1 și μ_2 ; varianță σ_1^2 și σ_2^2 și mărimea regiunilor n_1 și n_2 . Pentru a lua decizia de unire a acestora, se poate folosi distanța Fisher:

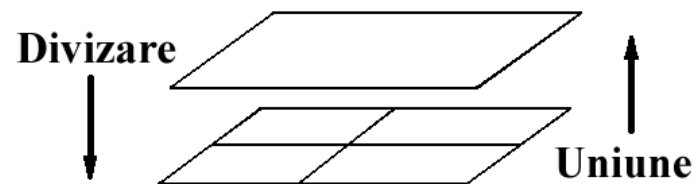
$$\frac{(n_1 + n_2)(\mu_1 + \mu_2)^2}{n_1\sigma_1^2 + n_2\sigma_2^2} = \frac{n\sigma^2}{n_1\sigma_1^2 + n_2\sigma_2^2} - 1 \quad (5.3)$$

unde $n = n_1 + n_2$ și σ^2 este varianța regiunii rezultată în urma unirii celor două. Dacă valorile statistice obținute se află sub un anumit prag, cele două regiuni sunt reunite.

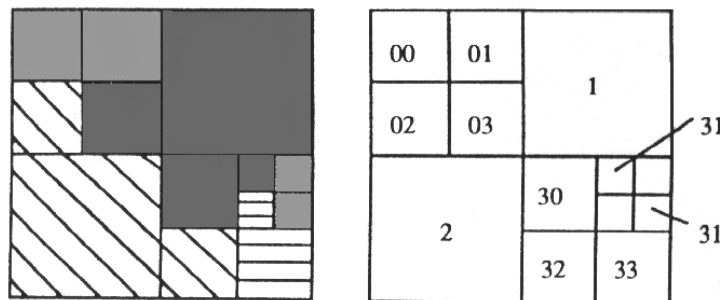
Segmentarea bazată pe region growing (cont.)

Divizarea și unirea regiunilor

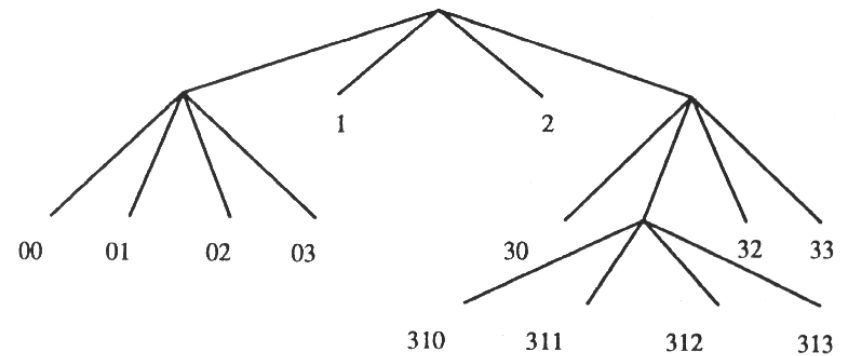
O combinație dintre divizarea și uniunea regiunilor (*split and merge*) poate oferi avantajele evidente ale celor două metode. Aceste tehnici folosesc reprezentări piramidale ale imaginii, în care regiunile au forme pătrate și corespund elementelor din structura piramidală.



a)



b)



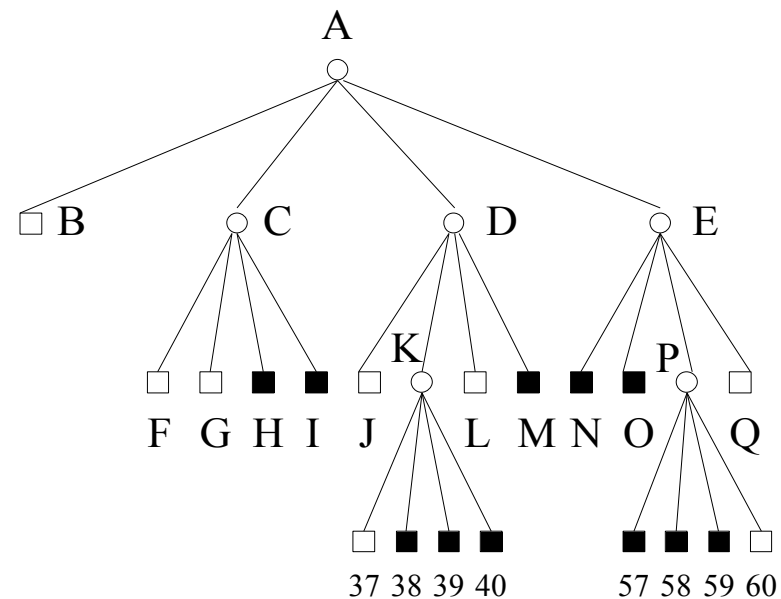
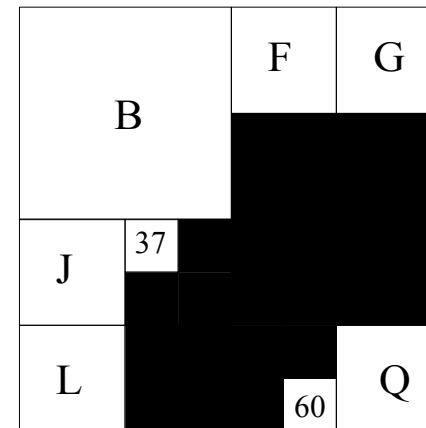
c)

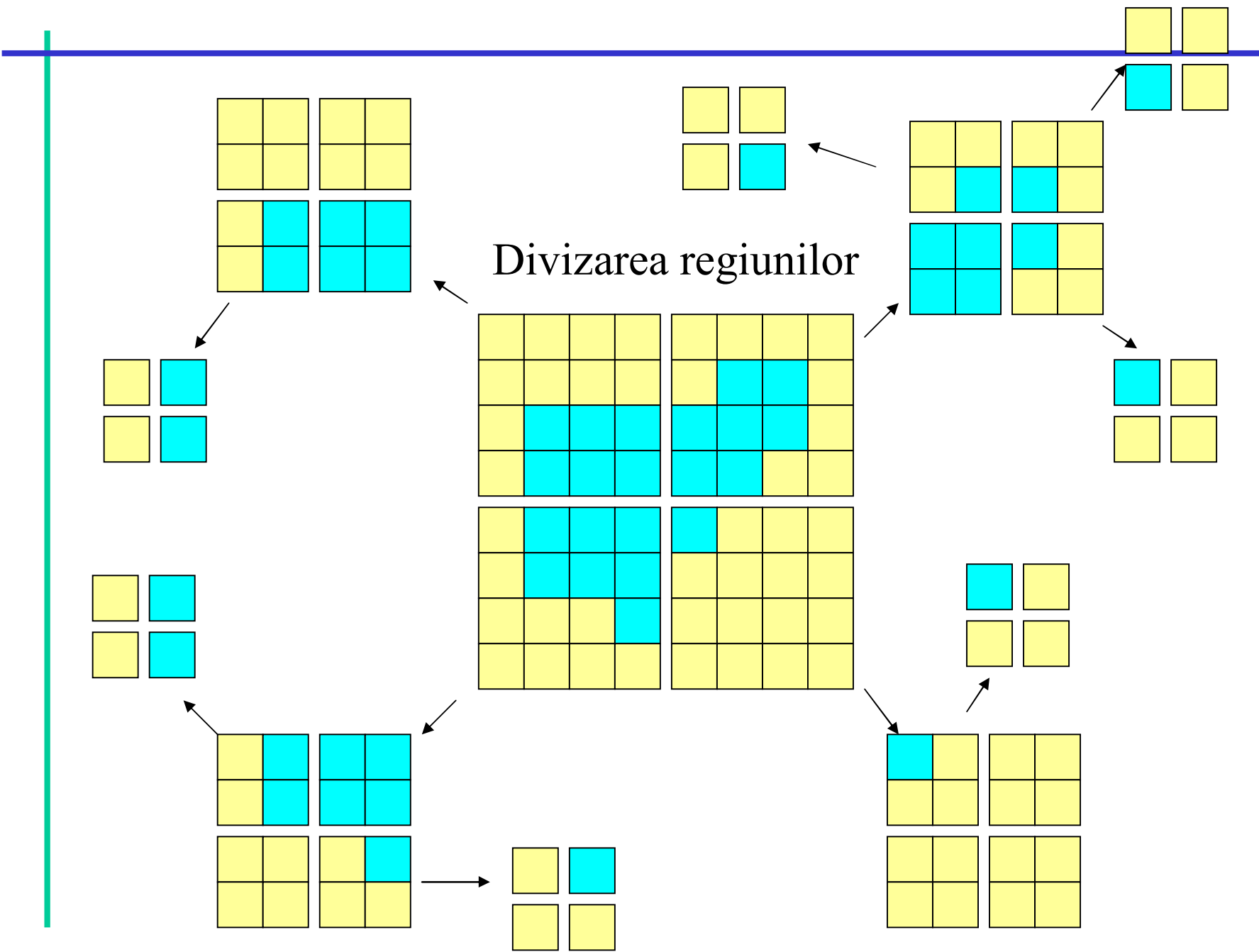
Metoda de segmentare split and merge. a) Direcțiile de parcurgere ale structurii, b) rezultatul segmentării și marcajul regiunilor, c) arborele pătratic de segmentare.

Segmentarea bazată pe region growing (cont.)

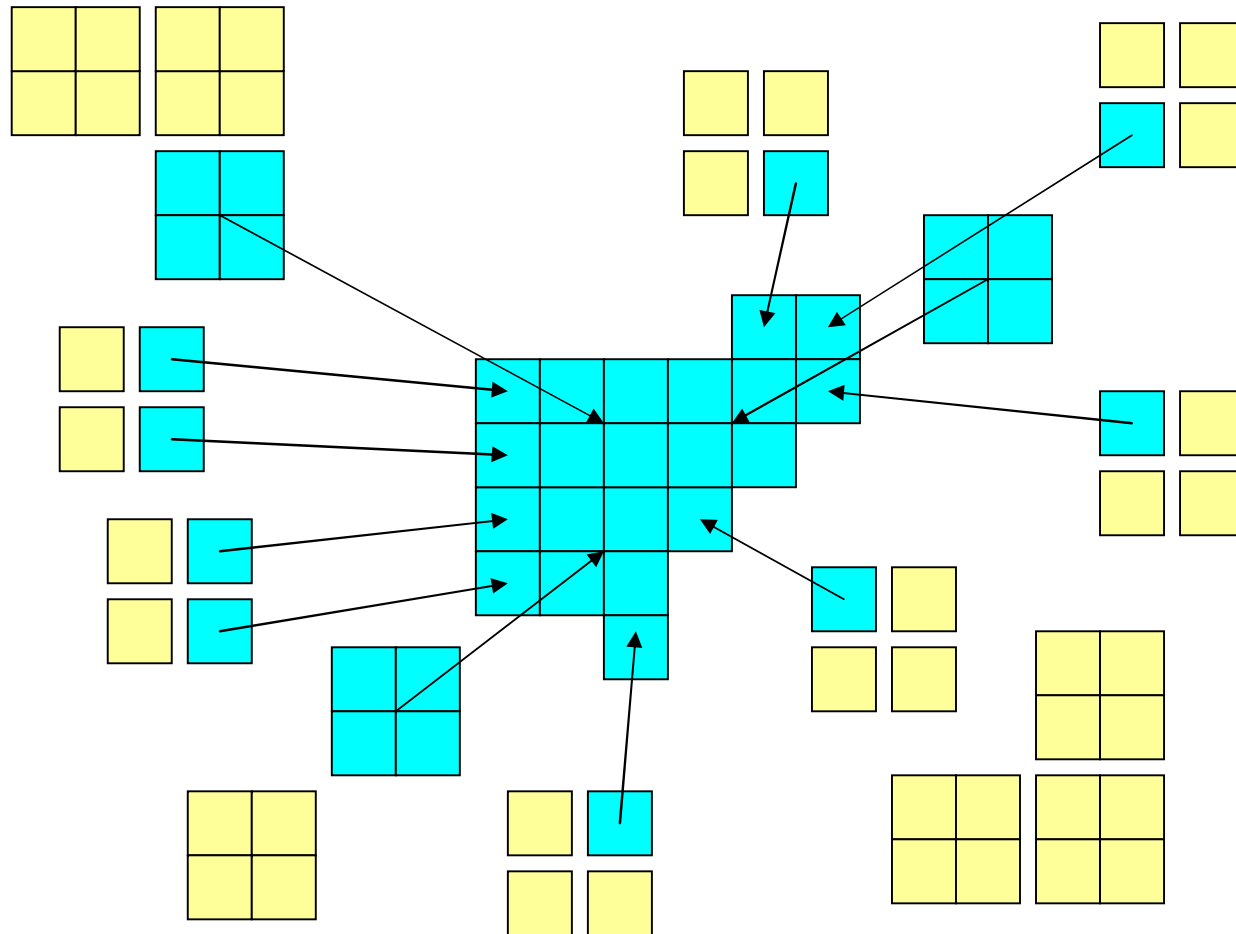
Divizarea și unirea regiunilor

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





Unirea regiunilor



Segmentarea bazată pe region growing (cont.)

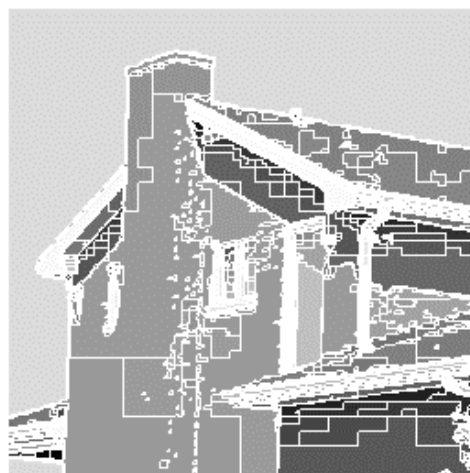
Divizarea și uniunea regiunilor



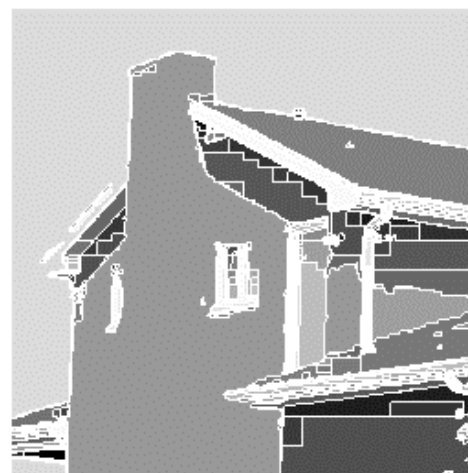
a)



b)



c)



d)

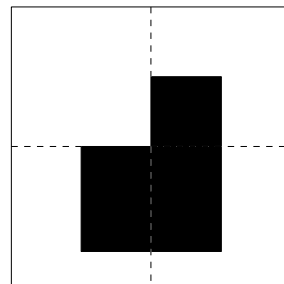
Segmentare prin split and merge; a) rezultatul segmentării pentru varianta 30 și b) 60;
c) și d) contururile regiunilor obținute suprapuse peste imaginea segmentată în cele două cazuri

Segmentarea bazată pe region growing (cont.)

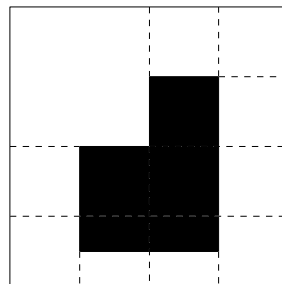
Divizarea și unirea regiunilor

Algoritmul *split and merge*:

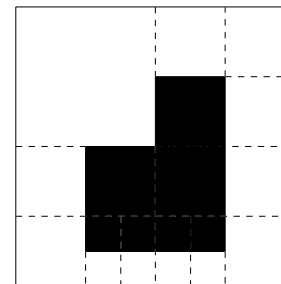
1. Se definește o segmentare inițială în regiuni, un criteriu de omogenitate și o structură de date piramidală.
2. Dacă o regiune R din structura piramidală nu este omogenă, se divide în 4 regiuni copil; dacă există 4 regiuni având același părinte, ele pot fi unite într-o singură regiune omogenă. Dacă nici o regiune nu poate fi divizată sau unită, se trece la pasul 3.
3. Dacă există două regiuni adiacente R_i și R_j (chiar dacă sunt pe nivele diferite sau nu au același părinte), ele se unesc, dacă îndeplinesc condițiile de omogenitate.
4. Se unesc regiunile de dimensiuni mici cu cele mai asemănătoare regiuni adiacente, dacă este necesar.



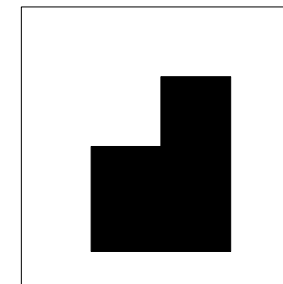
(a)



(b)



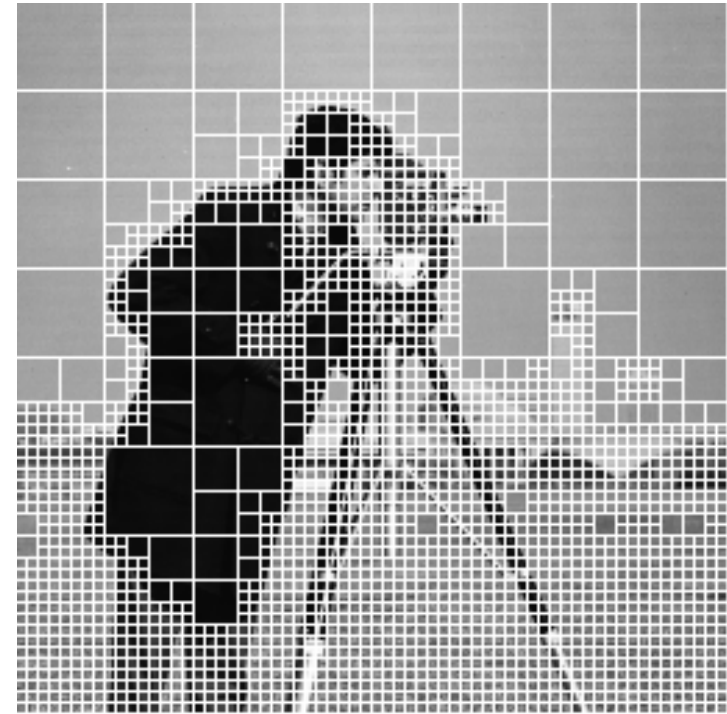
(c)



(d)

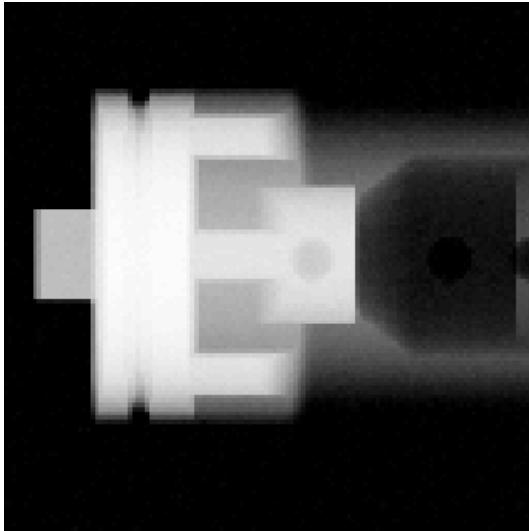
Segmentarea bazată pe region growing (cont.)

Divizarea și uniunea regiunilor

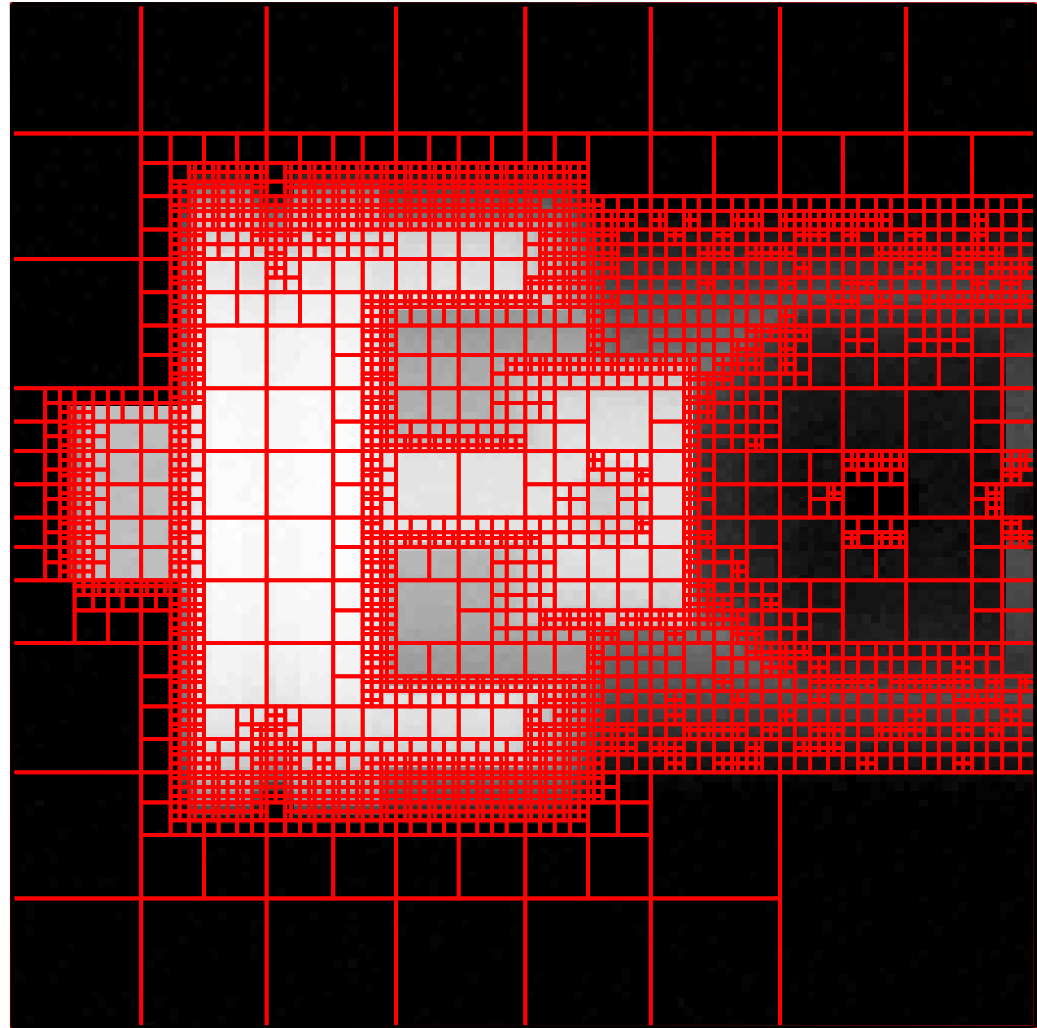


Segmentarea bazată pe region growing (cont.)

Divizarea și uniunea regiunilor



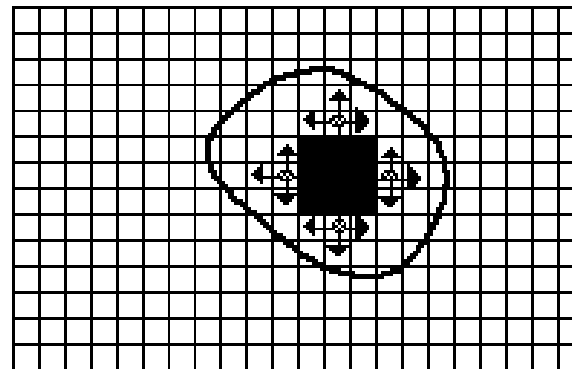
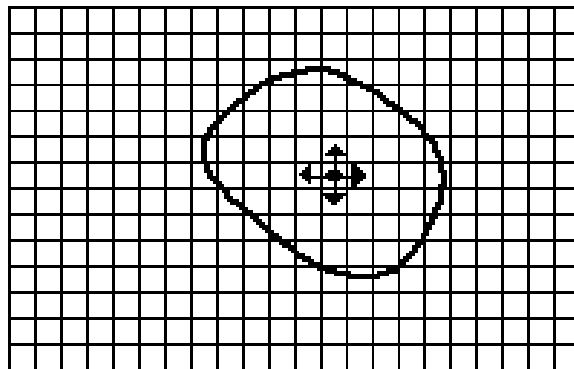
Criteriul de omogenitate
deviatia standard = 5



Segmentarea bazată pe region growing (cont.)

Region Growing

Se pornește de la un pixel sau un grup de pixeli și se examinează toți pixelii vecini ai acestora. Dacă unul dintre aceștia îndeplinește un anumit criteriu, este adăugat grupului de pixeli deja format. Acest proces este continuat până când nu se mai adaugă pixeli în grupul respectiv, deci până când a fost identificată o regiune.

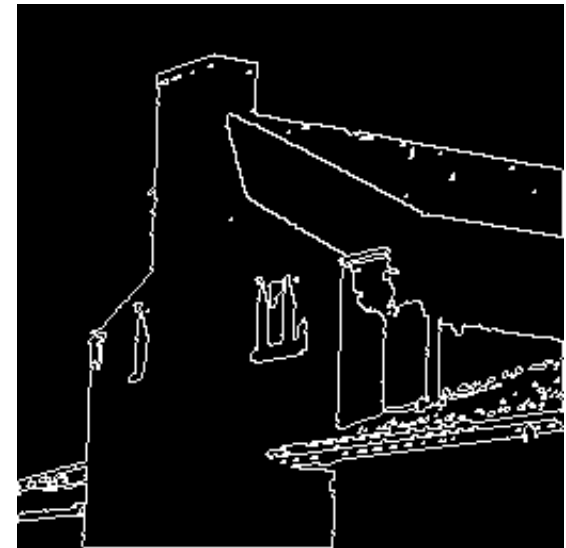


- pixel de start
- ↑ direcția de creștere
- pixeli grupați
- pixeli ce urmează a fi procesați

Algoritmul region growing

Segmentarea bazată pe region growing (cont.)

Region Growing



Segmentare prin region grow

Segmentarea bazată pe region growing (cont.)

Region Growing



pixel initial



dupa 25 pixeli



dupa 100 pixeli



dupa 500 pixeli



dupa 1000 pixeli



regiunea finala

Segmentarea bazată pe region growing (cont.)

Region Growing

