Concepte și aplicații în Vederea Artificială

Bogdan Alexe
bogdan.alexe@fmi.unibuc.ro

Radu Ionescu
radu.ionescu@fmi.unibuc.ro

Curs opțional

anii III/IV, semestrul I, 2024-2025

CONSILIUL FMI 17.19.2024

Ordinea de zi

1. Aprobarea minutelor ședințelor CF (cu vot electronic) din Minuta CF din 12.07.2024, CF din 16.07.2024, Minuta CF din 22.07.2024, Minuta CF din 01.08.2024, Minuta CF din 12.09.2024, Minuta CF din 14-15.09.2024, Minuta CF din 18.09.2024, Minuta CF din 24.09.2024, Minuta CF din 26.09.2024
2. Aprobarea standardelor științifice minimale pentru cadrele didactice din FMI.
3. Aprobarea Metodologiei pentru acordarea burselor studenților FMI (licență si master).
4. Prezentarea raportului CDOS 2024.
5. Aprobarea scoaterii la concurs a unor posturi didactice pe semestrul I.
6. Aprobarea comisiei de atribuire a burselor la nivel de facultate
7. Avizarea rezultatelor alegerilor reprezentaților studenților FMI in CF si in Senat.
8. Avizarea propunerii Departamentului de Informatica pentru candidatura domnului prof. univ. dr. Victor Mitrana pentru Premiile Senatului la categoria profesorul emerit al anului.
9. Aprobarea listelor de experti ce vor tine ore in regim de plata cu ora.
10. Diverse

- Olimpiada de Inteligenta Artificiala
- Standarde pentru redactarea lucrarii de licenta sau disertație la informatica
- Modificări ale criteriilor de acordare a gradațiilor de merit la departamentul de informatica.
- Discuții lista concursurilor acceptate de FMI pentru admitere
- Modificarea tipului de subiect de matematica la admitere

CONSILIUL FMI 17.19.2024

Ordinea de zi

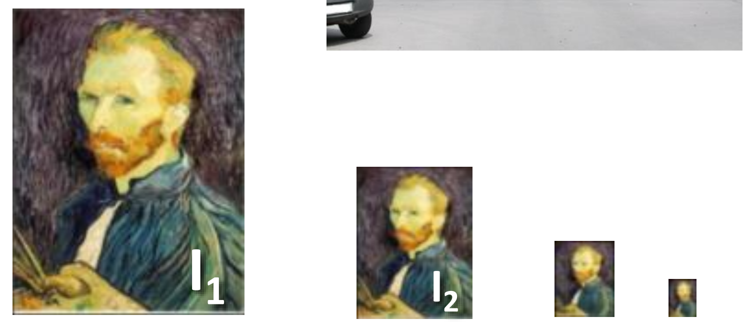
1. Aprobarea minutelor ședințelor CF (cu vot electronic) din Minuta CF din 12.07.2024, CF din 16.07.2024, Minuta CF din 22.07.2024, Minuta CF din 01.08.2024, Minuta CF din 12.09.2024, Minuta CF din 14-15.09.2024, Minuta CF din 18.09.2024, Minuta CF din 24.09.2024, Minuta CF din 26.09.2024
2. Aprobarea standardelor științifice minimale pentru cadrele didactice din FMI.
3. Aprobarea Metodologiei pentru acordarea burselor studenților FMI (licență si master).
4. Prezentarea raportului CDOS 2024.
5. Aprobarea scoaterii la concurs a unor posturi didactice pe semestrul I.
6. Aprobarea comisiei de atribuire a burselor la nivel de facultate
7. Avizarea rezultatelor alegerilor reprezentaților studenților FMI in CF si in Senat.
8. Avizarea propunerii Departamentului de Informatica pentru candidatura domnului prof. univ. dr. Victor Mitrana pentru Premiile Senatului la categoria profesorul emerit al anului.
9. Aprobarea listelor de experti ce vor tine ore in regim de plata cu ora.
10. Diverse

- Olimpiada de Inteligenta Artificiala
- Standarde pentru redactarea lucrarii de licenta sau disertație la informatica
- Modificări ale criteriilor de acordare a gradațiilor de merit la departamentul de informatica.
- Discuții lista concursurilor acceptate de FMI pentru admitere
- Modificarea tipului de subiect de matematica la admitere

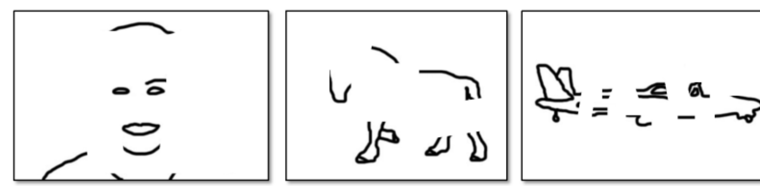
Cursul trecut

- Aplicații ale filtrelor:
  - găsirea şabloanelor

  - redimensionarea imaginilor

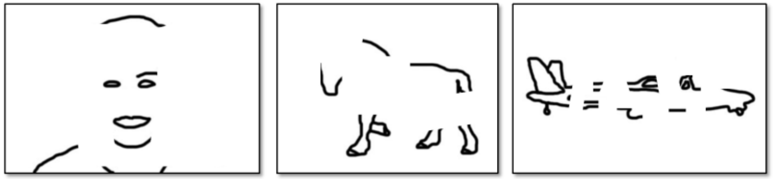


  - extragerea informației (gradienți, muchii, textură)

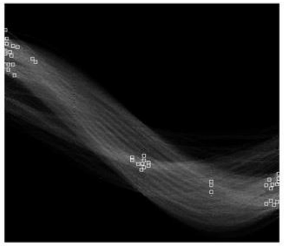


Cursul de azi

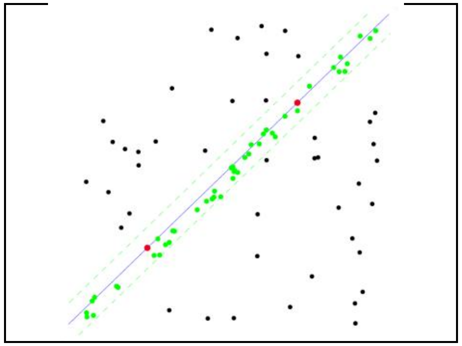
- Aplicații ale filtrelor: extragerea informației (muchii)



- Aplicație: detectarea liniilor cu
  - transformata Hough



  - RANSAC (cursul viitor)



Aplicatie laborator: Redimensionarea imaginilor cu păstrarea conținutului

Redimensionarea imaginilor cu păstrarea conținutului

Imagine inițiala

Redimensionare imagine: vreau să măresc/micșorez lățimea/înălțimea imaginii

Idee: adaug/elimin înșiruiri de pixeli ce conectează extremitățile imaginii ← Cum le aleg?

[Shai & Avidan, SIGGRAPH 2007]

Slide adaptat după K. Grauman

Redimensionarea imaginilor cu păstrarea conținutului

Redimensionare cu păstrarea conținutului

Redimensionare uzuală
(funcția cv.resize în OpenCV)

[Shai & Avidan, SIGGRAPH 2007]

Slide adaptat după K. Grauman

Ideea de bază

https://www.youtube.com/watch?v=6NcIJXTlugc

Slide adaptat după K. Grauman

Ideea de bază

Redimensionare cu păstrarea conținutului

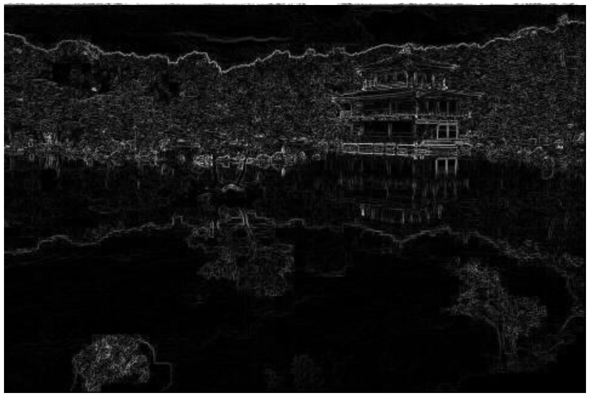
Intuiție:

- Păstrăm conținutul cel mai "interesant"
  → Preferăm să eliminăm/adăugăm pixeli cu gradient mic
- Pentru a reduce/mări o dimensiune , eliminăm/adăugăm înşiruiri de pixeli (drumuri neregulate) ce conectează extremitățile
  → Soluție optimă folosind programarea dinamică

Slide adaptat după K. Grauman

Ideea de bază – eliminare de drumuri

imaginea I



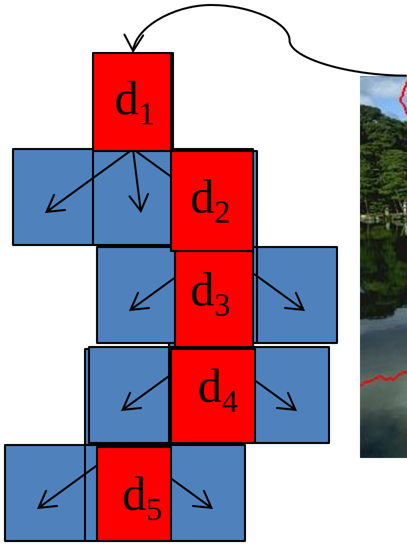
gradientul imaginii I (magnitudine)

$$\nabla I = \sqrt{(\frac{\partial I}{\partial x})^2 + (\frac{\partial I}{\partial y})^2}$$

- Eliminăm ‘drumuri’ (înşiruiri de pixeli) fără a afecta vizibil imaginea:
  - măsurăm ‘costul unui drum’ ca suma magnitudinilor gradienților pixelilor ce alcătuiesc drumul
- La fiecare iterație, alegem drumul cu **costul cel mai mic** din imagine

Slide adaptat după K. Grauman

Algoritmul



imaginea I

gradientul imaginii I

$$\nabla I = \sqrt{\left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2}$$

Fie d un drum vertical format din N pixeli: **d** = (d_1, d_2, … ,d_N)

Definim costul unui drum : $Cost(\mathbf{d}) = \sum_{i=1}^{N} \nabla I(d_i)$

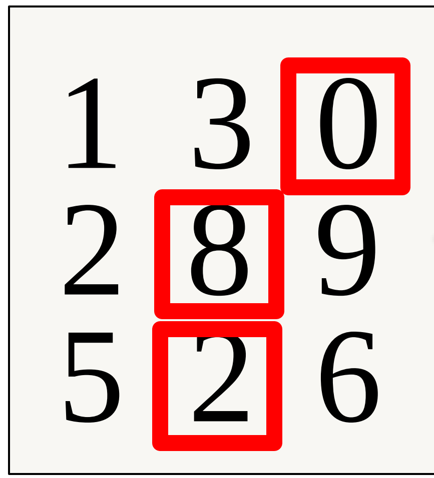
Drumul optim minimizează acest cost: $\mathbf{d}^* = \min_{\mathbf{d}} Cost(\mathbf{d})$

Calculăm drumul optim în mod eficient folosind programarea dinamică

Slide adaptat după K. Grauman

Cum identificăm drumul de cost minim?

- Mai întâi, considerăm o strategie **greedy**:



La fiecare pas aleg cea mai bună soluție locală
(aleg pixelii cu gradientul cel mai mic)

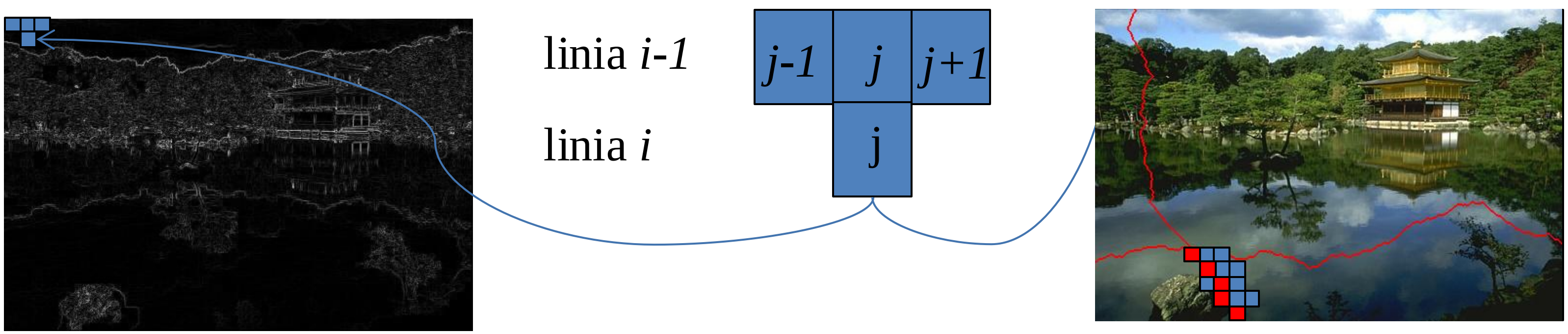
Combinarea optimelelor locale NU conduce la optim global

gradientul imaginii

Algoritmul

- Calculăm pentru fiecare pixel (i,j) drumul de cost minim până la (i,j) cu ajutorul celor trei vecini:

$$\mathbf{M}(i,j) = MagGradient(i,j) + \min(\mathbf{M}(i-1,j-1), \mathbf{M}(i-1,j), \mathbf{M}(i-1,j+1))$$



gradientul imaginii

Matricea M de costuri ale drumurilor (aici pentru drumuri verticale)

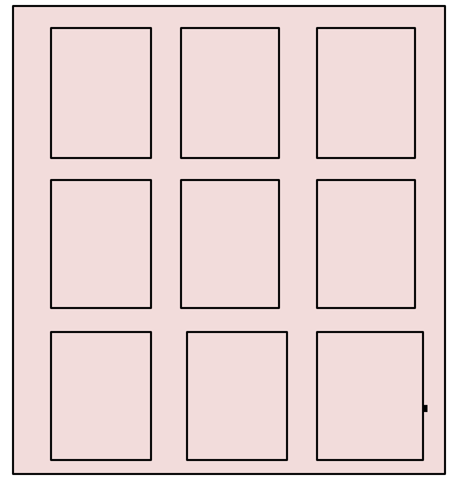
- Valoarea minimă din ultima linie din **M** reprezintă valoarea drumului vertical de cost minim. Poziția costului minim pe ultima linie localizează ultimul pixel din drum.
- Găsim drumul de cost minim mergând înapoi si găsind drumul minim cu ajutrul celor cei 3 vecini de sus din **M**

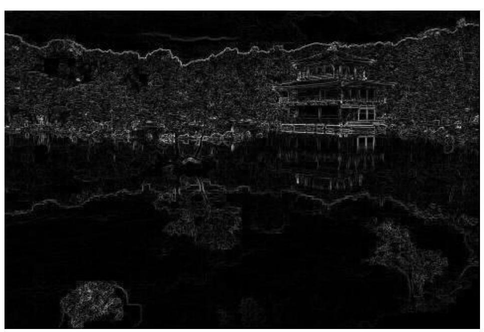
Slide adaptat după K. Grauman

Exemplu – programare dinamică

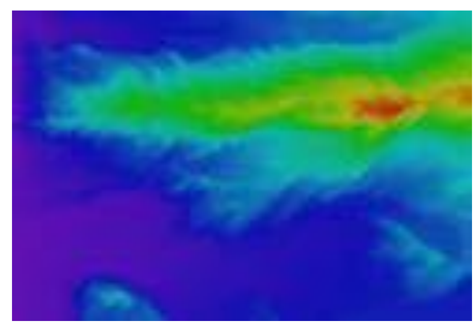
$$\mathbf{M}(i,j) = MagGradient(i,j) + \min(\mathbf{M}(i-1,j-1), \mathbf{M}(i-1,j), \mathbf{M}(i-1,j+1))$$

1	3	0
2	8	9
5	2	6





gradientul imaginii

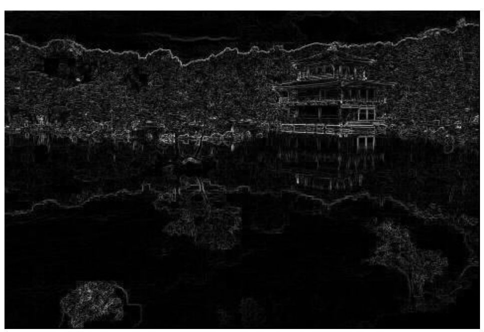


Matricea M de costuri ale drumurilor (aici pentru drumuri verticale)

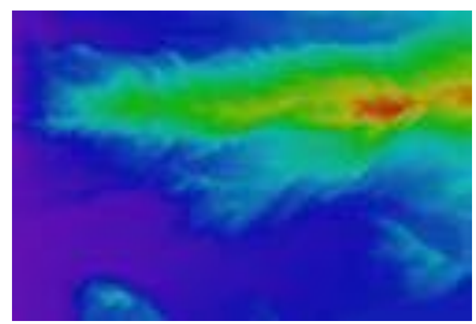
Exemplu – programare dinamică

$$\mathbf{M}(i,j) = MagGradient(i,j) + \min(\mathbf{M}(i-1,j-1), \mathbf{M}(i-1,j), \mathbf{M}(i-1,j+1))$$

$$\begin{matrix} \boxed{1} & 3 & 0 \\ \boxed{2} & 8 & 9 \\ 5 & \boxed{2} & 6 \end{matrix} \qquad \begin{matrix} \boxed{1} & 3 & 0 \\ \boxed{3} & 8 & 9 \\ 8 & \boxed{5} & 14 \end{matrix}$$



gradientul imaginii

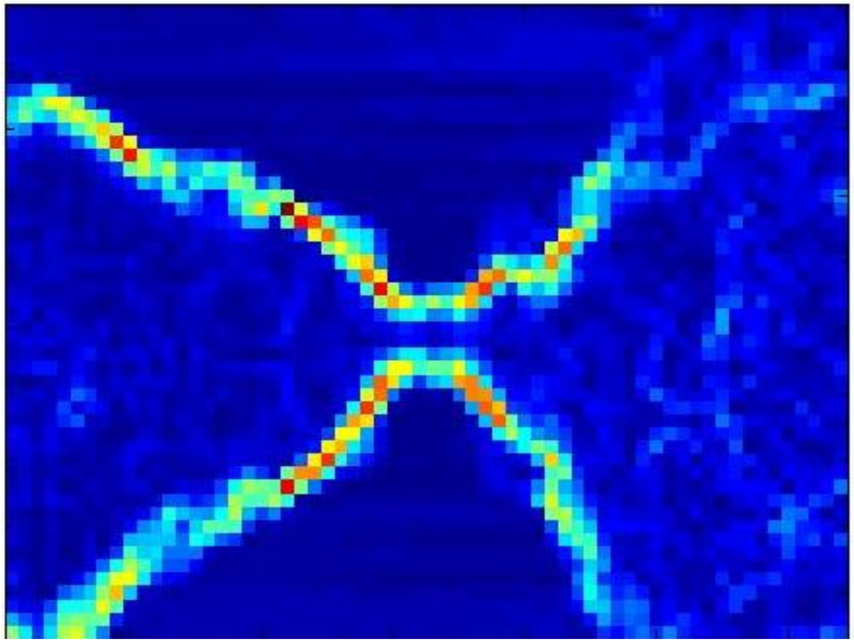


Matricea M de costuri ale drumurilor (aici pentru drumuri verticale)

Exemplu

Imagine inițială

Matricea **M** de costuri ale drumurilor



Albastru = cost mic

Roşu = cost mare

Slide adaptat după K. Grauman

Exemplu

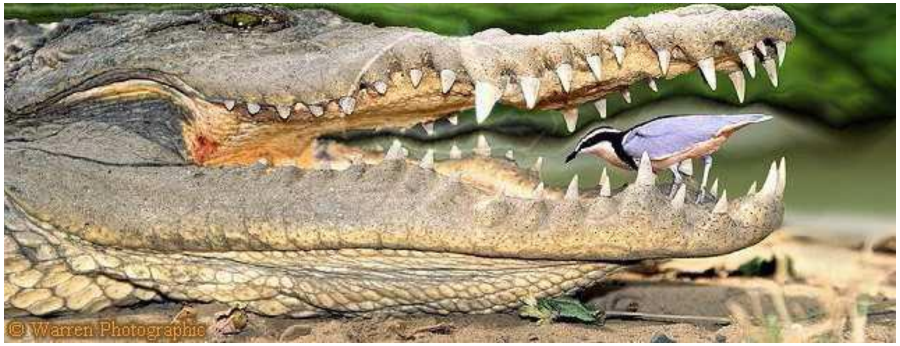
Slide adaptat după K. Grauman

Rezultate

Redimensionare cu păstrarea conținutului

Redimensionare uzuală (imresize)

Rezultate



Redimensionare cu păstrarea conținutului

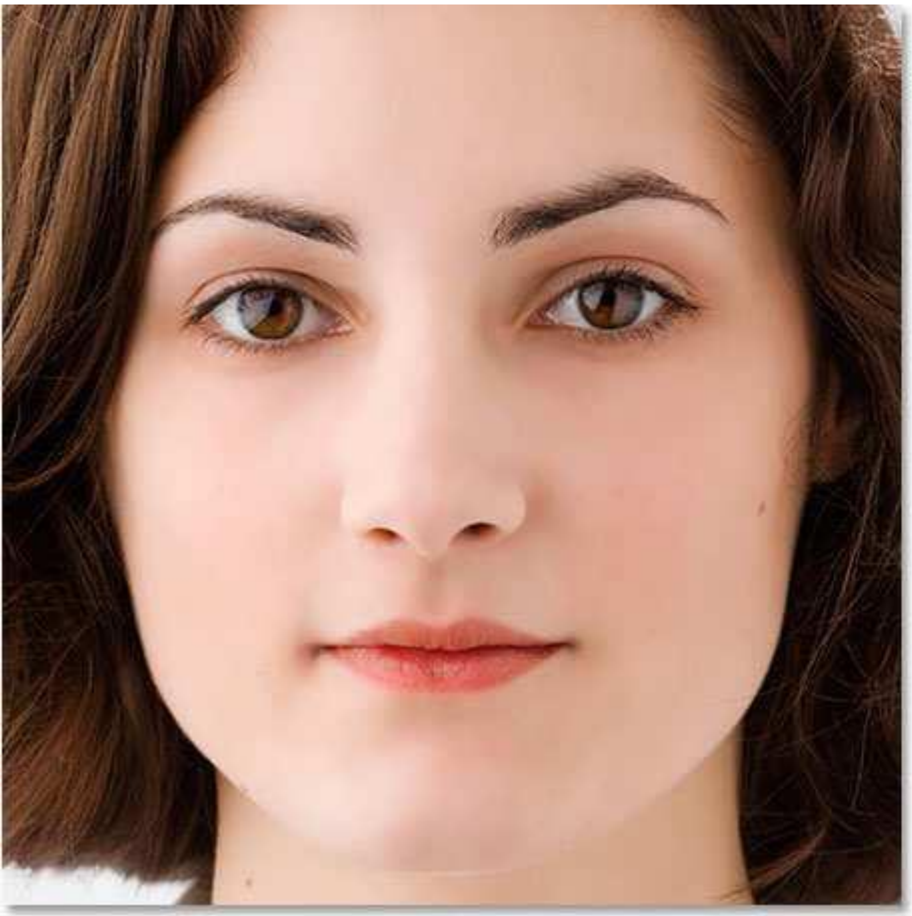
Redimensionare uzuală (imresize)

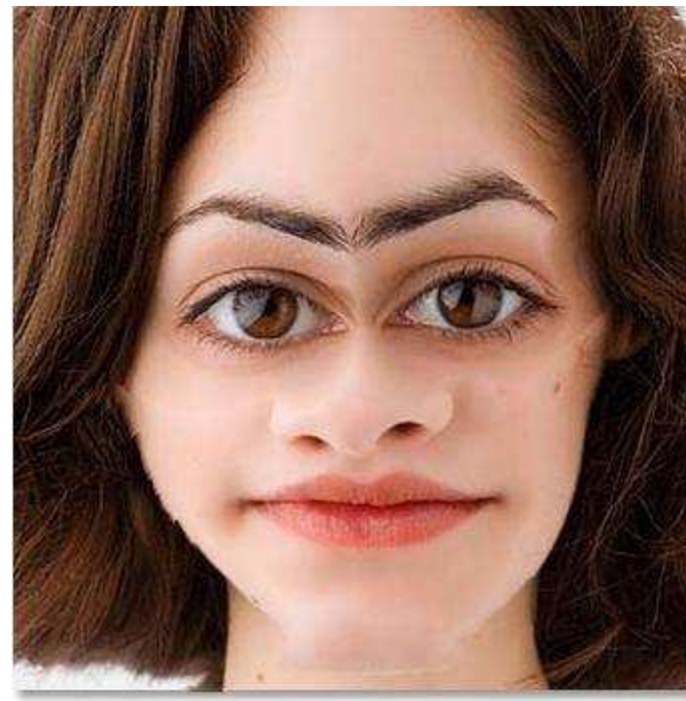
Slide adaptat după K. Grauman

Rezultate - failures

Slide adaptat după K. Grauman

Rezultate - failures



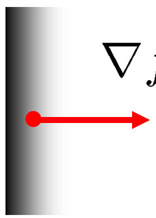


Eliminarea unei regiuni din imagine

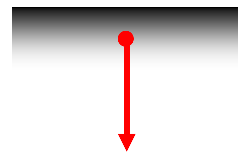
Cum transformăm gradienții în muchii?

Gradientul unei imagini

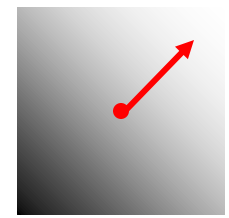
Gradientul unei imagini în fiecare punct: $\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right]$



$$\nabla f = \left[\frac{\partial f}{\partial x}, 0\right]$$



$$\nabla f = \left[0, \frac{\partial f}{\partial y}\right]$$



$$\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right]$$

Gradientul arată direcția celei mai rapide schimbări în intensitate

- Care este legătura dintre direcția gradientului si direcția muchiei?
- Direcția gradientului este perpendiculară pe direcția muchiei

Direcția gradientului este dată de: $\theta = \tan^{-1}\left(\frac{\partial f}{\partial y} / \frac{\partial f}{\partial x}\right)$

Caracterizăm o muchie prin magnitudinea gradientului:

$$\|\nabla f\| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2} \quad \text{sau} \quad \|\nabla f\| = \left|\frac{\partial f}{\partial x}\right| + \left|\frac{\partial f}{\partial y}\right|$$

Metode de detectare a muchiilor

Putem găsi muchii după 6 metode:

1. metoda Sobel
2. metoda Prewitt
3. metoda Roberts
4. metoda Laplacian-ului unei funcții Gaussiene
5. metoda “zero-crossings”
6. metoda Canny

Metodele Sobel + Roberts + Prewitt

Găsește muchii ca punctele unde gradientul are valori mari (mai mari decât un prag). Calculează gradientul pe baza filtrelor corespunzătoare (Sobel, Roberts, Prewitt).

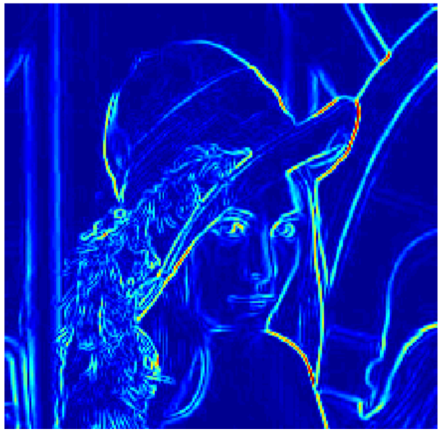
Prewitt: $M_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$; $M_y = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$

Sobel: $M_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$; $M_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$

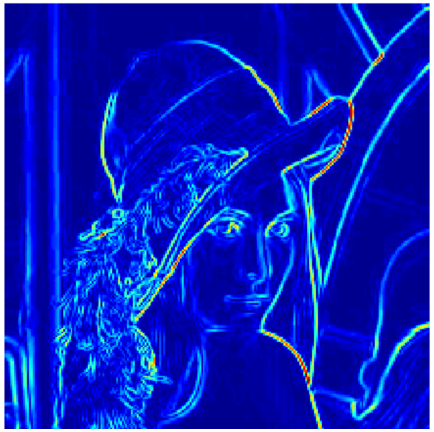
Roberts: $M_x = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$; $M_y = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$

Metodele Sobel + Roberts + Prewitt

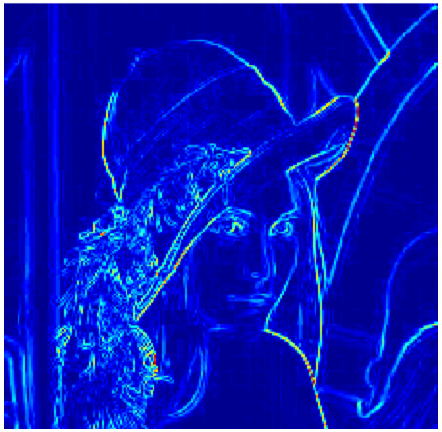
Găsește muchii ca punctele unde gradientul are valori mari (mai mari decât un prag). Calculează gradientul pe baza filtrelor corespunzătoare (Sobel, Roberts, Prewitt).



Gradientul imaginii calculat prin metoda Sobel



Gradientul imaginii calculat prin metoda Roberts



Gradientul imaginii calculat prin metoda Prewitt

Metodele Sobel + Roberts + Prewitt

Găsește muchii ca punctele unde gradientul are valori mari (mai mari decât un prag). Calculează gradientul pe baza filtrelor corespunzătoare (Sobel, Roberts, Prewitt).

Sobel

Roberts

Prewitt

Avem nevoie de un prag = threshold pentru a obține muchii (pixelii din muchie = edgels)
Toți pixelii cu gradient > prag devin 1 (edgels – parte din muchie), restul devin 0.

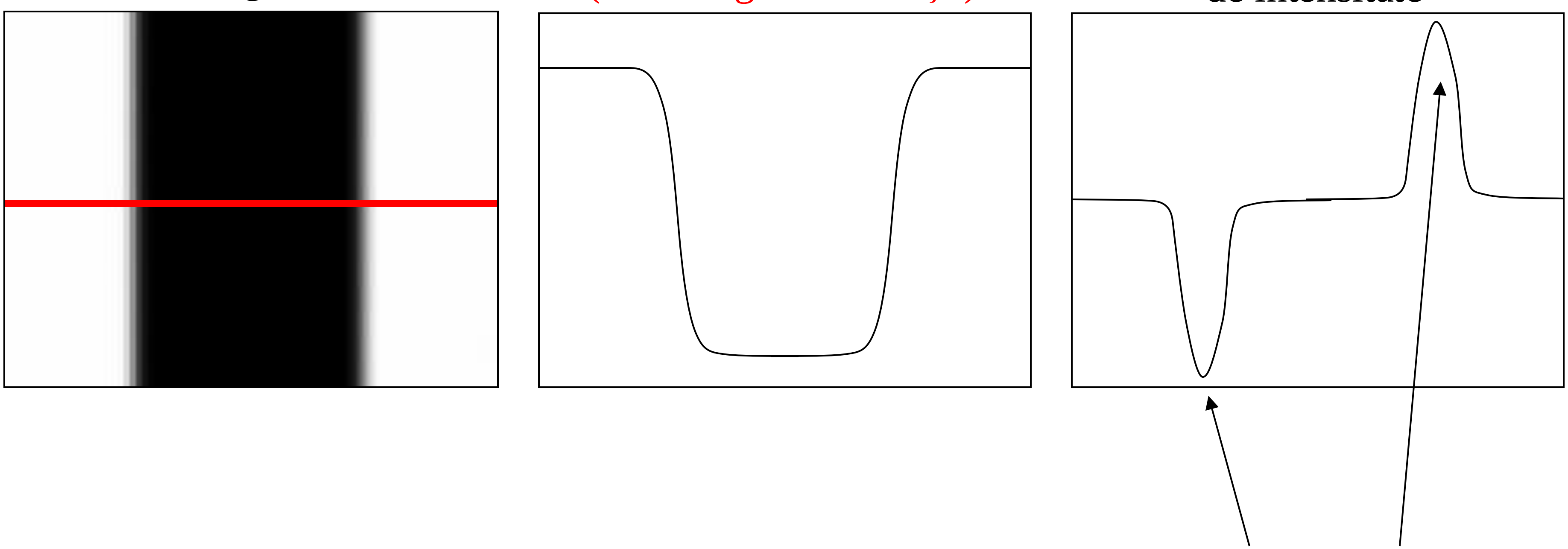
Obținerea filtrului Laplacian

O muchie este locul în care se produce o schimbare bruscă a funcției de intensitate

imagine

funcția de intensitate
(de-a lungul liniei roşii)

derivata funcției
de intensitate



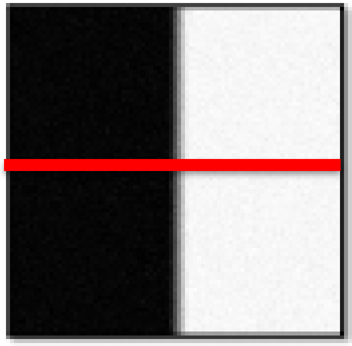
muchiile corespund punctelor
de extrem ale derivatei

Slide adaptat după S. Lazebnik

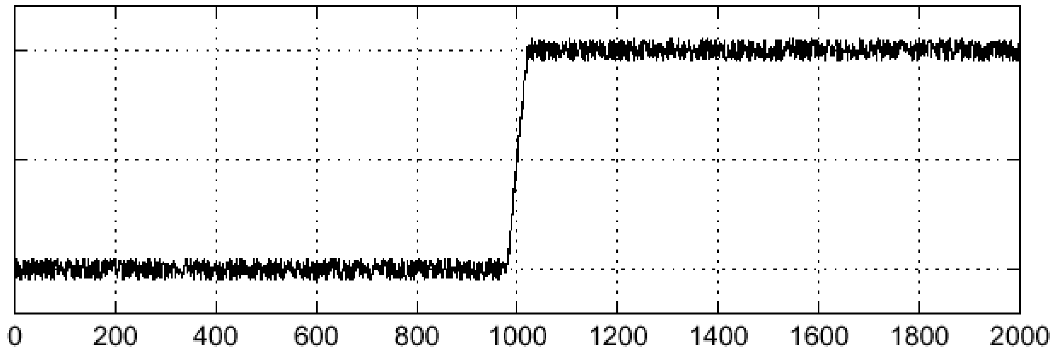
Efectul zgomotului – 1D

Considerăm linia roşie a imaginii

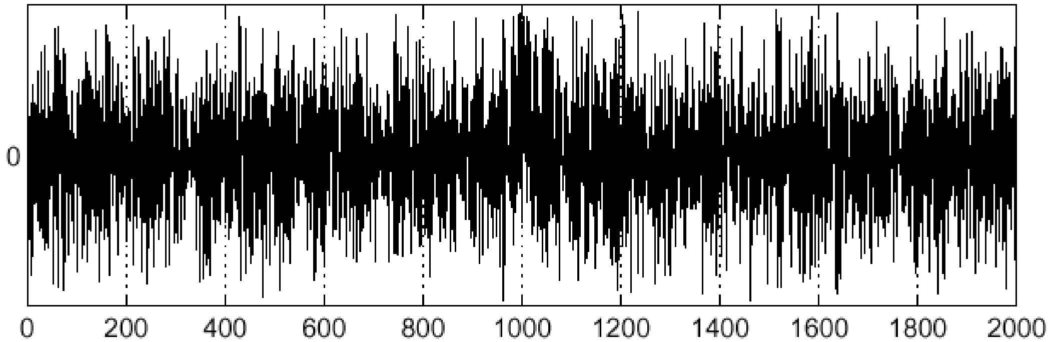
– plotăm intensitatea în funcție de poziție



$f(x)$



$\frac{d}{dx}f(x)$



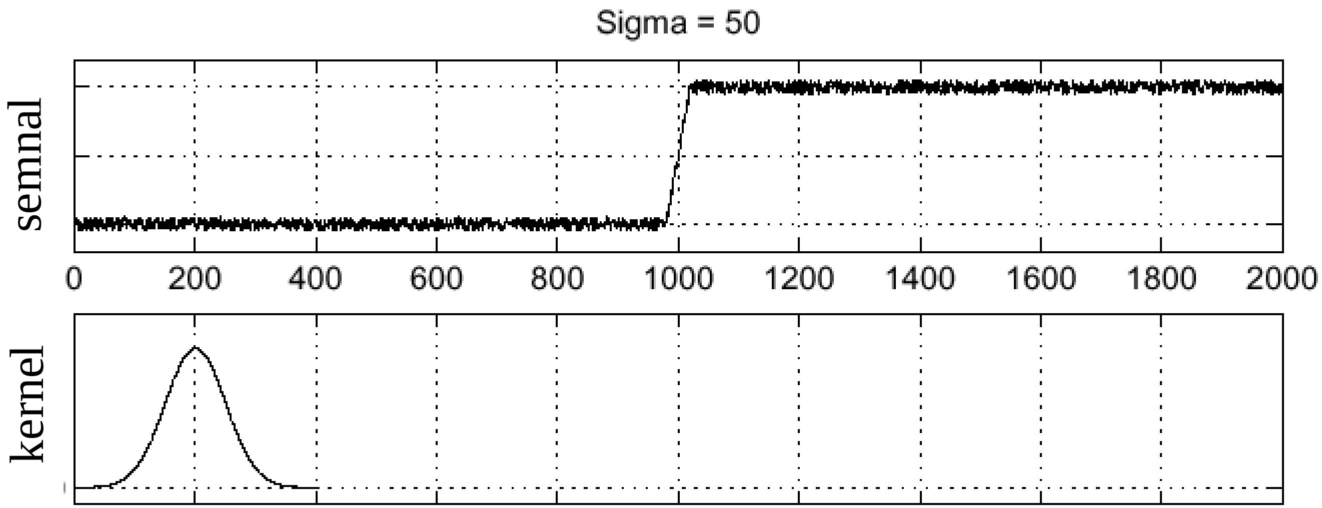
Unde este muchia?

Slide adaptat după Steve Seitz

Soluție: blurăm cu un filtru Gaussian – 1D

f

filtru Gaussian h



convoluție

Unde este muchia?

Căutăm punctele de extrem local în $\frac{\partial}{\partial x}(h \star f)$ = zerourile funcției $\frac{\partial^2}{\partial x^2}(h \star f)$

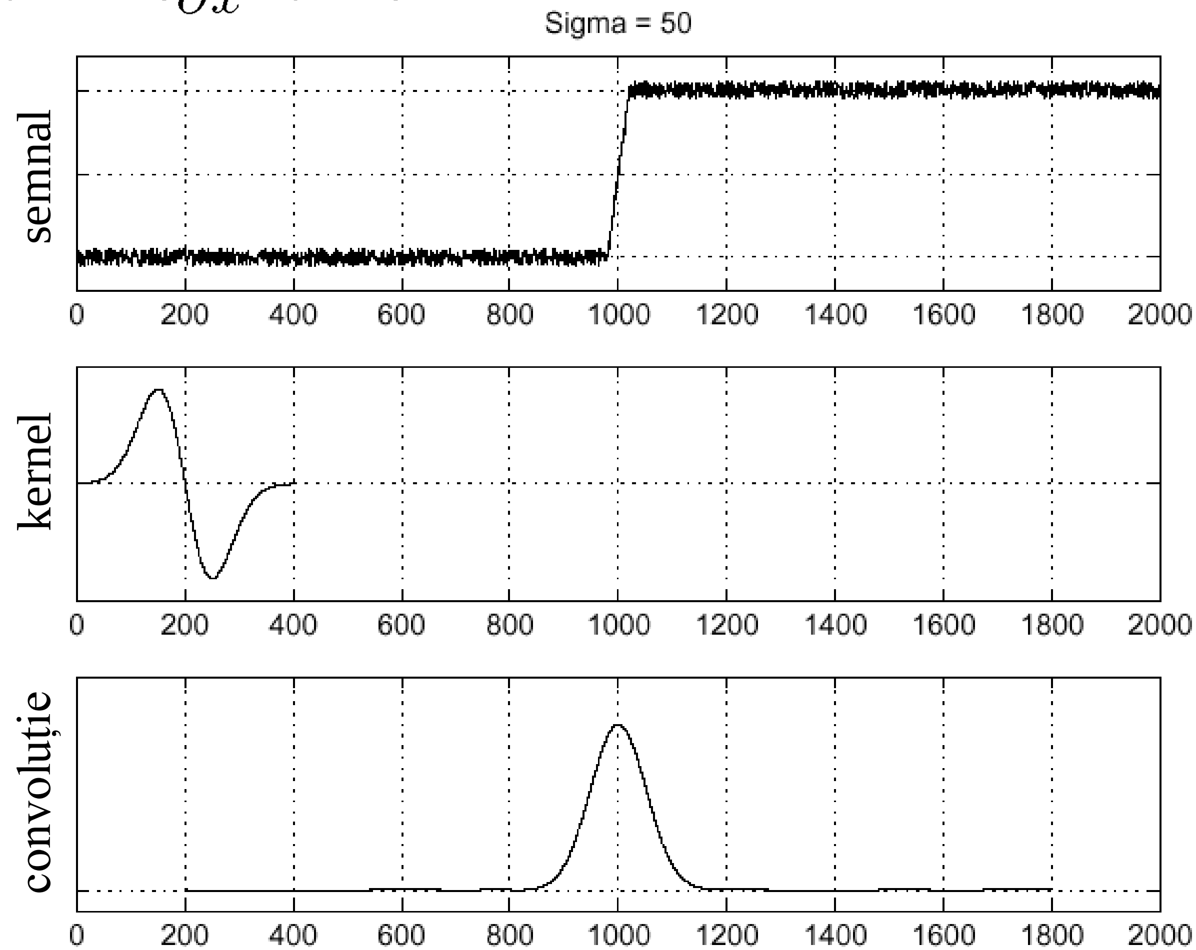
Proprietatea de diferențiabilitate a convoluției

$$\frac{\partial}{\partial x}(h \star f) = (\frac{\partial}{\partial x}h) \star f$$

f

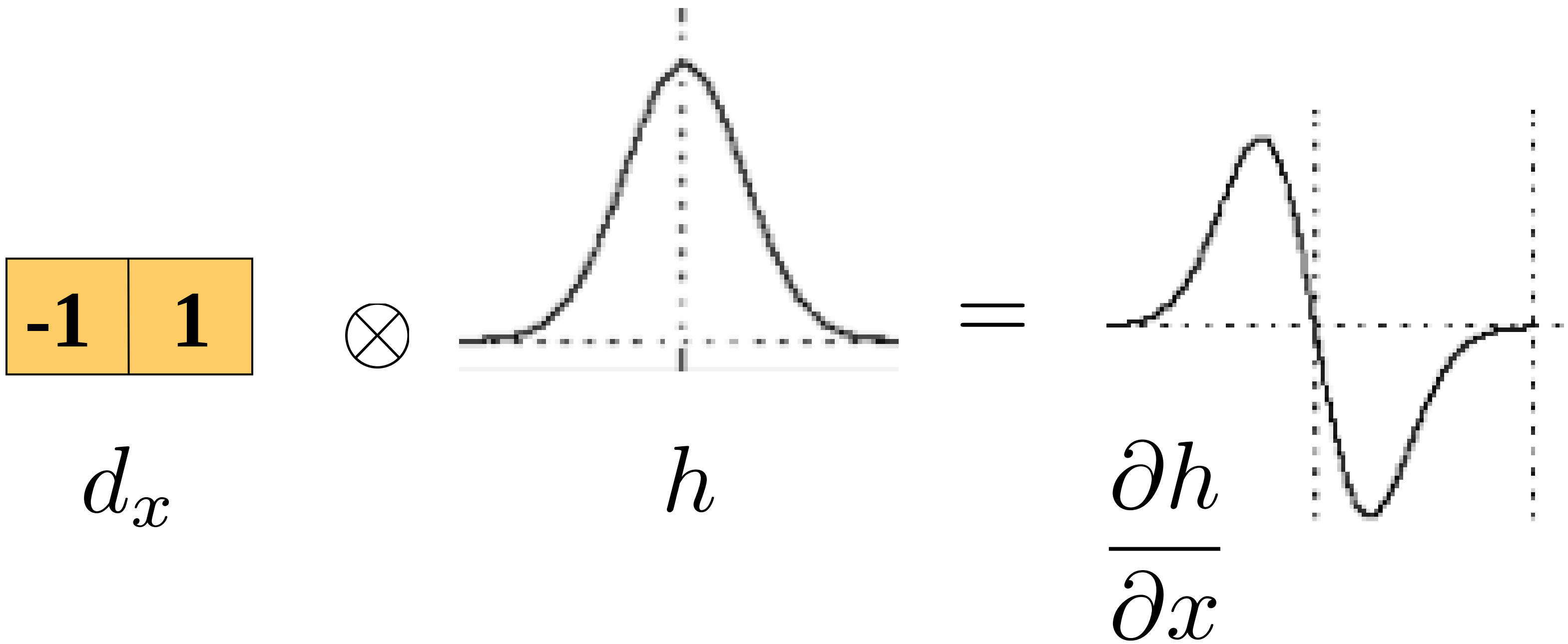
$\frac{\partial}{\partial x}h$

$(\frac{\partial}{\partial x}h) \star f$



Slide adaptat după Steve Seitz

1D



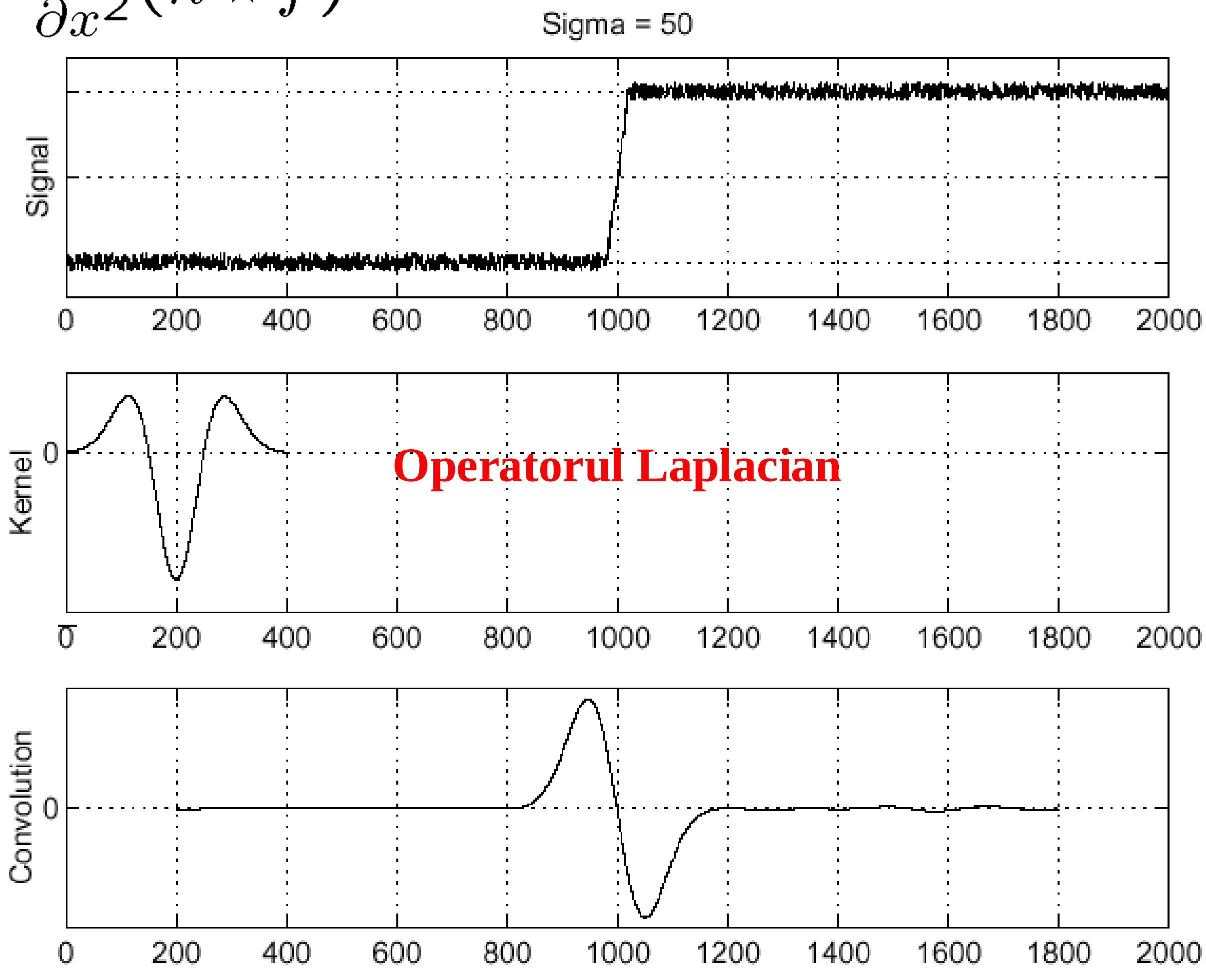
Laplacian-ul unei funcții Gaussiene – 1D

Considerăm $\frac{\partial^2}{\partial x^2}(h \star f)$

f

$\frac{\partial^2}{\partial x^2}h$

$(\frac{\partial^2}{\partial x^2}h) \star f$



Unde este muchia?

Zero-crossing ale funcției $(\frac{\partial^2}{\partial x^2}h) \star f$

Slide adaptat după Steve Seitz

Derivata filtrelor Gaussiane - 2D

$$d_x \otimes (h \otimes I) = (d_x \otimes h) \otimes I$$

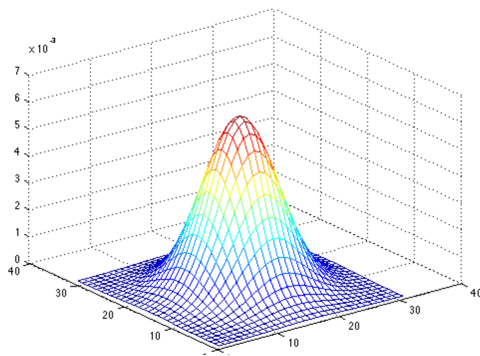
filtru de derivare filtru Gaussian imagine

0	**0**	**0**
0	**-1**	**1**
0	**0**	**0**

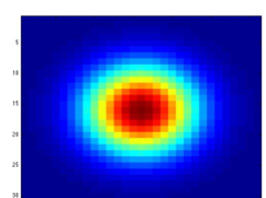
$$\otimes \begin{bmatrix} 0.0030 & 0.0133 & 0.0219 & 0.0133 & 0.0030 \\ 0.0133 & 0.0596 & 0.0983 & 0.0596 & 0.0133 \\ 0.0219 & 0.0983 & 0.1621 & 0.0983 & 0.0219 \\ 0.0133 & 0.0596 & 0.0983 & 0.0596 & 0.0133 \\ 0.0030 & 0.0133 & 0.0219 & 0.0133 & 0.0030 \end{bmatrix}$$

0	**0**	**0**
0	**-1**	**1**
0	**0**	**0**

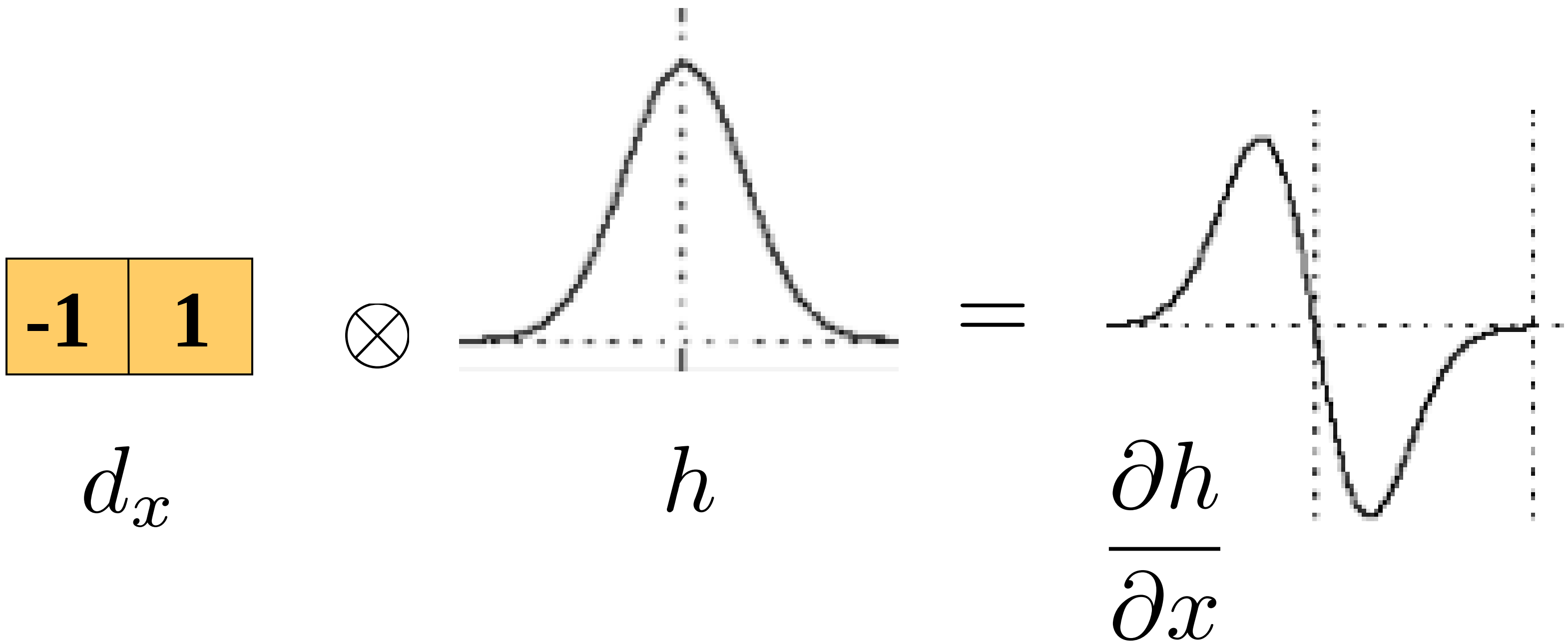
$\otimes$



$=$ **?**



1D



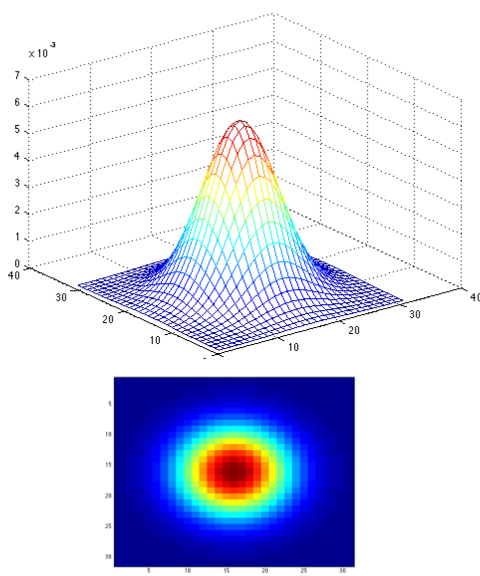
Derivata filtrelor Gaussiane - 2D

$$d_x \otimes (h \otimes I) = (d_x \otimes h) \otimes I$$

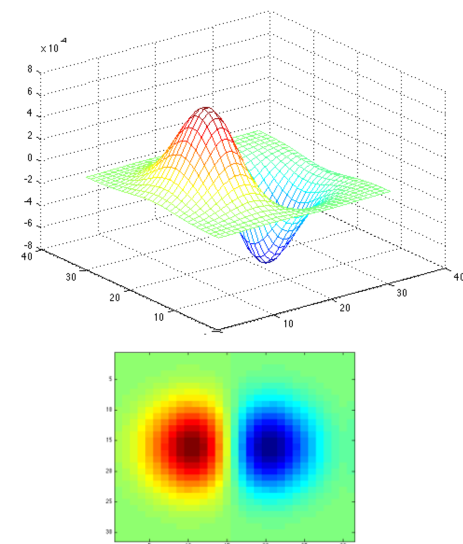
filtru de derivare   filtru Gaussian   imagine

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & 0 \end{bmatrix} \otimes \begin{bmatrix} 0.0030 & 0.0133 & 0.0219 & 0.0133 & 0.0030 \\ 0.0133 & 0.0596 & 0.0983 & 0.0596 & 0.0133 \\ 0.0219 & 0.0983 & 0.1621 & 0.0983 & 0.0219 \\ 0.0133 & 0.0596 & 0.0983 & 0.0596 & 0.0133 \\ 0.0030 & 0.0133 & 0.0219 & 0.0133 & 0.0030 \end{bmatrix}$$

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



$$=$$

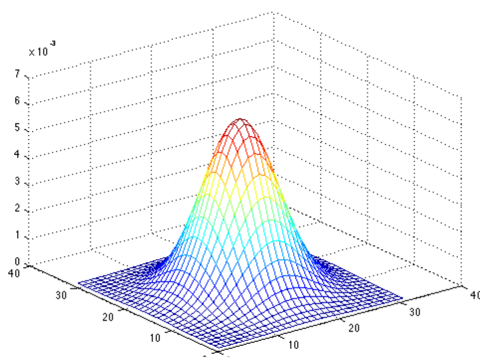


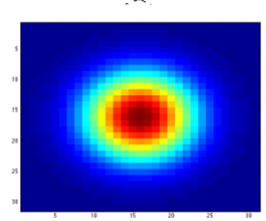
Derivata filtrelor Gaussiane - 2D

$$d_y \otimes (h \otimes I) = (d_y \otimes h) \otimes I$$

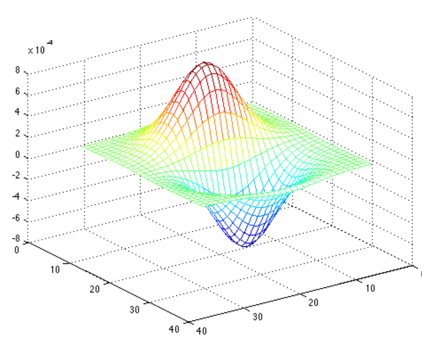
$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 1 & 0 \end{bmatrix} \otimes \begin{bmatrix} 0.0030 & 0.0133 & 0.0219 & 0.0133 & 0.0030 \\ 0.0133 & 0.0596 & 0.0983 & 0.0596 & 0.0133 \\ 0.0219 & 0.0983 & 0.1621 & 0.0983 & 0.0219 \\ 0.0133 & 0.0596 & 0.0983 & 0.0596 & 0.0133 \\ 0.0030 & 0.0133 & 0.0219 & 0.0133 & 0.0030 \end{bmatrix}$$

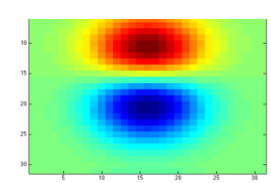
$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 1 & 0 \end{bmatrix} \otimes$$



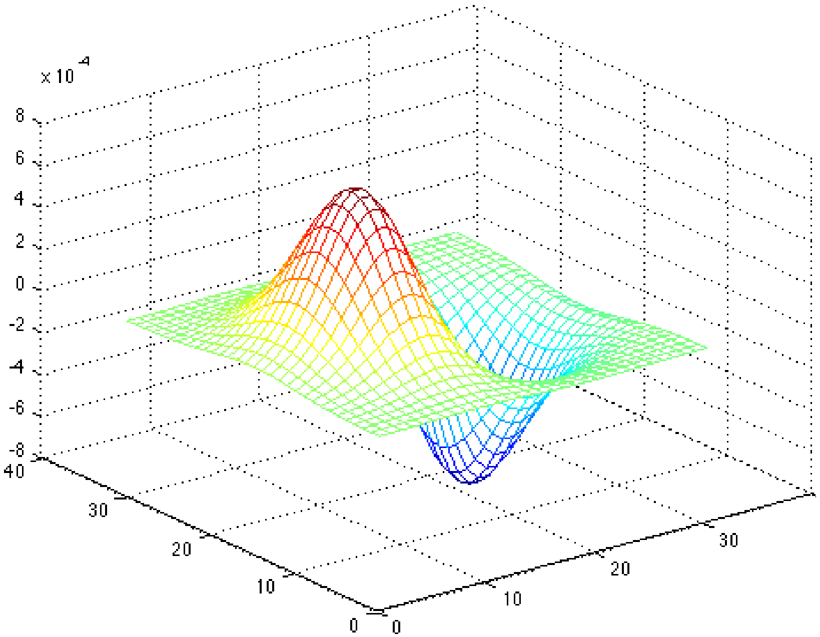


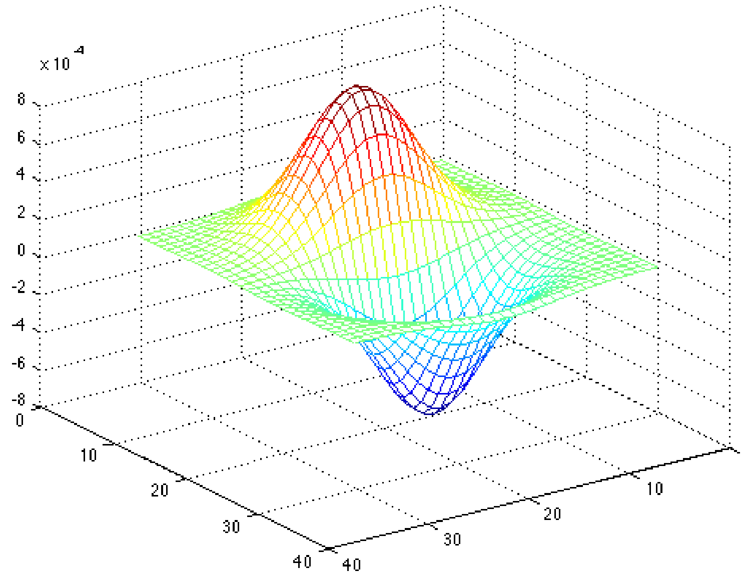
$$=$$





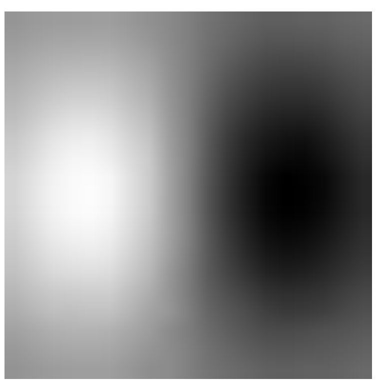
Derivata filtrelor Gaussiane - 2D

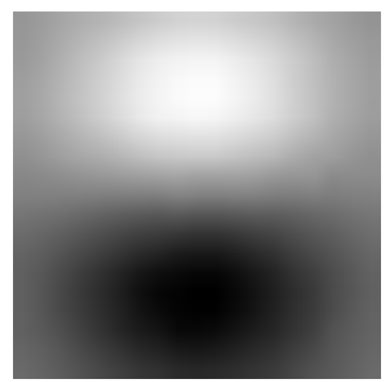




direcția x

direcția y





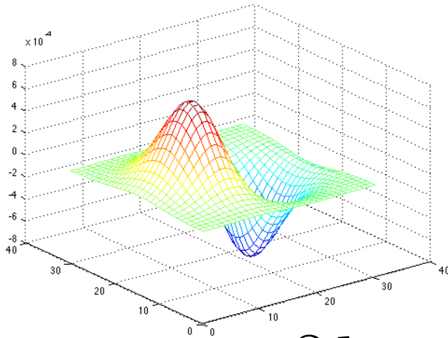
Slide adaptat după S. Lazebnik

Derivata a doua a filtrelor Gaussiane

0	**0**	**0**
0	**-1**	**1**
0	**0**	**0**

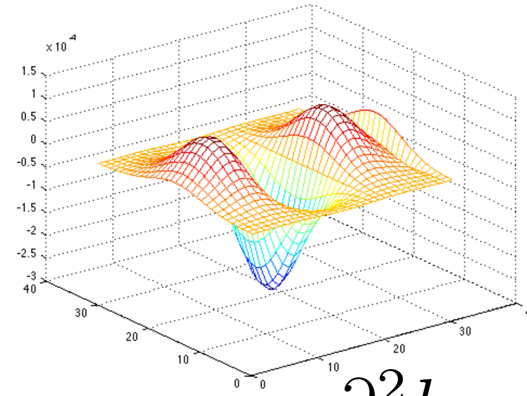
direcția x

$\otimes$



$$\frac{\partial h}{\partial x}$$

$=$

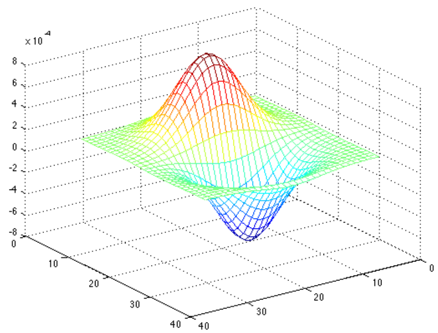


$$\frac{\partial^2 h}{\partial x^2}$$

0	**0**	**0**
0	**-1**	**0**
0	**1**	**0**

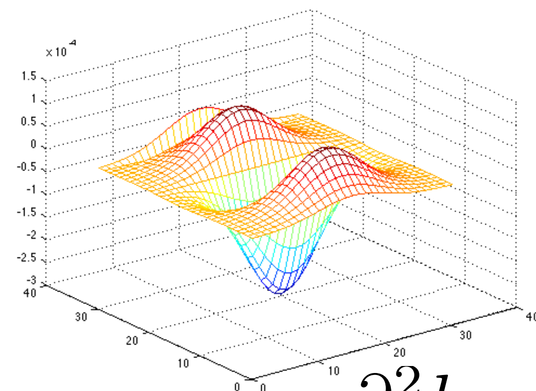
direcția y

$\otimes$



$$\frac{\partial h}{\partial y}$$

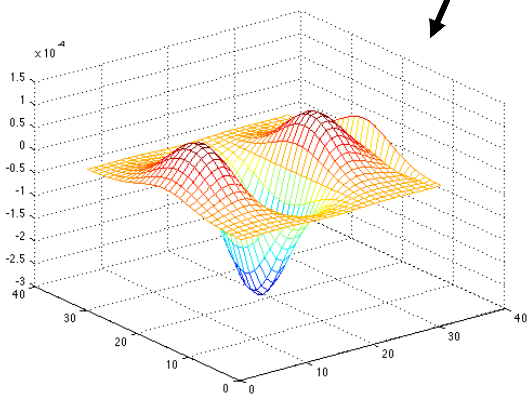
$=$



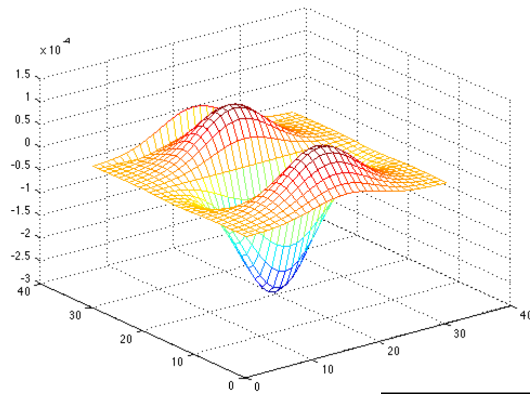
$$\frac{\partial^2 h}{\partial y^2}$$

Laplacian-ul unei funcții Gaussiene – 2D

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

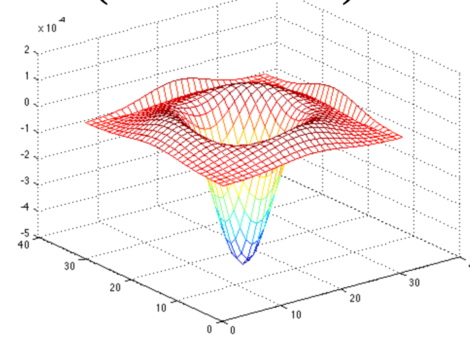


$+$



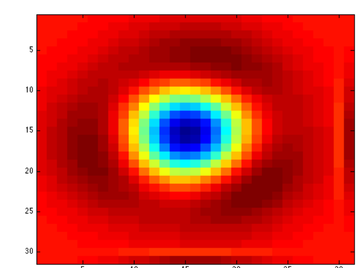
$=$

Forma generală a
unui filtru Laplacian
(mexican hat)



Exemplu particular
de filtru Laplacian

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



Metoda Laplacianul unei funcții Gaussiene

Sobel

Roberts

Prewitt

Laplacian

Proprietățile filtrelor

- Filtre de blurare (cursul trecut)
  1. valori pozitive;
  2. suma lor = 1 → pentru regiuni constante, output = input;
  3. gradul de blurare proporțional cu dimensiunea filtrului;
  4. elimină regiunile de pixeli cu varianță mare în intensitate (“high-frequency”); se mai numesc filtre trece-jos (“low-pass”)

- Filtre pentru detectarea muchiilor
  1. valori opuse pentru a avea un răspuns mare (în valoare absolută = modul) în regiuni cu contrast mare
  2. suma lor = 0 → pentru regiuni constante, răspuns = 0;
  3. cele mai mari valori în punctele de contrast maxim

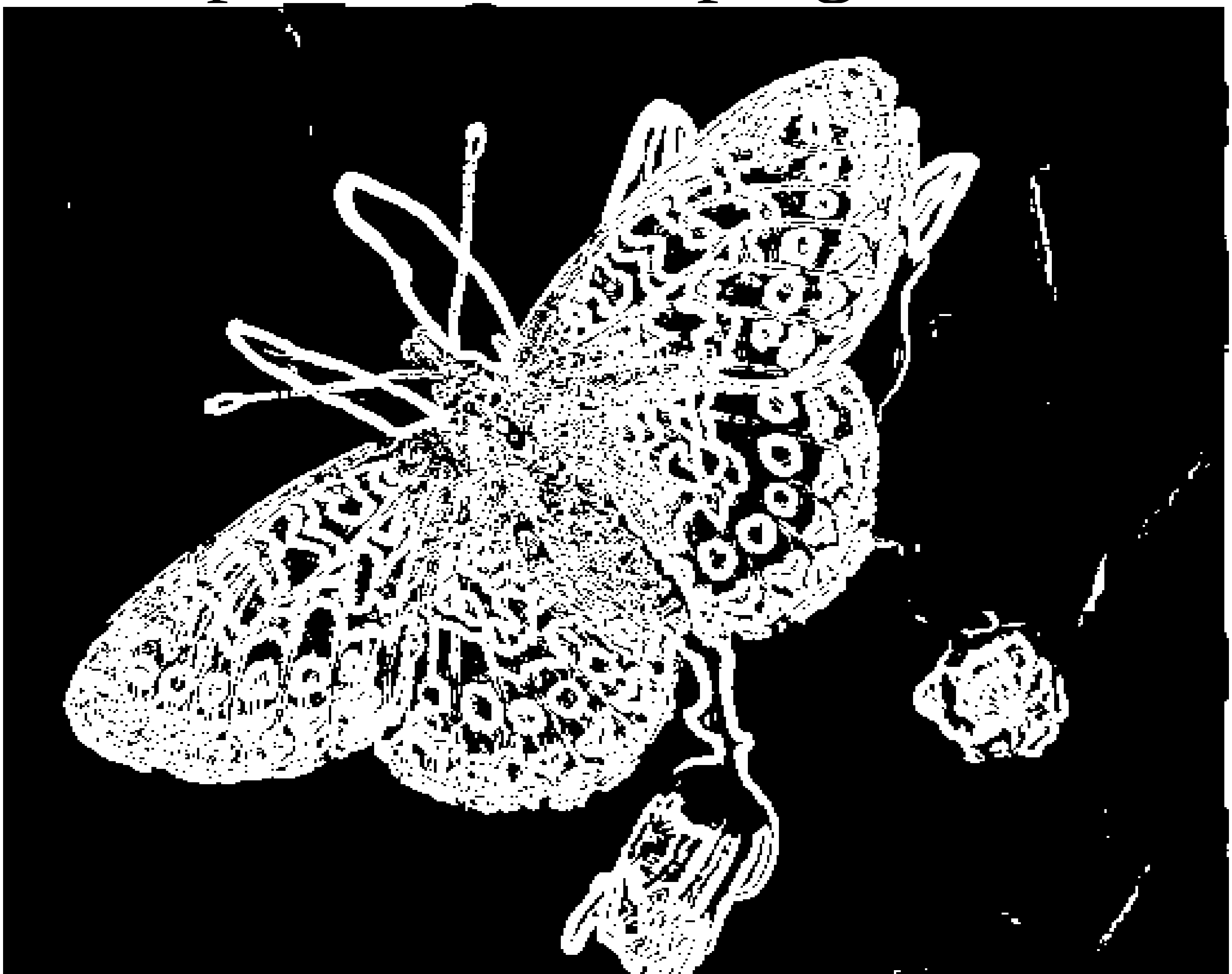
Slide adaptat după K. Grauman

Exemplu

Gradientul imaginii

Gradientul imaginii

Aplicarea unui prag mic



Aplicarea unui prag mare

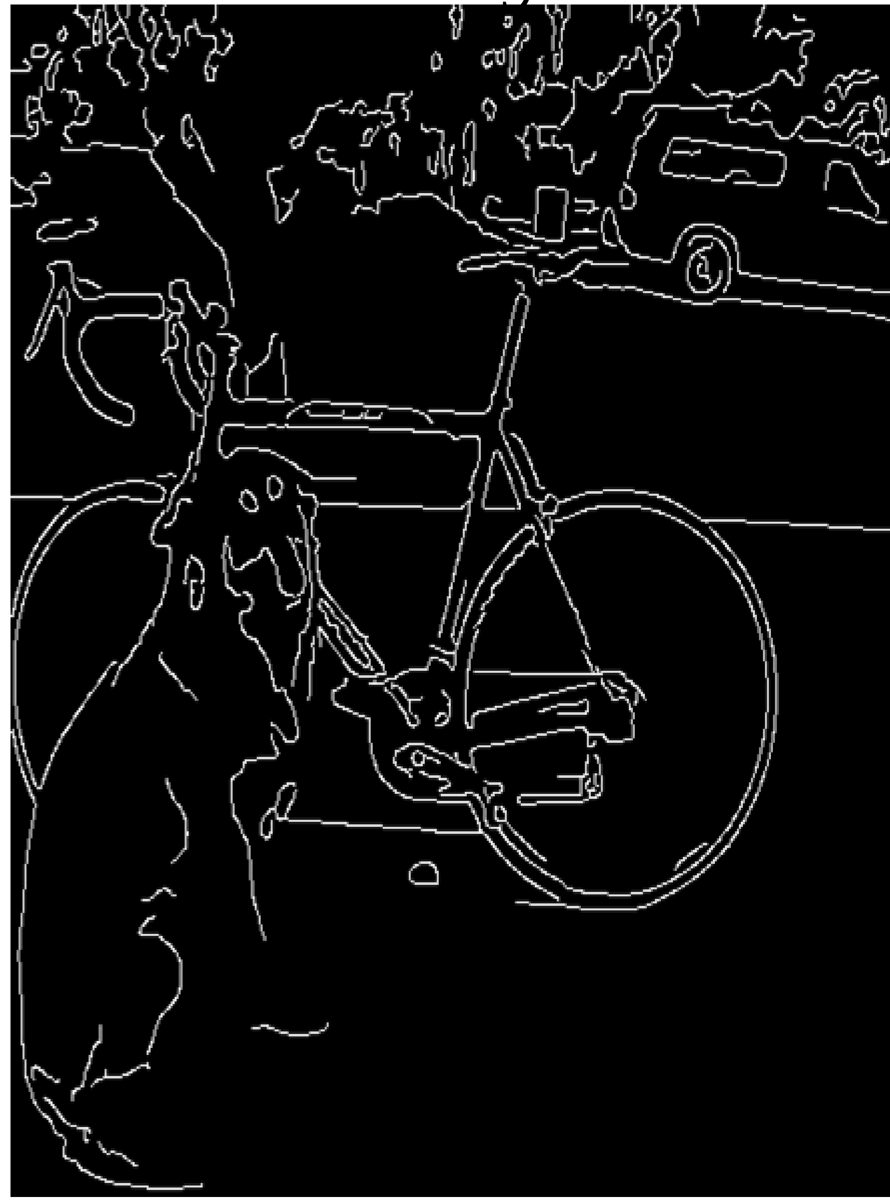
Metoda Canny

Imagine inițială

Gradientul imaginii

Rezultat metoda Canny

Imagine inițială



Imagine binară

Metoda Canny

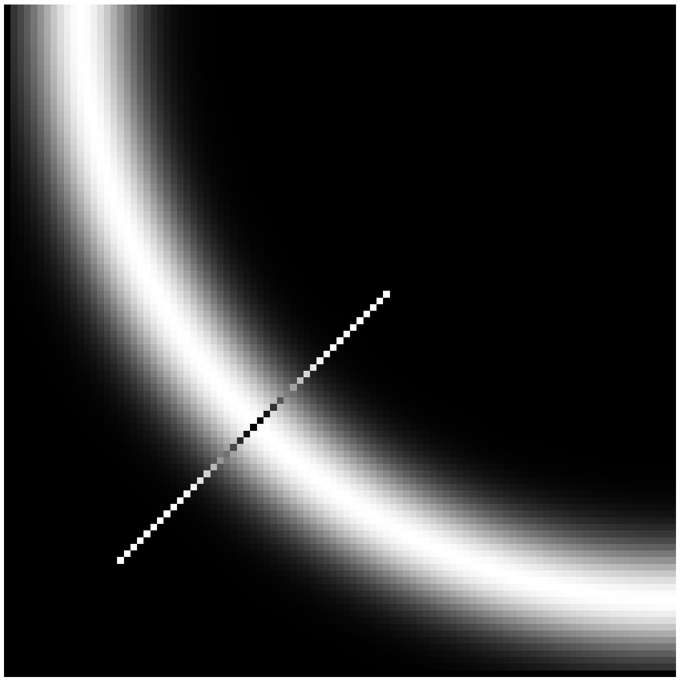
Cum transformăm aceste regiuni ale gradientului în muchii de lățime de un 1 pixel?

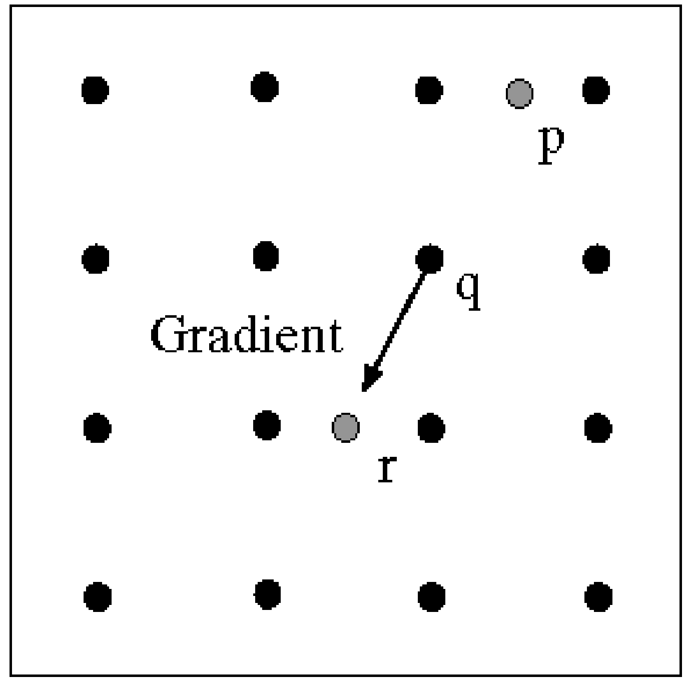
Slide adaptat după K. Grauman

Metoda Canny

Cum transformăm aceste regiuni ale gradientului în muchii de lățime de un 1 pixel?

Eliminarea non-maximelor

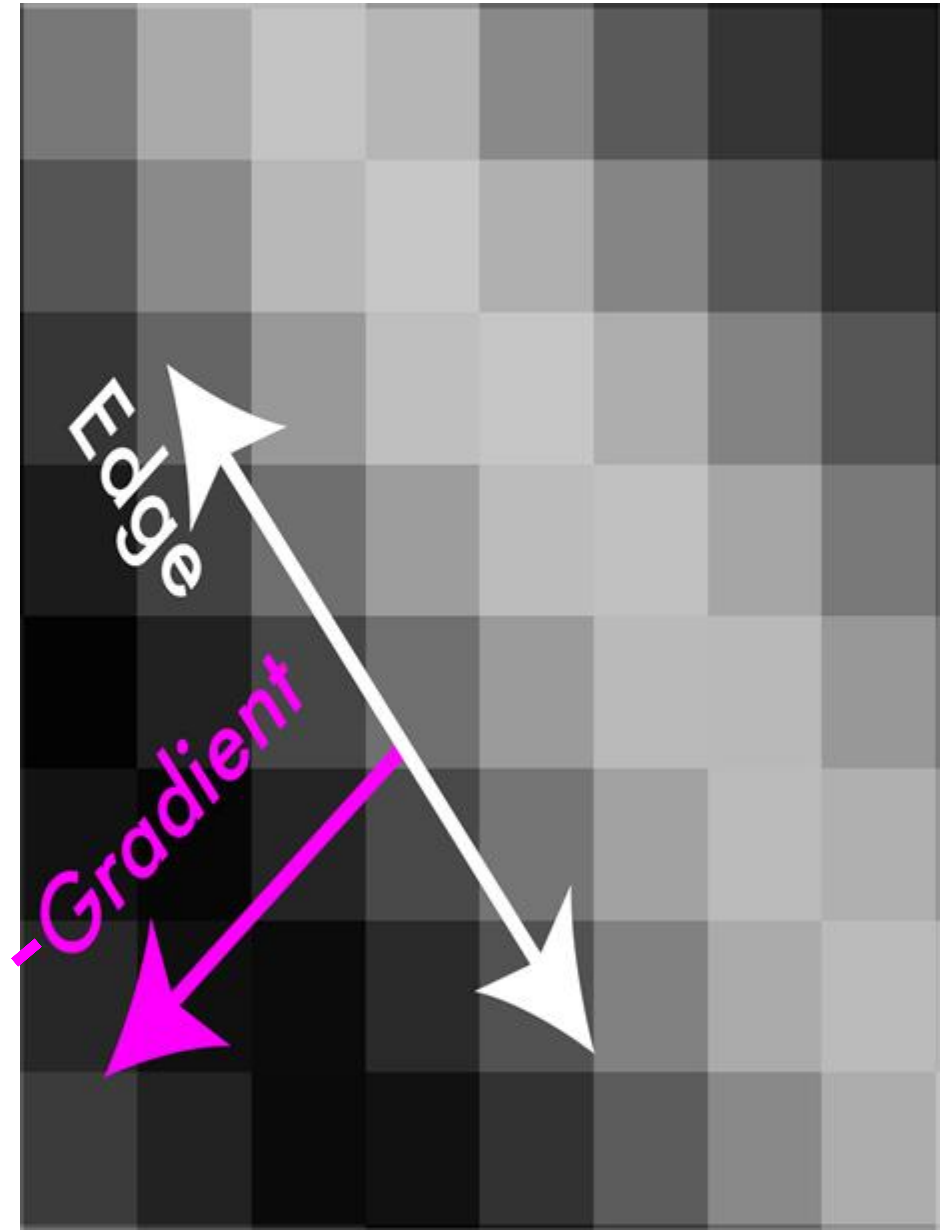




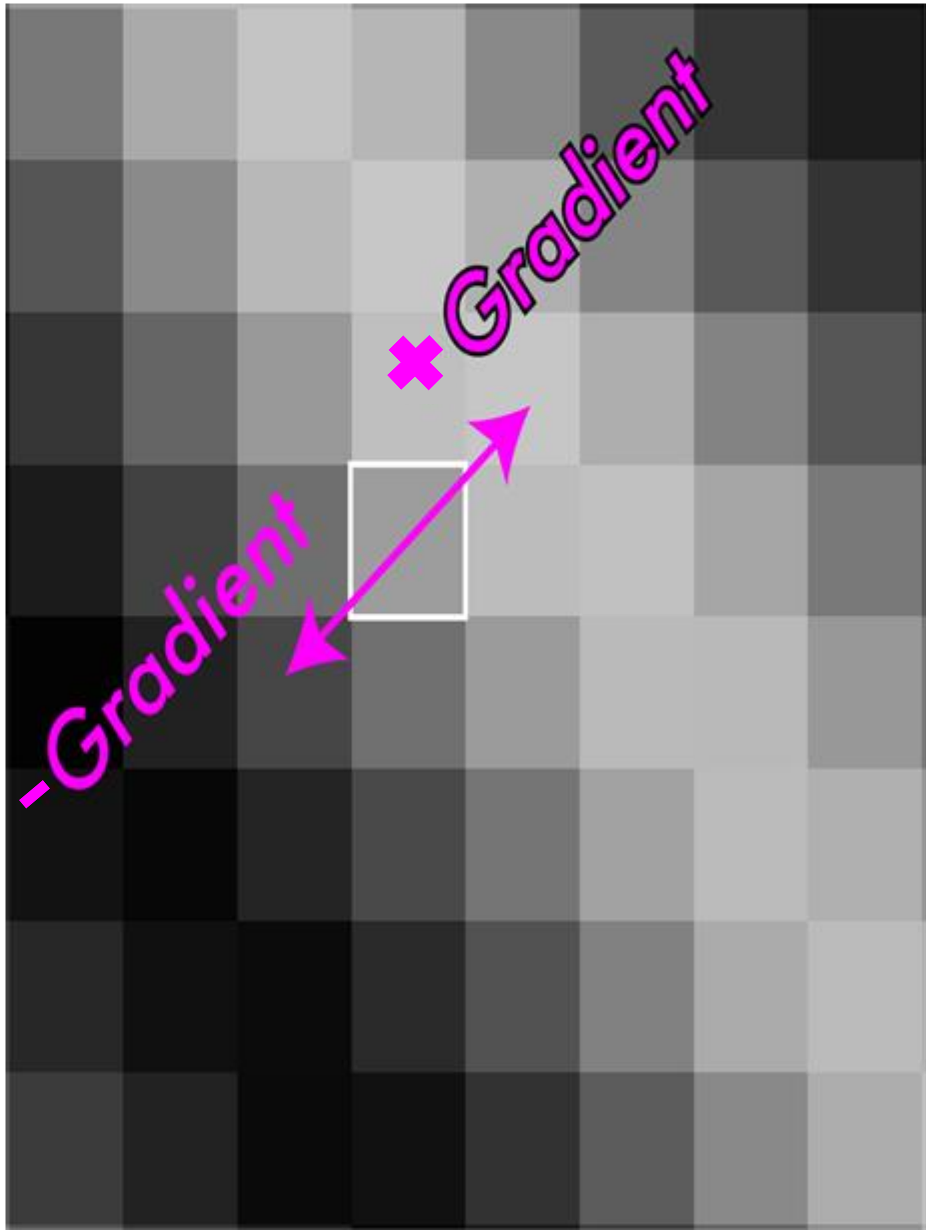
Verifică dacă pixelul curent este maxim local de-a lungul direcției gradientului, selectează maximul de-a lungul lățimii muchiei

– necesită verificarea cu pixelii p şi r obținuți prin interpolare

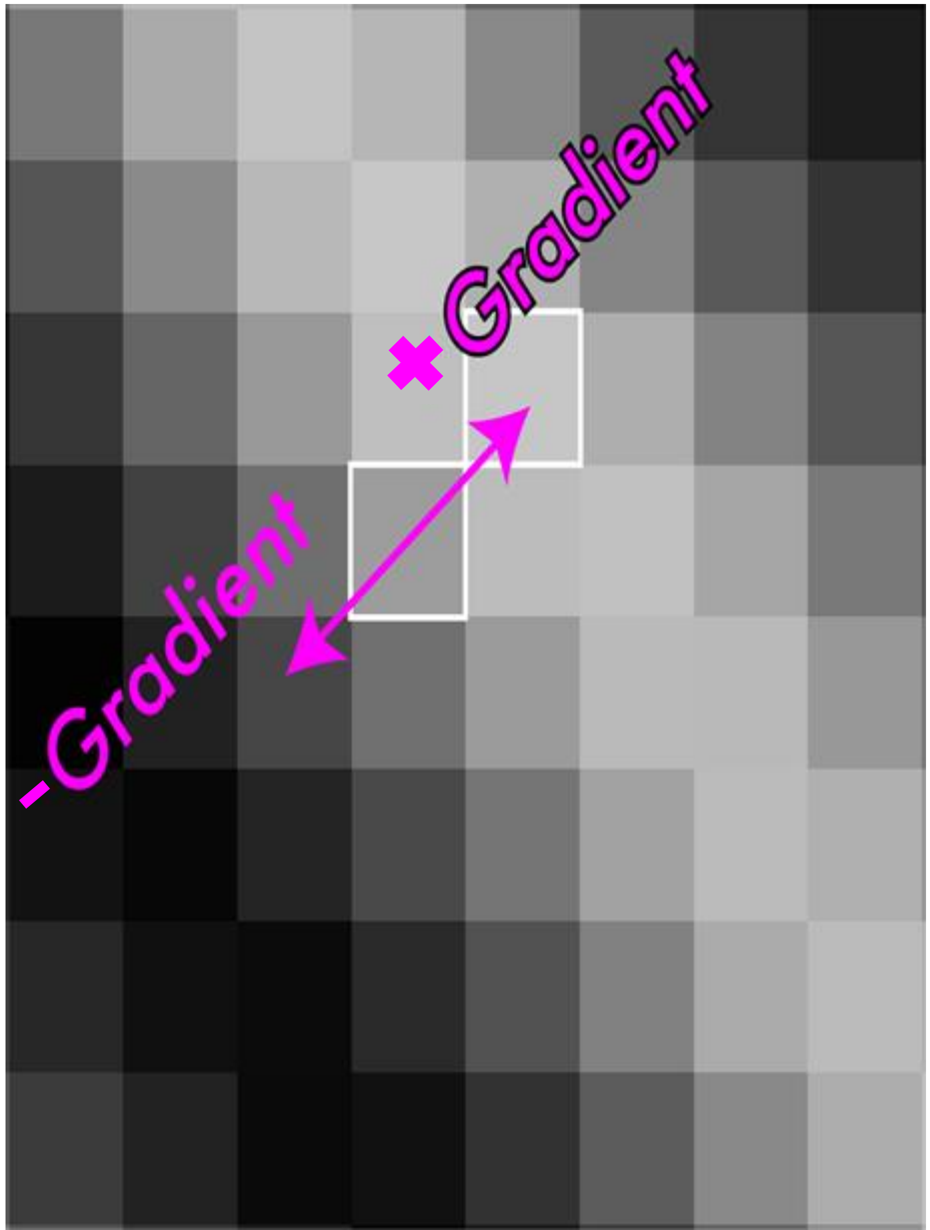
Eliminarea non-maximelor



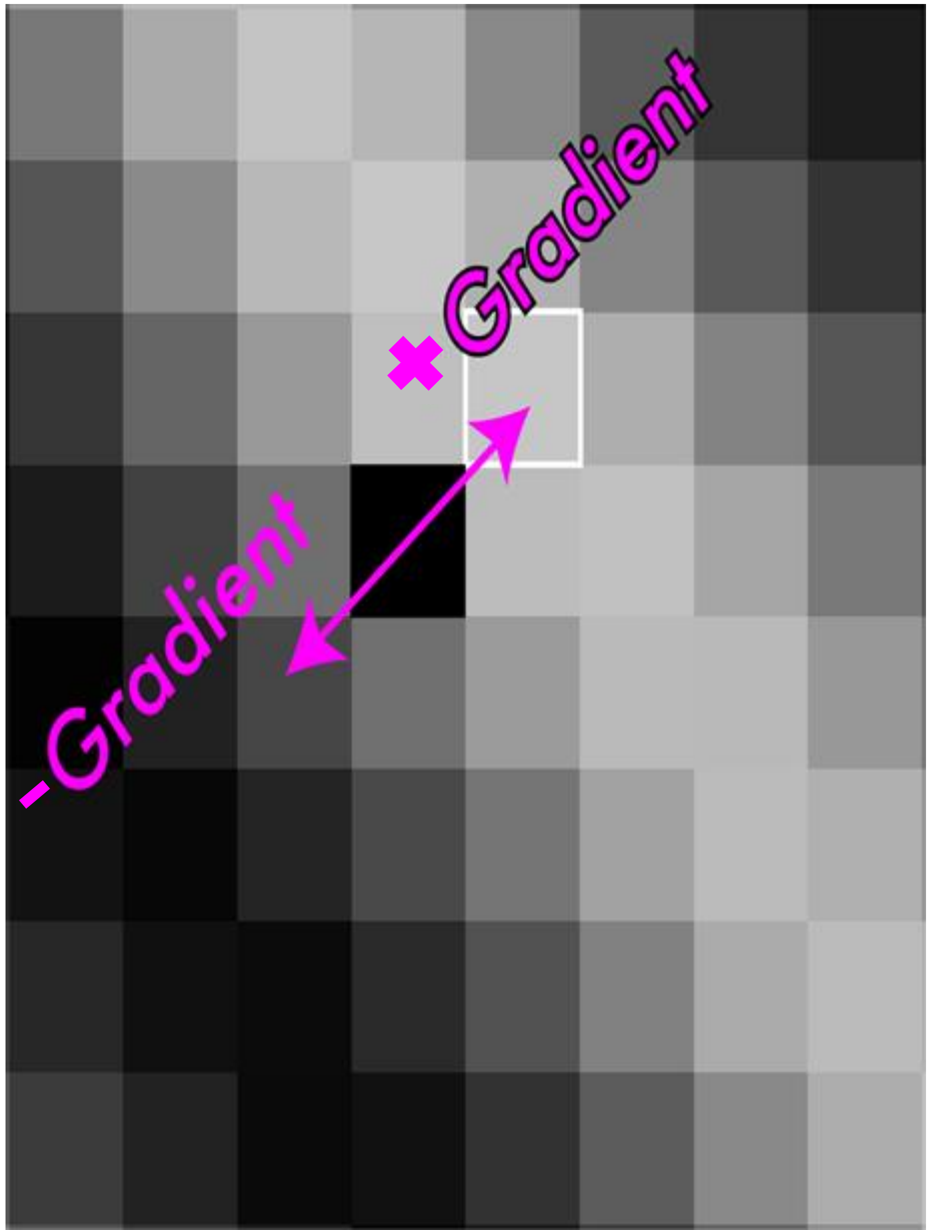
Eliminarea non-maximelor



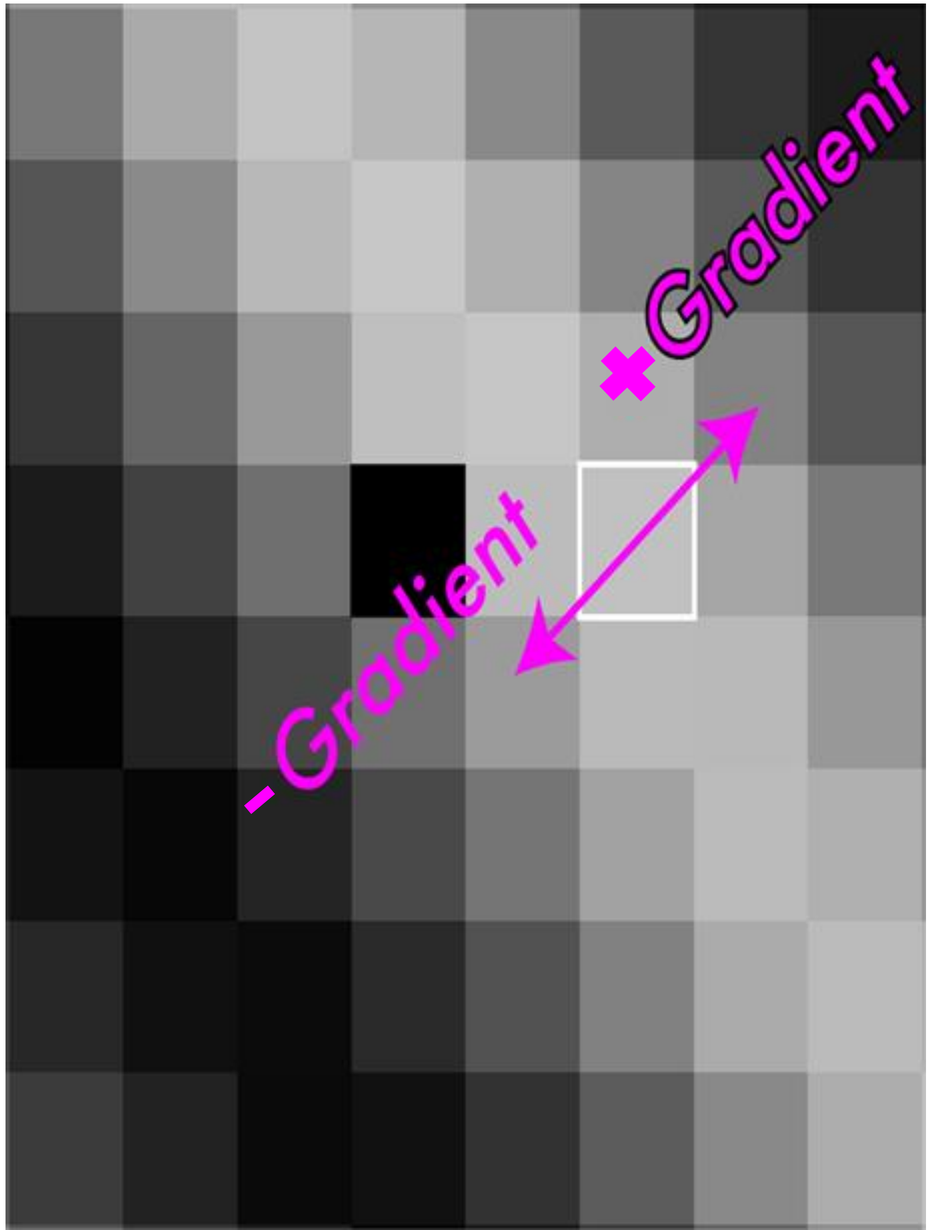
Eliminarea non-maximelor



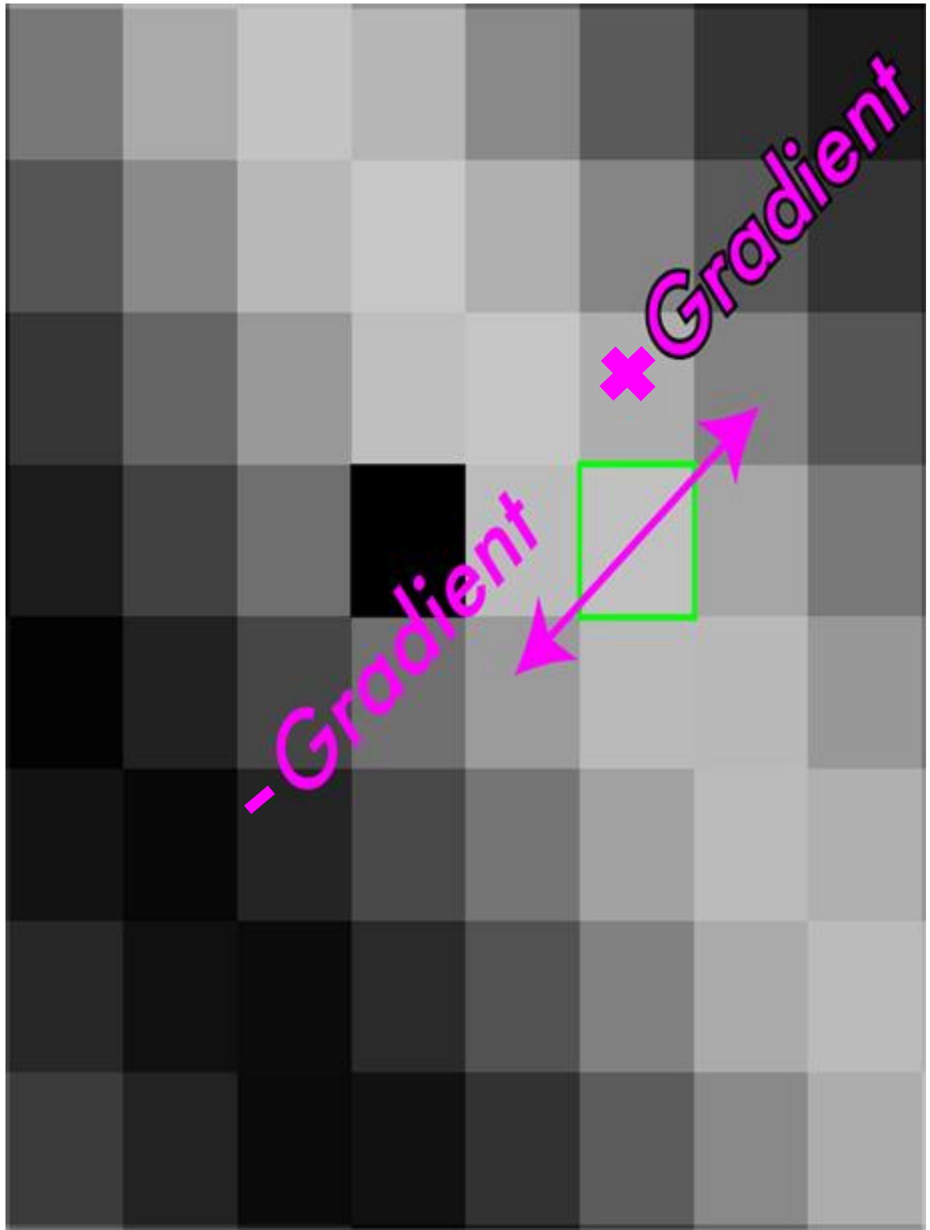
Eliminarea non-maximelor



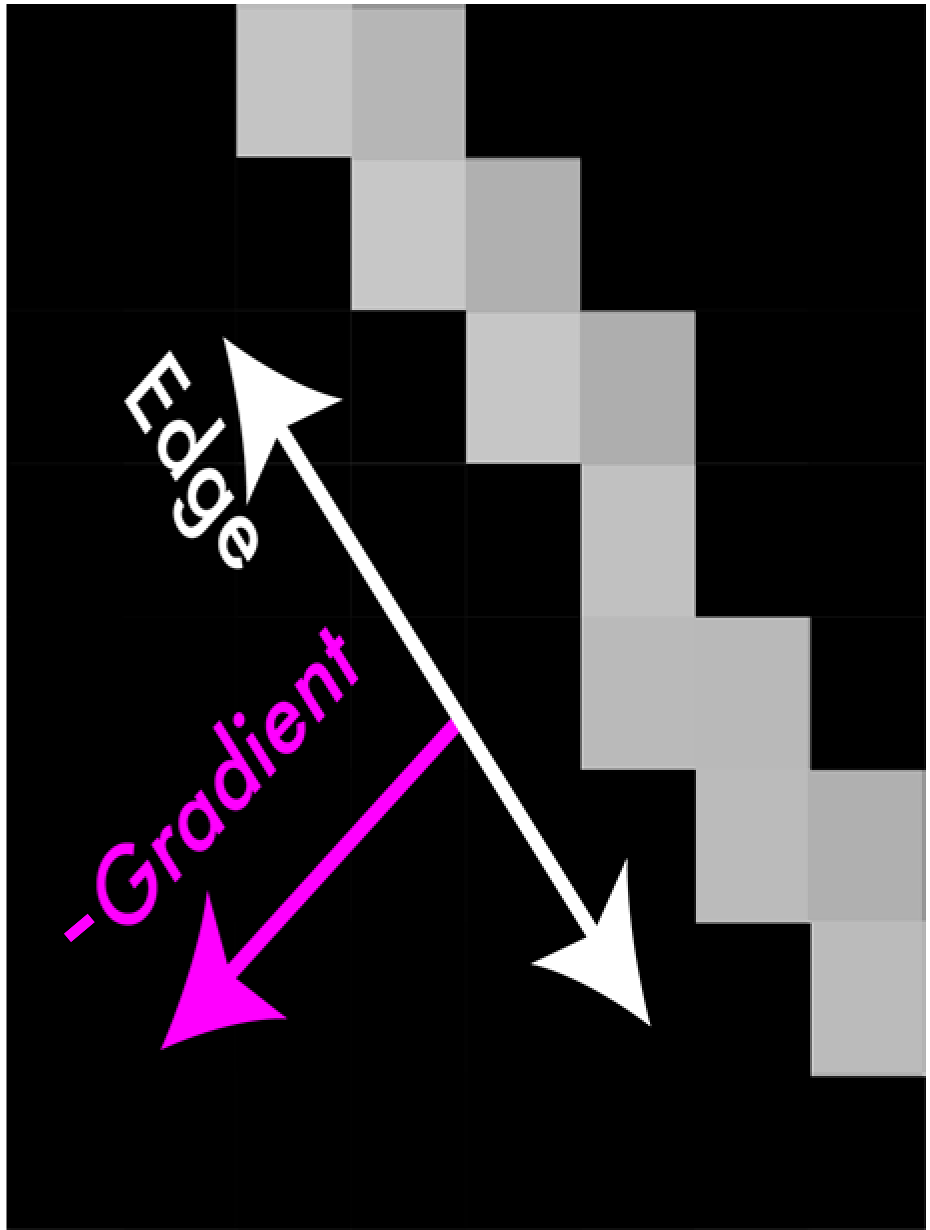
Eliminarea non-maximelor



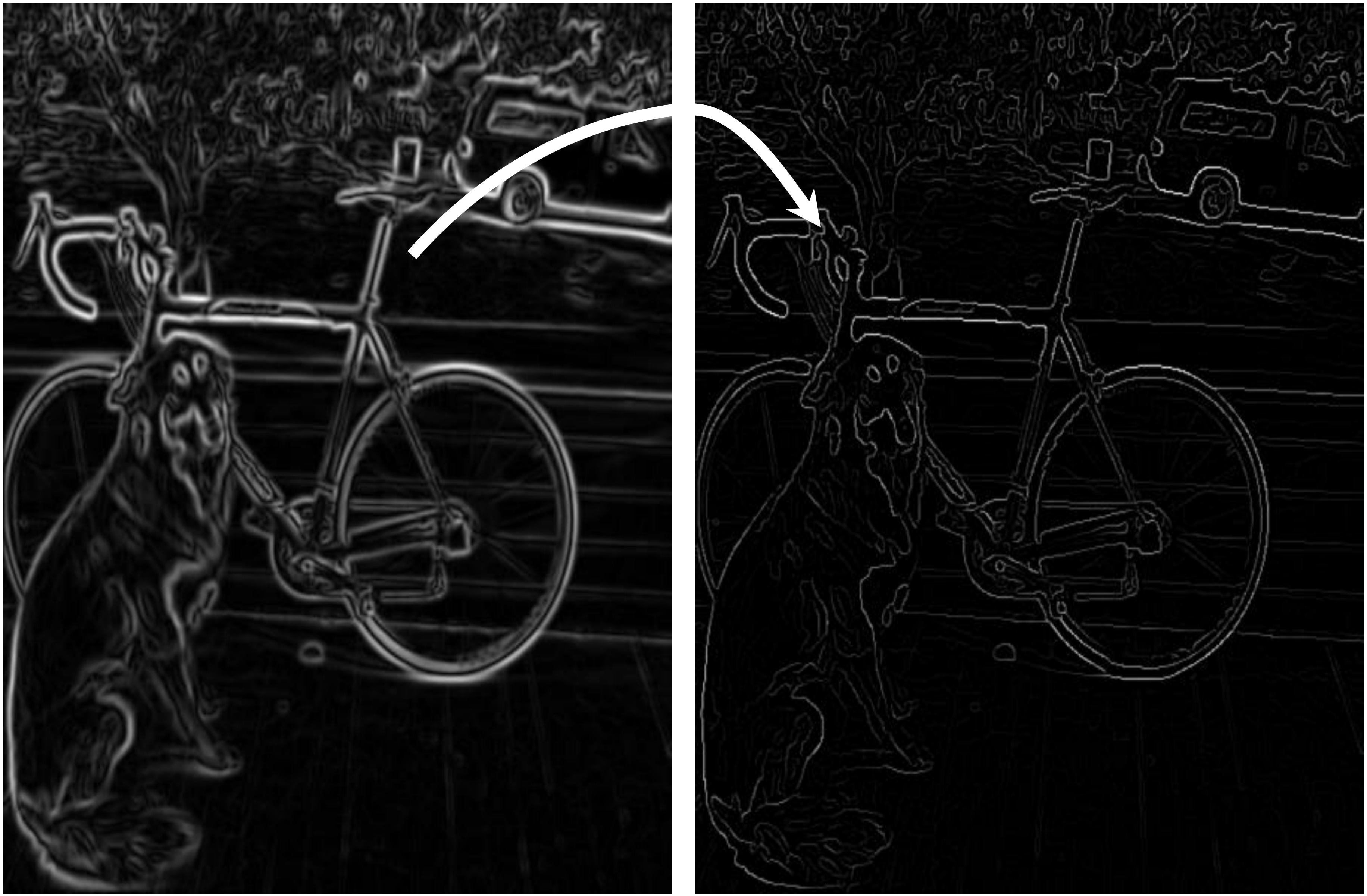
Eliminarea non-maximelor



Eliminarea non-maximelor



Eliminarea non-maximelor



De la gradienți la muchii

- după eliminarea non-maximelor mai avem zgomot în imagine

- am dori ca după aplicarea unui prag (threshold) să obținem o imagine binară numai cu muchii proeminente (tari)

- 2 praguri, 3 cazuri
  - R > T: muchie tare
  - R < T și R > t: muchie slabă
  - R < t: nu e muchie

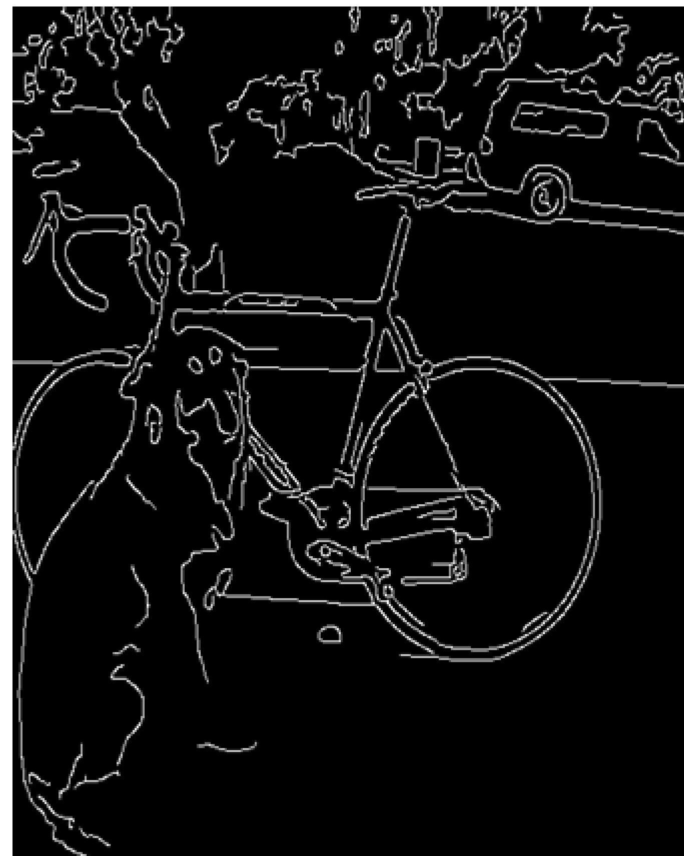
- de ce să folosim două praguri?

Conectează muhciile tari și slabe!

- muchiile tari (R > T) sunt muchii!

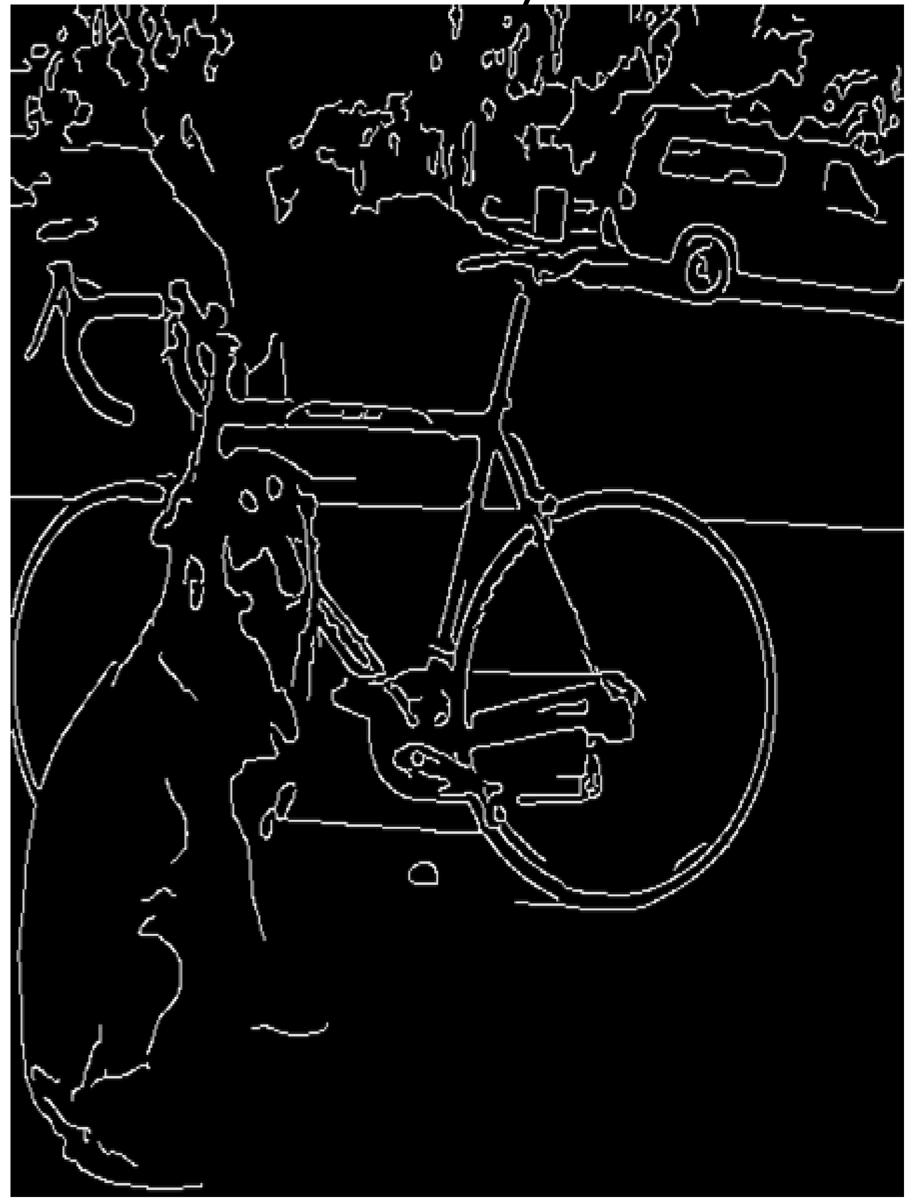
- muchiile slabe (R < T și R > t) sunt muchii dacă și numai dacă se învecinează cu muchii tari

- vecinătate dată de cei mai apropiați
8 vecini



Rezultat metoda Canny

Imagine inițială



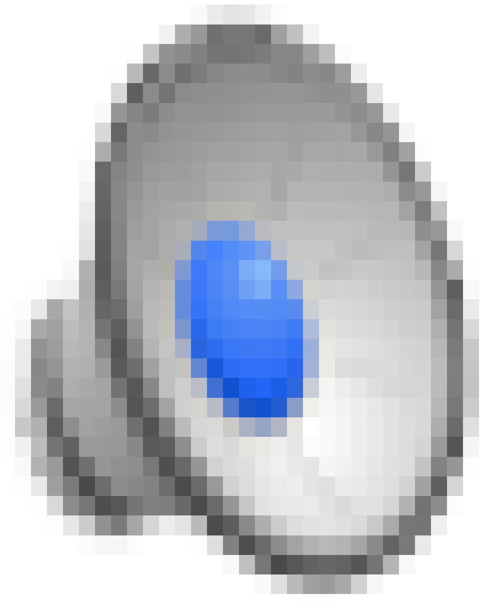
Imagine binară

Metoda Canny - demo

http://bigwww.epfl.ch/demo/ip/demos/edgeDetector/

Aplicație: detectarea linilor cu transformata Hough

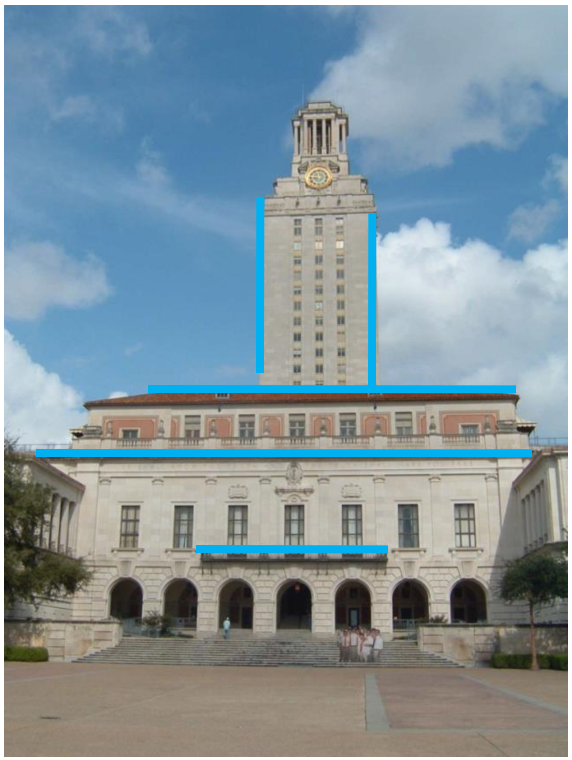
Aplicații – detectarea liniilor

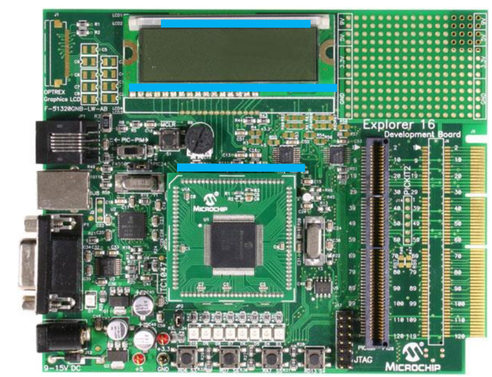


https://www.youtube.com/watch?v=SFqAAseL_1g

Aplicație: detectarea liniilor

- Multe obiecte/suprafețe sunt caracterizate de prezența unor linii drepte

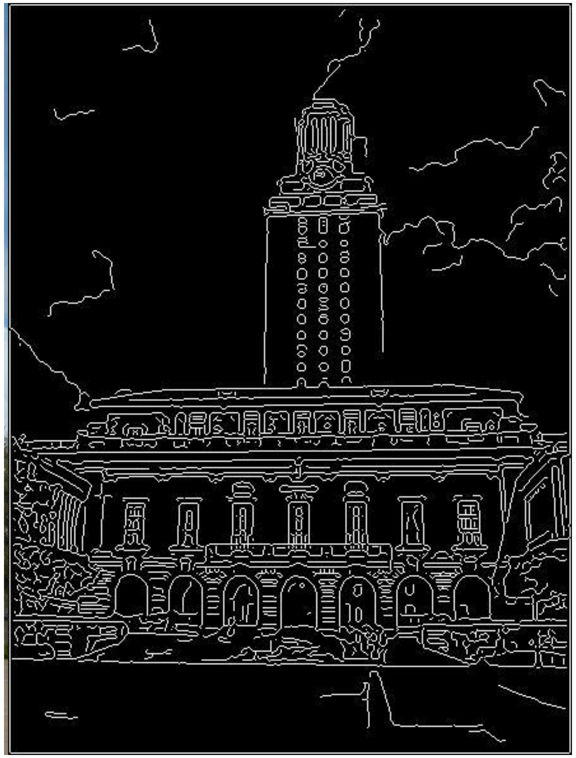




- De ce nu ne rezumăm la a rula un detector de muchii?

De ce e dificilă detectarea liniilor

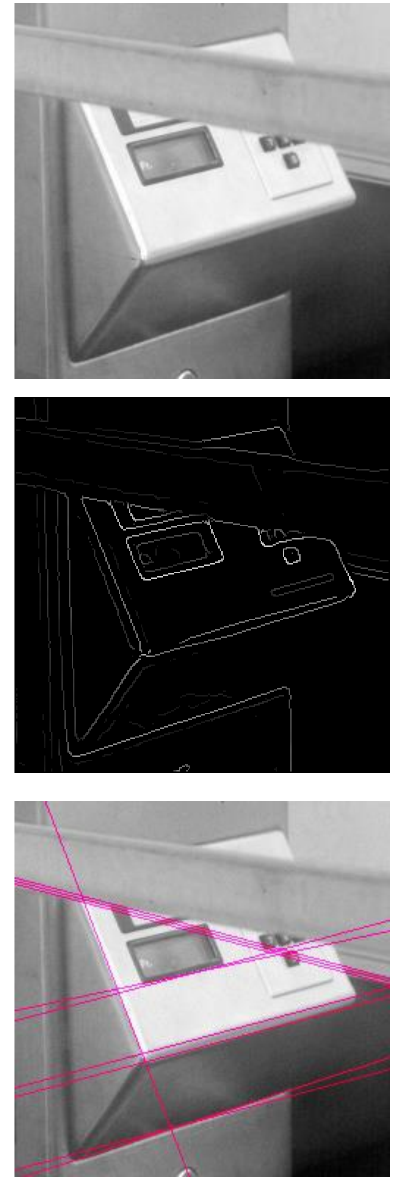
- multe puncte din background considerate ca fiind edgels
- zgomot transformat în edgels



- linii detectate parțial
- cum grupăm punctele în linii?

Slide adaptat după S. Lazebnik

Detectarea liniilor



- date fiind câteva puncte care aparțin unei linii, cum găsim linia?
- câte linii sunt în imagine?
- fiecare punct cărei linii aparține?

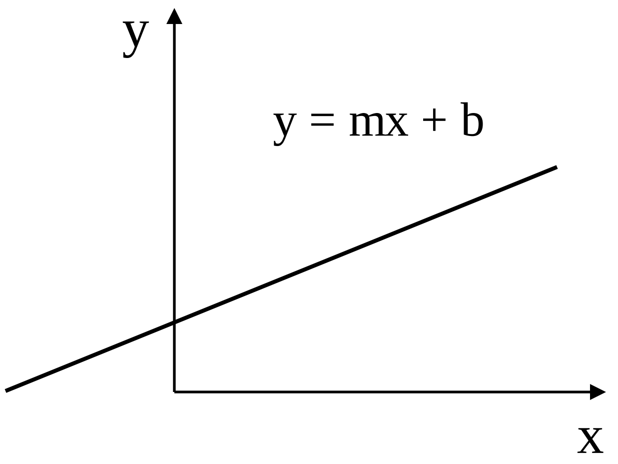
- **Transformata Hough** este o tehnică bazată pe “votare” care răspunde la aceste întrebări.

  Ideea principală:

  1. înregistrează toate liniile posibile care pot conține un punct detectat ca fiind edgel.
  2. găsește linia care primește cele mai multe voturi.

Slide adaptat după S. Lazebnik

Parametrizarea dreptelor



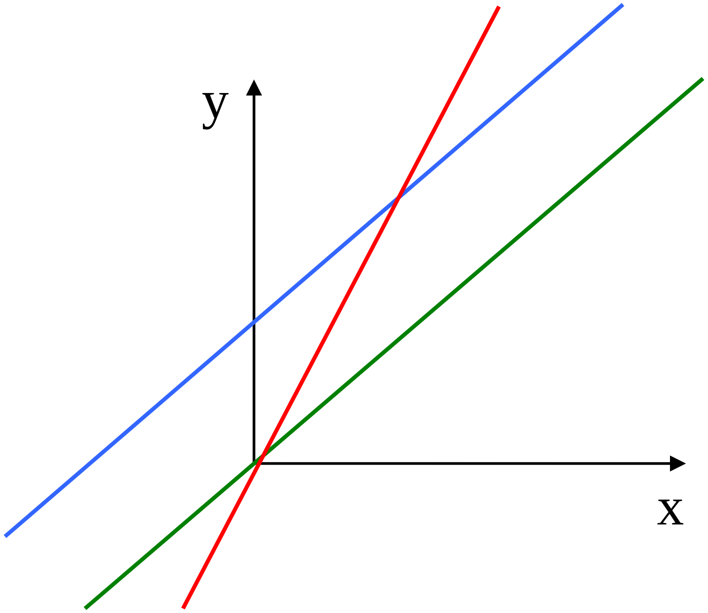
Ecuația unei drepte în planul imaginii: $y = m * x + b$,

m – panta dreptei (definește înclinarea față de axa Ox),

b – deplasarea (definește deplasarea față de originea O(0,0))

(m,b) sunt parametri dreptei

Parametrizarea dreptelor. Exemple



d_1: y = x, adică m=1, b = 0

d_2: y = x + 2, adică m=1, b = 2

d_3: y = 2x, adică m=2, b = 0

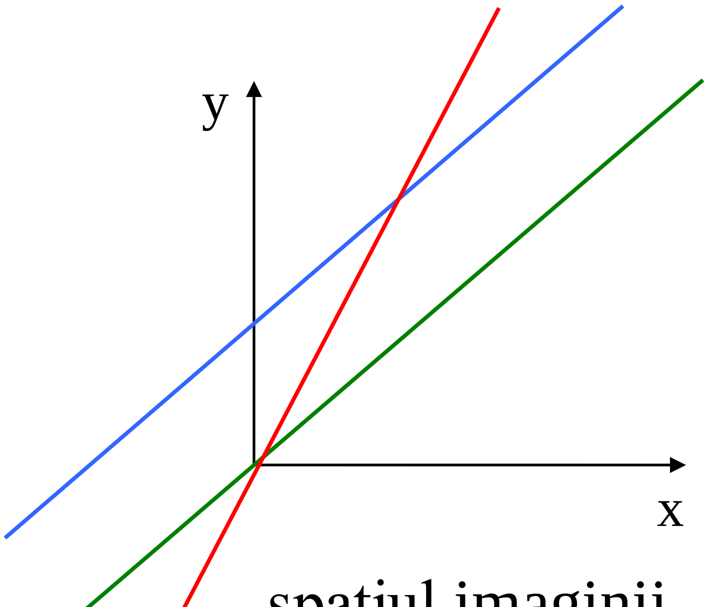
Ecuația unei drepte în planul imaginii: $y = m * x + b$,

m – panta dreptei (definește înclinarea față de axa Ox),

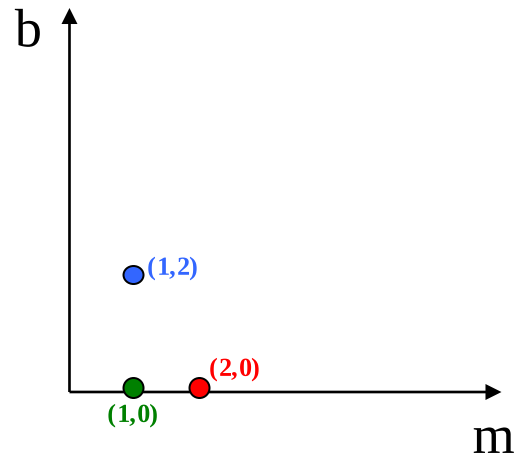
b – deplasarea (definește deplasarea față de originea O(0,0))

(m,b) sunt parametri dreptei

Spațiul Hough al parametrilor



spațiul imaginii



spațiul Hough
(al parametrilor)

d_1: y = x, adică m=1, b = 0

d_1: m=1, b = 0 -> (1,0)

d_2: y = x + 2, adică m=1, b = 2

d_2: m=1, b = 2 -> (1,2)

d_3: y = 2x, adică m=2, b = 0

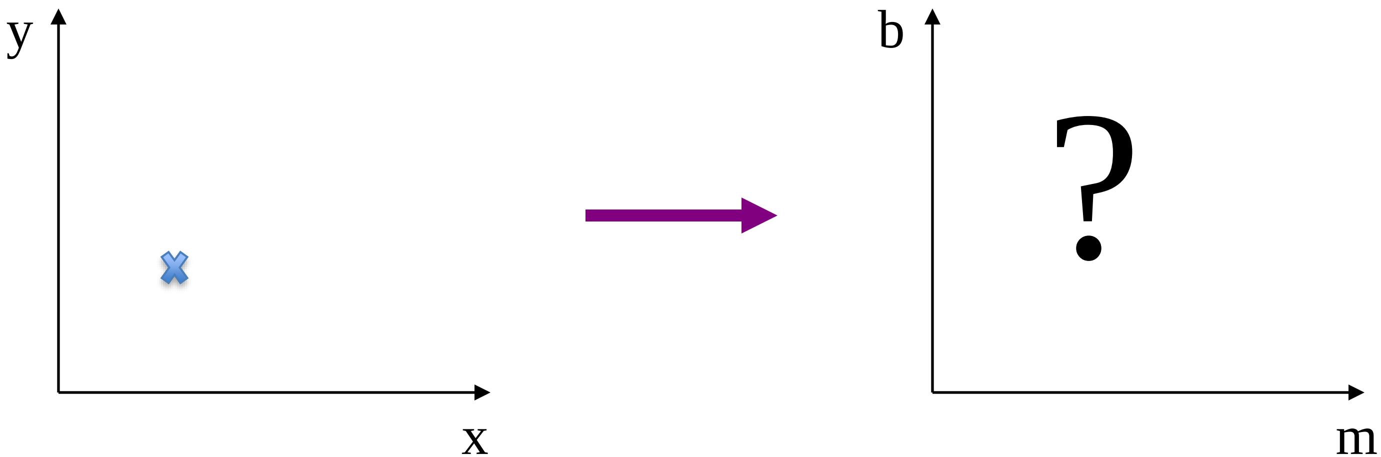
d_3: m=2, b = 0 -> (2,0)

O dreaptă într-o imagine corespunde unui punct în spațiul Hough.

Corespondența dintre cele două spații

O dreaptă într-o imagine corespunde unui punct în spațiul Hough.

Se dă un punct (x_0,y_0) în spațiul imaginii. Ce îi corespunde în spațiul Hough?



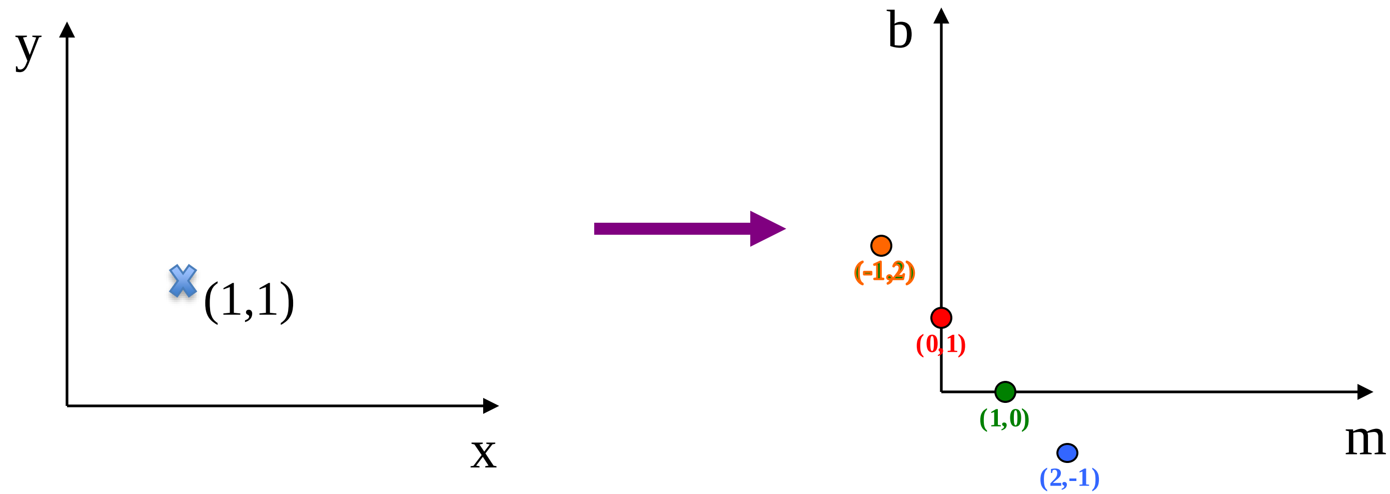
spațiul imaginii

spațiul Hough (al parametrilor)

Corespondența dintre cele două spații

O dreaptă într-o imagine corespunde unui punct în spațiul Hough.

Se dă un punct (x_0,y_0) în spațiul imaginii. Ce îi corespunde în spațiul Hough?



spațiul imaginii

spațiul Hough (al parametrilor)

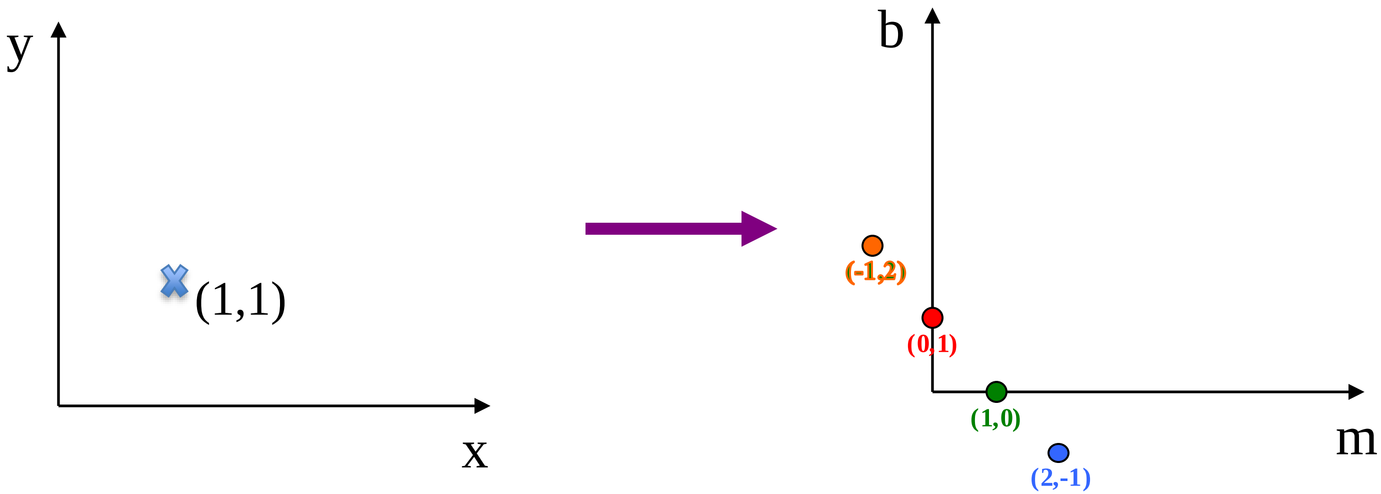
Prin punctul (1,1) trec o infinitate de drepte de forma y = mx+b:

y = x (m=1, b=0); y = 2x-1 (m=2,b=-1), y = 1 (m=0, b = 1), y = -x + 2 (m=-1, b =2),

Corespondența dintre cele două spații

O dreaptă într-o imagine corespunde unui punct în spațiul Hough.

Un punct (x_0,y_0) în spațiul imaginii corespunde unei drepte în spațiul Hough.



spațiul imaginii

spațiul Hough (al parametrilor)

Prin punctul (1,1) trec o infinitate de drepte de forma y = mx+b:

y = x (m=1, b=0); y = 2x-1 (m=2,b=-1), y = 1 (m=0, b = -1), y = -x + 2 (m=-1, b =2),

Toate cele 4 puncte (1,0), (2,-1), (0,-1), (-1,2) stau pe dreapta b = -m+1

Corespondența dintre cele două spații

O dreaptă într-o imagine corespunde unui punct în spațiul Hough.

Un punct (x_0,y_0) în spațiul imaginii corespunde unei drepte în spațiul Hough.

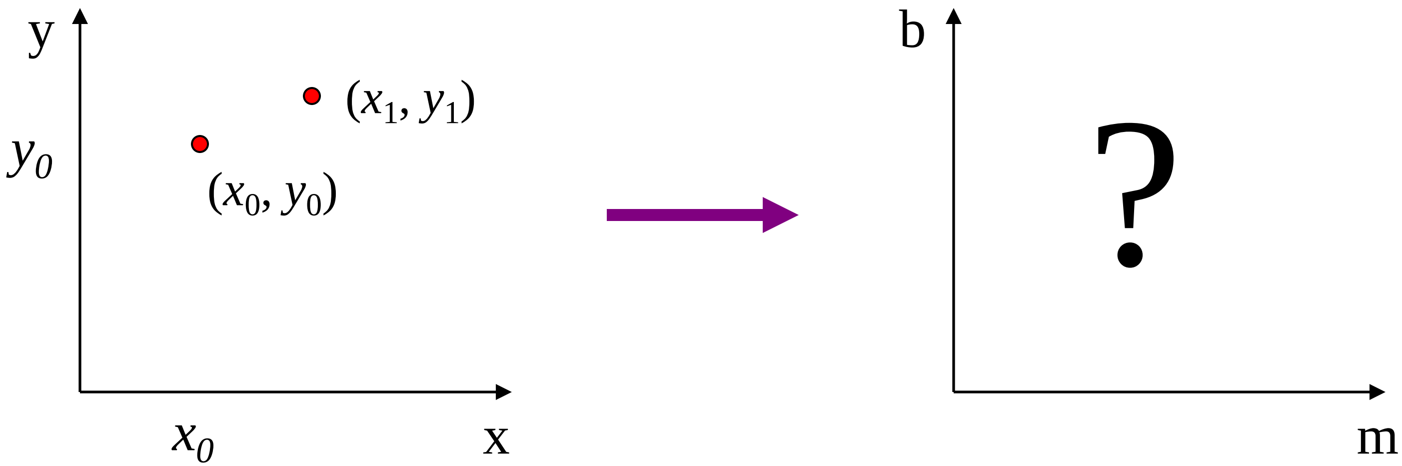
Ecuația unei drepte în planul imaginii: $y = m * x + b$,

Dreptele care trec prin punctul (x_0,y_0) pot avea orice pantă m și orice deplasare b cu condiția ca $y_0 = m * x_0 + b$.

$y_0 = m * x_0 + b \Leftrightarrow b = -x_0 * m + y_0$ (dreapta cu panta $-x_0$ și deplasare y_0)

$(x_0,y_0) = (1,1) \Rightarrow b = -m+1$

Găsirea liniilor într-o imagine folosing spațiul Hough



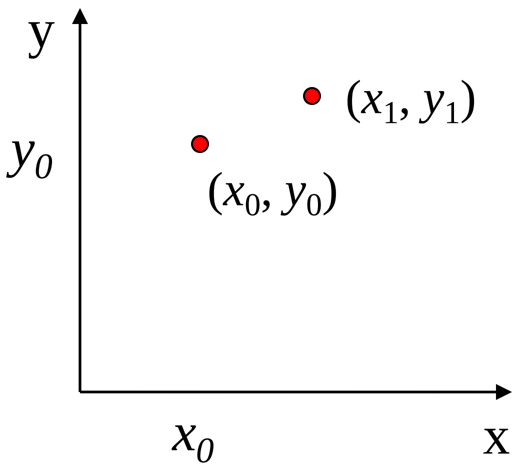
spațiul imaginii

spațiul Hough (al parametrilor)

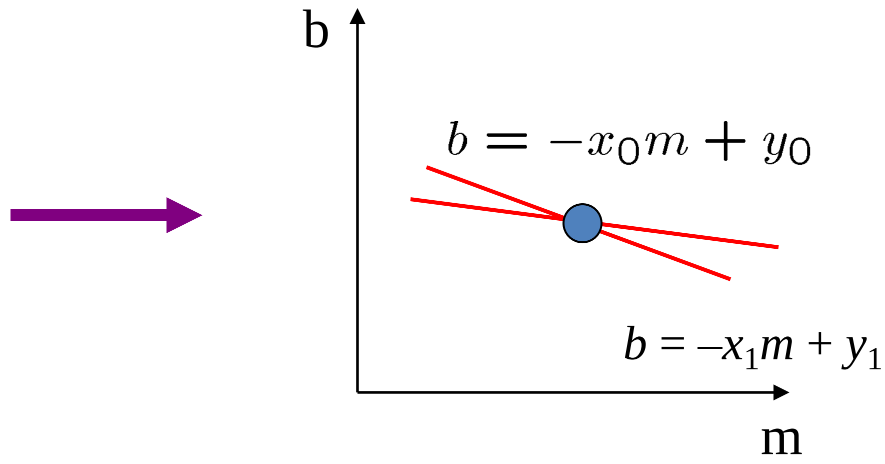
Care sunt parametri liniei care conține punctele (x_0, y_0) și (x_1, y_1)?

Slide adaptat după S. Lazebnik

Găsirea liniilor într-o imagine folosing spațiul Hough



spațiul imaginii



spațiul Hough (al parametrilor)

Care sunt parametri liniei care conține punctele (x_0, y_0) și (x_1, y_1)?

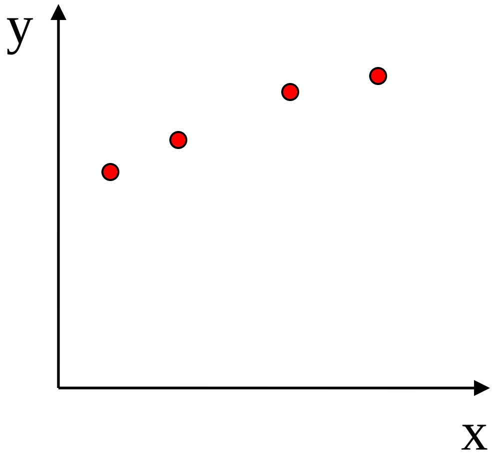
Punctului (x_0,y_0) îi corespunde dreapta de ecuație $b = -x_0m + y_0$

Punctului (x_1,y_1) îi corespunde dreapta de ecuație $b = -x_1m + y_1$

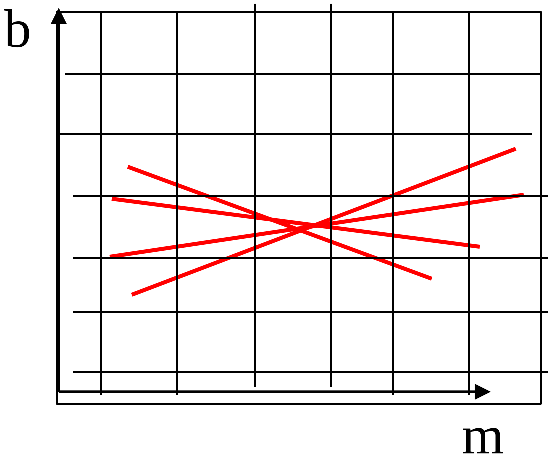
Dreptei d care conține punctele (x_0,y_0), (x_1,y_1) îi corespunde în spațiul Hough un punct. Acest punct se obține ca fiind intersecția dreptelor de ecuație $b = -x_0m + y_0$ și $b = -x_1m + y_1$

Slide adaptat după S. Lazebnik

Găsirea liniilor într-o imagine: algoritmul Hough



spațiul imaginii



spațiul Hough
(al parametrilor)

Cum putem folosi observația anterioară pentru a găsi parametri (m,b) ce definesc linia cea mai probabilă în spațiul imaginii?

- fiecare punct detectat ca fiind edgel în spațiul imaginii va vota pentru o mulțime de parametri în spațiul Hough
- acumulăm voturi în intervale discrete; parametri cu cel mai mare număr de voturi determină linia din spațul imaginii

Slide adaptat după S. Lazebnik