



Technical University of Cluj - Napoca
Computer Science Department

Interactiune Om-Calculator

Curs 4

Metode de segmentare bazate pe discontinuitati

Detectia colturilor



Metode de segmentare

Segmentare := izolare obiecte / zone de interes

⇒ detectie (cuantificare) trasaturi

⇒ clasificare/caracterizare obiecte pe baza trasaturilor

Abordari (pt. IOC)

1. Bazate pe omogenitatea (uniformitatea) pixelilor

- Segmentare bazata pe regiuni

- Clustering/grupare in spatiul trasaturilor

2. Bazate pe discontinuitatea pixelilor (intensitatii/culorii)

- Detectie de muchii

- Detectie colturi



Dectia muchiilor, conturului si a trasaturilor de forma din contur

Detectia muchiilor ([1]-cap3, [2], [3]-cap15,[4]-cap4,[5]-cap4,[6]-cap2)

- Metode bazate pe Gradientul imaginii (Roberts, Prewit, Sobel, Frei-Chen,Canny)
- Metode bazate pe Laplacianul imaginii (LoG / Mar-Hilderth)

[1] G. X.Ritter, J.N. Wilson, Handbook of computer vision algorithms in image algebra - 2nd ed, 2001 CRC Press.

[2]. J. F. Canny, Finding Edges and Lines in Images, technical Report, MIT Artificial Intelligence Laboratory, 1983.

[3] W.K. Pratt, Digital Image Processing: PIKS Inside, *Third Edition*. 2001 John Wiley & Sons, Inc.

[4] E. Trucco, A. Verri, "Introductory Techniques for 3-D Computer Vision", *Prentice Hall, 1998*.

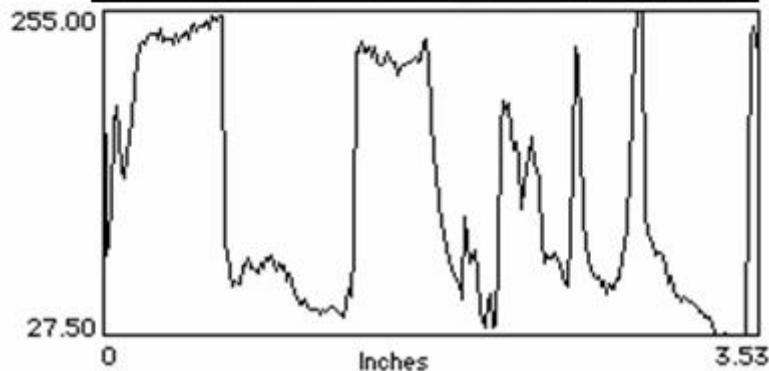
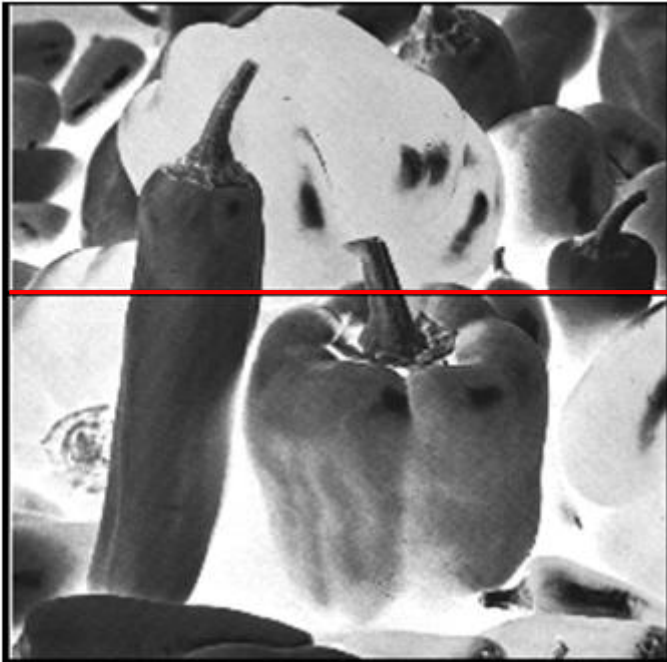
[5] M.S. Nixon, A.S. Aguado, Feature Extraction and Image Processing, Newnes, 2002

[6] D.A. Forsyth, J. Ponce, Computer Vision. A Modern Approach, *Prentice Hall, 2002*.

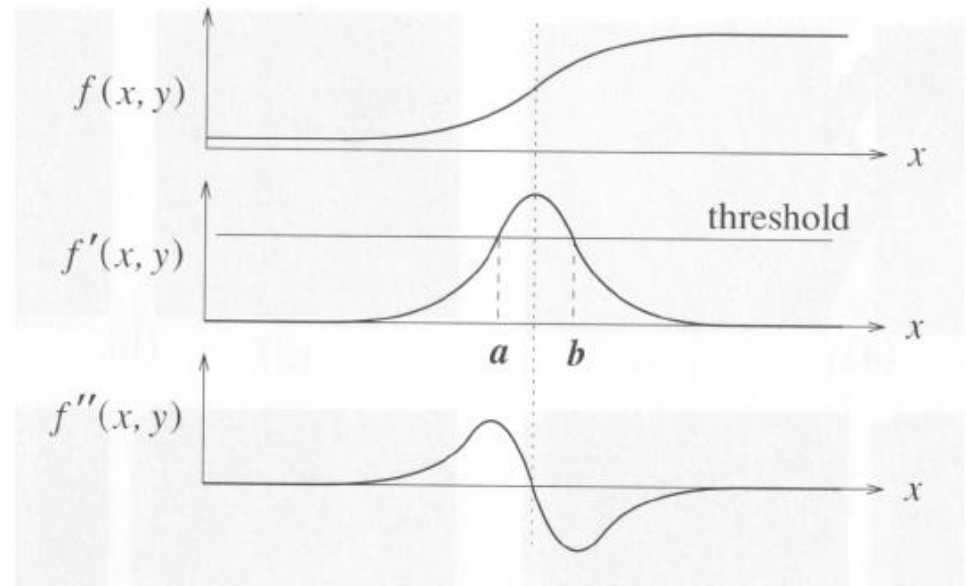


Definitie

Muchie := frontiera care separa doua regiuni de intensitate diferita
(intensitatea variaza brusc intre cele doua regiuni)

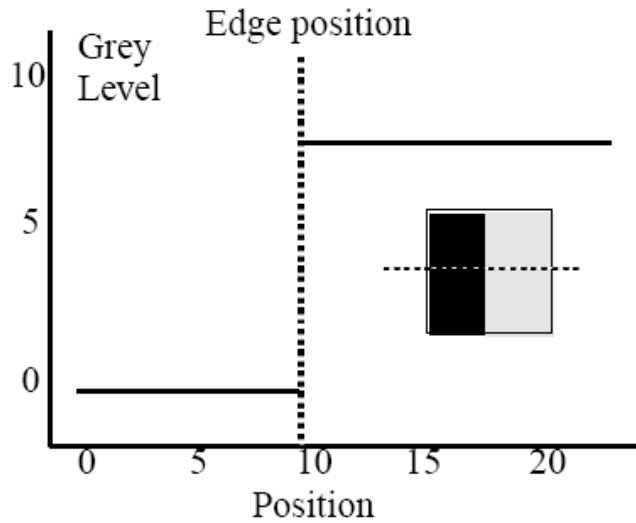


Cum putem detecta o muchie?

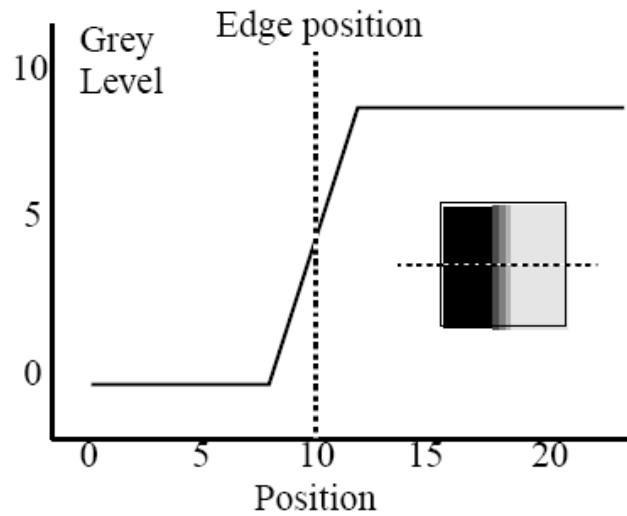




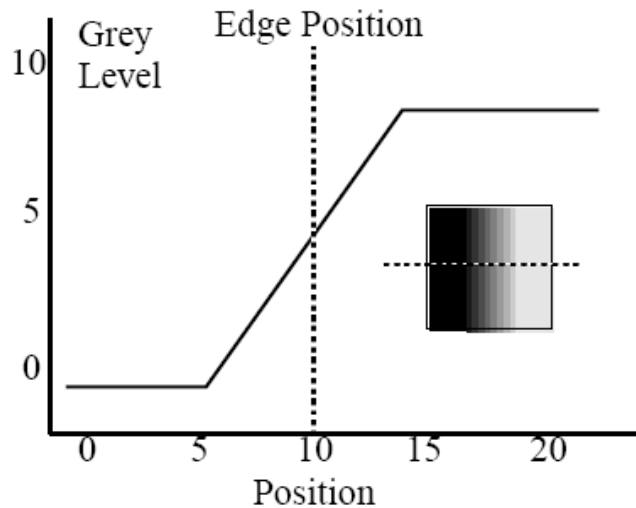
Profilul intensitatii in puntele de muchie



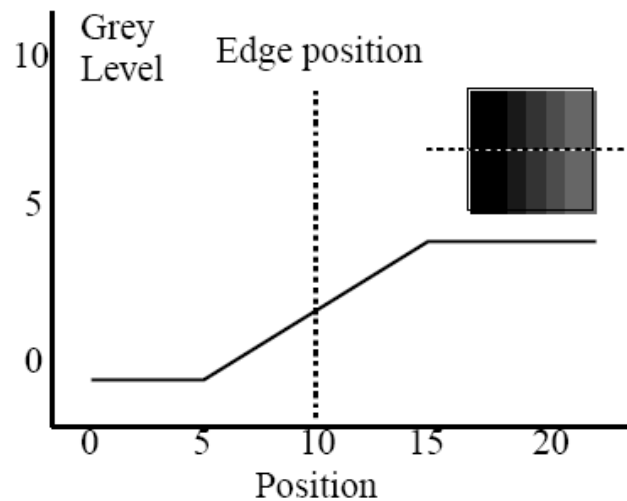
(a)



(b)



(c)



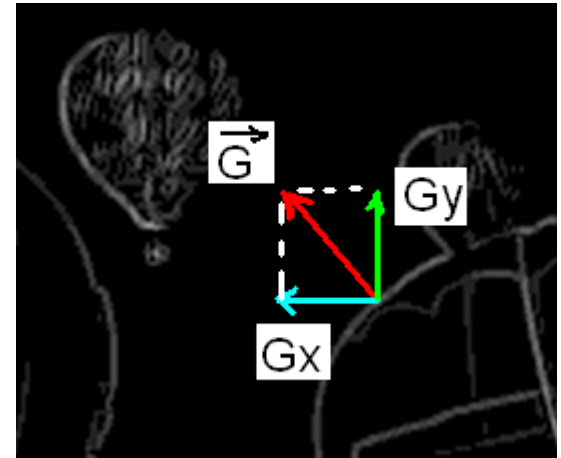
(d)



Gradientul imaginii (derivata de ordin 1)

Gradientul unei functii 2D

$$G[f(x, y)] = \begin{bmatrix} G_{f_x} \\ G_{f_y} \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$
$$= \begin{bmatrix} \lim_{\Delta x \rightarrow 0} \frac{f(x + \Delta x, y) - f(x, y)}{\Delta x} \\ \lim_{\Delta y \rightarrow 0} \frac{f(x, y + \Delta y) - f(x, y)}{\Delta y} \end{bmatrix}$$



Pentru o imagine digitala: $\Delta x = \Delta y = 1$

$$G[f[i, j]] = \begin{bmatrix} f[i + 1, j] - f[i, j] \\ f[i, j + 1] - f[i, j] \end{bmatrix} \Rightarrow$$

$$G_x = \begin{bmatrix} -1 & 1 \end{bmatrix}$$

$$G_y = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$$



Aproximari ale gradientului

Operatorul Roberts

$$G_x = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

$$G_y = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$



Operatorul Prewitt

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$

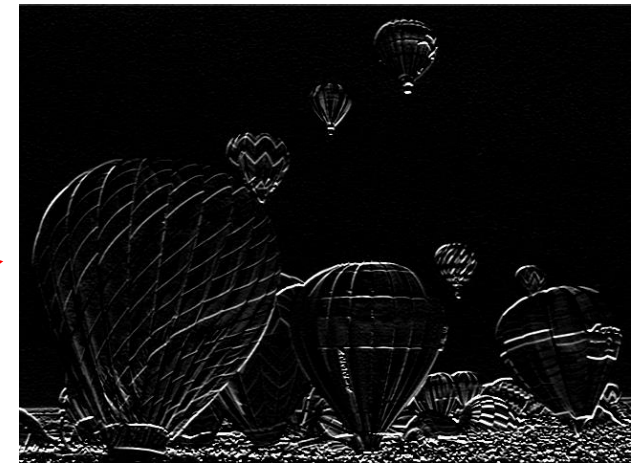
$$G_y = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$$



Operatorul Sobel

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

$$G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$





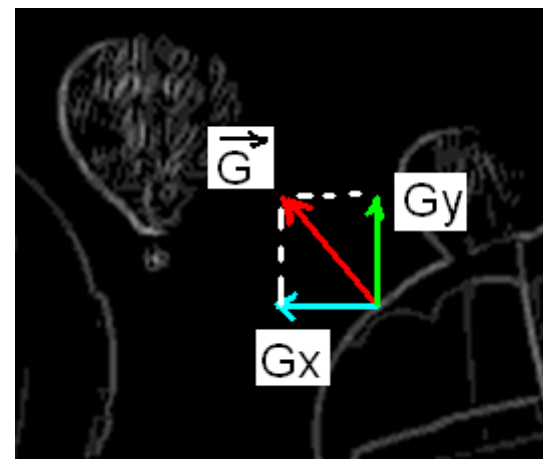
Marimi caracteristice (ale gradientului)

Magnitudinea

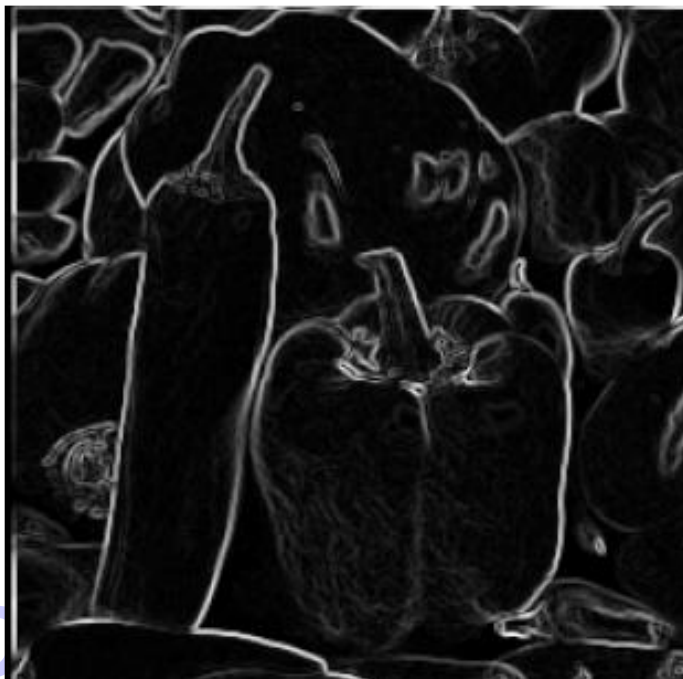
$$|G| = \sqrt{G_{fx}^2 + G_{fy}^2}$$

Directia

$$dir = \arctg\left(\frac{G_{fy}}{G_{fx}}\right)$$



Imaginea $|G|$

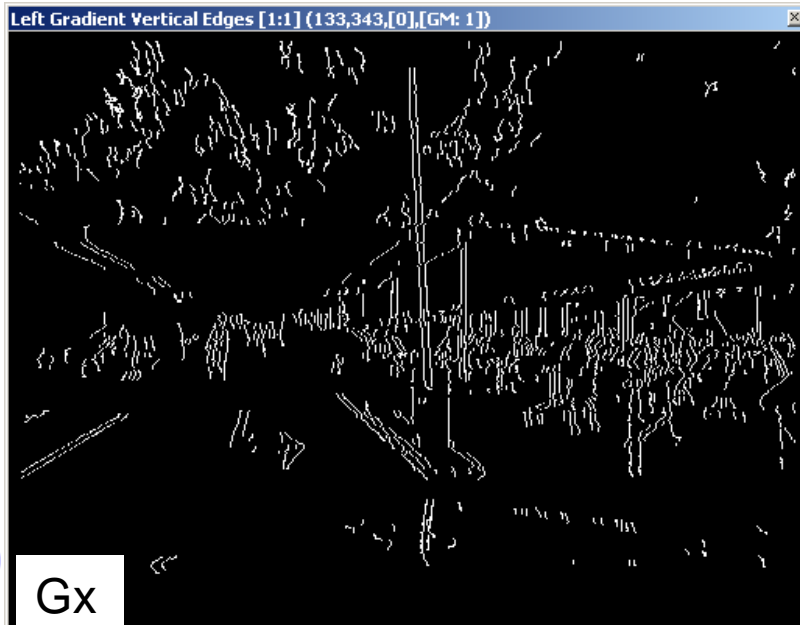
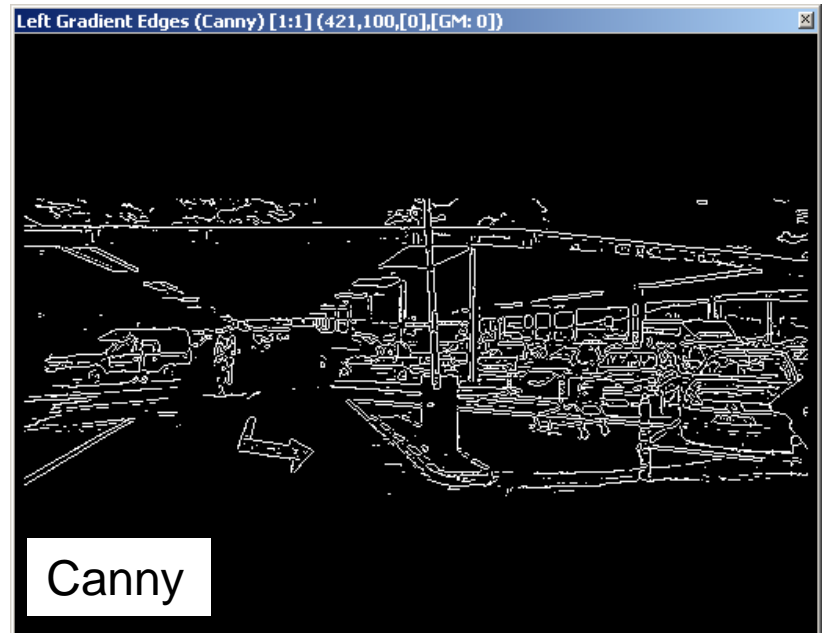


Binarizare
prag T





Exemple





Metoda Canny de detectie a muchilor

Caracteristici

- Maximizarea raportului semnal zgomot pt. o detectie corecta.
- Localizare buna la marcarea punctelor de muchie
- Minimizarea nr. de raspunsuri pozitive la o singura muchie (eliminarea non-muchiilor)

Algoritm

1. Filtrare zgomot cu un filtru Gaussian.
2. Calculul modulului si directiei gradientului (ex. Sobel)
3. Supresia non-maximelor
4. Trhresholding cu histereza



Detectia colturilor

Colt (corner) :=intersectia a cel puțin 2 muchii (de orientari diferite)

- Un punct in a carui vecinatate exista (cel puțin) 2 directii dominante (diferite) ale muchiilor (gradientului)

Punct de interes (interest point) \Rightarrow pozitie bine definita in imagine si detectat in mod robust: colt, minim sau maxim local de intensitate, terminatie a unei linii, punct de pe un contur in care curbura are un maxim local.

Pentru ca un punct de interes sa fie colt \Rightarrow **analiza si validari suplimentare**

Metoda de detectie de colturi \approx robusta \Rightarrow permite detectia aceluiasi set de colturi in conditii de iluminare diferita, sau daca imaginea obiectului sufera anumite transformari geometrice (rotatie, scalare, transformare de perspectiva)

Metode (http://en.wikipedia.org/wiki/Corner_detection):

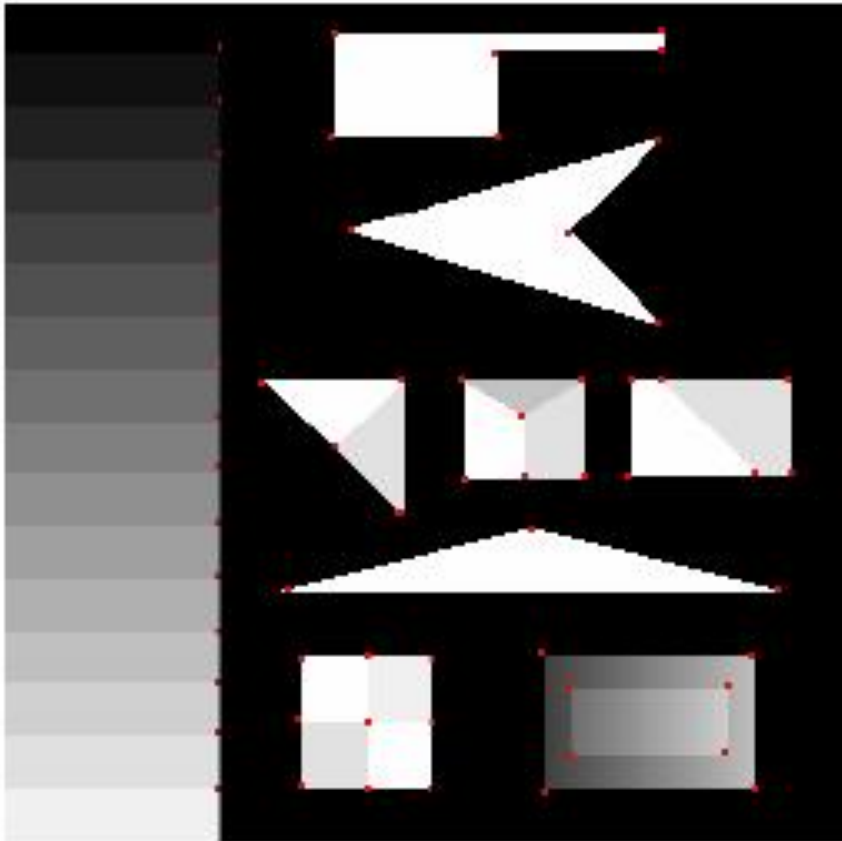
Moravec \Rightarrow Harris & Stephens \Rightarrow Shi/Kandc & Tomasi (OpenCV GoddfeaturesToTrack)

Wang&Brady, SUSAN, Trajkovic and Hedley



Detectia colturilor

Exemple:



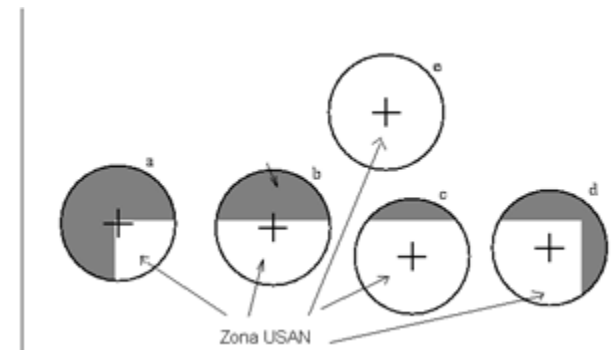
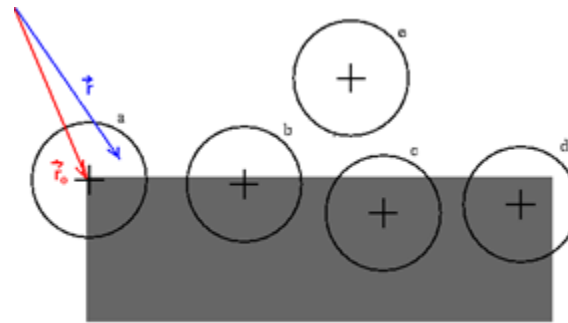


Detectia colturilor

Metoda SUSAN (Smallest Univalued Segment Assimilating Nucleus) [1]

- Număr minim de raportări false
- Localizare precisă
- Independență față de variații de luminozitate
- Număr redus de parametri pentru configurare și posibilitatea stabilirii automate a valorilor parametrilor
- Performanțe bune în prezența zgomotelor

$$\text{Kernel} = \begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & X & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix}$$



Pentru fiecare pixel r se calculează:

$$\text{diff} = |I(r) * \text{Kernel}(k, l) - I(r_0)|$$

$$k = -3 \dots 3$$

$$l = -3 \dots 3$$

$$(1) \quad c(\vec{r}, \vec{r}_0) = \begin{cases} 1, & \text{diff} < \theta \quad (\in \text{USAN}) \\ 0, & \text{altfel} \quad (\notin \text{USAN}) \end{cases}$$

$$(2) \quad c(\vec{r}, \vec{r}_0) = e^{-\left(\frac{I(\vec{r}) - I(\vec{r}_0)}{\theta}\right)^6}$$



Metoda SUSAN

Pentru fiecare poziție curentă a nucleului se calculează un număr care este de fapt **aria zonei USAN**:

$$n(\vec{r}_0) = \sum_{r \in \text{Kernel}} c(\vec{r}, \vec{r}_0)$$

Răspunsul inițial al detectorului de colțuri SUSAN in fiecare punct din imagine $= (i, j)$ se calculează după formula de mai jos:

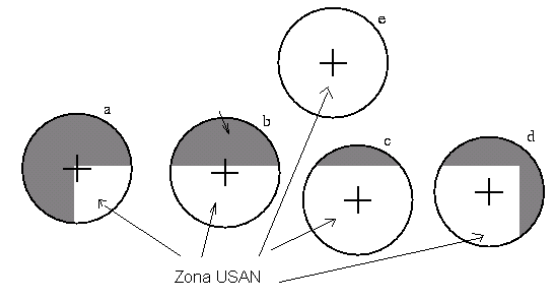
$$R(\vec{r}_0) = g - n(\vec{r}_0)$$

unde: g - prag geometric setat la valoarea $n_{\max} / 2$ (n_{\max} – aria nucleului USAN)

Colt: $R > 0$ (aria zonei USAN $< \frac{1}{2}$ aria nucleului)

Muchie: $R \approx 0$

Pixel intr-o zona uniforma: $R < 0$



Valoarea g & (1) – prag *calitativ* - forma coltului (pt. colturi mai ascutite se micsoreaza g)

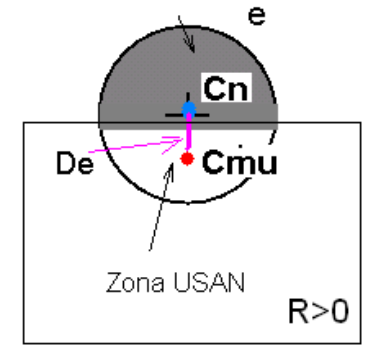
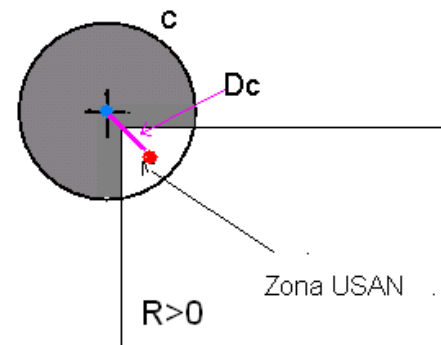
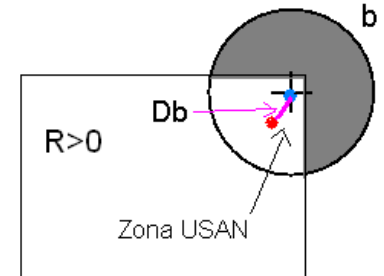
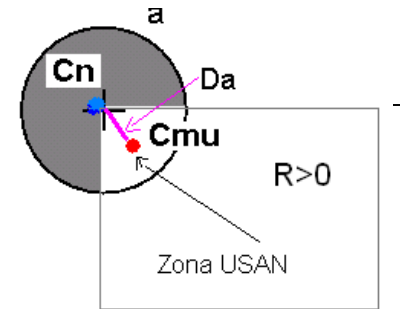
Valoare g & (2) – prag de intensitate (*cantitativ*) \Rightarrow nr. de colturi raportate



Metoda SUSAN

Filtrarea raspunsului initial R

1. Calcularea centrului masa al USAN-ului (C_{mu}) si compararea distantei D de la acesta la centrul nucleului (C_n). Pentru un colț real, distanta D este maxima (a) pe când pentru segmente D care sunt continute in zona USAN (b), centrul de greutate este mai aproape de cevtrul nucleului.
2. Continuitatea USAN-ului. Toți pixelii de pe segmentul D , trebuie sa facă parte din USAN pentru ca un colt detectat sa fie valid: (a), (b). Aceasta metoda forțează continuitatea USAN-ului și reduce din nou numărul de colturi fals raportate.



Cazurile (c) si (e) sunt invalide.

Non-maxima supression

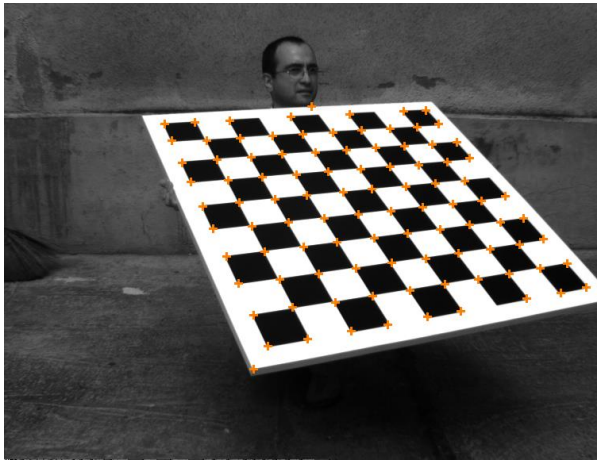
Căutarea în R folosind o matrice pătratică $n \times n$ (ex. $n=3, 5, ..$) a valorilor maxime locale (non-maxima supression) si raportarea colțurilor



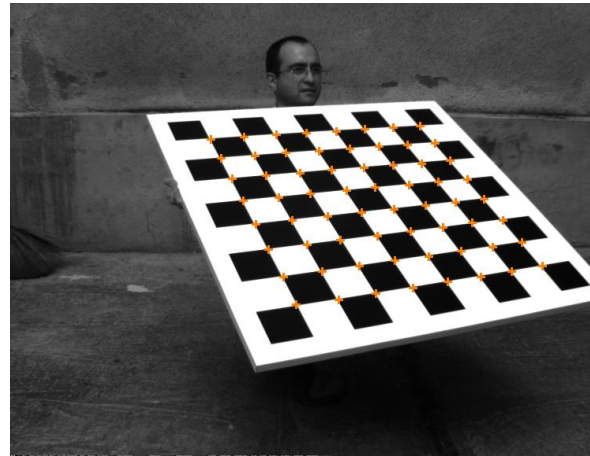
Metoda SUSAN

Pașii algoritmului USAN (pentru fiecare punct din imagine):

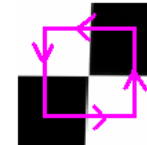
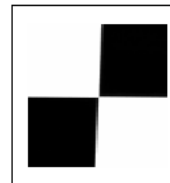
1. Plasarea unei măști circulare in jurul pixelului studiat.
2. Calcularea numărului de pixeli din USAN (n).
3. Calculul răspunsului inițial (R) prin scăderea din pragul geometric a valorii USAN-ului.
4. Eliminarea raportărilor false de colțuri prin folosirea metodelor ce implica centrul de greutate si continuitatea USAN-ului.
5. Aplicare non-maxima suppression \Rightarrow colturi finale



SUSAN



SUSAN + Rafinare bazata pe model





Metoda Harris

Variatiei intensitatii in pct (x,y) pentru o frereastră w shiftata cu deplasamentul (u,v):

$$E(u, v) = \sum_{x,y} w(x, y) [I(x + u, y + v) - I(x, y)]^2$$

Colturi \Rightarrow puncte cu $E(uv)$ maxim

Aproximare prin serii Taylor:

$$E(u, v) \approx \sum_{x,y} [I(x, y) + uI_x + vI_y - I(x, y)]^2$$

$$E(u, v) \approx \sum_{x,y} u^2 I_x^2 + 2uv I_x I_y + v^2 I_y^2$$

$$E(u, v) \approx [u \quad v] \left(\sum_{x,y} w(x, y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \right) \begin{bmatrix} u \\ v \end{bmatrix}$$

$$M = \sum_{x,y} w(x, y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}$$

w poate fi o ferestra cu ponderi gaussiene !



Metoda Harris

$M = \begin{bmatrix} A & C \\ C & B \end{bmatrix}$ **Matricea de autocorelatie (covarianta a derivatelor)** \Rightarrow contine totii operatorii diferentiali care descriu geometria “suprafetei” imaginii in punctul (x,y) (dpdv al variatiei de intensitate)

Matricea de autocorelatie (simetrica) poate fi diagonalizata prin rotatia axelor de coordonate [<http://civile.utcb.ro/cmat/cursrt/an04.pdf>]

$$\Rightarrow \begin{bmatrix} \lambda_{\max} & 0 \\ 0 & \lambda_{\min} \end{bmatrix}, \quad \lambda_{\max} \geq \lambda_{\min} \geq 0$$

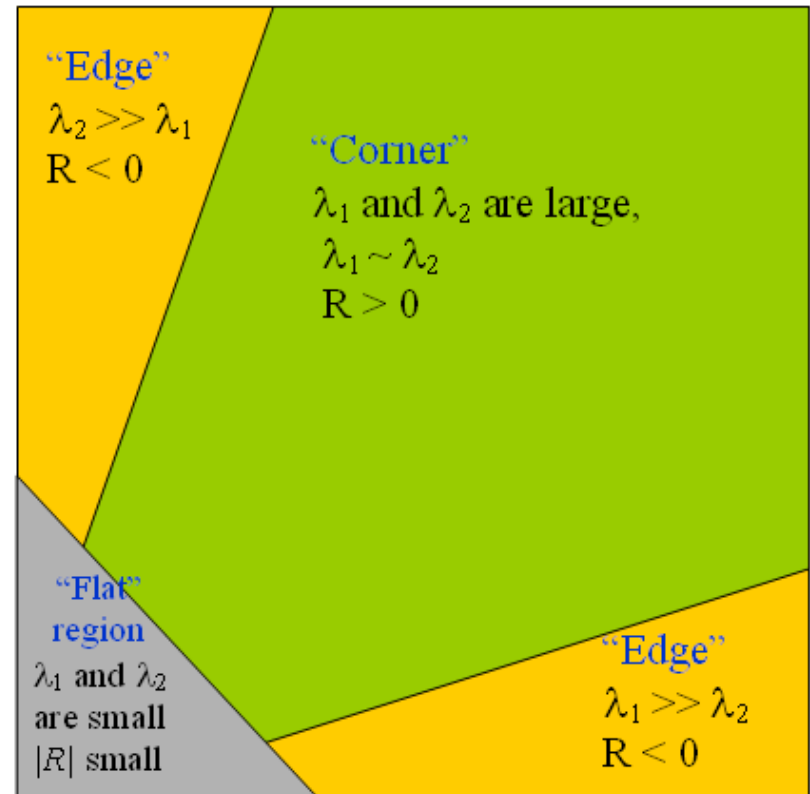
Valorile proprii ale lui M: $\lambda_1, \lambda_2 \Rightarrow$ functia de raspuns $R(x,y)$ (masura a “tariei” coltului in punctul P(x,y))

$$\det(M) = \lambda_1 \lambda_2 = AB - C^2$$

$$\text{trace}(M) = \lambda_1 + \lambda_2 = A + B$$

$$R(x, y) = \det(M) - k(\text{trace}(M))^2$$

$$k = 0.04 \dots 0.15$$





Algoritmul Harris:

1. Pt. fiecare pixel din imagine $P(x, y)$ se calculeaza matricea de autocorelatie M .
2. Se calculeaza "harta" (matricea) functiei de raspuns $R(x, y)$ (in fiecare pixel $P(x, y)$).
3. Se filtreaza $R(x, y)$ prin thresholding (ex: dc. $R < T \Rightarrow R=0$).
4. Se aplica "non-maximum suppression" \Rightarrow se retin maximele locale (restul se elimina: $R=0$).
5. Toate punctele ramase ($R > 0$) vor fi colturile rezultate.
6. Optional se poate limita numarul maxim de colturi raportate.

[2] A. Koschan, M. Abidi, Digital Color Image Processing, Wiley & Sons, 2008. - cap 6, pag 143 -144

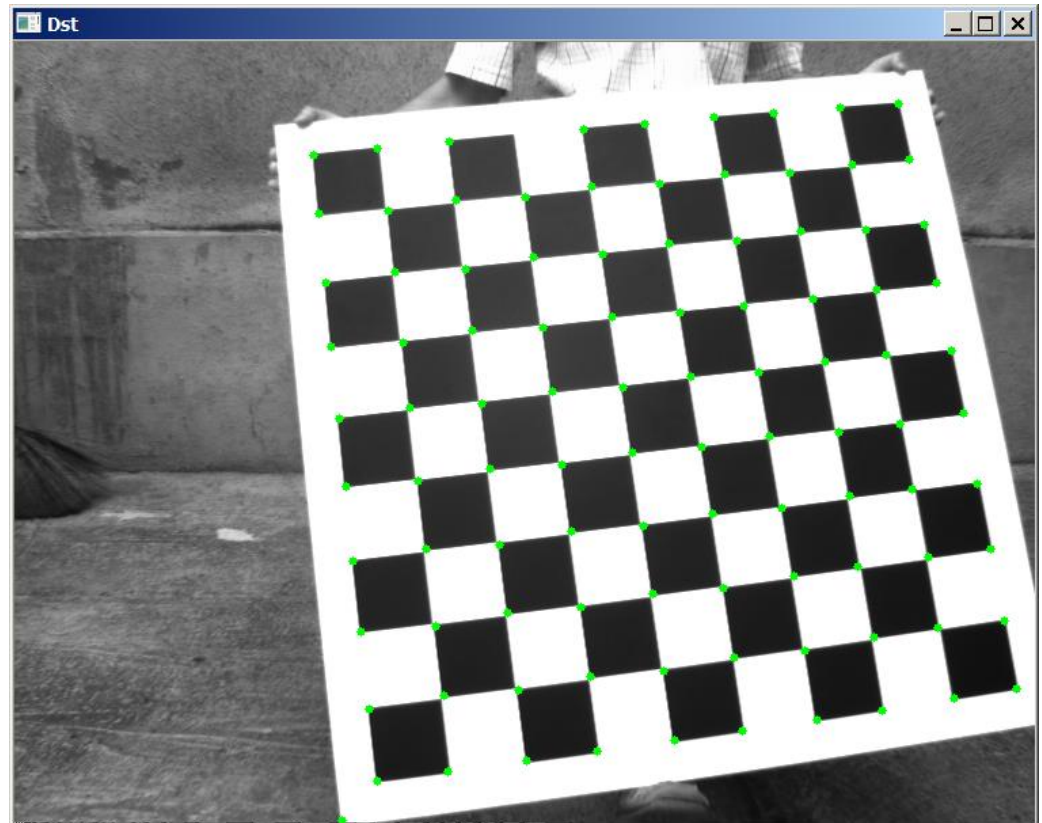
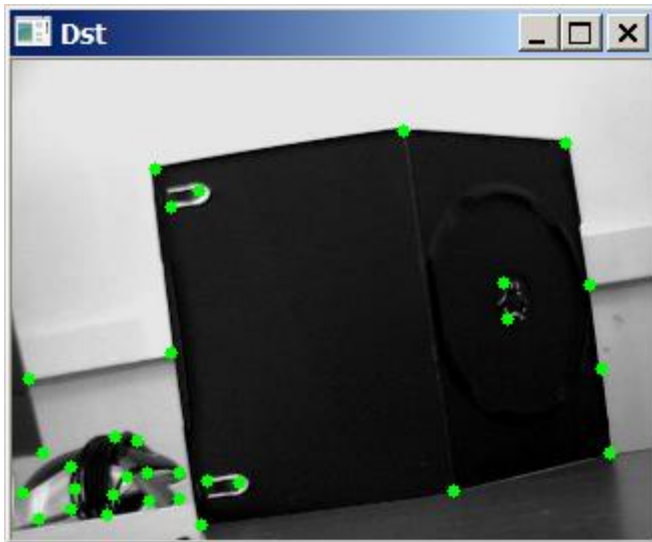


Metoda Harris

Rezultate: metoda Harris din openCV

goodFeaturesToTrack (... , useHarrisDetector = true, ...)

```
int maxCorners = 100;  
minDistance = 10;  
int blockSize = 3;  
double k = 0.04;
```





Metoda Harris

Algoritmul Harris (color) [2]:

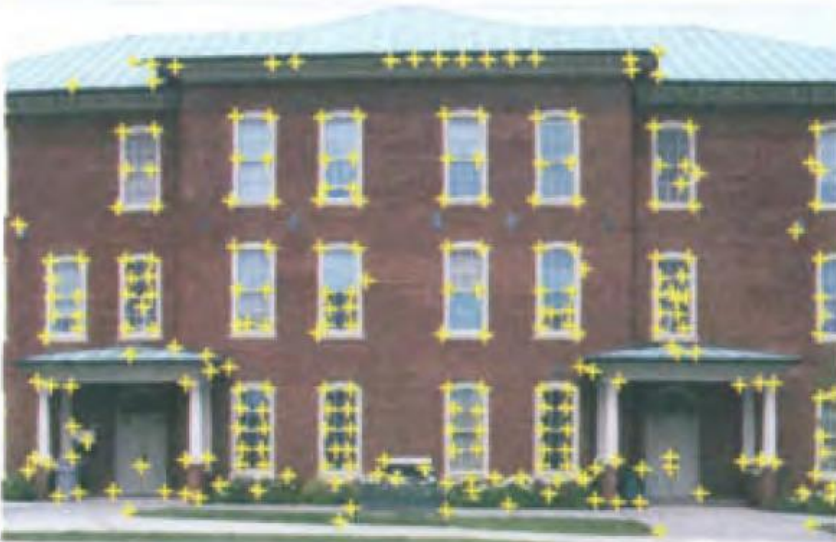
1'. Calculul matricei de autocorelatie (M) pt. imagini color:

$$M'_{11} = S_{\sigma}(R_x^2 + G_x^2 + B_x^2),$$

$$\mathbf{M}' = \begin{bmatrix} M'_{11} & M'_{12} \\ M'_{21} & M'_{22} \end{bmatrix}$$

$$M'_{22} = S_{\sigma}(R_y^2 + G_y^2 + B_y^2), \text{ and}$$

$$M'_{12} = M'_{21} = S_{\sigma}(R_x R_y + G_x G_y + B_x B_y)$$





Metoda KLT (Kanade-Lucas-Tomasi)

Algoritmul KLT [3]

Parametrii de intrare: λ_{th} , d – dim ferestrei w

1. Calculeaza I_x si I_y (derivata I) pt. fiecare pixel $p(x,y)$ din imagine.
2. Pt fiecare pixel p din imagine:
 - calculeaza matricea M peste vecinatatea $w(dx,dy)$
 - calculeaza λ_{min} : daca $\lambda_{min} > \lambda_{th}$ salveaza p in lista L
3. Sorteaza L in ordine descrescatoare.
4. Pt. fiecare punct p' din lista L realizeaza non-maxima supresion:
 - daca exista alte puncte in lista cu $\lambda > \lambda_{p'}$, si care sunt vecine spatial cu p' , atunci p' este eliminat.

λ_{th} , poate fi estimat din histograma construita pe toate valorile lui λ_{min}

O variatie a algoritmului este metoda *goodFeaturesToTrack* din OpenCV



Detectia colturilor

Aplicatii

Detectia de *puncte de interes* pentru:

- Detectia miscarii
- Tracking (trasaturi si obiecte)
- Image registration
- Adnotare semantica/ regasire (retrieval) automata de imagini
- Asamblare imagini panoramice (din parti)
- Modelare 3D
- Recunoastere de obiecte

Pentru aplicatii complexe (ex. image registration, adnotare semantica) se asociaza colturilor trasaturi invariante (SIFT, SURF etc.)



Referinte

- [1] S.M. Smith. *SUSAN - a new approach to low level image processing*. Internal Technical Report TR95SMS1, Defence Research Agency, Chobham Lane, Chertsey, Surrey, UK, 1995.
- [2] A. Koschan, M. Abidi, *Digital Color Image Processing*, Wiley & Sons, 2008. - cap 6, pag 143 -144
- [3] J. Shi and C. Tomasi. *Good features to track*. In Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR94), Seattle, June 1994.