Vedere Artificială
(Computer Vision)

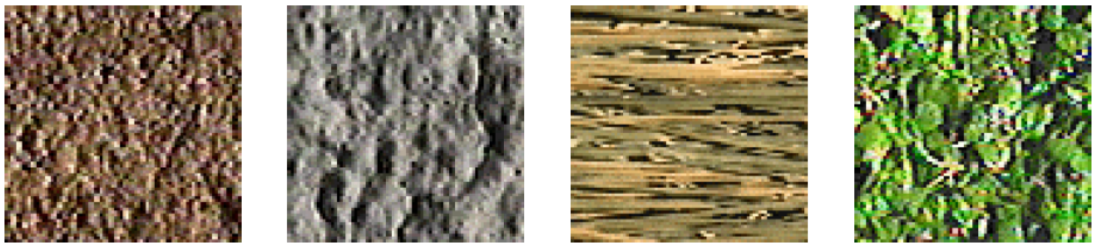
Bogdan Alexe

bogdan.alexe@fmi.unibuc.ro

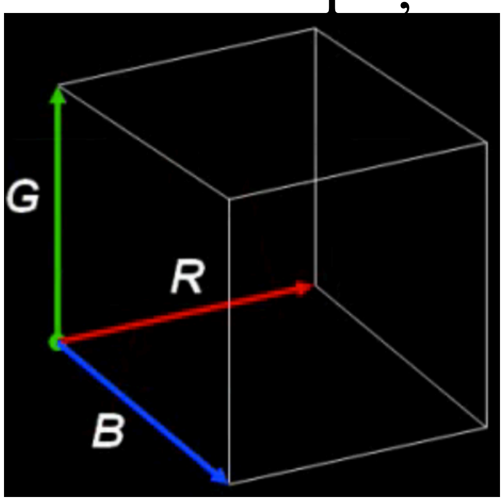
anul 2, master Informatică, semestrul I, 2019-2020

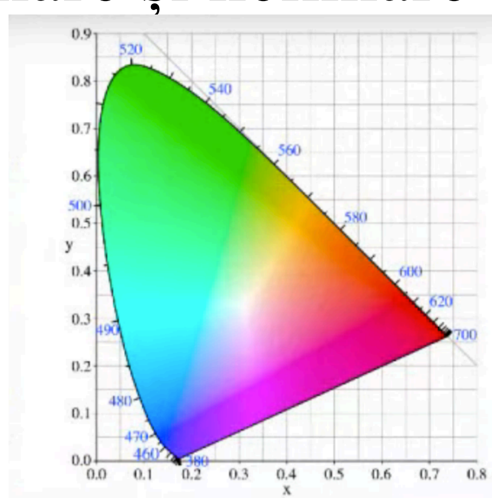
Recapitulare – cursul trecut

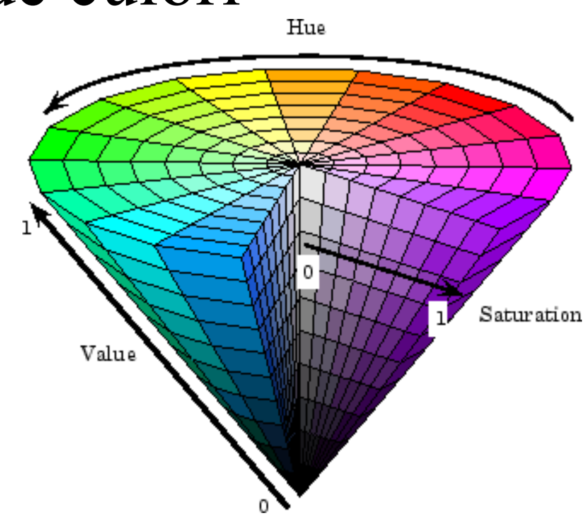
1. Textură: sinteză, transfer



2. Culoare – spații liniare și neliniare de culori





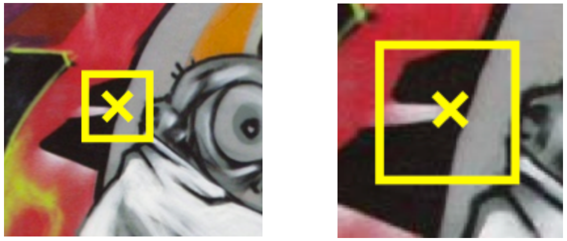


Planul cursului de azi

- trăsături locale: detectare + descriere

- detectorul Harris

- detectorul Harris Laplace



Trăsături locale

Trăsături locale: terminologie

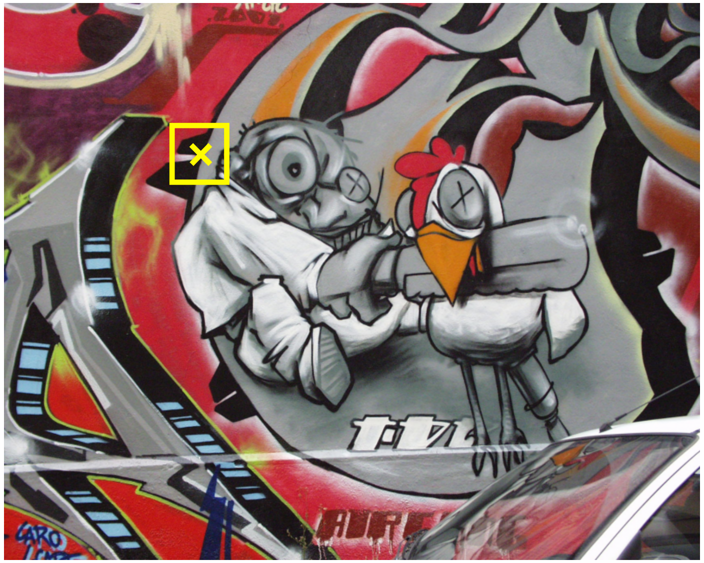
- în limba engleză: local feature = interest point + region descriptor = keypoint + region descriptor

- în limba română: trăsătură locală = caracteristică locală = regiune din imagine centrată într-un punct de interes (punct cheie) + descriptorul conținutului vizual al regiunii din imagine;

- trăsătură locală = o regiune din imagine cu proprietăți speciale, în sensul că poate fi detectată dacă imaginea sufera transformări geometrice sau fotometrice (nu foarte mari)

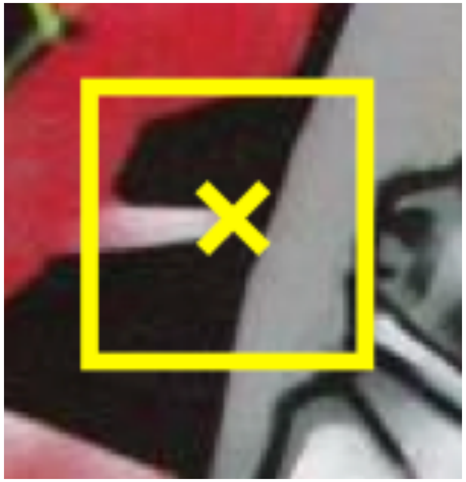
Trăsături locale

- trăsătură locală = o regiune din imagine cu proprietăți speciale, în sensul că poate fi detectată dacă imaginea sufera transformări geometrice sau fotometrice (nu foarte mari)

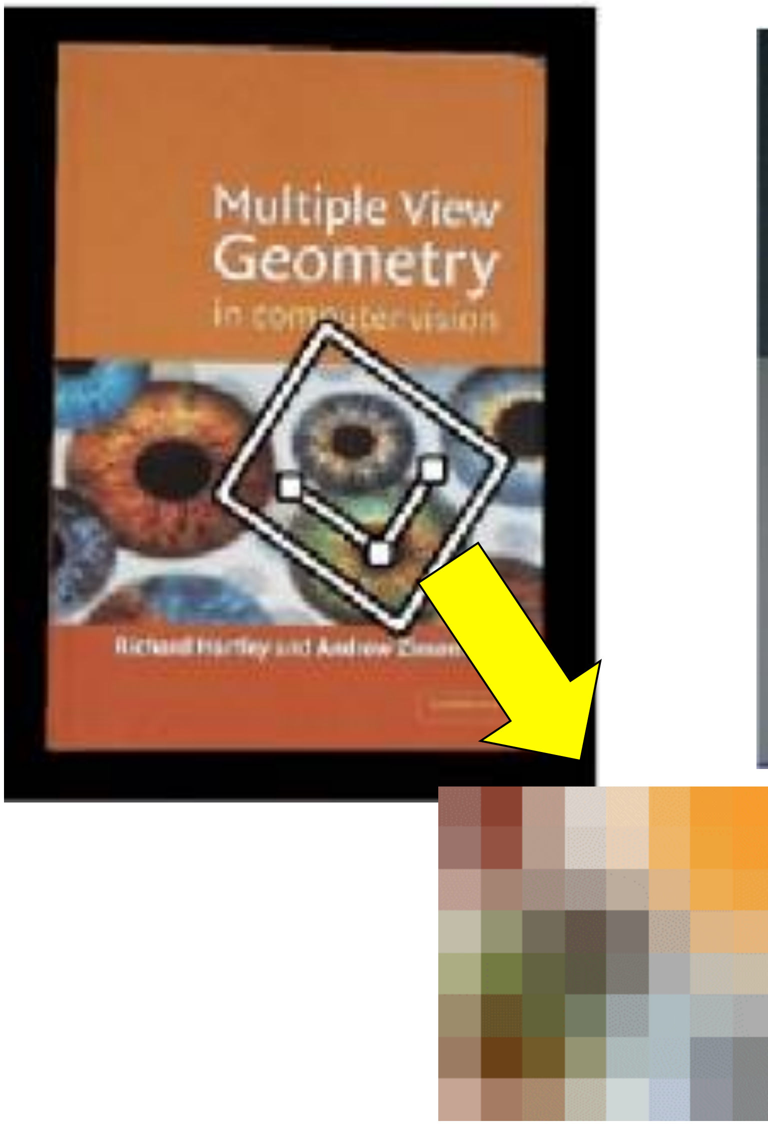


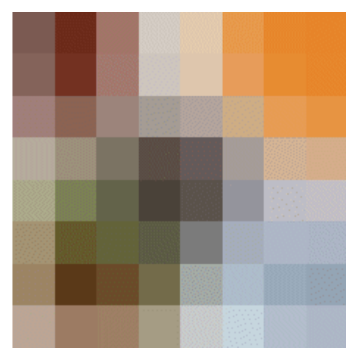
Trăsături locale

- trăsătură locală = o regiune din imagine cu proprietăți speciale, în sensul că poate fi detectată dacă imaginea sufera transformări geometrice sau fotometrice (nu foarte mari)



Transformări geometrice





Exemplu:
translație +
rotație +
scalare

Transformări fotometrice

Slide adaptat dupa T. Tuytelaars

Trăsături locale

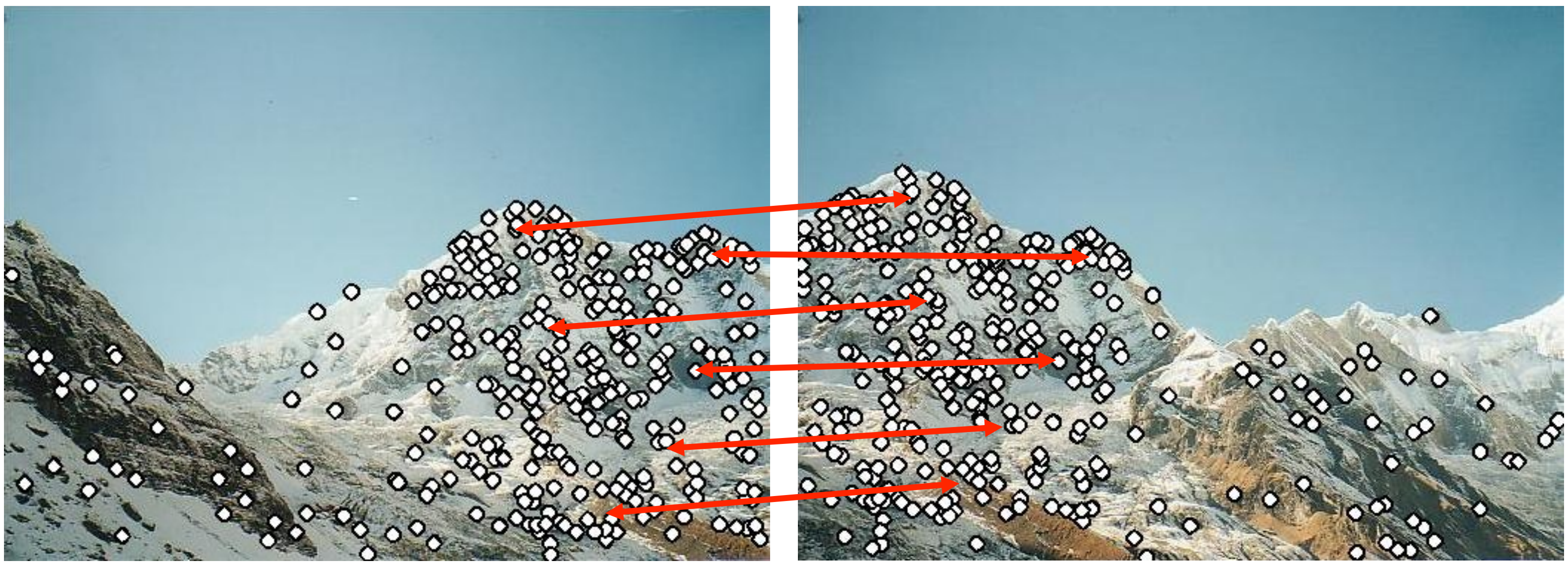
- posibilă aplicație: realizare de panorame

  - avem 2 imagini ale aceleași scene
  - cum le combinăm astfel încât să obținem o imagine mai mare (o panoramă)?

Slide adaptat după S. Lazebnik

Trăsături locale

- posibilă aplicație: realizare de panorame



Pasul 1: extrage trăsături (puncte din imagini + descriptori ai conținutului vizual)

Pasul 2: găsește corespondențe între trăsăturile extrase

Trăsături locale

- posibilă aplicație: realizare de panorame

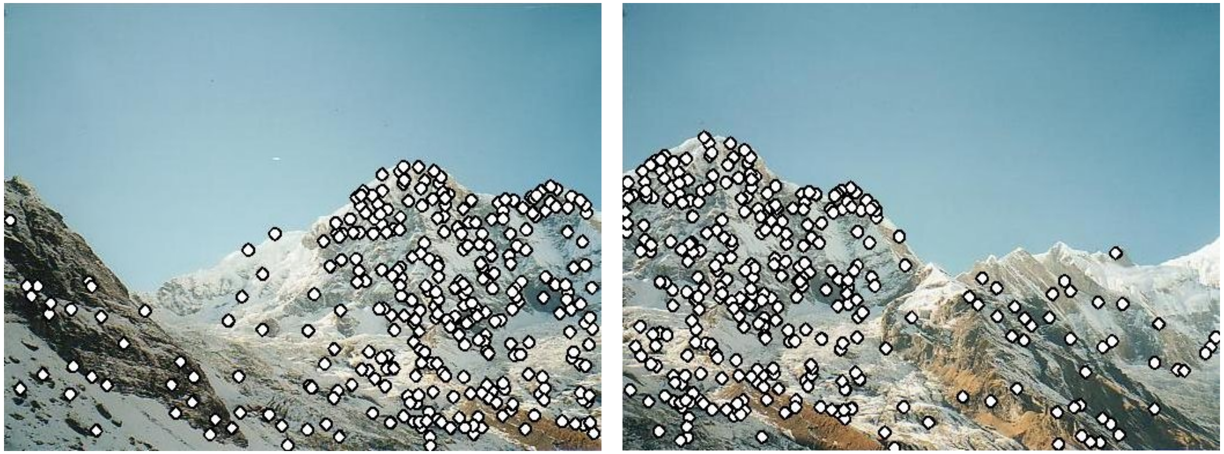
Pasul 1: extrage trăsături (puncte din imagini + descriptori ai conținutului vizual)

Pasul 2: găsește corespondențe între trăsăturile extrase

Pasul 3: aliniază imaginile (găsește homografia care aliniază imaginile)

Slide adaptat după S. Lazebnik

Ce proprietăți vrem să aibă trăsăturile locale?



- Repetabilitate
  - aceeași trăsătură locală să poată fi găsită în alte imagini obținute din imaginea inițială prin transformări geometrice (rotație, perspectivă, etc.) sau fotometrice (iluminarea scenei diferă zi/noapte)
- Unicitate
  - fiecare trăsătură locală are o descriere unică
- Reprezentare compactă a imaginii
  - mult mai puține trăsături locale decât pixeli
- Localitate
  - o trăsătură locală ocupă o porțiune mică dintr-o imagine; reprezentare robustă la mascarea obiectelor, background

Slide adaptat după S. Lazebnik

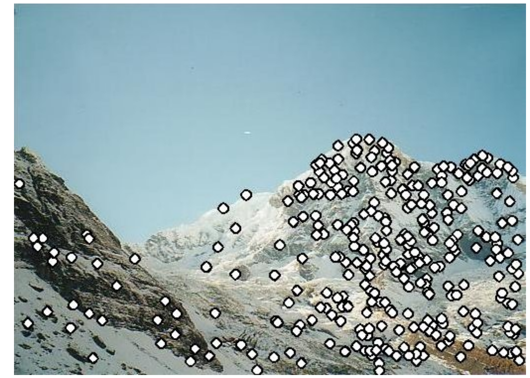
Trăsături locale: aplicații

- alinierea imaginilor
- reconstrucție 3D
- tracking
- navigarea roboților
- indexarea și regăsirea informației într-o colecție mare de imagini (milioane de imagini)
- recunoașterea obiectelor

Slide adaptat după S. Lazebnik

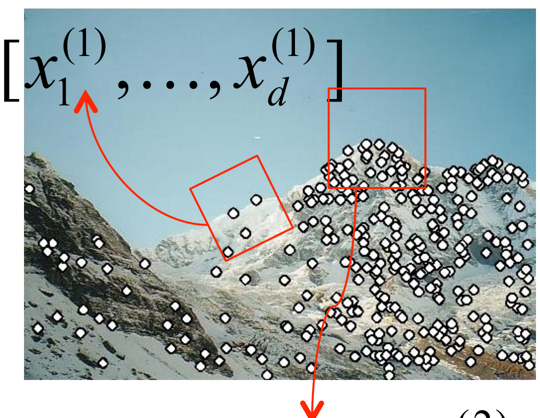
Trăsături locale: componentele principale

1) **Detectare**: localizează punctele de interes în imagine



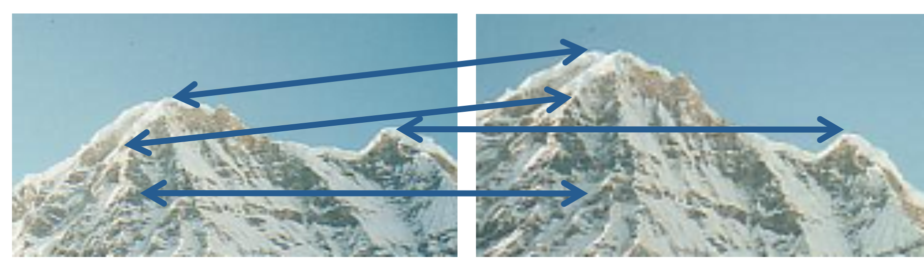
2) **Descriere**: descrie conținutul vizual din vecinătatea fiecărui punct de interes printr-un vector ce reprezintă descriptorul vizual (feature vector)

$$\mathbf{x}_1 = [x_1^{(1)}, \ldots, x_d^{(1)}]$$



$$\mathbf{x}_2 = [x_1^{(2)}, \ldots, x_d^{(2)}]$$

3) **Matching**: determină corespondențele dintre descriptorii dintre imagini



Slide adaptat după S. Lazebnik

Trăsături locale: repetabilitate

- vrem să găsim (mare parte din) aceleași puncte din ambele imagini.

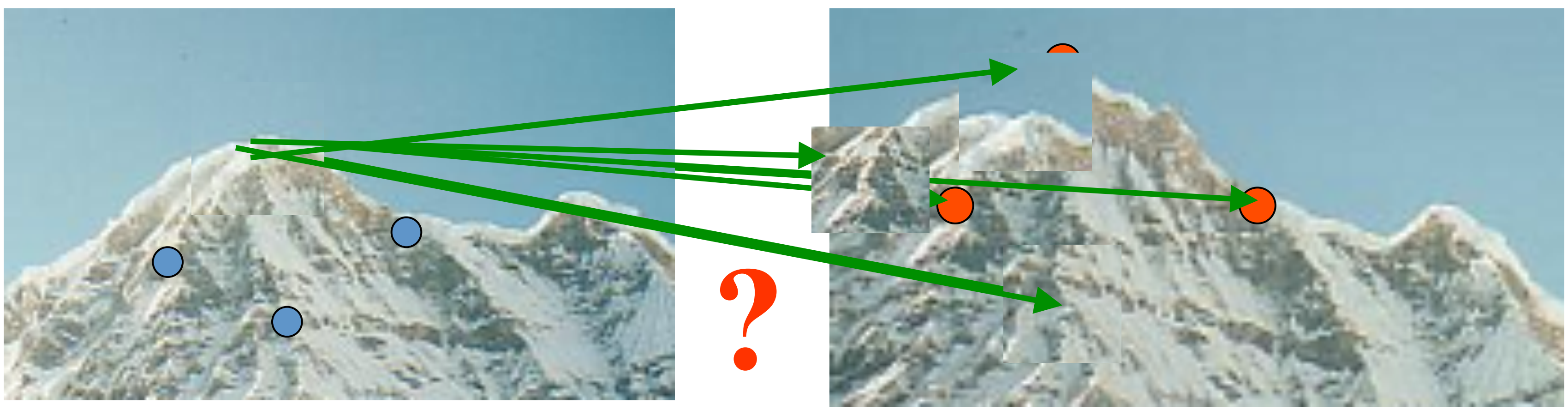
În acest caz nu am putea găsi corespondențe

- procedura de detectare a trăsăturilor locale rulează *independent* pentru fiecare imagine.

Slide adaptat după S. Lazebnik

Trăsături locale: unicitatea descrierii

- abilitate de a determina în mod corect corespondența dintre puncte

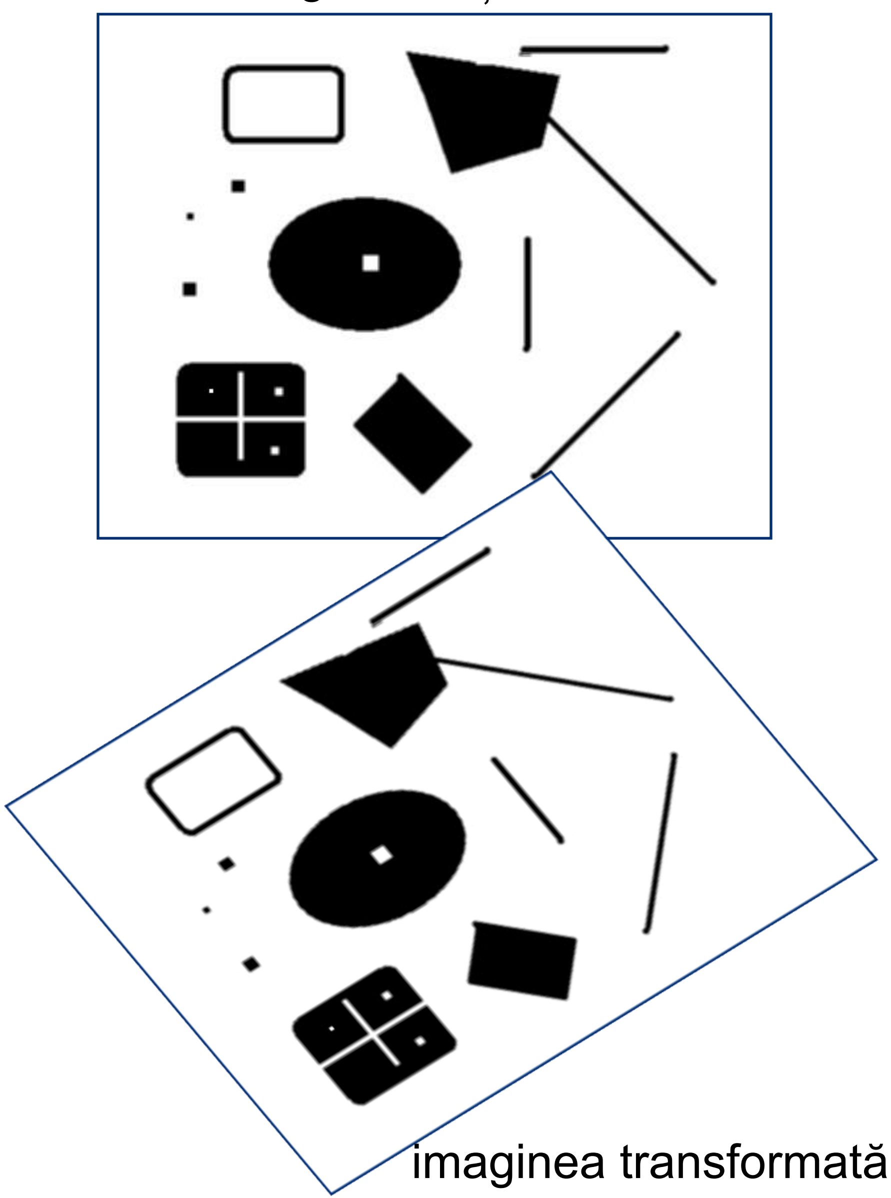


- descrierea regiunilor ocupate de trăsăturile locale trebuie să furnizeze invarianță la transformările geometrice și fotometrice dintre cele două imagini

Slide adaptat după S. Lazebnik

Trăsături locale: detectare

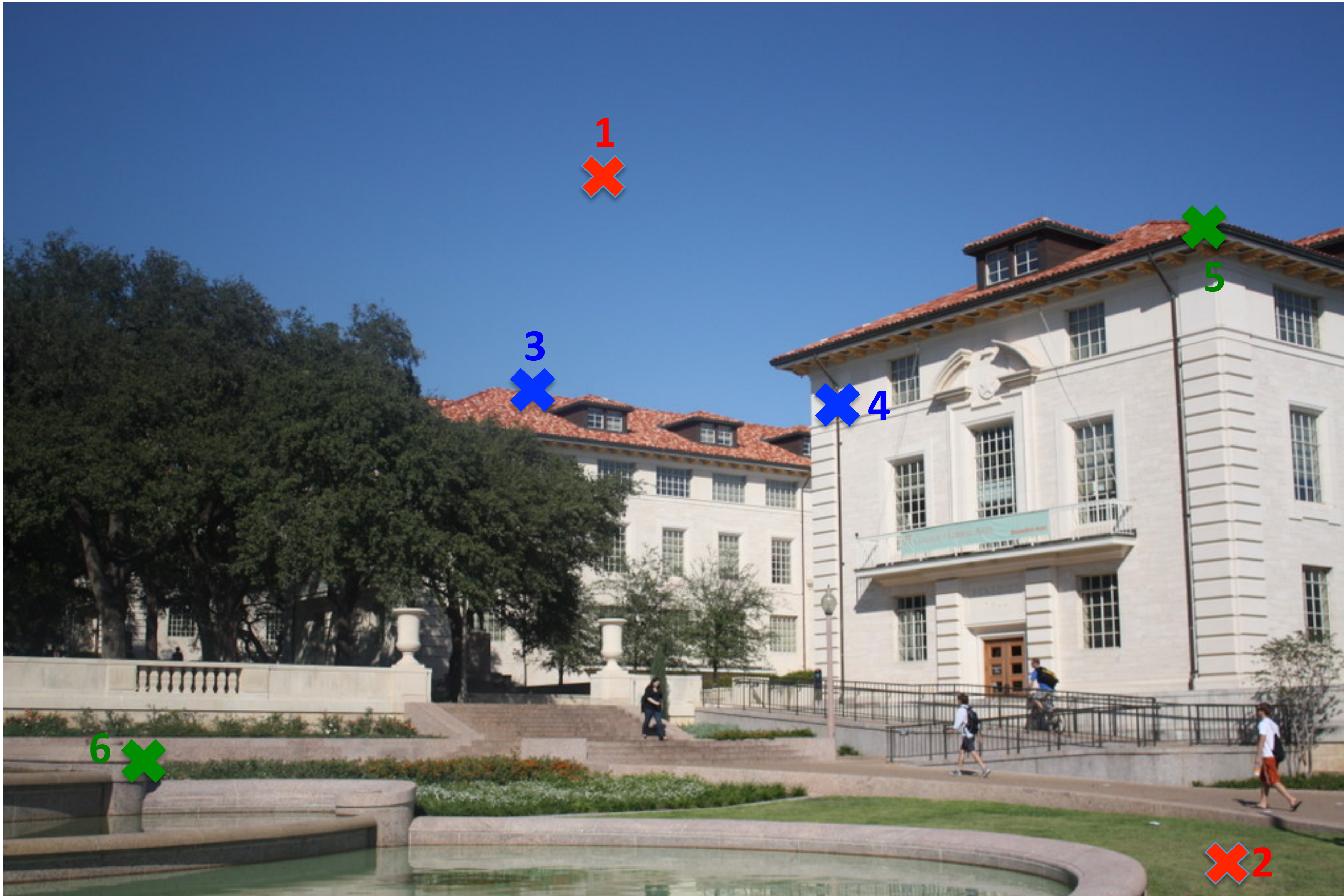
- trebuie să dați un click pe un punct din imagine
- plecați apoi din fața calculatorului și reveniți după ce am transformat imaginea
- trebuie să dați click pe punctul pe care l-ați ales inițial (îi găsiți corespondentul)
- ce puncte ați alege?



Trăsături locale: detectare

Detectare: localizează punctele de interes în imagine

Ce puncte ați alege?



1
2
3
4
5
6

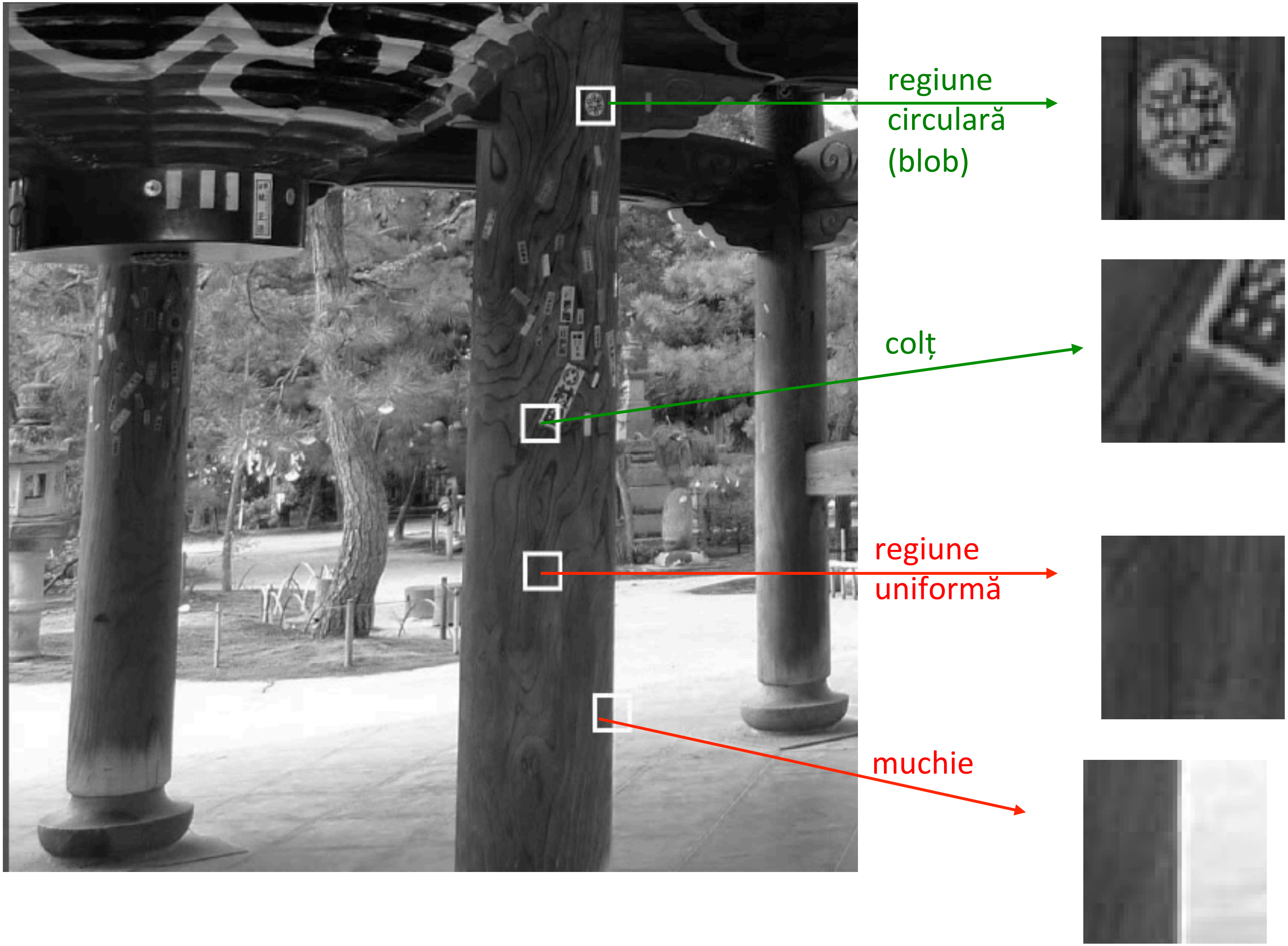
Trăsături locale: detectare

Detectare: localizează punctele de interes în imagine

Regiuni uniforme ambigue pentru matching

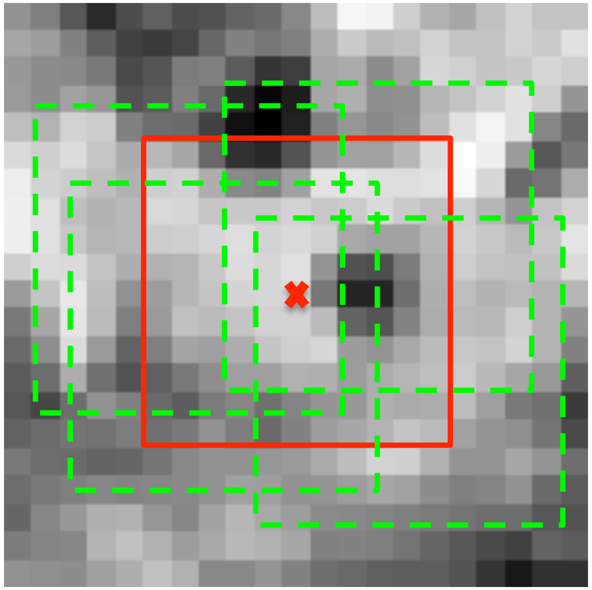
Muchii ambigue pentru matching

Colțuri bune pentru matching


regiune circulară (blob)
colț
regiune uniformă
muchie

Detectarea trăsăturilor locale

- cum formulăm matematic ce înseamnă o trăsătură locală bună?

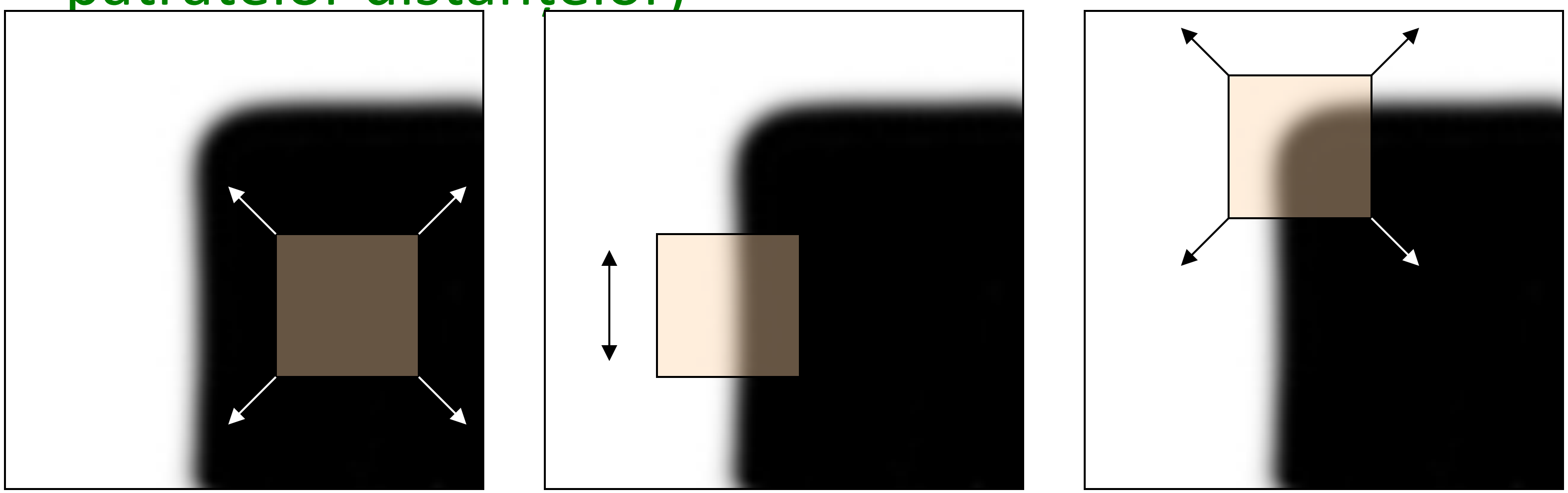


- caracterizăm punctul din mijloc cu o fereastră mică din jurul lui

- deplasarea ferestrei ***în orice direcție*** ar trebui să ducă la diferențe mari în intensitate (suma pătratelor distanțelor)

Detectarea colțurilor

- caracterizăm punctul din mijloc cu o fereastră mică din jurul lui
- deplasare ferestrei ***în orice direcție*** ar trebui să ducă la diferențe mari în intensitate (suma pătratelor distanțelor)



regiune "flat" : nicio schimbare în toate direcțiile

"muchie": nicio schimbare în direcția muchiei

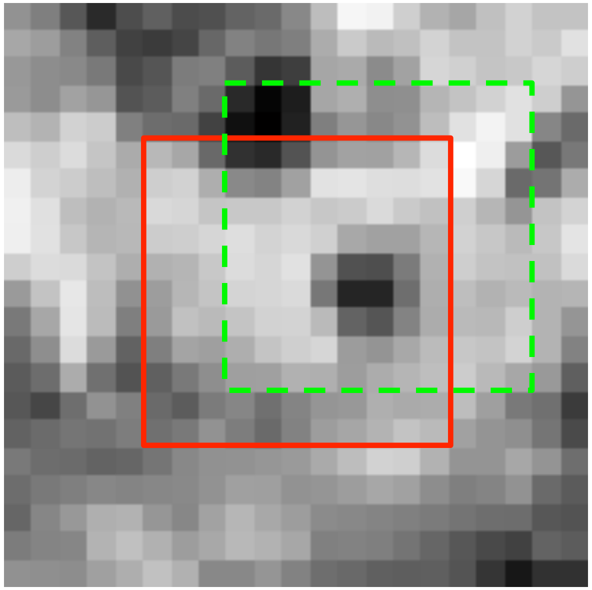
"colț": schimbări semnificative în toate direcțiile

Detectarea colțurilor

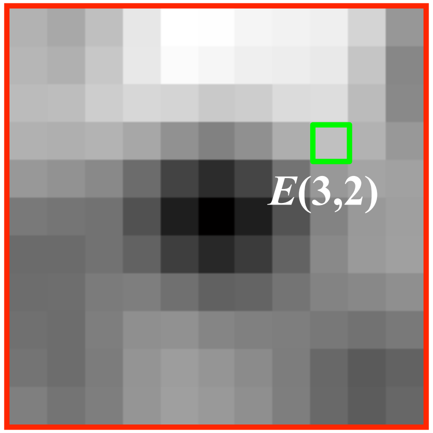
Schimbarea în înfățișare a ferestrei W la deplasarea [u,v]:

$$E(u,v) = \sum_{(x,y)\in W} \left(I(x+u, y+v) - I(x,y)\right)^2$$

$I(x, y)$



$E(u, v)$

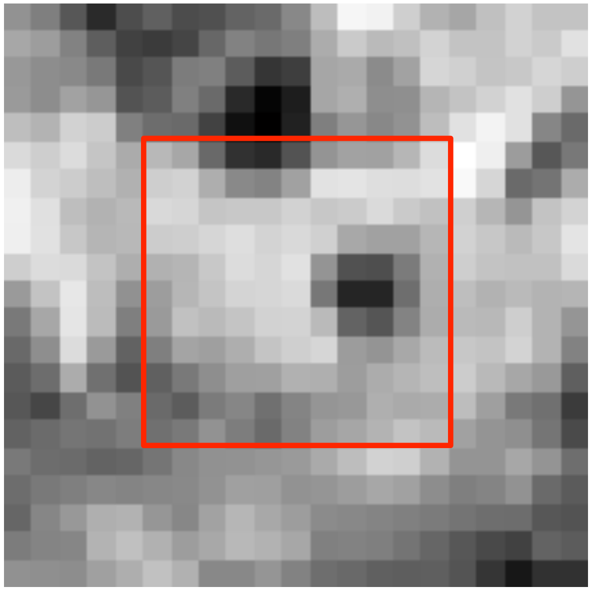


Detectarea colțurilor

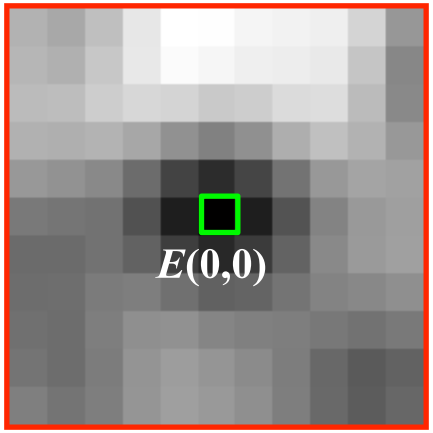
Schimbarea în înfățișare a ferestrei W la deplasarea [u,v]:

$$E(u,v) = \sum_{(x,y)\in W} \left(I(x+u, y+v) - I(x,y) \right)^2$$

$I(x, y)$



$E(u, v)$



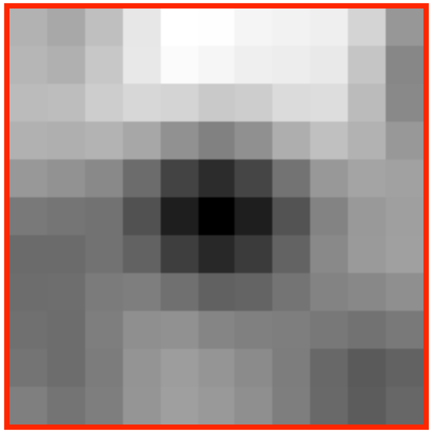
Detectarea colțurilor

Schimbarea în înfățișare a ferestrei W la deplasarea [u,v]:

$$E(u,v) = \sum_{(x,y)\in W} \left(I(x+u, y+v) - I(x,y) \right)^2$$

Vrem să studiem comportamentul funcției E(u,v) pentru mici deplasări [u,v]

$E(u, v)$



Detectarea colțurilor

Dezvoltarea în serie Taylor a funcțiilor de două variabile I(x+u,y+v) și E(u,v) în jurul lui (u,v) = (0,0)

$$I(x+u, y+v) \approx I(x,y) + u \cdot \frac{\partial I}{\partial x}(x,y) + v \cdot \frac{\partial I}{\partial y}(x,y)$$

$$I(x+u, y+v) \approx I(x,y) + u \cdot I_x + v \cdot I_y$$

Înlocuim în expresia funcției E(u,v) pentru mici deplasări [u,v]

$$E(u,v) = \sum_{(x,y)\in W} \left(I(x+u, y+v) - I(x,y)\right)^2$$

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

Detectarea colțurilor

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

E(u,v) poate fi scris ca:

$$E(u,v) \approx \begin{bmatrix} u & v \end{bmatrix} M \begin{bmatrix} u \\ v \end{bmatrix}$$

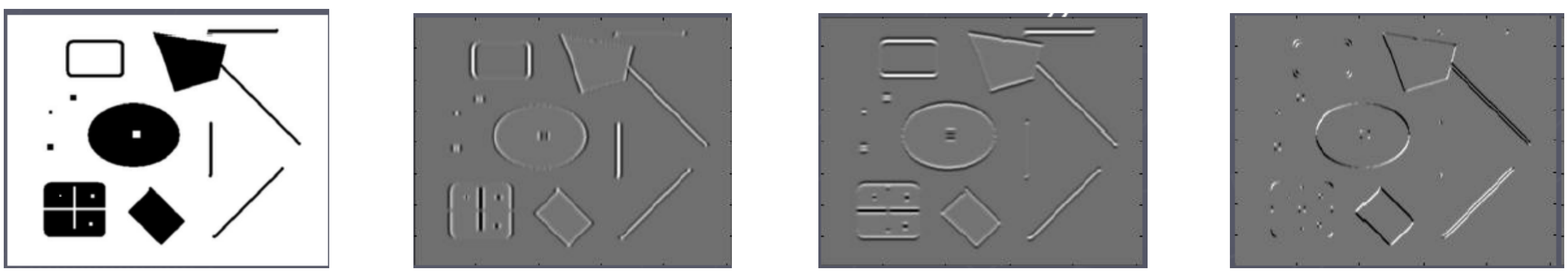
unde *M* este *matricea de ordin doi cu derivatele imaginii* I_x *și* I_y

$$M = \begin{bmatrix} \sum_{x,y} I_x I_x & \sum_{x,y} I_x I_y \\ \sum_{x,y} I_x I_y & \sum_{x,y} I_y I_y \end{bmatrix}$$

Detectarea colțurilor

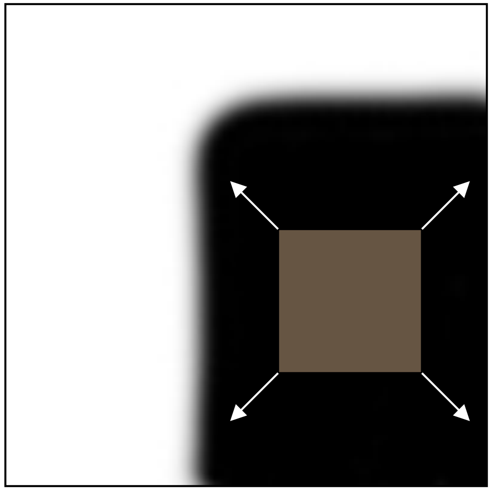
$$M = \begin{bmatrix} \sum_{x,y} I_x I_x & \sum_{x,y} I_x I_y \\ \sum_{x,y} I_x I_y & \sum_{x,y} I_y I_y \end{bmatrix}$$

M este *matricea de ordin doi cu derivatele imaginii I_x și I_y în vecinătatea unui punct*



$$I_x \Leftrightarrow \frac{\partial I}{\partial x} \qquad I_y \Leftrightarrow \frac{\partial I}{\partial y} \qquad I_x I_y \Leftrightarrow \frac{\partial I}{\partial x}\frac{\partial I}{\partial y}$$

Ce ne spune matricea M?



regiune “flat” :
nicio schimbare
în toate direcțiile

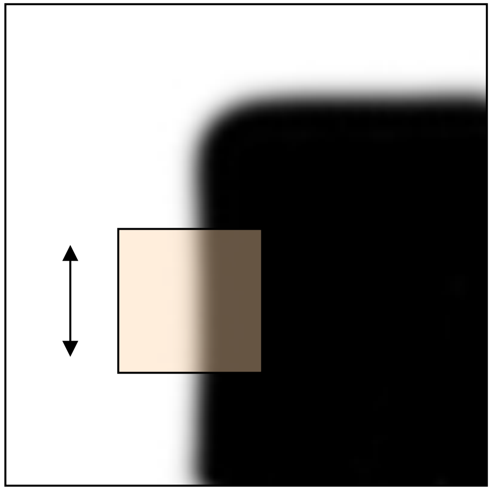
$$M = \begin{bmatrix} \sum_{x,y} I_x I_x & \sum_{x,y} I_x I_y \\ \sum_{x,y} I_x I_y & \sum_{x,y} I_y I_y \end{bmatrix}$$

$$M \approx \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

Valorile proprii ale lui M = soluțiile ecuației $\det(M - \lambda I_2) = 0$ sunt $\lambda_1 \approx \lambda_2 \approx 0$

Ce ne spune matricea M?

Cazul în care muchiile sunt paralele cu axele imaginii



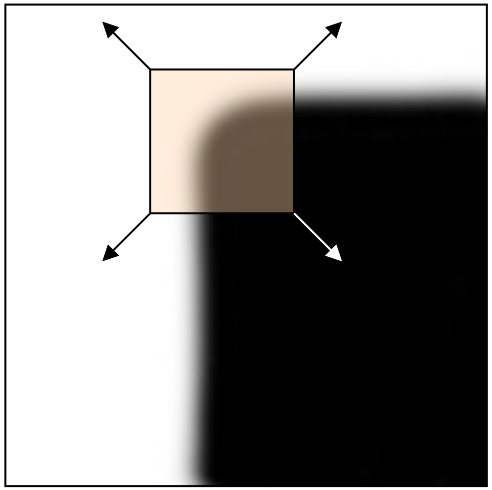
$$M = \begin{bmatrix} \sum\limits_{x,y} I_x I_x & \sum\limits_{x,y} I_x I_y \\ \sum\limits_{x,y} I_x I_y & \sum\limits_{x,y} I_y I_y \end{bmatrix}$$

"muchie":
nicio schimbare în direcția muchiei

$$M \approx \begin{bmatrix} \lambda & 0 \\ 0 & 0 \end{bmatrix}$$

Valorile proprii ale lui M = soluțiile ecuației det(M – λI_2) = 0 sunt $\lambda_1 >> 0$, $\lambda_2 \approx 0$

Ce ne spune matricea M?



"colț" aliniat cu axele:
schimbări semnificative
în toate direcțiile

$$M = \begin{bmatrix} \sum_{x,y} I_x I_x & \sum_{x,y} I_x I_y \\ \sum_{x,y} I_x I_y & \sum_{x,y} I_y I_y \end{bmatrix}$$

$$M \approx \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}$$

Valorile proprii ale lui M = soluțiile ecuației $\det(M - \lambda I_2) = 0$ sunt λ_1, $\lambda_2 >> 0$

Ce ne spune matricea M?

$$M = \begin{bmatrix} \sum_{x,y} I_x I_x & \sum_{x,y} I_x I_y \\ \sum_{x,y} I_x I_y & \sum_{x,y} I_y I_y \end{bmatrix} \qquad M \approx \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}$$

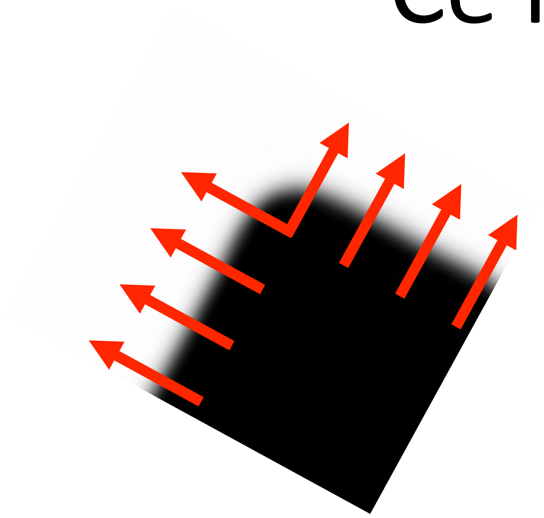
Valorile proprii ale lui M = soluțiile ecuației $\det(M - \lambda I_2) = 0$ sunt λ_1, λ_2

Caută punctele unde ambele valori proprii λ sunt mari.

Dacă o valoare proprie λ este aproape de 0, atunci punctul **NU** este un colț.

Ce se întâmplă dacă colțul nu este aliniat cu axele imaginii, Ox și Oy?

Ce ne spune matricea M?



$$M = \begin{bmatrix} \sum\limits_{x,y} I_x I_x & \sum\limits_{x,y} I_x I_y \\ \sum\limits_{x,y} I_x I_y & \sum\limits_{x,y} I_y I_y \end{bmatrix}$$

Întrucât *M* e simetrică avem:

$$M = X \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} X^T \qquad Mx_i = \lambda_i x_i$$

Valorile proprii ale lui M codifică cantitatea schimbării în intensitate în direcțiile date de vectorii proprii ai lui M

Vectori proprii și valori proprii

$$A = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}.$$

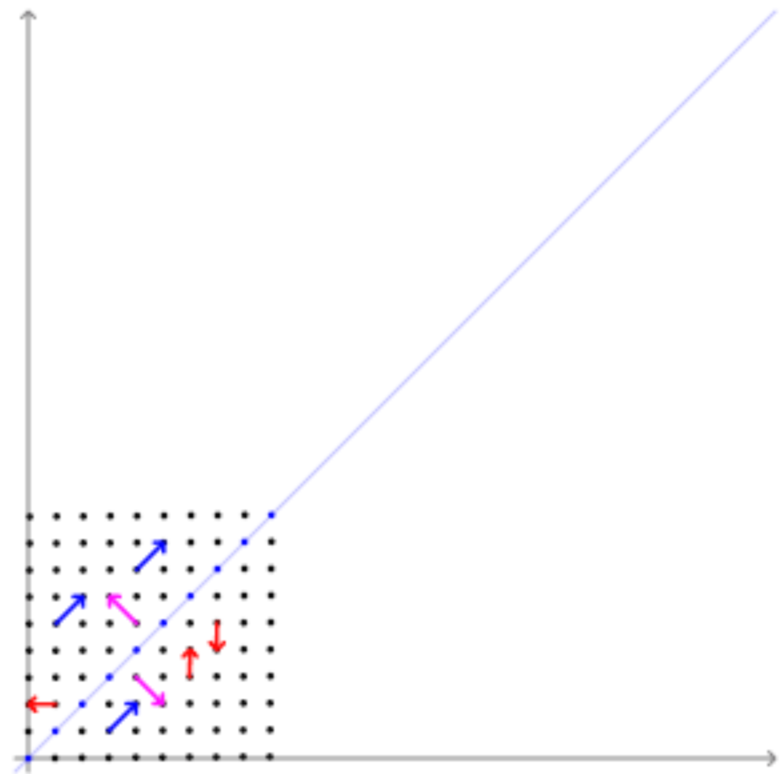
Valorile proprii ale lui M = soluțiile ecuației det(M – λI_2) = 0

$$|A - \lambda I| = \left| \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right| = \begin{vmatrix} 2-\lambda & 1 \\ 1 & 2-\lambda \end{vmatrix}, \\ = 3 - 4\lambda + \lambda^2.$$

$\lambda_1 = 1$, $\lambda_2 = 3$

Vectorii proprii ai lui A

$$v_{\lambda=1} = \begin{bmatrix} 1 \\ -1 \end{bmatrix} \qquad v_{\lambda=3} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$



Vectorii proprii ai lui A:

- rămân paraleli după transformarea A
- lungimea lor se modifică conform valorilor proprii

https://en.wikipedia.org/wiki/Eigenvalues_and_eigenvectors

Vectori proprii și valori proprii

$$A = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}.$$

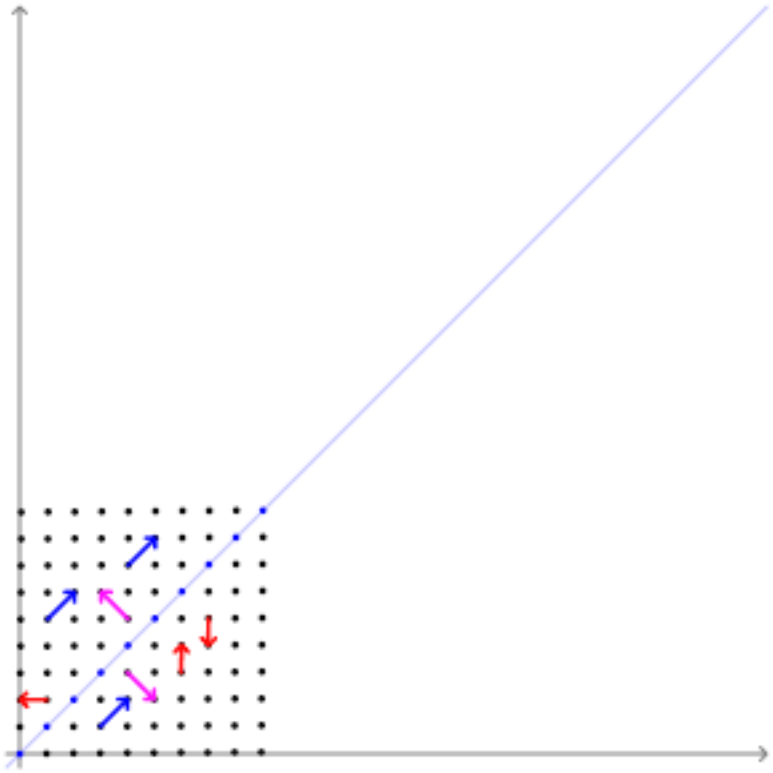
Valorile proprii ale lui M = soluțiile ecuației $\det(M - \lambda I_2) = 0$

$$|A - \lambda I| = \left| \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right| = \begin{vmatrix} 2-\lambda & 1 \\ 1 & 2-\lambda \end{vmatrix},$$
$$= 3 - 4\lambda + \lambda^2.$$

$\lambda_1 = 1$, $\lambda_2 = 3$

Vectorii proprii ai lui A

$$v_{\lambda=1} = \begin{bmatrix} 1 \\ -1 \end{bmatrix} \qquad v_{\lambda=3} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

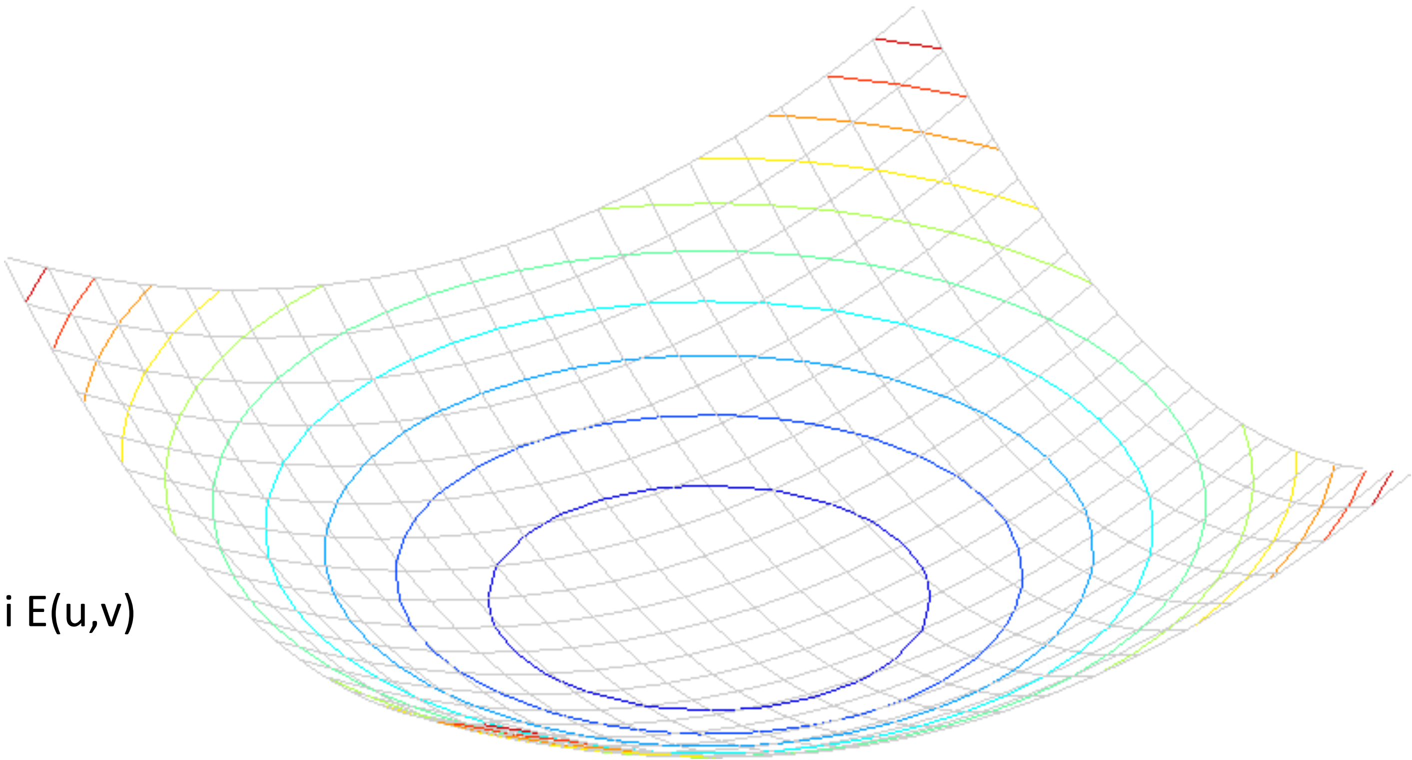


$$Av_{\lambda=1} = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ -1 \end{bmatrix} = \begin{bmatrix} 1 \\ -1 \end{bmatrix} \qquad Av_{\lambda=3} = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 3 \\ 3 \end{bmatrix}$$

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

https://en.wikipedia.org/wiki/Eigenvalues_and_eigenvectors

Interpretarea lui E(u,v)



Graficul lui E(u,v)

Considerăm o secțiune orizontală al ui *E*(*u*, *v*): $[u \;\; v] \; M \begin{bmatrix} u \\ v \end{bmatrix} = \text{const}$

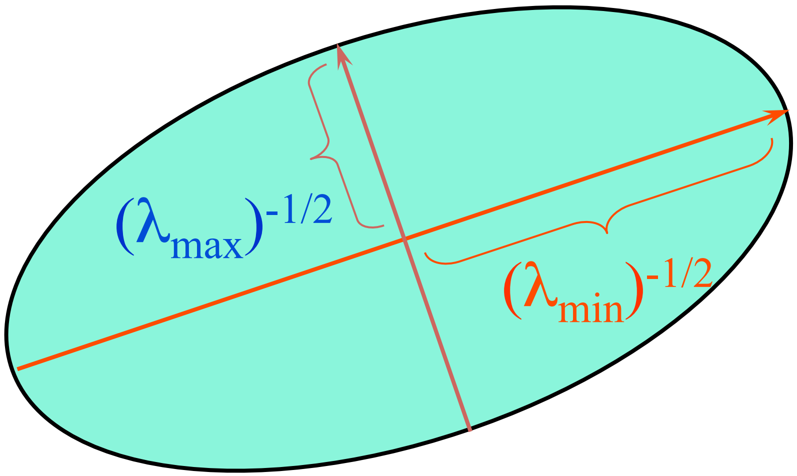
Obținem ecuația unei elipse

Interpretarea lui E(u,v)

Considerăm o secțiune orizontală al ui *E*(*u*, *v*): $[u \ \ v] \ M \begin{bmatrix} u \\ v \end{bmatrix} = \text{const}$

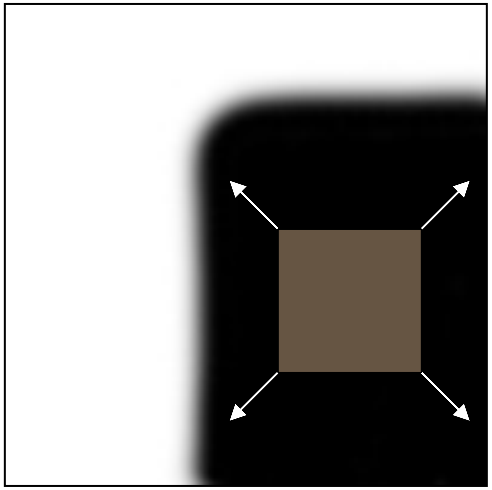
Obținem ecuația unei elipse

Lungimile axelor elipsei sunt determinate de valorile proprii, orientarea elipsei e dată de X



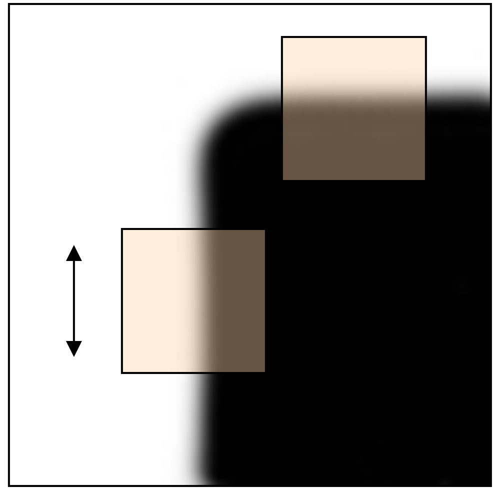
$$M = X \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} X^T$$

Funcția cornerness



“uniform”:

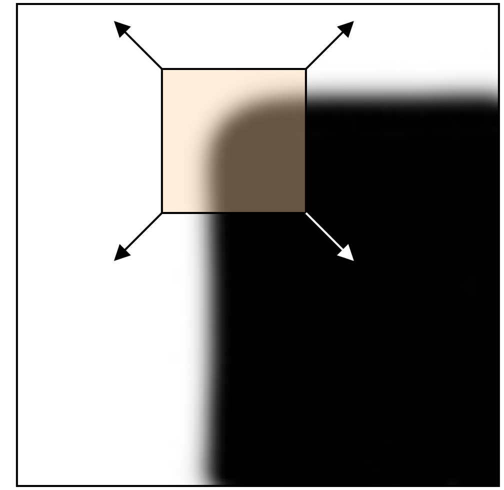
λ_1 și λ_2 sunt mici;



“muchie”:

$\lambda_1 >> \lambda_2$

$\lambda_2 >> \lambda_1$



“colț”:

λ_1 și λ_2 sunt mari,
$\lambda_1 \sim \lambda_2$;

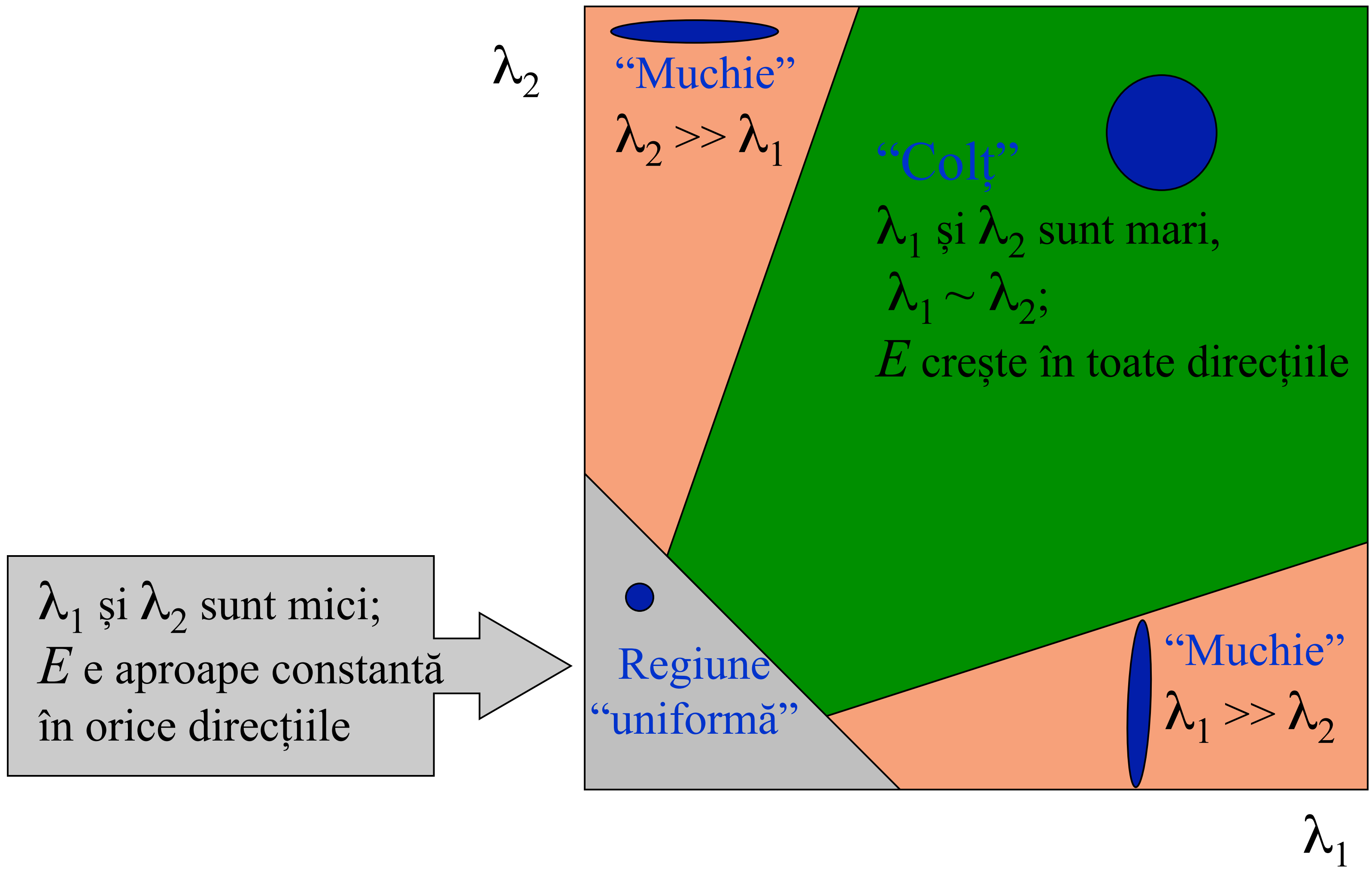
Funcția cornerness R măsoară cât de mult un punct este “colț”.
Posibile formulări:

$$R = \frac{\lambda_1 \cdot \lambda_2}{\lambda_1 + \lambda_2}$$

$$R = det(M) - \alpha \cdot trace(M)^2 = \lambda_1 \cdot \lambda_2 - \alpha * (\lambda_1 + \lambda_2)^2$$

Interpretarea valorilor proprii

Clasificarea punctelor imaginii pe baza valorilor proprii ale lui *M*:

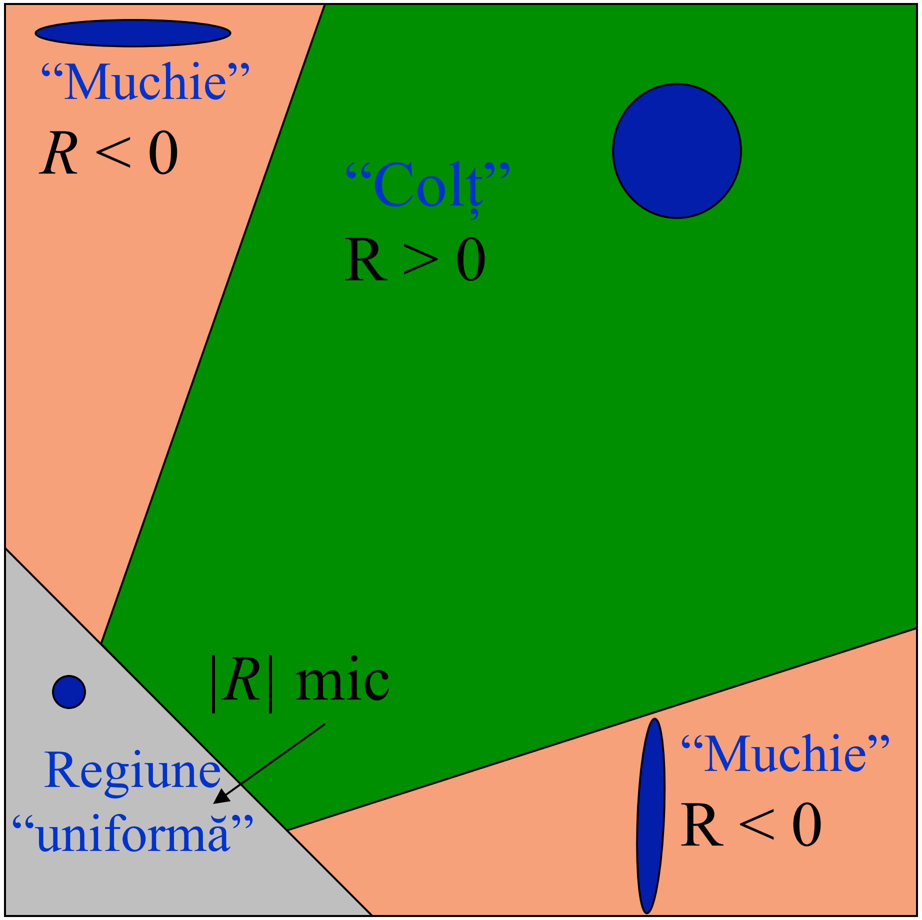


Slide adaptat după S. Lazebnik

Valori ale funcției cornerness R

$$R = det(M) - \alpha \cdot trace(M)^2 = \lambda_1 \cdot \lambda_2 - \alpha * (\lambda_1 + \lambda_2)^2$$

α: constant (0.04 - 0.06)



Slide adaptat după S. Lazebnik

Detectorul de colțuri Harris

1) Calculează derivate parțiale I_x și I_y la fiecare pixel (x,y)

2) Calculează matricea M pentru o fereastră centrată în fiecare pixel (x,y) din imagine

3) Calculează funcția cornerness R la fiecare pixel (x,y)

4) Găsește punctele pentru care R are valori foarte mari (> threshold).

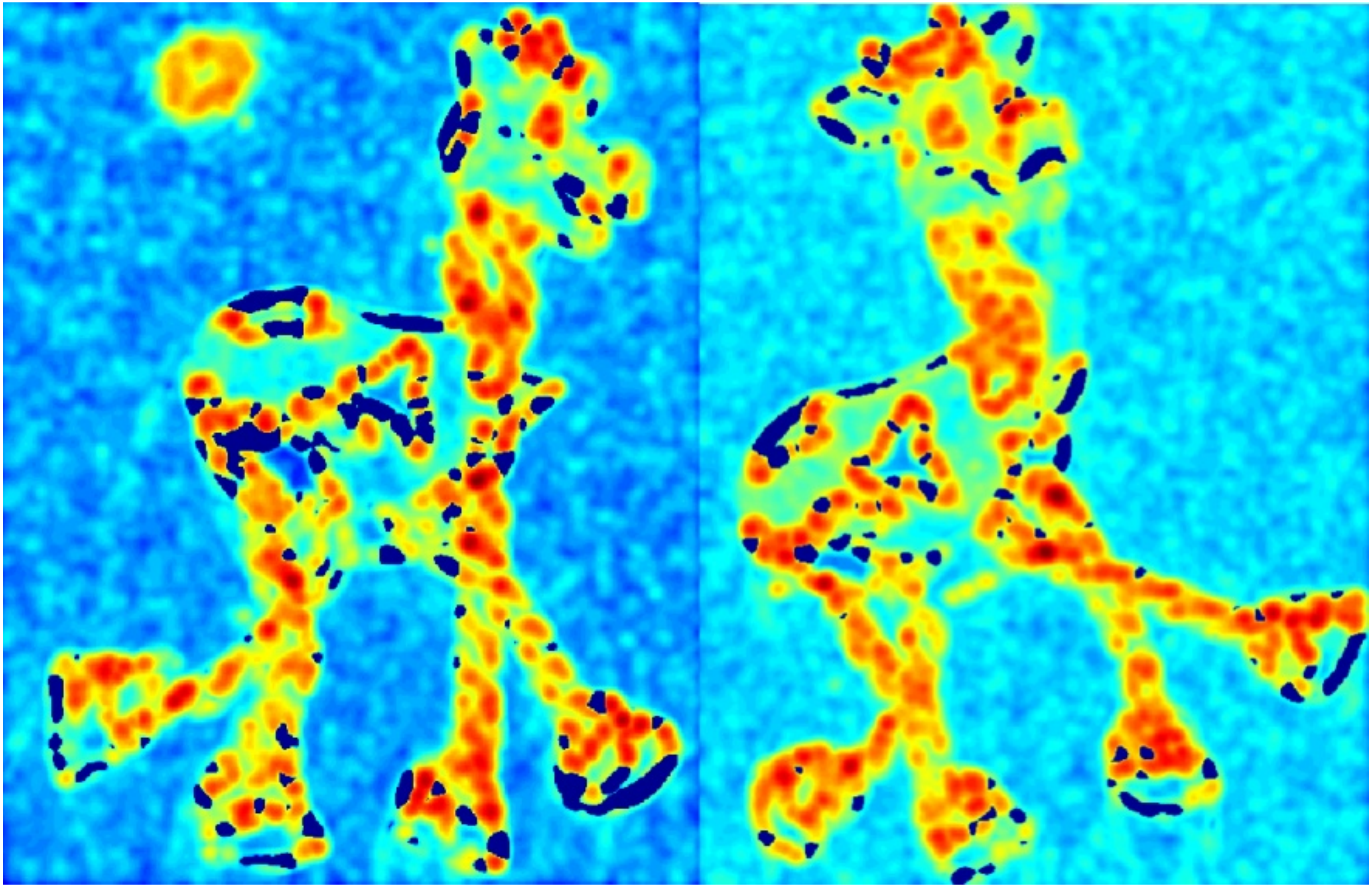
5) Găsește maximele locale ale funcției, realizează suprimarea maximelor (non-maximum supression)

C.Harris and M.Stephens. [“A Combined Corner and Edge Detector.”] *Proceedings of the 4th Alvey Vision Conference*: pages 147—151, 1988.

Detectorul de colțuri Harris: pași

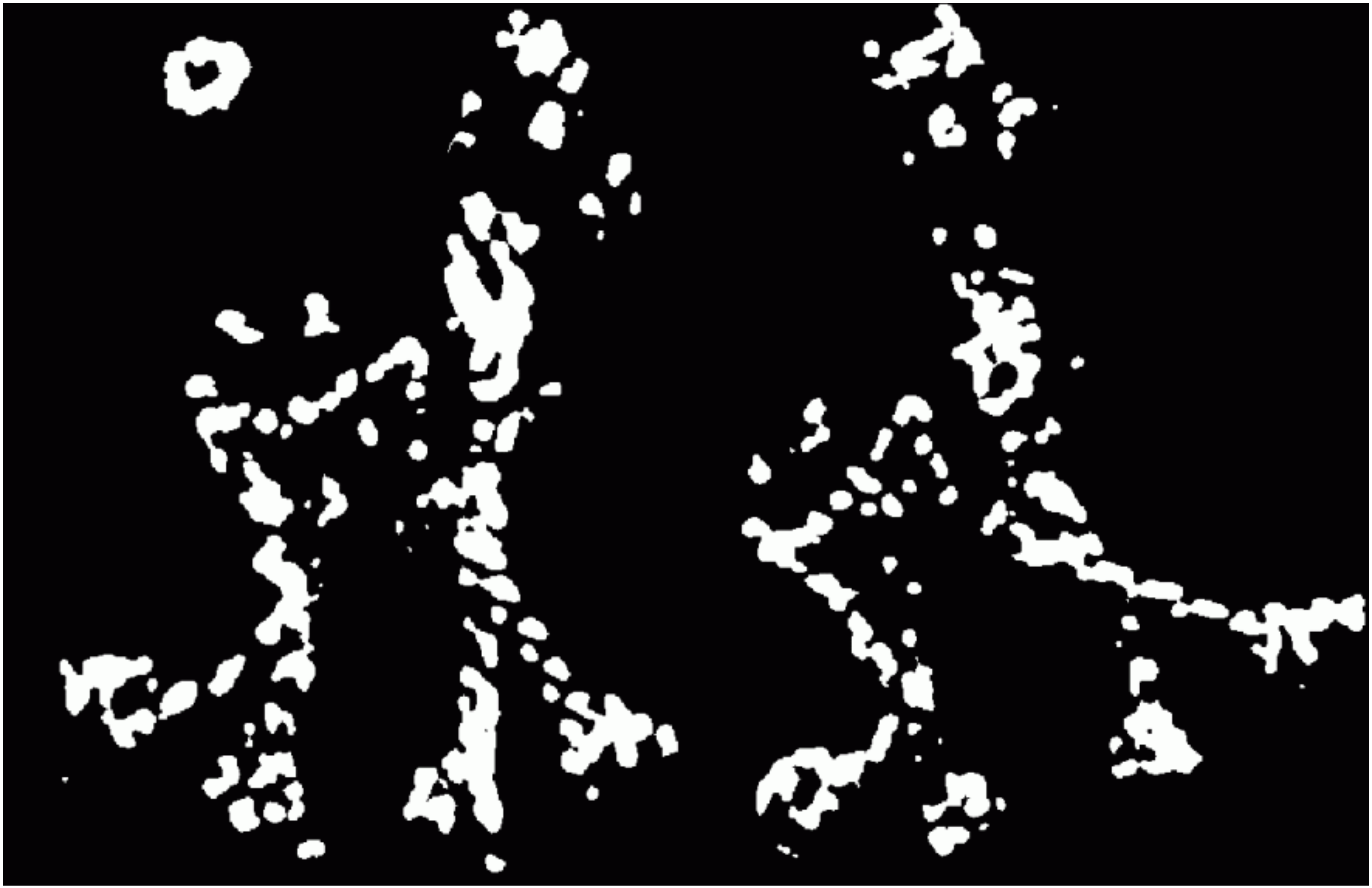
Detectorul de colțuri Harris: pași

Calculul funcției cornerness R



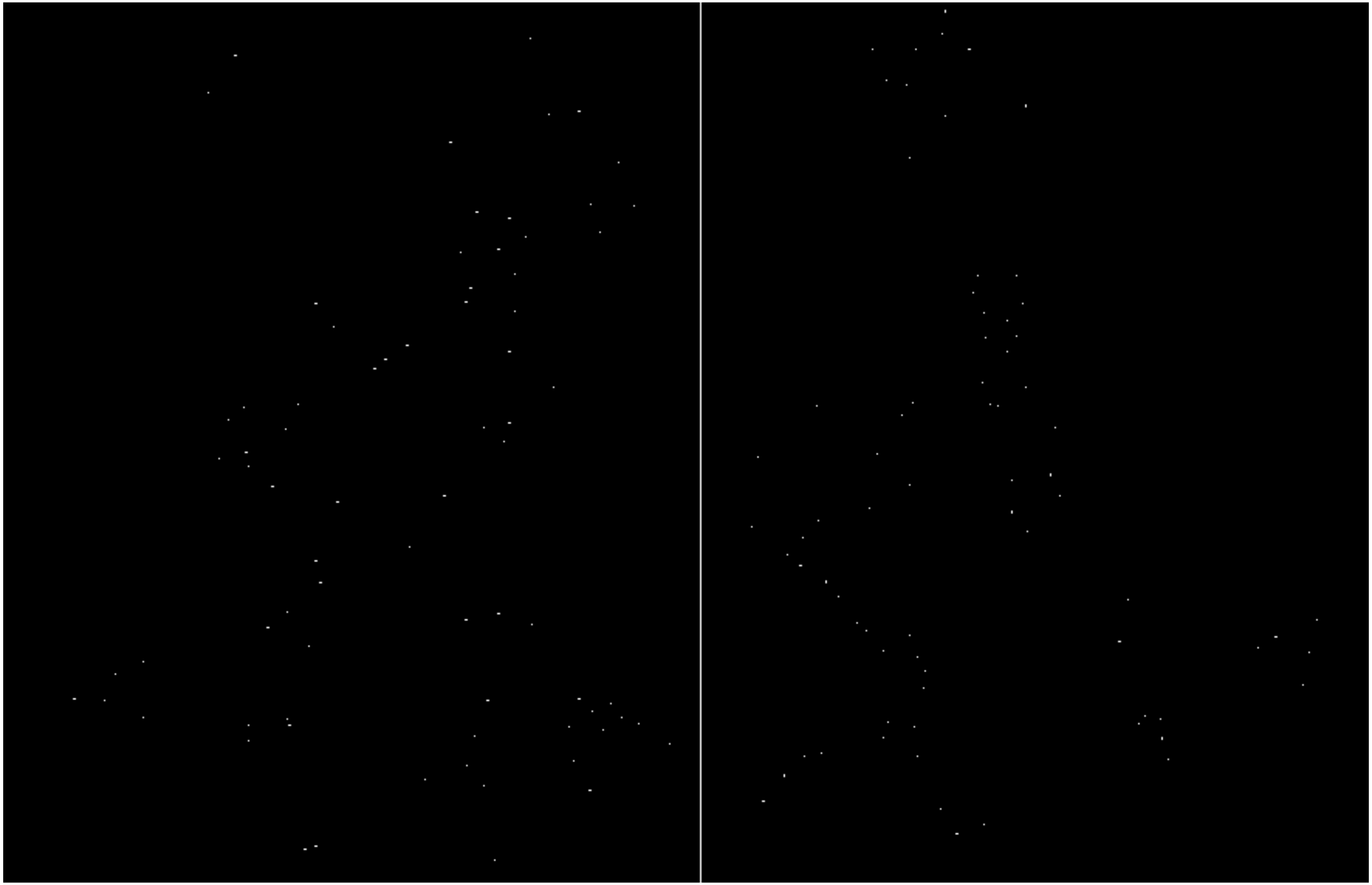
Detectorul de colțuri Harris: pași

Găsește pixelii (x,y) din imagine cu R(x,y) > threshold



Detectorul de colțuri Harris: pași

Păstrează pixelii (x,y) din imagine cu R(x,y) maxim local

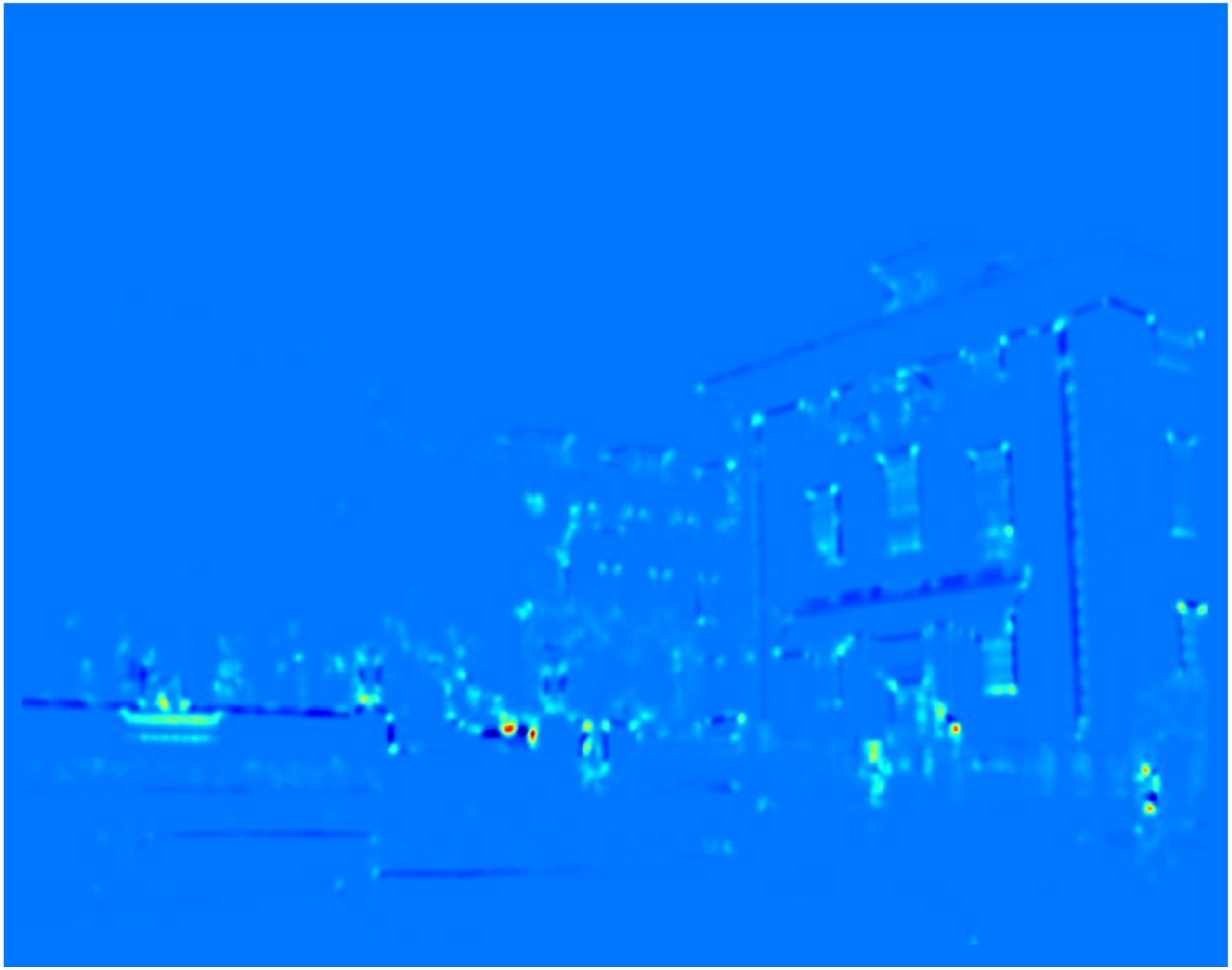


Exemple pentru detectorul Harris

Slide adaptat după K. Graumann

Exemple pentru detectorul Harris

Calculul funcției cornerness R



Slide adaptat după K. Graumann

Exemple pentru detectorul Harris

Slide adaptat după K. Graumann

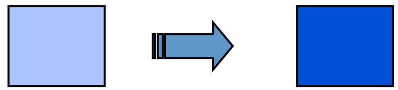
Invarianță și covarianță

Vrem să găsim pixeli din imagini care sunt colțuri care să fie ***invariante*** la transformări fotometrice și ***covariante*** la transformări geometrice

Invarianță: imaginea e transformată fotometric și poziția colțului nu se schimbă

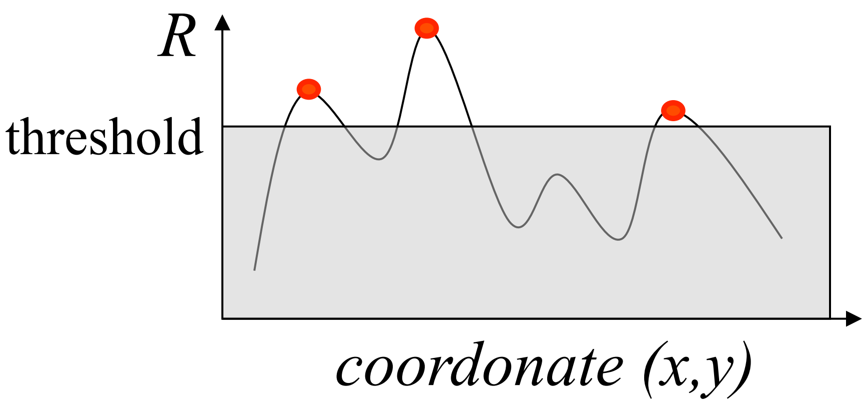
Covarianță: dacă avem două versiuni ale aceleași imagini obținute prin transformări geometrice (translație, rotație, trandformare afină, perspectivă), colțurile ar trebui detectate în poziții corespondente.

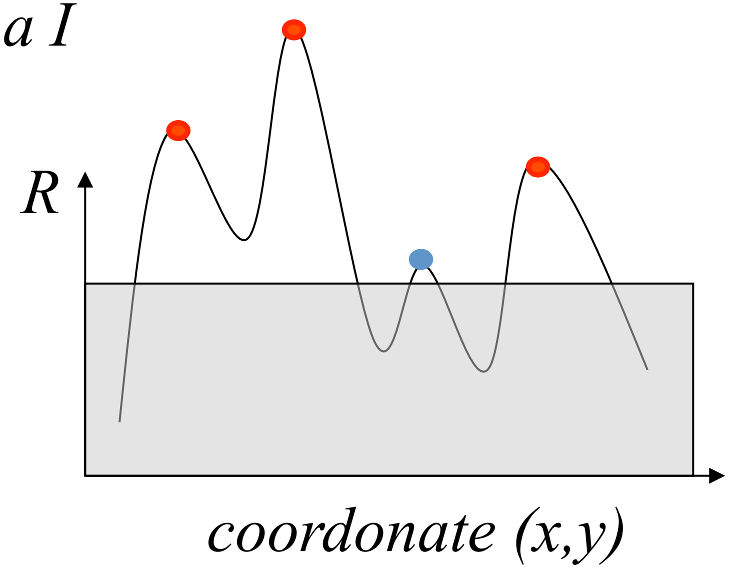
Transformări fotometrice: schimbări afine de intensitate



$I \rightarrow a\,I + b$

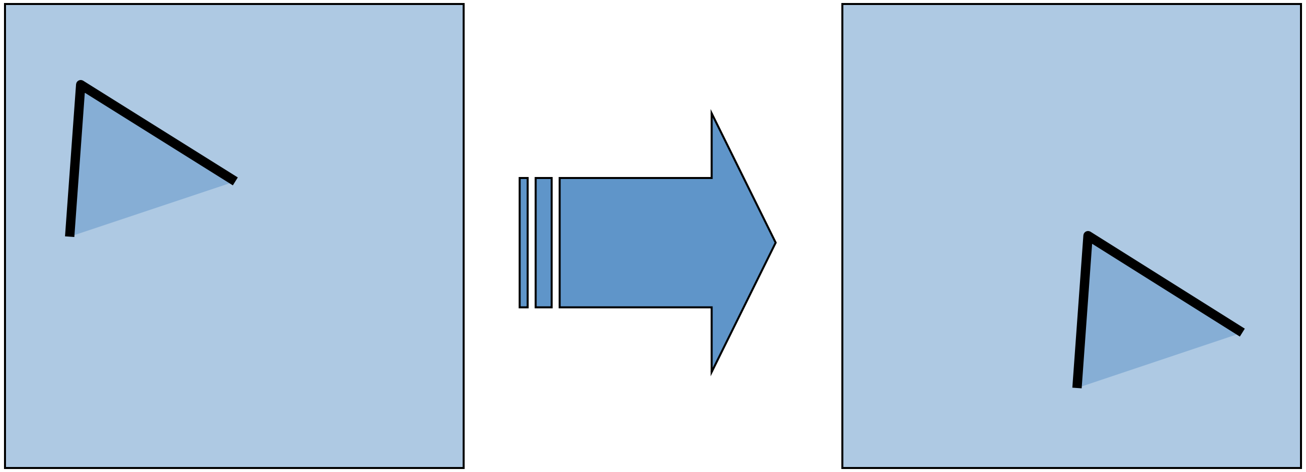
- folosim numai derivate => invarianță la adunare/scădere în $I \rightarrow I + b$
- înmulțire cu scalar: $I \rightarrow a\,I$





Parțial invarianță la schimbări afine de intensitate

Transformări geometrice: translații

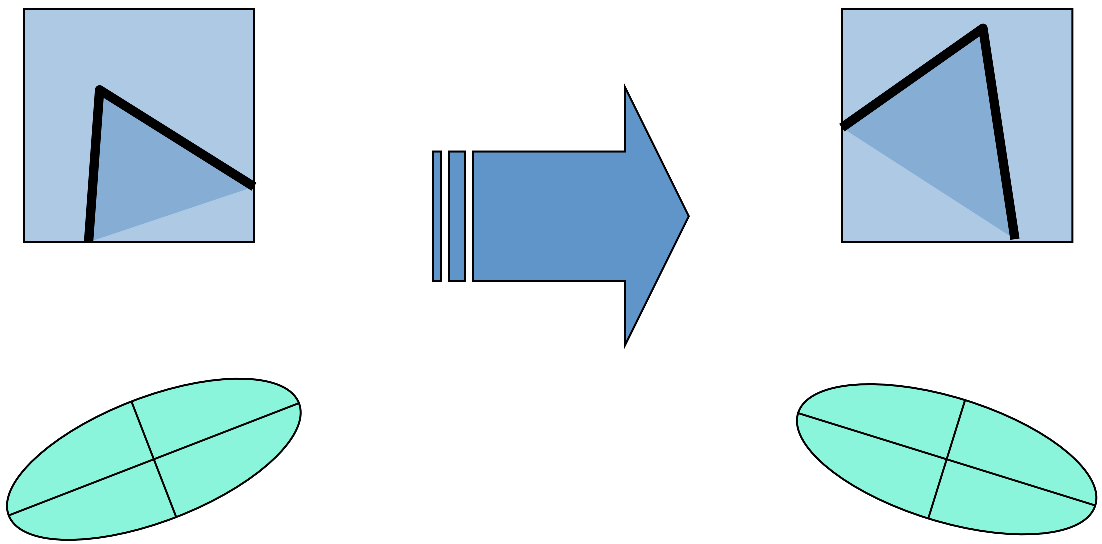


- derivatele sunt covariante la translație

Poziția colțului este covariantă la translații

Slide adaptat după K. Graumann

Transformări geometrice: rotații

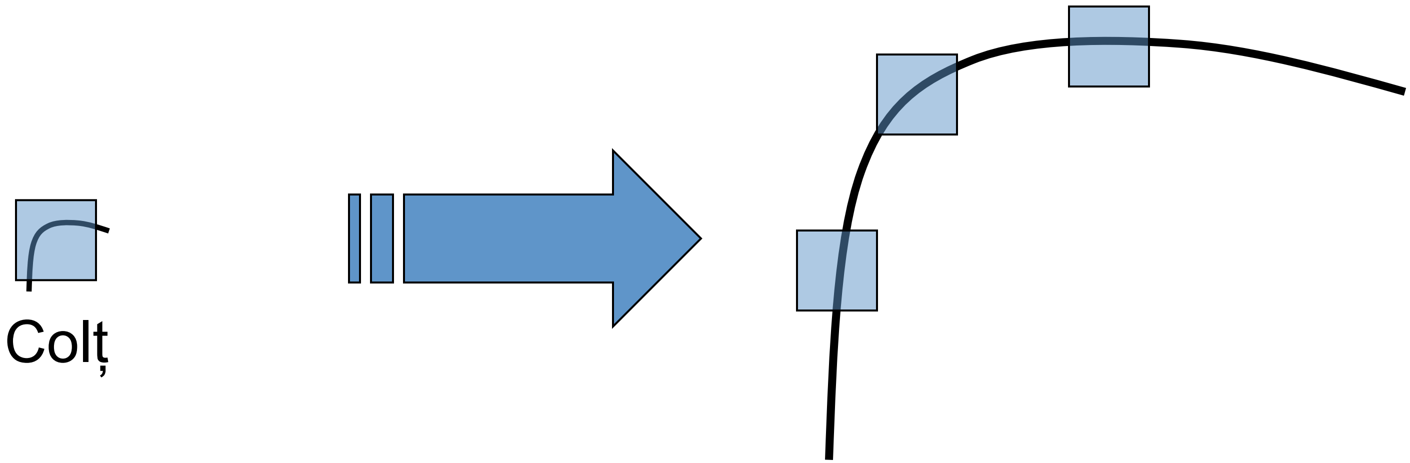


Elipsa determinată de M se rotește însă forma sa (dată de valorile proprii) rămâne aceeași

Poziția colțului este covariantă la translații

Slide adaptat după K. Graumann

Transformări geometrice: scalare a imaginii (mărire, micșorare)



Toate punctele sunt clasificate ca fiind muchii

Poziția colțului nu este covariantă la scalarea imaginilor

Slide adaptat după K. Graumann

Detectorul de colțuri Harris

1) Calculează derivate parțiale I_x și I_y la fiecare pixel (x,y)

2) Calculează matricea M pentru o fereastră W centrată în fiecare pixel (x,y) din imagine

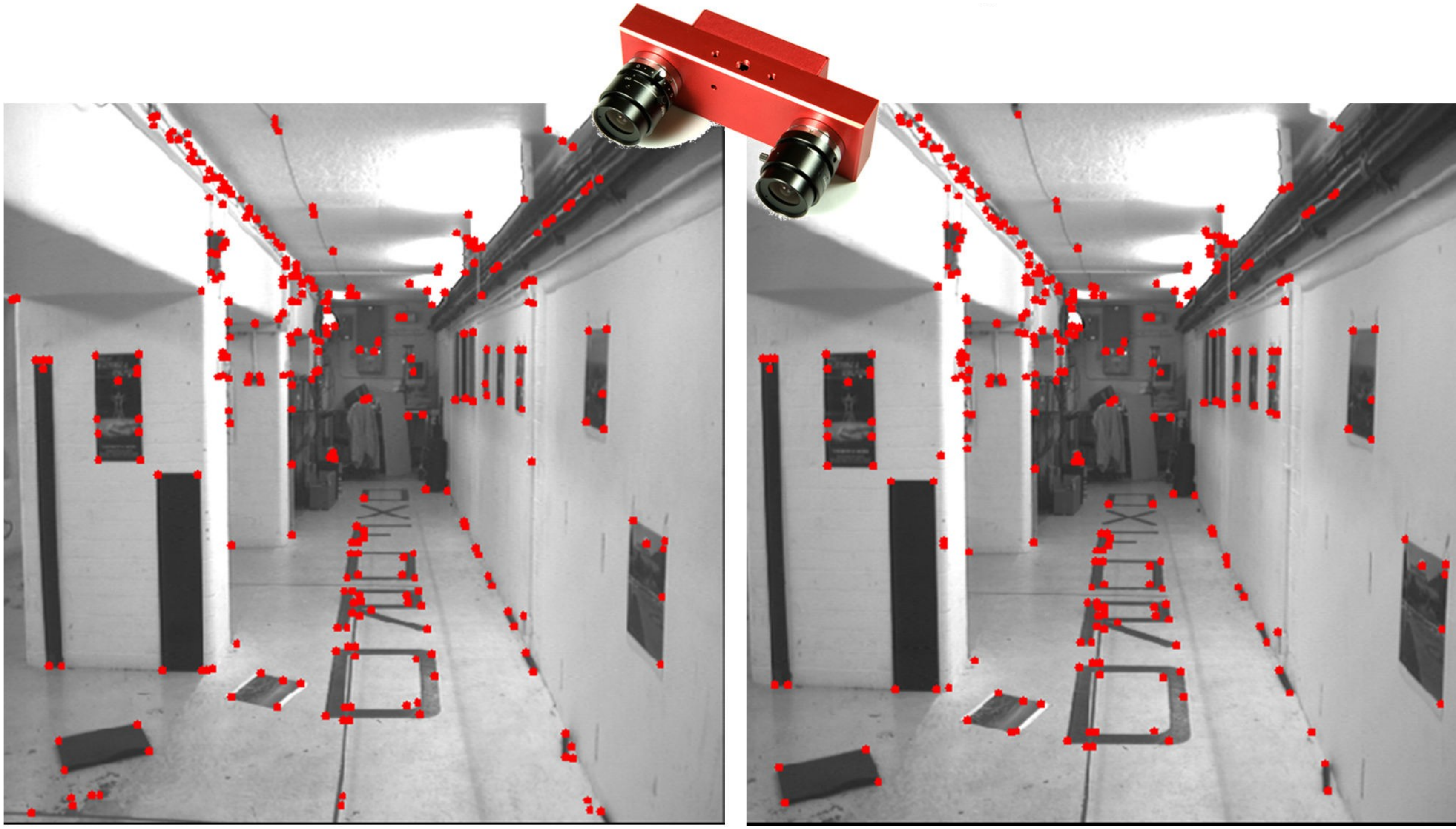
3) Calculează funcția cornerness R la fiecare pixel (x,y)

4) Găsește punctele pentru care R are valori foarte mari (> threshold).

5) Găsește maximele locale ale funcției, realizează suprimarea maximelor (non-maximum supression)

Nu e invariant la scală -> pot rula detectorul la mai multe scale

Rezultate detector Harris



- localizare precisa;
- repetabilitate crescuta;
- aceeași scală

Detectorul Harris

Detectorul Harris detectează colțuri la o singură scală = mărime = scale

Vrem să găsim un detector multi-scale, fiecare trăsătură locală detectată să aibă o scală caracteristică, covariantă cu zoom-ul într-o imagine.

Găsirea scalei specificei conduce și la ajustarea dimensiunii ferestrei din jurul punctului detectat ca trăsătură

Slide adaptat după K. Graumann

Detectarea punctelor de interes invariante la scală

Cum putem detecta puncte de interes în fiecare imagine independent, astfel încât detecțiile se regăsesc chiar la scale diferite?

Slide adaptat după K. Graumann