

Analiza și Prelucrarea Secvențelor de Imagini

Ș.I.dr.ing. Bogdan IONESCU

București, 2010

> M5. Informația de mișcare

5.1. [Introducere]

5.2. [Detecția mișcării]

5.3. [Estimarea mișcării]

Analiza și Prelucrarea Secvențelor de Imagini, Ș.I. Bogdan IONESCU

1

> M5. Informația de mișcare [Introducere]

Analiza și Prelucrarea Secvențelor de Imagini, Ș.I. Bogdan IONESCU

2

Informația de mișcare

= informația fundamentală a unei secvențe de imagini (imagini în mișcare).

> două direcții de studiu (vezi și MPEG-7):

↳ analiza mișcării globale

- este realizată la nivel de segment video (ansamblu de cadre);
- aplicații: exemple, recunoașterea mișcării camerei video, detecția activității de mișcare sau generarea imaginilor de tip "mozaic".



→ tipul mișcării: mișcare translațională, mișcare de rotație, mișcare de apropiere;

→ informații reținute: amplitudinea mișcării, durata mișcării, localizarea în secvență.

Analiza și Prelucrarea Secvențelor de Imagini, Ș.I. Bogdan IONESCU

3

Informația de mișcare

= informația fundamentală a unei secvențe de imagini (imagini în mișcare).

> două direcții de studiu (vezi și MPEG-7):

↳ analiza mișcării globale (continuare)



→ o măsură a percepției vizuale pe care o avem asupra mișcării conținute în secvență;

→ niveluri de activitate: absență, intensă (ex. secvențe de gol, urmăriri, ...), redusă (ex. dialog, interviu, ...);



← prin recalare geometrică și suprapunere imagini.

Analiza și Prelucrarea Secvențelor de Imagini, Ș.I. Bogdan IONESCU

4

Informația de mișcare

> două direcții de studiu (vezi și MPEG-7):

↳ analiza mișcării locale

- este realizată la nivel de regiuni spațiale de pixeli = obiecte;
- aplicații principale: segmentarea și urmărirea temporală a obiectelor în mișcare.



→ determinarea regiunilor din imagine care se deplasează (sau se schimbă, ~segmentare);

→ analiza traiectoriei acestor obiecte (ex. evoluție centru de greutate, puncte de contur, ...) + evoluție formă în timp (= tracking);

> indiferent de abordare, numitor comun: **estimarea/detecția mișcării !**

Analiza și Prelucrarea Secvențelor de Imagini, Ș.I. Bogdan IONESCU

5

> M5. Informația de mișcare [Detecția mișcării]

Detecția mișcării

> cea mai simplă abordare a conținutului de mișcare,

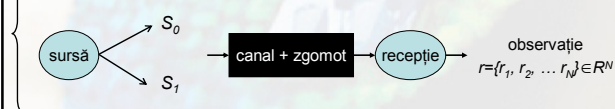
> obiectiv: identificarea pixelilor din imagine (sau a regiunilor) care se schimbă de la imagine la alta,

> mișcarea nu este percepută direct, ci prin intermediul schimbărilor intensității pixelilor din imagine (pot fi produse și de zgomot sau fluctuații de intensitate luminoasă!),

decizie cu prag fix

> criteriul de decizie Bayes:

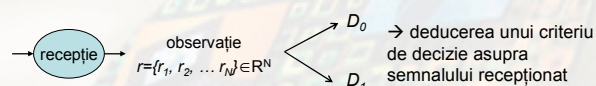
[M. Ciuc, C. Vertan, Prelucrarea Statistică a Semnalelor, 2005]



Detecția mișcării

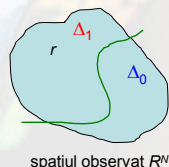
decizie cu prag fix

> criteriul de decizie Bayes (continuare):



= determinarea unui hiperplan ce separă R^N în două clase: Δ_0 și Δ_1 ,

$$\begin{aligned} \Delta_0 \cap \Delta_1 &= \emptyset \\ \Delta_0 \cup \Delta_1 &= R^N \end{aligned} \Rightarrow \begin{aligned} r \in \Delta_0 &\Rightarrow D_0 \\ r \in \Delta_1 &\Rightarrow D_1 \end{aligned}$$



spațiul observat R^N

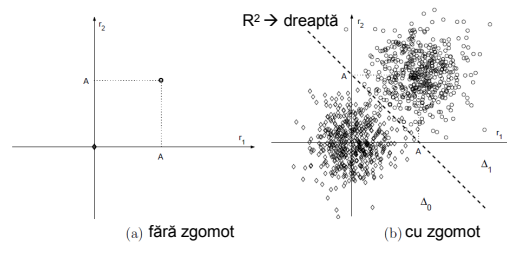
Detecția mișcării

[M. Ciuc, C. Vertan, Prelucrarea Statistică a Semnalelor, 2005]

decizie cu prag fix

> criteriul de decizie Bayes (continuare): exemplu

$$\begin{aligned} S_0 &= S_0(t) = 0, S_1 = S_1(t) = A \quad t \in [0; T] \\ r &= (r_1, r_2) \in R^2, \quad N = 2 \text{ eşantioane,} \end{aligned}$$



Detecția mișcării

decizie cu prag fix

> criteriul de decizie Bayes (continuare):

→ situații posibile:

- | | |
|---|----------|
| 1. am emis S_0 (ipoteza H_0) → am decis D_0 (corect) | C_{00} |
| 2. am emis S_0 (ipoteza H_0) → am decis D_1 (alarmă falsă) | C_{10} |
| 3. am emis S_1 (ipoteza H_1) → am decis D_0 (non detecție) | C_{01} |
| 4. am emis S_1 (ipoteza H_1) → am decis D_1 (corect) | C_{11} |

$$C_{00} < C_{10}, C_{11} < C_{01}$$

→ regula de decizie Bayes este în sensul minimizării costului mediu:

$$\begin{aligned} \bar{C} &= C_{00}P(D_0 \cap S_0) + C_{01}P(D_1 \cap S_0) + \\ &C_{10}P(D_0 \cap S_1) + C_{11}P(D_1 \cap S_1) \end{aligned}$$

Detecția mișcării

decizie cu prag fix

> criteriul de decizie Bayes (continuare):

$$\begin{aligned} \Rightarrow \bar{C} &= C_{00} \cdot P_0 \cdot P(D_0 / S_0) + C_{01} \cdot P_0 \cdot P(D_1 / S_0) + \\ &C_{10} \cdot P_1 \cdot P(D_0 / S_1) + C_{11} \cdot P_1 \cdot P(D_1 / S_1) \end{aligned}$$

unde P_0 este probabilitatea de emisie a simbolului S_0 iar P_1 a simbolului S_1 .

$$\begin{aligned} \Rightarrow \bar{C} &= C_{00} \cdot P_0 \cdot \int_{\Delta_0} w(r / S_0) dr + C_{01} \cdot P_0 \cdot \int_{\Delta_1} w(r / S_0) dr + \\ &C_{10} \cdot P_1 \cdot \int_{\Delta_0} w(r / S_1) dr + C_{11} \cdot P_1 \cdot \int_{\Delta_1} w(r / S_1) dr \end{aligned}$$

unde $w(r/S_0)$ reprezintă densitatea de probabilitate de ordinul N în ipoteza emiterii simbolului S_0 (densit. de prob. a punctelor \diamond anterioare),

Detecția mișcării

decizie cu prag fix

> criteriul de decizie Bayes (continuare):

$$\begin{aligned}\bar{C} &= C_{00} \cdot P_0 \cdot \int_{\Delta_0} \omega(r/S_0) dr + C_{01} \cdot P_0 \cdot \int_{\Delta_1} \omega(r/S_0) dr + \\ &C_{10} \cdot P_1 \cdot \int_{\Delta_0} \omega(r/S_1) dr + C_{11} \cdot P_1 \cdot \int_{\Delta_1} \omega(r/S_1) dr \\ \int_{\Delta_0} \omega(r/S_0) dr + \int_{\Delta_1} \omega(r/S_0) dr &= \int_{R^N} \omega(r/S_0) dr = 1 \\ \Rightarrow \bar{C} &= C_{01} \cdot P_0 + C_{11} \cdot P_1 + \\ &\int_{\Delta_0} [P_1 \cdot (C_{10} - C_{11}) \cdot \omega(r/S_1) - P_0 \cdot (C_{01} - C_{00}) \cdot \omega(r/S_0)] dr\end{aligned}$$

Detecția mișcării

decizie cu prag fix

> criteriul de decizie Bayes (continuare):

$$\bar{C} = C_{01} \cdot P_0 + C_{11} \cdot P_1 + \int_{\Delta_0} [P_1 \cdot (C_{10} - C_{11}) \cdot \omega(r/S_1) - P_0 \cdot (C_{01} - C_{00}) \cdot \omega(r/S_0)] dr$$

→ alegerea intervalului Δ_0 care minimizează costul mediu:

$$\Delta_0 = \{r \in R^N \mid P_1 \cdot (C_{10} - C_{11}) \cdot \omega(r/S_1) - P_0 \cdot (C_{01} - C_{00}) \cdot \omega(r/S_0) < 0\}$$

$$\Delta_1 = \{r \in R^N \mid P_1 \cdot (C_{10} - C_{11}) \cdot \omega(r/S_1) - P_0 \cdot (C_{01} - C_{00}) \cdot \omega(r/S_0) > 0\}$$

Detecția mișcării

decizie cu prag fix

> criteriul de decizie Bayes (continuare):

$$\left. \begin{aligned} r \in \Delta_0 \Rightarrow D_0 &\Rightarrow \frac{\omega(r/S_1)}{\omega(r/S_0)} > \frac{P_0 \cdot (C_{01} - C_{00})}{P_1 \cdot (C_{10} - C_{11})} \\ r \in \Delta_1 \Rightarrow D_1 &\Rightarrow \frac{\omega(r/S_1)}{\omega(r/S_0)} < \frac{P_0 \cdot (C_{01} - C_{00})}{P_1 \cdot (C_{10} - C_{11})} \end{aligned} \right\}$$

$$\Lambda(r) = \frac{\omega(r/S_1)}{\omega(r/S_0)} \begin{cases} > \frac{P_0}{P_1} \cdot g \\ < \frac{P_0}{P_1} \cdot g \end{cases}$$

raport de plauzibilitate pragul testului

Detecția mișcării

decizie cu prag fix

> aplicarea criteriului Bayes la detecția mișcării:
[J. Konrad, Motion Detection and Estimation, 2009]

- ipoteza $H_S (\sim H_0) \rightarrow$ un pixel este staționar,
- ipoteza $H_M (\sim H_1) \rightarrow$ un pixel este în mișcare.

$$I_k(n) = I_{k-1}(n) + w$$

unde $I_k(n)$ reprezintă pixelul de coordonate $n=(x,y)$ la momentul k iar w este zgomotul survenit în imagine.

→ modelare zgomot w :

- H_S : zgomot alb Gaussian de medie 0 și dispersie σ^2 (motiv de faptul că în zonele staționare schimbarea este dată de zgomotul camerei video);
- H_M : zgomot cu distribuție uniformă în $[-L;L]$ (diferențele sunt date de mișcare fiind nepredictibile);

Detecția mișcării

decizie cu prag fix

> aplicarea criteriului Bayes la detecția mișcării (continuare):

→ dacă ceea ce observăm (echivalent vector r) este:

$$\rho_k(n) = I_k(n) - I_{k-1}(n)$$

→ criteriul de decizie Bayes poate fi formulat astfel:

$$\frac{\omega(\rho_k(n)/S_M)}{\omega(\rho_k(n)/S_S)} > \frac{P_S}{P_M} \cdot g$$

$$\frac{\omega(\rho_k(n)/S_M)}{\omega(\rho_k(n)/S_S)} < \frac{P_S}{P_M} \cdot g$$

unde P_S probabilitatea absenței mișcării, P_M este probabilitatea mișcării, D_M și D_S sunt deciziile prezenței mișcării și respectiv absenței iar g depinde de costurile deciziilor.

Detecția mișcării

decizie cu prag fix

> aplicarea criteriului Bayes la detecția mișcării (continuare):

$$\frac{\omega(\rho_k(n)/S_M)}{\omega(\rho_k(n)/S_S)} > \frac{P_S}{P_M} \cdot g \Rightarrow \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{\rho_k(n)^2}{2\sigma^2}} > \frac{P_S}{P_M} \cdot g$$

$$\Rightarrow e^{\frac{\rho_k(n)^2}{2\sigma^2}} > \frac{2 \cdot L \cdot P_S \cdot g}{P_M \cdot \sqrt{2\pi\sigma^2}} \Rightarrow \frac{\rho_k(n)}{2\sigma^2} > \ln \frac{2 \cdot L \cdot P_S \cdot g}{P_M \cdot \sqrt{2\pi\sigma^2}}$$

Detecția mișcării

decizie cu prag fix > aplicarea criteriului Bayes la detecția mișcării (continuare):

$$\Rightarrow \rho_k(n) = \frac{D_M}{D_S} > 2 \cdot \sigma^2 \cdot \ln \frac{2 \cdot L \cdot P_S \cdot \theta}{P_M \cdot \sqrt{2\pi\sigma^2}} = \theta$$

- > **limitări**:
- pentru valori mici ale lui θ se marchează și zgomotul (regiuni mici izolate);
 - pentru valori mari ale lui θ tendința este să fie marcate doar contururile obiectelor cât și zonele mai texturate;

$$\Rightarrow \frac{1}{N} \sum_{m \in W_n} \rho_k(m) > \theta$$

: medierea pixelilor într-o anumită fereastră centrată în punctul $n(W_n)$ (optimală pentru zgomot alb);

Detecția mișcării

decizie cu prag fix > aplicarea criteriului Bayes la detecția mișcării (continuare):

$$\frac{1}{N} \sum_{m \in W_n} \rho_k(m) > \theta$$

> **limitări**: obiecte mari netexturate, tendința de a selecta doar conturul exterior;

[J. Konrad, Motion Detection and Estimation, 2009]



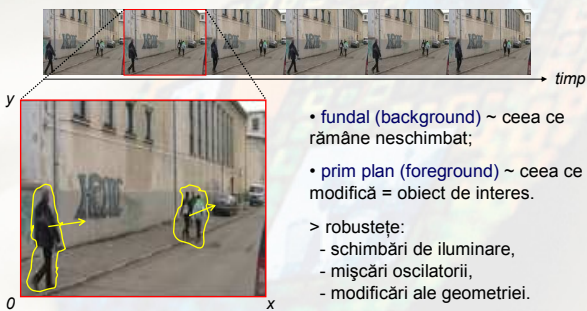
→ ameliorare:

$$\rho_k(n) = I_k(n) - B_k(n)$$

unde $B_k(n)$ este cea ce numim fundalul secveței la momentul k .

Detecția mișcării

estimarea fundalului



- **fundal (background)** ~ ceea ce rămâne neschimbat;
 - **prim plan (foreground)** ~ ceea ce se modifică = obiect de interes.
- > **robustețe**:
- schimbări de iluminare,
 - mișcări oscilatorii,
 - modificări ale geometriei.

Detecția mișcării

estimarea fundalului

↳ **metode nerecursive**

- diferența cadrelor succesive,
- aplicarea principiului filtrului median;
- metode statistice neparametrice, ...

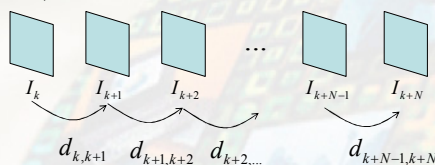
↳ **metode recursive**

- amestec de Gaussiene,
- medie alunecătoare;
- bazate pe filtrul Kalman, ...

Detecția mișcării

estimarea fundalului

> diferența cadrelor succesive:



$B_{k+1} = I_k$ unde B_{k+1} reprezintă fundalul estimat la momentul $k+1$;

$$d_{k,k+1} = |I_{k+1} - B_{k+1}|$$

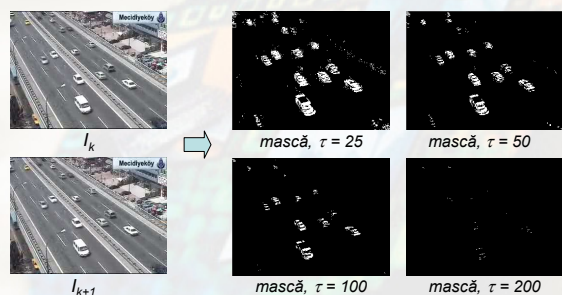
> dacă $d_{k,k+1}(n) > \tau \Rightarrow$ pixelul $n(x,y)$ face parte din obiect.

Detecția mișcării

[Tamersoy 09]

estimarea fundalului

> diferența cadrelor succesive, exemple:



Detecția mișcării

estimarea fundalului

> filtrul medie:

$$B_k(n) = \frac{1}{N} \cdot \sum_{l=0}^{N-1} I_{k-l}(n)$$

unde B_k reprezintă fundalul estimat la momentul k , $n=(x,y)$ coordonatele pixelului curent iar N este numărul de imagini considerate (fereastra temporală),

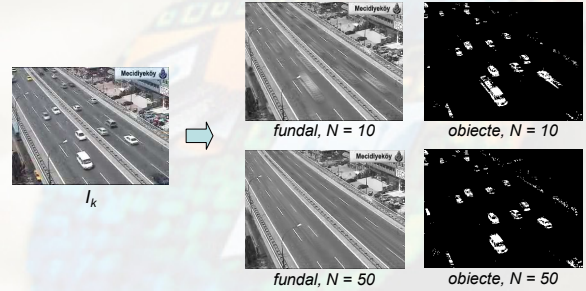
$$\rightarrow d_k = |I_k - B_k|$$

> dacă $d_k(n) > \tau \Rightarrow$ pixelul $n=(x,y)$ face parte din obiect.

Detecția mișcării

estimarea fundalului

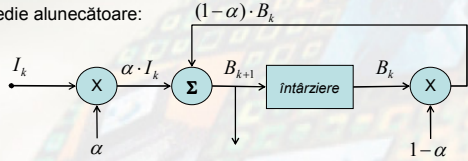
> filtrul medie, exemple:



Detecția mișcării

estimarea fundalului

> medie alunecătoare:



$$B_{k+1} = \alpha \cdot I_k + (1 - \alpha) \cdot B_k$$

unde B_{k+1} reprezintă fundalul estimat la momentul $k+1$ iar α este un parametru (rată de învățare, ex. $\alpha=0.05$);

$$\rightarrow d_{k+1} = |I_{k+1} - B_{k+1}|$$

> dacă $d_{k+1}(n) > \tau \Rightarrow$ pixelul $n=(x,y)$ face parte din obiect.

Detecția mișcării

estimarea fundalului

> filtrul median:

$$B_k(n) = \text{median}_{l \in \{0, \dots, N-1\}} \{I_{k-l}(n)\}$$

unde B_k reprezintă fundalul estimat la momentul k , $n=(x,y)$ coordonatele pixelului curent, $\text{median}\{\}$ este operatorul ce returnează valoarea mediană a unui vector iar N reprezintă numărul de imagini considerate (fereastra temporală);

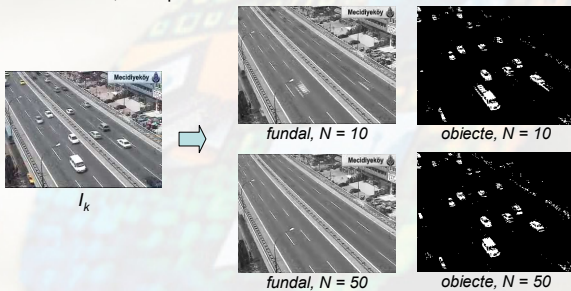
$$\rightarrow d_k = |I_k - B_k|$$

> dacă $d_k(n) > \tau \Rightarrow$ pixelul $n=(x,y)$ face parte din obiect.

Detecția mișcării

estimarea fundalului

> filtrul median, exemple:



Detecția mișcării

estimarea fundalului

> aproximarea filtrului median:

$$B_k(n) = B_{k-1}(n) + \text{sgn}(I_k(n) - B_{k-1}(n))$$

unde B_k reprezintă fundalul estimat la momentul k iar $\text{sgn}()$ este funcția signum:

$$\text{sgn}(x) = \begin{cases} -1 & x < 0 \\ 0 & x = 0 \\ 1 & x > 0 \end{cases} \quad \begin{aligned} & \text{- dacă } I_k(n) > B_{k-1}(n) \\ & \Rightarrow \text{pixelul de fundal în } n, +1; \\ & \text{- dacă } I_k(n) < B_{k-1}(n) \\ & \Rightarrow \text{pixelul de fundal în } n, -1; \end{aligned}$$

$$\rightarrow d_k = |I_k - B_k|$$

> dacă $d_k(n) > \tau \Rightarrow$ pixelul $n=(x,y)$ face parte din obiect.

> ~ converge către un filtru median.

Detecția mișcării

estimarea fundalului

> estimarea fundalului, exemple comparative:



diferența cadrelor consecutive

media alunecătoare

aproximarea filtrului median

[R. Dumitrescu, D. Grama, B. Ionescu, 2010]

Detecția mișcării

decizie cu prag fix

> estimarea probabilității de distribuție a fundalului:
[J. Konrad, Motion Detection and Estimation, 2009]

$$\Gamma_S(I_k(n)) = \frac{1}{N} \sum_{i=1}^N \Gamma(I_k(n) - I_{k-i}(n))$$

unde Γ este o distribuție Gaussiană de medie 0 și dispersie σ^2 care este considerată a fi constantă pe parcursul secvenței iar N este fereastra temporală.

→ presupunând o distribuție uniformă a intensităților pixelilor de mișcare în intervalul $[-L; L]$, atunci decizia devine:

$$\Rightarrow \Gamma_S(I_k(n)) \stackrel{D_S}{>} \frac{P_M}{2 \cdot L \cdot g \cdot P_S} = \theta \quad \text{decizia are loc în spațiul probabilităților,}$$

Detecția mișcării

[J. Konrad, Motion Detection and Estimation, 2009]

decizie cu prag fix

> estimarea probabilității de distribuție a fundalului (continuare):

$$\Rightarrow \Gamma_S(I_k(n)) \stackrel{D_S}{>} \frac{P_M}{2 \cdot L \cdot g \cdot P_S} = \theta$$

→ pixelul la locația n este considerat staționar dacă este probabil ca aceasta să fie extras din $\Gamma_S()$,

→ $\Gamma_S()$ este bazat pe N cadre ceea ce îi permite să se adapteze la schimbări lente de fundal (variații de iluminare, zi-noapte),



Detecția mișcării

decizie cu prag adaptiv

> metodele anterioare plecau de la ipoteza că nu există informații despre distribuția pixelilor de mișcare (considerați uniform distribuiți);

> *ipoteza*: pe măsură ce un obiect se deplasează în timp, intensitățile $I_k(n)$, $I_{k+1}(n)$, ... sunt de fapt valori ce aparțin acestui obiect;

→ putem folosi **istoricul spațial** în locul celui temporal (ergodicitate în domeniul spațial);

→ model local-în-timp \approx model local-în-spațiu;

→ astfel, la un moment dat, trebuie identificați toți pixelii obiectului ce se deplasează din care face parte pixelul curent n ;

Detecția mișcării

[J. Konrad, Motion Detection and Estimation, 2009]

decizie cu prag adaptiv

> dacă notăm cu $e_k(n) = \begin{cases} S \text{ (staționar)} \\ M \text{ (mișcare)} \end{cases}$ o decizie la momentul k pentru pixelul $n=(x,y)$;

$$a. e_k^0(n) \leftarrow \Gamma_S(I_k(n)) \stackrel{D_S}{>} \frac{P_M}{2 \cdot L \cdot g \cdot P_S} = \theta$$

> iterația 0 = o mască inițială de detecție;

$$b. set_i(n) = \{m \in N(n) / e_k^{i-1}(m) = M\}$$

> pornind de la n se identifică pixelii vecini ce aparțin obiectului în mișcare din care n face parte, unde i este iterația curentă, $N(n)$ este o vecinătate mică iar M = decizia de mișcare.

Detecția mișcării

decizie cu prag adaptiv

> probabilitatea de mișcare a pixelului este calculată ca fiind:

$$P_{M_i}(I_k(n)) = \frac{1}{|set_i(n)|} \sum_{m \in set_i(n)} \Gamma(I_k(n) - I_k(m))$$

unde Γ este o distribuție Gaussiană de medie 0 și dispersie σ^2 iar $set_i(n)$ este setul de pixeli determinat anterior.

> decizia devine:

$$\Rightarrow \frac{P_S(I_k(n))}{P_{M_i}(I_k(n))} \stackrel{D_S}{>} \frac{P_M}{g \cdot P_S} = \theta \quad \text{ce produce decizia} \rightarrow e_k^i(n)$$

Detekția mișcării

[J. Konrad, Motion Detection and Estimation, 2009]

decizie cu prag adaptiv

> decizia devine:

$$\Rightarrow \frac{P_S(I_k(n))}{P_{M_i}(I_k(n))} \stackrel{D_S}{>} \frac{P_M}{D_M} = \theta \text{ ce produce decizia } \rightarrow e_k^i(n)$$

c. $e_k^i(n)$ devine masca inițială și se determină în continuare noul $set_{i+1}(n)$ și noua probabilitate $P_{M_{i+1}}$ (se reia pasul b.)

> dacă la un moment dat $set_{i+1}(n)=0$ atunci se poate folosi ipoteza că P_M are o distribuție uniformă.

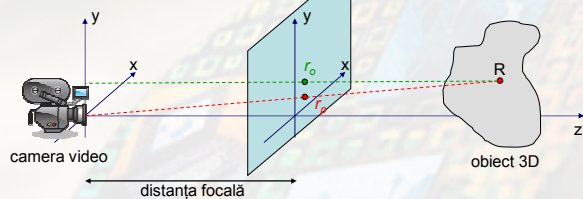


> M5. Informația de mișcare [Estimarea mișcării]

Estimarea mișcării

[Marichal 98]

> punerea problemei:

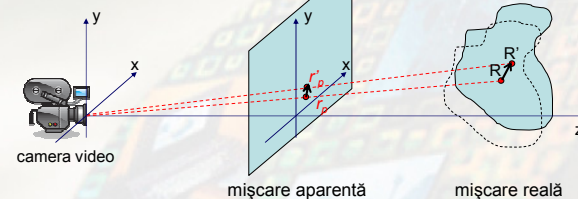


> formarea imaginii:

- **proiecție de perspectivă:** imaginea r_p a punctului R este obținută prin raza ce unește R cu lentila camerei;
- **proiecție ortogonală:** imaginea r_o a punctului R este obținută prin raza perpendiculară pe planul imaginii (~~distanță~~).

Estimarea mișcării

> punerea problemei (continuare):



- > scena 3D este proiectată în imagini plane 2D;
- > informația 3D de mișcare este de asemenea proiectată în plan;
- > în imagine mișcarea se manifestă prin schimbări ale intensității pixelilor de-a lungul timpului \Rightarrow estimarea este bazată pe intensitate.

Estimarea mișcării

> principiul estimării mișcării: estimarea deplasării unui pixel sau bloc de pixeli între cadre succesive:

A. minimizarea variației intensității

$$DFD(\vec{r}, \vec{d}, \Delta t) = I(\vec{r} + \vec{d}, t + \Delta t) - I(\vec{r}, t)$$

unde \vec{r} reprezintă poziția pixelului/blocului de pixeli în imaginea curentă $I(t)$, \vec{d} reprezintă vectorul de deplasare între momentele t și $t + \Delta t$ exprimat în funcție de deplasarea pe cele două axe, oX și respectiv oY :

DFD = Displaced Frame Difference

$$\vec{d} = (dx, dy), \vec{r} = (x, y)$$

\Rightarrow exemplu estimare ...

Estimarea mișcării



["The Wicker Man", Copyright 2006 Warner Bros Pictures]

Estimarea mișcării

$$DFD(\vec{r}, \vec{d}, \Delta t) = I(\vec{r} + \vec{d}, t + \Delta t) - I(\vec{r}, t)$$

B. flux optic

> dacă considerăm imaginile ca fiind funcții continue, atunci putem aplica o descompunere în serie Taylor (ordin 1), \Rightarrow

$$I(\vec{r} + \vec{d}, t + \Delta t) = I(\vec{r}, t) + \frac{\partial I(\vec{r}, t)}{\partial x} \cdot dx + \frac{\partial I(\vec{r}, t)}{\partial y} \cdot dy + \frac{\partial I(\vec{r}, t)}{\partial t} \cdot dt + \dots$$

> înlocuind în ecuația anterioară, rezultă:

$$DFD(\vec{r}, \vec{d}, \Delta t) = \frac{\partial I(\vec{r}, t)}{\partial x} \cdot dx + \frac{\partial I(\vec{r}, t)}{\partial y} \cdot dy + \frac{\partial I(\vec{r}, t)}{\partial t} \cdot dt$$

Estimarea mișcării

B. flux optic (continuare)

$$DFD(\vec{r}, \vec{d}, \Delta t) = \frac{\partial I(\vec{r}, t)}{\partial x} \cdot dx + \frac{\partial I(\vec{r}, t)}{\partial y} \cdot dy + \frac{\partial I(\vec{r}, t)}{\partial t} \cdot dt$$

> minimizând DFD obținem:

$$\frac{\partial I(\vec{r}, t)}{\partial x} \cdot u + \frac{\partial I(\vec{r}, t)}{\partial y} \cdot v + \frac{\partial I(\vec{r}, t)}{\partial t} = 0 \quad \text{ecuația fluxului optic}$$

unde u și v definesc vectorul de deplasare din imagine:

$$(u, v) = \left(\frac{dx}{dt}, \frac{dy}{dt} \right)$$

Estimarea mișcării

B. flux optic (continuare)

$$\frac{\partial I(\vec{r}, t)}{\partial x} \cdot u + \frac{\partial I(\vec{r}, t)}{\partial y} \cdot v + \frac{\partial I(\vec{r}, t)}{\partial t} = 0$$

flux optic = imaginea în care valoarea fiecărui pixel reprezintă estimarea proiecției vitezei de translație ce corespunde unui punct de pe suprafața obiectului ce se află în mișcare relativă în câmpul vizual al camerei video.

> mișcare este estimată doar pe direcția gradientului spațial.

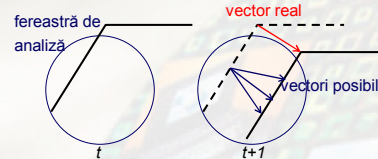
> este posibil ca fluxul optic să nu poată fi estimat în toate punctele.

Estimarea mișcării

[Marichal 98]

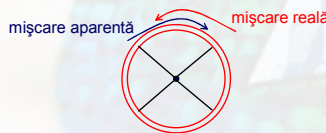
> principalele limitări:

↳ problema estimării într-o fereastră (apertură):



→ fluxul optic nu este unic determinat de informația locală din imaginea ce se schimbă;

↳ problema corespondenței:



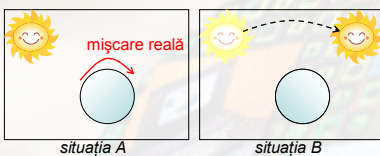
→ dacă viteza unghiulară a roții $> \frac{\pi}{nr.spite} \cdot cadre / s$ roata se va roti aparent invers;

Estimarea mișcării

[Marichal 98]

> principalele limitări (continuare):

↳ sensibilitate la fluctuații de intensitate:



→ zgomot, schimbare intensitate = mișcare;
A. câmp mișcare nul vs. în realitate mișcare;
B. câmp mișcare vs. în realitate inexistent;

↳ suprapunere obiecte:

> apariția/dispariția precum și suprapunerea obiectelor conduce la crearea de regiuni unde intensitatea observată la momentul t nu are corespondență în imaginea la momentul $t+1$.

← absență informație de mișcare !



Estimarea mișcării

> pentru a reduce erorile implicite de estimare sunt adoptate o serie de ipoteze inițiale, exemplu:

- ↳ pixelii blocurilor/regiunilor din imagine au același tip de mișcare de translație de la o imagine la alta;
- ↳ funcția DFD trebuie să aibă o evoluție monoton crescătoare;
- ↳ intensitatea pixelilor din imagine este constantă cu mișcarea;
- ↳ mișcarea este considerată a fi constantă pentru mici volume spațio-temporale;

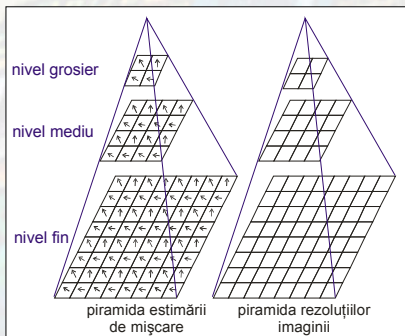
> în funcție de metodă pot fi adoptate una, mai multe sau toate ipotezele anterioare.

Estimarea mișcării

> În funcție de nivelul de detaliu și de precizia câmpului vectorial de mișcare (ansamblul vectorilor de mișcare), întâlnim două tipuri de implementări:

1. multi-rezoluție

→ estimarea mișcării se face pentru mai multe reprezentări, de rezoluții diferite, ale aceleiași imagini;

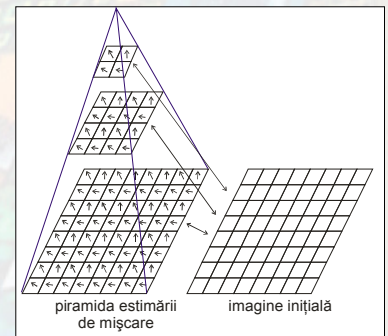


Estimarea mișcării

> două tipuri de implementări (continuare):

2. multi-scală

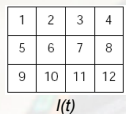
→ folosește imaginile în rezoluția inițială, dar furnizează mai multe niveluri de detaliu pentru vectorii de mișcare;



Estimarea mișcării

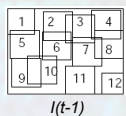
> din punct de vedere temporal, întâlnim:

↳ estimare de tip "forward"



> noua poziție este căutată în imaginea următoare;

↳ estimare de tip "backward"



> noua poziție este căutată în imaginea anterioară;

Estimarea mișcării

> tehnici de estimare a mișcării:

↳ metode diferențiale (bazate pe flux optic)

→ câmp de mișcare dens, sensibil la zgomot, complexitate de calcul importantă;

↳ metode parametrice

→ estimarea parametrilor unui anumit model de mișcare;

↳ metode stohastice

→ explorarea spațiului parametrilor este ghidată în acest caz de procese aleatoare (ex. modele Bayesiene, Markov, ...);

↳ metode bazate pe blocuri

→ estimare la nivel de bloc, compromis complexitate de calcul – precizie estimare, adoptate de MPEG;

Estimarea mișcării

1. metode diferențiale

> bazate pe calculul gradientului (ecuație flux optic):

$$\frac{\partial I(\vec{r}, t)}{\partial t} = - \frac{\partial I(\vec{r}, t)}{\partial x} \cdot u - \frac{\partial I(\vec{r}, t)}{\partial y} \cdot v \rightarrow u \text{ și } v \text{ sunt deplasările}$$

diferența dintre cadre succesive

componentele orizontale și verticale ale gradientului

↳ metode directe

→ principiu: anularea valorilor gradientului;

↳ metode indirecte

→ principiu: convergența funcției DFD către o soluție în direcția gradientului;

Estimarea mișcării

1. metode diferențiale

↳ metode directe: metode iterative

• [Horn 81] pentru a rezolva ecuația fluxului optic adoptă o ipoteză suplimentară: modulul gradientului să aibă valori mici, condiție valabilă în realitate doar pentru micile deplasări ale pixelilor din imagine;

→ problema estimării = problema minimizării funcției de cost:

$$\iint [(I_x \cdot u + I_y \cdot v + I_t)^2 + \lambda \cdot (u_x^2 + u_y^2 + v_x^2 + v_y^2)] dx \cdot dy$$

unde I_a reprezintă derivata parțială a I după componenta a , u_x , u_y , v_x și respectiv v_y reprezintă derivatele parțiale de ordinul întâi ale celor două componente (u , v) ale fluxului optic iar λ este multiplicatorul Lagrange.

Estimarea mișcării

I. metode diferențiale

↳ metode directe: metode iterative (continuare)

$$\iint [(I_x \cdot u + I_y \cdot v + I_t)^2 + \lambda \cdot (u_x^2 + u_y^2 + v_x^2 + v_y^2)] dx \cdot dy$$

→ o posibilă soluție sunt:

$$\hat{u} = u_m - I_x \cdot \frac{P}{D}, \quad \hat{v} = v_m - I_y \cdot \frac{P}{D}$$

unde u_m și v_m sunt valorile medii locale ale lui u și v , iar P și D sunt doi parametri definiți astfel:

$$P = I_x \cdot u_m + I_y \cdot v_m + I_t, \quad D = \lambda + I_x^2 + I_y^2$$

Estimarea mișcării

I. metode diferențiale

↳ metode directe: metode iterative (continuare)

> o posibilă soluție nerecursivă:

$$\frac{\partial I(x, y, t)}{\partial x} \approx \frac{1}{4} \left(I(x, y+1, t) - I(x, y, t) + I(x+1, y+1, t) - I(x+1, y, t) + \right. \\ \left. I(x, y+1, t+1) - I(x, y, t+1) + I(x+1, y+1, t+1) - I(x+1, y, t+1) \right)$$

$$\frac{\partial I(x, y, t)}{\partial y} \approx \frac{1}{4} \left(I(x, y+1, t) - I(x+1, y+1, t) + I(x, y, t) - I(x+1, y, t) + \right. \\ \left. I(x, y+1, t+1) - I(x+1, y+1, t+1) + I(x, y, t+1) - I(x+1, y, t+1) \right)$$

$$\frac{\partial I(x, y, t)}{\partial t} \approx \frac{1}{4} \left(I(x, y, t+1) - I(x, y, t) + I(x+1, y, t+1) - I(x+1, y, t) + \right. \\ \left. I(x, y+1, t+1) - I(x, y+1, t) + I(x+1, y+1, t+1) - I(x+1, y+1, t) \right)$$

Estimarea mișcării

I. metode diferențiale

↳ metode directe: metode iterative (continuare)

→ soluția este calculată iterativ:

$$(\hat{u}_i, \hat{v}_i) = f\{(\hat{u}_{i-1}, \hat{v}_{i-1})\}$$

unde i reprezintă iterația curentă; soluția adoptată fiind soluția furnizată de iterația pentru care este îndeplinit un anumit criteriu de convergență.

- câmp vectorial dens (fiecare pixel are asociat un vector de mișcare);
- complexitate de calcul ridicată (nu este folosit la codare → folosit la analiză);
- deplasări importante ale pixelilor ?

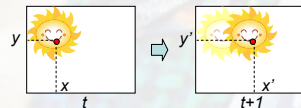
Estimarea mișcării

II. metode parametrice

> deplasarea pixelilor din imagine este modelată pe baza unui anumit set de parametri (model).

→ estimare = determinare parametri.

> ipoteză geometrică: obiectele din imagine sunt suprafețe plane și rigide.



$$\begin{cases} x' = x + u \\ y' = y + v \end{cases}$$

$$\begin{aligned} u &= u(x, y) \\ v &= v(x, y) \end{aligned} = \text{componentele de mișcare}$$

Estimarea mișcării

II. metode parametrice

> având la dispoziție un vector de parametri de model, p , mișcarea poate fi exprimată astfel:

$$\begin{aligned} u &= f(x, y; p) \\ v &= g(x, y; p) \end{aligned} \quad \text{unde } f \text{ și } g \text{ determină modelul de mișcare;}$$

• modelul translațional (cel mai simplu):

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

unde (x, y) reprezintă coordonatele pixelului curent, (x', y') noua sa poziție iar t_x și t_y deplasările pe cele două axe.

Estimarea mișcării

II. metode parametrice

• model cu 3 parametri:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = C \cdot \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

unde C este un parametru de scală suplimentar.

• model cu 4 parametri:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} C_x & 0 \\ 0 & C_y \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

unde C_x și C_y reprezintă factorii de scală pe cele două axe.

Estimarea mișcării

II. metode parametrice

• model de rotație:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

unde factorul de scală a fost înlocuit cu un factor de rotație cu un anumit unghi θ .

• model cu 5 parametri (combinația ultimelor două modele):

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \cdot \begin{bmatrix} C_x & 0 \\ 0 & C_y \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

unde C_x și C_y reprezintă factorii de scală pe cele două axe.

Estimarea mișcării

II. metode parametrice

• model afin (se separă rotația pe cele două axe):

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} C_x \cdot \cos \theta_x & -C_y \cdot \sin \theta_y \\ C_x \cdot \sin \theta_x & C_y \cdot \cos \theta_y \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

unde θ_x și θ_y reprezintă rotațiile pe cele două axe.

→ rezultatul proiecției ortogonale a mișcării pe suprafața imaginii.

• model cu 8 parametri:

$$x' = \frac{a_1 + a_2 \cdot x + a_3 \cdot y}{1 + a_7 \cdot x + a_8 \cdot y}, \quad y' = \frac{a_4 + a_5 \cdot x + a_6 \cdot y}{1 + a_7 \cdot x + a_8 \cdot y}$$

→ rezultatul proiecției de perspectivă.

Estimarea mișcării

II. metode parametrice

• model al transformării biliniare:

$$x' = a_1 \cdot x + a_2 \cdot y + a_3 \cdot x \cdot y + a_4$$

$$y' = a_5 \cdot x + a_6 \cdot y + a_7 \cdot x \cdot y + a_8$$

• modele ce iau în calcul accelerația mișcării, [Sanson 81]:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a_x^x & a_x^y \\ a_y^x & a_y^y \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} b_x^x & b_x^{xy} & b_x^{y^2} \\ b_y^x & b_y^{xy} & b_y^{y^2} \end{bmatrix} \cdot \begin{bmatrix} x^2 \\ xy \\ y^2 \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

unde a și b reprezintă seturile de parametri folosiți la modelare.

Estimarea mișcării

$$u = f(x, y; p)$$

$$v = g(x, y; p)$$

II. metode parametrice

> estimarea mișcării este realizată prin introducerea în ecuația fluxului optic a modelului de mișcare și estimarea parametrilor prin minimizarea unei funcții de cost:

$$p = \arg \min_p \Psi(I_x \cdot f(x, y; p) + I_y \cdot g(x, y; p) + I_t)$$

unde p reprezintă modelul parametric, Ψ este o funcție de cost iar I_a reprezintă derivata lui I după componenta a .

• Motion2D: bibliotecă de funcții C pentru estimarea parametrică a mișcării [IRISA – INRIA <http://www.irisa.fr/vista/>],

Estimarea mișcării

III. metode bazate pe blocuri

> estimarea mișcării este realizată pe baza diferențelor dintre blocuri de pixeli;

→ un vector de mișcare pentru fiecare bloc;

→ cel mai bun compromis între complexitatea de calcul și precizia estimării (adoptată standarde de compresie MPEG):

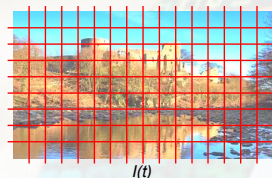
↳ **sensibilitate ridicată:** blocuri de pixeli de dimensiuni reduse, utilă când este nevoie de precizie (analiza deplasărilor fine) dar sensibilă la zgomot;

↳ **robustețe ridicată:** blocuri de dimensiuni mai mari ⇒ vectorii de mișcare reprezintă o aproximare mai grosieră a fluxului optic, utilă pentru o analiză globală, complexitate 7.

Estimarea mișcării

III. metode bazate pe blocuri

> principiul estimării:



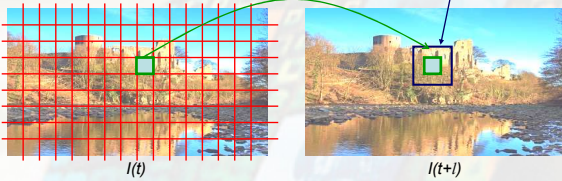
• imaginea curentă analizată la momentul t , $I(t)$, este împărțită în blocuri disjuncte de pixeli de dimensiune $B \times B$;

→ B este ales ca 2^x , $x=1,2,\dots$ (optimizare hardware).

Estimarea mișcării

III. metode bazate pe blocuri

> principiul estimării (continuare):

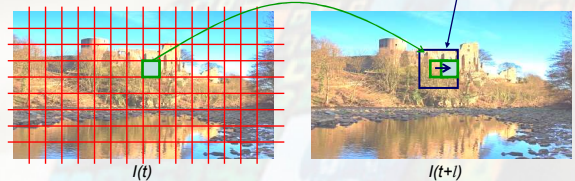


- pentru fiecare bloc din $I(t)$ se caută noua sa poziție în imaginea următoare la momentul $t+l$ (l = pasul de analiză);
- reducere complexitate, căutarea se face doar într-o fereastră S de dimensiune $(2B+1) \times (2B+1)$.

Estimarea mișcării

III. metode bazate pe blocuri

> principiul estimării (continuare):



- noua poziție a blocului curent este determinată pe baza minimizării unei funcții de cost, F_C , ce estimează eroarea de aproximare a blocului curent cu blocurile din fereastra S .

Estimarea mișcării

III. metode bazate pe blocuri

> principiul estimării (continuare):

$$\vec{d}_{\min} = \arg \min_{\vec{d} \in S} F_C(I(\vec{r} + \vec{d}, t+l), I(\vec{r}, t))$$

unde \vec{d}_{\min} reprezintă deplasarea blocului curent $I(\vec{r}, t)$ pentru care funcția F_C este minimă, iar valorile lui \vec{d} sunt toate deplasările posibile ale blocului de comparare în interiorul ferestrei S .

> **căutare completă** (full search) = compararea se face cu toate blocurile posibile din fereastra S ;

⇒ complexitate de calcul maximă vs. precizie maximă.

Estimarea mișcării

III. metode bazate pe blocuri

> funcții de cost uzuale (F_C):

↳ Mean Absolute Difference (MAD)

$$MAD(b_1, b_2) = \frac{1}{X \cdot Y} \sum_{x=1}^X \sum_{y=1}^Y |b_1(x, y) - b_2(x, y)|$$

unde b_1 și b_2 reprezintă două blocuri de pixeli iar X și Y dimensiunile imaginii.

↳ Mean Square Difference (MSD)

$$MSD(b_1, b_2) = \frac{1}{X \cdot Y} \sum_{x=1}^X \sum_{y=1}^Y [b_1(x, y) - b_2(x, y)]^2$$

Estimarea mișcării

III. metode bazate pe blocuri

> funcții de cost uzuale (F_C , continuare):

↳ Pel Difference Classification (PDC)

$$PDC(b_1, b_2) = \sum_{x=1}^X \sum_{y=1}^Y \text{ord}\{|b_1(x, y) - b_2(x, y)| \leq \tau\}$$

unde $\text{ord}(P)$ returnează valoarea 1 dacă propoziția P este adevărată și 0 altfel, iar τ este un prag ales arbitrar.

↳ Integral Projection (IP)

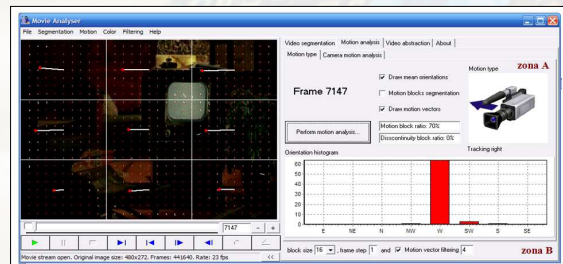
$$IP(b_1, b_2) = \sum_{x=1}^X \left| \sum_{y=1}^Y b_1(x, y) - \sum_{y=1}^Y b_2(x, y) \right| + \sum_{y=1}^Y \left| \sum_{x=1}^X b_1(x, y) - \sum_{x=1}^X b_2(x, y) \right|$$

Estimarea mișcării

III. metode bazate pe blocuri

> demonstrație:

- HDUncomp: frame 17;
 - The Holiday frame 31623;
- frame 45796;
frame 109038;
frame 1;
frame 7636.



Estimarea mișcării

III. metode bazate pe blocuri

> *observații practice:* în general în urma minimizării funcției de cost apar următoarele situații:

- valoarea minimală este obținută pentru blocul de pixeli din imaginea la momentul $t+1$ pe aceeași poziție cu blocul curent,
→ vector nul = **absența mișcării**;
- valoarea minimală a funcției de cost este foarte mare ($> \tau_{disc}$),
→ vector inexistent = **discontinuitatea mișcării**;
- valoarea minimală este nenulă și $< \tau_{disc}$,
→ **mișcare**.

Estimarea mișcării

III. metode bazate pe blocuri

> complexitatea de calcul depinde de:

- modul de căutare în fereastra S ,
- dimensiunea ferestrei S ,
- funcția de cost folosită, etc.

> metodele existente diferă în funcție de modul de căutare a noii poziții a blocului curent în imaginea următoare:

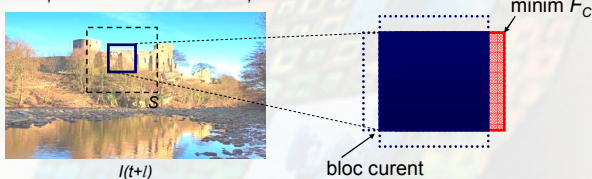
- căutare în 3 pași,
- căutare intercalată,
- căutare logaritmică,
- căutare ierarhică,
- căutare binară,
- căutare hibridă,
- căutare ortogonală,
- ...



Estimarea mișcării

III. metode bazate pe blocuri

> **căutarea în 3 pași (varianta H.263+):** noua poziție este căutată în direcția minimului local al funcției de cost:



- I**
- funcția de cost este estimată între blocul curent (la t) și cele patru blocuri vecine la distanță de 1 pixel (la $t+1$): N, E, S și V,
 - D_1 = valoarea minimă F_C ; blocul respectiv devine bloc curent în imaginea la $t+1$;

Estimarea mișcării

III. metode bazate pe blocuri

> **căutarea în 3 pași (continuare):**



- II**
- se repetă procesul pentru noul bloc curent (la $t+1$) și vecinii acestuia: N, E, S și V,
 - $D_2 \leftarrow D_1$; D_1 = noua valoare minimă F_C ; blocul respectiv devine bloc curent în imaginea la $t+1$;

Estimarea mișcării

III. metode bazate pe blocuri

> **căutarea în 3 pași (continuare):**



- III**
- se repetă procesul pentru noul bloc curent (la $t+1$),
 - $D_3 \leftarrow D_2$; $D_2 \leftarrow D_1$; D_1 = noua valoare minimă F_C ;
 - dacă $D_3 \leq D_1$ și $D_2 \leq D_1 \rightarrow$ STOP
altfel blocul ce a furnizat F_C minim devine bloc curent la $t+1$;

Estimarea mișcării

III. metode bazate pe blocuri

> **căutarea în 3 pași (continuare):**

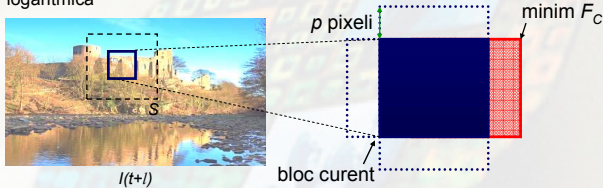


- IV**
- se repetă procesul pentru noul bloc curent (la $t+1$),
 - $D_3 \leftarrow D_2$; $D_2 \leftarrow D_1$; D_1 = noua valoare minimă F_C ;
 - dacă D_1 , D_2 și D_3 nu satisfac condiția anterioară \rightarrow repetă **IV**,
dacă condiția este satisfăcută \rightarrow vector de mișcare.

Estimarea mișcării

III. metode bazate pe blocuri

> căutarea logaritmică (mai multe variante): complexitatea de calcul este logaritmică



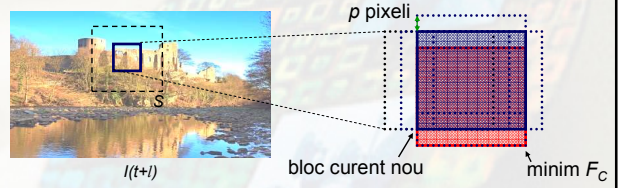
it.1

- funcția de cost este estimată între blocul curent (la t) și cele patru blocuri vecine la distanță de p pixeli (la $t+1$): N, E, S și V,
- blocul pentru care F_C este minim devine bloc curent în imaginea la $t+1$; $p \leftarrow p/2$,

Estimarea mișcării

III. metode bazate pe blocuri

> căutarea logaritmică (continuare):



it.2

- funcția de cost este estimată între blocul curent (la t) și cele patru blocuri vecine la distanță de p pixeli (la $t+1$): N, E, S și V,
- blocul pentru care F_C este minim devine bloc curent în imaginea la $t+1$;

Estimarea mișcării

III. metode bazate pe blocuri

> căutarea logaritmică (continuare):



it.n

- funcția de cost este estimată între blocul curent (la t) și cele patru blocuri vecine la distanță de p pixeli (la $t+1$): N, E, S și V,
- $p == 1$ STOP; blocul pentru care F_C este minim determină vectorul de mișcare.

Estimarea mișcării

III. metode bazate pe blocuri

> căutarea intercalată: similară cu cea logaritmică, căutare în X:

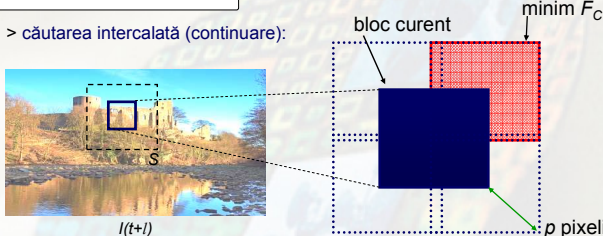


- funcția de cost este estimată între blocul curent (la t) și blocul omolog (la $t+1$),
- dacă $F_C < \tau$ atunci STOP,

Estimarea mișcării

III. metode bazate pe blocuri

> căutarea intercalată (continuare):



- altfel, funcția de cost este estimată între blocul curent (la t) și patru blocuri vecine în X la distanță de p pixeli (la $t+1$),
- blocul pentru care F_C este minim devine bloc curent în imaginea la $t+1$;

Estimarea mișcării

III. metode bazate pe blocuri

> căutarea intercalată (continuare):

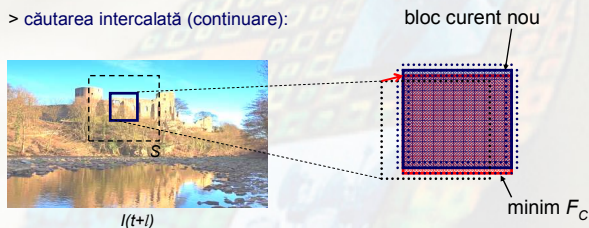


- dacă $p > 1$ atunci $p \leftarrow p/2$;
- se repetă pasul anterior → blocul pentru care F_C este minim devine bloc curent în imaginea la $t+1$;

Estimarea mișcării

III. metode bazate pe blocuri

> căutarea intercalată (continuare):

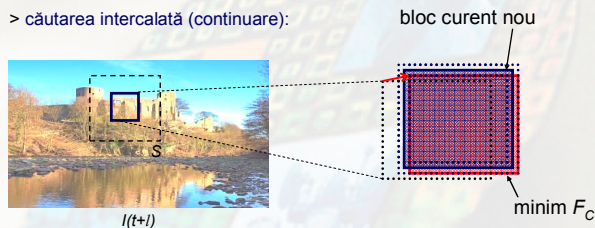


- dacă $p=1$ și F_C minim este obținut pentru vecinul din stânga jos sau dreapta sus al ultimului bloc curent
→ se re-evaluează F_C în +;

Estimarea mișcării

III. metode bazate pe blocuri

> căutarea intercalată (continuare):

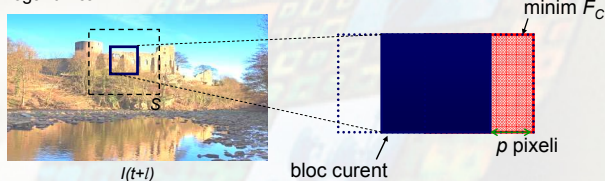


- dacă $p=1$ și F_C minim este obținut pentru vecinul din stânga sus sau dreapta jos al ultimului bloc curent
→ se re-evaluează F_C în X;

Estimarea mișcării

III. metode bazate pe blocuri

> căutarea ortogonală: combinație între căutarea în trei pași și cea logaritmică

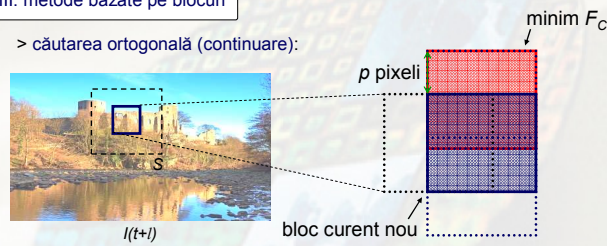


- inițial, funcția de cost este estimată între blocul curent (la t) și două blocuri vecine la distanță de p pixeli pe orizontală (la $t+1$),
- blocul pentru care F_C este minim devine bloc curent în imaginea la $t+1$;

Estimarea mișcării

III. metode bazate pe blocuri

> căutarea ortogonală (continuare):



- funcția de cost este estimată între blocul curent (la t) și alte două blocuri vecine la distanță de p pixeli pe verticală (la $t+1$),
- blocul pentru care F_C este minim devine bloc curent în imaginea la $t+1$;

Estimarea mișcării

III. metode bazate pe blocuri

> căutarea ortogonală (continuare):



- dacă $p>1$, $p \leftarrow p/2$ și se repetă pașii anteriori;
- altfel → STOP

Estimarea mișcării

III. metode bazate pe blocuri

> căutarea binară (MPEG): fereastra de căutare este împărțită în mai multe regiuni urmând să se realizeze o căutare completă în una dintre acestea

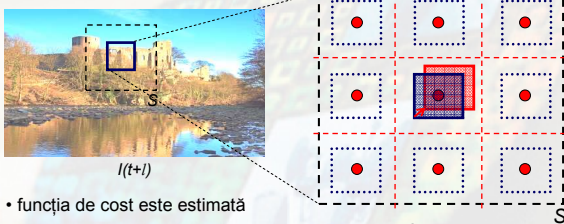


- se definește o grilă de 9 puncte repartizată după cele 8 direcții cardinale (accentul se pune pe regiunea centrală ~ cea mai probabilă);

Estimarea mișcării

III. metode bazate pe blocuri

> căutarea binară (continuare):

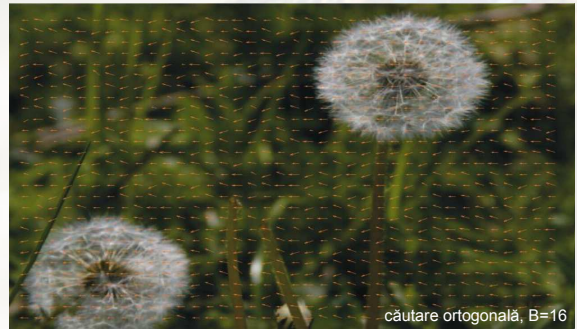


- funcția de cost este estimată între blocul curent (la t) și 9 blocuri din regiunile definite;
- în regiunea ce furnizează F_C minim este realizată o căutare completă.

Estimarea mișcării

III. metode bazate pe blocuri

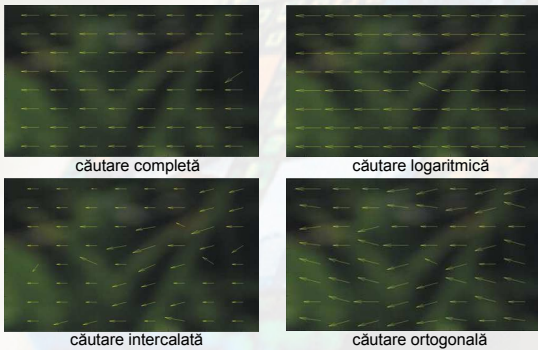
> exemple: mișcare de translație spre stânga:



Estimarea mișcării

III. metode bazate pe blocuri

> exemple: mișcare de translație spre stânga:



Estimarea mișcării

III. metode bazate pe blocuri

> exemple: mișcare de mărire (zoom-in):



Estimarea mișcării

III. metode bazate pe blocuri

> câteva cifre estimative:

Tip mișcare	Metoda de căutare	T[s] (B=8)	T[s] (B=16)	T[s] (B=32)	Precizie subiectivă
translație	completă	7.2	9.7	11.5	excelentă
	logaritmică	2.6	1.1	0.4	excelentă
	intercalată	1.6	0.8	0.3	medie
	ortogonală	1.6	0.8	0.3	scăzută
zoom-in	completă	5.3	9.6	11.5	excelentă
	logaritmică	1.9	1.1	0.4	excelentă
	intercalată	1.3	0.8	0.3	scăzută
	ortogonală	1.9	0.8	0.3	medie
mișcare obiecte	completă	1.8	7.1	6.8	bună
	logaritmică	0.8	0.8	0.2	bună
	intercalată	0.6	0.6	0.2	medie
	ortogonală	0.6	0.6	0.3	medie
statică	completă	0.3	0.08	0.04	excelentă
	logaritmică				
	intercalată				
	ortogonală				

> Sfârșit M5. Informația de mișcare