

PROCESAREA IMAGINILOR

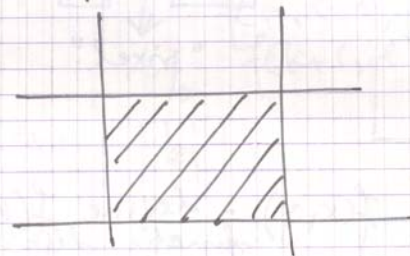
Mihai Ivanovici

4261.

INTRODUCERE

Modelul matematic al imaginii

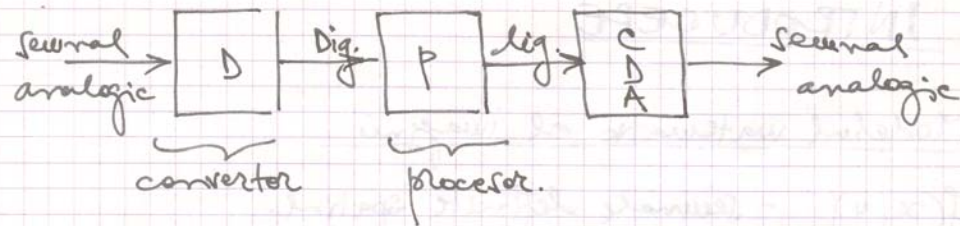
- $f(x, y)$ - semnale definite spațial.
- o imagine în tonuri de gri se poate modela cu $f(x, y)$.
- modelul poate fi în $L^2(\mathbb{R}^2)$.
(funcții de energie (repartizate în spațiu) finite).
- Spațiul $L^2(\mathbb{R}^2)$ poate fi limitat pe un domeniu finit D :
 $\Rightarrow$ spațiul $L^2(D)$:



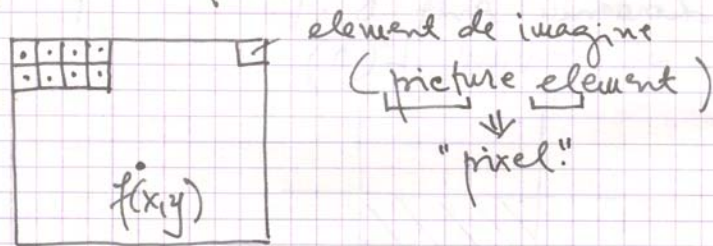
- clasele de obiecte poate fi reprezentat ca un spațiu Banach (normat). $\downarrow$ V vectorial.
- norma de spațiu Hilbert:

$$\|f\| = \sqrt{\int_D |f|^2 dx}$$

$$\langle f, g \rangle = \int_D f \cdot g dx$$



- prelucrări numerice digitale - în ac. caz se calculează cu valori aproximative. (din cauza reprezentării).
- teoria - modele de funcții continue
- în practică: funcții eșantionate



- funcția continuă $f(x,y)$ - devine $f(k,l)$ o funcție discretă.
- tipuri de procesări de imagini:
 - îmbunătățire (enhancement) - nu...
 - restaurare - cunoașterea modelului de degradare
 - refacerea imaginilor din proiectii
 - segmentare = separarea obiectelor ce compun imaginea
- operații de analiză:
 - clasificări
 - recunoaștere de forme.

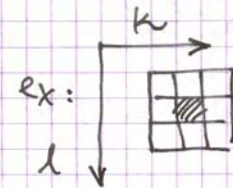
- compresia (taxată pe diverse modele).

Cazul modelului continuu (analogic)

$$f(x,y) \in L^2(\mathbb{R}^2).$$

- transformări ale imaginilor:
 - punctuale: $f(x_0, y_0) \rightarrow g(x_0, y_0) = \mathcal{C}(f(x_0, y_0))$
 - de vecinătate: $g(x_0, y_0) = \mathcal{C}(\text{valorile dintr-o vecinătate a lui } (x_0, y_0)).$

$$g(k,l) = \mathcal{C}(f(k,l), f(k-1,l), f(k-1,l-1), f(k-1,l+1), f(k,l-1), f(k,l+1), f(k+1,l-1), f(k+1,l), f(k+1,l+1)).$$



- integrale - de obicei transformări liniare.
- Obs: cele punctuale sînt în gn. neliniare.
cele de vecinătate: hî-așa - hî-așa.

Transformata Fourier bidimensională

$$F(u,v) = \iint_{-\infty}^{+\infty} f(x,y) \cdot e^{-j(ux+vy)} dx dy.$$

Obs: există și T.F. n-dimensionale:

$$\hat{f}(\bar{u}) = \int_{\mathbb{R}^n} f(\bar{x}) \cdot e^{-j\bar{u}\bar{x}} d\bar{x}.$$

$$\mathcal{F}[f(x,y)] = \hat{f}(u,v).$$

- lineare transformare este tot o transformare liniară:

$$\mathcal{F}(\alpha f + \beta g) = \alpha \cdot \mathcal{F}f + \beta \cdot \mathcal{F}g.$$

$$f(x,y) \rightarrow f'(x-x_0, y-y_0) \rightarrow$$

(funcția translatată)

$$f(x-x_0, y-y_0) = f'(x,y).$$

$$\begin{aligned} \hat{f}'(u,v) &= \iint_{-\infty}^{\infty} f(x-x_0, y-y_0) \cdot e^{-j(ux+vy)} dx dy \\ &= \iint_{-\infty}^{\infty} f(x', y') \cdot e^{-j(x'u + y'v + x_0u + y_0v)} dx' dy' \\ &= e^{-j(x_0u + y_0v)} \cdot \hat{f}(u,v) \end{aligned}$$

Transformata Fourier inversă

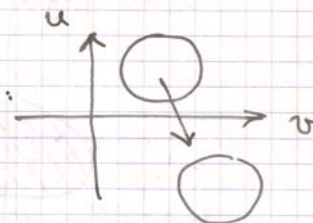
$$f(x,y) = \frac{1}{4\pi^2} \iint \hat{f}(u,v) \cdot e^{j(ux+vy)} du dv.$$

$$\hat{f}(u,v) = \int_{-\infty}^{\infty} e^{-jvy} \left(\int_{-\infty}^{\infty} f(x,y) \cdot e^{-jux} dx \right) dy =$$

$$= \int_{-\infty}^{\infty} \hat{f}(u,y) \cdot e^{-jvy} dy$$

$$\hat{f}(u,y) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \hat{f}(u,v) \cdot e^{jvy} dv$$

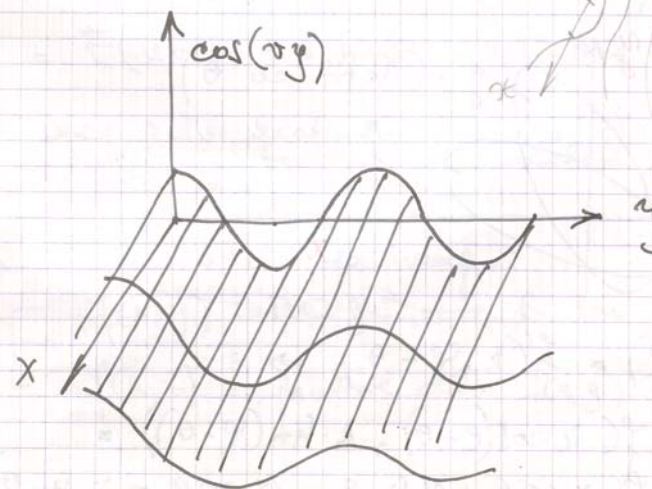
- translatarea spectrului: ex:



$$f'(x,y) = f(x,y) \cdot e^{j(u_0x + v_0y)}.$$

• funcții "cos" bidimensionale
 $\cos(ux + vy)$

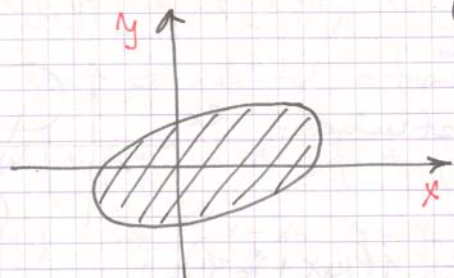
ex: $u=0 \rightarrow \cos(vy)$.



- de calculat perioada (intrinsecă) - pt. $u, v \neq 0$.

Proprietăți transformatei Fourier bidimensionale

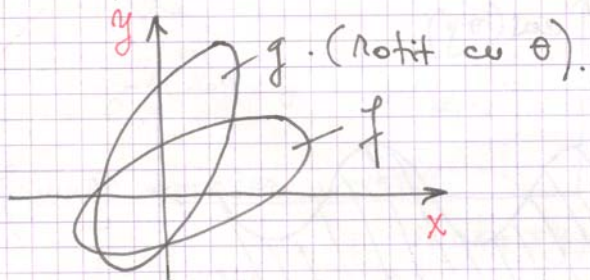
fie $f(x, y)$ - având un suport "finit"
(mulțimea punctelor, inclusiv
marginile, diferită de $\emptyset$)



- fie g - suportul rotit al funcției f .
 $x, y \rightarrow R, \varphi$ - coordonate polare

$$\begin{cases} x = R \cdot \cos \varphi \\ y = R \cdot \sin \varphi \end{cases}$$

$$f(x, y) = f(R \cos \varphi, R \sin \varphi) = f'(R, \varphi).$$



$$\Rightarrow g'(R, \varphi) = f'(R, \varphi - \theta) \Rightarrow$$

$$\begin{aligned} g(x, y) &= f(R \cos(\varphi - \theta), R \sin(\varphi - \theta)) = \\ &= f(x \cdot \cos \theta + y \cdot \sin \theta, y \cdot \cos \theta - x \cdot \sin \theta) \end{aligned}$$

$$G(u, v) = \iint_{\mathbb{R}^2} f(x \cdot \cos \theta + y \cdot \sin \theta, y \cdot \cos \theta - x \cdot \sin \theta) \cdot e^{-j(ux + vy)} dx dy =$$

$$\begin{cases} x \cdot \cos \theta + y \cdot \sin \theta = x' \\ -x \cdot \sin \theta + y \cdot \cos \theta = y' \end{cases} \begin{cases} | \cdot \sin \theta \\ | \cdot \cos \theta \end{cases}$$

$$\begin{cases} y = x' \cdot \sin \theta + y' \cdot \cos \theta \\ x = x' \cdot \cos \theta - y' \cdot \sin \theta \end{cases}$$

$$\Rightarrow G(u, v) = \iint_{\mathbb{R}^2} f(x', y') \cdot e^{-j(ux' \cdot \cos \theta - y' \cdot \sin \theta + vx' \cdot \sin \theta + vy' \cdot \cos \theta)} dx' dy' =$$

$$= \iint_{\mathbb{R}^2} f(x', y') \cdot e^{-j[x'(u \cdot \cos \theta + v \cdot \sin \theta) + y'(-u \cdot \sin \theta + v \cdot \cos \theta)]} dx' dy' =$$

$$= F(u \cos \theta + v \sin \theta, -u \sin \theta + v \cos \theta).$$

©: Dacă rotim ~~spatiul~~ imaginea $f(x, y) \Rightarrow$ se
poate în ~~spatiul~~ în același ~~spatiu~~ $\Rightarrow$ se
cei același unghi.

- obs: 2 modele de imagini:

$$\begin{cases} f(x, y) : D \subset \mathbb{R}^2 \rightarrow \mathbb{R} \text{ sau } \mathbb{R}^3 \text{ pt. color} \\ f(k, \varphi) - \text{provine din } f(x, y) \text{ prin eșantionare} \end{cases}$$

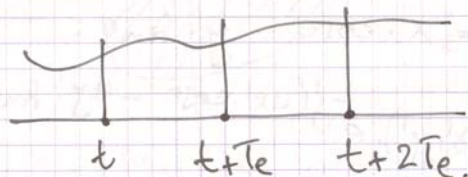
Teorema eșantionării 2D

- t. eș. unidim. :

dacă $f(t)$ și $F(\omega) = 0$ pt. $|\omega| > \Omega$ atunci:

$f(t)$ poate fi refăcut din eșantioanele sale a

$$f(t) = \sum_{-\infty}^{\infty} f(kT_e) \cdot \frac{\sin \omega_0(t - kT_e)}{\omega_0(t - kT_e)}$$

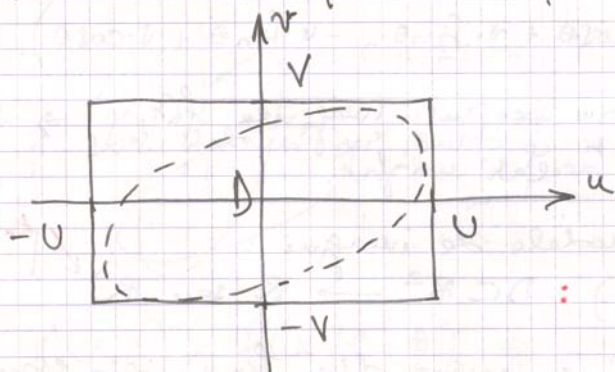


$$\omega_0 = \frac{2\pi}{T_e}$$

$$f_e = \frac{1}{T_e} > 2 f_{\max}$$

$$\frac{1}{T_e} > 2 \cdot \frac{\Omega}{2\pi} \quad \Omega < \frac{\pi}{T_e}$$

⑦: $f(x, y)$ - are un spectru de forma :



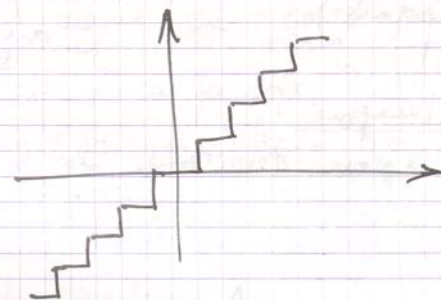
$$F(u, v) = 0 \text{ pt. } (u, v) \notin D.$$

$$f(x, y) = \sum_{-\infty}^{\infty} f(x - kx_0, y) \cdot \frac{\sin U(x - kx_0)}{U(x - kx_0)} =$$

$$= \sum_{-\infty}^{\infty} \sum_{-\infty}^{\infty} f(x - kx_0, y - ly_0) \cdot \frac{\sin U(x - kx_0)}{U(x - kx_0)} \cdot \frac{\sin V(y - ly_0)}{V(y - ly_0)}$$

↳ propoziția care ne trece din continuu în domeniul discret.

A doua aproximație: o val. analitică se transformă într-o val. discretă printr-un proces de cuantizare



- de câte eșantioane avem nevoie pt. a reprezenta un spectru cu o anumită precizie?
- o culoare = o an. curbă spectrală.
- avem 3 feluri de traductori $\Rightarrow$ spațiul imaginii (teoretic n dimensional) se proiectează pe 3 plane: R, G, B $\Rightarrow$ sint coeficienți 3 dim. pt. a reprezenta culorile.
- la nivelul ochiului are loc o reducere a informației de la nivelul traductoarelor retinei.
- $\exists$ și alte modele de culoare în care distanța între 2 vectori să fie egală cu distanța "perceptuală" între acei 2 vectori - în spațiul real al culorilor.

Constanta culorilor $\sqrt{\text{fen. ce}}$ compensează variabile de culoare datorate diferențelor de iluminare
 Obs: 3 componente este minimul de reprezentare a culorilor.

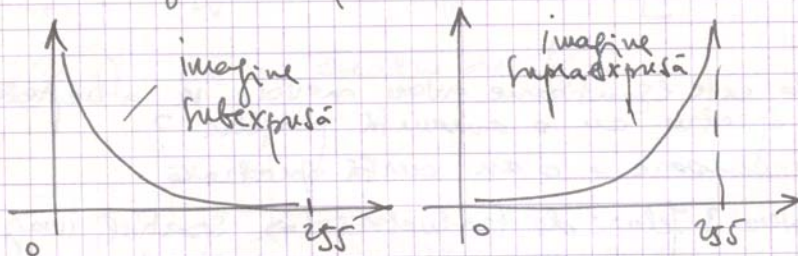
Modelarea matematică a imaginii

modelare deterministă
 modelare statistică. $\Rightarrow$ val. pixelilor sînt variabile aleatoare

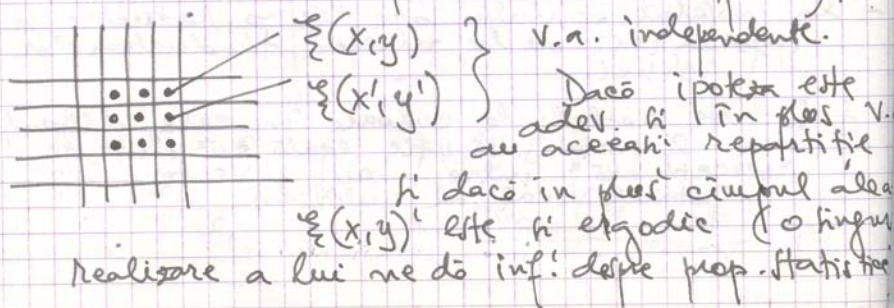
- vom lucra cu cîmpuri aleatoare
 (nu cu procese aleatoare care au o desfășurare temporală)
 au o desfășurare spațială.

Histograma unei imagini

- o imagine are aprox. 1 milion de pixeli. (eșantionare).
- ex: histogramă: pe 256 de valori.



- pixelii au valori corelate $\Rightarrow$ ipoteză simplificatoare: valorile sînt independente (ipoteză este totuși falsă).



atunci histograma reprez. estimatul densității de probabilitate pt. imaginea respectivă.

Dacă v.a. sînt indep. atunci ac. f. de repartiție nu ne dă informația maximă.

pt. 1, 2, 3 realizări: - informația medie este proporțională cu entropia sursei.

Sursa de informație: adică histograma ne dă informație maximă dacă "imprimăm" valorile pe toate valorile din domeniu $\Rightarrow$

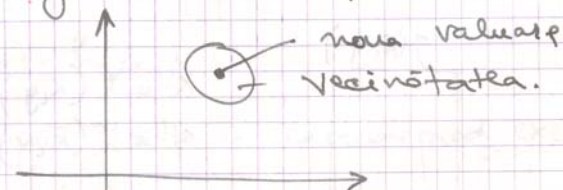
ex. f.v. egalizare de histogramă.

Avem o f. de repartiție, avem a-2-a f. de repartiție nouă ce reprez. histograma egalizată H este funcție de prima f. de repartiție $\Rightarrow$ care este acea funcție?

Problema este mai dificilă în dom. discret. Egalizarea în dom. discret este mai degrabă o împănare pe întreg domeniul a val. decât o egalizare globală ca în dom. continuu.

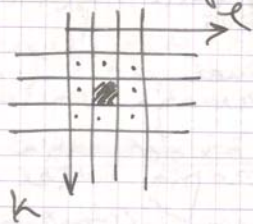
1° - preluări punctuale: $f(x,y)$ -
 pt. x, y : avem o funcție care "mută" val. din $f(x,y)$ în altă valoare: y
 $f(x,y) [f(x,y)] \Rightarrow \varphi(f) = g$

2° - preluări de vecinătate: Noua valoare în punctul $g(x,y)$ se obține ca un " φ ":
 $g(x,y) = \varphi(f \text{ în vecinătatea lui } (x,y))$



- în cazul imaginii continue.

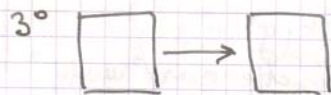
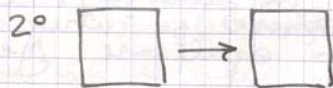
- în cazul imaginii discretizate.



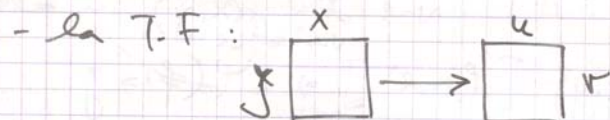
$$g(k,l) = \mathcal{C} \left(\begin{matrix} f(k-1, l-1), \\ f(k-1, l), f(k-1, l+1) \\ \dots \end{matrix} \right)$$

5° - prelucrări integrale - descrise de transformări integrale.
[ex: T. Fourier bidimensionale]
- sunt practic numai liniare.

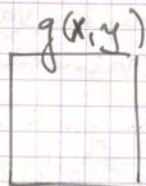
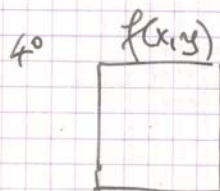
4° - transformări geometrice:



- dar au aceeași semnificație (coordonatele x și y)

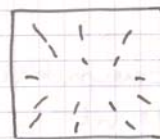
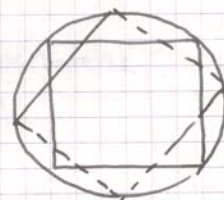


Obs.: $F(u) \cdot H(u) = G(u)$ - pt. filtrare unidim.
 $F(u,v) \cdot H(u,v) = G(u,v)$ - filtrare bidim.
= transf. punctuale?

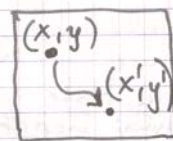


$$\begin{aligned} x' &= \mathcal{C}(x,y) \\ y' &= \mathcal{C}(x,y) \end{aligned}$$

$\mathcal{C}$ - poate pătrî sau nu domeniul de definiție
- ex. la o rotație: sau la o translație



- imaginea se "strânge" spre mijloc.
Determinați o astfel de transformare conformă.
(adică păstrează domeniul).



$$g(x',y') = f(x,y)$$

$$(x,y) \rightarrow (x',y') = \Phi(x,y)$$

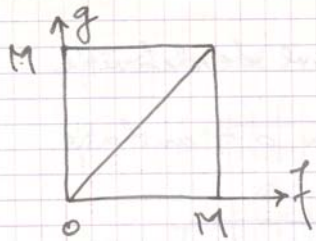
$$\Rightarrow g(x,y) = f(\Phi^{-1}(x,y))$$

PRELUCRĂRI PUNCTUALE

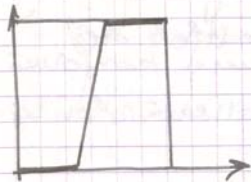
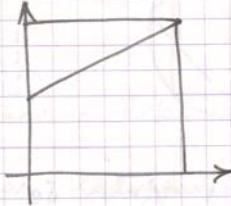
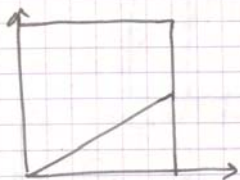
$$\begin{aligned} g(x,y) &= \mathcal{C}_{(x,y)}(f(x,y)) - \text{adică } \mathcal{C} \text{ depinde de } (x,y) \\ &= \mathcal{C}(f(x,y), x,y) \end{aligned}$$

(asta se întâmplă cînd facem o filtrare a spectrului)
sau $f(x,y) \cdot h(x,y)$.

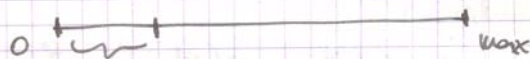
Prelucrări punctuale care depind exclusiv de valoarea.



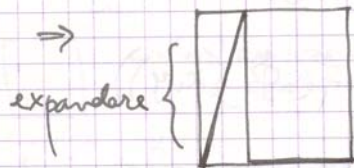
- bisectoarea $\Rightarrow$ nu are loc
nici o transformare



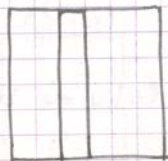
ex: tomografie computerizată:
standard: 11 bit = 2048 nivel



dacă vrem să vedem toate valorile
de aici și numai acestea:



- imagini
 - binare
 - cu tonuri de gri
 - color
 - multispectrale

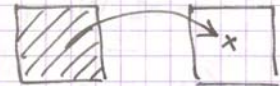


PRELUCRĂRI DE VECINĂTATE

Filtrarea liniară a imaginilor

- operatori de vecinătate $\Rightarrow$ prelucrări de vecinătate.

- operatori
 - punctuali
 - de vecinătate - Val. modificată a unui pixel în noua imagine depinde de o vecinătate a lui din imaginea inițială.
 - integrale



Operator liniar = un op. care respectă
principiul superpoziției.

- dacă avem 2 imagini: $f_1(u,v)$ și $f_2(u,v)$
și $a, b \in \mathbb{R}$, atunci:

$$O(a f_1 + b f_2) = a \cdot O(f_1) + b \cdot O(f_2)$$

Expresia unui filtru liniar se def. astfel:

dacă $f(u,v)$ - imaginea originală

$g(u,v)$ - imag. transformată

cu $u = 0, \dots, M-1$

$v = 0, \dots, N-1$

atunci:

$$g(u,v) = \sum_{(k,l) \in W} w_{kl} \cdot f(u-k, v-l)$$

unde W = o structură de puncte care definește
vecinătatea unui pixel, iar

w_{kl} = sint constante care s.n.
coeficienți filtrului

Un filtru liniar este definit de vecinătatea W a fiecărui pixel și de coeficienții w_{kl} .

El poate fi specificat prin precizarea unei matrice V , care s.n. masca de convoluție (masca de filtrare).

- caracterizată - de formă, coef. și origine.

ex:

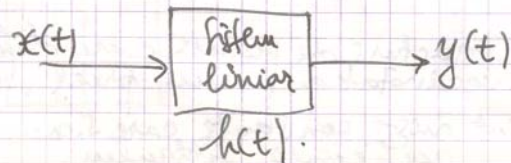
$$V = \begin{bmatrix} w_{-1,-1} & w_{-1,0} & w_{-1,1} \\ w_{0,-1} & \boxed{w_{0,0}} & w_{0,1} \\ w_{1,-1} & w_{1,0} & w_{1,1} \end{bmatrix}$$

Obs.: forma în gen - pătrată, de dim. impară.
- originea: $w_{0,0}$.

Filtrarea liniară poate fi descrisă astfel:
(operația de...)

- Se suprapune masca de convoluție peste fiecare pt. fiecare pixel al imaginii originale)
- a.i. originea măști să cadă peste pixelul considerat
- apoi se fac toate produsele între coeficienții măști și valorile pixelilor peste care s-au suprapus aceștia, iar suma acestor produse reprezintă noua valoare a pixelului considerat.
- (s.n. tehnica ferestrei glisante)

↑ operația descrisă = convoluție bidimensională.
- cazul convoluției unidimensionale:



9.

$$y(t) = x(t) * h(t) = \int_{-\infty}^{\infty} h(\tau) \cdot x(t-\tau) d\tau.$$

↑ cazul continuu.

↓ cazul discret.

$$y(n) = \sum_m h(m) \cdot x(n-m).$$

($= -\infty$) cazul general

$$= \sum_{m=0}^{M-1} h(m) \cdot x(n-m) - \text{cazul practic. (restul coef. sunt zero -)}.$$

Alegerea coeficienților filtrului:

Filtre de netezire

- echivalente FTJ
- FTJ - folosite în gen. pt. eliminarea zgomotului (care este de bandă largă).
- în general informația în principal se regăsește în componentele de joasă frecvență
- prin filtrare TJ se "face" zgomotul - care va avea putere mult mai mică. (în realitate se "face" și puțin din semnalul util).
- pt. imagine - scopul principal = reducerea puterii zgomotului în imagine. Modelul cel mai utilizat: zgomotul este aditiv și alb.
- pt. o imagine $f(i,j) \Rightarrow$

$$f_h(i,j) = \underbrace{f(i,j)}_{\text{imagine curată}} + \underbrace{n(i,j)}_{\text{perturbatie = zgomot (alb și independent față de imagine)}} = \text{imagine cu "puceci".}$$

ipoteze:

n = signal stationar (comportamentul statistic nu depinde de i și j).

$\bar{n} = 0$ = media

σ_n - dispersia

varianța

$$\overbrace{n(i,j) \cdot n(i,j)}^{\text{v.a. indep.}} = \begin{cases} \sigma_n^2 & i=i, j=j \\ 0 & \text{în rest} \end{cases}$$

⇒ FTJ - reduce cantitatea de zgomot din imagine.

Condiția de normare a coef. filtrului de netezire

* componenta continuă nu trebuie să fie alterată de către filtru.

- condiția de normare se deduce impunând condiția de mai sus. (*)

, cu alte cuvinte, filtrul să conserve luminanța medie a imaginii.

$$f(i,j) = f(i,j) = A \quad \forall i,j$$

$$\Rightarrow g(i,j) = A \quad \forall i,j \quad (g = \text{imaginea filtrată})$$

$$\Rightarrow A = \sum_{(k,l) \in W} w_{kl} \cdot A \Rightarrow$$

$$\boxed{\sum_{(k,l) \in W} w_{kl} = 1}$$

- condiția de normare pt. FTJ.

$$\boxed{w_{kl} \geq 0}$$

Cel mai simplu ex. de FTJ (de netezire) este filtrul de mediere.

Filtrul de mediere

- toți coef. sînt egale.

- mască unui f. de med. 3x3:

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

5x5.

1/25	1/25			
1/25	--	--		

Efectul filtrului de mediere asupra unei imagini cu zgomot.

$$f_1(i,j) = f(i,j) + n(i,j) \quad \forall i,j$$

$$N \times N \quad w_{kl} = \frac{1}{N^2} \quad \forall k,l, \quad N = 2k+1 \quad (\text{impar}).$$

$$\text{notăm } N^2 = P.$$

$$g(i,j) = \sum_{(k,l) \in W} w_{kl} \cdot f_1(i-k, j-l) =$$

$$= \underbrace{\sum_{(k,l) \in W} \frac{1}{P} \cdot f(i-k, j-l)}_{\text{ce-a ieșit din semnalul util}} + \underbrace{\sum_{(k,l) \in W} \frac{1}{P} \cdot n(i-k, j-l)}_{\text{ce-a rămas din zgomot după filtrare.}}$$

$$n(i,j)$$

$$n_f(i,j) = \sum_{(k,l) \in W} \frac{1}{P} \cdot n(i-k, j-l)$$

$$n_f(i,j) = \frac{1}{P} \sum_{i=1}^P n_i$$

n_i - eșantioanele
(P la număr)
de zgomot din
vecinătate.

$$n_f(i,j) = \text{fol. o var. aleatoare}$$

(medie de v.a.)

$$\Rightarrow \text{media} = \overline{n_f(i,j)} = \frac{1}{P} \sum_{i=1}^P n_i = \frac{1}{P} \sum_{i=1}^P \bar{n}_i = 0$$

- varianța zgomotului filtrat:

$$\sigma_{n_f}^2 = \overline{n_f^2(i,j)} = \overline{\left(\frac{1}{P} \sum_{i=1}^P n_i \right)^2} =$$

$$= \frac{1}{P^2} \cdot \sum_{i,j=1}^P n_i n_j = \frac{1}{P^2} \cdot \sum_{i,j=1}^P \overline{n_i n_j}$$

dacă $\begin{cases} i \neq j & n_i, n_j - \text{v.a. indep.} \\ i = j & \Rightarrow \overline{n_i n_j} = \overline{n_i} \cdot \overline{n_j} = 0 \end{cases}$

$\Rightarrow \overline{n_i n_j} = \overline{n_i} \cdot \overline{n_j} = 0$

$\Rightarrow \overline{n_i n_j} = \overline{n_i^2} = \sigma_n^2$

$$= \dots = \frac{P \cdot \sigma_n^2}{P^2} \Rightarrow \boxed{\sigma_{n_f}^2 = \frac{\sigma_n^2}{P}}$$

$$\boxed{\sigma_{n_f} = \frac{\sigma_n}{N}}$$

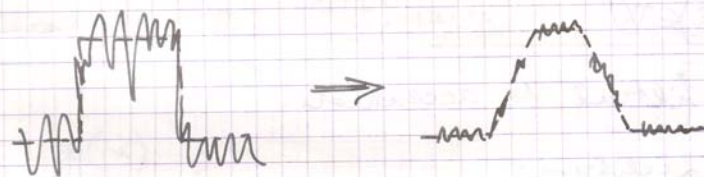
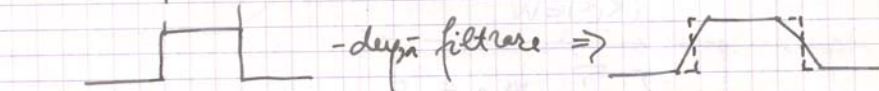
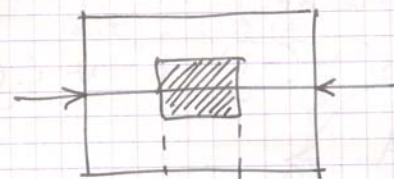
- puterea zgomotului scade de P ori (N^2 ori)
(varianța scade de P ori $\Rightarrow$ valori
mult mai apropiate)

Dezavantaj - în semnalul util are de suferit.

- filtrarea este dezanjantă în zonele în care
(de mediere) imaginea conține frecvențe înalte
(variații, tranziții bruste).

$\Rightarrow$ apare un fenomen de întețosare
(de "blurring").
blurring :

ex:



- d.p.d.v. al zgomotului - este utilă o mască mare
- d.p.d.v. al imaginii - este utilă o mască mică.

© - filtrul de mediere nu se prea folosește.

Alte nuclee de convoluție:

$$\frac{1}{16} \cdot \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

- însă toate au în principal aceleași rezultate.
 $\rightarrow$ în gen se fol. filtre adaptive.

Filtrarea trece-sus

- componente de frecvență înaltă = variațiile bruște (în imagine).

Condiția de volumare a coef.

- se deduce impunând condiția ca comp. continuă a semn. să nu treacă prin filtru (să fie complet atenuată de către el.)

$$f(i,j) = A \quad \forall(i,j) \Rightarrow g(i,j) = 0 \quad \forall i,j.$$

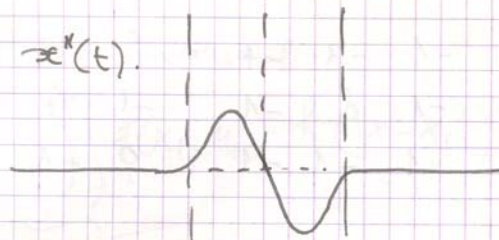
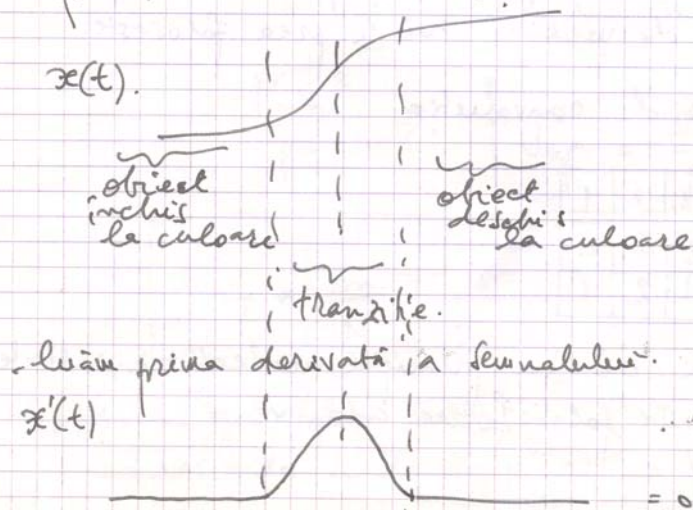
$$0 = \sum_{(k,l) \in W} w_{kl} \cdot A \Rightarrow$$

$$\boxed{\sum_{(k,l) \in W} w_{kl} = 0}$$

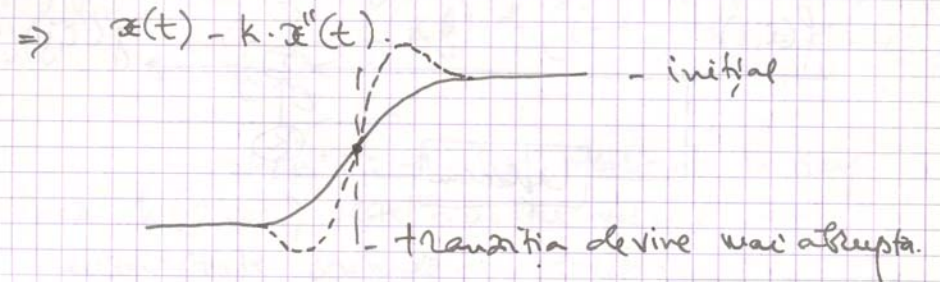
Aplicație - filtrul de accentuare.

Filtrul de accentuare

- nu este FTS, ci folosește filtrarea TS. pt. a realiza accentuarea.
- ex. fie cazul unidimensional.



Dacă scădem un anumit procent $x'(t)$ din $x(t)$.



$$x'(t_0) = \lim_{t \rightarrow t_0} \frac{x(t) - x(t_0)}{t - t_0} = \text{cazul continuu.}$$

$$x(n) \Rightarrow$$

$$x'(n) = x(n) - x(n-1) \Rightarrow \text{masca unidimensională de filtrare.}$$

$$\text{sau } x'(n) = x(n+1) - x(n).$$

sau

$$\boxed{1 \quad -1}$$

$$\boxed{-1 \quad 1}$$

$$x''(n) = x(n+1) - 2x(n) + x(n-1)$$

$$\Rightarrow \boxed{1 \quad -2 \quad 1}$$

Obs. fiind cazul discret $\Rightarrow$ și mai multe moduri de aproximare.

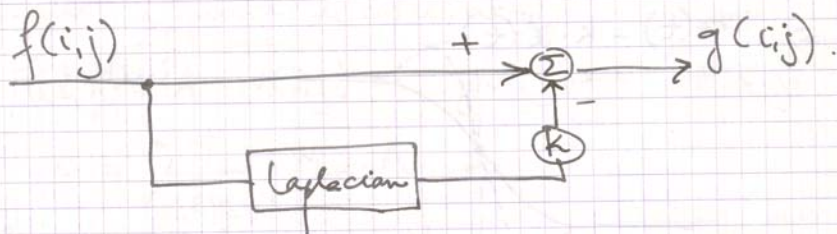
derivata a 2-a = Laplacian.

Mașă bidim.

1	-2	1
-2	4	-2
1	-2	1

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

⇒ filtrul de accentuare.

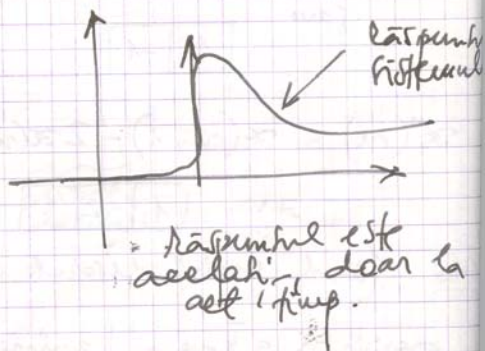
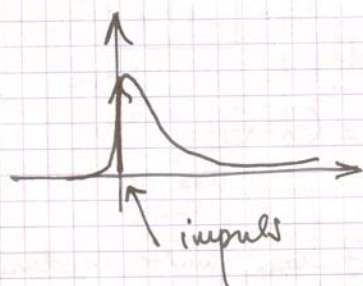


op. de vecinătate / liniare

$$\int s(\tau) \cdot h(t, \tau) d\tau$$

dacă histereul este invariant în timp

$$\int s(\tau) \cdot h(t - \tau) d\tau$$



- bidim:

$$g(x, y) = \iint f(x_1, y_1) \cdot h(x, y, x_1, y_1) dx_1 dy_1$$

- pt. hist. invariante:

$$\iint f(x_1, y_1) \cdot h(x - x_1, y - y_1) dx_1 dy_1$$

integrală de convoluție bidim.

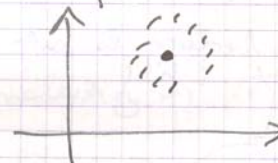
$$\iint \text{derive: } \sum_k \sum_l f(k, l) \cdot h(i - k, j - l) = g(i, j)$$

variabile pt. hist. invariante în timp.

Dacă $h(2 \text{ var.})$ este "nucleu reie":

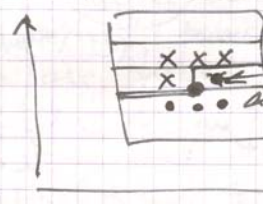
- cauzalitate spațială nu există.

(dacă avem un punct luminos, el va radia:



3 ni modele cauzale.

'la TV:

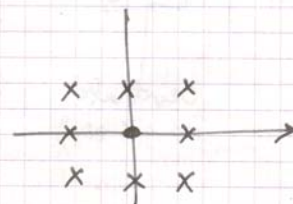


depend de "central" "x" - nu.

$$h(0, 0) = 0$$

$$h(i, j), |i|, |j| \leq 1$$

nucleu de 3x3.



dacă $|i|, |j| \leq 2 \Rightarrow$ nucleu 5×5 .

Când nucleul are 9 elemente:

$$\sum_{k,l} f(k,l) \cdot h(i-k, j-l)$$

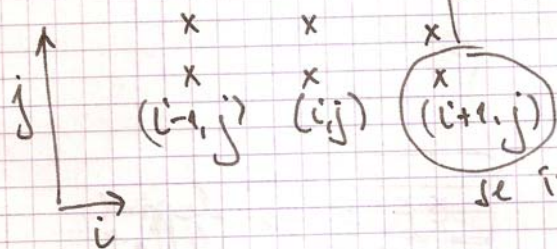
(i, j sunt
fixați)

dacă $k = i, i-1, i+1 \rightarrow$ prima coordonată
0, 1, -1.

$l = j, j-1, j+1$.

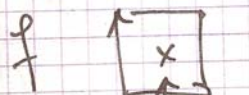
dacă $k, l \neq i, j$ cu mai mult de unul
 $\Rightarrow$ argumentele sunt mai mult de 1
 $\Rightarrow h = 0 \rightarrow$

suma se reduce la calc. a 9 termenii
pt. i, j .



se înmulțește cu $h(-1, 0)$

pt. o vecinătate:

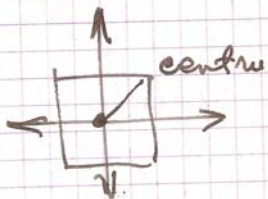


centrală
în x

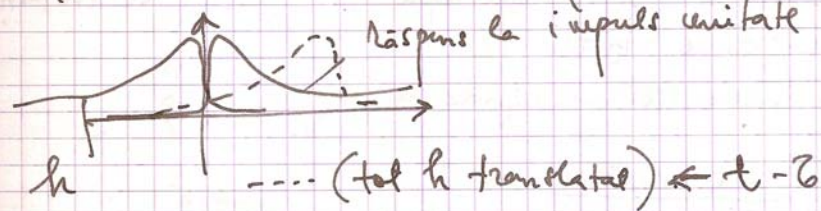
h devine o vecinătate



adică



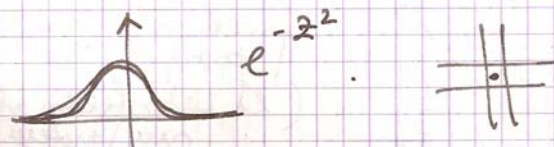
h_i în cazul unidim. e la fel:



Imaginile în gr. sunt însoțite de zgomot:

$$v(x, y) = i(x, y) + z(x, y)$$

- o ipoteză: este ca pixelii din vecinătatea
unui pixel sunt pe o distribuție
Gaussiană:



$\Rightarrow$ concluzia $\Rightarrow$ se fol. filtre liniare.

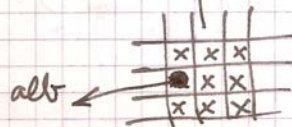
- însă în realitate zgomotul este de tip "sare și
pipet".
 $\Rightarrow$ zgomotul are amplitudine mare
(puncte albe și puncte negre).

- în gr. imaginea are nivele între 5% și 95%,
iar zgomotul: 0 sau 255.

Zgomotul are o distribuție de forma:



$\rightarrow$ opțiune nu mai este filtrul linear.



- pt. un filtru median - punctul "vizibil" este
 imaginea în jurul lui în toate direcțiile
 în care intră.

- dacă vrem să eliminăm acest punct $\Rightarrow$

Filtre de ordine

- ac. filtre nu sînt liniare.

- pt. un filtru linear:

$$L(\alpha f + \beta g) = \alpha L(f) + \beta L(g).$$

ex: notăm val. pixelilor: $v_0, v_1, \dots, v_8 = v_k$
 $k=0,8$

4	3	2
5	0	1
6	7	8

- facem abstracție de
 poziția lor în spațiu
 (la un filtru linear
 nu interesează
 ap. lucr.).

$$v_k \rightarrow v(k) \quad k=0,8$$

$$\Downarrow v(0), v(1), v(2), \dots, v(8)$$

$$\text{unde } v(0) \leq v(1) \leq v(2) \leq \dots \leq v(8).$$

un filtru linear:

$$\text{pt. } x_1, \dots, x_n \Rightarrow x_{(1)}, \dots, x_{(n)}$$

$$y_1, \dots, y_n \Rightarrow y_{(1)}, \dots, y_{(n)}$$

$$z = x + y: z_1, \dots, z_n \Rightarrow z_{(1)}, \dots, z_{(n)}$$

atunci în general $z_{(k)} \neq x_{(k)} + y_{(k)}$.

- ~~contra~~ exemplu:

$$x: 0, 1, 2, \dots, 8$$

$$y: 8, 7, 6, \dots, 0$$

$$y_{\text{ordonat}} = 0, \dots, 7, 8.$$

$$\Rightarrow z: 8, 8, 8, \dots, 8.$$

$$z_{\text{ordonat}} = 0, 2, 4, 6, \dots, 16$$

$$\text{fie } \mathcal{C}(v_1, \dots, v_n) = (v_{(1)}, \dots, v_{(n)})$$

$\mathcal{C}$ = funcția de ordonare a n -tuplului.

- pt. filtrare avem nevoie de 1 singură valoare
 $\Rightarrow$

Filtru median

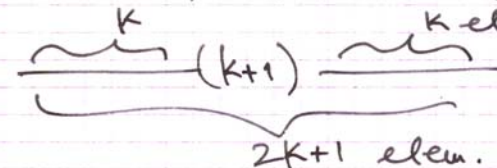
median $\neq$ medie.

$$\text{pt. } n = 2k + 1$$



mediana este $v_{(k+1)}$ element.

are k - val. mai mici în stînga
 k - val. mai mari în dreapta.

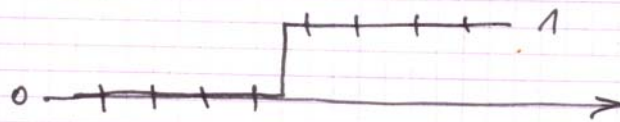


- în general $n = 2k + 1$. (poate fi și $2k$).

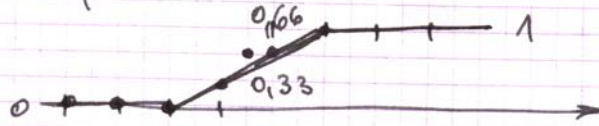
Filtrul de minim și filtrul de maxim

- f. de min = ia val. cea mai mică

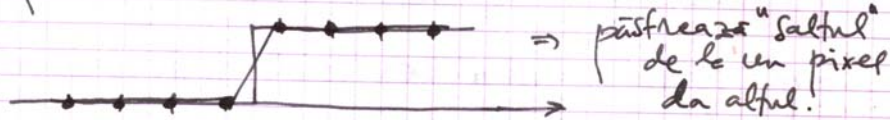
- f. de max = ia val. cea mai mare.



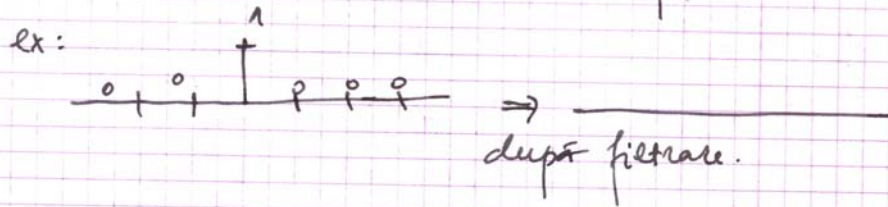
- pt. un filtru de mediere cu vecinătate 3 : $\rightarrow$



- un filtru de median va realiza la ieș. :



adică păstrează mai bine
prezențele înalte
(trec frecv. foarte înalte, care
ntăre compoanente continue)

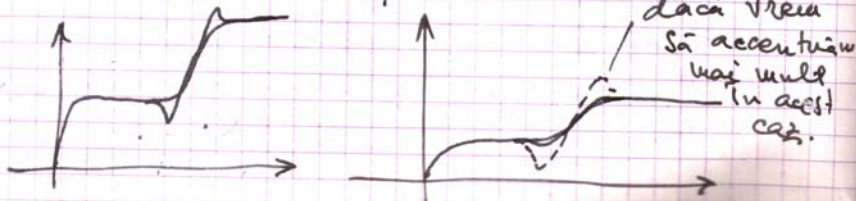


Filtre adaptive

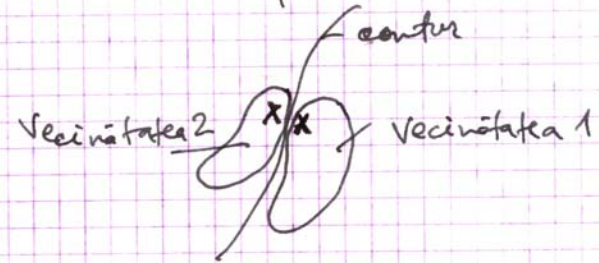
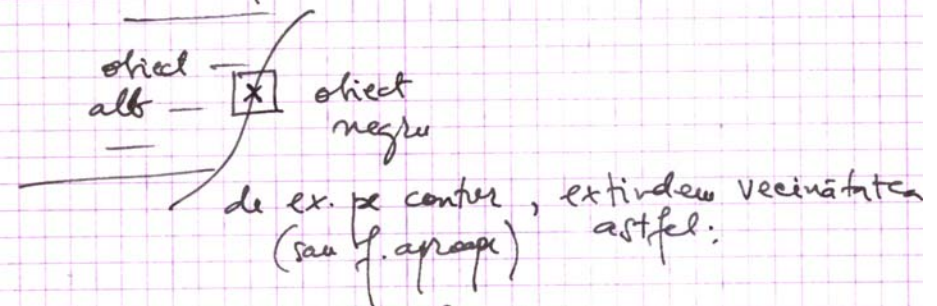
- un filtru "normal" : $\sum_1^n \alpha_i f_i$

- un f. adaptiv liniear : $\sum_1^n \alpha_i(f) \cdot f_i$

ex: accentuări "MAH" ?



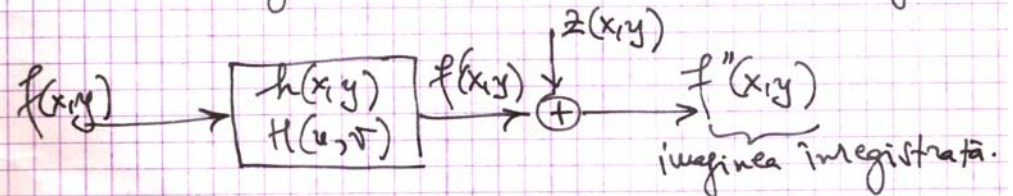
- vecinătățile pot fi alese adaptiv.



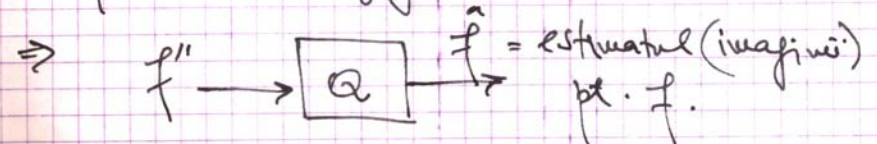
$\rightarrow$ Îmbunătățirea imaginilor
avem $f(x,y)$ - și nu se știe cum a fost degradată.

$\rightarrow$ Restaurarea imaginilor
- degradare / deteriorare
cu zgomot aleator.

- în general se cunoaște modelul de degradare.



- în general f efecte de margine, o anumită invarianță spațială a aparatelor de captat imaginea pe care le neglijăm.

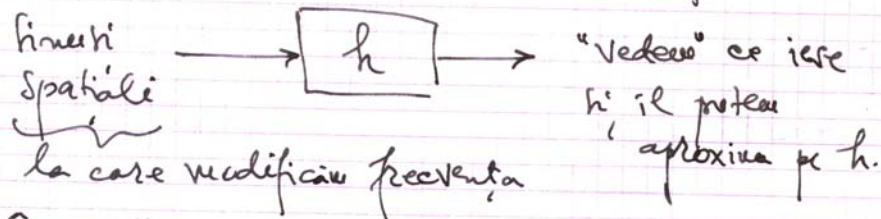


$\hat{f}$ - trebuie să fie optim într-un anumit sens.
 $\hat{f}$ - tr. să fie cât mai apropiat de f .

$$f'' = f' + 2 = f ** h + 2$$

dacă cunoaştem $2 \Rightarrow f' = f'' - 2$
 dar nu cunoaştem 2 !!

- am putea pp. că îl cunoaştem pe h - dar de obicei nu putem face o astfel de presupunere.
- dar ne achiziţionăm doar un f , ci mai multe imagini - pt. care " h " nu se modifică.



$$\| \hat{f} - \hat{f} ** h \| - \text{să fie micșorat.}$$

$$\| \cdot \|^2 - \text{min.}$$

$$\| \hat{f} - \hat{f} ** h \|^2 = 0 \text{ de ex.}$$

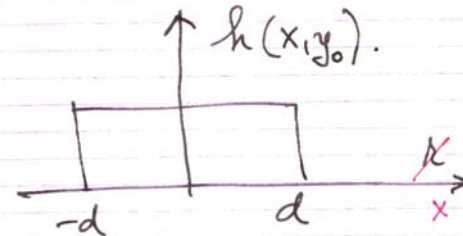
$$\| f \| ^2 = \| \hat{f} \|^2$$

$$\| F''(u, v) - \hat{F} * H(u, v) \| = 0 = \text{filter invers.}$$

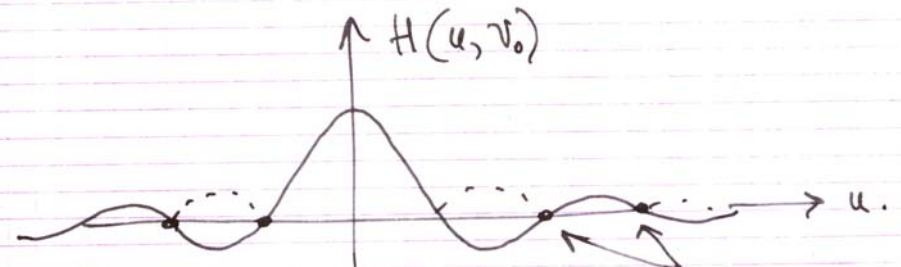
$$F'' = \hat{F} \cdot H$$

$$\hat{F}(u, v) = \frac{F''(u, v)}{H(u, v)}$$

- pp. că avem un sistem cu ieșire finită
 (cazul curent):

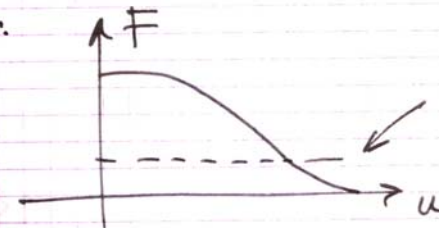


$$\text{transformata dreptunghiului} = \frac{\sin x}{x}$$

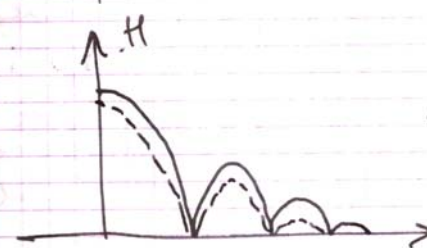


În punctele în care $H(u, v_0) = 0$ vom avea "probleme" cu calcularea lui $\hat{f}$.

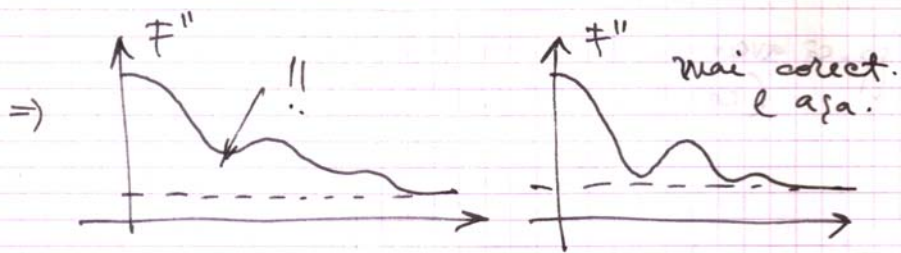
- pp:



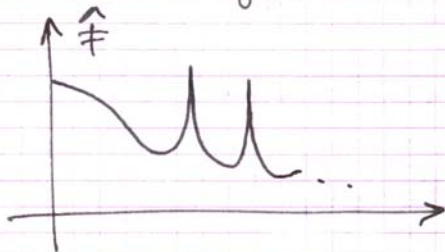
Spectrul dreptunghiului
 (= relativ constant.)



$$F \cdot H = \text{---}$$



din cauza zgomotului $F'' \neq 0$.



$\Rightarrow$ trebuie aduse niște îmbunătățiri.

dacă $|Q| = \frac{1}{|H|} > \max$. - crește peste o valoare maximă atunci o trencem (pt. că știm că se dat. / 0).

sau când crește mult o facem zero

Metode de optimizare

$$f'' = f * h + z \quad (z = \text{zgomot}; \quad * = ** \quad \text{convoluție bidimensională})$$

$$z = f'' - f * h$$

dacă estimăm f cu $\hat{f}$

$$\Rightarrow z \approx f'' - \hat{f} * h$$

- dacă nu în toate punctele, neșar să respecte anumite condiții globale.

$$E_{zg} = \|z\|^2 = \|f'' - \hat{f} * h\|^2$$

$$E_{zg} = \text{energia de zgomot} = E_z$$

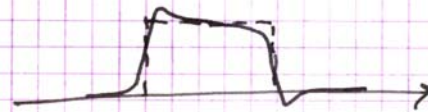
$\Rightarrow$ constrângere pt. $\hat{f}$:

$$\|f'' - \hat{f} * h\|^2 - E_z = 0.$$

ex: filtrare inversă pe un semnal unidimensional.



- trecut prin filtru cu bandă finită $\Rightarrow$

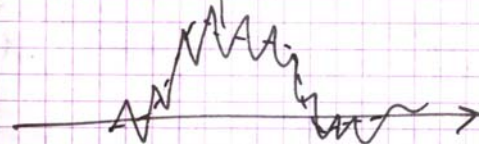


+ zgomot $\Rightarrow$



~

inversul fn. de integrare = derivare $\Rightarrow$
se va reface panta, prin filtrare inversă, ce presupune variații mari $\Rightarrow$



$$\|\hat{f} * h\| - \text{să fie minimă.}$$

(pb. de minim cu constrângeri).

- dintre toate $\hat{f}$ alegem un $\hat{f}$ care de ex. să ne dea o altă derivată minimă.

→ se poate scrie:

$$\|F'' - \hat{F} \cdot H\|^2 - E_2 = 0.$$

$$n, \| \hat{F} \cdot C \|^2 \rightarrow 0$$

$$\| \hat{F} \cdot C \|^2 + \lambda (\|F'' - \hat{F} \cdot H\|^2 - E_2) \rightarrow \text{minimum}$$

Lagrange (limitator?).

Dacă trecem din dom. continuu - în discret →

$$\sum_{k,l} \hat{F}(k,l) \cdot F(k,l) \quad \underline{F(k,l) = A(k,l) + j \cdot B(k,l)}.$$

notarea devine astfel o hruș →

$$\sum_{k,l} |\hat{F}(k,l) \cdot C(k,l)|^2 + \lambda \left(\sum_{k,l} |F''(k,l) - (A+jB)H|^2 - E_2 \right) \rightarrow \text{minimum}$$

$E_2 = E_2$

$$\sum_{k,l} \left[(A+jB)C(A-jB) \cdot C^* + \lambda (F'' - (A+jB)H) \cdot (F''^* - (A-jB)H^*) \right] - \lambda E_2 \rightarrow \text{minimum}$$

$C^* = \text{complex conjugate}$

$$|(A+jB)|^2 = (A+jB)(A+jB)^* = (A+jB)(A-jB)$$

- notarea este: $A, B(k,l)$.

- derivăm în funcție de A, B .

pt. fiecare k, l avem o ecuație independentă.

$$\sum \left[(A^2 + B^2) \cdot |C|^2 + \lambda (|F''|^2 - F''(A-jB)H^* - (A+jB)H \cdot F''^* + (A^2 + B^2) \cdot |H|^2) \right] \rightarrow \text{min.}$$

$$\text{avem } (A-jB) \cdot F'' \cdot H^*$$

$$+ (A+jB) \cdot (F'' \cdot H^*)^*$$

$$\Leftrightarrow (A-jB) \cdot (C+jD) + (A+jB) \cdot (C-jD)$$

= ... trebuie să tăiem partea imaginară?!

$$(AC + AjD - jBC + \dots = 2(AC + BD).$$

$$C = \text{Re} \{ F'' \cdot H^* \}$$

$$D = \text{Im} \{ F'' \cdot H^* \}.$$

$$\Rightarrow -(A-jB) \cdot F'' \cdot H^* - (A+jB)H \cdot F''^* = -2A \cdot \text{Re} \{ F'' \cdot H^* \} - 2B \cdot \text{Im} \{ F'' \cdot H^* \}.$$

$$\frac{\partial}{\partial A} \Big|_{k,l} \cdot \text{Expresie} = 0. \Rightarrow$$

$$\frac{\partial}{\partial A(k,e)} = 2A|c|^2 - 2\lambda \operatorname{Re}\{F''H^*\} + 2A|H|^2 = 0$$

$$A(|c|^2 + \lambda|H|^2) = \lambda \operatorname{Re}\{F''H^*\}$$

$$\frac{\partial}{\partial B(k,e)} = 2B|c|^2 - 2\lambda \operatorname{Im}\{F''H^*\} + 2B|H|^2 = 0$$

$$\Rightarrow B(|c|^2 + \lambda|H|^2) = \lambda \operatorname{Im}\{F''H^*\}$$

pt. $\forall k, e$.

$$\Rightarrow \hat{F}(k,e) = A + jB = \lambda \frac{\operatorname{Re}\{F''H^*\} + j\operatorname{Im}\{F''H^*\}}{|c|^2 + \lambda|H|^2}$$

$$\Rightarrow \hat{F}(k,e) = \frac{\lambda \cdot F''H^*}{|c|^2 + \lambda|H|^2} \cdot \frac{H}{H} \Rightarrow$$

$$\hat{F}(k,e) = \frac{1}{H(k,e)} \cdot \frac{\lambda \cdot F'' \cdot |H|^2}{\lambda|H|^2 + |c|^2}$$

$\Rightarrow$ avem de-a face cu un filtru linear -
pt. $\forall k, e$.

$$\hat{F}(k,e) = Q(k,e) \cdot F''(k,e)$$

$$Q(k,e) = \frac{1}{H(k,e)} \cdot \frac{\lambda \cdot |H(k,e)|^2}{\lambda|H(k,e)|^2 + |c(k,e)|^2}$$

$$\|F'' - \hat{F} \cdot H\| = E_2 \Leftrightarrow$$

$$\sum_{k,e} |F''(k,e) - \hat{F}(k,e) \cdot H(k,e)|^2 = E_2 = E_n$$

$$\hat{F} = F'' \cdot Q \Rightarrow$$

$$\left| F''(k,e) - \frac{\lambda \cdot |H(k,e)|^2 \cdot F''(k,e)}{\lambda|H(k,e)|^2 + |c(k,e)|^2} \right|^2 \Rightarrow$$

$$\sum_{k,e} |F''(k,e)|^2 \cdot \frac{(\lambda|H(k,e)|^2 + |c(k,e)|^2)^2}{(\lambda|H(k,e)|^2 + |c(k,e)|^2)^2} = E_n$$

- pt. $H(k,e) \approx 0$. sau foarte mic

$$Q(k,e) \approx |H^*| \approx 0$$

$\Rightarrow$ unde H este f. mic $\Rightarrow$ Q va fi f. mic.

$$E_n = \sum_{k,e} |F''(k,e)|^2 \cdot \frac{|c(k,e)|^4}{(\lambda|H(k,e)|^2 + |c(k,e)|^2)^2}$$

- pt. a afla λ - se rezolvă ec. prin
aproximări succesive



ne interesează doar pt. $\lambda \geq 0$.

facem funcția $E_n(\lambda)$ și pt. un E_n dat
îl calculăm și pe λ .

- dacă zgomotul e f. mic $\Rightarrow \lambda \rightarrow \infty$.

$H =$ cunoscut

$C =$ filtru de derivare

$$\Rightarrow \lambda |H(k, l)|^2 \gg |C(k, l)|^2$$

$$\Rightarrow Q(k, l) \approx 1.$$

- în acel loc $\Rightarrow Q = \frac{1}{H} =$ filtru invers.

- unde zgomotul e foarte mare w , $H(k, l) \approx 0$.

$$\Rightarrow Q = \dots$$

$Q =$ filtru cu constrângeri.

TRANSFORMĂRI UNITARE

- se referă la o clasă de matrice utilizate pt. reprezentarea semnalelor (imaginiilor)

- o matrice este unitară (de tip $N \times N$) dacă inversa ei este transpusa: A^{*T} :

$$A \cdot A^{*T} = A^{*T} \cdot A = I_N. \quad A \in \mathbb{C}_{N \times N}.$$

Cazul 1-D.

- fie $\underline{u} =$ vector:

$$\underline{u} = [u(0) \ u(1) \ \dots \ u(N-1)]^T$$

exemplu de timp ale unui semnal.

$\underline{v} = A \cdot \underline{u}$ - transformarea unitară directă.

$$\underline{v} = [v(0) \ v(1) \ \dots \ v(N-1)]^T$$

$\underline{v} =$ transformata lui $\underline{u}$.

mat. A specifică transformarea.

$$v(k) = \sum_{n=0}^{N-1} a(k, n) \cdot u(n) \quad + k = \overline{0, N-1}.$$

$$A = \{a(k, n)\}_{k, n = \overline{0, N-1}}.$$

Transformarea inversă:

$$\underline{u} = A^{-1} \cdot \underline{v} \Rightarrow \underline{u} = A^{*T} \cdot \underline{v}.$$

$$u(k) = \sum_{n=0}^{N-1} a^*(k, n) \cdot v(k) \quad + k = \overline{0, N-1}.$$

$v(n) !!$

$$A^{*T} = [\underline{a}_0^* \underline{a}_1^* \dots \underline{a}_{N-1}^*]$$

$\underline{a}_k^*$ - a k-a coloană a matricei A^{*T} .

Transf. Inversa

$\underline{u} = A^{*T} \cdot \underline{v}$ - se scrie sub formă

$$\underline{u} = [\underline{a}_0^* \underline{a}_1^* \dots \underline{a}_{N-1}^*] \cdot \begin{bmatrix} v(0) \\ \vdots \\ v(N-1) \end{bmatrix} =$$

$$= \sum_{k=0}^{N-1} v(k) \cdot \underline{a}_k^*$$

formă de combinație liniară
cu coef. $v(k)$.

= descompunerea vectorului $\underline{u}$ în altă
bază.

$$\text{ex: } \underline{u} = \begin{pmatrix} 2 \\ 5 \end{pmatrix} = 2 \begin{pmatrix} 1 \\ 0 \end{pmatrix} + 5 \begin{pmatrix} 0 \\ 1 \end{pmatrix} = 2\underline{e}_1 + 5\underline{e}_2$$

$\underline{e}_1, \underline{e}_2$ = vectori
ai bazei canonice.

Obs. #2 vectori independenți pot forma o bază
în $\mathbb{R}^2$.

- dacă sînt $\perp \Rightarrow$ bază ortogonală
- dacă sînt de normă 1 $\Rightarrow$ bază ortonormală.

Transf. unitară = descomp. unui vector într-o
altă bază formată din coloanele matricei
 A^{*T} , iar elementele vectorului transformat

$v(k)$ sînt coef. transf. vectorului $\underline{u}$ în
bază formată de coloanele mat. A^{*T} ,
 $\{\underline{a}_k^*\}_{k=0 \dots N-1}$

$\{\underline{a}_k^*\} = 0$ față de lui $\mathbb{R}^N$.

Mat. unitară $\Rightarrow A \cdot A^{*T} = I_N$

$$\begin{bmatrix} \underline{a}_0^T \\ \underline{a}_1^T \\ \vdots \\ \underline{a}_{N-1}^T \end{bmatrix} \begin{bmatrix} \underline{a}_0^* & \underline{a}_1^* & \dots & \underline{a}_{N-1}^* \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & \ddots & 1 \end{bmatrix}$$

$$\begin{cases} \underline{a}_0^T \cdot \underline{a}_0^* = 1 \\ \underline{a}_0^T \cdot \underline{a}_1^* = 0 \\ \dots \end{cases}$$

$\Rightarrow$ din condiția de unitaritate $\Rightarrow$

$$\underline{a}_i^T \cdot \underline{a}_j^* = \delta_{ij} = \begin{cases} 1 & i=j \\ 0 & i \neq j \end{cases}$$

$\Rightarrow$ asta implică ortonormalitatea ^{faței} mat. în
care se face descomp. ^{mat.} $\underline{u}$ în
bază.

Cazul 2-D

$$U = \{u(w, n)\}_{w, n=0 \dots N-1}$$

$$V = \{v(k, l)\}_{k, l=0 \dots N-1}$$

- Relația directă:
$$v(k, l) = \sum_w \sum_n a(w, n) \cdot u(w, n)$$

+ $k, l = 0, N-1$

- Rel. inversă:

$$u(w, n) = \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} q_{ke}^*(w, n) \cdot v(k, l) \quad \text{+ } w, n = \overline{0, N-1}$$

- transf. este definită de N^2 matrice de dim $N \times N$. (q_{ke}).

- mat. trebuie să respecte condiția de unitaritate. (ortogonalitate).

$$\sum_{w, n=0}^{N-1} q_{ke}(w, n) \cdot q_{k'l'}^*(w, n) = \delta(k-k', l-l') = \begin{cases} 1 & k=k' \text{ și } l=l' \\ 0 & \text{în rest.} \end{cases}$$

- pt. calculul unei elem. hai nevoie de N^2 înmulțiri.
pt. k, l fixate

⇒ complexitatea întregului calcul = $O(N^4)$.

- poate fi redusă la $O(N^3)$ în ipoteza de separabilitate.

O transf. bidim. unitară este separabilă dacă elementele $q_{ke}(w, n)$ pot fi scrise:

$$q_{ke}(w, n) = q_k(w) \cdot q_l(n) = q(k, w) \cdot q(l, n)$$

pt. + $k, l = \overline{0, N-1}$

adică dacă cele N^4 elem. care definesc transformarea se pot obține pornind de la elementele a numai 2 matrici $N \times N$,

respectiv: $A = \{a(k, w)\} \quad k, w = \overline{0, N-1}$
 $B = \{b(l, n)\} \quad l, n = \overline{0, N-1}$.

hi: $A^{-1} = A^*T$

hi: $B^{-1} = B^*T$

- în practică se alege $A = B$ (de fapt $B = A$):

$$q_{ke}(w, n) = a(k, w) \cdot a(l, n)$$

- în ipoteza separabilității + $A = B \Rightarrow$
 expresie transformatei Levin:

Cazul 2-D separabil:

$$v(k, l) = \sum_{w, n=0}^{N-1} a(k, w) \cdot u(w, n) \cdot a(l, n) \quad \text{+ } k, l = \overline{0, N-1}$$

$$u(w, n) = \sum_{k, l=0}^{N-1} a^*(k, w) \cdot v(k, l) \cdot a^*(l, n) \quad \text{+ } w, n = \overline{0, N-1}$$

$$\begin{cases} V = A \cdot U \cdot A^* \\ U = A^{*T} \cdot V \cdot A^* \end{cases} \quad \text{V} = \text{A} \cdot \text{U} \cdot \text{A}^T$$

Observații:

• - în cazul în care $U \in \mathbb{R}^{M \times N}$ atunci tr. unitară presupune / necesită 2 matrici:

$$\begin{aligned} A_M &\in \mathbb{C}^{M \times M} & A_M^{-1} &= A_M^{*T} \\ A_N &\in \mathbb{C}^{N \times N} & A_N^{-1} &= A_N^{*T} \end{aligned} \quad \Rightarrow$$

$$\begin{cases} V = A_M U \cdot A_N^T \\ U = A_M^{*T} V \cdot A_N^* \end{cases} \quad V \in \mathbb{C}_{M \times N}$$

• de ce tr. "separabile"?

$$V = A \cdot U \cdot A^T \Leftrightarrow V = [A \cdot (AU)^T]^T$$

$$AU = A \cdot \underbrace{\begin{bmatrix} | & | & | & | \end{bmatrix}}_{\text{mat. } U \text{ evidențind coloanele}}$$

$$AU = \begin{bmatrix} | & | & | & | \end{bmatrix}$$

- fiecare coloană din AU = transf. 1-D
a primei coloane din V .

- este o tr. separabilă - pt. că se aplică întâi
o tr. 1-D pt. coloane apoi una 1-D
pt. linii.

Bază de matrici

pt. 2 semnale 2-D: $F, G \in \mathbb{C}_{N \times N}$

definim: produsul scalar:

$$\langle F, G \rangle = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} f(u,v) \cdot g^*(u,v)$$

$$A^{*T} = \underbrace{\begin{bmatrix} \underline{a}_0^* & \underline{a}_1^* & \dots & \underline{a}_{N-1}^* \end{bmatrix}}_{\text{coloanele mat. } A^{*T}}$$

24.

- definim: $A_{ke}^* \stackrel{\text{def}}{=} \underline{a}_k^* \cdot \underline{a}_e^{*T}$
 $(N \times 1 \cdot 1 \times N) = (N \times N)$
 $\forall k, e = \overline{0, N-1}$

Relațiile tr. 2-D separabile devin:

$$\Rightarrow V(k,e) = \langle U, A_{ke}^* \rangle \quad \forall k, e = \overline{0, N-1}$$

tr. inv:

$$U = \sum_{k=0}^{N-1} \sum_{e=0}^{N-1} v(k,e) \cdot A_{ke}^*$$

Exemplu:

$$U = \begin{bmatrix} 3 & 7 \\ -1 & 5 \end{bmatrix}$$

$$A = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

mat. unei rotații cu 45° .

24.10.11

$$A^{*T} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

$$A \cdot A^{*T} = A \cdot A = \frac{1}{\sqrt{2}} \cdot \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \cdot \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$

$$V = A \cdot U \cdot A^{*T} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \cdot \begin{bmatrix} 3 & 7 \\ -1 & 5 \end{bmatrix} \cdot \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 2 & 12 \\ 4 & 2 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 14 & -10 \\ 6 & 2 \end{bmatrix} =$$

$$\Rightarrow V = \begin{bmatrix} 7 & -5 \\ 3 & 1 \end{bmatrix}$$

$$U = \begin{bmatrix} 3 & 7 \\ -1 & 5 \end{bmatrix} = 3 \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} + 7 \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} + (-1) \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} + 5 \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$

$$A^{*T} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad \underline{a}_0^* = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad \underline{a}_1^* = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

$$A_{00}^* = \underline{a}_0^* \cdot \underline{a}_0^{*T} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \cdot \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

$$A_{01}^* = \underline{a}_0^* \cdot \underline{a}_1^{*T} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \cdot \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & -1 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix}$$

$$A_{10}^* = \underline{a}_1^* \cdot \underline{a}_0^{*T} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ -1 \end{bmatrix} \cdot \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix}$$

$$A_{11}^* = \underline{a}_1^* \cdot \underline{a}_1^{*T} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ -1 \end{bmatrix} \cdot \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & -1 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

formăm o altă tablă a matricilor (din $\mathbb{R}^2$)

$\rightarrow$ + mat 2x2 poate fi scrisă ca o combinație liniară.

• verificăm ce proiecție are U pe mod. A_{ke}^* = coef. lui V_{ke}

$$\langle U, A_{00}^* \rangle = \left\langle \begin{bmatrix} 3 & 7 \\ -1 & 5 \end{bmatrix}, \frac{1}{2} \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \right\rangle = \frac{3 \cdot 1 + 7 \cdot 1 + (-1) \cdot 1 + 5 \cdot 1}{2} = 7 = V(0,0).$$

$$\langle U, A_{01}^* \rangle = -5 = V(0,1).$$

• verific. că calculăm:

$$\sum_k \sum_{l=0}^{N-1} V(k,l) \cdot A_{ke}^* = 7 \cdot \frac{1}{2} \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} +$$

$$+ (-5) \cdot \frac{1}{2} \begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix} + 3 \cdot \frac{1}{2} \begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix} + 1 \cdot \frac{1}{2} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} = \begin{bmatrix} 3 & 7 \\ -1 & 5 \end{bmatrix} = U.$$

- utilizare < analiza - cu te. Fourier
compresia.

Proprietăți ale transf. unitare

1. Conservarea energiei.

$$\begin{aligned} 1-D: E_v &= \|v\|^2 = v^{*T} \cdot v = (A \cdot u)^{*T} \cdot A \cdot u = \\ &= u^{*T} \cdot \underbrace{A^{*T} \cdot A}_{I_n} \cdot u = u^{*T} \cdot u = \|u\|^2 = E_u. \end{aligned}$$

2. Compactarea energiei.

- în general transformarea este aleasă a.i.

energia să fie distribuită inegal în spațiul transformatei, chiar dacă ea era uniform distribuită în spațiul original.

- în plus, transformatele sînt alese a.i. coef. în spațiul transformatei să fie mai decorelați decât în spațiul original.

- transformata optimă care compactează maximum de energie într-un nr. de coef. dați n' are în același timp decorelați complet coeficienții în spațiul transformatei s.n. fr. Karhunen-Loève

Transformata Fourier Discretă (TFD).

Cazul 1-D:

$$\underline{u} = [u(0) \dots u(N-1)]^T$$

$$\underline{v} = [v(0) \dots v(N-1)]^T$$

$$\text{unde } v(k) = \sum_{n=0}^{N-1} u(n) \cdot \exp\left(-\frac{2\pi j kn}{N}\right) \quad \forall k=0 \dots N-1$$

TFD:

$$u(n) = \frac{1}{N} \sum_{k=0}^{N-1} v(k) \cdot \exp\left(\frac{2\pi j kn}{N}\right) \quad \forall n=0 \dots N-1$$

- astfel definită TFD nu este unitară
(nu conservă energia).

$\Rightarrow$ TFD unitară:

$$\begin{cases} v(k) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} \dots \\ u(n) = \frac{1}{\sqrt{N}} \sum \dots \end{cases}$$

$$\underline{F} = \{f(k, n)\}_{k, n=0 \dots N-1}$$

$$f(k, n) = \frac{1}{\sqrt{N}} \exp\left(-\frac{2\pi j kn}{N}\right)$$

$\Rightarrow$ TFD se poate scrie matricial:

$$\begin{cases} \underline{v} = \underline{F} \cdot \underline{u} \\ \underline{u} = \underline{F}^* \underline{v} \end{cases} \text{ ținând cont de } \underline{F} = \underline{F}^T.$$

Obs. 3 algoritmi rapizi FFT

- pt. cazul 1D: $O(N^2)$ se reduce la $O(N \log_2 N)$.

- pt. cazul 2D:

$$\underline{U} = \{u(w, n)\}_{w, n=0 \dots N-1}$$

$$\Rightarrow \text{prin TFD } \underline{V} = \{v(k, l)\}_{k, l=0 \dots N-1}.$$

$$v(k, l) = \frac{1}{N} \sum_{w=0}^{N-1} \sum_{n=0}^{N-1} u(w, n) \cdot \exp\left[-\frac{2\pi j (kw + ln)}{N}\right]$$

$$\forall k, l = 0 \dots N-1.$$

cazul
separabil.

$$\underline{V} = \underline{F} \cdot \underline{U} \cdot \underline{F}$$

$$u(w, n) = \frac{1}{N} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} v(k, l) \cdot \exp\left[\frac{2\pi j (kw + ln)}{N}\right]$$
$$\forall w, n = 0 \dots N-1.$$

$$\underline{U} = \underline{F}^* \cdot \underline{V} \cdot \underline{F}^*$$

- FFT: reduce $O(N^3) \rightarrow O(N^2 \log_2 N)$.

de citit: pag. 141 ÷ 150.
+ TCD (oct).

COMPRESIA IMAGINILOR

Term. de compresie se referă la totalitatea metodelor ce au drept scop reducerea cantității de date necesare pt. stocarea și/sau transmiterea imaginilor.

Clasificare:

1. metode de compresie la nivel de pixel.
 - = met. care nu țin cont de corelația între pixelii vecini pe linii și coloane.
 - sînt tehnici de compresie fără pierdere de inf., adică imaginea originală poate fi refăcută perfect din cea comprimată (codată).
 - ex: - codarea Huffman, LZW (Lempel-Ziv-Walsh), RLE (Run Length Encoding), WBS (White-Block Skipping).

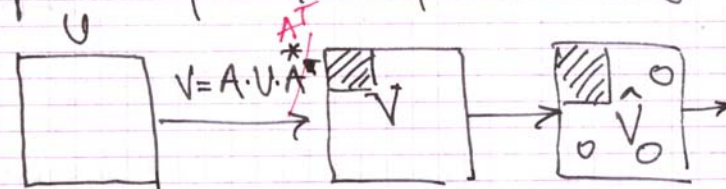
2. Metode de codare predictivă.

- sînt met. care realizează compresia pe baza exploatarea corelației între pixelii vecini pe linii și coloane.
- ex: DPCM (Differential Pulse Code Modulation)

3. Metode de compresie cu transformate.

- bazate pe scrierea imaginii într-o altă fază prin aplicarea unei transf. unitare (integrabile) a.î. energia în spațiul transformat să fie concentrată într-un nr. cît mai mic de coef. Prin eliminarea coef. de energie joasă (care se dovedește a fi în

nr. cît mai mare) și stocarea numai a acelor de energie mare, se obțin factori de compresie importanți cu pierderi neglijabile.

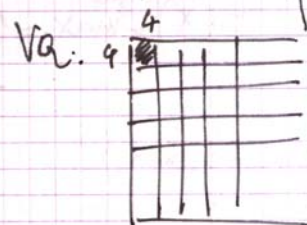


$$\hat{U} = A^T \cdot \hat{V} \cdot A$$

JPEG - factor de compresie?

4. Alte metode:

- VQ (Vector Quantization) - cuantizare vectorială
- codare cu fractale.



- pt. foarte multe imagini - se face clustering pe blocuri 4x4.

(de ex. în 256 de clase de blocuri 4x4 - și

pt. fiecare clasă se alege un vector reprezentativ - de ex. media vectorilor dintr-o clasă respectivă).

Metode de codare la nivel de pixel

1. Codarea Huffman

$$[S] = \{s_1, \dots, s_N\} \quad P(s_1) = p_1 \dots P(s_N) = p_N$$

p_i - cunoscut.

$$H(S) = - \sum_{i=1}^N p_i \cdot \log p_i \rightarrow \text{inf. medie \& limit. (biti/simbol)}.$$

$$H(S) = \max \Leftrightarrow p_1 = p_2 = \dots = p_N = \frac{1}{N}.$$

$[X] = \{0, 1\}$ - darem să codăm simbolurile sursei S cu simbolii din sursa X .
a.i. entropia sursei secundare să fie maximizată.

Eficiența codului este dată de lungimea medie:

$$\bar{l} = \sum_{i=1}^N l_i \cdot p_i \quad l_i = \text{lung. codului alocat simb. } S_i$$

$$\bar{l} = \frac{H(S)}{H(X)} \quad \bar{l} = \bar{l}_{\min} \Leftrightarrow H(X) = \max. = 1.$$

limita min. teoretică.
(cod absolut optimal)

$$\bar{l}_{\text{realizat}} \geq \bar{l}_{\min}$$

ex:

	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8
	0,15	0,0625	0,14	0,0625	0,21	0,0625	0,0625	0,25
S_8	0,25			0,25				
S_5	0,21			0,21				
S_1	0,15			0,15				
S_3	0,14			0,14				

* $\left. \begin{array}{l} 0,25 \\ 0,25 \\ 0,25 \\ 0,21 \end{array} \right\} \rightarrow 0,29$

28

$$\left. \begin{array}{l} S_2 \quad 0,0625 \\ S_4 \quad 0,0625 \\ S_6 \quad 0,0625 \\ S_7 \quad 0,0625 \end{array} \right\} \rightarrow \left. \begin{array}{l} 0,15 \\ 0,0625 \\ 0,0625 \end{array} \right\} \rightarrow \left. \begin{array}{l} 0,125 \\ 0,125 \end{array} \right\} \rightarrow 0,25$$

$$* \dots \left. \begin{array}{l} 0,46 \\ 0,29 \\ 0,25 \end{array} \right\} \rightarrow \left. \begin{array}{l} 0,54 \\ 0,46 \end{array} \right\} \rightarrow 1.$$

S_8 : 10
 S_5 : 11
 S_1 : 000
 S_3 : 001
 S_2 : 0110
 S_4 : 0111
 S_6 : 0100
 S_7 : 0101.

$$\begin{aligned} \bar{l} &= \sum_i l_i \cdot p_i = 2 \cdot 0,25 + 2 \cdot 0,21 + 3 \cdot (0,15 + 0,14) + 4 \cdot 4 \cdot 0,0625 = \\ &= 0,92 + 0,84 + 1 = 2,76 \text{ biti/simbol.} \end{aligned}$$

Dacă n-am fi avut cont de probabilități, am fi codat cele 8 simbo. pe 3 biti $\rightarrow \bar{l} = 3$.

-dezav: - probabilitățile nu sunt cunoscute $\Rightarrow$ trebuie calculată histograma imaginii.
 - trebuie stocată și tabela de codare.

Obs.: dacă histograma este uniformă $\Rightarrow$ compresie 1 la 1.

LZW - Lempel - Ziv - Welch

2. Metoda Ziv-Lempel

- fol. o tabelă de codare care este construită pe măsură ce firul este parcurs, atât la codare cât și la decodare. $\Rightarrow$ nu este necesară stocarea sau transmiterea tabelului de codare.

ex: 1 0 1 1 1 0 0 0 0 1 1 0 1 1 1 0 0 1 1 0 ...

curv. | cod.

0	0
1	1
...	...
10	2
01	3
11	4
110	5
00	6
000	7
011	8
101	9
111	10
1100	11
0110	12

tabelul de codare

inicial:

0:0
1:1.

1 0 1 4 0 6 3 2 4 5 8 ...

Obs. se folosește un nr. fix pt. lungimea codului.

Obs. spre sfârșit - un nr. mare de bitti se va regăsi printre cuvintele din tabelă și astfel vor fi codate cu doar 8 bitti de ex. (deci fixăm codul pe 8 bitti).

- luăm un bit: 1 - 7 în tabelă?
dacă 7 - mai luăm un bit.
dacă nu 7 - îl trecem în tabelă.

- în firul codat - scriem doar prefixul cuvântului. care nu se află în tabelă.
- începem apoi cu un bit mai înainte de prefix.

Decodarea:

curv. | cod

0	0
1	1
10	2
01	3
11	4
110	5
00	6
000	7
011	8
101	9
111	10
1100	11
0110	12

1 0 1 4 0 6 3 2 4 5 8 ...

(1) 1x, urmează 0 $\Rightarrow x=0$.

(0) 10x, urmează 1 $\Rightarrow x=1$.

(1) 101x - urmează 4
1x $\Rightarrow x=1$.

(4) 10111x $\Rightarrow$ 101110y.
0yx $\Rightarrow y=0$.

(6) $\rightarrow$ 101110002.
01x $\Rightarrow x=0$.

(3) 1011100001x
de tip 10y

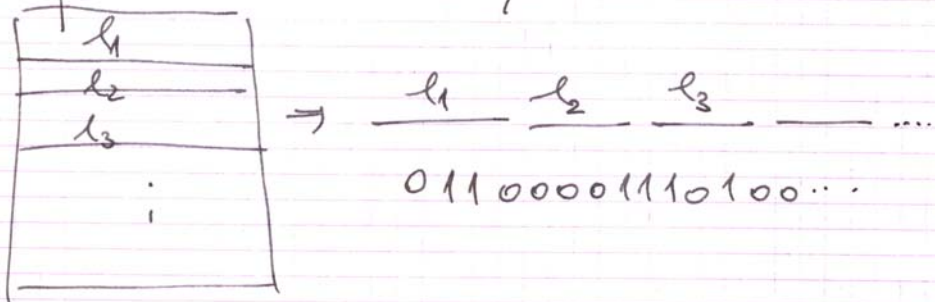
(2) 101110000110y
01x

1 0 1 1 1 0 0 0 0 1 1 1 1 0 0 1 1 u

3. Codarea RLE.

a) Imagini binare.

- imaginea se transformă într-un fir unidimensional prin concatenarea liniilor / coloanelor acesteia



- ~~apare~~ în firul c

primul elem. al firului codat este primul el. al firului decodat. (copiem. 0).
după care se scrie în firul codat lungimea fiecărui subfir constant al firului.

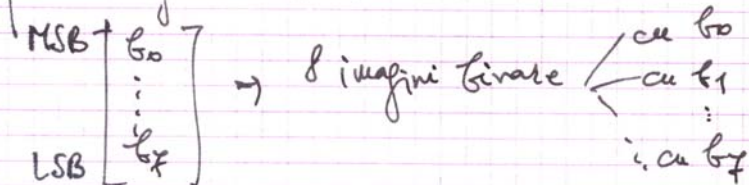
01243112

- pt. imagini prin fax.

6) Imagini în tonuri de gri.

⇒ codare RLE pe plane de bîte

- pt. o imag. cu 256 de nivele :



- imaginea corespunzătoare nivelului MSB va fi comprimată cel mai bine cu RLE.

- imaginea LSB = imagine de "pruci" - care aproape că nu merită să fie codată.

- factorul de compresie !!

Parametri de evaluare a calitatii compresiei

1. Raportul de compresie : $C = \frac{\text{nr. de bîte necesare pt. stocarea imaginii originale}}{\text{nr. de bîte necesare pt. stocarea imaginii comprimate}}$
BMP: $N \times M \times 8$
JPG: $200 \times 500 = 60 \text{ KB}$

2. Numărul de bîte/pixel:

$$N = \frac{\text{nr. de bîte nec. pt. stoc. imag. comprimate}}{\text{nr. total de pixeli.}}$$

$$N = N_{bp} = [bpp].$$

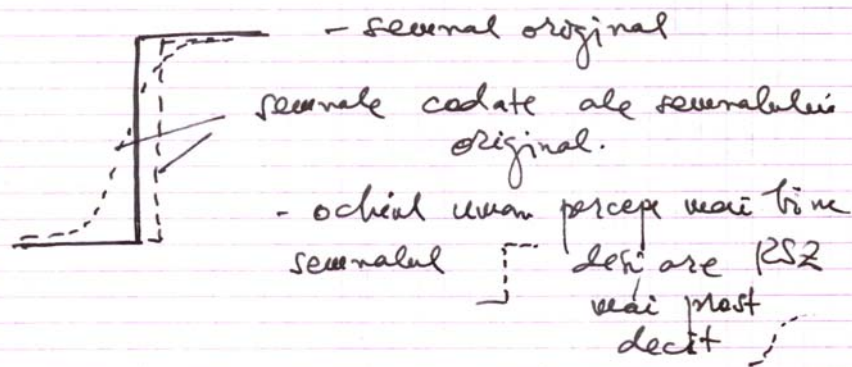
3. Pt. metode de compresie cu pierderi: evaluarea calitatii imag. comprimate se face prin RSZ (Raportul Semnal-Zgomot).

$$RSZ = 10 \log \frac{\sum_i \sum_j I_{orig}^2(i,j)}{\sum_i \sum_j (I_{orig}(i,j) - I_{codat}(i,j))^2}$$

$$RSZ = \frac{\text{energia imaginii originale}}{\text{energia de zgomot (care apare prin codare)}}$$

- * pt. met. cu pierderi : C și RSZ contestă. (Să fie 'veri amindono').

Obs: RSZ = o măs. obiectivă și nu exprimă subiectivitatea ochiului uman.



Metode de codare predictive

Cazul 1D:

pp. un seversal $u(n)$
 pe baza a N^M es. ale seversalului sa se prezice esantionul urmator al seversalului.

$$u(n), u(n-1), u(n-2), \dots, u(n-M+1), \dots, u(0)$$

$$\hat{u}(n+1) = \sum_{i=0}^{M-1} a_i \cdot u(n-i), \text{ unde } a_i \text{ s\u00ednt}$$

nieste coef. care s\u00ednt optimizati a.i. estimarea esantionului urm. sa fie cit mai buna folosind un criteriu de eroare.

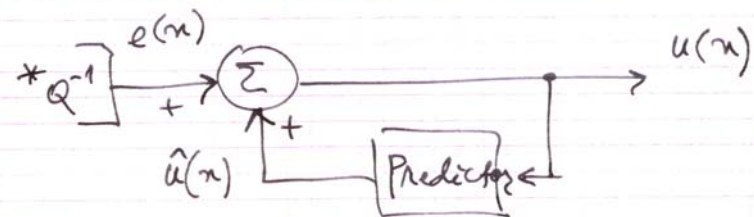
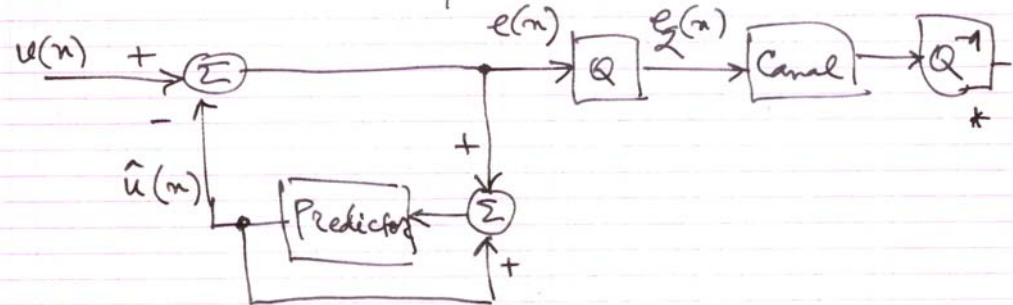
Obs: tehnica se bazeaza pe faptul c\u00e2 esant. s\u00ednt corelate,

$$e(n) = \text{eroarea}$$

$$e(n) = u(n+1) - \hat{u}(n+1).$$

Obs: de obicei criteriul: eroarea patratic\u00e2 medie minim\u00e2 (pe baza functiei de autocorelatie a seversalului)

Schema de codare cu predictie:



si cuantizare.
 Q = bloc de codare (Q^{-1} = decodare).

Obs: merge o codare Huffman

Determinarea coef.:

$$\hat{u}(n+1) = \sum_{i=0}^{M-1} a_i \cdot u(n-i).$$

$$e(n) = u(n+1) - \hat{u}(n+1) = u(n+1) - \sum_{i=0}^{M-1} a_i \cdot u(n-i)$$

$$J = \frac{|e(n)|^2}{e^2(n)} \rightarrow \text{minim\u00e2} \rightarrow a_0, a_1, \dots, a_{M-1} = ?$$

$$\frac{\partial J}{\partial a_i} = 0 \text{ pt. } \forall i = 0, M-1 \rightarrow a_i.$$

$$\frac{\partial J}{\partial a_i} = \frac{\partial \overline{e^2(n)}}{\partial a_i} = \frac{\partial \left[u(n-1) - \sum_{k=0}^{M-1} a_k \cdot u(n-k) \right]^2}{\partial a_i} =$$

$$= -2 \left(u(n-1) - \sum_{k=0}^{M-1} a_k \cdot u(n-k) \right) \cdot u(n-i) = 0$$

$$\Rightarrow \overbrace{u(n+1) u(n-1)}^{n-i?} = \sum_{k=0}^{M-1} a_k \cdot \overbrace{u(n-k) \cdot u(n-i)}^{i=0, M-1}$$

dar $u(n) \cdot u(n-k) = R_u(k) = \text{funct. de autocorelație.}$

$$\Rightarrow \begin{cases} R_u(i+1) = \sum_{k=0}^{M-1} a_k \cdot R_u(i-k) & i=0, M-1 \end{cases}$$

= sistem de M ecuații cu M necunoscute

$$\Rightarrow a_k = \dots \quad k=0, M-1$$

Matriceal se poate scrie:

$$\underbrace{\begin{bmatrix} R_u(0) & R_u(1) & \dots & R_u(M-1) \\ R_u(-1) & R_u(0) & \dots & R_u(M-2) \\ \dots & \dots & \dots & \dots \\ R_u(-M+1) & R_u(-M+2) & \dots & R_u(0) \end{bmatrix}}_R \underbrace{\begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{M-1} \end{bmatrix}}_a = \underbrace{\begin{bmatrix} R_u(1) \\ ? \\ \vdots \\ R_u(M) \end{bmatrix}}_n$$

$$\underline{a} = R^{-1} \cdot \underline{n}$$

- funct. de autocorelație este pară $\Rightarrow R_u(-1) = R_u(1)$.

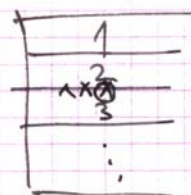
Deci $u = \text{zgomot alb.} \Rightarrow R_u(i) = 0 \text{ pt. } i \neq 0.$

$$R_u(k) = \begin{cases} \sigma_u^2 & k=0 \\ 0 & k \neq 0 \end{cases}$$

$\Rightarrow$ coef. $a_k = 0 \Rightarrow$ crearea de predicție = esențial
 $\Rightarrow$ nu se poate face nici o predicție
 pt. zgomotul alb.

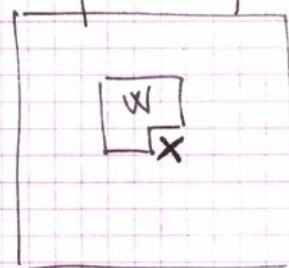
(predicția în cazul acesta = 0, adică cea
 mai bună predicție = media semn.
 (zgomotului)).

Cazul 2D.



un pixel va depinde doar
 de vecinătăți dintr-un
 obs. se pierde astfel corelația spațială
 (pe coloane) ale pixelilor

Sau.: putem defini o vecinătate bidimensională: W .



$$\text{pt. } x: \hat{u}(n, x) = \sum_{i,j \in W} a_{ij} \cdot u(n-i, x-j)$$

Compresia cu transformate
Transformarea Karhunen-Loève

Cazul 1-D.

$$\underline{u} = [u(0) \dots u(N-1)]^T \quad \text{stationar în sens larg.}$$

dorim: $A = [\underline{e}_0 \ \underline{e}_1 \ \dots \ \underline{e}_{N-1}]^T$
 în det. transf. unitară $(A^{-1} = A^T)$.

a.î: $\underline{v} = A \cdot \underline{u} \quad \underline{v} = [v(0) \dots v(N-1)]^T$

Să îndeplinească urm. prop.: dacă înlocuim ultimele valori ale vect. $\underline{v}$ cu niște constante atunci eroarea indusă în semnalul original să fie minimă. Dorim ca începând de la rangul "m" să înlocuim $c(u) = v(u) \dots c(N-1) = v(N-1)$ și obținem:

$$\hat{\underline{v}} = [v(0) \dots v(u-1) \ c(u) \ \dots \ c(N-1)]^T$$

prin transf. inversă:

$$\hat{\underline{u}} = A^T \cdot \hat{\underline{v}}$$

$\Rightarrow A = ?$ a.î. $\varepsilon = \|\underline{u} - \hat{\underline{u}}\|^2 \rightarrow$ minimă.
 (er. pătratică medie).

$$\underline{u} = A^T \cdot \underline{v} = [\underline{e}_0 \ \underline{e}_1 \ \dots \ \underline{e}_{N-1}] \cdot \begin{bmatrix} v(0) \\ \vdots \\ v(N-1) \end{bmatrix}$$

$$\underline{u} = \sum_{i=0}^{N-1} v(i) \cdot \underline{e}_i$$

$$\hat{\underline{u}} = \sum_{i=0}^{u-1} v(i) \underline{e}_i + \sum_{i=u}^{N-1} c(i) \cdot \underline{e}_i$$

$$\underline{u} - \hat{\underline{u}} = \sum_{i=u}^{N-1} (v(i) - c(i)) \cdot \underline{e}_i$$

$$\varepsilon = \|\underline{u} - \hat{\underline{u}}\|^2 = (\underline{u} - \hat{\underline{u}})^T \cdot (\underline{u} - \hat{\underline{u}}) =$$

$$= \left(\sum_{i=u}^{N-1} (v(i) - c(i)) \cdot \underline{e}_i^T \right) \left(\sum_{j=u}^{N-1} (v(j) - c(j)) \cdot \underline{e}_j \right) =$$

$$\varepsilon = \sum_{i=u}^{N-1} \sum_{j=u}^{N-1} (v(i) - c(i)) (v(j) - c(j)) \cdot \underline{e}_i^T \cdot \underline{e}_j$$

$A =$ transf. unitară.

$$\underline{e}_i^T \cdot \underline{e}_j = \delta_{ij} \quad (\text{veea ca transf. să fie unitară})$$

$$\Rightarrow \varepsilon = \sum_{i=u}^{N-1} (v(i) - c(i))^2$$

- Dorim să aflăm constantele respective?

- pt. a det. $c(k)$: $\frac{\partial \varepsilon}{\partial c(k)} = 0 \quad \forall k = u \dots N-1$
 $\Rightarrow c(k)$

$$\frac{\partial \varepsilon}{\partial c(k)} = -2(v(k) - c(k)) = 0 \Rightarrow \boxed{c(k) = v(k)}$$

$$\forall k = u \dots N-1.$$

$\rightarrow$ val. constantelor = val. medie statistică a eșantioanelor înlocuite

$$\varepsilon = \sum_{k=m}^{N-1} \overline{(v(k) - \bar{v}(k))^2}$$

$$v(k) = \langle \underline{\varphi}_k, \underline{u} \rangle = \underline{\varphi}_k^T \cdot \underline{u} = \underline{u}^T \cdot \underline{\varphi}_k$$

proiecția lui $\underline{u}(k)$ pe $\underline{\varphi}_k$

$$\Rightarrow \varepsilon = \sum_{k=m}^{N-1} \overline{(v(k) - \bar{v}(k))(v(k) - \bar{v}(k))} =$$

$$= \sum_{k=m}^{N-1} \overline{(\underline{\varphi}_k^T \cdot \underline{u} - \underline{\varphi}_k^T \cdot \bar{\underline{u}})(\underline{u}^T \cdot \underline{\varphi}_k - \bar{\underline{u}}^T \cdot \underline{\varphi}_k)}$$

$$\bar{v}(k) = \overline{\underline{\varphi}_k^T \cdot \underline{u}} = \underline{\varphi}_k^T \cdot \bar{\underline{u}} \quad (\underline{\varphi}_k = \text{vector determinist})$$

$$\varepsilon = \sum_{k=m}^{N-1} \underline{\varphi}_k^T (\underline{u} - \bar{\underline{u}}) (\underline{u} - \bar{\underline{u}})^T \cdot \underline{\varphi}_k =$$

$$= \sum_{k=m}^{N-1} \underline{\varphi}_k^T \cdot \underbrace{(\underline{u} - \bar{\underline{u}})(\underline{u} - \bar{\underline{u}})^T}_{\text{mat. } N \times N} \cdot \underline{\varphi}_k$$

s. not. $K_u =$ matricea de covarianță a lui $\underline{u}$.

$$\varepsilon = \sum_{k=m}^{N-1} \underbrace{\underline{\varphi}_k^T}_{1 \times N} \cdot \underbrace{K_u}_{N \times N} \cdot \underbrace{\underline{\varphi}_k}_{N \times 1}$$

$$K_u = \overline{(\underline{u} - \bar{\underline{u}})(\underline{u} - \bar{\underline{u}})^T} \quad \begin{bmatrix} \vdots \\ \vdots \end{bmatrix} \begin{bmatrix} \dots \end{bmatrix}$$

$$K_u(i,j) = \overline{(u(i) - \bar{u}(i))(u(j) - \bar{u}(j))}$$

reprez. covarianța dintre al i -lea și al j -lea elem. al vectorului $\underline{u}$.

(arată cât de corelate statistic sînt eg. i și j între ele).

$$K_u(i,i) = \overline{(u(i) - \bar{u}(i))^2} = \sigma_u^2(i) = \sigma_u^2$$

(def. între mat. de corelație și mat. de covarianță: este o constantă).

$\begin{bmatrix} \text{valori} & \dots \\ & \text{mari} \end{bmatrix}$ - pt. un semnal real
← teoretic

den ce în ce mai mici
cu cît sînt mai departe de diag. principală.

$$\varepsilon = \sum_{k=m}^{N-1} \underline{\varphi}_k^T \cdot K_u \cdot \underline{\varphi}_k$$

$$\underline{\varphi}_k = ? \text{ a.î. } \varepsilon \rightarrow \min.$$

condiție suplimentară: $\underline{\varphi}_k^T \cdot \underline{\varphi}_k = 1$. (constrîngere).

Veri. minimizare cu constrîngere. pt. $\forall k = m, N-1$

ex: $f(x)$ - x care min. $f(x)$ a.î. $g(x) = a$.

Se minimizează: $\Psi(x) = f(x) + \lambda(g(x) - a)$.

$\lambda =$ multiplicatorul Lagrange

- minimizăm:

$$\Psi = \sum_{k=u}^{N-1} \underline{\varphi}_k^T \cdot K_u \cdot \underline{\varphi}_k + \sum_{k=u}^{N-1} \lambda_k (\underline{\varphi}_k^T \cdot \underline{\varphi}_k - 1).$$

Ex: un vector $\underline{\varphi} = \begin{bmatrix} \varphi(0) \\ \vdots \\ \varphi_{N-1} \end{bmatrix}$

$h_i \quad E(\underline{\varphi}) \in \mathbb{R}$

expresie scalară care dep. de val. vect. $\underline{\varphi}$

$\underline{\varphi} = ?$ a.i. E min.:

$$\Rightarrow \frac{\partial E}{\partial \varphi_0} = 0 \quad \frac{\partial E}{\partial \varphi_1} = 0 \quad \dots \quad \frac{\partial E}{\partial \varphi_{N-1}} = 0 \Rightarrow$$

$$\begin{bmatrix} \frac{\partial E}{\partial \varphi_0} \\ \vdots \\ \frac{\partial E}{\partial \varphi_{N-1}} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

$$\Rightarrow \nabla_{\underline{\varphi}} E = \begin{bmatrix} \frac{\partial E}{\partial \varphi_0} \\ \vdots \\ \frac{\partial E}{\partial \varphi_{N-1}} \end{bmatrix}$$

$\nabla_{\underline{\varphi}} E$ = gradientul expresiei E în raport cu vect. $\underline{\varphi}$

dacă $A = 0$ matrice simetrică $B = B^T$

$$\boxed{\nabla_{\underline{\varphi}} (\underline{\varphi}^T \cdot B \cdot \underline{\varphi}) = 2 \cdot B \cdot \underline{\varphi}} \quad \text{de dem. !!}$$

$$\nabla_{\underline{\varphi}} (\underline{\varphi}^T \cdot \underline{\varphi}) = 2 \underline{\varphi} \quad (B = I_N).$$

$$\underline{\varphi} = \sum_{k=u}^{N-1} \underline{\varphi}_k^T \cdot K_u \cdot \underline{\varphi}_k + \sum_{k=u}^{N-1} \lambda_k (\underline{\varphi}_k^T \cdot \underline{\varphi}_k - 1).$$

$$\nabla_{\underline{\varphi}_k} \Psi = 0 \Rightarrow \underline{\varphi}_k \quad \forall k = \overline{u, N-1}.$$

având în vedere $K_u = K_u^T \quad K_u(i,j) = K_u(j,i).$

$$\Rightarrow \nabla_{\underline{\varphi}_k} \Psi = 2 \cdot K_u \underline{\varphi}_k + 2 \lambda_k \cdot \underline{\varphi}_k = 0.$$

$$\Rightarrow \boxed{K_u \cdot \underline{\varphi}_k = -\lambda_k \cdot \underline{\varphi}_k}$$

- vectorii $\underline{\varphi}_k$ care verif. ec. păt. medec ca constrângere să fie ortogonali și de modul unitari.:

dacă notăm $-\lambda_k = \lambda'_k$

$$\boxed{K_u \cdot \underline{\varphi}_k = \lambda'_k \cdot \underline{\varphi}_k} \quad \forall k = \overline{u, N-1}.$$

$\Rightarrow \underline{\varphi}_k$ = vect. proprii ai mat. de covarianță a semnalului.

Obs. relația se păstrează pt. $\forall u$.

pt. $u=0 \Rightarrow$ expresia se poate generaliza și pt. $u=0. (\forall k = \overline{0, N-1}).$

=> practic te. KL nu este unică. Ea depinde de statistica semnalului, adică de mat. de auto-covarianță a semnalului.

Val. λ'_k sînt valorile proprii corespunzătoare (valorile) vectorilor proprii $\underline{\psi}_k$.

$$\varepsilon = \sum_{k=u}^{N-1} \underbrace{\underline{\psi}_k^T \cdot K_u \cdot \underline{\psi}_k}_{\lambda'_k} = \sum_{k=u}^{N-1} \underbrace{\lambda'_k \cdot \underbrace{\underline{\psi}_k^T \cdot \underline{\psi}_k}_{=1}}_{\lambda'_k} =$$

$$= \sum_{k=u}^{N-1} \lambda'_k = \text{suma } \text{resp.} \text{ valorilor proprii ale vectorilor din mat. } A \text{ care au fost înlocuiți.}$$

$$A = [\underline{\psi}_0 \ \underline{\psi}_1 \ \dots \ \underline{\psi}_{N-1}]^T$$

cu condiția: $\lambda'_0 > \lambda'_1 > \dots > \lambda'_{N-1}$

Obs. ordinea vectorilor $\underline{\psi}_k$ este în ordinea descrescătoare a val. proprii ale vect $\underline{\psi}_k$ a. i. eroarea să fie minimă.

$$K_v = \overbrace{(\underline{v} - \bar{\underline{v}})(\underline{v} - \bar{\underline{v}})^T} = \overbrace{(A \cdot \underline{u} - A \cdot \bar{\underline{u}})} = (A \cdot \underline{u} - A \cdot \bar{\underline{u}})^T =$$

$$= A \cdot (\underline{u} - \bar{\underline{u}}) \cdot (\underline{u} - \bar{\underline{u}})^T \cdot A^T = A \cdot K_u \cdot A^T$$

$$\text{dar } K_u \cdot A^T = K_u \cdot [\underline{\psi}_0 \ \underline{\psi}_1 \ \dots \ \underline{\psi}_{N-1}]$$

$$K_u \cdot A^T = \begin{bmatrix} K_u \cdot \underline{\psi}_0 & K_u \cdot \underline{\psi}_1 & \dots & K_u \cdot \underline{\psi}_{N-1} \\ \parallel & \parallel & & \parallel \\ \lambda'_0 \underline{\psi}_0 & \lambda'_1 \underline{\psi}_1 & \dots & \lambda'_{N-1} \underline{\psi}_{N-1} \end{bmatrix}$$

$$\Rightarrow A = \begin{bmatrix} \underline{\psi}_0^T \\ \vdots \\ \underline{\psi}_{N-1}^T \end{bmatrix}$$

$$A \cdot K_u \cdot A^T = \begin{bmatrix} \underline{\psi}_0^T \\ \vdots \\ \underline{\psi}_{N-1}^T \end{bmatrix} \cdot [\lambda'_0 \underline{\psi}_0 \ \dots \ \lambda'_{N-1} \underline{\psi}_{N-1}] =$$

$$= \begin{bmatrix} \lambda'_0 & 0 \\ 0 & \ddots \\ 0 & & \lambda'_{N-1} \end{bmatrix} \Rightarrow \text{coef. în spațiul transformatei sînt perfect decorelați.}$$

Tr. K.L nu se folosește în practică:

- ea nu este unică ci este specifică fiecărei clase de semnale.
- nu există algoritmi rapizi pt. implementare

În practică este înlocuită cu transf. suboptimale. Pt. imagini - cea mai potrivită este tr. codinilor discreți.

MORFOLOGIE MATEMATICĂ

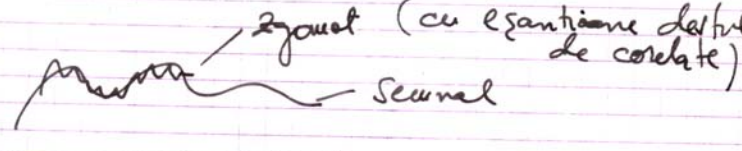
- morfologie = știința formelor.

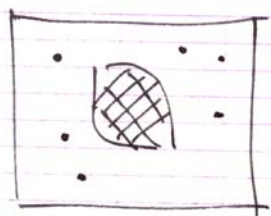
[preprocesarea imaginii = curățire, etc.]

în 1-D - o formă poate arăta astfel _____

în 2-D - 

imagini $\left\{ \begin{array}{l} \text{binare - nivele } \{0, 1\} \text{ sau } \{0, 255\} \\ \text{gră} \\ \text{color} \end{array} \right. \Rightarrow \text{cele mai "clare" forme.}$

pt. 1-D : 

pt. 2-D :  - 2gouetul nu mai este la fel de bine corelat. (sau chiar de loc)

- pt. un pixel - la Lore și Piper, nu $\exists$ o distribuție gaussiană

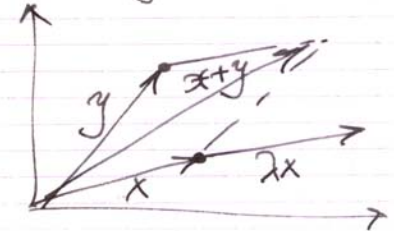
- pt. imagini binare - cu forme alterate
nu pot fi preluate cu filtre pe valori, ci cu filtre de formă. 

- mulțimea punctelor $A \subset E$ - mulțimea tuturor punctelor

E = o mulțime dotată cu o structură topologică (știu ce e un punct și vecinii săi); în plus considerăm că este un spațiu vectorial adică știu : $x+y$ sau λx

$x: (x_1, x_2)$

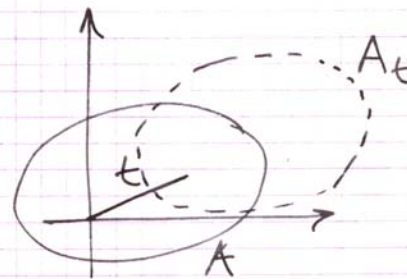
$y: (y_1, y_2)$



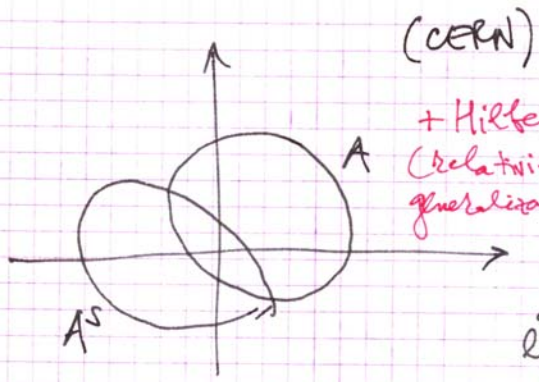
- dacă avem o mulțime A , atunci :

$A_t = \{y \mid \exists x \in A : y = x+t\}$
punct.

$A_t = \{y \mid \exists x \in A : y = x+t\}$
punct din plan



$A, \Rightarrow A^S = \text{mulțimea simetrică lui } A, \text{ în raport cu originea :}$



(CERN)
 + Hilbert
 (relativitatea generalizată)
 Minkowski - a fost profesorul lui Einstein
 Articolul despre relativitatea restrânsă a fost scris de prima lui nevastă, de la care l-a comparat.

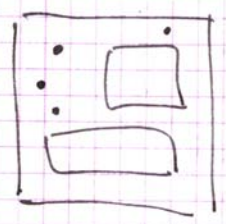
- În începutul sec. Minkowski - a introdus operații cu mulțimi: $A+B$, $A-B$.

$$A+B = \{z \mid \exists a \in A \wedge b \in B : z = a+b\}$$

$$A-B = \{z \mid \dots : z = a-b\}$$

- Matheron ; Jean Serra.

- dacă avem o imagine binară :



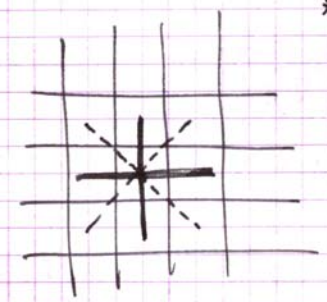
punctele + zonele pe care vrem să le îndepărtăm

Obs: putem număra pixelii și calcula ariile obiectelor din imagine - și apoi să înlocuim obiectele de arie mică.

Dilatare și erodare

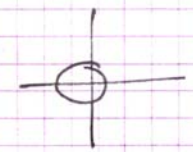
- avem o mulțime A = punctele din imagine = spațiu, topologic
 și o mult. B - numită element structural

- erodarea (sau dilatarea) se pot face cu elemente structurale diferite
- morfologia - pe mulțimi continue
- imaginile sunt mulțimi discrete, oricât de mare ar fi rezoluția.
- pixelii pot fi dispuși în rețea carteziană sau în rețea hexagonală:



— = vecinătate V_4
 — + --- = V_8 .

- în cazul continuu = vecinătatea = un disc



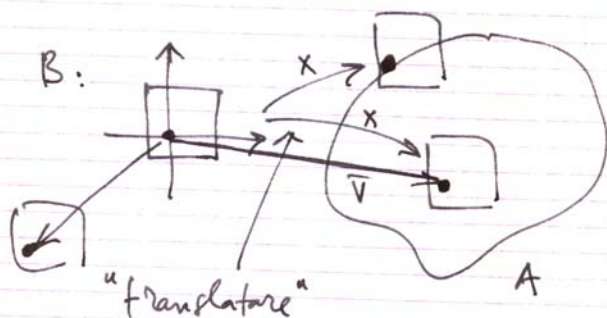
$$\|x - x_0\| < \varepsilon$$

in rețea: $d(x,y) = \max(|x_1 - y_1|, |x_2 - y_2|)$
 x, y = pixeli. (dist. pt. 8 vecini)
 sau $d_4 = |x_1 - y_1| + |x_2 - y_2|$

Obs: după erodare obținem "erodată" mulțimii A prin elem. structural B .

erodată: $A \ominus B$
 dilatăta: $A \oplus B$.

$$A \ominus B =$$



$$B \neq \emptyset$$

$$A \neq \emptyset$$

- punctul • are calitatea că B-ul translataat cu vectorul de translatare, e cuprins tot în A.

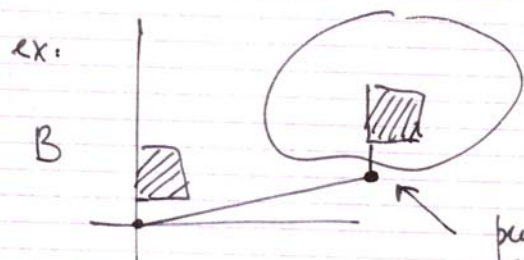
$$A \ominus B = \{x \mid B_x \subset A\}$$

$$A \oplus B = \{x \mid B_x \cap A \neq \emptyset\}$$

dacă $B_x \subset A \Rightarrow$ un x care se găsește în erodata lui A se găsește și în dilatarea lui A.

$$\Rightarrow A \ominus B \subset A \oplus B$$

dar $A \ominus B \subset A$? Nu neapărat.



(zero nu este conținut)

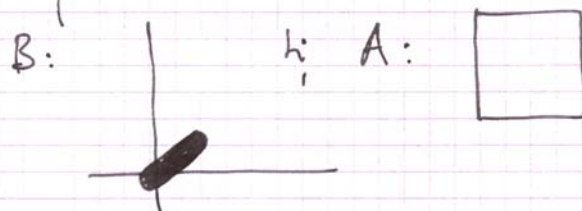
punctul • este conținut în erodata lui A

$$\Rightarrow A \ominus B \not\subset A$$

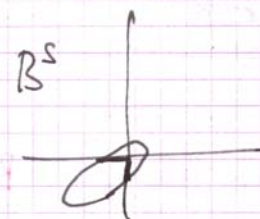
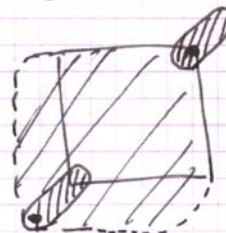
$$A \not\subset A \oplus B$$

Obs: dacă $0 \in B \Rightarrow A \ominus B \subset A$
și $A \subset A \oplus B$.

- caz particular:



$$\Rightarrow A \oplus B = A + B^s$$



- B^s - simetric

- dacă luăm un elem. structural simetric față de origine $\Rightarrow B = B^s$

$$\Rightarrow A \oplus B = A + B^s = A + B$$

$$A \ominus B = A - B^s$$



- Obs: dilatarea și erodarea = operații duale.

$$C(A \oplus B) = C A \ominus B$$

- mai ușor de exprimat cu definițiile Serra (Matheron).

- definirea unei eratori $\Rightarrow$ un mecanism, care poate fi aplicat de mai multe ori.

$$(A \ominus B) \ominus B$$

$$(A \ominus B) \oplus B \neq A$$

$$(A \oplus B) \ominus B \neq A$$

$\ominus$ și $\oplus$ nu sînt
operatori inverse

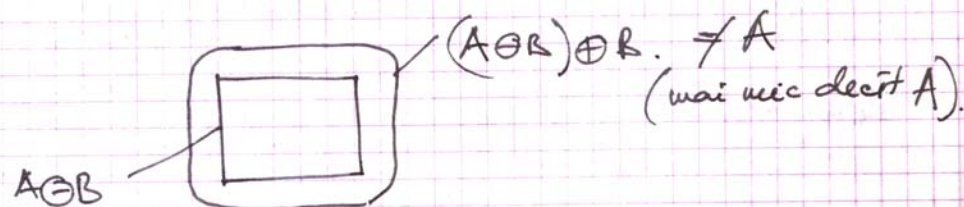
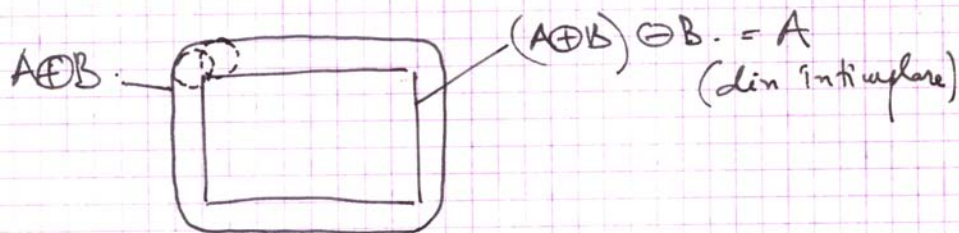
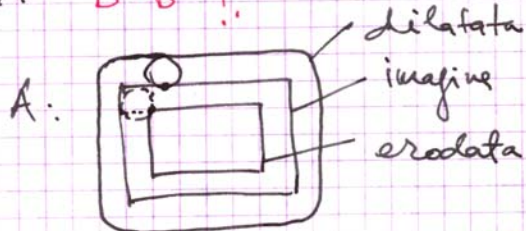
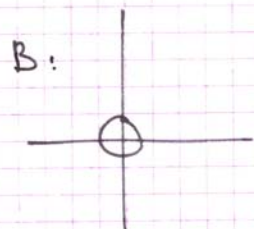
$$(A \oplus B) \ominus B^{\text{not}} = A \bullet B \quad - \text{încădarea lui } A \text{ prin } B$$

$$(A \ominus B) \oplus B^{\text{not}} = A \oslash B \quad - \text{deschiderea lui } A \text{ prin } B.$$

$$A \oslash B \subset A \subset A \bullet B \quad (C \text{ poate fi } \subseteq) \text{ dacă originea face parte din } B.$$

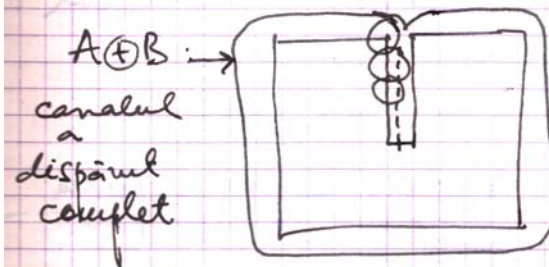
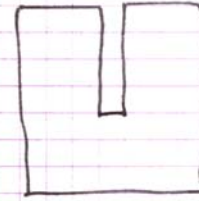
- un caz simplu:

$$B = B^S !!$$

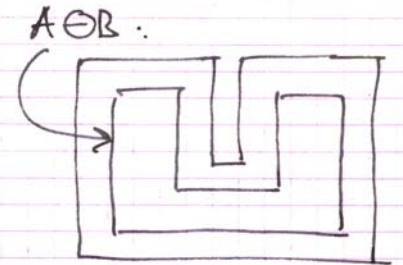
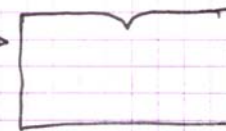


- alt caz:

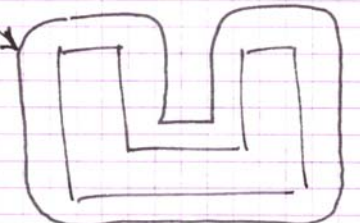
∇ - mai îngustă decît $2 \cdot r =$
diametrul elementului
structural B.



$$(A \oslash B) \ominus B$$



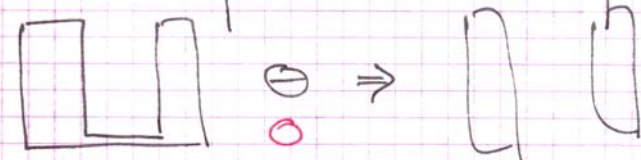
$$(A \oslash B) \oplus B$$

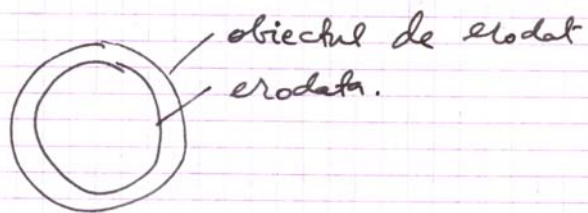


- și la închidere și la deschidere - colțurile se netezesc.

- la închidere - canalele strîmte se unesc și dispar obiecte care erau separate (apare un obiect)

- la deschidere - istmurile strîmte dispar și apar 2 inele $\rightarrow$ apar 2 obiecte.





- nu are nici un punct a.i. elementul structural să fie inclus în el.
 $\Rightarrow$ la erodare el va dispărea.

$\rightarrow$ deschiderea este o operație cu care putem elimina punctele izolate.
 închiderea - putem "închide" canale.

$A \circ B \subset A$ - monotonie
 $A \bullet B \supset A$ - monotonie.

obs.: o funcție monotonă și idempotentă s.n. filtrare.

$$(A \circ B) \circ B = A \circ B$$

definiții echivalente:

$$A \ominus B = \{x \mid B_x \subset A\} = \{x \mid \forall b \in B \exists a \in A \text{ a.i. } x = a - b\}$$

$$= \bigcap_{b \in B} A_{-b}$$

$$A \oplus B = \{x \mid B_x \cap A \neq \emptyset\} =$$

$$= \{x \mid \exists b \in B, \exists a \in A : x = a + b\}$$

$$= \bigcup_{b \in B} A_{+b} \quad -b = \text{simetricul lui } b$$

Proprietăți:

- dualitatea:

$$C(A \oplus B) = C A \ominus B$$

$$C(A \ominus B) = C A \oplus B$$

- translație cu elementul structural (nu $\emptyset$).

$$A \oplus B_t = (A \oplus B)_t$$

$$B_t = B \text{ translatat cu } t.$$

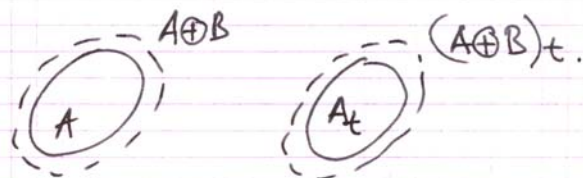
$$A \ominus B_t = (A \ominus B)_{-t}$$

- translația mulțimii A:

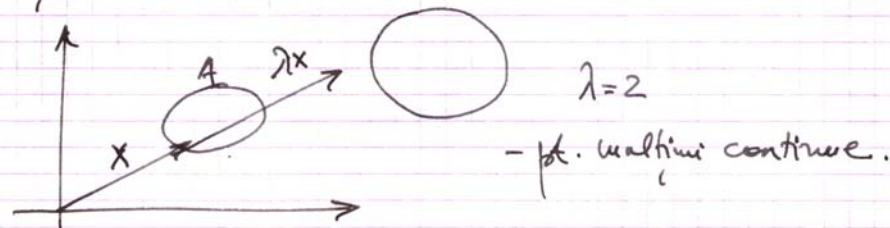
$$A_t = A \text{ translatat cu } t.$$

$$A_t \oplus B = (A \oplus B)_t$$

$$A_t \ominus B = (A \ominus B)_t$$



- mulțimea $\lambda \cdot A$, unde $\lambda = \text{real}$.



$$\frac{1}{\lambda}(A \oplus B) = A \oplus \frac{1}{\lambda}B$$

$$\frac{1}{\lambda}(A \ominus B) = A \ominus \frac{1}{\lambda}B$$

- monotonicitate:

$$A_1 \subset A_2 \Rightarrow A_1 \oplus B \subset A_2 \oplus B$$

$$A_1 \ominus B \subset A_2 \ominus B.$$

$$\text{dacă } B_1 \subset B_2 \Rightarrow A \oplus B_1 \subset A \oplus B_2$$

$$A \ominus B_1 \supset A \ominus B_2 \quad !!$$

- dacă $0 \in B \Rightarrow A \subset A \oplus B$, altfel nu.
 $A \ominus B \subset A$.

- pseudo-comutativitate și asociativitate:

$$A \oplus B = (B \oplus A)^s$$

$$A \oplus (B \oplus C) = (A \oplus B) \oplus C^s$$

$$(A \ominus B) \ominus C = A \ominus (B \oplus C).$$

$$(A \cup B) \oplus C = (A \oplus C) \cup (B \oplus C)$$

$$A \oplus (B \cup C) = (A \oplus B) \cup (A \oplus C)$$

Obs - dacă elem. structurant e mai complex, îl putem descompune în mai multe elemente.
 și aplicăm proprietățile de mai sus.

$$(A \cap B) \ominus C = (A \ominus C) \cap (B \ominus C).$$

$$A \ominus (B \cup C) = (A \ominus B) \cap (A \ominus C).$$

$$A \oplus (B \cap C) \subset (A \oplus B) \cap (A \oplus C)$$

$$A \ominus (B \cap C) \supset (A \ominus B) \cup (A \ominus C).$$

$$(B \cap C) \oplus A \subset (B \oplus A) \cap (C \oplus A)$$

$$(B \cap C) \ominus A \supset (B \ominus A) \cap (C \ominus A).$$

CORECTURĂ:

$$\begin{cases} A \circ B = (A \ominus B) \oplus B^s \\ A \bullet B = (A \oplus B) \ominus B^s \end{cases}$$

$$\begin{cases} A \circ B = (A \ominus B) \oplus B^s \\ A \bullet B = (A \oplus B) \ominus B^s \end{cases}$$

$$C(A \circ B) = C_A \bullet B$$

$$C(A \bullet B) = C_A \circ B.$$

$$A \circ B \subset A \subset A \bullet B - \text{valabil pt } \forall.$$

idempotență:

$$\begin{cases} (A \circ B) \circ B = A \circ B \\ (A \bullet B) \bullet B = A \bullet B. \end{cases}$$

- translate:

$$A_t \circ B = (A \circ B)_t$$

$\circ$ = deschis

$$A_t \bullet B = (A \bullet B)_t.$$

$\bullet$ = închis

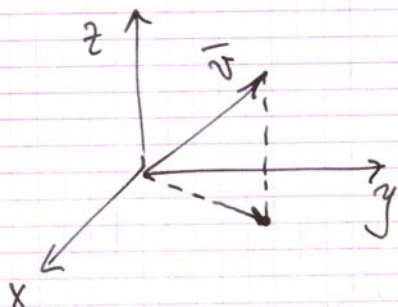
$$\frac{1}{\lambda}(A \circ B) = A \circ \frac{1}{\lambda}B.$$

$$\frac{1}{\lambda}(\lambda A \cdot B) = A \cdot \frac{1}{\lambda} B$$

$$\begin{cases} A_1 \subset A_2 \Rightarrow A_1 \cap B \subset A_2 \cap B & A_1 \cdot B \subset A_2 \cdot B \\ B_1 \subset B_2 \Rightarrow A \cap B_1 \supset A \cap B_2 & A \cdot B_1 \subset A \cdot B_2 \end{cases}$$

$$\begin{cases} A \cap B = \bigcup_{B_x \subset A} B_x \\ A \cdot B = \bigcap_{B_x \cap A \neq \emptyset} B_x \end{cases}$$

ex.: de operații idempotente: proiecția



Filtre alternate Secvențiale

$$(((A \cap B) \cdot B) \cap 2B) \cdot 2B) \cap 3B \dots$$

Obs. Wavelette - au un grad de libertate în plus față de funcțiile clasice "sin" și "cos".

Support compact = intervalul pe care funcția $\neq 0$.

- wavelet = 2 grade de libertate: translația și scalarea.
 { are o bază numerabilă pt. toate funcțiile L^2 din $\mathbb{R}$.

Operații morfologice de cântărire

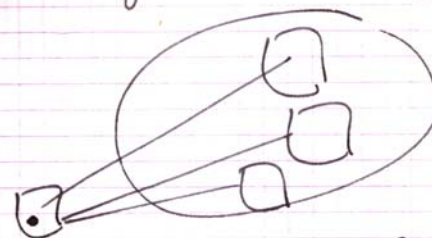
- un cântărire interior $\delta A = A - (A \ominus B)$

B = poate fi un cerculeț.

- cântărirul exterior $\Delta A = (A \oplus B) - A$.

- gradientul morfologic:

$$\text{grad } A = (A \oplus B) - (A \ominus B).$$



→ procesul se poate paraleliza

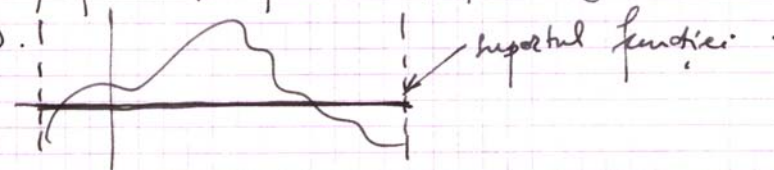
Obs.: 3 mașini cu atâtea procesoare câte pixeli are imaginea.

↑ Morfologie mat. pt. semnale binare

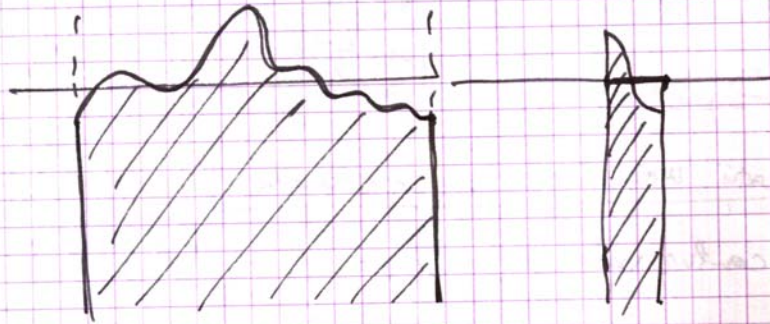
↓ morf. mat. pt. imagini cu nivele de gri.

Obs.: morf. poate fi aplicată și pt. imagini 3-D.

pt. 1-D.



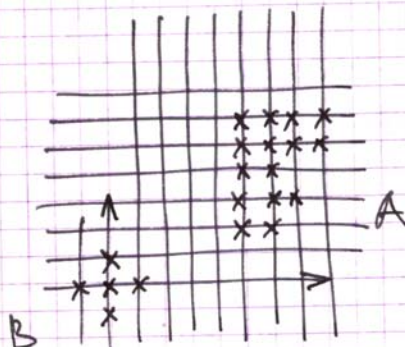
- funcția 1-D se pune în corelație cu imaginea 2-D: (biunivoc)



și alegem elementul structurant

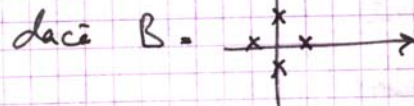
caz particular: B:  - identică cu D.

Ex. morfologie discretă.

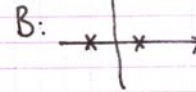
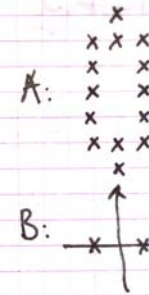
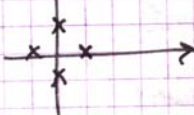


un "ceruleț" care conține originea

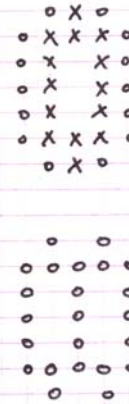
$A \oplus B \Rightarrow$



dacă B =



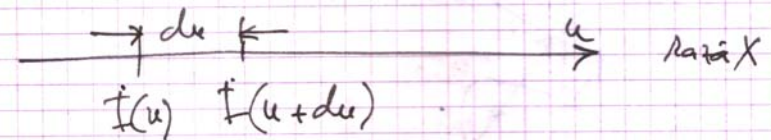
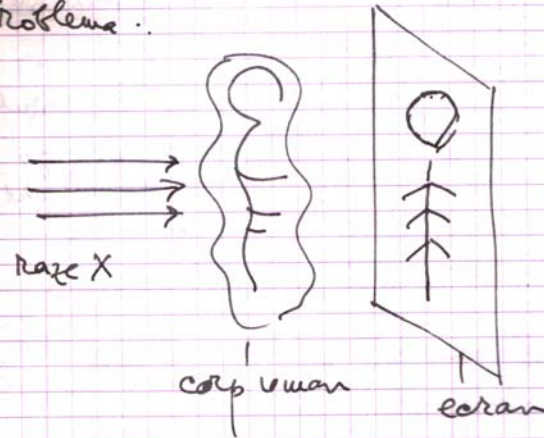
$A \oplus B$



punctul acesta nu va mai fi

REFACEREA IMAGINILOR DIN PROIECTII

Problema:



$$dI = I(u+du) - I(u) = -k \cdot I \cdot du$$

tomografie de absorbție
tomografie de absorbție

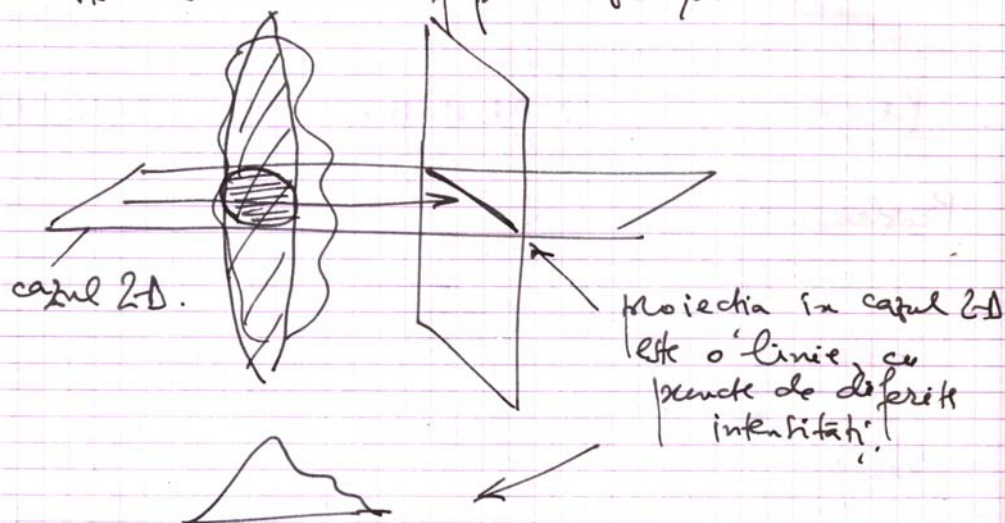
$k = k(u)$
coeficient de material

$$\frac{dI}{I} = -k \cdot du_{u_1} \quad \text{— dacă } k = \text{constant}$$

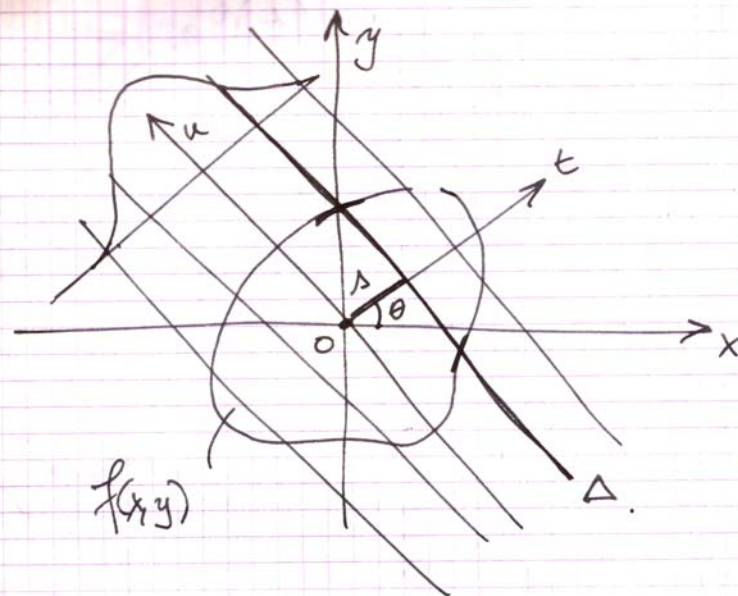
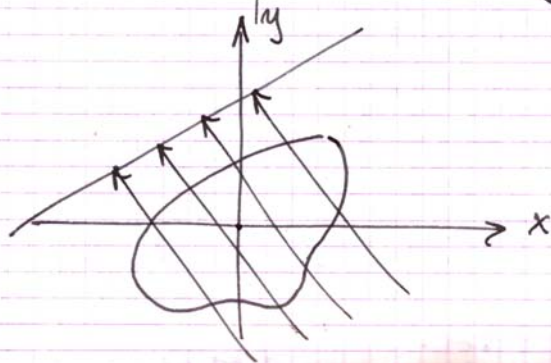
$$I = I_0 \cdot e^{-\int_{u_0}^u k \cdot du} \quad \text{dacă nu } \Rightarrow$$

$$a = \ln \frac{I_0}{I} = \int_{u_0}^{u_1} k(u) du.$$

- pe ecran se vede ceva proporțional cu atenuarea a
- vrem să vedem de fapt • secțiune



și vrem să aflăm valoarea (toate val.)
punctelor din



Transformata Radon.

Tr. Radon a lui $f(x, y)$: $Rf = g = g(\Delta, \theta)$

pt. dreapta Δ : ec. dreptei:

$$x \cdot \cos \theta + y \cdot \sin \theta - \Delta = 0.$$

$$g(\Delta, \theta) = \iint f(x, y) \cdot \delta(x \cdot \cos \theta + y \cdot \sin \theta - \Delta) dx dy.$$

Schimbăm coordonatele din Δ, θ în t, u :

$$\begin{cases} t = x \cdot \cos \theta + y \cdot \sin \theta \\ u = -x \cdot \sin \theta + y \cdot \cos \theta \end{cases}$$

$$\begin{pmatrix} t \\ u \end{pmatrix} = \begin{pmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \Rightarrow$$

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} t \\ u \end{pmatrix}$$

$$\Rightarrow g = \iint_{\mathbb{R}^2} f(t-u) \cdot f(t \cos\theta - u \sin\theta, t \sin\theta + u \cos\theta) dt du$$

integrate în rap. cu "t" mai întâi

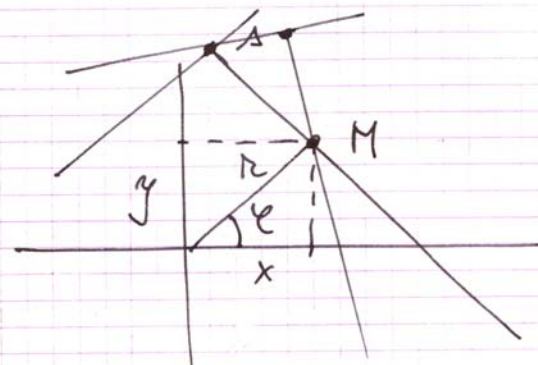
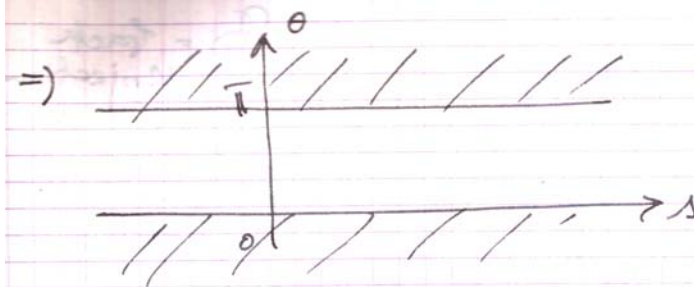
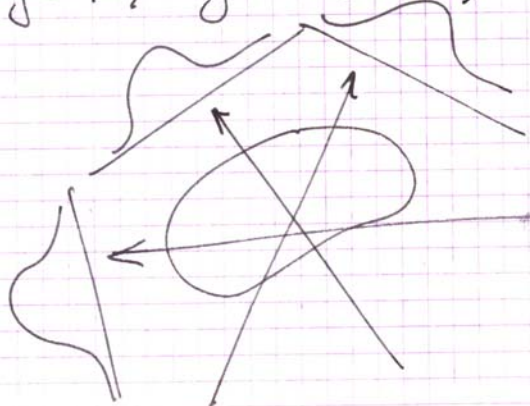
$$= \int_{\mathbb{R}} f(s \cos\theta - u \sin\theta, s \sin\theta + u \cos\theta) du$$

f - îi corespunde lui k din formula intensității laterale (I).

s - variază de la $-\infty$ la $+\infty$
 $\theta = 0, \bar{\alpha}$

$$g(s, \theta) = g(-s, \theta + \bar{\alpha})$$

$$g(s, \theta) = g(s, \theta + 2k\bar{\alpha})$$



- care este leg.
 între s, θ
 și x, y -)



$$x = r \cdot \cos\varphi$$

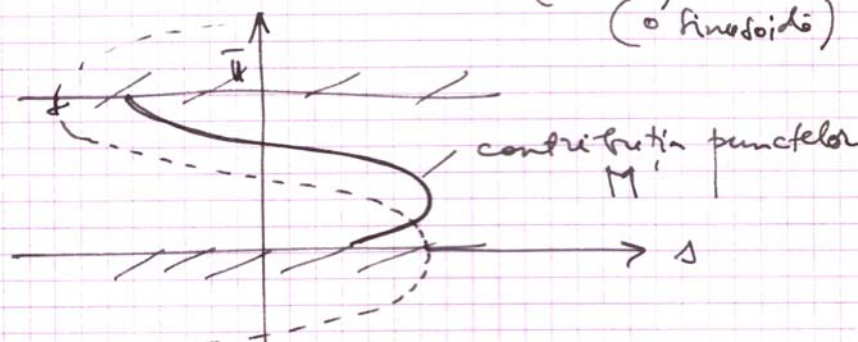
$$y = r \cdot \sin\varphi$$

$$x \cdot \cos\theta + y \cdot \sin\theta - s = 0$$

$$r \cdot \cos\varphi \cdot \cos\theta + r \cdot \sin\varphi \cdot \sin\theta - s = 0$$

$$r \cdot \cos(\theta - \varphi) = s$$

(o sinusoidă)



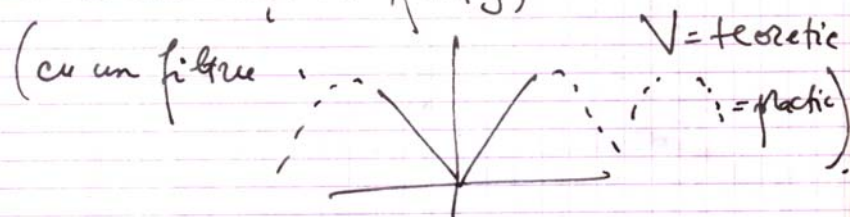
$$\hat{f}(x, y) = \int_0^{\bar{\alpha}} g(x \cdot \cos\theta + y \cdot \sin\theta, \theta) d\theta$$

$$r \cdot \cos\varphi \cdot \cos\theta + r \cdot \sin\varphi \cdot \sin\theta$$

iar $\hat{f}(x,y) = Bg$ unde B = back
projection
operator

$\hat{f}(x,y)$ - este aproape ce ne trebuie

si este necesara o filtrare TS pt.
a se obtine $\approx f(x,y)$



+ teorema proiectiei
+ segmentare.