

# Concepte și aplicații în Vederea Artificială

Bogdan Alexe

[bogdan.alexe@fmi.unibuc.ro](mailto:bogdan.alexe@fmi.unibuc.ro)

Cursul 6

anul III, Opțional Informatică, semestrul I, 2018-2019

# Predare proiect 2 pe 11 noiembrie

## 1.6 Predarea proiectului

Puneți într-o arhivă cu numele *tema2.cod.zip* codul vostru Matlab. Puneți într-un document cu numele *tema2\_rezultate.pdf* următoarele:

- (a) (1 punct) rezultatele obținute la punctul (1.1) pentru imaginea 'castel.jpg' ;
- (b) (1 punct) rezultatele obținute la punctul (1.2) pentru imaginea 'praga.jpg' ;
- (c) (1 punct) rezultatele obținute la punctul (1.3) pentru imaginea 'delfin.jpg' ;
- (d) (1.5 puncte) rezultatele obținute la punctul (1.4) pentru imaginea 'arcTriumf.jpg' ;
- (e) (1.5 puncte) rezultatele obținute la punctul (1.5) pentru imaginea 'lac.jpg' .
- (f) (2 puncte) rezultatele obținute de voi pentru alte imagini (cel puțin 5) decât cele furnizate folosind cele 3 metode de selectare a drumurilor: metoda de selecție aleatoare, metoda greedy și metoda programării dinamice. Puteți exemplifica rezultatele obținute pentru oricare din operațiile enumerate mai sus (în secțiunile 1.1 - 1.5). Includeți cel puțin 3 exemple reușite și cel puțin 2 exemple nereușite explicând de fiecare dată de ce algoritmul vostru a reușit sau nu a reușit.

# Predare proiect 2 pe 11 noiembrie

Se va nota cu 1 punct prezentarea proiectului în format pdf. Vom lua în calcul aspecte precum: așezarea în pagină, comentariile ce însoresc imaginile prezentate, exemplele alese.

Oficiu: 1 punct.

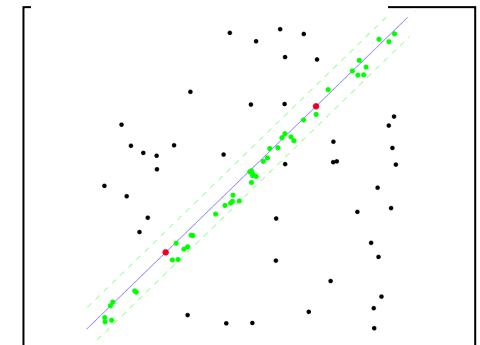
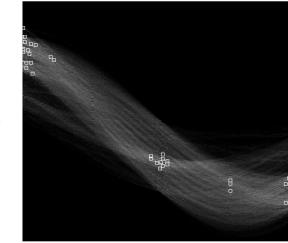
Pentru fiecare exemplu de la punctele (a) - (e) includeți 3 imagini: imaginea inițială, imaginea obținută cu algoritmul implementat și imaginea obținută cu redimensionarea uzuală - folosind funcția imresize (acolo unde e posibil). La punctul (f) includeți pentru fiecare exemplu 5 imagini: imaginea inițială, imaginea obținută cu algoritmul implementat pentru toate cele 3 metode (aleatoare, greedy, programare dinamică) și imaginea obținută cu redimensionarea uzuală - folosind funcția imresize.

Trimiteti cele două fișiere (*tema2\_cod.zip* și *tema2\_rezultate.pdf*) la adresa de email a Iulianei Georgescu, [georgescu.lily@yahoo.com](mailto:georgescu.lily@yahoo.com).

Termenul limită de predare a proiectului este duminică, 11 noiembrie 2018, ora 23:59. Fiecare zi de întârziere în predarea proiectului se penalizează cu 1 punct în minus.

# Cursul trecut

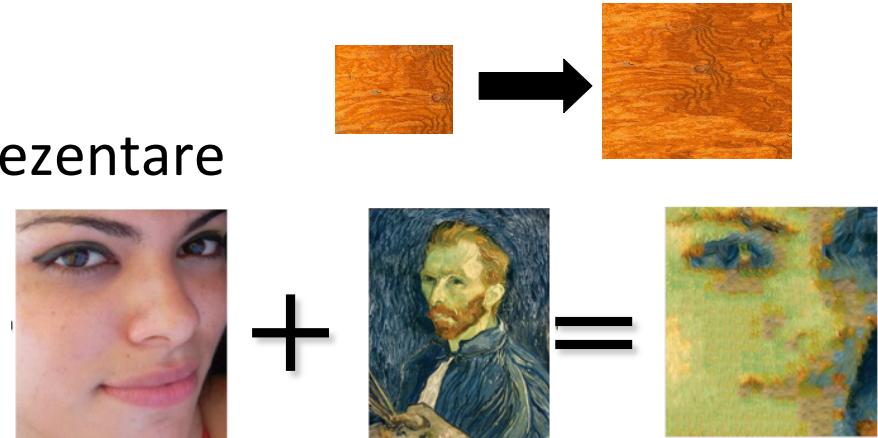
- Aplicație: detectarea liniilor cu transformata Hough
- Aplicație: detectarea liniilor cu RANSAC
- Compararea contururilor
- Textură – generalități



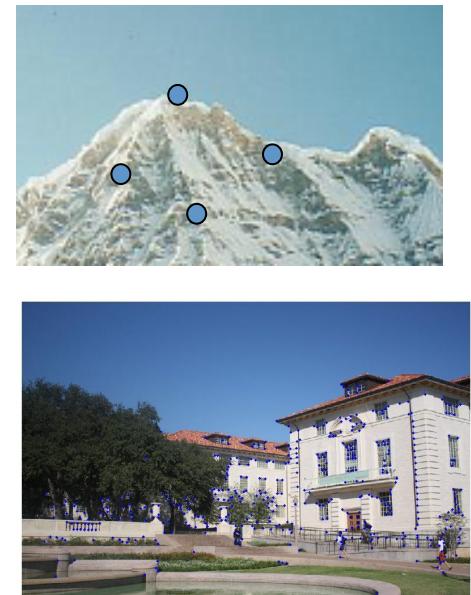
Σ Σ

# Cursul de azi

- Textură
  - Reprezentare
  - Colecții de filtre pentru reprezentare
  - Sinteză texturii (tema 3)
  - Transferul texturii (tema 3)



- Trăsături locale
- Detectorul Harris



# De ce studiem textura?

- **Segmentare**
  - grupăm (segmentăm) regiunile din imagine cu aceeași textură
- **Clasificare**
  - clasificăm obiecte ca aparținând unei clase de obiecte
- **Sinteză**
  - dată o mostră de textură (dimensiuni mici) vrem să generăm o textură similară (dimensiuni mari)



mostră de textură

textură generată

# Percepția vizuală a texturii

- Julesz – 1975, studiază sistemul vizual uman:
  - cum reprezintă textura
  - ce reguli urmează pentru procesarea unei imagini cu textură
- Diferențierea texturilor - experiment
  - sistemul vizual uman nu poate distinge între texturi cu anumite proprietăți statistice identice

Texturi la fel sau diferite?

A 7x7 grid of black L-shaped blocks on a white background. Each L-shape is composed of three squares: one vertical column of two squares and one horizontal square extending from the middle of the left side of the column. The blocks are arranged in a staggered pattern, where each row is offset by one square compared to the row above it. The top-left square of the grid is empty.

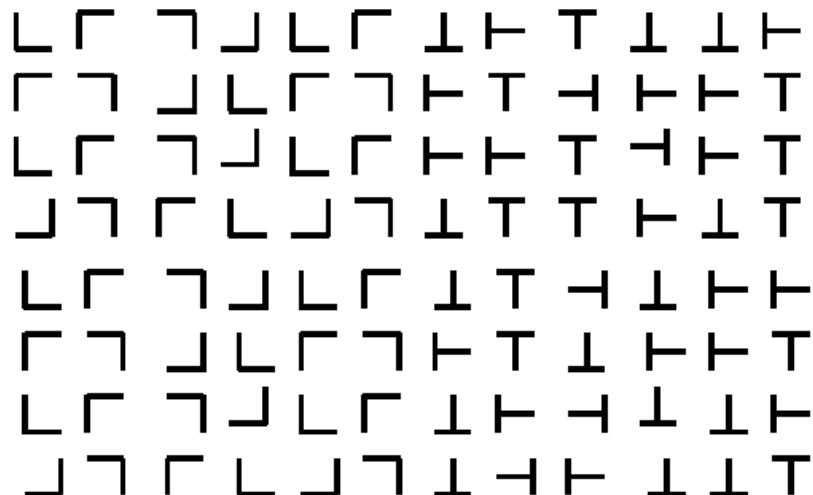
The image shows a large, dense grid of black L-shaped blocks on a white background. The blocks are arranged in a staggered pattern, creating a complex and intricate design. The grid consists of approximately 10 columns and 10 rows of these L-shaped blocks, resulting in a total of about 100 individual blocks. The blocks overlap and nest within each other, forming a continuous pattern that covers most of the frame.

A 10x10 grid of black L-shaped blocks on a white background. The blocks are arranged in a staggered pattern, where each block is positioned such that its vertical stem is aligned with the horizontal bar of the block directly above it. This creates a continuous, winding path across the entire grid.

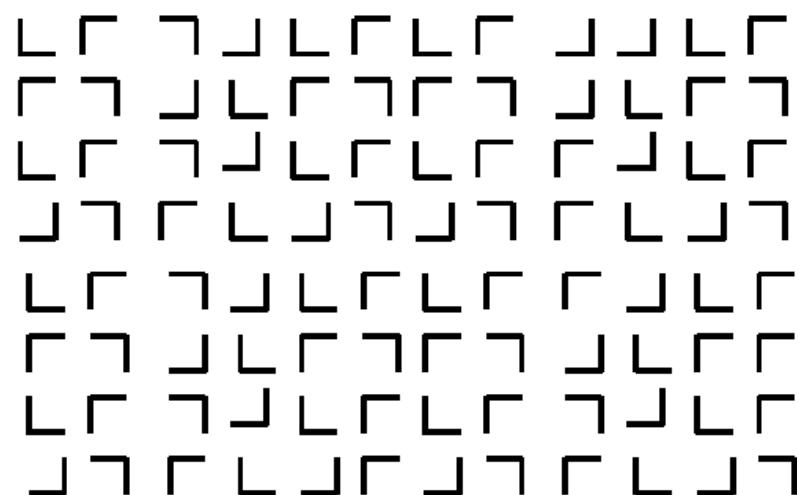
A 10x10 grid of black L-shaped blocks on a white background. The blocks are arranged in a staggered pattern, creating a tessellated effect. Each block is composed of three squares: one vertical column of two squares and one horizontal row of two squares, meeting at the top-left square of the column.

# Texteli

- texteli = **texture elements**
- elementele constitutive ale unei texturi
- analiza unei texturi: analiza relației statistice dintre texteli



texteli



texteli

# Reprezentarea texturii

Texturile sunt formate din pattern-uri locale ce se repetă, aşa încât:

- găseşte aceste pattern-uri (texteli)
  - foloseşte filtre care arată ca pattern-urile
  - reține valoarea răspunsului după filtrare
- descrie datele d.p.d.v. statistic pentru fiecare fereastră locală
  - medie, deviaţie standard
  - histograme

# Filtre pentru calculul derivatelor partiale – cursurile trecute

corelație

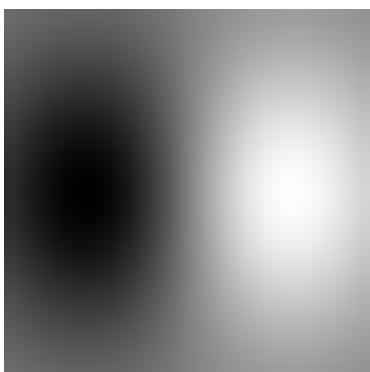
0	0	0
0	-1	1
0	0	0

$$\frac{\partial f(x, y)}{\partial x} \approx \frac{f(x+1, y) - f(x, y)}{1}$$

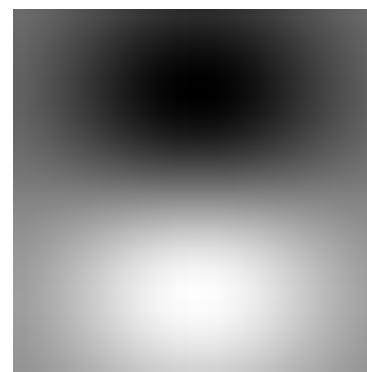
-1	0	1
-2	0	2
-1	0	1

Filtru Sobel vertical pentru calculul  
derivatei partiale în raport cu x

**Generalizare pentru filtre  
care calculează**  $\frac{\partial f(x, y)}{\partial x}$



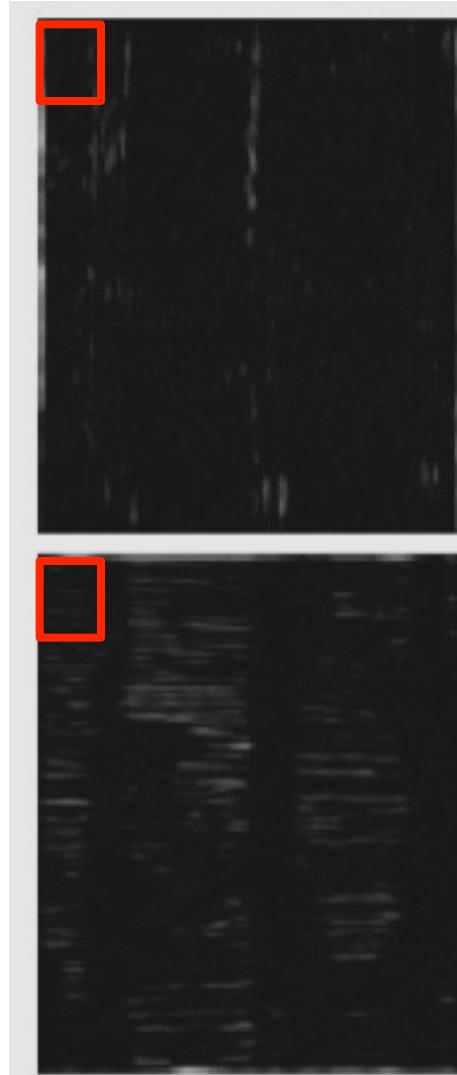
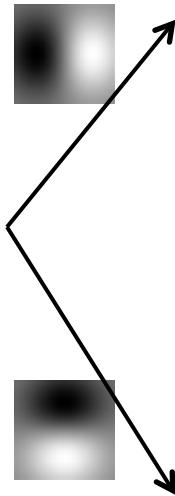
**Generalizare pentru filtre  
care calculează**  $\frac{\partial f(x, y)}{\partial y}$



# Reprezentarea texturii: exemplu



imagine inițială



imagini filtrate  
(filtre pentru derive)

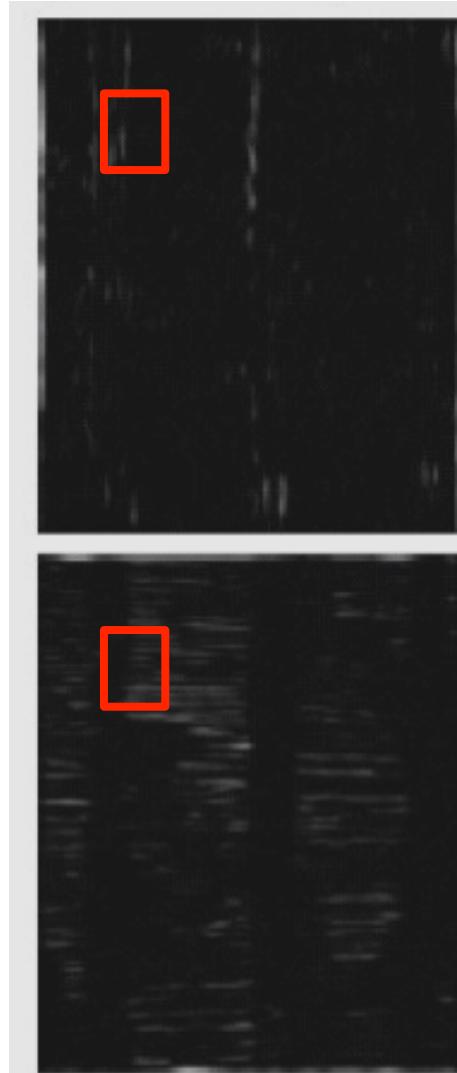
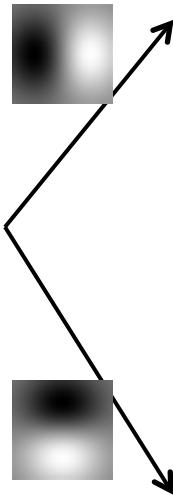
	<u>media</u> $d/dx$	<u>media</u> $d/dy$
fereastra #1	4	10
⋮	⋮	⋮

Statistică pentru a descrie pattern-urile în ferestre mici

# Reprezentarea texturii: exemplu



imagine inițială



imagini filtrate  
(filtre pentru derive)

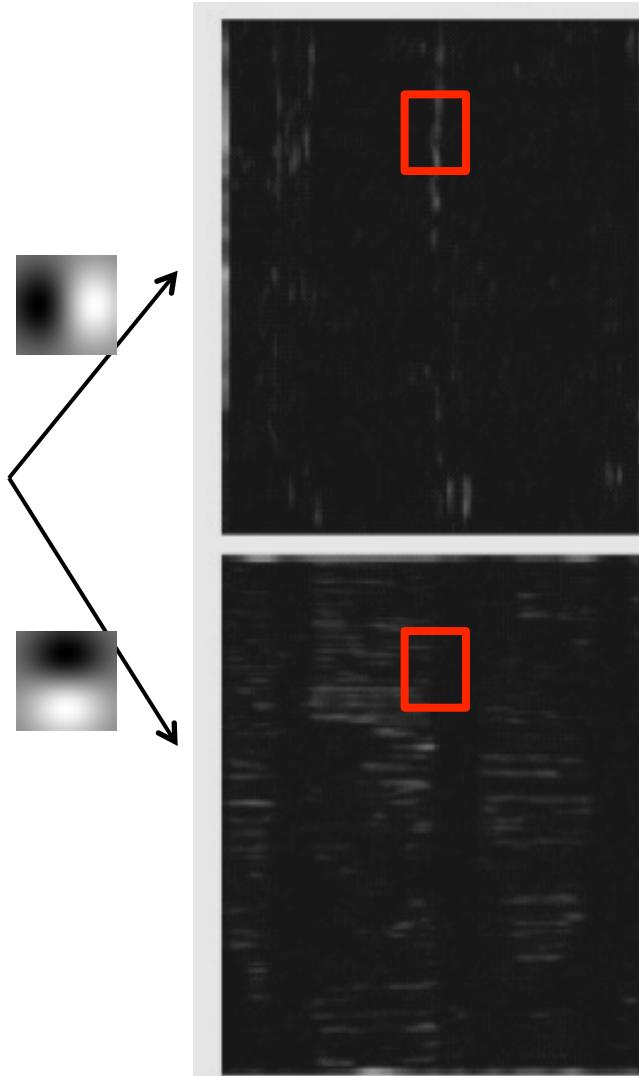
	<u>media</u> $d/dx$	<u>media</u> $d/dy$
fereastra #1	4	10
fereastra #2	25	7
⋮	⋮	⋮

Statistică pentru a descrie pattern-urile în ferestre mici

# Reprezentarea texturii: exemplu



# Imagine inițială

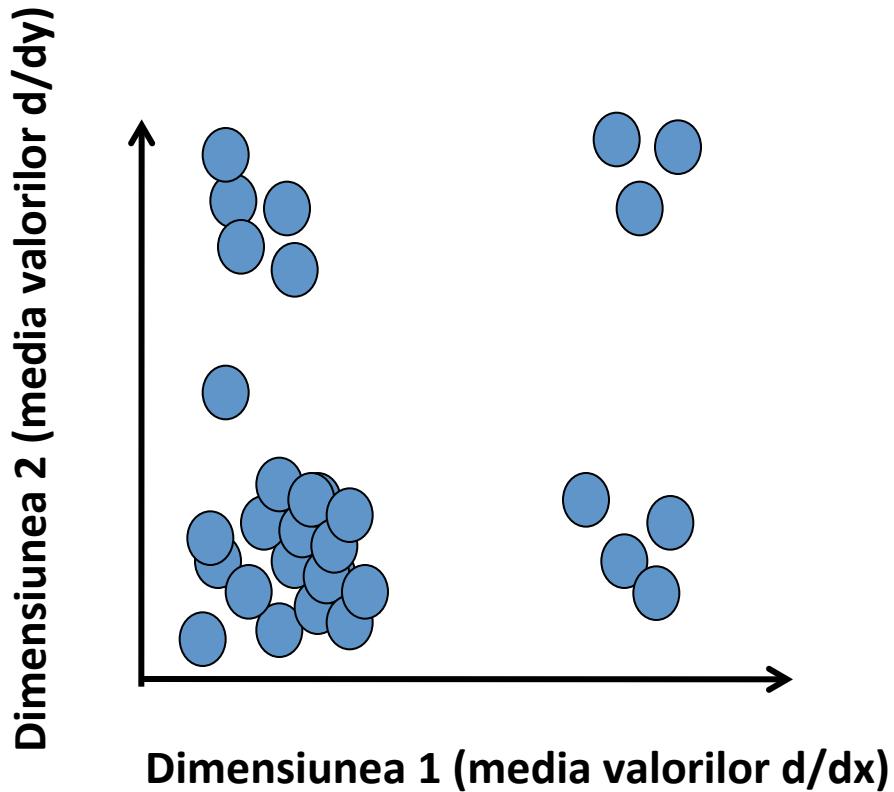


## **imagini filtrate (filtre pentru derivate)**

	<u>media</u> <u>d/dx</u>	<u>media</u> <u>d/dy</u>
fereastra #1	4	10
fereastra #2	25	7
fereastra #3	18	20
	•	•

## Statistici pentru a descrie pattern-urile în ferestre mici

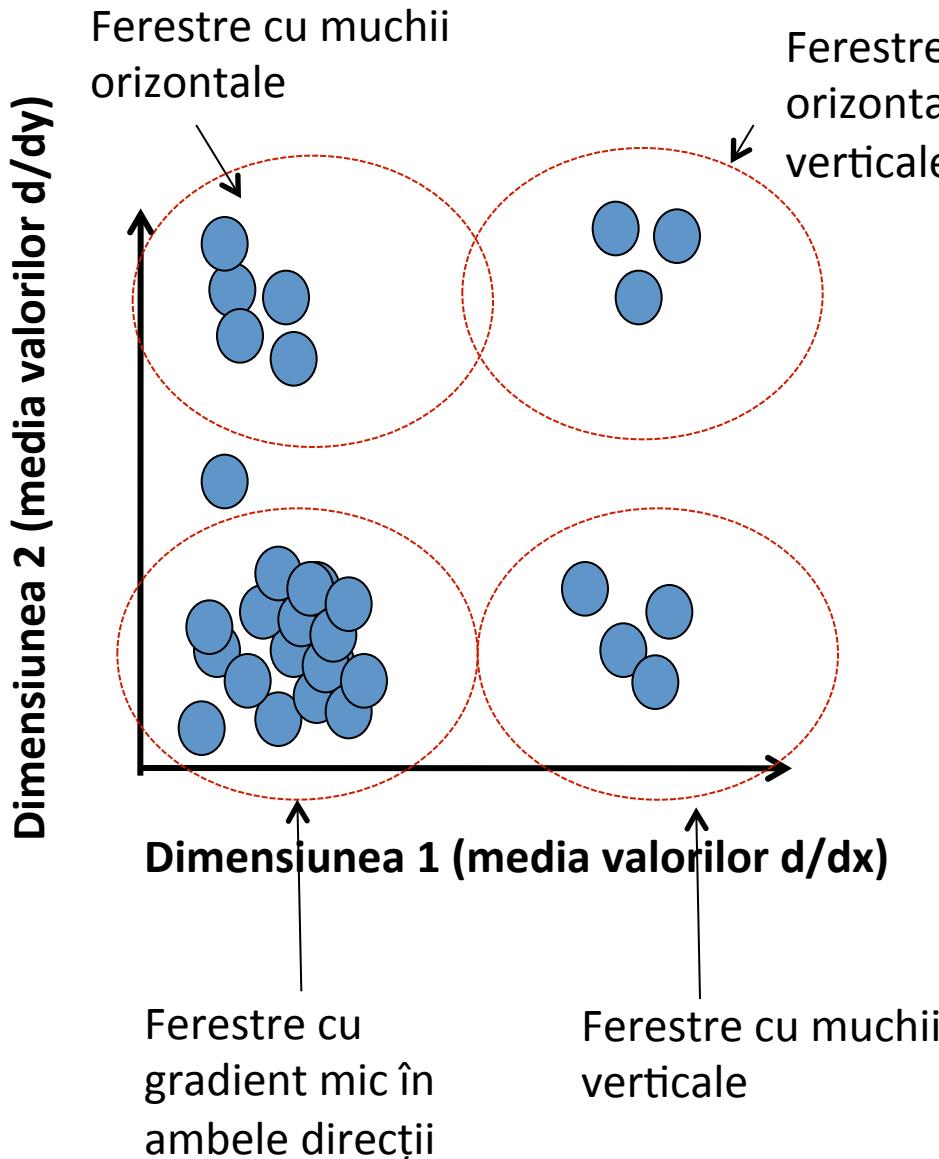
# Reprezentarea texturii: exemplu



	<u>media</u> <u>d/dx</u>	<u>media</u> <u>d/dy</u>
fereastra #1	4	10
fereastra #2	25	7
fereastra #3	18	20
	•	•

# Statistici pentru a descrie pattern-urile în ferestre mici

# Reprezentarea texturii: exemplu



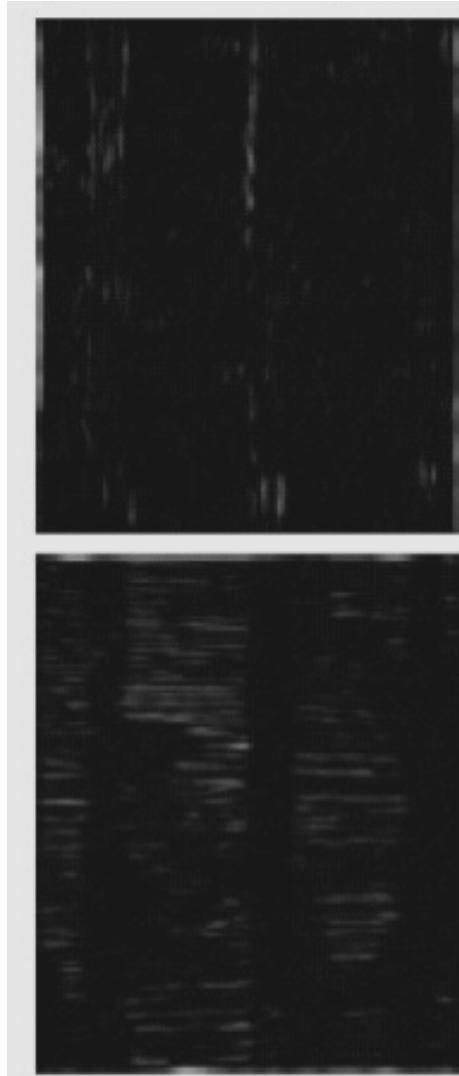
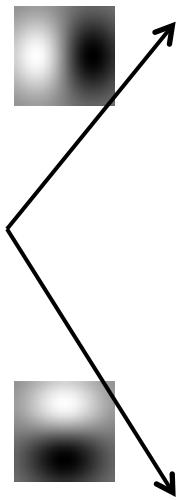
	<u>media</u> <u>d/dx</u>	<u>media</u> <u>d/dy</u>
fereastra #1	4	10
fereastra #2	25	7
fereastra #3	18	20
		⋮

Statistică pentru a descrie pattern-urile în ferestre mici

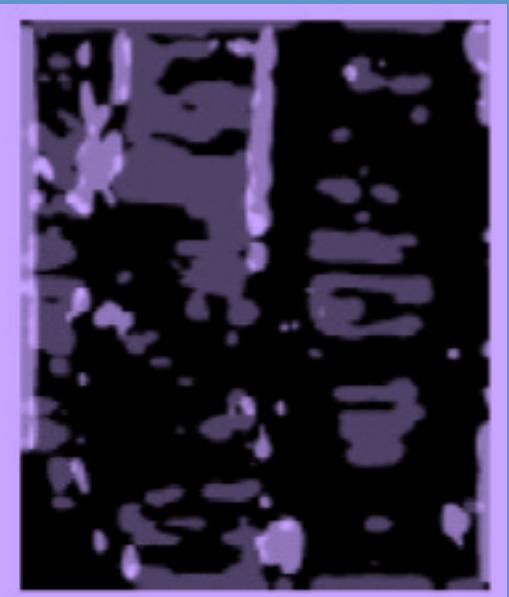
# Reprezentarea texturii: exemplu



Imagine inițială

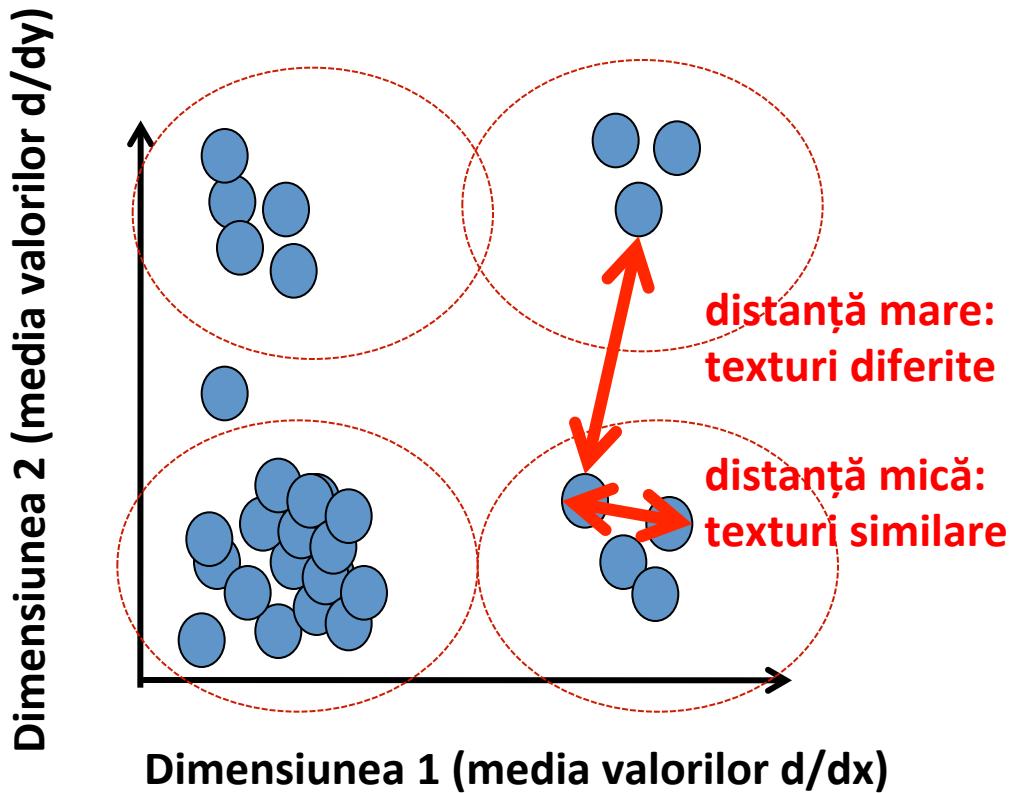


imagini filtrate  
(filtre pentru derive)



Vizualizare a tipurilor de textură.

# Reprezentarea texturii: exemplu



	media $d/dx$	media $d/dy$
fereastra #1	4	10
fereastra #2	25	7
fereastra #3	18	20

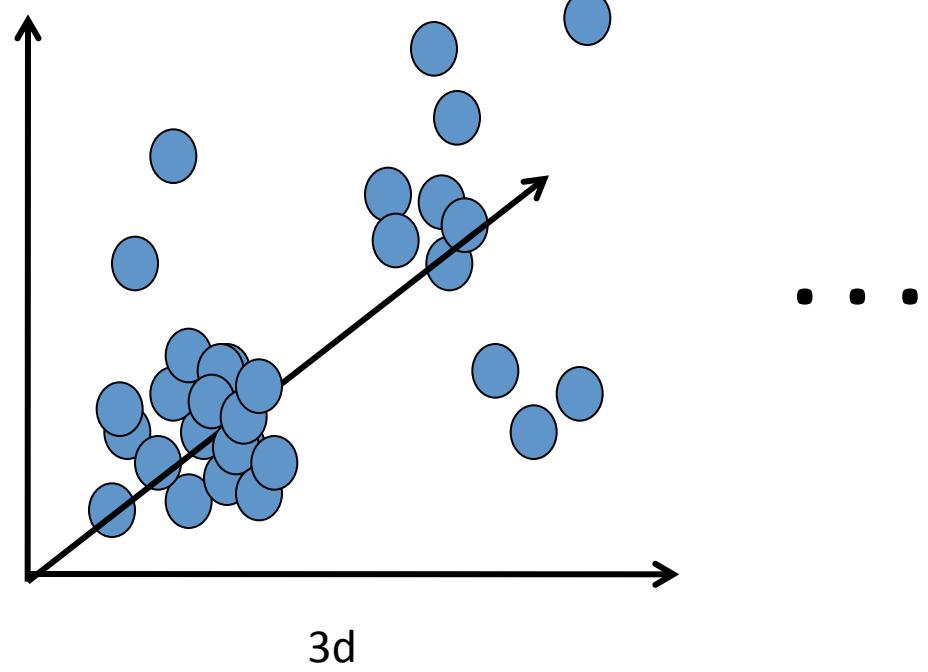
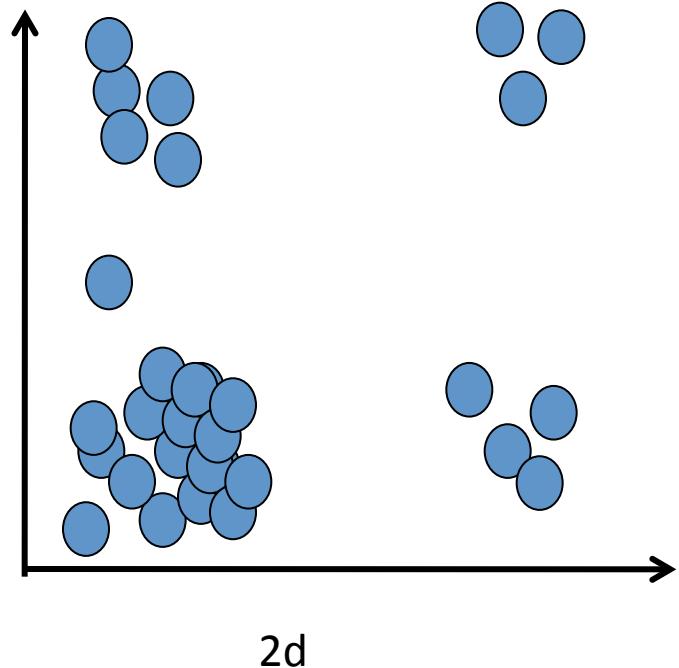
⋮

Statistici pentru a  
descrie pattern-urile în  
ferestre mici

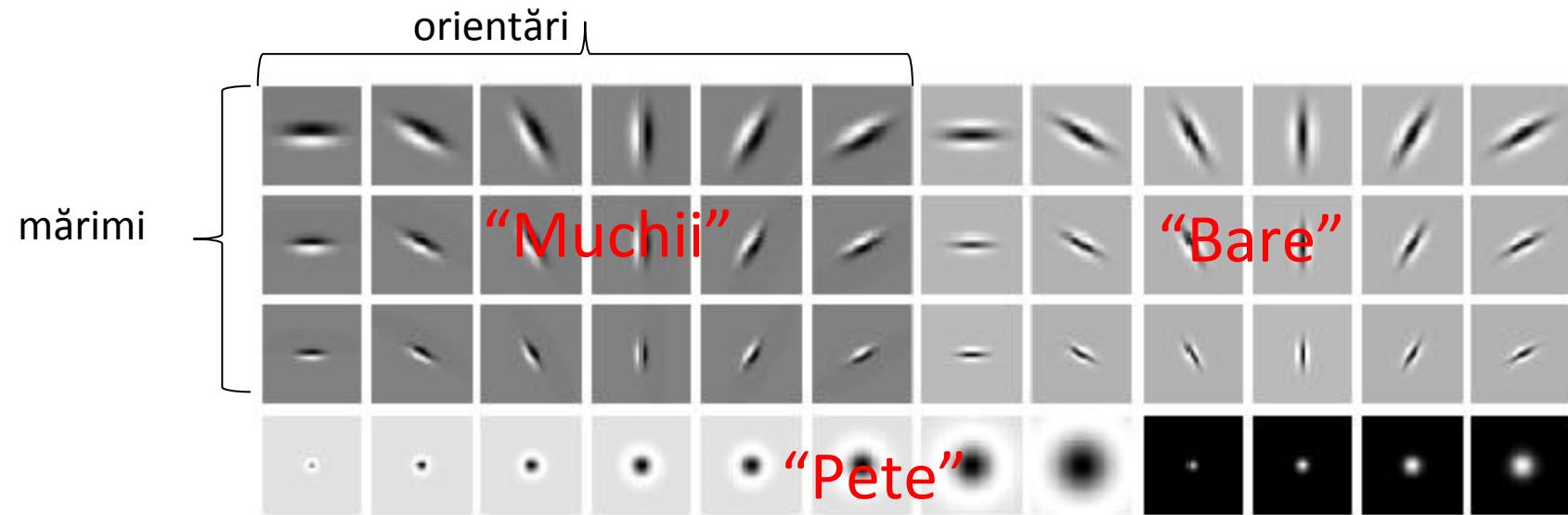
# Colecții de filtre

- În exemplu anterior am folosit două filtre, fiecare fereastră este descrisă d.p.d.v. al texturii printr-un vector cu 2 componente
  - derivatele parțiale  $x$  și  $y$  oferă informație despre structura locală.
- putem generaliza de la 2 dimensiuni la  $d$  dimensiuni: aplicăm o colecție de  $d$  filtre
- vectorul care descrie fereastra va avea  $d$ -componente.
  - texturi similare/disimilare în acest spațiu  $d$ -dimensional

# Spațiu de dimensiune d



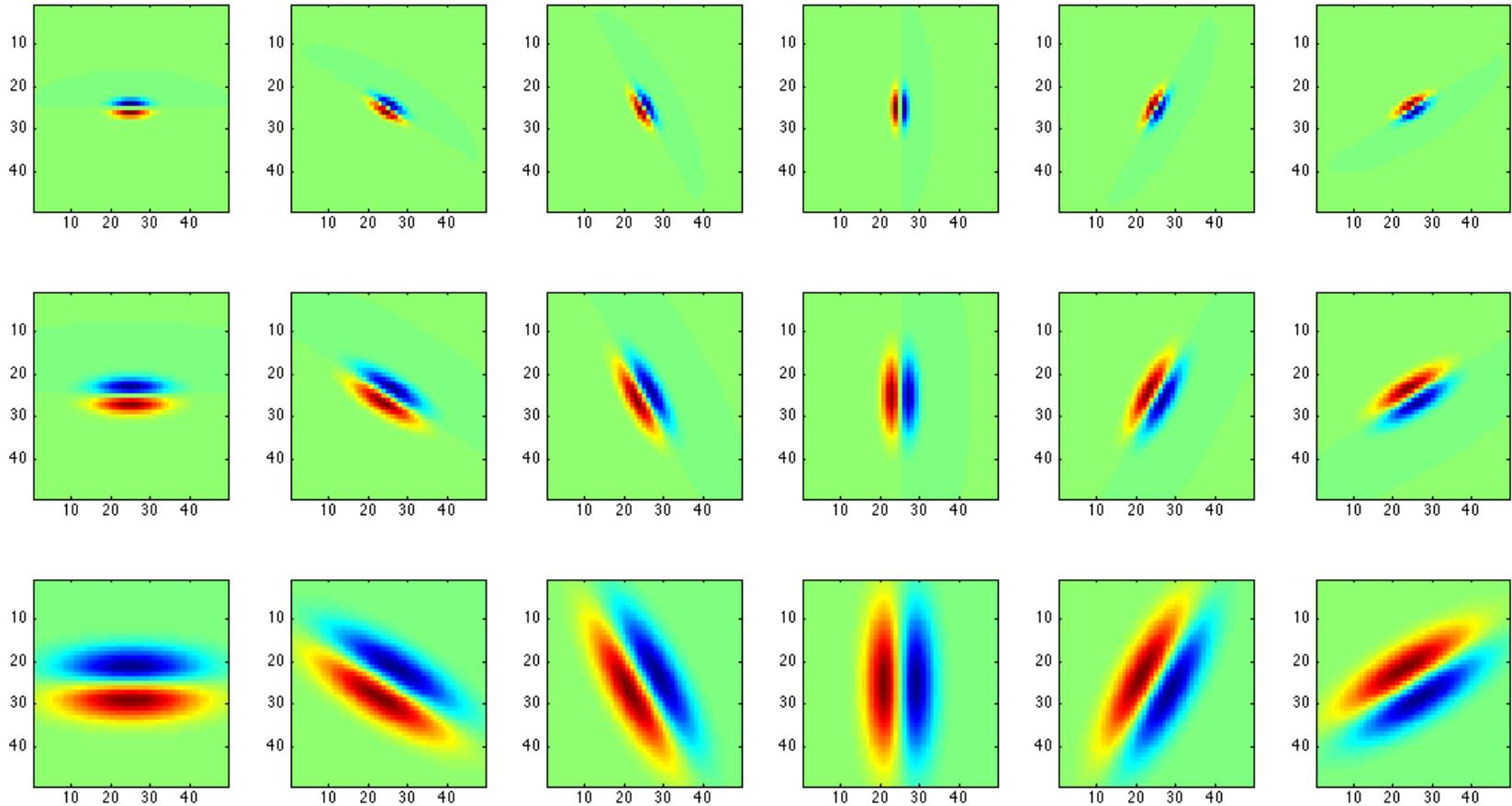
# Colecții de filtre



- Ce filtre să conțină colecția?
  - o combinație de mărimi și orientări, diverse tipuri de pattern-uri.

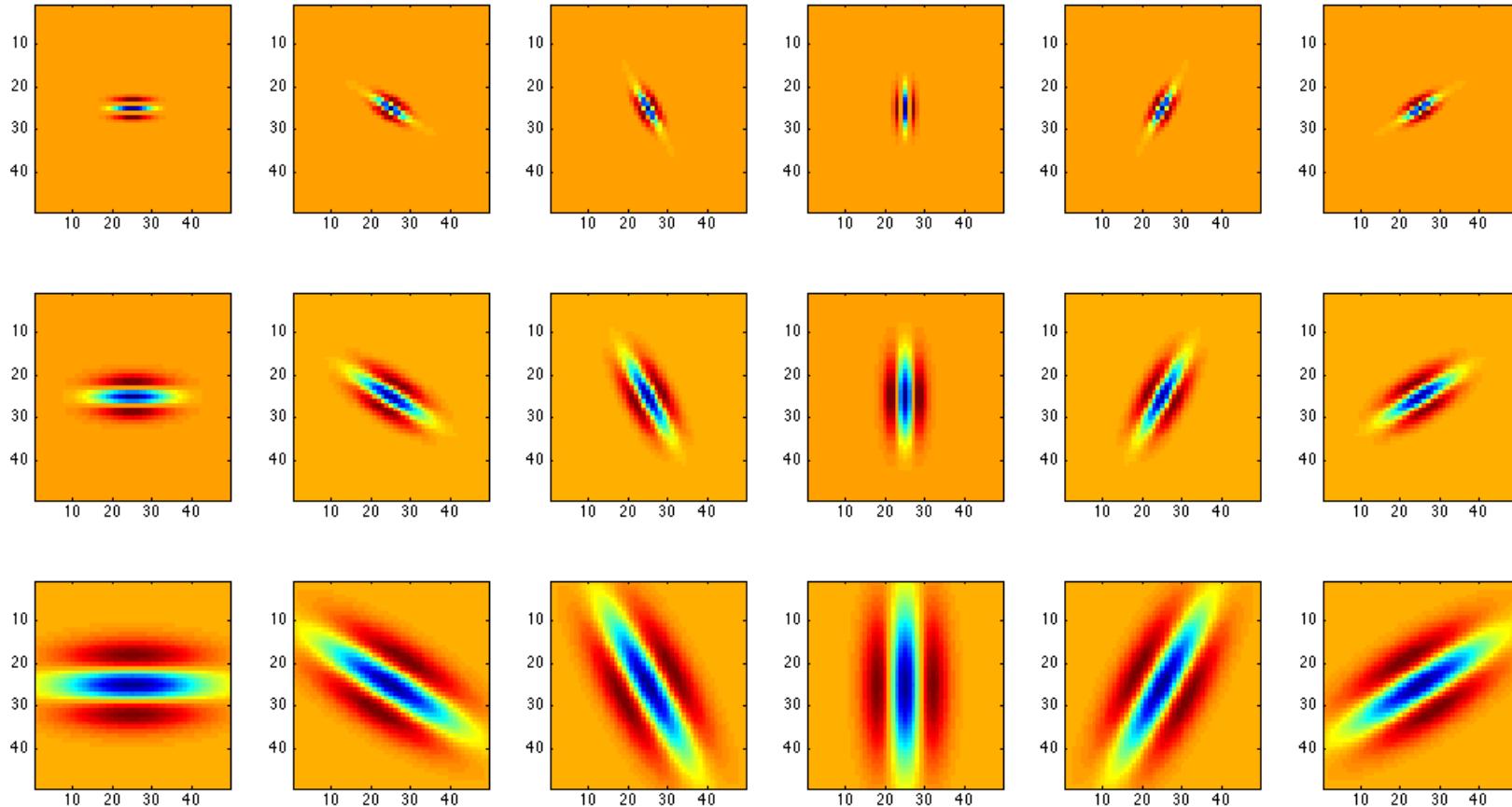
<http://www.robots.ox.ac.uk/~vgg/research/texclass/filters.html>

# Filtre pentru detectarea muchiilor



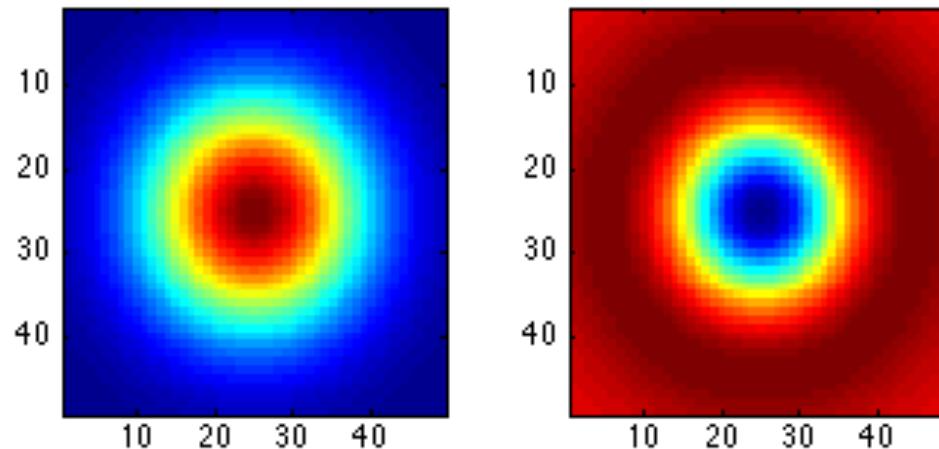
roșu – valori mari, albastru – valori mici

# Filtru pentru detectarea barelor



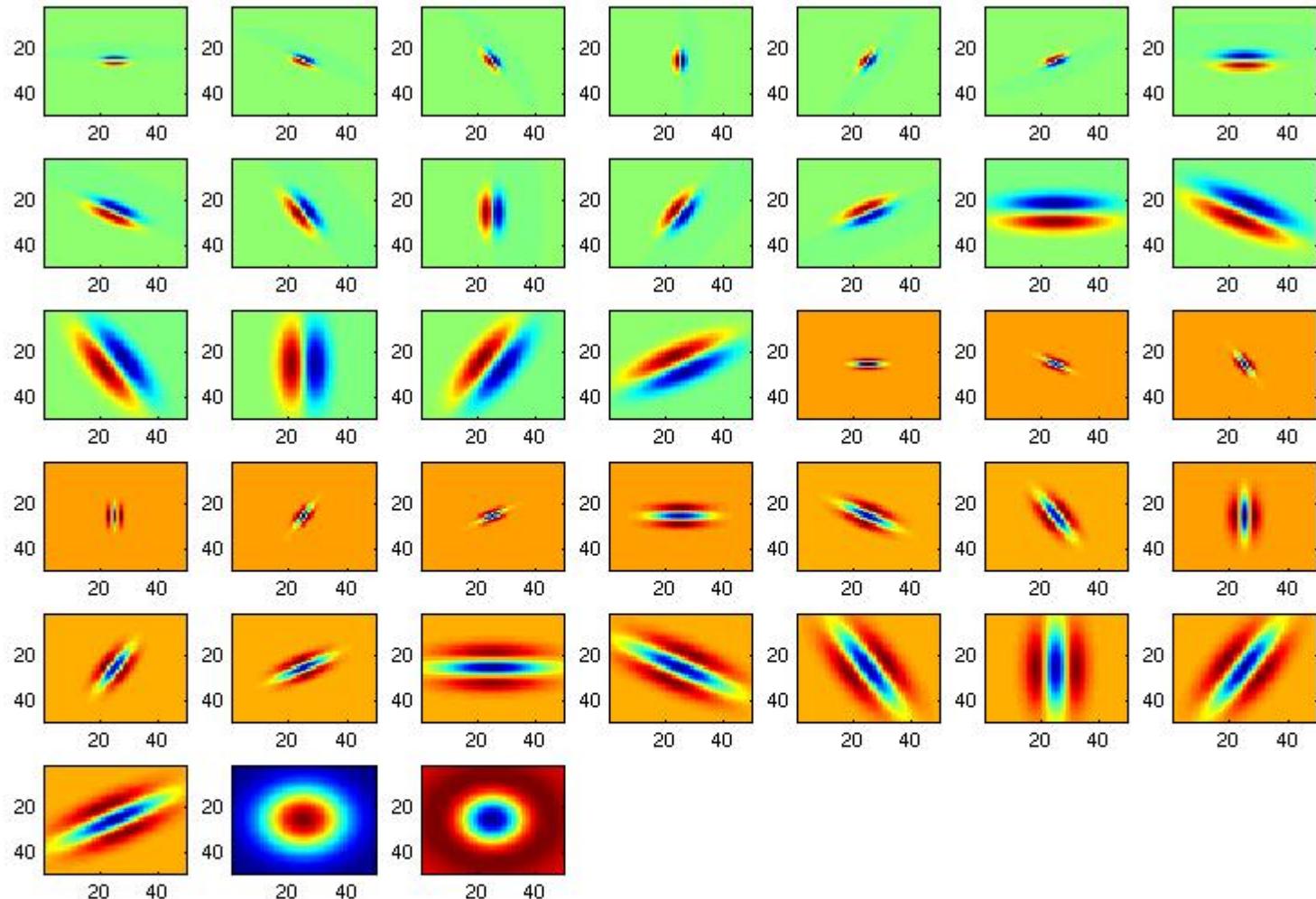
roșu – valori mari, albastru – valori mici

# Filtru pentru detectarea petelor



roșu – valori mari, albastru – valori mici

# Exemplu – colecția de filtre

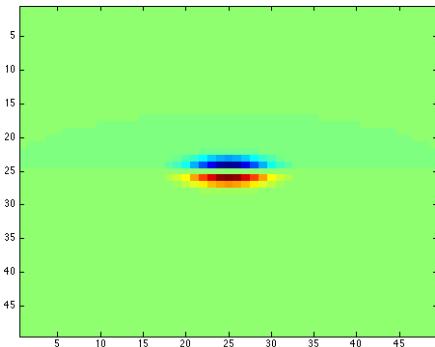


roșu – valori mari, albastru – valori mici

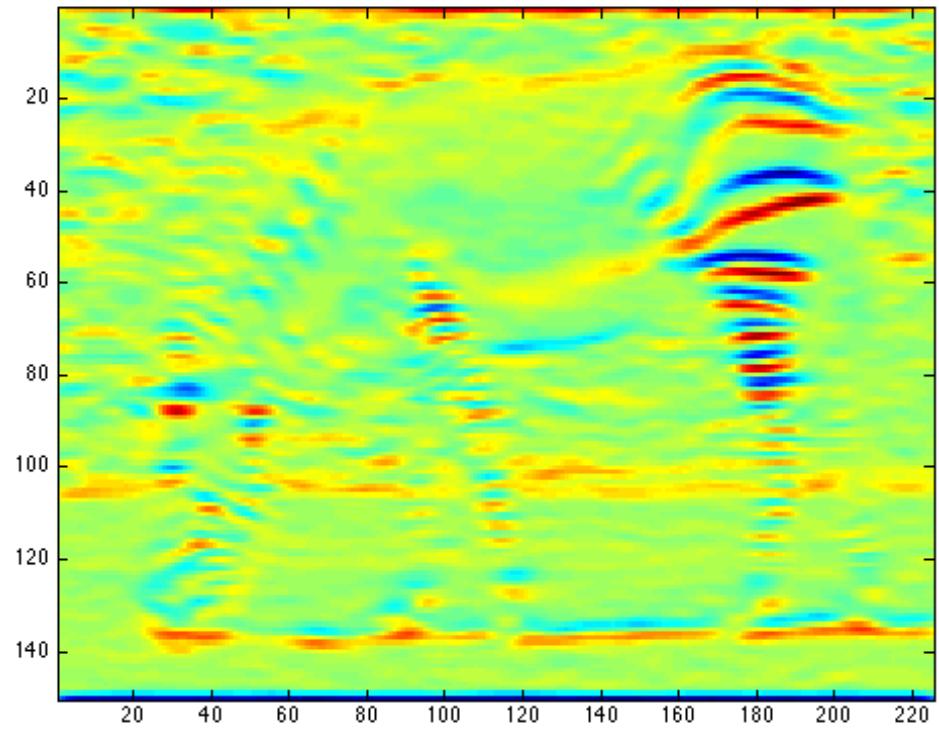
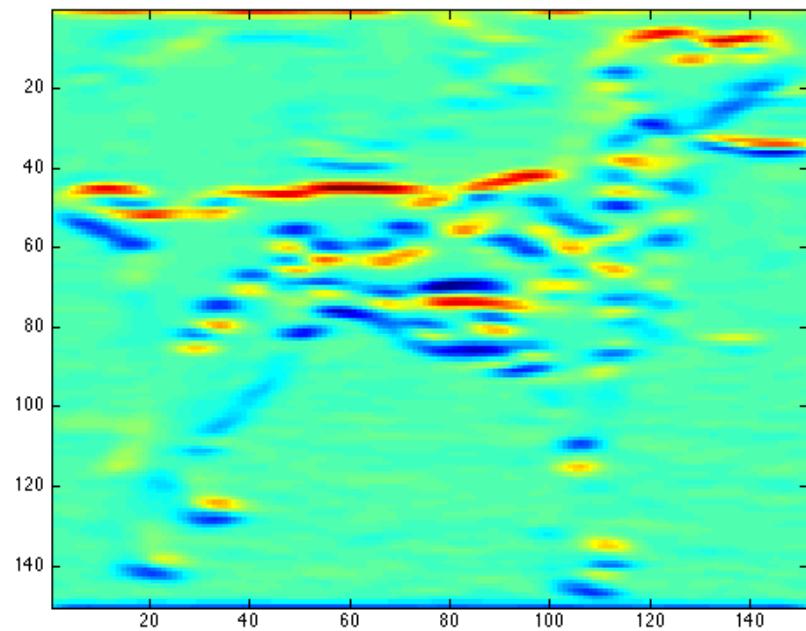
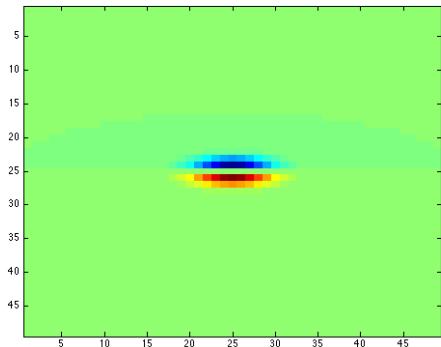
# Imagini



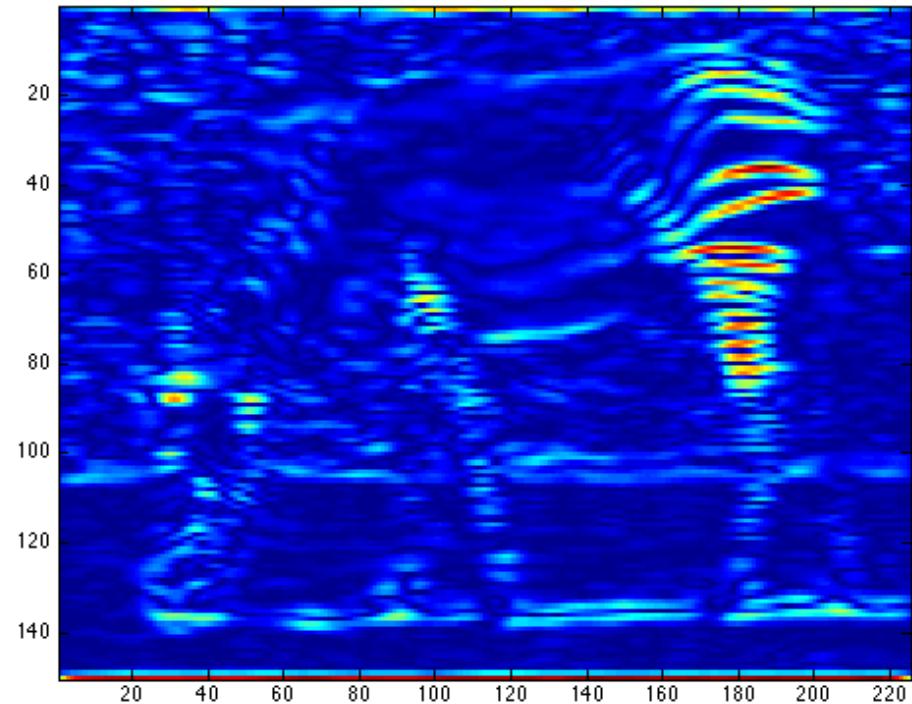
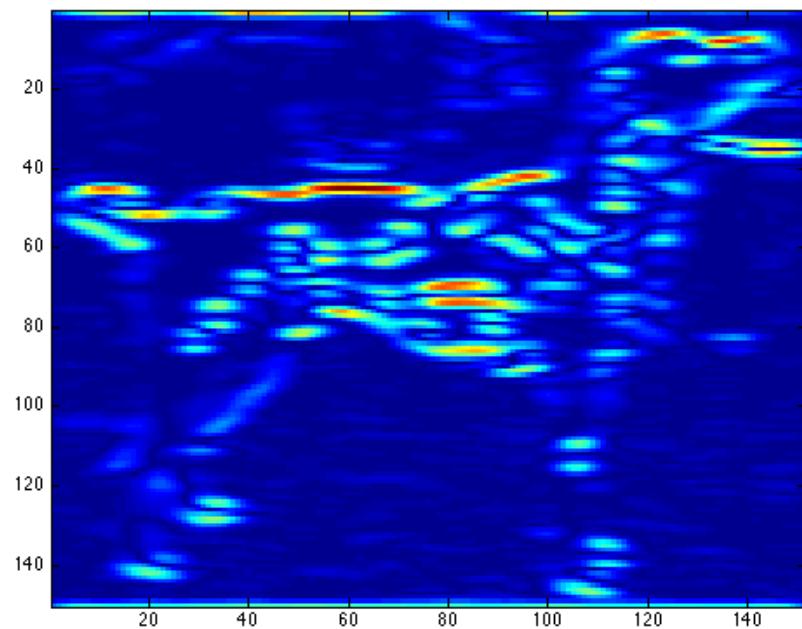
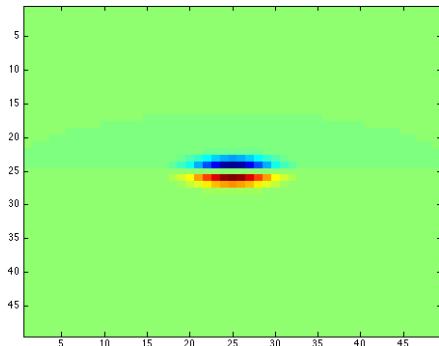
# Imagini + filtru



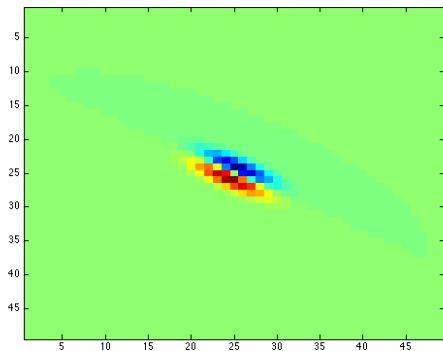
# Imagini filtrate



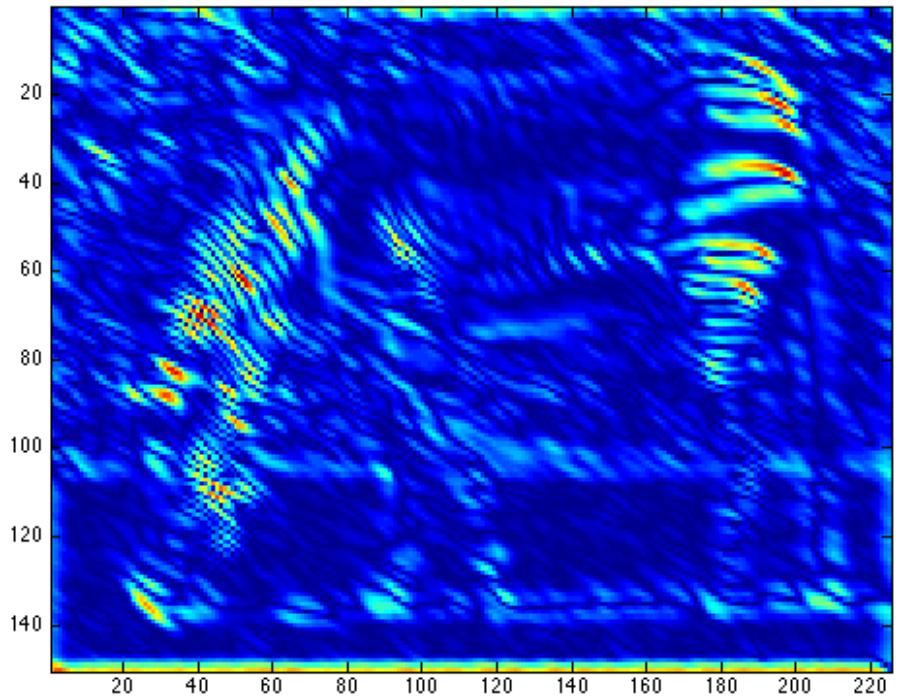
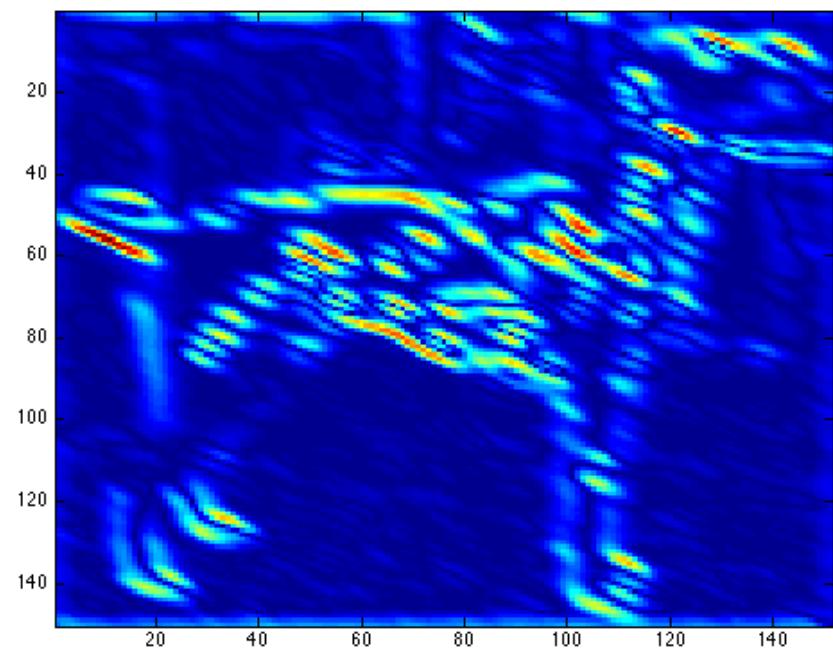
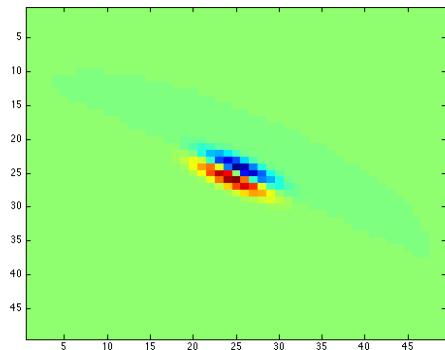
# Imagini filtrate cu răspuns absolut



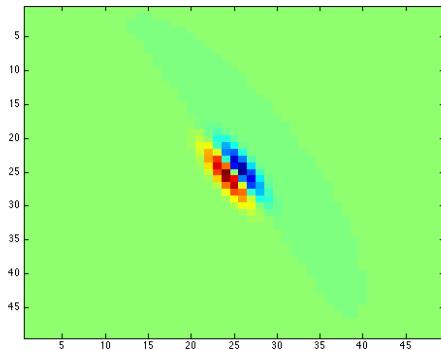
# Imagini + filtru



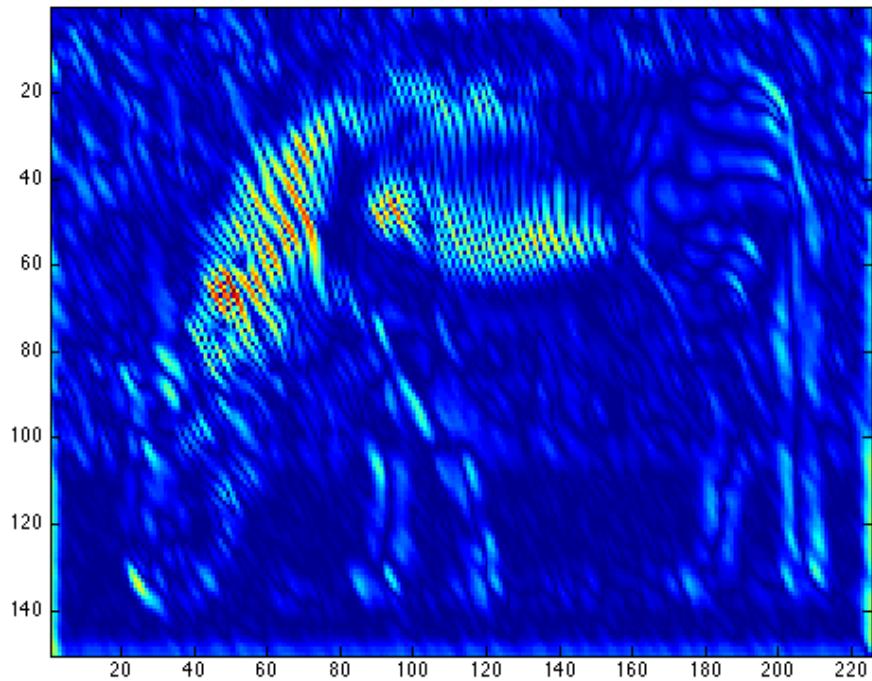
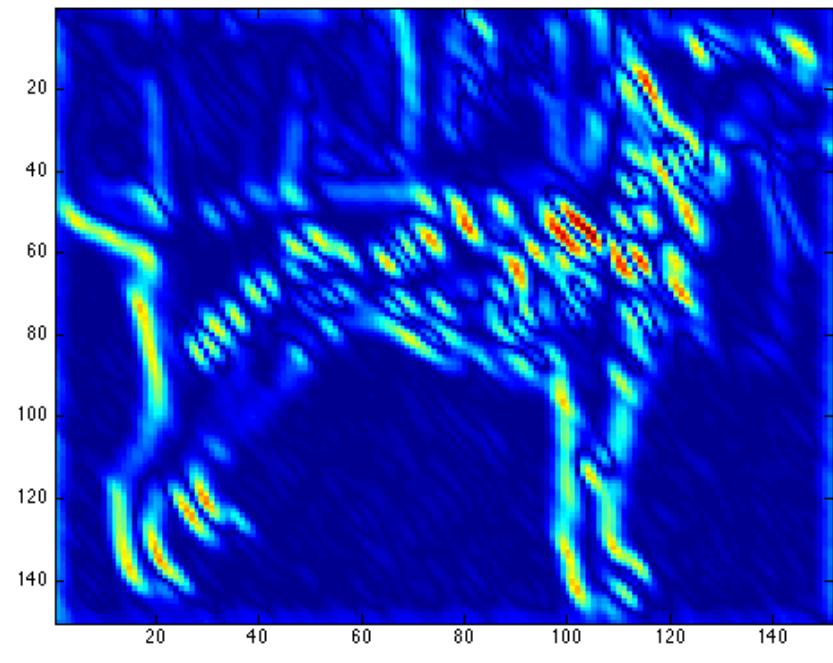
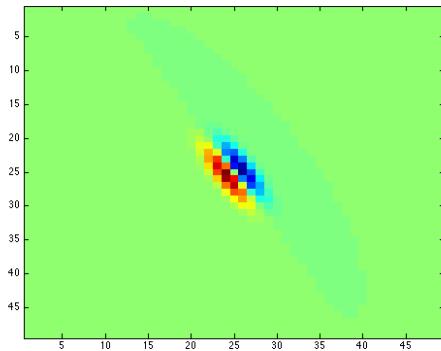
# Imagini filtrate cu răspuns absolut



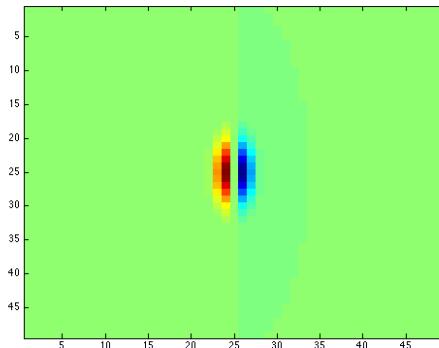
# Imagini + filtru



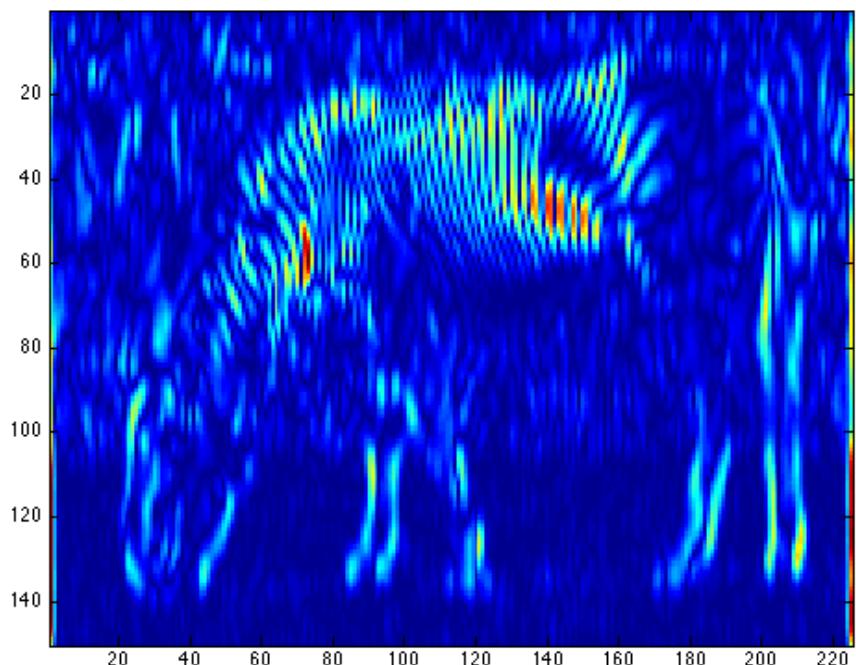
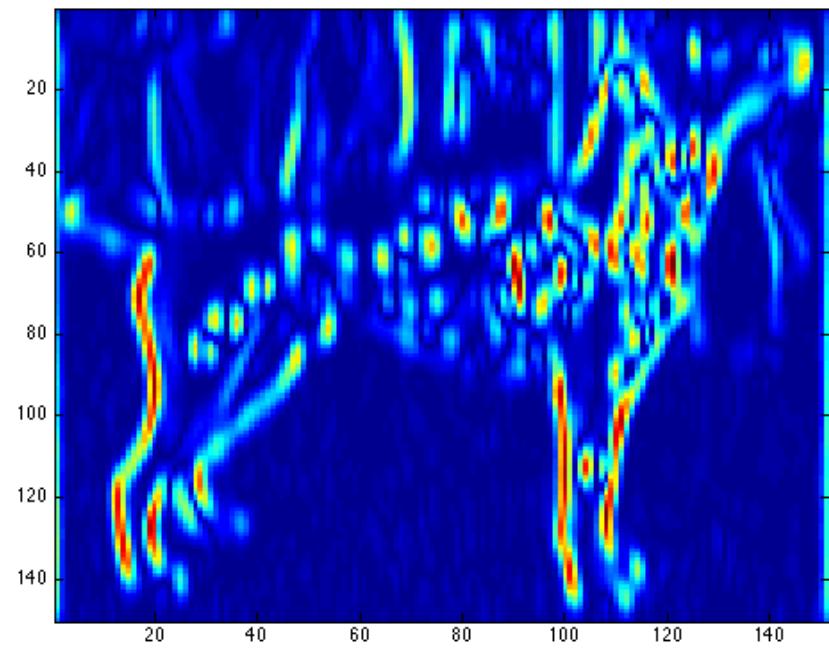
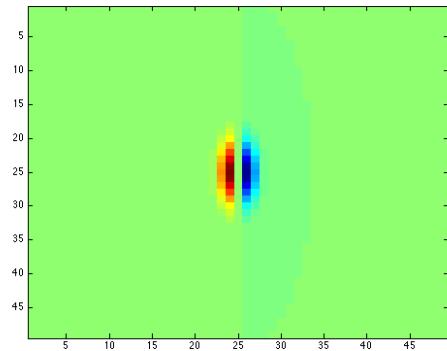
# Imagini filtrate cu răspuns absolut



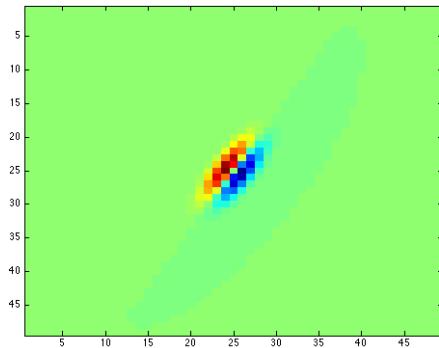
# Imagini + filtru



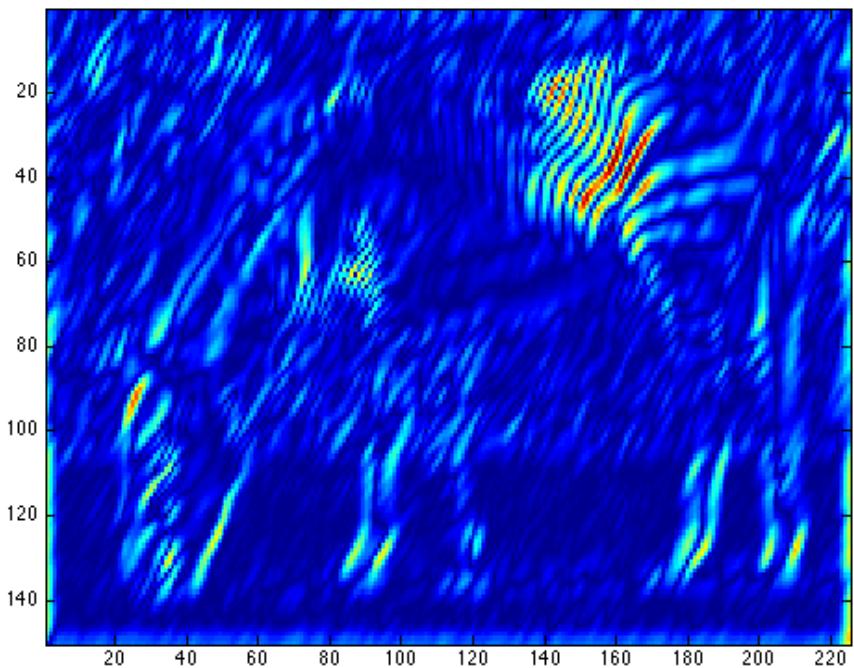
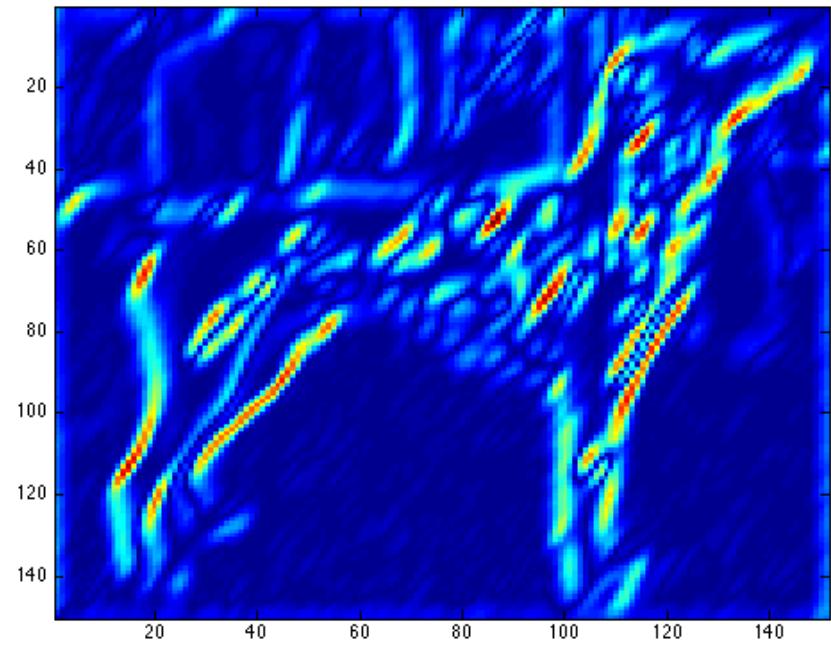
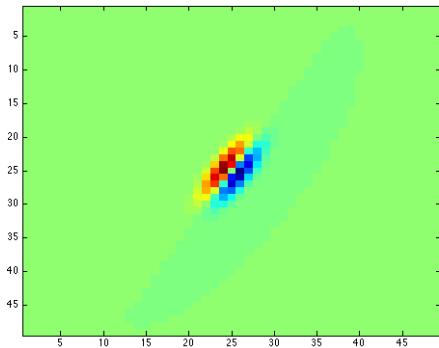
# Imagini filtrate cu răspuns absolut



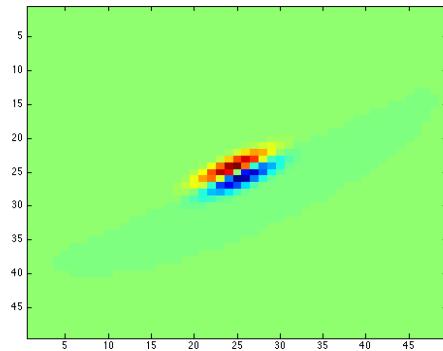
# Imagini + filtru



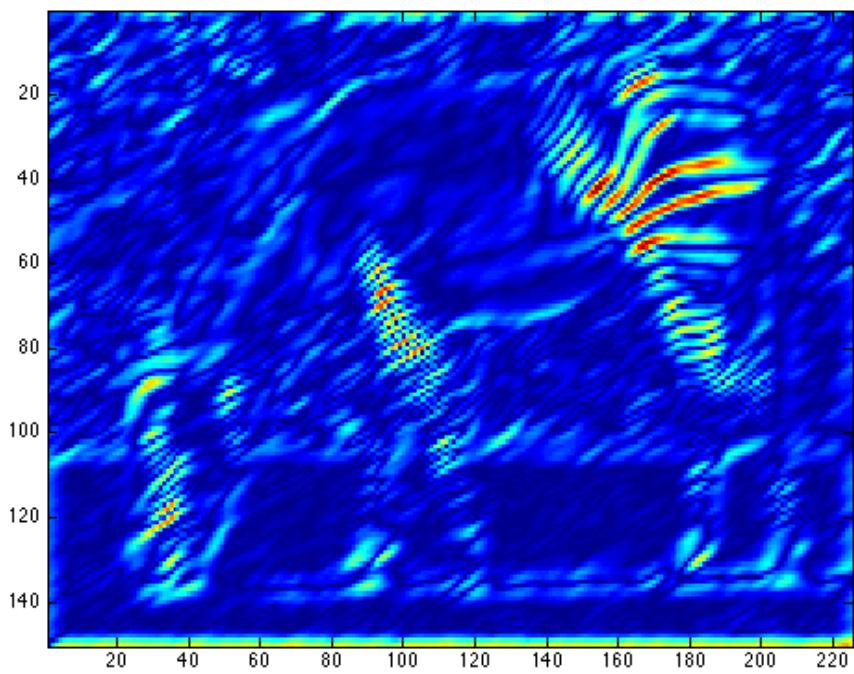
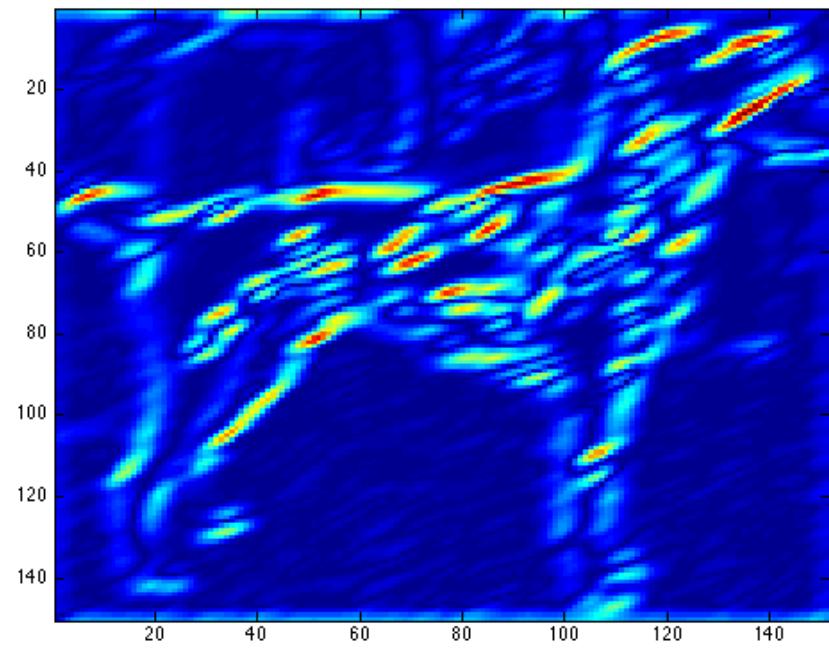
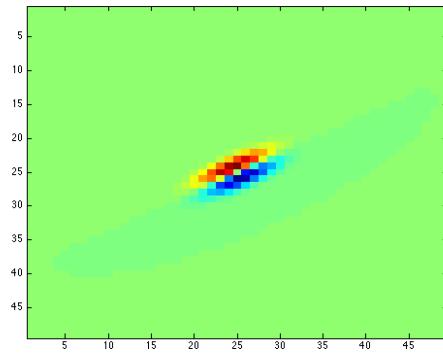
# Imagini filtrate cu răspuns absolut



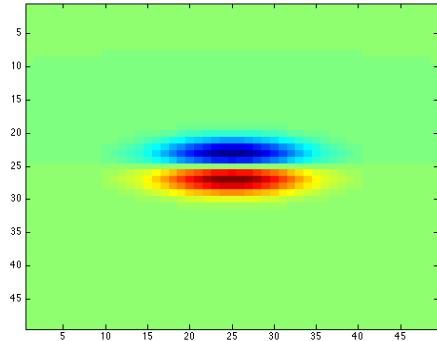
# Imagini + filtru



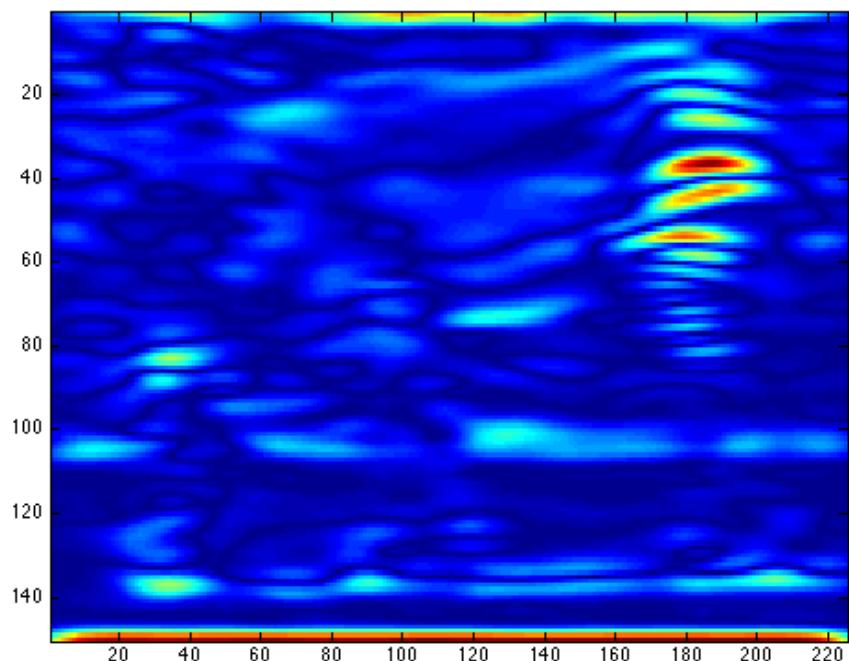
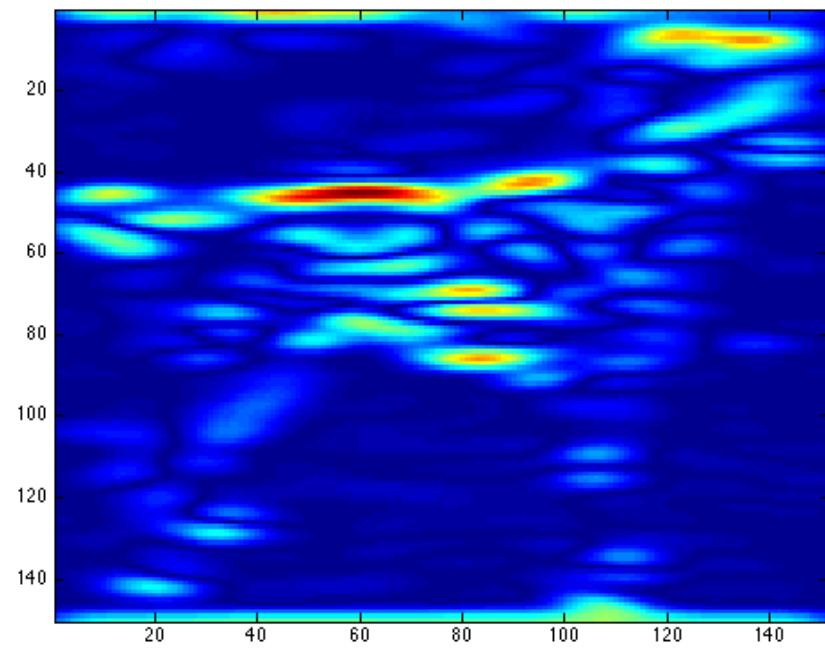
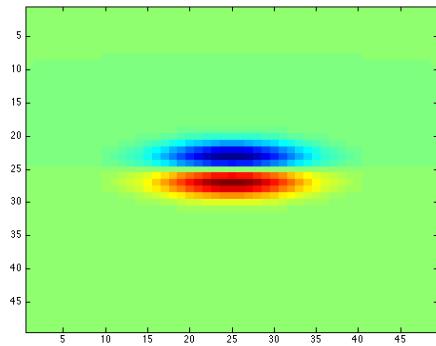
# Imagini filtrate cu răspuns absolut



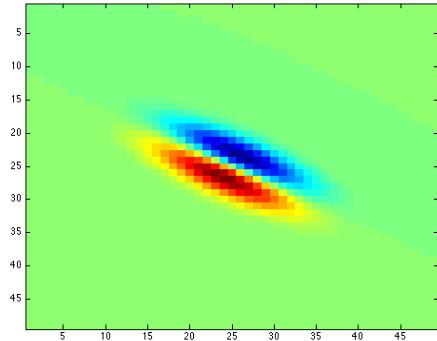
# Imagini + filtru



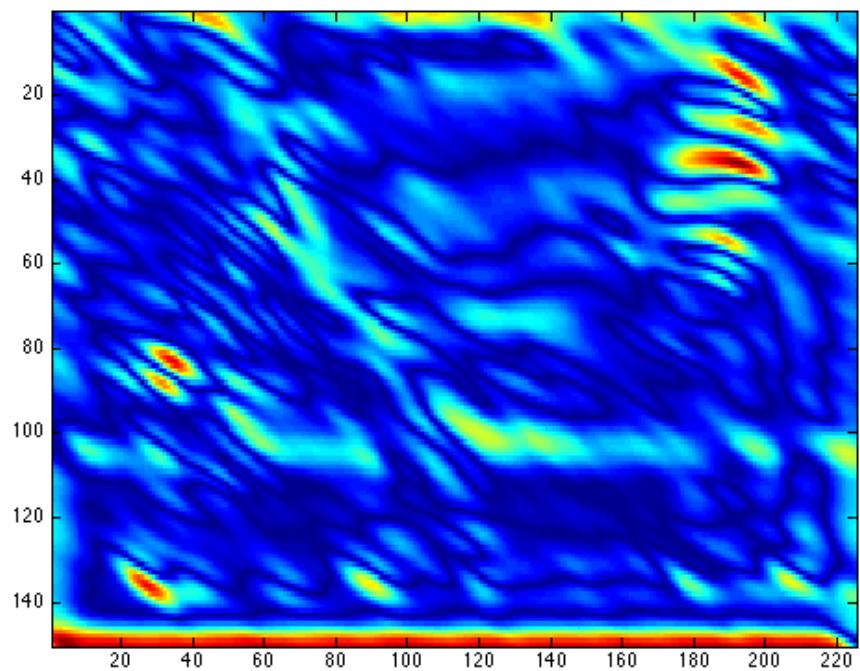
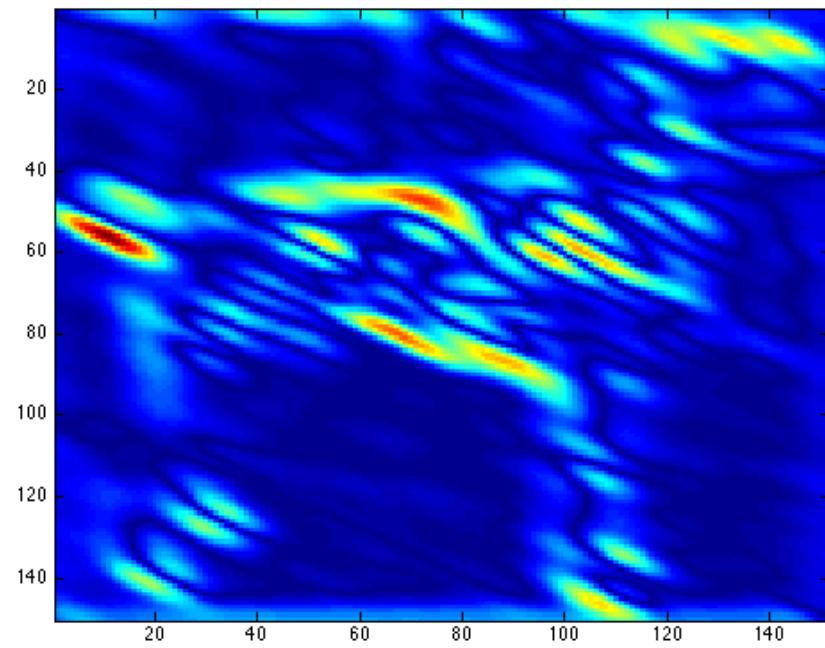
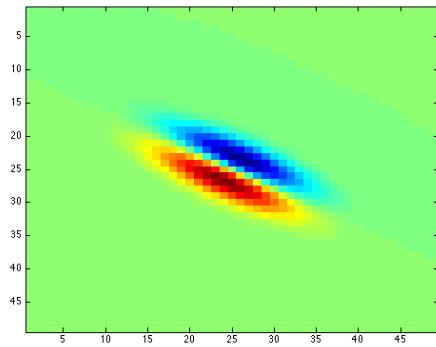
# Imagini filtrate cu răspuns absolut



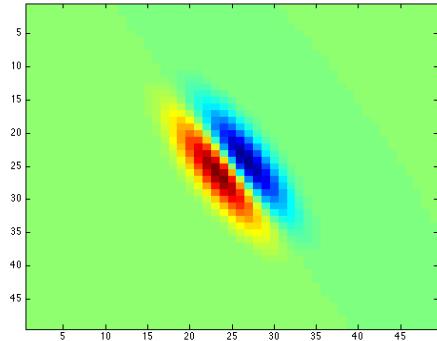
# Imagini + filtru



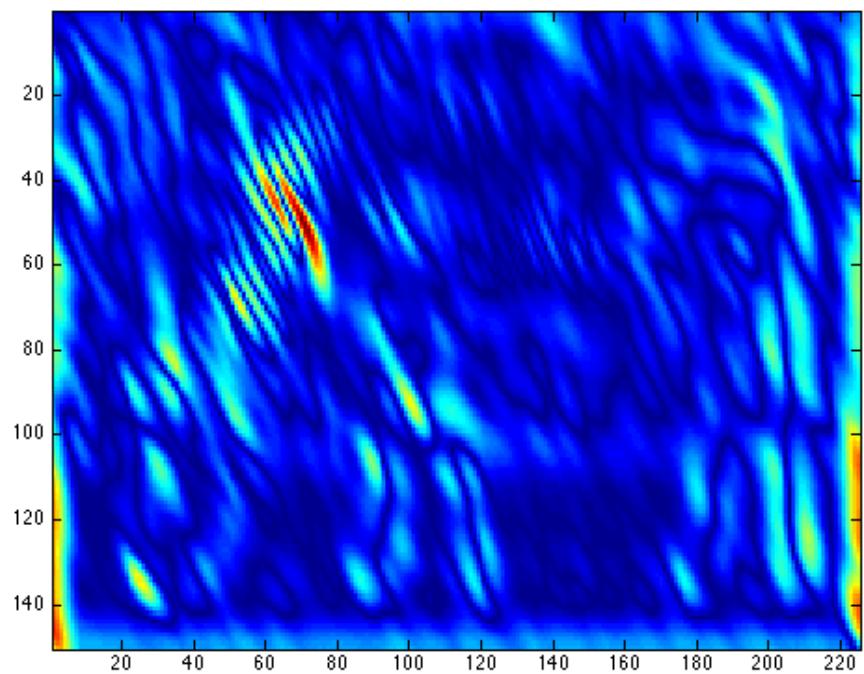
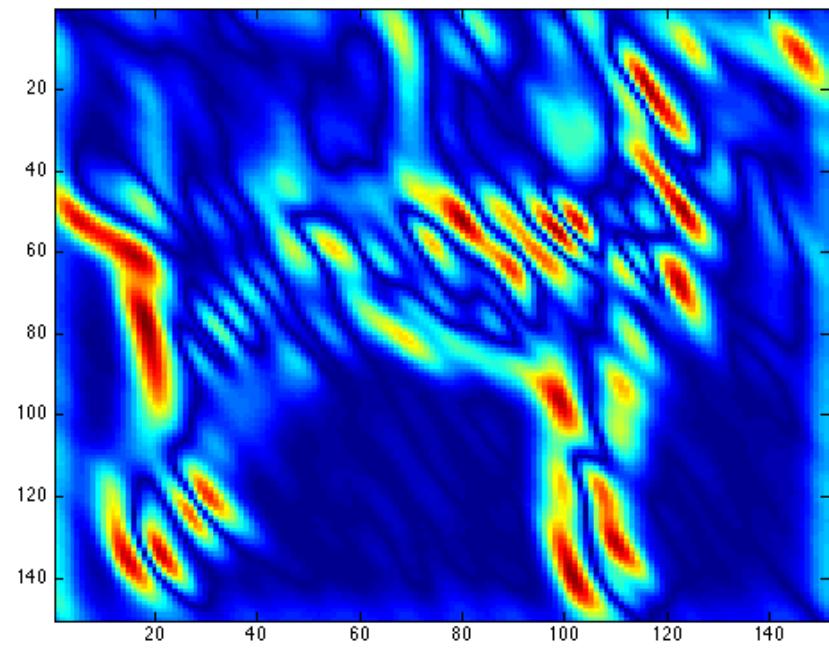
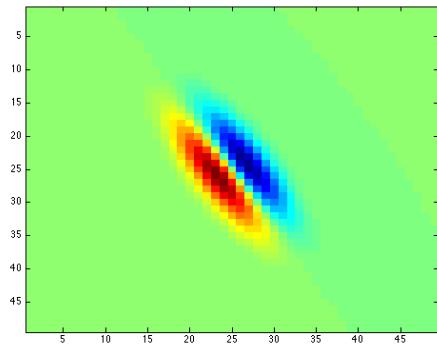
# Imagini filtrate cu răspuns absolut



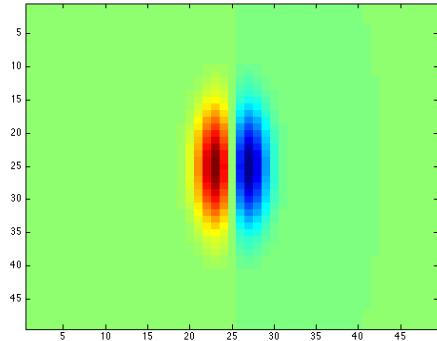
# Imagini + filtru



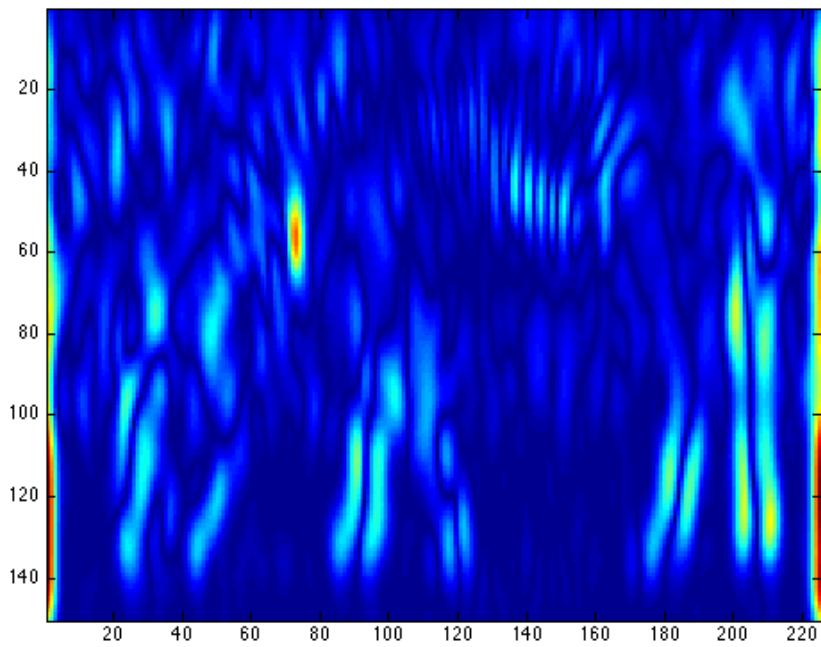
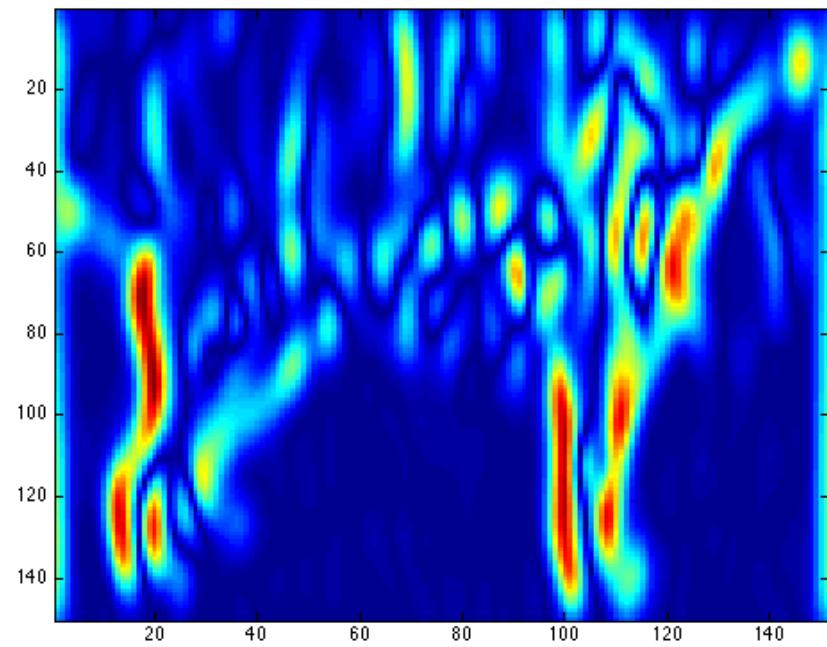
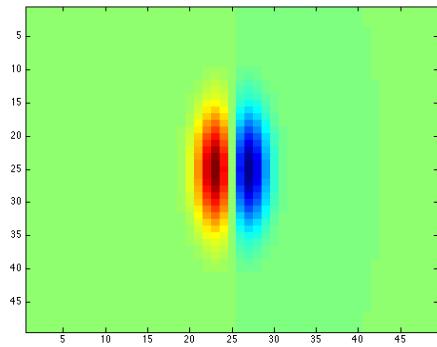
# Imagini filtrate cu răspuns absolut



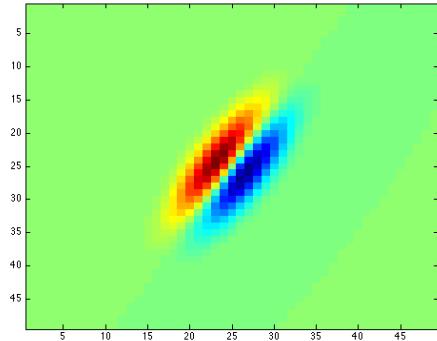
# Imagini + filtru



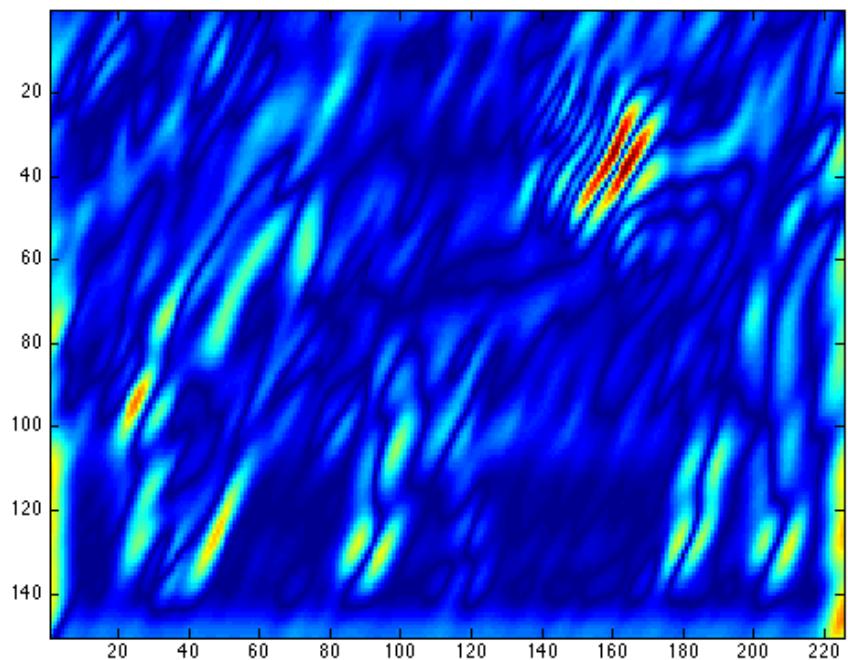
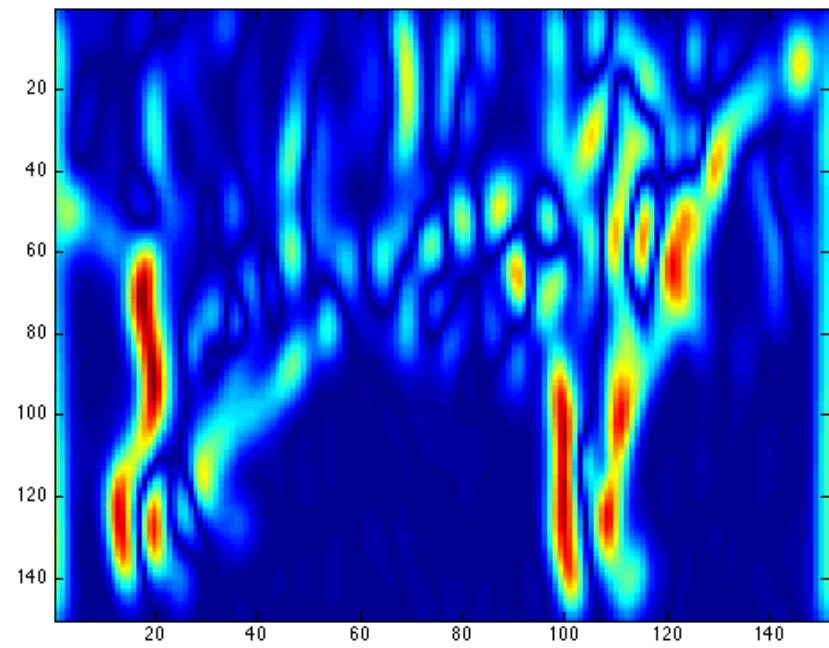
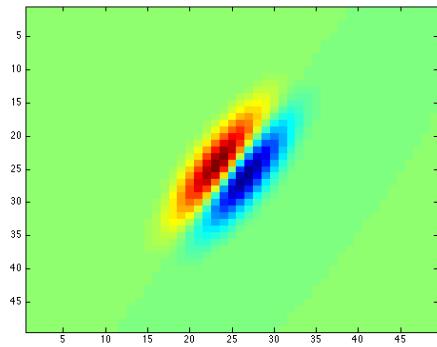
# Imagini filtrate cu răspuns absolut



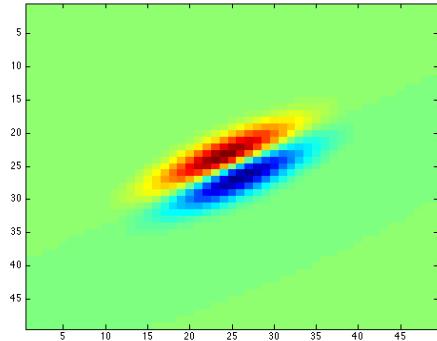
# Imagini + filtru



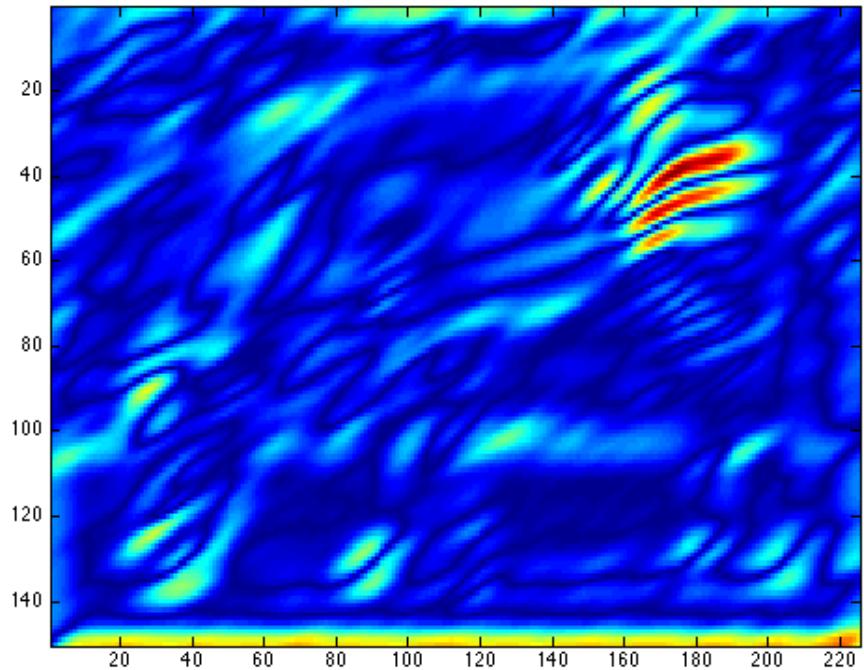
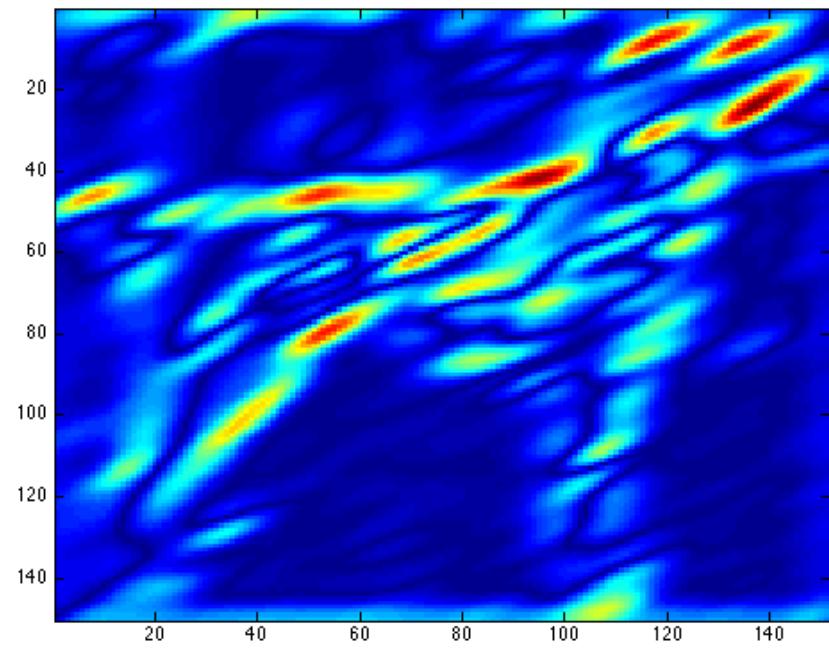
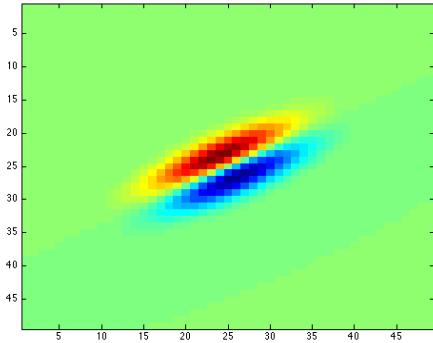
# Imagini filtrate cu răspuns absolut



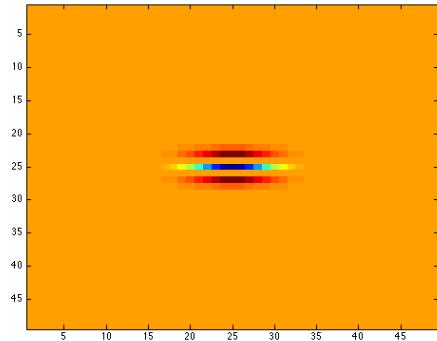
# Imagini + filtru



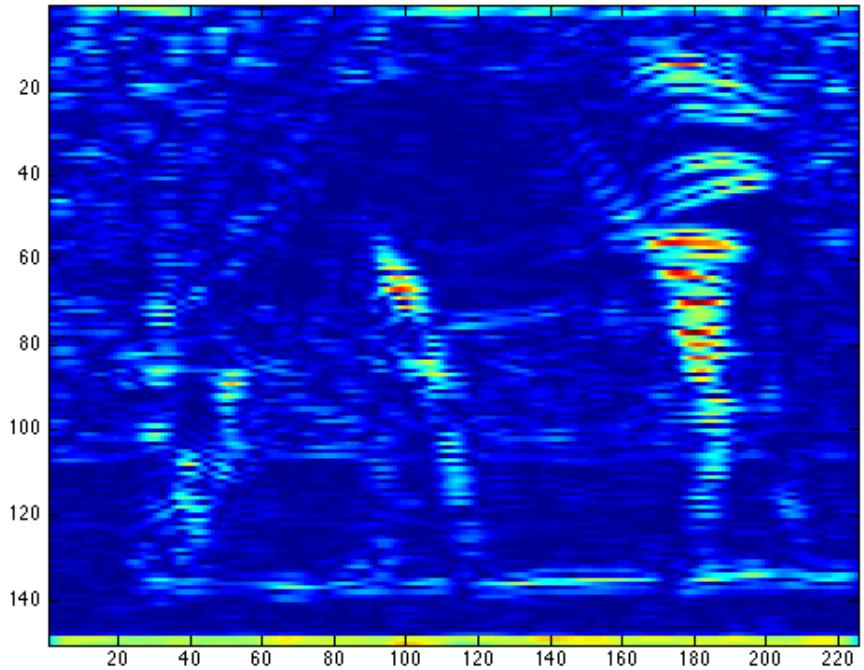
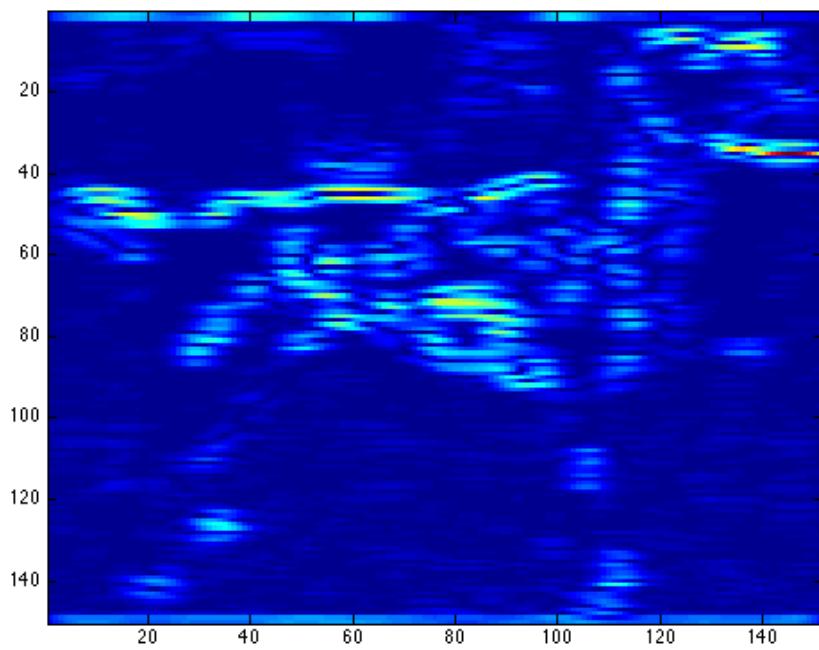
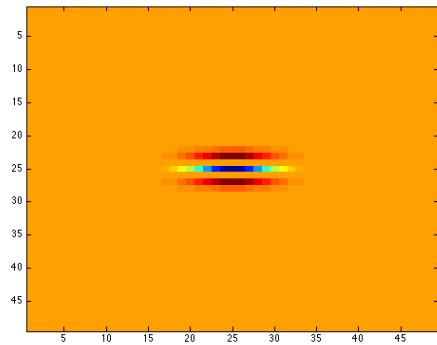
# Imagini filtrate cu răspuns absolut



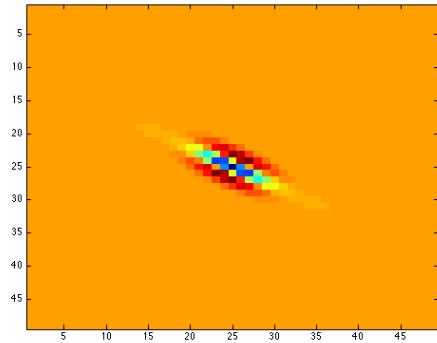
# Imagini + filtru



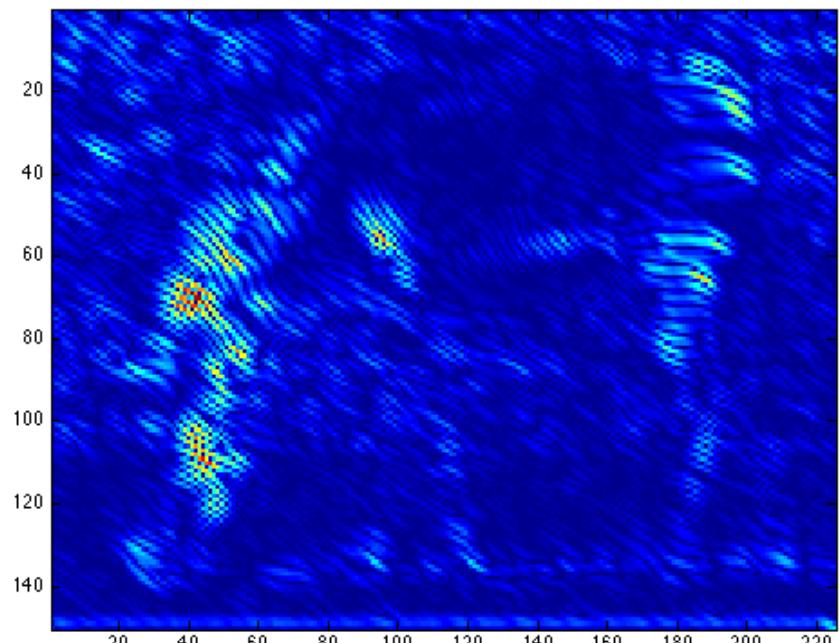
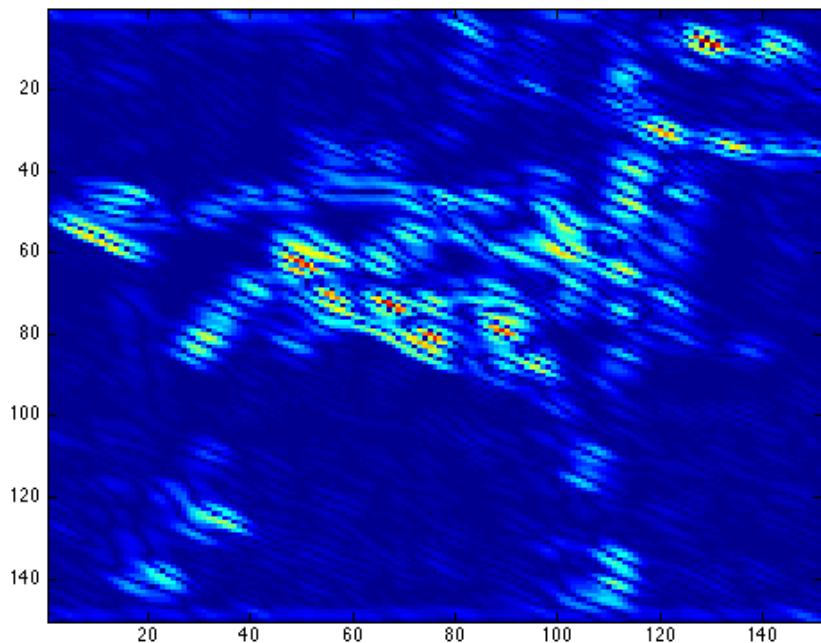
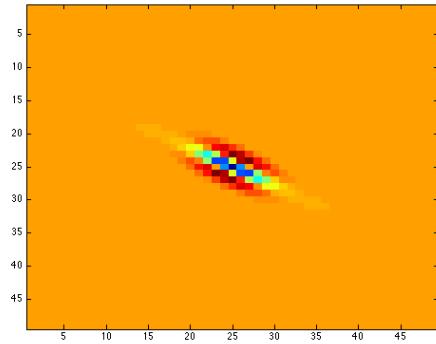
# Imagini filtrate cu răspuns absolut



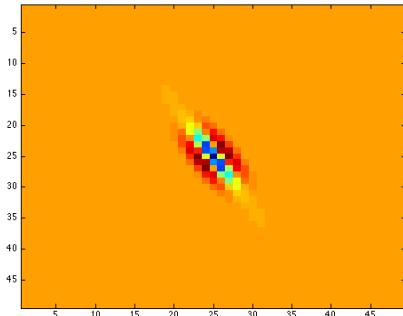
# Imagini + filtru



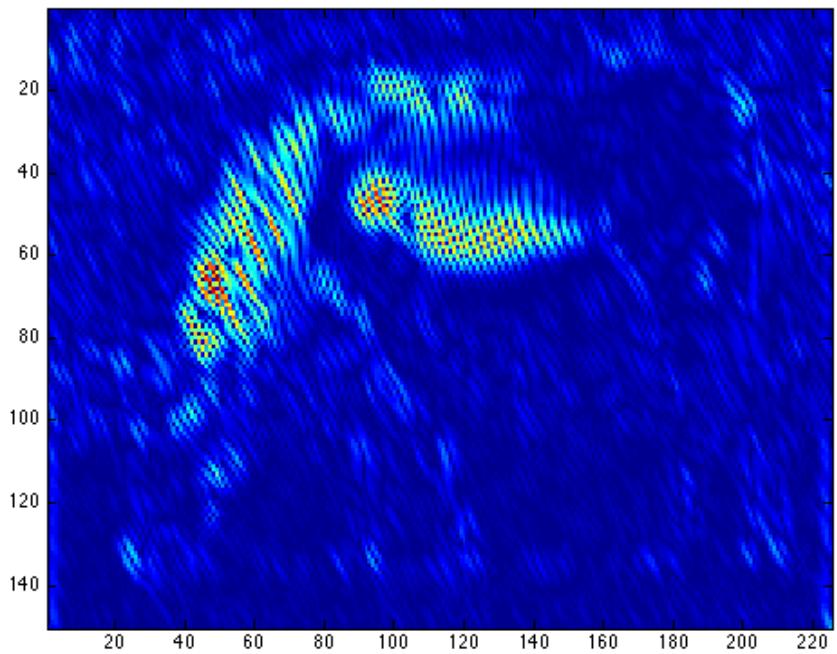
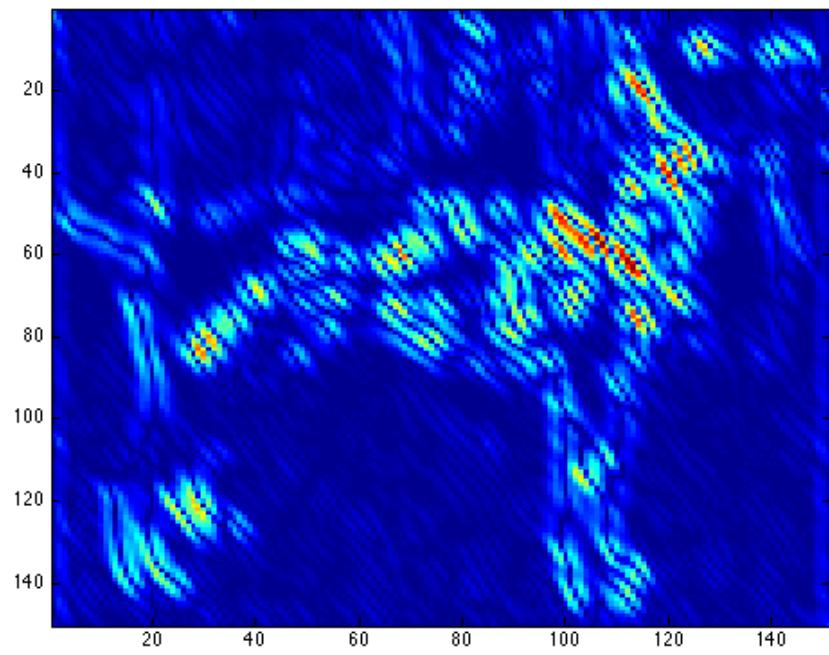
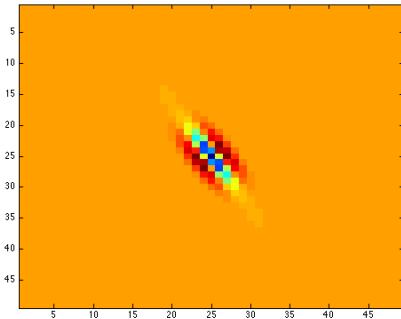
# Imagini filtrate cu răspuns absolut



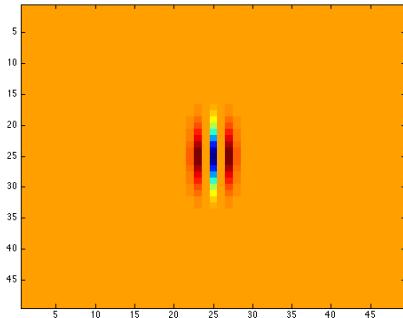
# Imagini + filtru



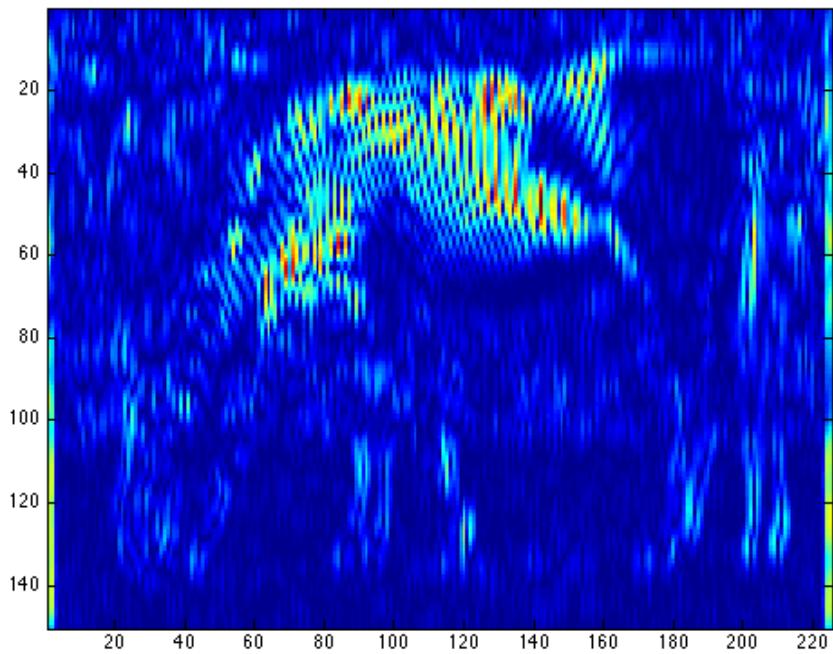
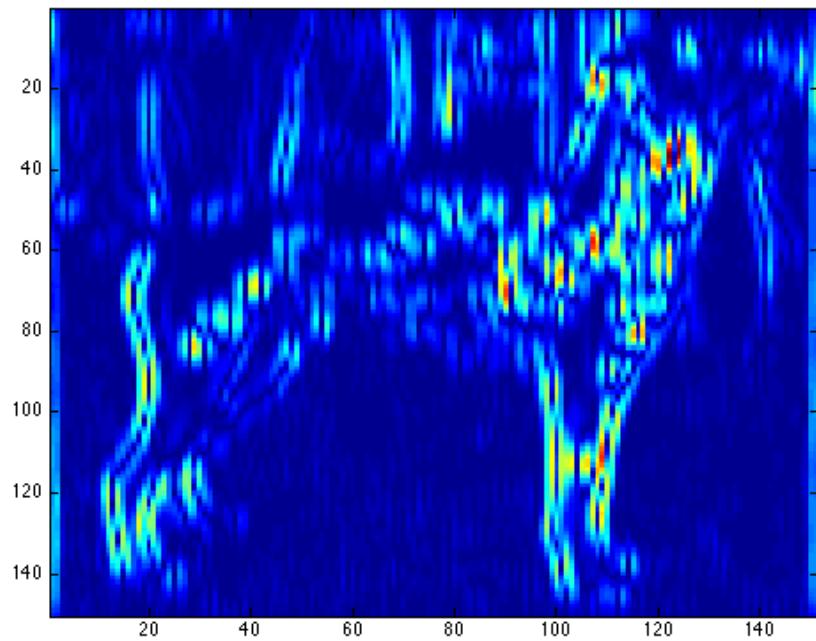
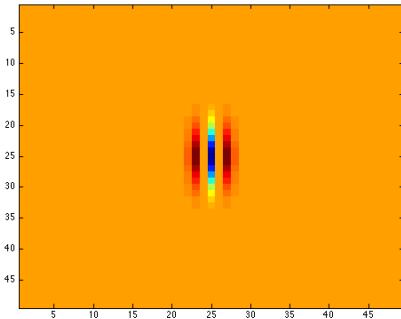
# Imagini filtrate cu răspuns absolut



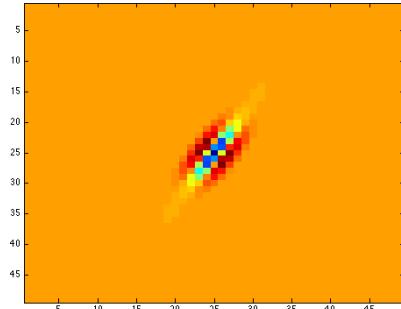
# Imagini + filtru



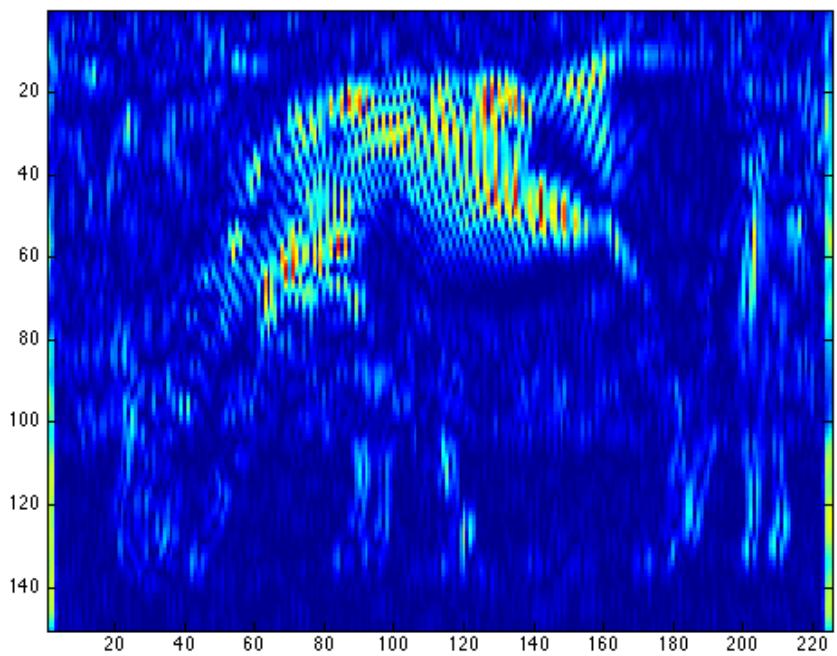
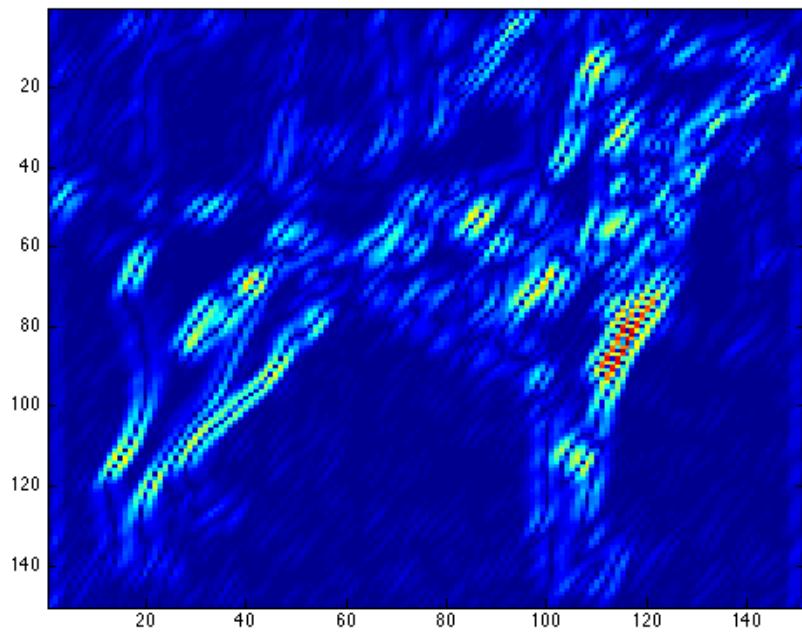
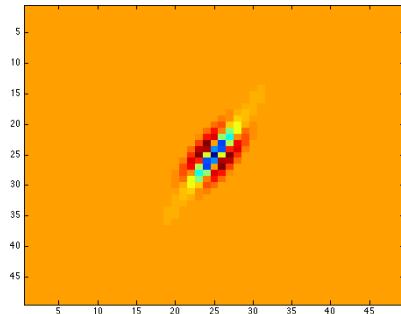
# Imagini filtrate cu răspuns absolut



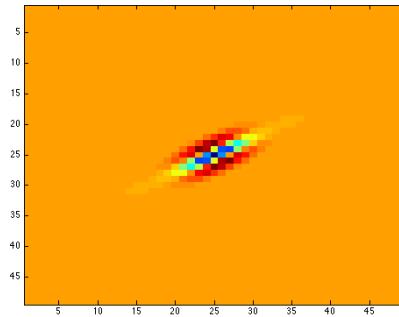
# Imagini + filtru



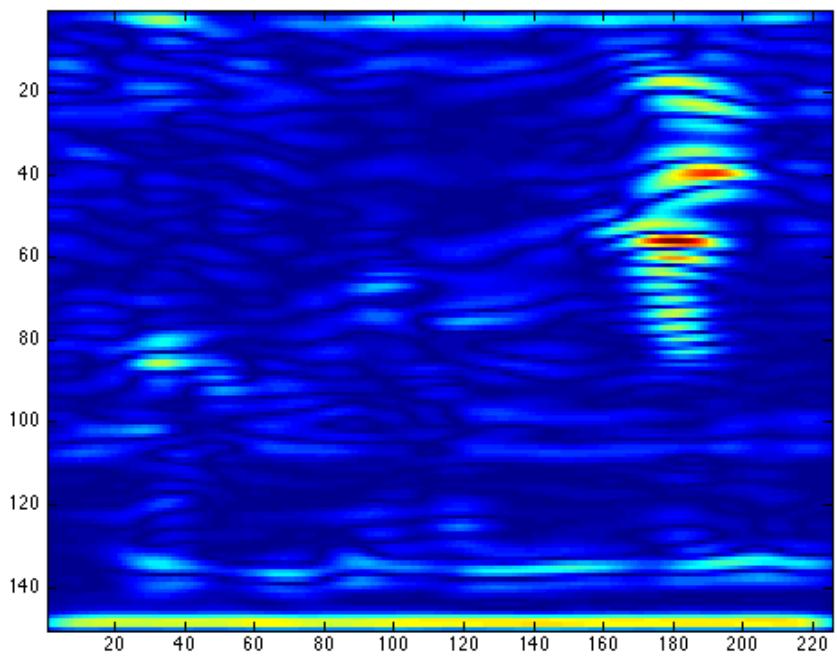
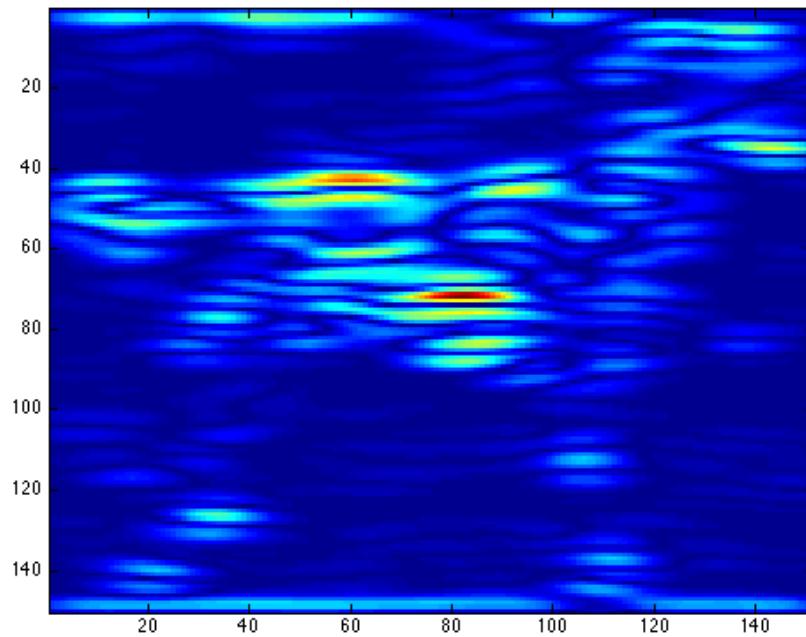
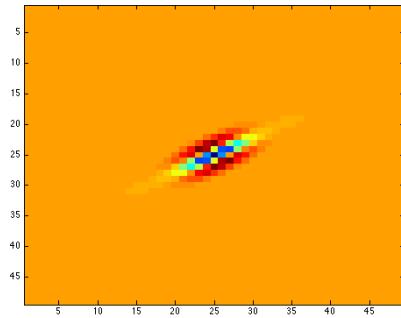
# Imagini filtrate cu răspuns absolut



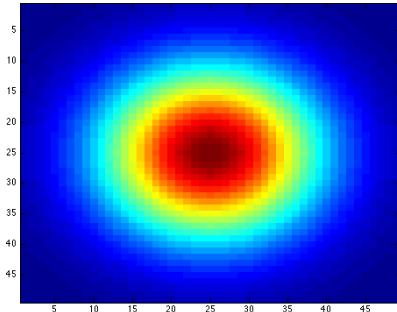
# Imagini + filtru



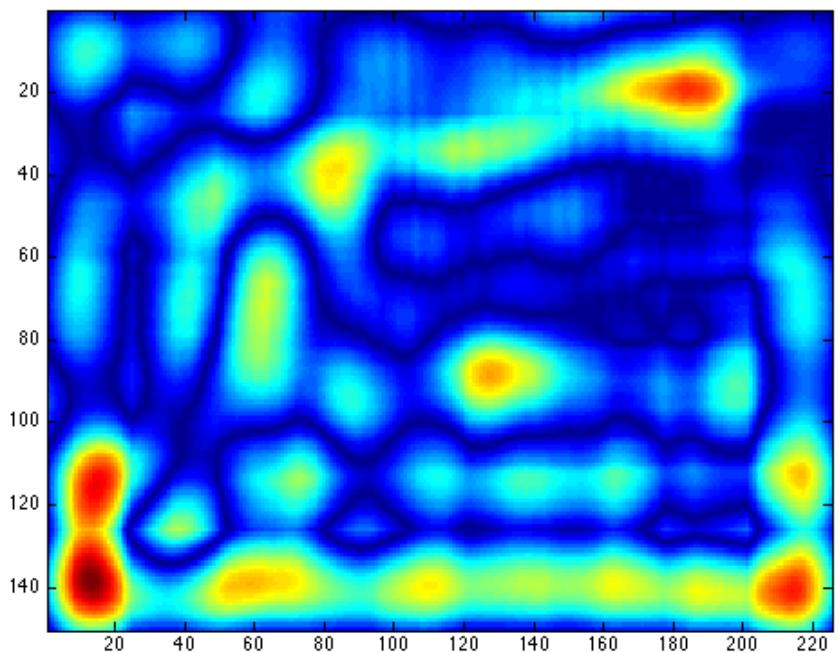
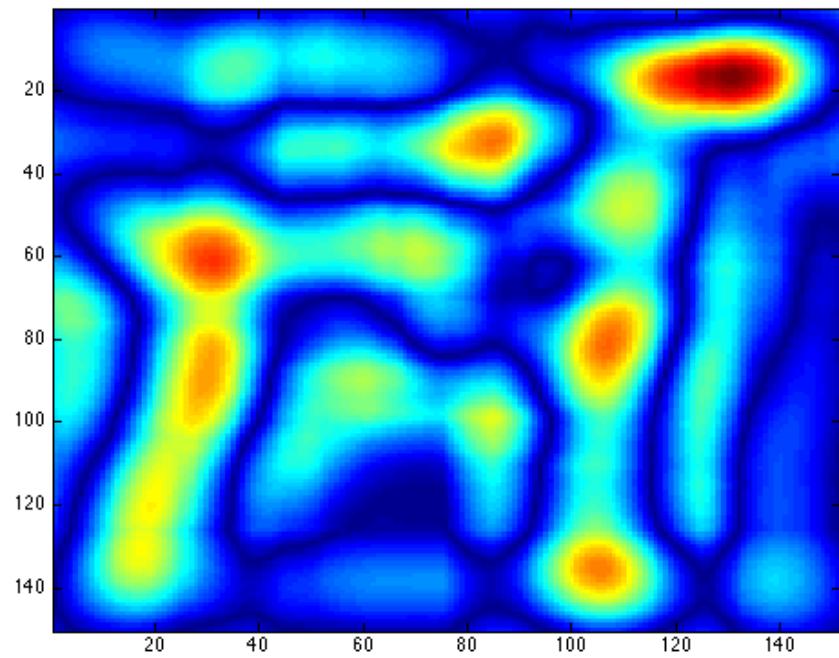
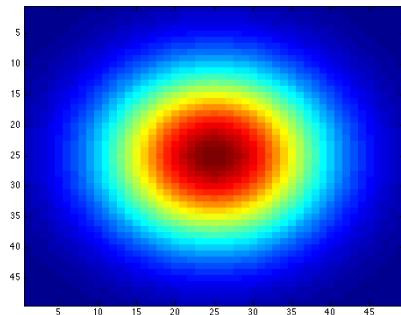
# Imagini filtrate cu răspuns absolut



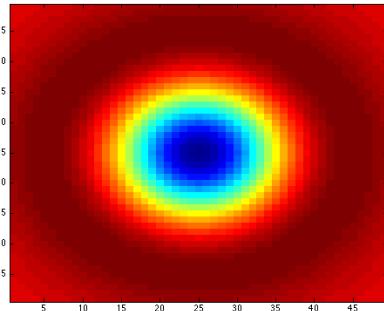
# Imagini + filtru



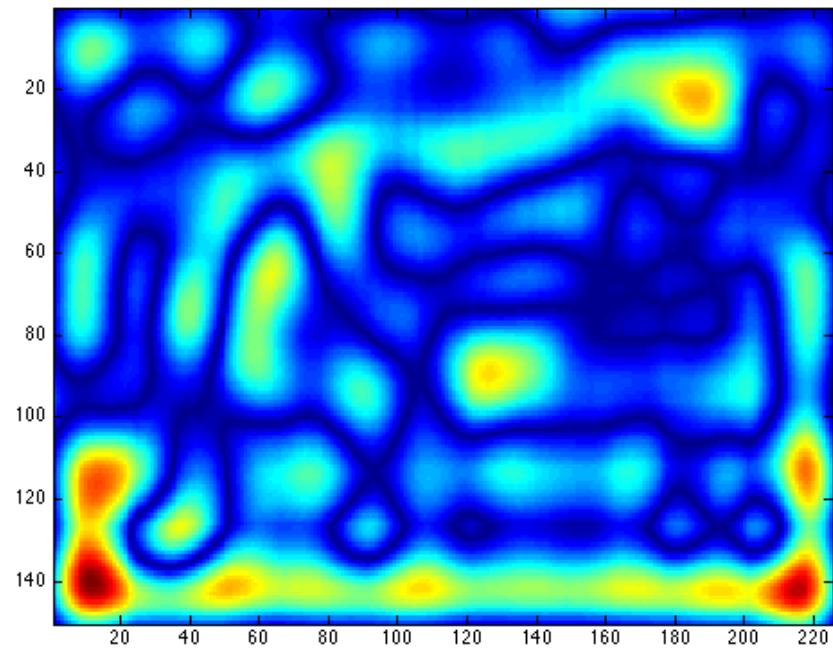
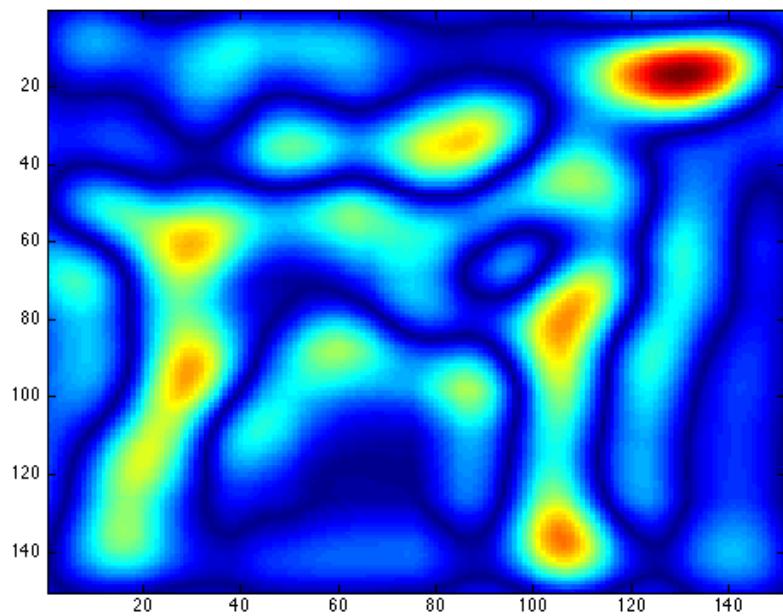
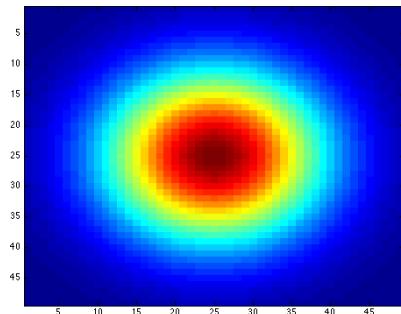
# Imagini filtrate cu răspuns absolut



# Imagini + filtru

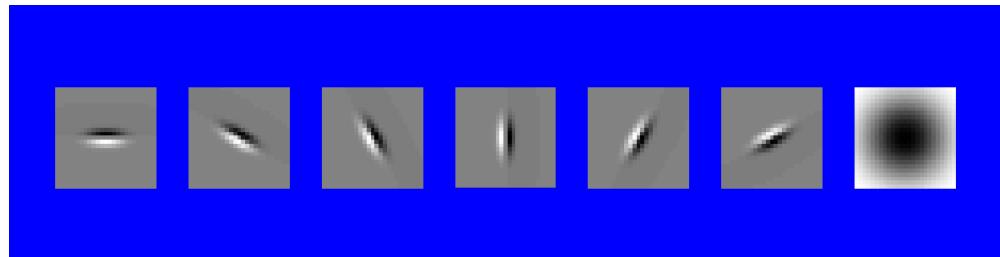


# Imagini filtrate cu răspuns absolut

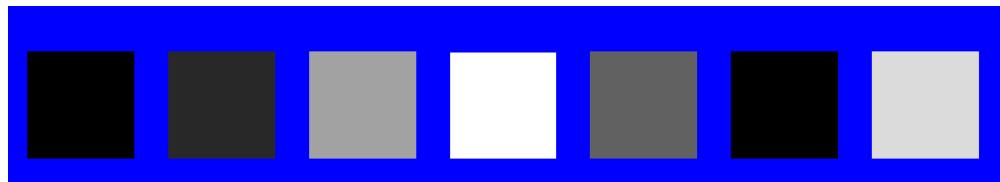


# Găsiți corespondența!

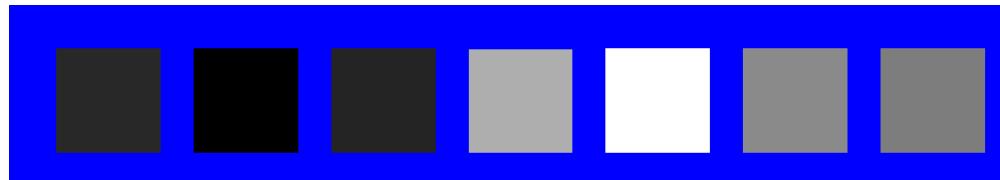
Colecție de filtre



1



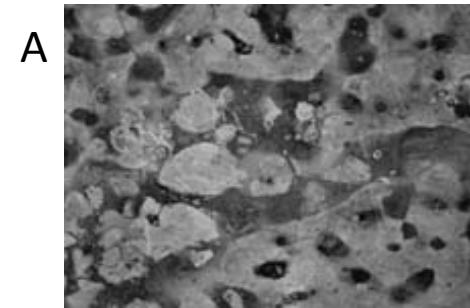
2



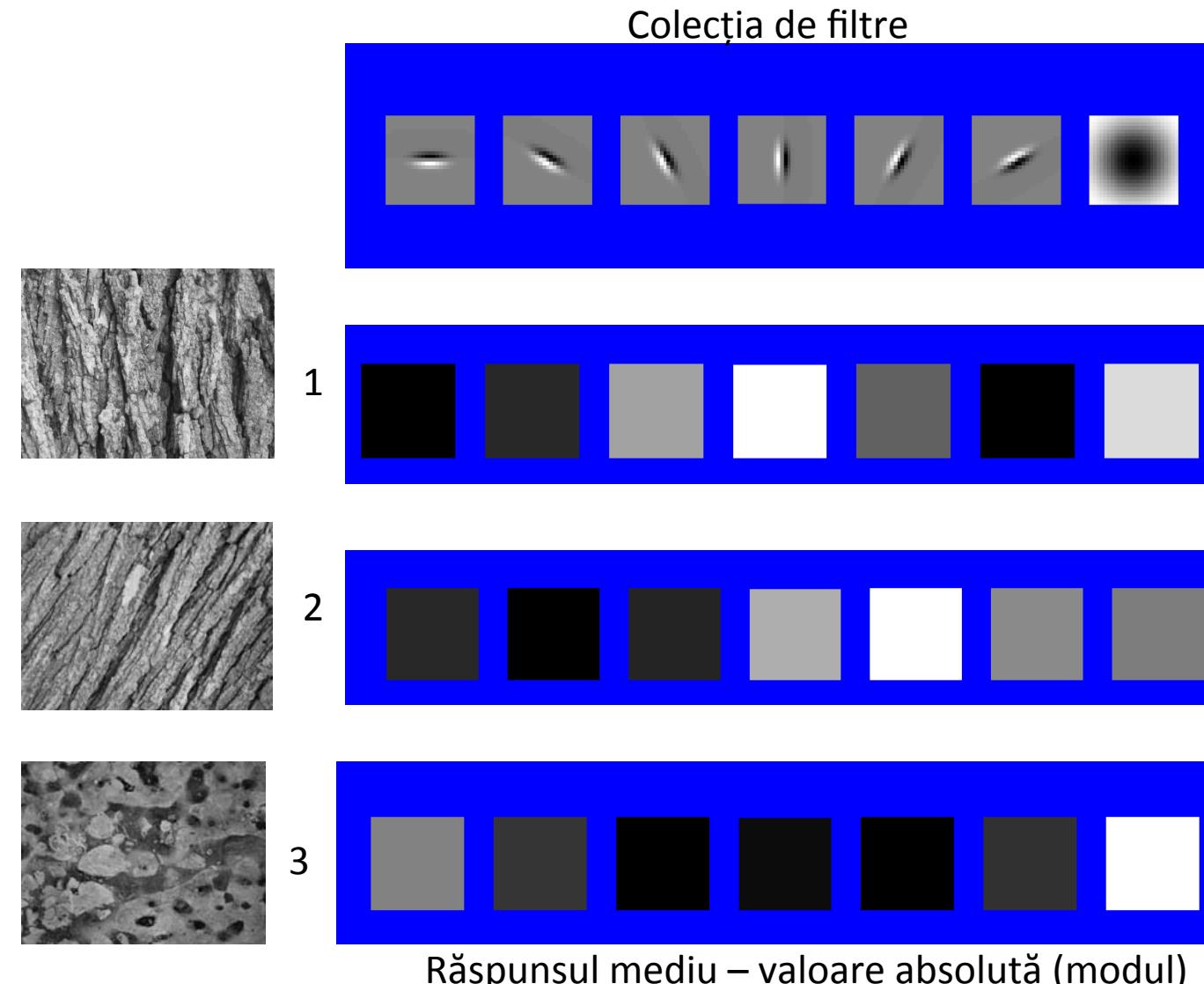
3



Răspunsul mediu – valoare absolută (modul)

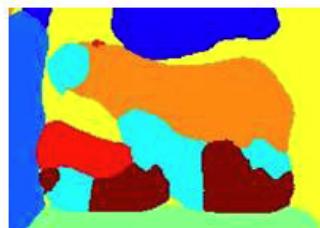


# Reprezentarea texturilor pe baza răspunsului mediu în modul



# De ce studiem textura?

- **Segmentare/clasificare**
  - analiza, reprezentarea texturii
  - grupează regiunile din imagine cu aceeași textură



- **Sinteză**
  - dată o mostră de textură (dimensiuni mici) vrem să generăm o textură similară (dimensiuni mari)

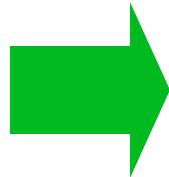


mostră de textură

textură generată

# Sinteza texturii

- Scop: generarea de noi exemple de textură (dimensiuni mari) pe baza unui mostre (dimensiuni mici)
- Aplicații: îmbogățirea mediilor virtuale cu textură, jocuri pe calculator, acoperirea găurilor

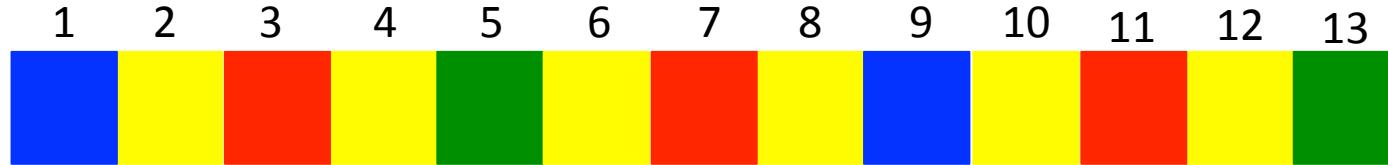


# Obstacole

- Analiza texturii: cum cuprindem esența texturii?
- Texturi regulate – texeli care se repetă
- Texturi stocastice – nu prezintă în mod explicit texeli
- Vrem să modelăm întreg spectrul de texturi: de la texturi ce se repetă la texturi stocastice



# Probabilitate



$$P(\text{pixel} = \boxed{\text{blue}}) = 2/13$$

$$P(\text{pixel} = \boxed{\text{red}}) = 3/13$$

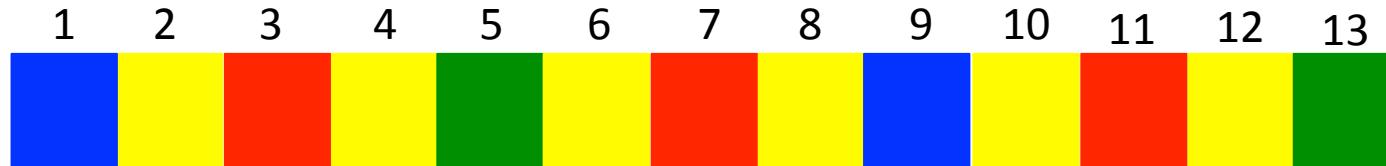
$$P(\text{pixel} = \boxed{\text{yellow}}) = 6/13$$

$$P(\text{pixel} = \boxed{\text{green}}) = 2/13$$

$$2/13 + 6/13 + 3/13 + 2/13 = 1$$

Probabilitate = Nr cazuri favorabile / Nr cazuri totale  
Suma = 1

# Probabilitate condiționată



$$P(\text{pixel} = \boxed{\text{red}} \mid \text{pixelul din stânga} = \boxed{\text{yellow}}) = 3/6$$

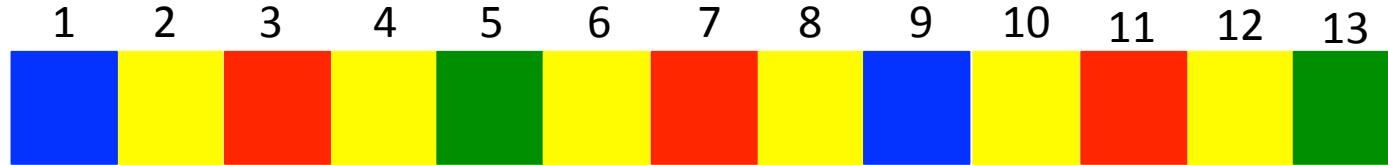
$$P(\text{pixel} = \boxed{\text{blue}} \mid \text{pixelul din stânga} = \boxed{\text{yellow}}) = 1/6$$

$$P(\text{pixel} = \boxed{\text{yellow}} \mid \text{pixelul din stânga} = \boxed{\text{yellow}}) = 0/6$$

$$P(\text{pixel} = \boxed{\text{green}} \mid \text{pixelul din stânga} = \boxed{\text{yellow}}) = 2/6$$

$$3/6 + 1/6 + 0/6 + 2/6 = 1$$

# Probabilitate condiționată



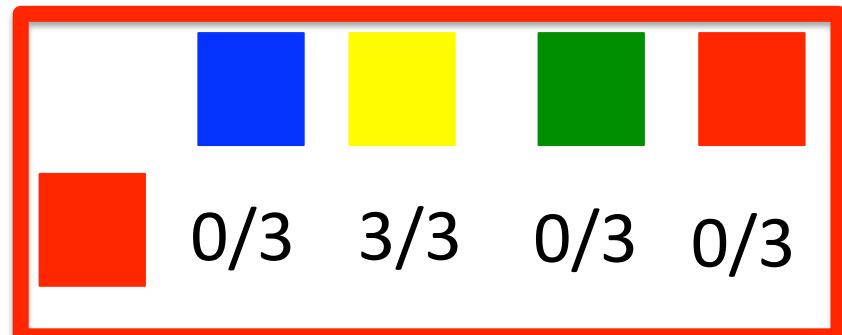
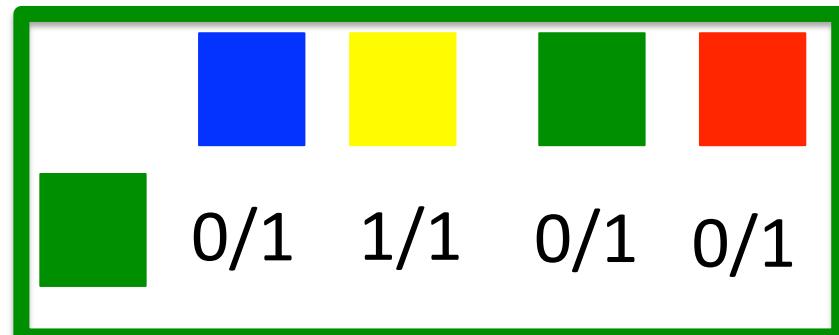
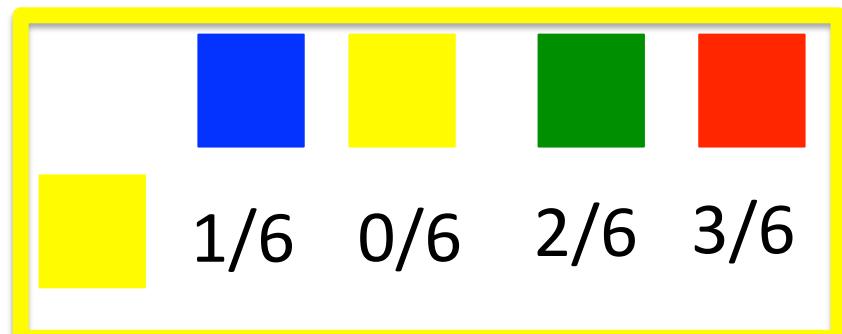
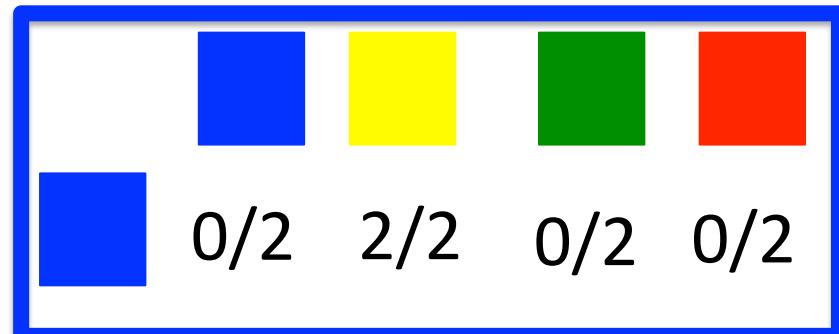
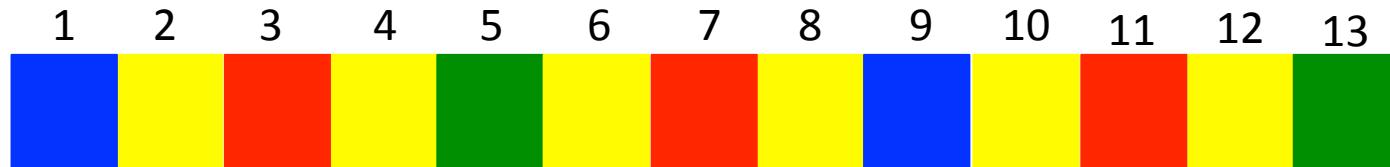
$P(\text{pixel} = \boxed{\text{red}} \mid \text{pixelul din stânga} = \boxed{\text{blue}}) = 0/2$

$P(\text{pixel} = \boxed{\text{blue}} \mid \text{pixelul din stânga} = \boxed{\text{blue}}) = 0/2$

$P(\text{pixel} = \boxed{\text{yellow}} \mid \text{pixelul din stânga} = \boxed{\text{blue}}) = 2/2$

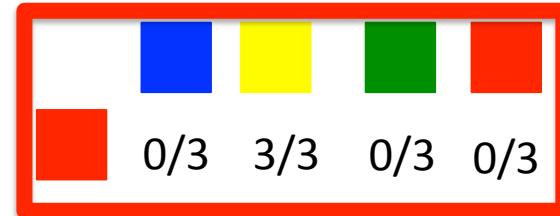
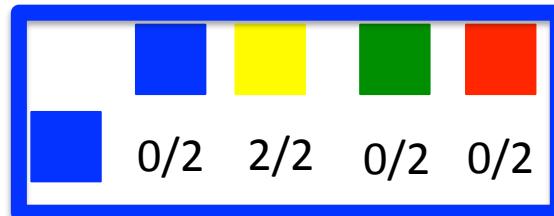
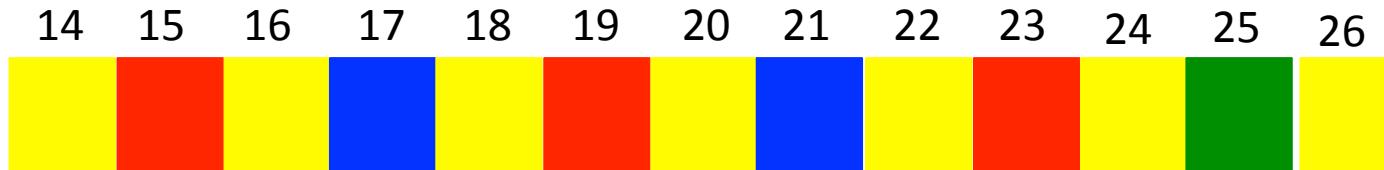
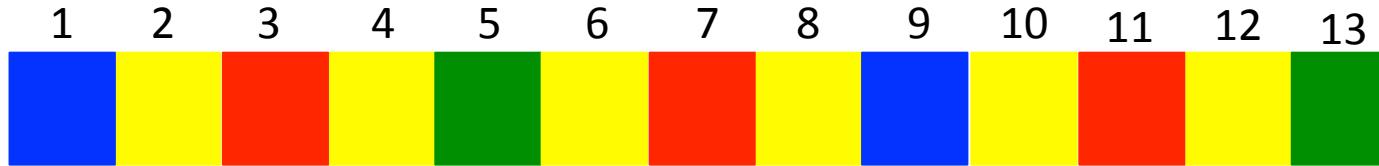
$P(\text{pixel} = \boxed{\text{green}} \mid \text{pixelul din stânga} = \boxed{\text{blue}}) = 0/2$

# Probabilitate condiționată



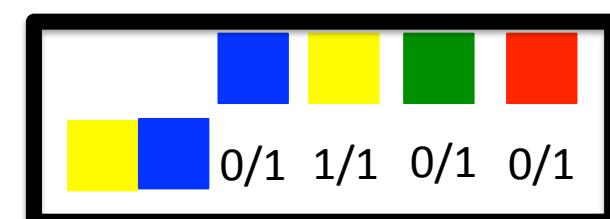
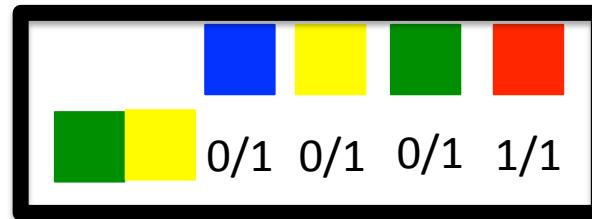
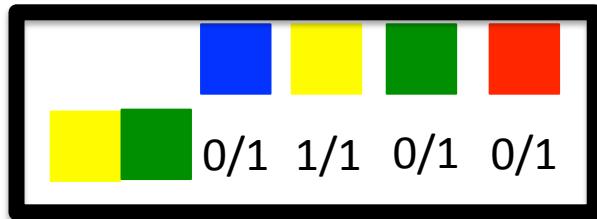
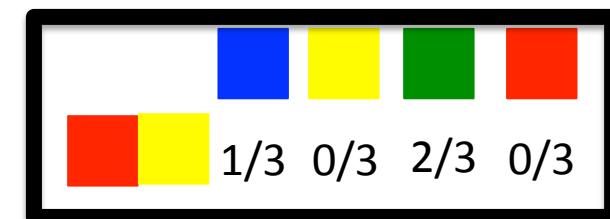
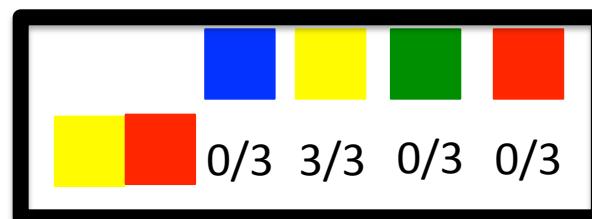
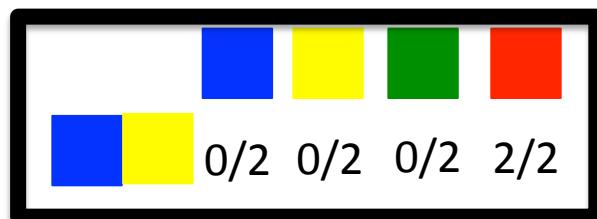
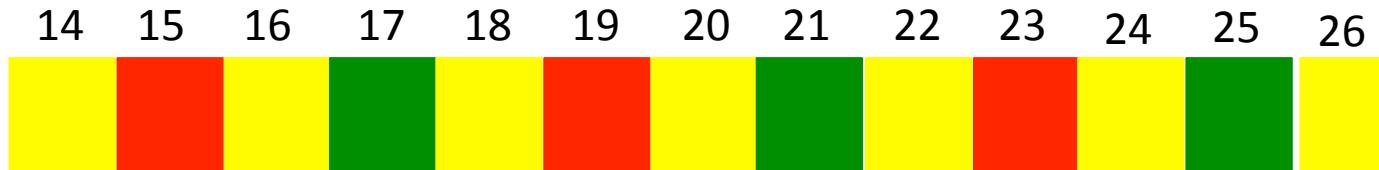
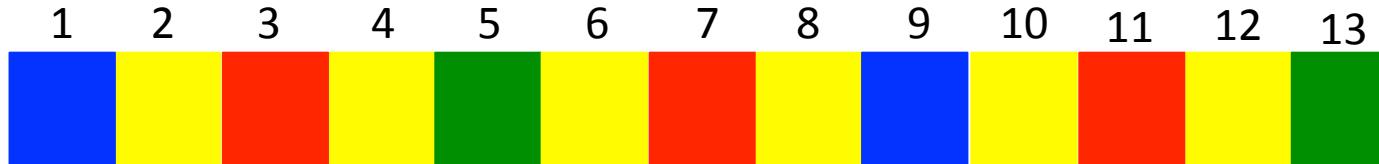
# Complețăm textura

Complețăm textura pixel cu pixel după următoarea regulă: observăm vecinul din stânga și apoi selectăm un pixel din distribuția condiționată (fără să o updatăm).



# Complețăm textura

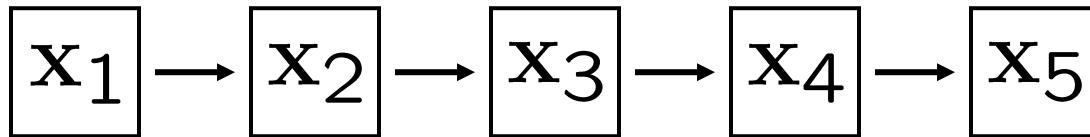
Complețăm textura pixel cu pixel după următoarea regulă: observăm cei doi vecini din stânga (conțează ordinea) și apoi selectăm un pixel din distribuția condiționată (fără să o updatăm).



# Lanțuri Markov

Lanț Markov:

- o secvență de variabile aleatoare  $x_1, x_2, \dots, x_n$
- $x_t$  reprezintă **starea** modelului la momentul t



- **Ipoteza de bază:** fiecare stare  $x_t$  depinde numai de starea precedentă  $x_{t-1}$

➤ dependența este dată de o **probabilitate condiționată**:

$$p(x_t | x_{t-1})$$

- Exemplul de mai sus: lanț Markov de ordinul 1
- Lanț Markov de ordinul N:

$$p(x_t | x_{t-1}, \dots, x_{t-N})$$

# Câmp Markov aleator

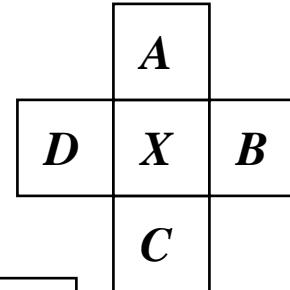
## Câmp Markov aleator

- Generalizare a lanțurilor Markov în 2 sau mai multe dimensiuni

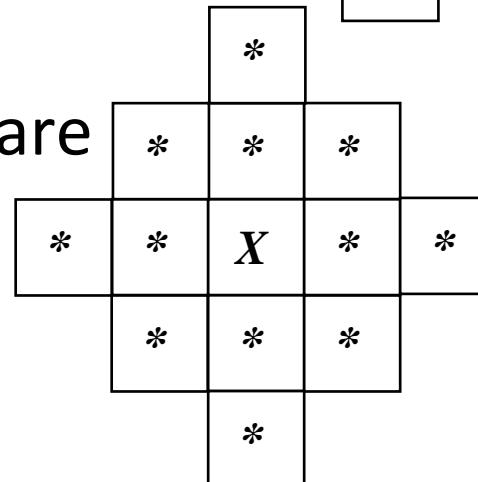
## Câmpuri Markov aleatoare de ordinul 1:

- probabilitatea ca un pixel  $X$  ia o anumită valoare depinde de valorile pixelilor vecini  $A, B, C$ , și  $D$ :

$$P(X|A, B, C, D)$$



## Câmpuri Markov aleatoare de ordin superior

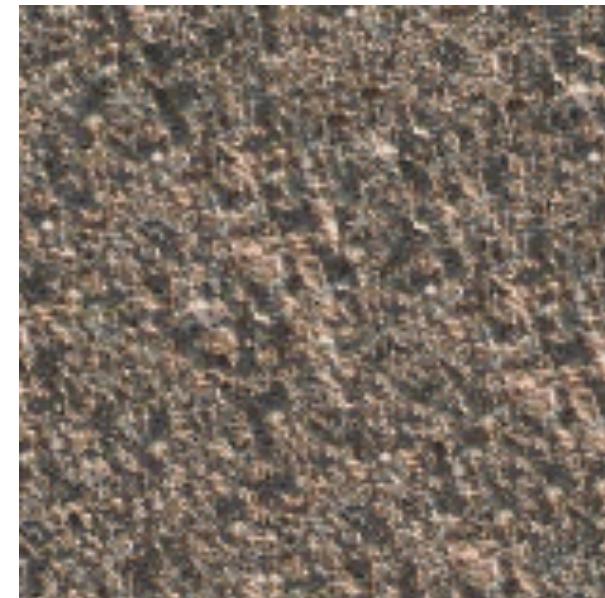
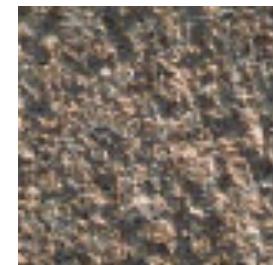


# Generarea texturii

Textură  
inițială

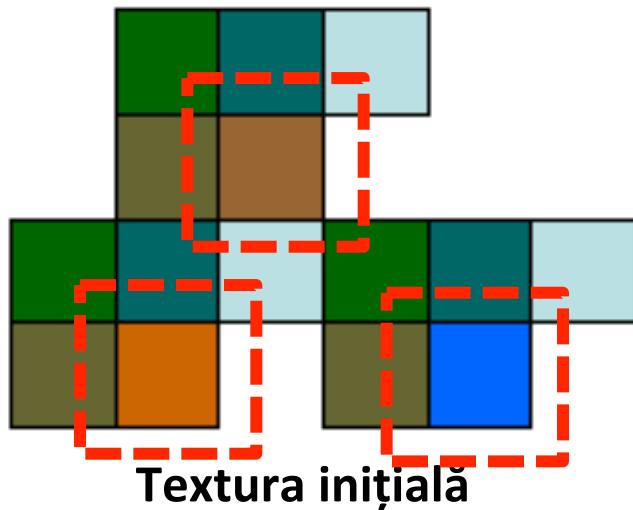


Textură  
generată



# Generarea texturii – intuiție

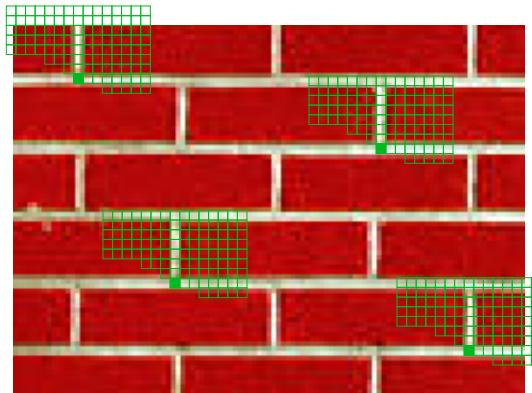
Inserăm pixeli cu o anumită intensitate pe baza valorilor pixelilor vecini



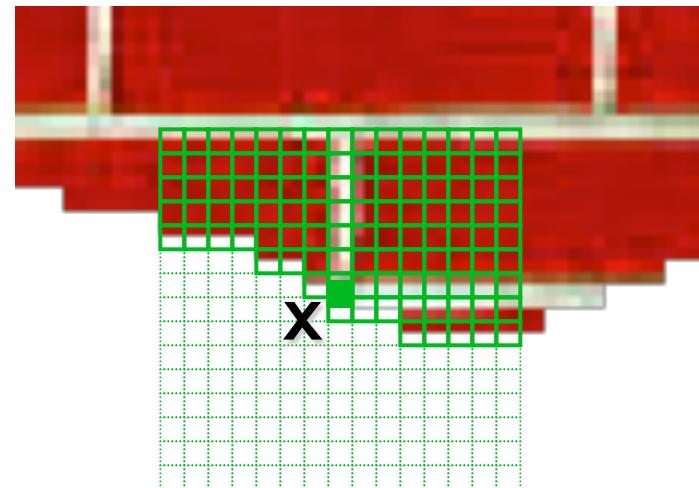
Vrem să inserăm un  
nou pixel aici

Distribuția intensității unui pixel este condiționată de vecinii lui (pentru acest exemplu).

# Generarea unui singur pixel



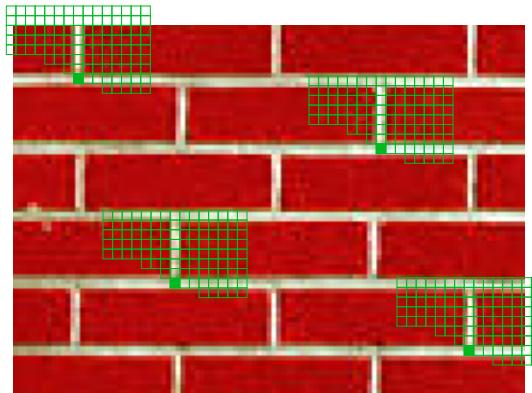
Textura inițială



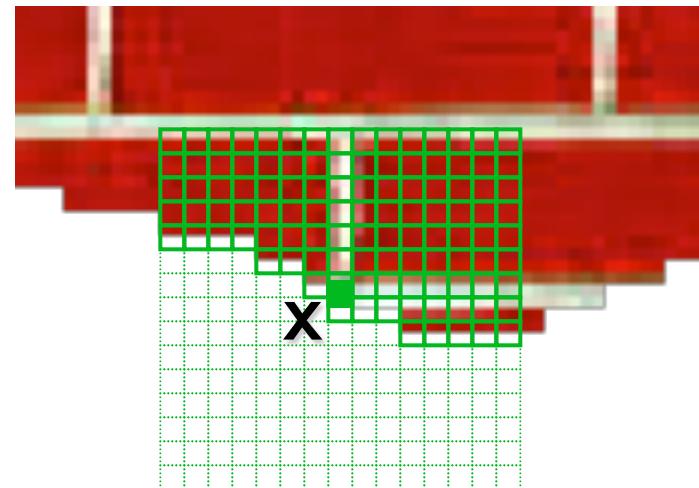
Imagine generată

- Care este  $P(x|\text{vecinătatea lui } x)$ ?
- Găsește toate ferestrele din textura inițială care seamănă perfect cu vecinătatea lui  $x$
- Pentru a genera  $x$ 
  - alege aleator o fereastă din cele găsite
  - valoarea lui  $x = \text{valoarea pixelului central al ferestrei}$

# Generarea unui singur pixel



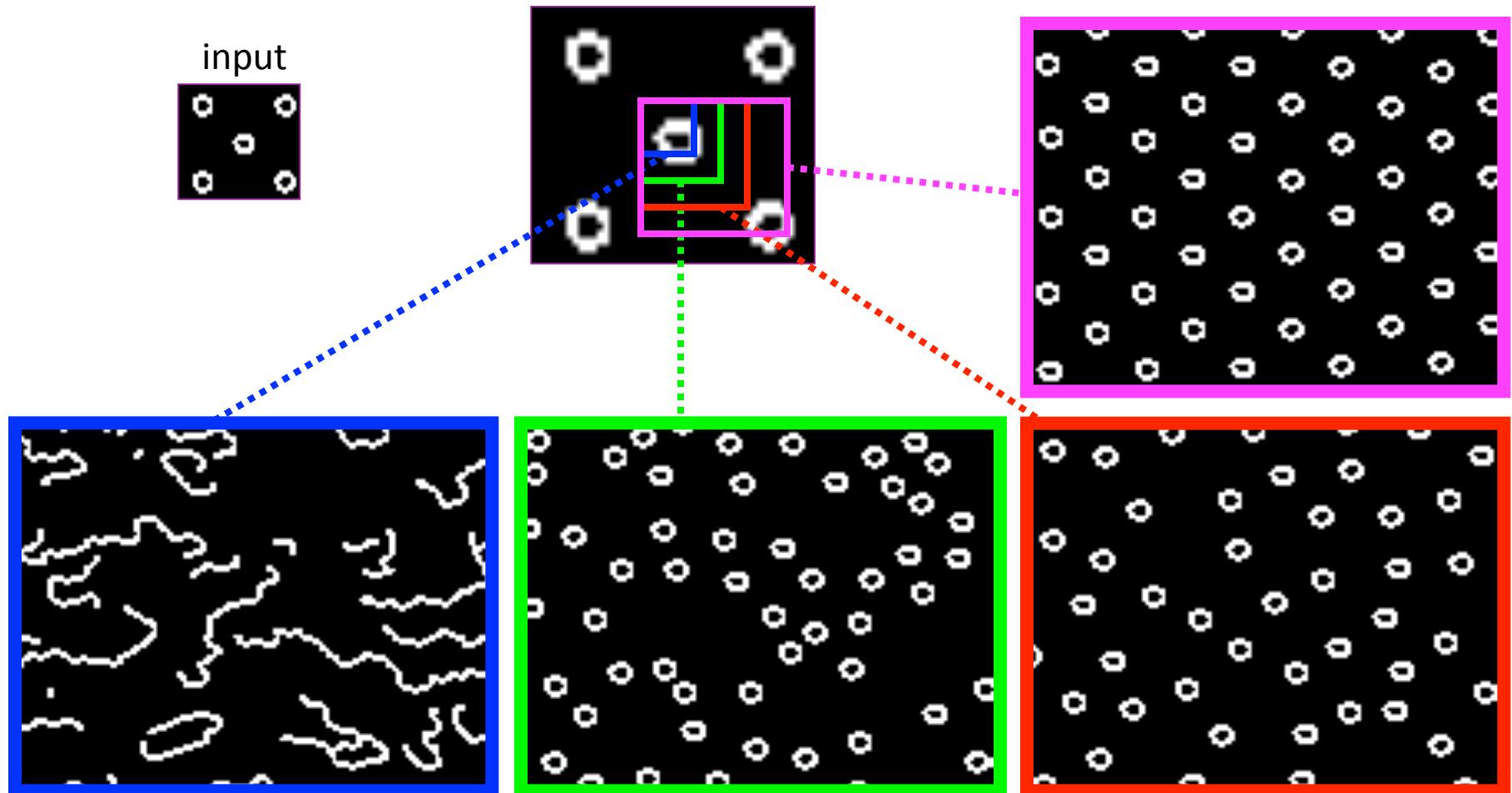
Textura inițială



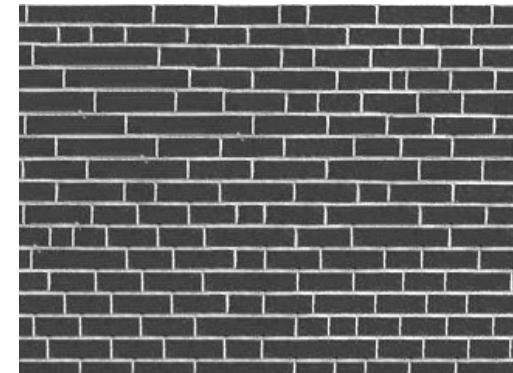
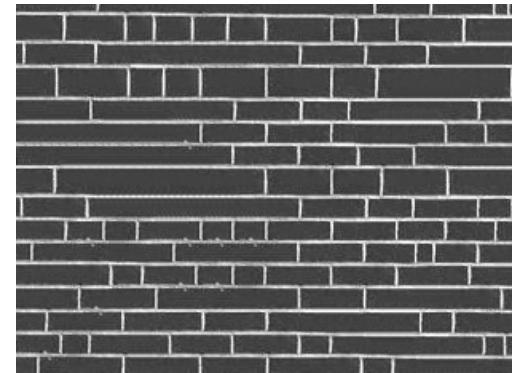
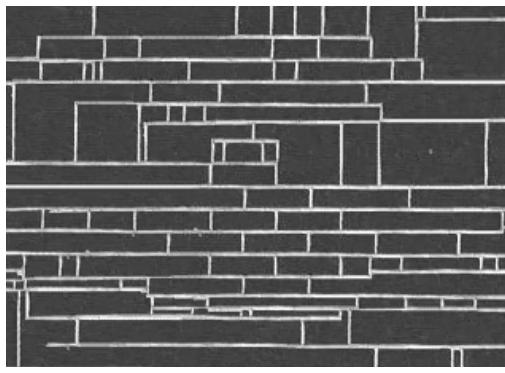
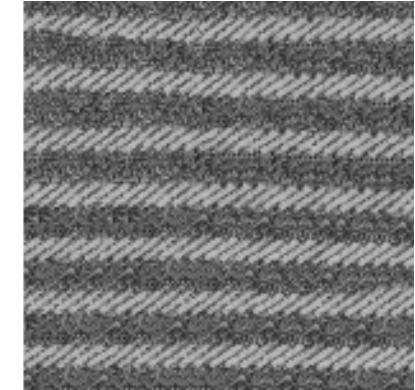
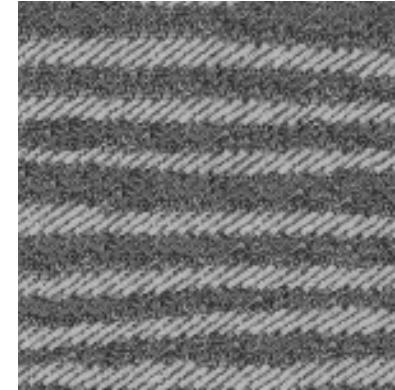
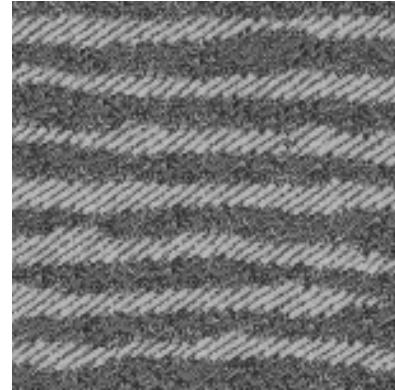
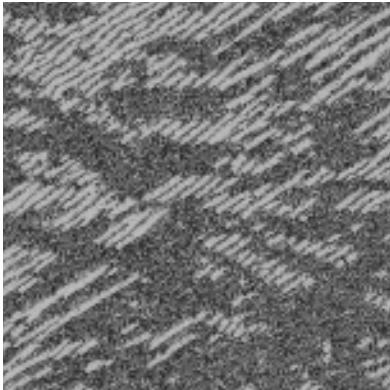
Imagine generată

- E posibil să nu existe nicio fereastră care seamănă perfect cu vecinătatea lui  $x$ .
- Găsește cele mai apropiate ferestre pe baza unei distanțe (suma pătratelor distanțelor) și alege una din aceste ferestre pe baza distanței
- Transformă distanța în probabilitate astfel încât: distanțe mari = prob. mică, distanțe mici = prob. mare

# Dimensiunea ferestrei vecinătății



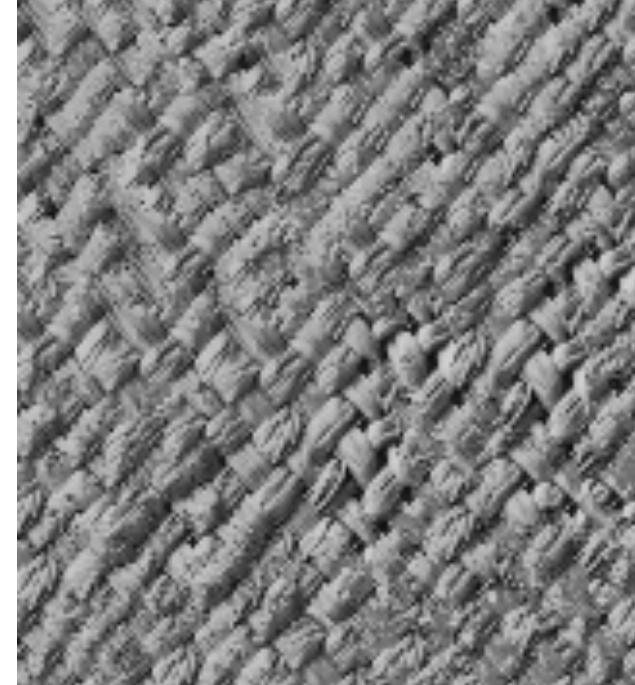
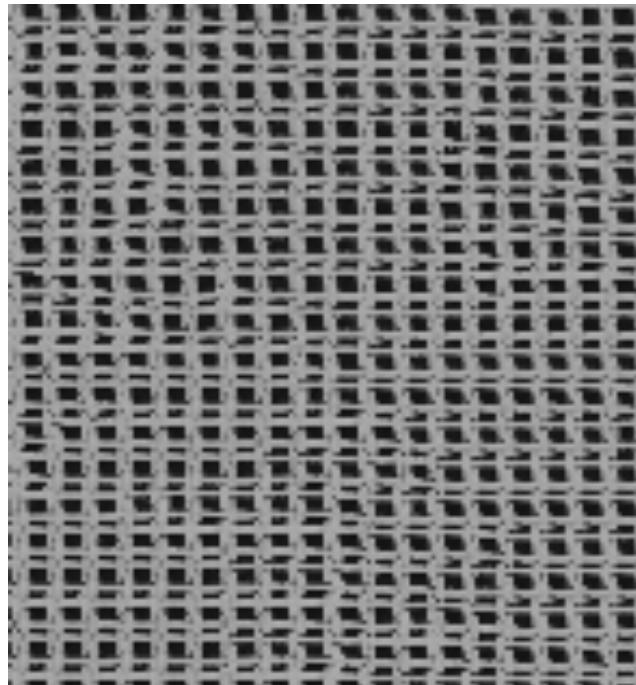
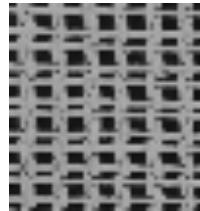
# Dimensiunea ferestrei vecinătății



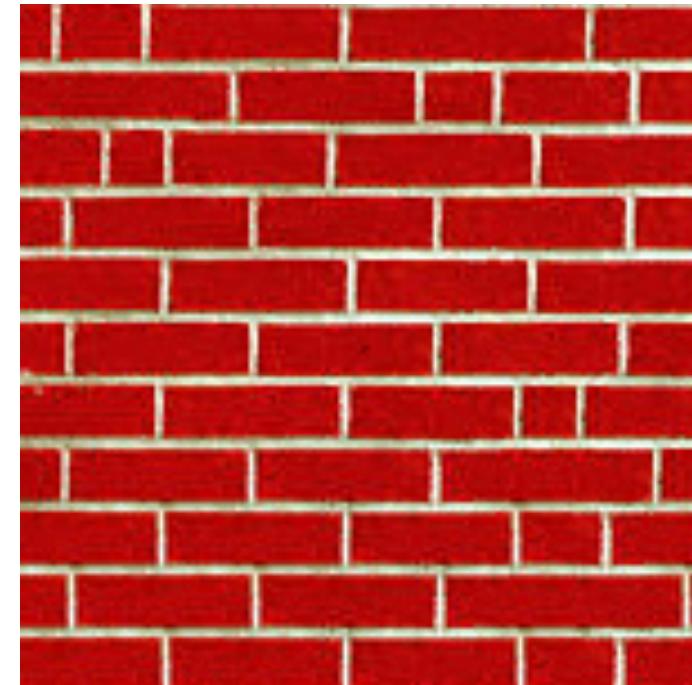
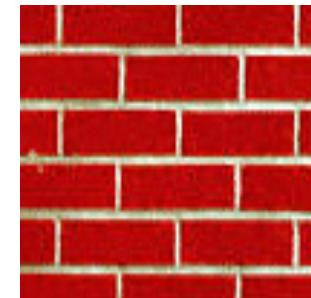
Creștem dimensiunea ferestrei



# Generarea texturii - rezultate



# Generarea texturii - rezultate

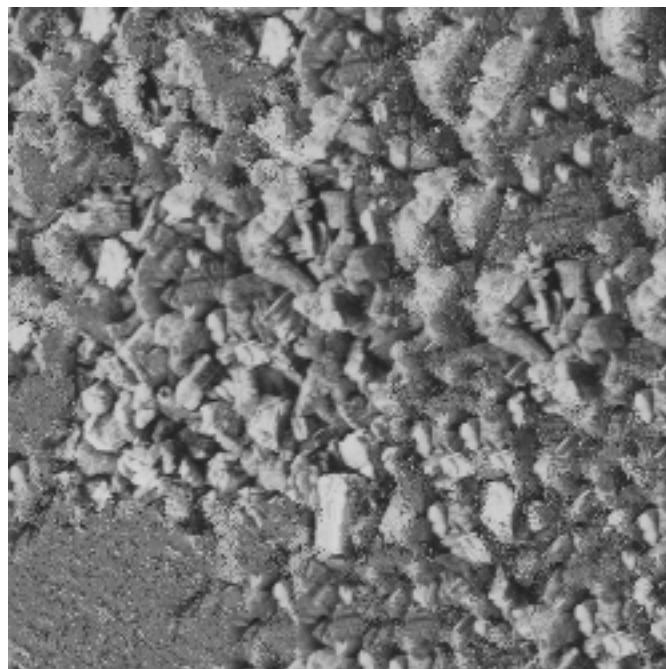
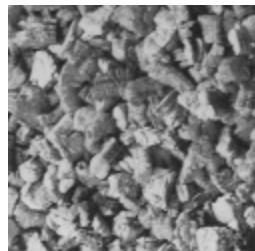


# Generarea texturii - rezultate

uring in the unsensato  
r Dick Gephardt was fai  
rful riff on the looming  
nly asked, "What's your  
tions?" A heartfelt sigh  
story about the emergenc  
es against Clinton. "Boy  
g people about continuin  
ardt began, patiently obs  
s, that the legal system h  
e with this latest tanger

ithairm, them . "Wmephartfe lartifelintomimen  
fel ck Clirtioout omaim thartfelins. f ait s anentc  
the ry onst wartfe lck Gephntoomimeationl sigak  
Chiooufit Clinut Cll riff on, hat's yordn, parut tly  
ons ycontonsteht wasked, paim t sahe loo' riff on l  
nskoneploourtfeas leil A nst Clit, "Wleontongal s  
k Cirtioouirtfepe óng pme abegal fartfenstemem  
tiensteneltorydt telemephminsverdt was agemer  
ff ons artientont Cling peme asurtfe atih, "Boui s  
hal s fartfelt sig pedr tl-dt ske abounutie aboutioo  
itfaonewwas yous aboonthardt thatins fain, ped,  
ains, them, pabout wasy arfuiu courtly d, ln A h  
ole emthriringbooreme agas fa bontinsyst Clinut  
ory about continst Clipeouinst Cloke agatiff out C  
stome zinemen tly ardt beoraboul n, thenly as t C  
cons faimeme Diontont wat coutlyohgans as fan  
ian, phrtfaul, "Wbaut cout congagal còmininga  
mifmst Cliiy abon al coountha.emungairt tf oun  
The looorystan loontieph. Intly on, theoplegatick  
aul tatiesontly atie Diontiomt wal s f thegæ ener  
mthahsæt's enenhhñmas fan. "intchthorw ahons v

# Cazuri nereușite

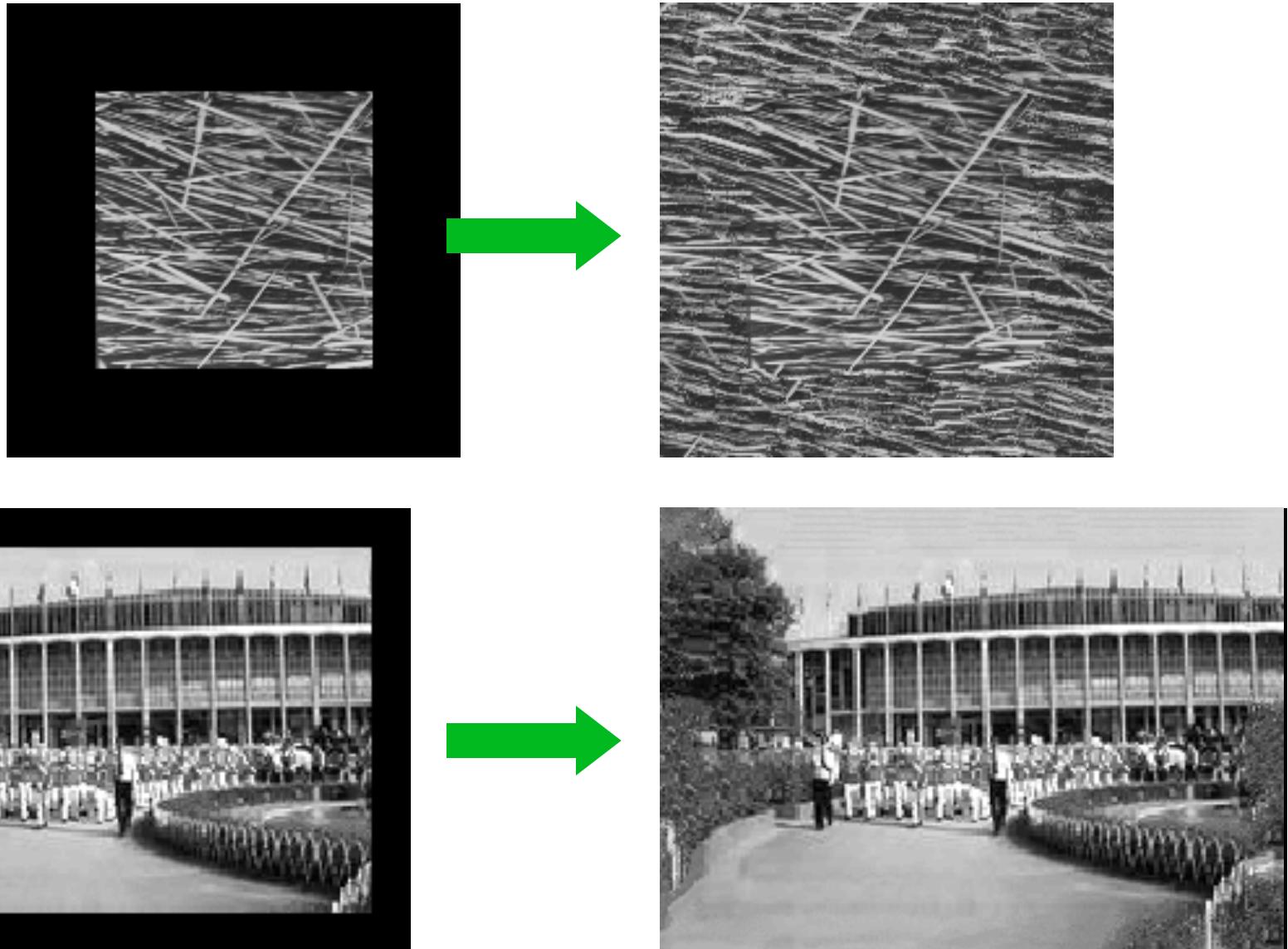


texteli care nu sunt  
în textura inițială

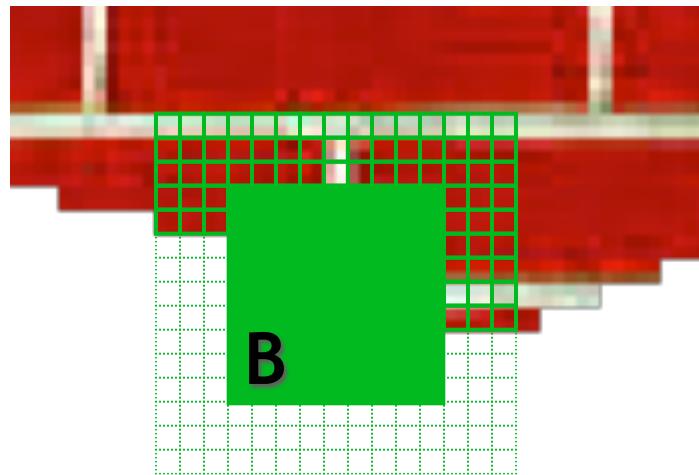


Copiază

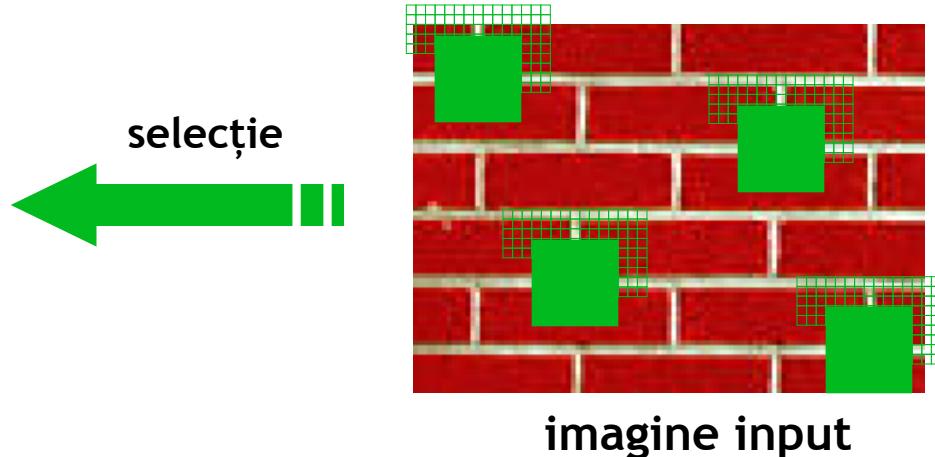
# Extrapolare



# Generarea texturii la nivel de blocuri



Sintetizăm un bloc

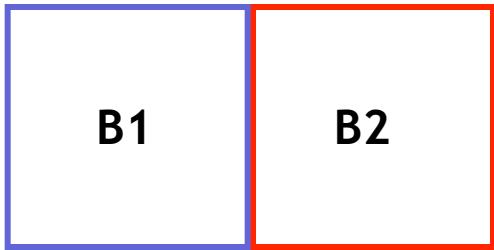
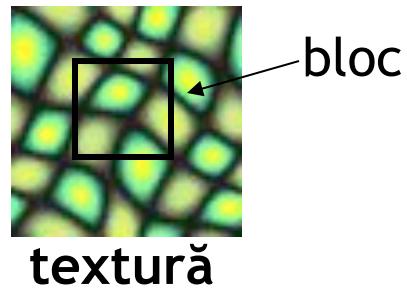


înțelegem

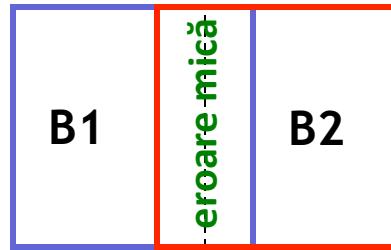
- Observație: pixelii vecini sunt foarte corelați

**Idee: unitate de sinteză = bloc de pixeli**

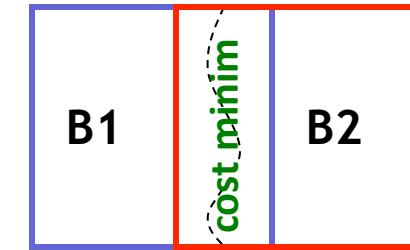
- Înlocuim un pixel cu un bloc :  $P(B | \text{Vecinătate}(B))$
- Mult mai rapid: sintetizăm toți pixelii dintr-un bloc odată



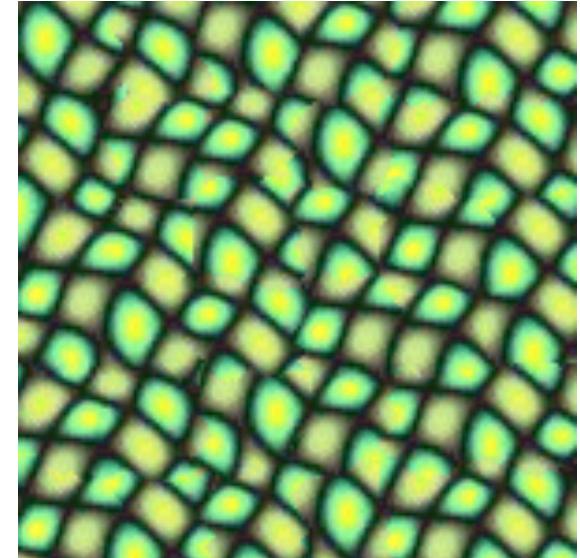
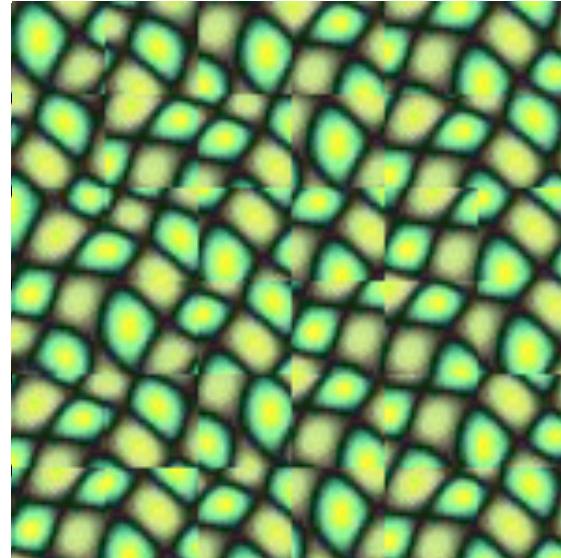
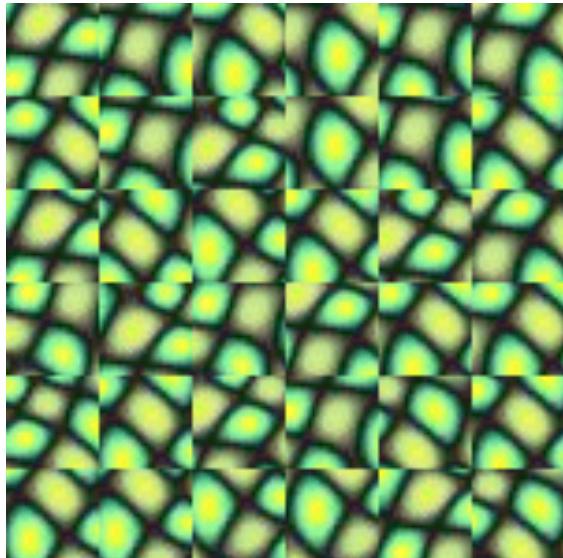
**blocuri aleatoare din textură  
pușe unele lângă altele**



**blocuri aleatoare din textură  
care se suprapun la frontieră**

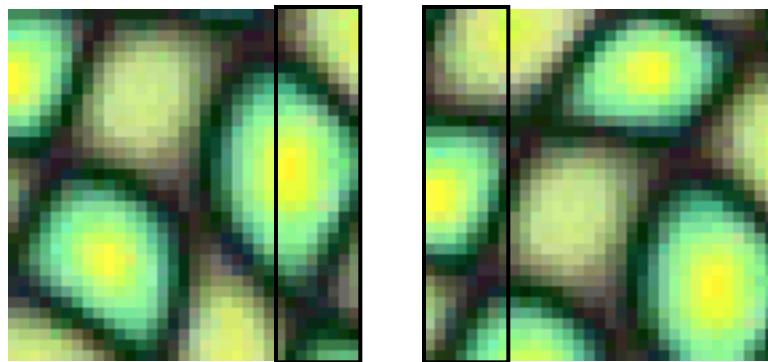


**blocuri aleatoare din textură  
care se suprapun la frontieră  
frontiera de cost minim**

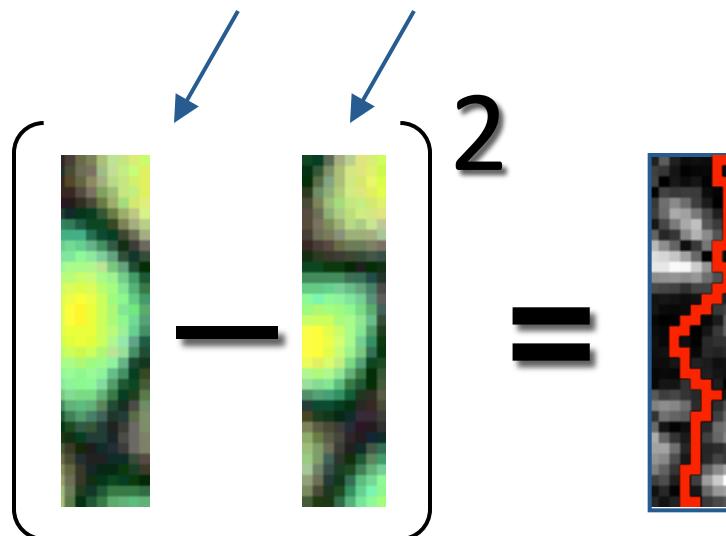
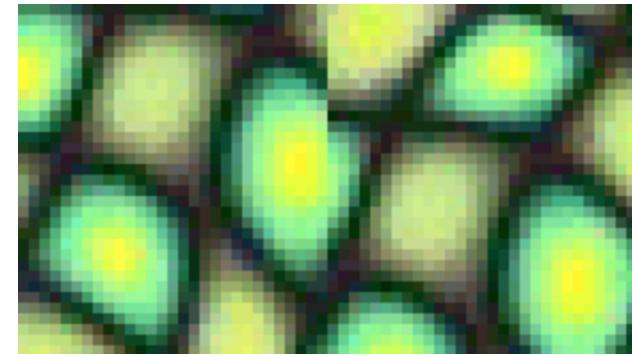


# Frontiera de eroare minimă

blocuri care se suprapun



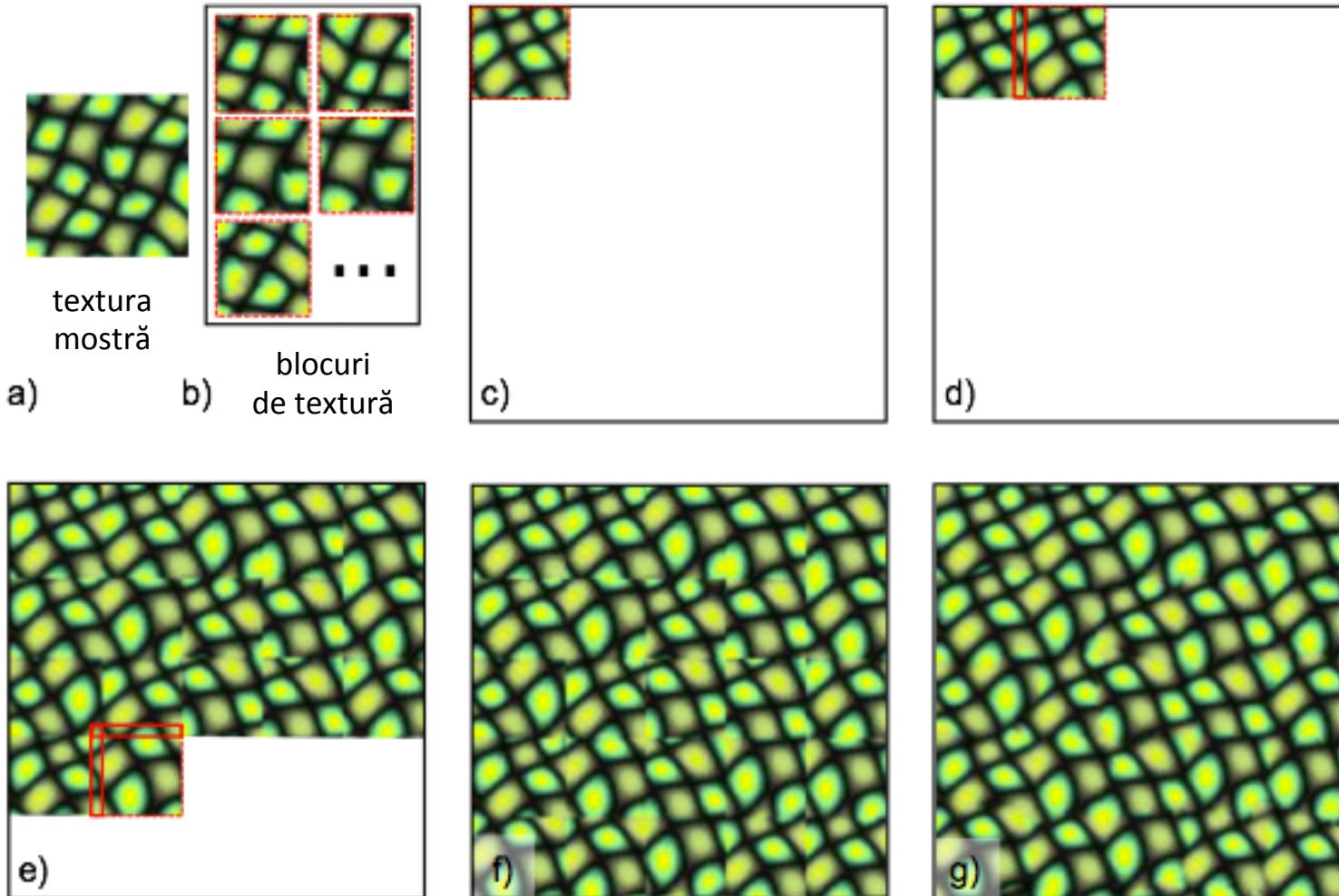
muchie/frontieră verticală

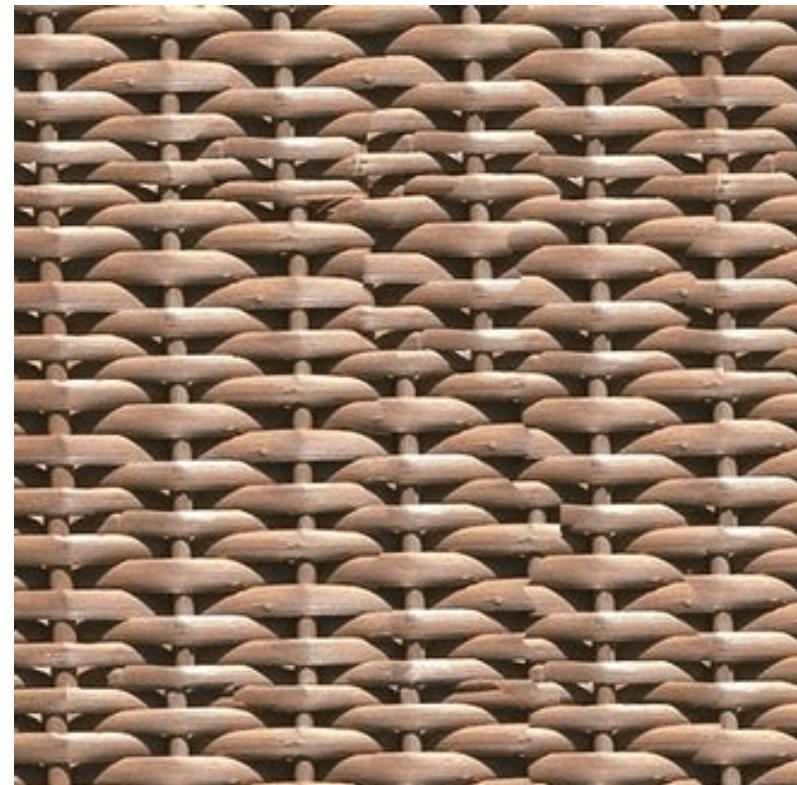
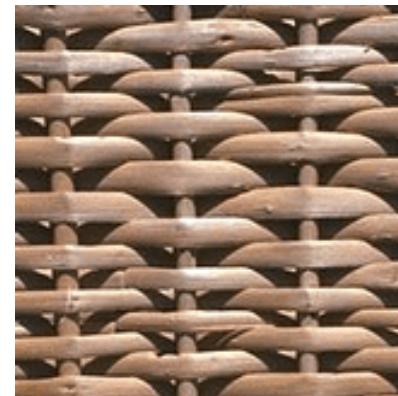


eroarea de suprapunere

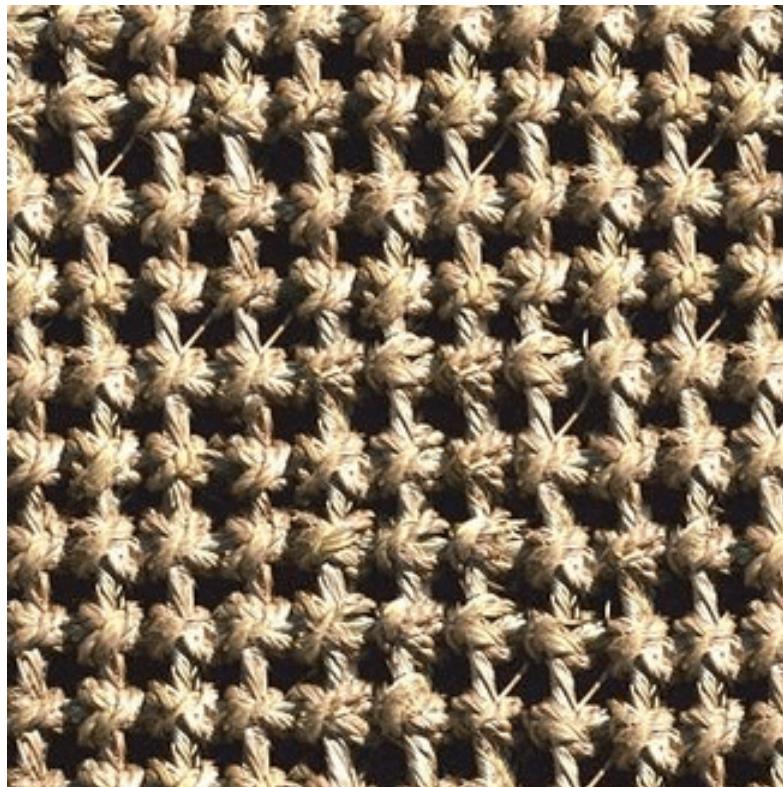
frontiera de eroare minimă

# Sinteza texturii la nivel de blocuri - algoritm

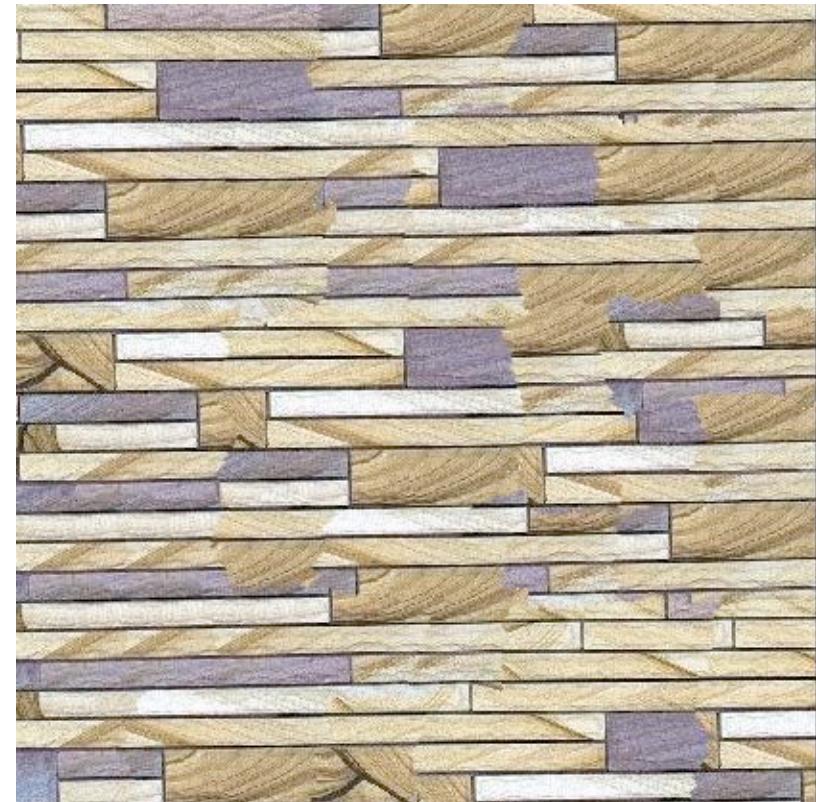




Slide adaptat după A. Efros



Slide adaptat după A. Efros



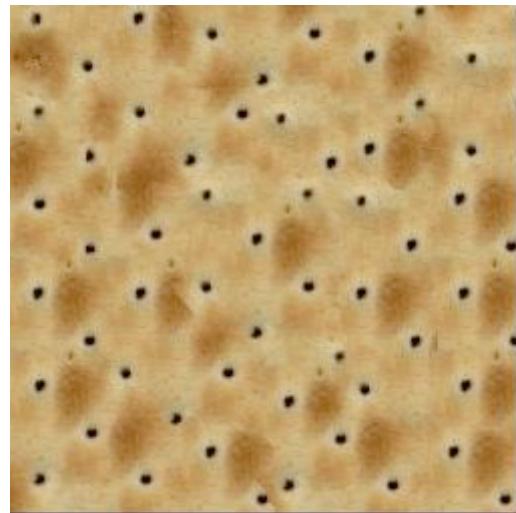
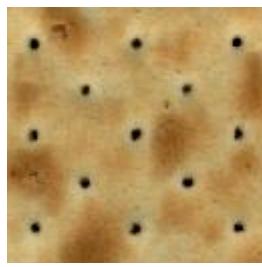
Slide adaptat după A. Efros



Slide adaptat după A. Efros



Slide adaptat după A. Efros



Slide adaptat după A. Efros

Mai puțin  
reușite



# Transferul texturii

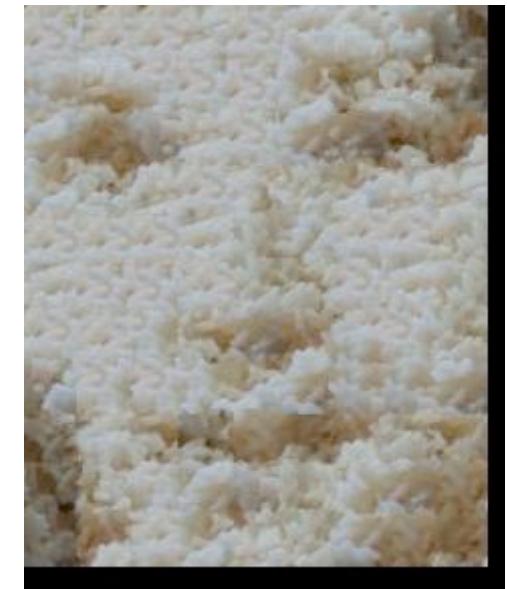


+

orez

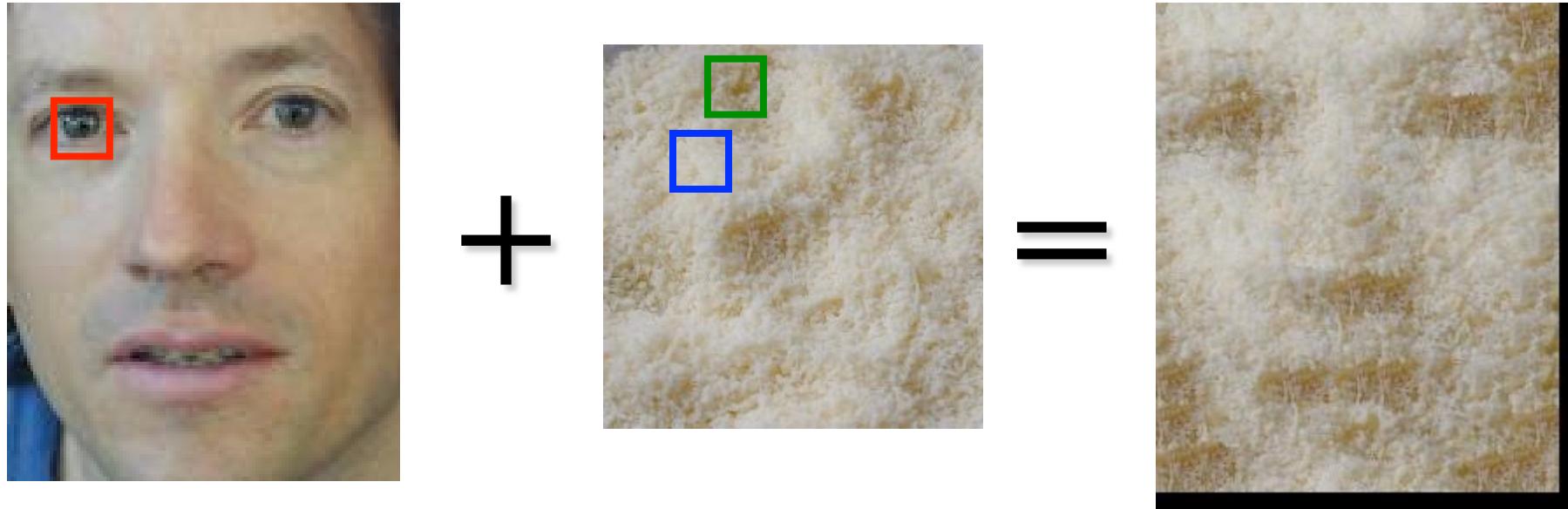


=



# Transferul texturii

- Transferăm textura unui obiect la un alt obiect



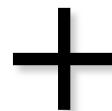
- Adăugăm un nou termen: similaritatea dintre blocul de textură și blocul din imagine
- Spre exemplu similaritate în termeni de intensitate

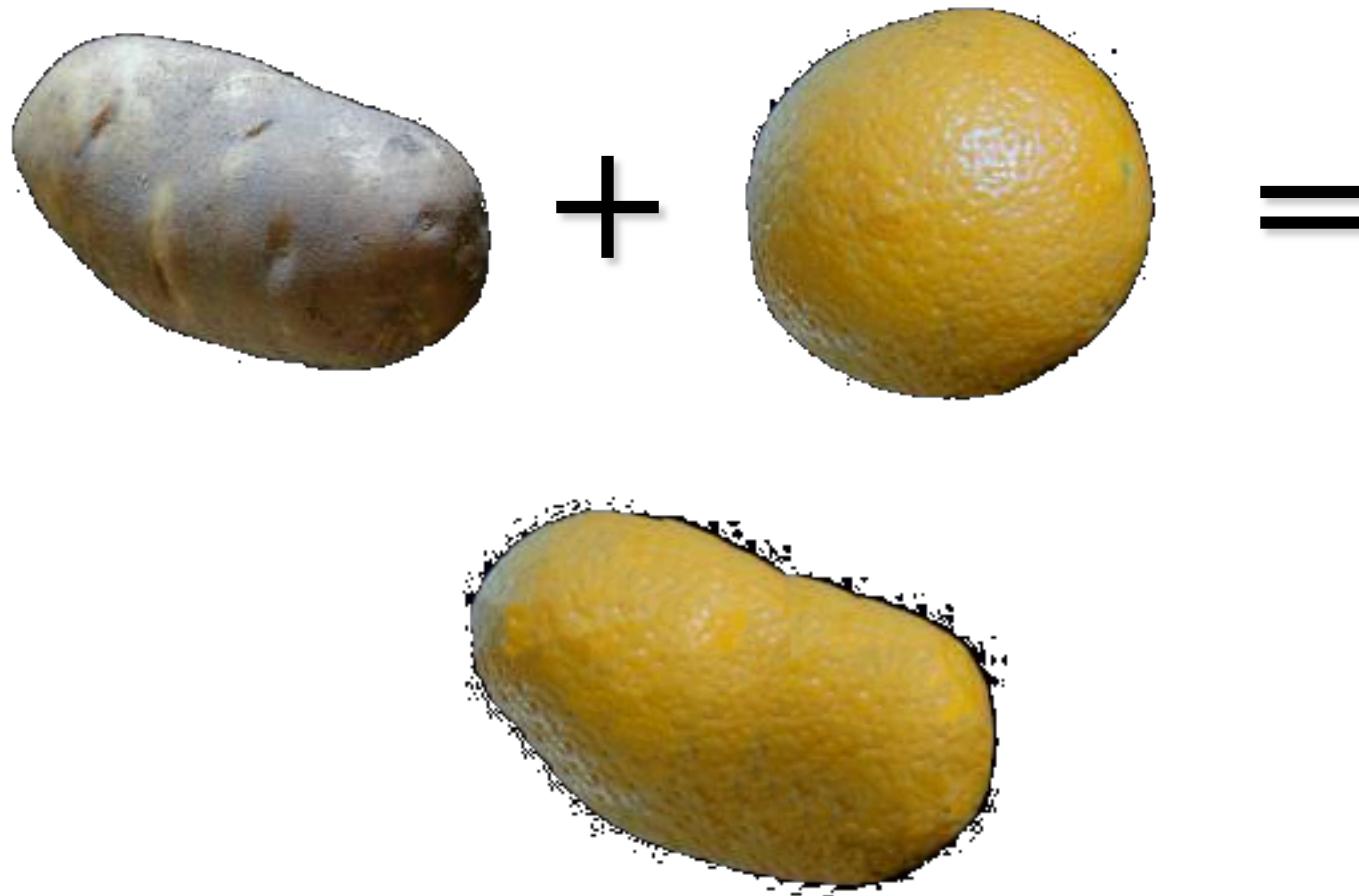


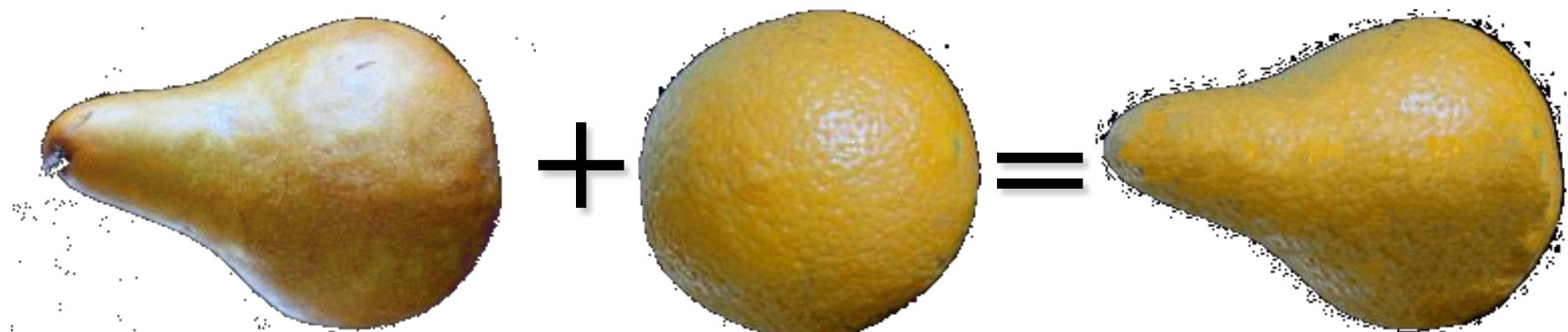
parmezan



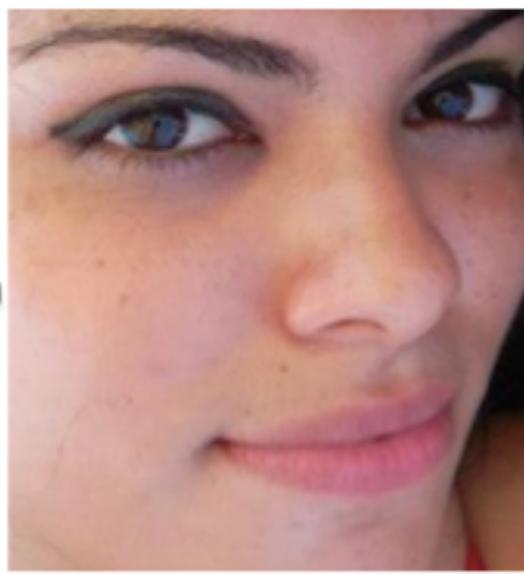
orez



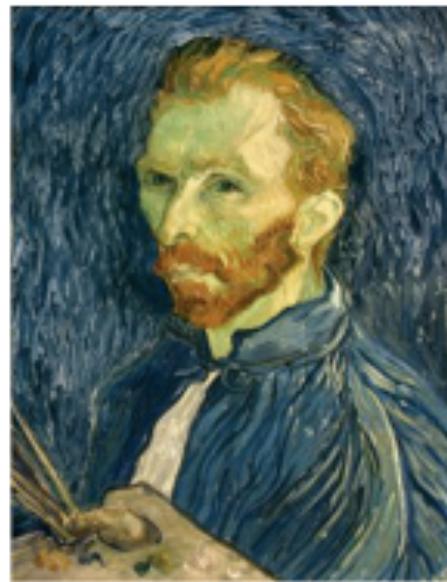




Slide adaptat după A. Efros

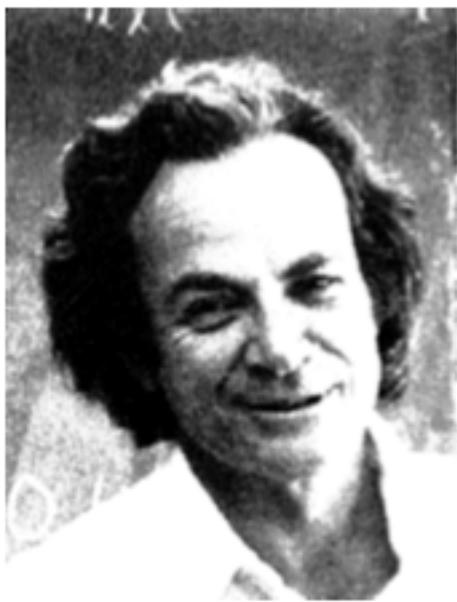


+



=





+



=



**DEMO**