

Concepte și aplicații în Vederea Artificială

Bogdan Alexe

bogdan.alexe@fmi.unibuc.ro

Cursul 13

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

						Note teme			
Nr.	Nume	IT	Prenume	Grupa	T1	T2	T3	T4	
1	AVRAM	S	ANDREI-ALEXANDRU	343		8	5.75	7.5	
2	BANU	M	ALEXANDRU	331	9.2	9.3		10	
3	BANU	T.C	ROBERT-EMANUEL	342	8	5.85			
4	BELCINEANU	V	ALEXANDRU-IOAN	344	9.5	10	10	9.5	
5	BOTEZATU	C	DANIEL-ANDREI	342	6	6.5	8		
6	CALINESCU	G	VALENTIN-GELU	331		8	9.25	10	
7	CALUIAN	V	IULIAN	343	10	9.5	9.75		
8	CAPATANA	N	DELIA-GABRIELA	343					
9	CONSTANTIN	A.P	TEODOR-CLAUDIU	331					
10	DRANCA	V	CONSTANTIN	334	5.95	9.4	9.5		
11	FERARU	D	ANDREI-IONUT	342	6	6.5			
12	FLORESCU	M.F	OANA-MIHAELA	334			5.75	7	
13	GIUMANCA	I	IOAN-DRAGOS-MARIAN	342	9.75	9.25		6	
14	IONESCU	A	TEODOR-STELIAN	331	9.3	9.25		10	
15	ISAIA	A	VLAD-LUCIAN	343	5.25	7.25			
16	JITCA	D	DAVID	334	9.8	9.6	10		
17	LEOPEA	S	CATALINA	343					
18	LUPASCU	I	MARIAN	331	10	10	10	10	
19	MARCOVSCHI	C.M	OCTAVIAN-MIHAI	341	8			10	
20	MARE	N	TUDOR-ALEXANDRU	344	7	9.5	8		
21	MISAILA	M.G	ALINA-CRISTINA	343					
22	MUNTEAN	C	RADU-ALEXANDRU	331		10	10	10	
23	NEDELCU	P	ANDREEA	344	10	9.25	10	10	
24	PANDELE	G.C	MARIA-SMARANDA	331		10		10	
25	POESINA	M	EDUARD-GABRIEL	334	10	10	9.5		
26	ROBU	V	COSTIN-STEFAN	332	9.5	10		10	
27	STANCU	P	ROBERT-GABRIEL	343	10	10	8		
28	TOMI	T	ANDRA-CORNELIA	344	5.2	8	7		
29	UTA	T.I	STEFANA-CRISTINA	344	5.5	9	8		
30	ZUGRAVU	B.G	ANDREI	331	10	10	10	10	
31	DUMITRESCU		GABRIEL HORIA	333	7		9.5		

Examen - evaluare

- În funcție de 2 aspecte:
 - teme de laborator (vor fi 5 de-a lungul semestrului);
 - lucrare finală de laborator (în săptămâna 14).
- puteți obține nota numai din teme dacă:
 - aveți cel puțin 3 teme peste nota 5;
 - nota finală = media celor mai mari 3 note (din 5 posibile) din teme.

SAU

- puteți obține nota numai din lucrarea finală de laborator (test pe calculator).

SAU

- puteți obține nota din media dintre lucrarea finală de laborator (trebuie să luați minim 5) și cea mai mare notă din temă (dacă vreti să o luăm în considerare)

Examen - evaluare

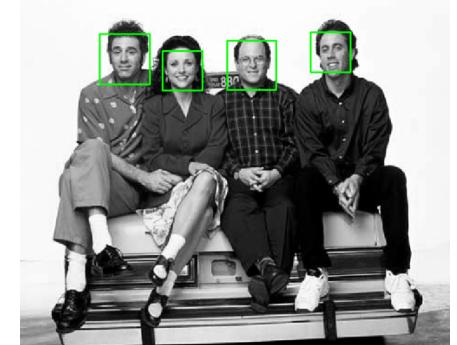
- Teme
 - tema 5 are termen limită vineri, 11 ianuarie;
 - laborator azi de la 14-16 în sala 201, finalizare tema 3;
 - sperăm să afișăm notele la tema 5 până luni 14 ianuarie;
- Lucrare de laborator
 - **miercuri, 16 ianuarie 2019, 10-12, sala 201** (studentii de la 343 au verificare la DAW)

Recapitulare – cursul trecut (2018)

- Localizarea claselor de obiecte
 - metoda ferestrei glisante

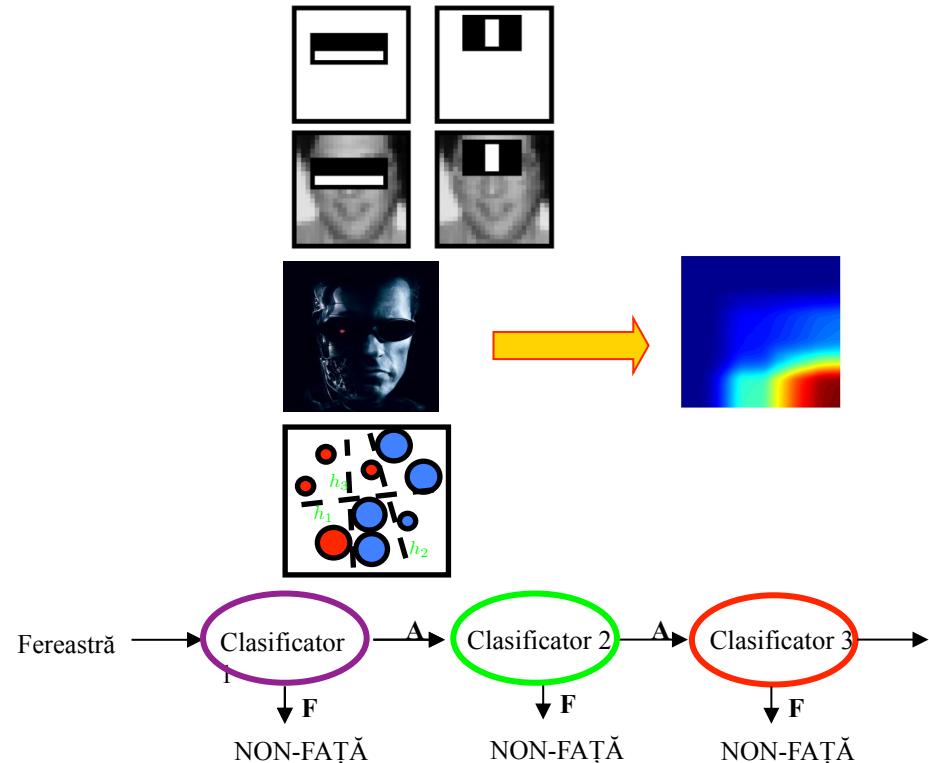


- Detectare facială folosind metoda glisării ferestrei și histograme de gradienți orientați (tema 5)

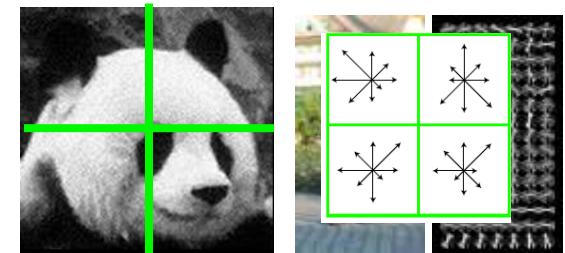


Cursul de azi

- Detectare facială folosind:
 - caracteristici Haar
 - imagini integrale
 - metoda AdaBoost
 - cascade de clasificatori



- Reprezentări bazate pe gradienți
 - histograme de gradienți orientați (HOG)
 - detectarea oamenilor pe bază de HOG

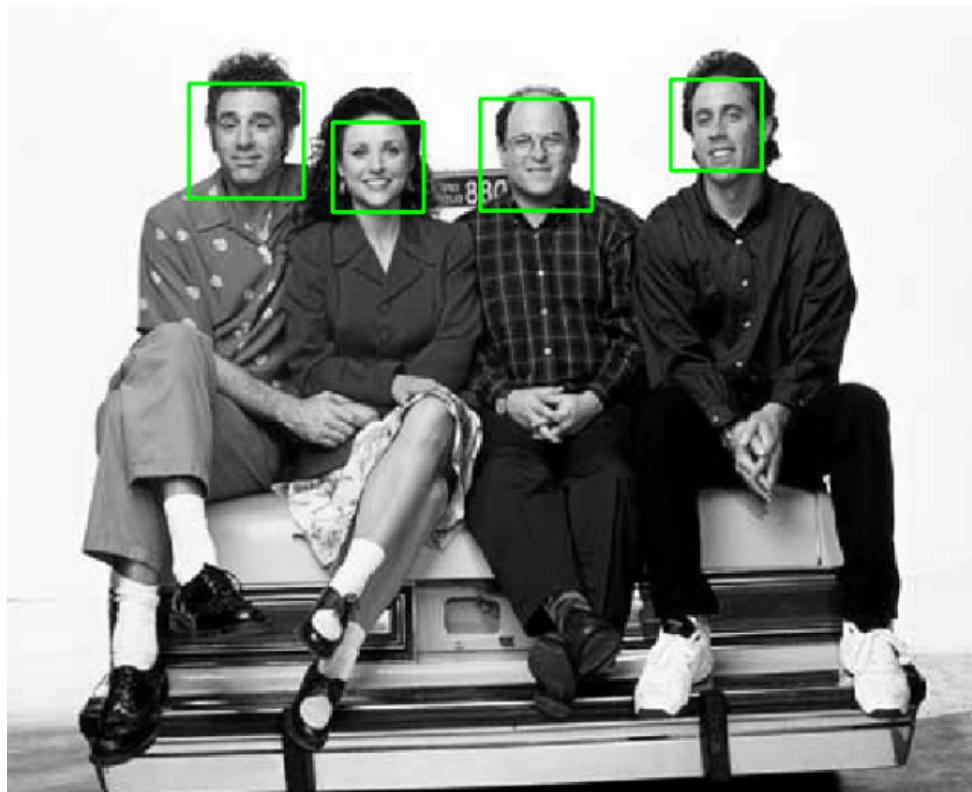


Abordări pentru detectarea facială

Există multe abordări de succes. Cele mai cunoscute:

1. detector bazat pe metoda glisării ferestrei și HOG (cursul trecut)
2. detectorul Viola-Jones (acest curs, implementat în Matlab)
3. detector bazat pe metoda valorilor proprii (eigen-faces)
4. detector bazat pe rețele neuronale de tip deep (la master)

Detectare facială folosind metoda AdaBoost și caracteristici Haar



Detectorul facial Viola-Jones

- detector facial real-time (15 frame-uri/sec) – 2001
- este implementat în MATLAB
- etapa de învățare decurge foarte încet, dar detectarea este foarte rapidă
- idei cheie:
 - folosirea de ***imagini integrale*** pentru evaluarea rapidă a caracteristicilor
 - selectarea caracteristicilor cu metoda '***boosting***'
 - ***cascade de clasificatori*** pentru eliminarea rapidă a ferestrelor ce nu conțin fețe

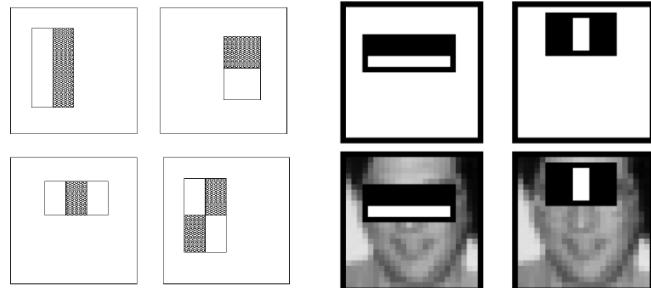
P. Viola and M. Jones. [Rapid object detection using a boosted cascade of simple features.](#) CVPR 2001.

P. Viola and M. Jones. [Robust real-time face detection.](#) IJCV 57(2), 2004.

Slide adaptat după S. Lazebnik

Extragere de caracteristici Haar

Filtre “dreptunghiulare” - caracteristici Haar



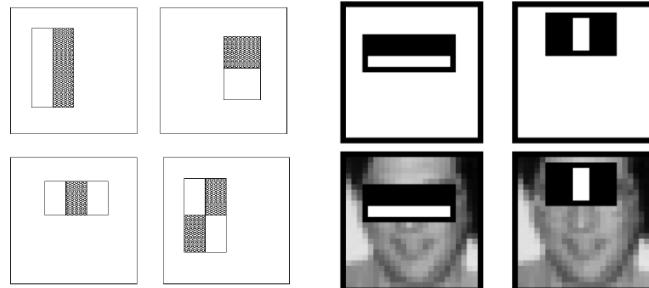
Caracteristici măsurate = diferență în intensitate dintre regiuni adiacente
(alb = +1, negru = -1)

$$\text{valoare} = \sum (\text{pixeli în regiunea albă}) - \sum (\text{pixeli în regiunea neagră})$$



Extragere de caracteristici Haar

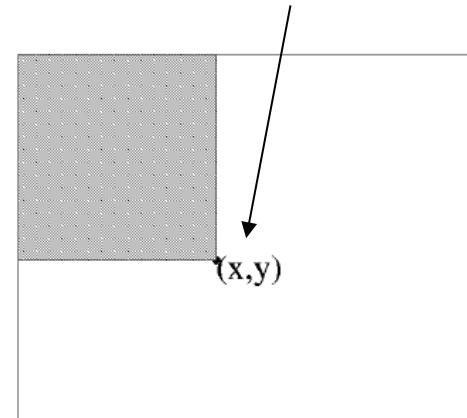
Filtre “dreptunghiulare” - caracteristici Haar



Caracteristici măsurate = diferență în intensitate dintre regiuni adiacente
(alb = +1, negru = -1)

Calcul eficient cu **imagini integrale**: suma intensităților din orice dreptunghi poate fi calculată în timp constant

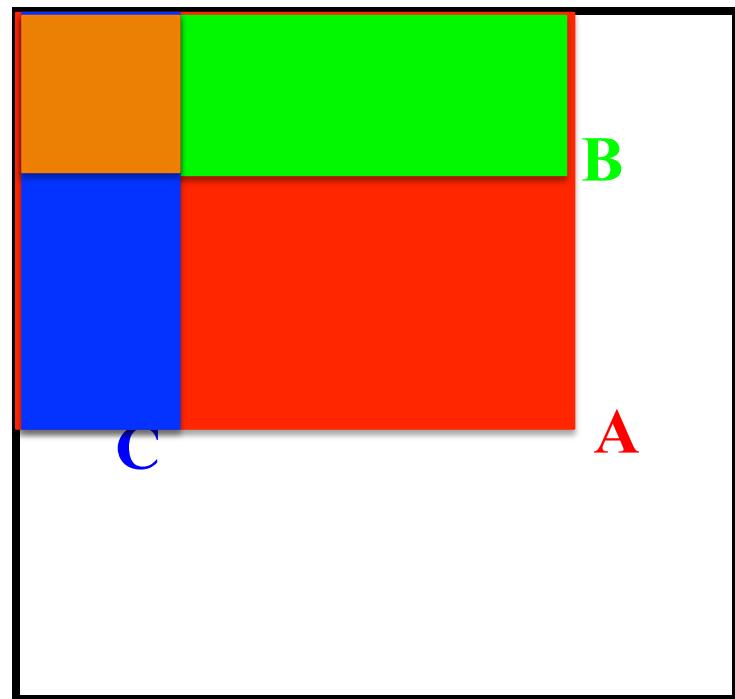
Valoarea la (x,y) este suma tuturor intensităților pixelilor deasupra și la stânga lui (x,y)



Imagine integrală

Extragere de caracteristici Haar

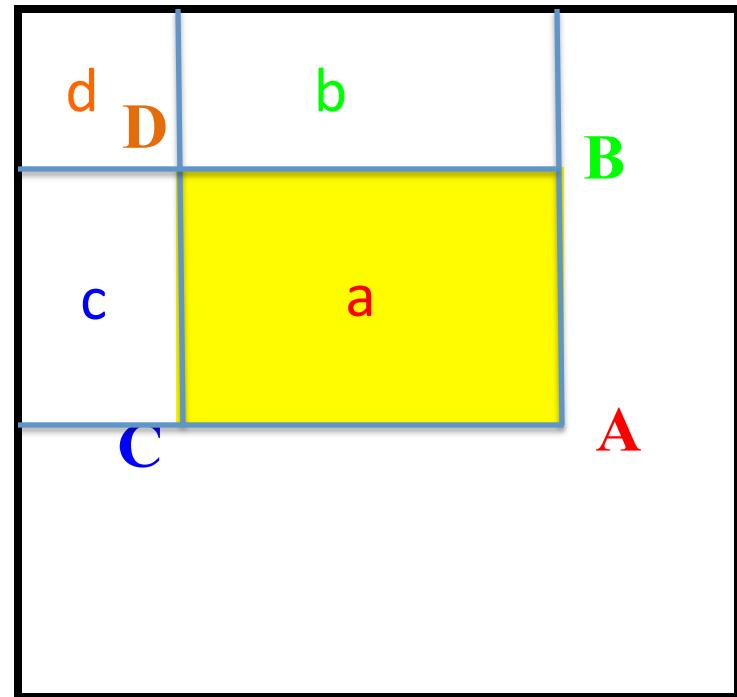
- Fie A,B,C si D valorile imaginii integrale pentru punctele din colțul dreptunghiului
- Suma intensităților din dreptunghiul galben poate fi calculată foarte simplu astfel:
$$\text{suma} = \textcolor{red}{A} - \textcolor{green}{B} - \textcolor{blue}{C} + \textcolor{orange}{D}$$



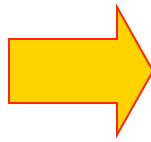
Imagine integrală

Extragere de caracteristici Haar

- Fie A,B,C si D valorile imaginii integrale pentru punctele din colțul dreptunghiului
- Suma intensităților din dreptunghiul galben poate fi calculată foarte simplu astfel:
$$\begin{aligned} \text{suma} &= A - B - C + D \\ &= (a + b + c + d) - (b + d) - (c + d) + d = a \end{aligned}$$
- Numai 3 operații (o adunare + 2 scăderi) necesare pentru a calcula suma intensităților din fiecare dreptunghi!

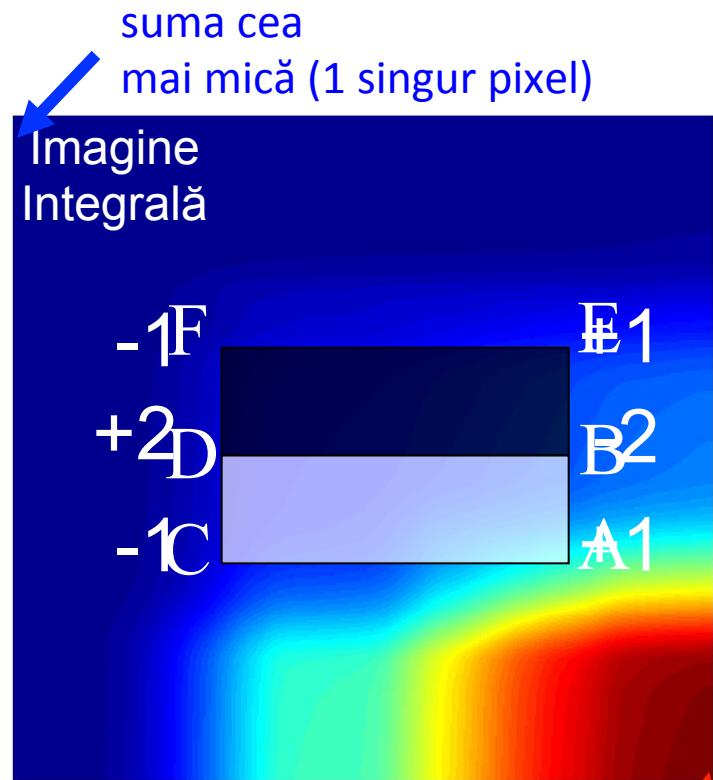
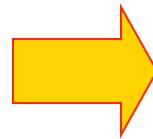


Exemplu



rgb2gray

Exemplu



valoare = $\sum (\text{pixeli în regiunea albă}) - \sum (\text{pixeli în regiunea neagră})$

$$(A - B - C + D) - (B - E - D + F) =$$

$$A - 2*B - C + 2*D + E - F$$

suma cea mai mare (toți pixelii)

Imagine integrală

34	11	33	3	19	19	18	18	2	24
34	8	36	11	5	11	5	6	15	33
17	22	17	4	6	3	5	7	35	18
8	3	15	22	5	1	20	10	12	22
8	7	1	22	19	29	6	20	9	27
16	7	11	17	15	2	25	19	29	10
34	26	29	31	5	6	30	17	4	10
33	28	30	4	28	21	26	5	32	21
1	18	13	5	27	16	28	19	32	23
12	13	16	23	13	7	21	5	2	15

Imagine inițială

34

Imagine integrală

-1	1
1	1

Filtru

Ponderi în Imaginea integrală

Pondere în Imaginea inițială

Imagine integrală

34	11	33	3	19	19	18	18	2	24
34	8	36	11	5	11	5	6	15	33
17	22	17	4	6	3	5	7	35	18
8	3	15	22	5	1	20	10	12	22
8	7	1	22	19	29	6	20	9	27
16	7	11	17	15	2	25	19	29	10
34	26	29	31	5	6	30	17	4	10
33	28	30	4	28	21	26	5	32	21
1	18	13	5	27	16	28	19	32	23
12	13	16	23	13	7	21	5	2	15

Imagine inițială

34 45

Imagine integrală

-1	1
1	1

Filtru

Ponderi în Imaginea integrală

Pondere în Imaginea inițială

Imagine integrală

34	11	33	3	19	19	18	18	2	24
34	8	36	11	5	11	5	6	15	33
17	22	17	4	6	3	5	7	35	18
8	3	15	22	5	1	20	10	12	22
8	7	1	22	19	29	6	20	9	27
16	7	11	17	15	2	25	19	29	10
34	26	29	31	5	6	30	17	4	10
33	28	30	4	28	21	26	5	32	21
1	18	13	5	27	16	28	19	32	23
12	13	16	23	13	7	21	5	2	15

Imagine inițială

34 45 78

-1	1
1	1

Filtru

Ponderi în Imaginea integrală

Pondere în Imaginea inițială

Imagine integrală

Imagine integrală

34	11	33	3	19	19	18	18	2	24
34	8	36	11	5	11	5	6	15	33
17	22	17	4	6	3	5	7	35	18
8	3	15	22	5	1	20	10	12	22
8	7	1	22	19	29	6	20	9	27
16	7	11	17	15	2	25	19	29	10
34	26	29	31	5	6	30	17	4	10
33	28	30	4	28	21	26	5	32	21
1	18	13	5	27	16	28	19	32	23
12	13	16	23	13	7	21	5	2	15

Imagine inițială

34	45	78	81	100	119	137	155	157	181
----	----	----	----	-----	-----	-----	-----	-----	-----

Imagine integrală

-1	1
1	1

Filtru

Ponderi în Imaginea integrală

Pondere în Imaginea inițială

Imagine integrală

34	11	33	3	19	19	18	18	2	24
34	8	36	11	5	11	5	6	15	33
17	22	17	4	6	3	5	7	35	18
8	3	15	22	5	1	20	10	12	22
8	7	1	22	19	29	6	20	9	27
16	7	11	17	15	2	25	19	29	10
34	26	29	31	5	6	30	17	4	10
33	28	30	4	28	21	26	5	32	21
1	18	13	5	27	16	28	19	32	23
12	13	16	23	13	7	21	5	2	15

Imagine inițială

34	45	78	81	100	119	137	155	157	181
68	87								

Imagine integrală

-1	1
1	1

Filtru

Ponderi în Imaginea integrală

Pondere în Imaginea inițială

Imagine integrală

34	11	33	3	19	19	18	18	2	24
34	8	36	11	5	11	5	6	15	33
17	22	17	4	6	3	5	7	35	18
8	3	15	22	5	1	20	10	12	22
8	7	1	22	19	29	6	20	9	27
16	7	11	17	15	2	25	19	29	10
34	26	29	31	5	6	30	17	4	10
33	28	30	4	28	21	26	5	32	21
1	18	13	5	27	16	28	19	32	23
12	13	16	23	13	7	21	5	2	15

Imagine inițială

34	45	78	81	100	119	137	155	157	181
68	87	156							

Imagine integrală

-1	1
1	1

Filtru

Ponderi în Imaginea integrală

Pondere în Imaginea inițială

Imagine integrală

34	11	33	3	19	19	18	18	2	24
34	8	36	11	5	11	5	6	15	33
17	22	17	4	6	3	5	7	35	18
8	3	15	22	5	1	20	10	12	22
8	7	1	22	19	29	6	20	9	27
16	7	11	17	15	2	25	19	29	10
34	26	29	31	5	6	30	17	4	10
33	28	30	4	28	21	26	5	32	21
1	18	13	5	27	16	28	19	32	23
12	13	16	23	13	7	21	5	2	15

Imagine inițială

34	45	78	81	100	119	137	155	157	181
68	87	156	170	194	224	247	271	288	345

Imagine integrală



Filtru

Ponderi în Imaginea integrală

Pondere în Imaginea inițială

Imagine integrală

34	11	33	3	19	19	18	18	2	24
34	8	36	11	5	11	5	6	15	33
17	22	17	4	6	3	5	7	35	18
8	3	15	22	5	1	20	10	12	22
8	7	1	22	19	29	6	20	9	27
16	7	11	17	15	2	25	19	29	10
34	26	29	31	5	6	30	17	4	10
33	28	30	4	28	21	26	5	32	21
1	18	13	5	27	16	28	19	32	23
12	13	16	23	13	7	21	5	2	15

Imagine inițială

34	45	78	81	100	119	137	155	157	181
68	87	156	170	194	224	247	271	288	345
85	126	212	230	260	293	321	352	404	479
93	137	238	278	313	347	395	436	500	597
101	152	254	316	370	433	487	548	621	745
117	175	288	367	436	501	580	660	762	896
151	235	377	487	561	632	741	838	944	1088
184	296	468	582	684	776	911	1013	1151	1316
185	315	500	619	748	856	1019	1140	1310	1498
197	340	541	683	825	940	1124	1250	1422	1625

Imagine integrală

Filtru

Ponderi în Imaginea integrală

Pondere în Imaginea inițială

O(N) operații!
~300 în acest caz

Imagine integrală

34	11	33	3	19	19	18	18	2	24
34	8	36	11	5	11	5	6	15	33
17	22	17	4	6	3	5	7	35	18
8	3	15	22	5	1	20	10	12	22
8	7	1	22	19	29	6	20	9	27
16	7	11	17	15	2	25	19	29	10
34	26	29	31	5	6	30	17	4	10
33	28	30	4	28	21	26	5	32	21
1	18	13	5	27	16	28	19	32	23
12	13	16	23	13	7	21	5	2	15

Imagine inițială

	Sum	Cost	Total Cost
$\Sigma\Sigma$	69	5	5
II	69	3	3

34	45	78	81	100	119	137	155	157	181
68	87	156	170	194	224	247	271	288	345
85	126	212	230	260	293	321	352	404	479
93	137	238	278	313	347	395	436	500	597
101	152	254	316	370	433	487	548	621	745
117	175	288	367	436	501	580	660	762	896
151	235	377	487	561	632	741	838	944	1088
184	296	468	582	684	776	911	1013	1151	1316
185	315	500	619	748	856	1019	1140	1310	1498
197	340	541	683	825	940	1124	1250	1422	1625

$$\begin{aligned} \Sigma\Sigma: & 17+4+6+15+22+5= 69 \\ II: & 313+87-194-137= 69 \end{aligned}$$

Imagine integrală

Imagine integrală

34	11	33	3	19	19	18	18	2	24
34	8	36	11	5	11	5	6	15	33
17	22	17	4	6	3	5	7	35	18
8	3	15	22	5	1	20	10	12	22
8	7	1	22	19	29	6	20	9	27
16	7	11	17	15	2	25	19	29	10
34	26	29	31	5	6	30	17	4	10
33	28	30	4	28	21	26	5	32	21
1	18	13	5	27	16	28	19	32	23
12	13	16	23	13	7	21	5	2	15

	Sum	Cost	Total Cost
$\Sigma\Sigma$	352	23	28
II	352	3	6

Imagine inițială

34	45	78	81	100	119	137	155	157	181
68	87	156	170	194	224	247	271	288	345
85	126	212	230	260	293	321	352	404	479
93	137	238	278	313	347	395	436	500	597
101	152	254	316	370	433	487	548	621	745
117	175	288	367	436	501	580	660	762	896
151	235	377	487	561	632	741	838	944	1088
184	296	468	582	684	776	911	1013	1151	1316
185	315	500	619	748	856	1019	1140	1310	1498
197	340	541	683	825	940	1124	1250	1422	1625

$$\text{II: } 911 + 156 - 247 - 468 = 352$$

Imagine integrală

Imagine integrală

34	11	33	3	19	19	18	18	2	24
34	8	36	11	5	11	5	6	15	33
17	22	17	4	6	3	5	7	35	18
8	3	15	22	5	1	20	10	12	22
8	7	1	22	19	29	6	20	9	27
16	7	11	17	15	2	25	19	29	10
34	26	29	31	5	6	30	17	4	10
33	28	30	4	28	21	26	5	32	21
1	18	13	5	27	16	28	19	32	23
12	13	16	23	13	7	21	5	2	15

Imagine inițială

	Sum	Cost	Total Cost
$\Sigma\Sigma$	141	8	36
II	141	3	9

34	45	78	81	100	119	137	155	157	181
68	87	156	170	194	224	247	271	288	345
85	126	212	230	260	293	321	352	404	479
93	137	238	278	313	347	395	436	500	597
101	152	254	316	370	433	487	548	621	745
117	175	288	367	436	501	580	660	762	896
151	235	377	487	561	632	741	838	944	1088
184	296	468	582	684	776	911	1013	1151	1316
185	315	500	619	748	856	1019	1140	1310	1498
197	340	541	683	825	940	1124	1250	1422	1625

Imagine integrală

$$\text{II: } 762 - 621 = 141$$

Exemplu de învățare/antrenare

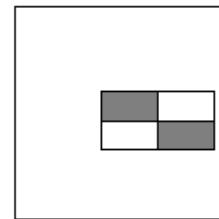
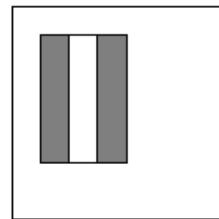
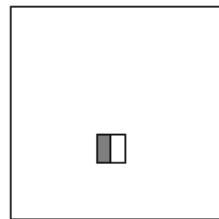
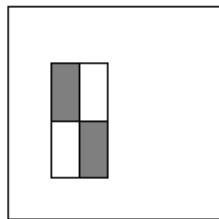
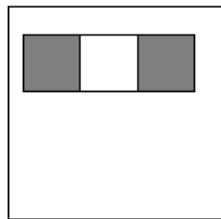
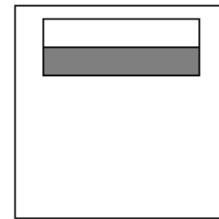
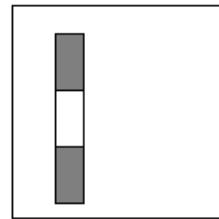
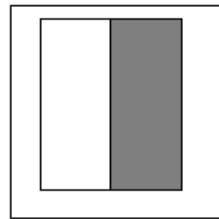
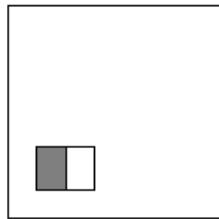
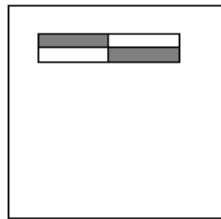
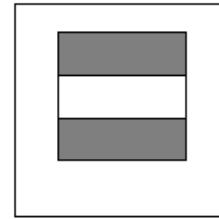
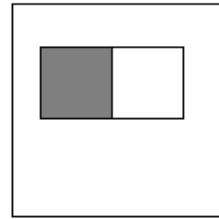
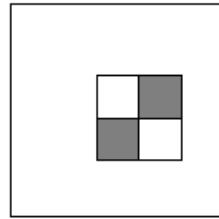
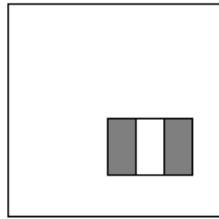
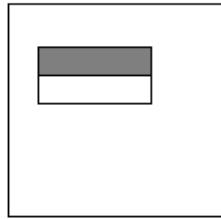
Fețe



non-Fețe



Colectie mare de filtre



Se consideră toate
filtrele posibile în
funcție de :
poziție, scală/
mărime și tip:
160,000+
caracteristici
măsurate pentru o
fereastră de
dimensiuni 24 x 24
pixeli

Selectarea caracteristicilor

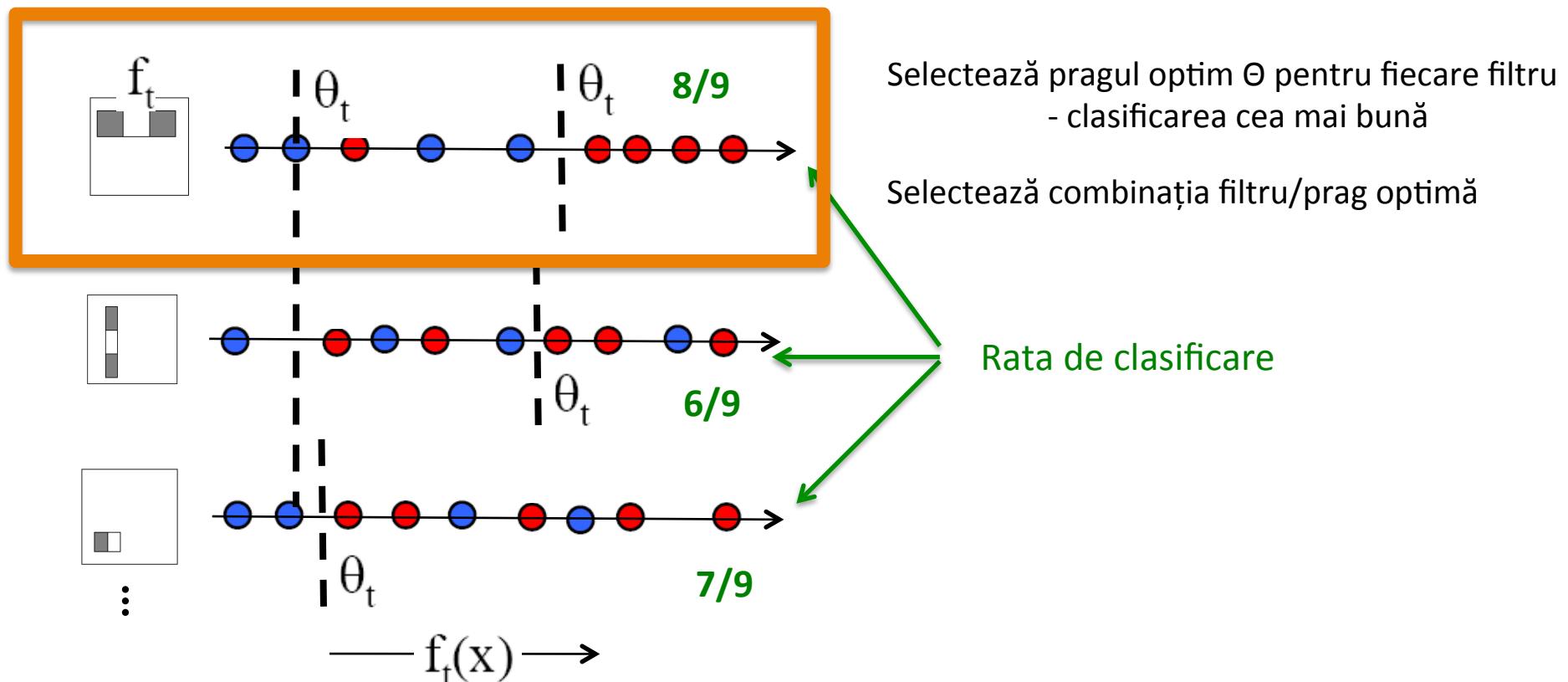
- pentru o fereastră de dimensiuni 24x24 pixeli, numărul de caracteristici Haar posibile (filtre dreptunghiulare) este $\sim 160,000$!
- pentru o imagine test, este foarte lent să evaluăm întregul set de 160,000 de caracteristici
- putem crea un clasificator bun folosind numai o submulțime mică de caracteristici (din cele $\sim 160,000$ posibile) ?
- cum să selectăm o asemenea submulțime?

Boosting

- Boosting – obține un clasificator binar ‘puternic’ (cu acuratețe mare) prin combinarea mai multor clasificatori binari ‘slabi’ (cu acuratețe mică)
 - un clasificator slab trebuie doar să performeze mai bine decât 50% în clasificarea binară (mai bine decât o simplă ‘ghicire’)
- Învățarea/antrenarea constă din mai mulți pasi:
 - la fiecare pas, selectăm clasificatorul ‘slab’ care clasifică cel mai bine exemplele care erau ‘greu’ de clasificat pentru clasificatorii anteriori
 - exemplele “greu” de clasificat – capătă ponderi (Adaptive Boosting – AdaBoost)

AdaBoost pentru selectarea caracteristicii + clasificatorului

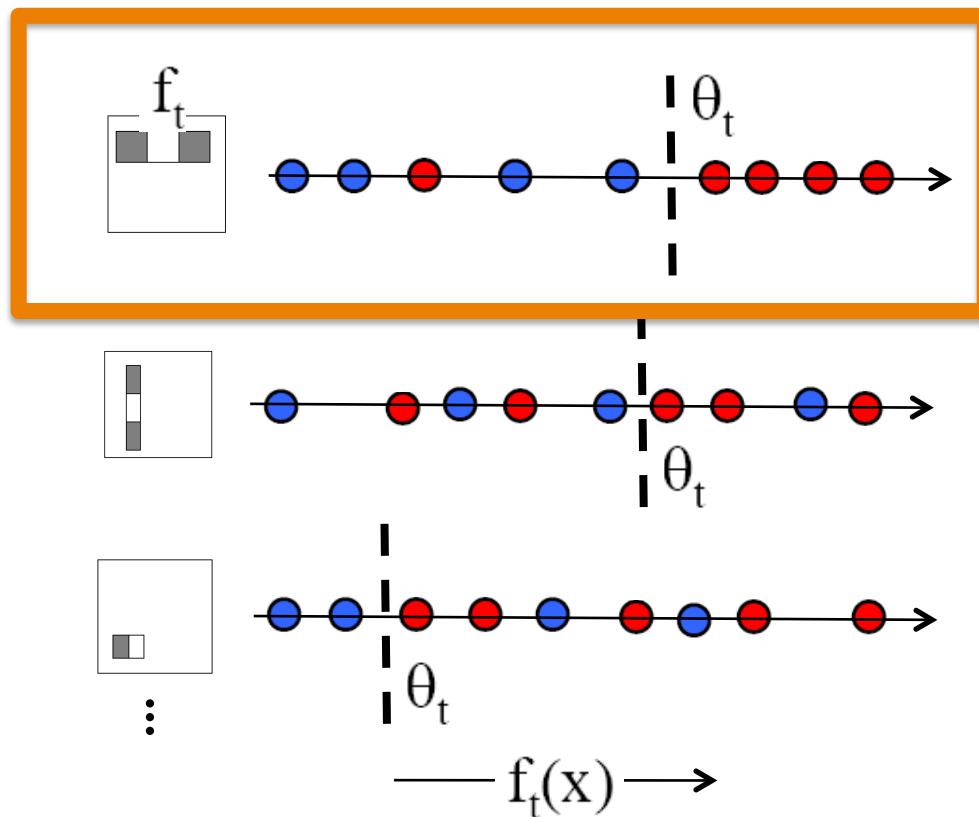
- Vrem să selectăm filtrul dreptunghiular și pragul care separă cel mai bine exemplele de învățare **pozitive (fete)** de cele **negative (non-fete)**, în termeni de #exemple incorect clasificate.



Output-ul unui filtru rectangular pe diverse imagini (**fete** sau **non-fete**)

AdaBoost pentru selectarea caracteristicii + clasificatorului

- Vrem să selectăm filtrul dreptunghiular și pragul care separă cel mai bine exemplele de învățare **pozitive (fete)** de cele **negative (non-fete)**, în termeni de #exemple incorect clasificate.



Output-ul unui filtru rectangular pe diverse imagini (**fete** sau **non-fete**)

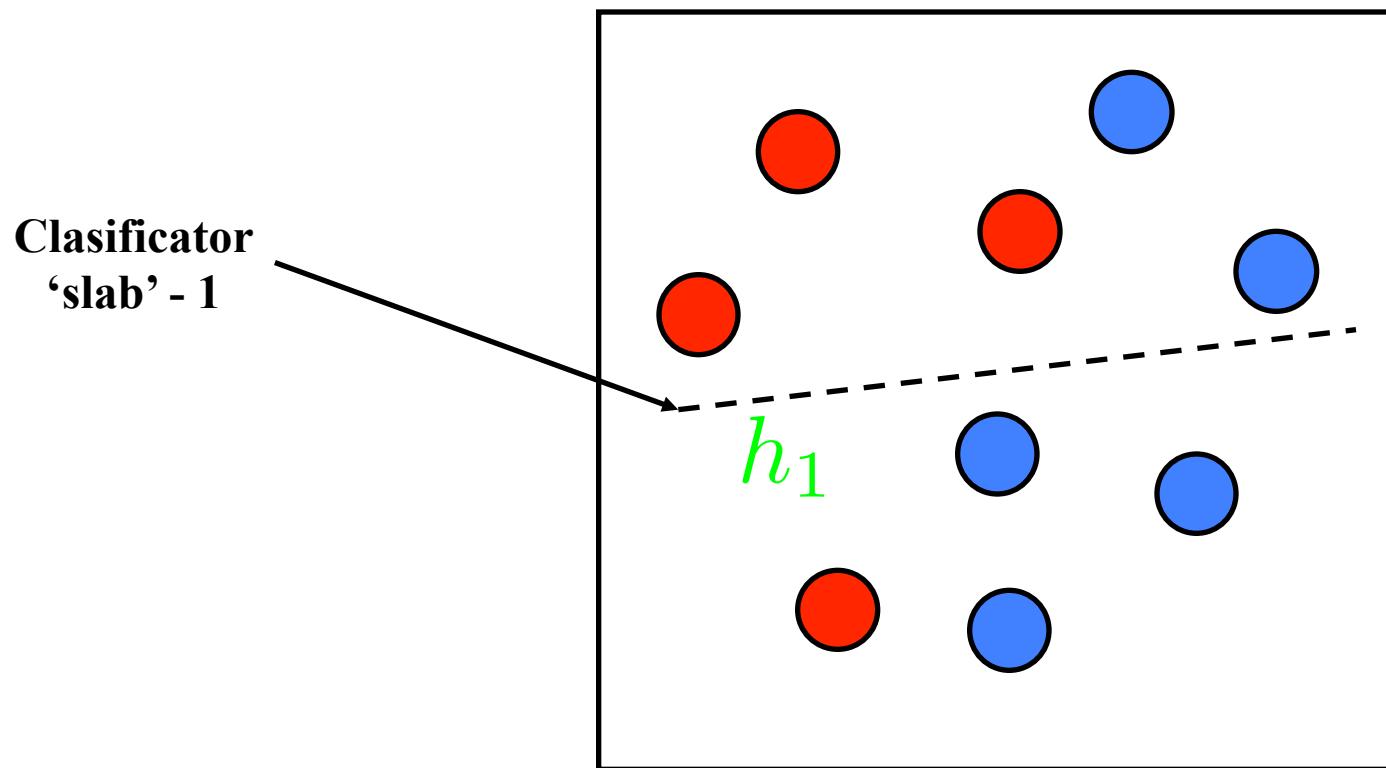
Rezultă un clasificator slab:


$$h_t(x) = \begin{cases} +1 & \text{if } f_t(x) > \theta_t \\ -1 & \text{altfel} \end{cases}$$

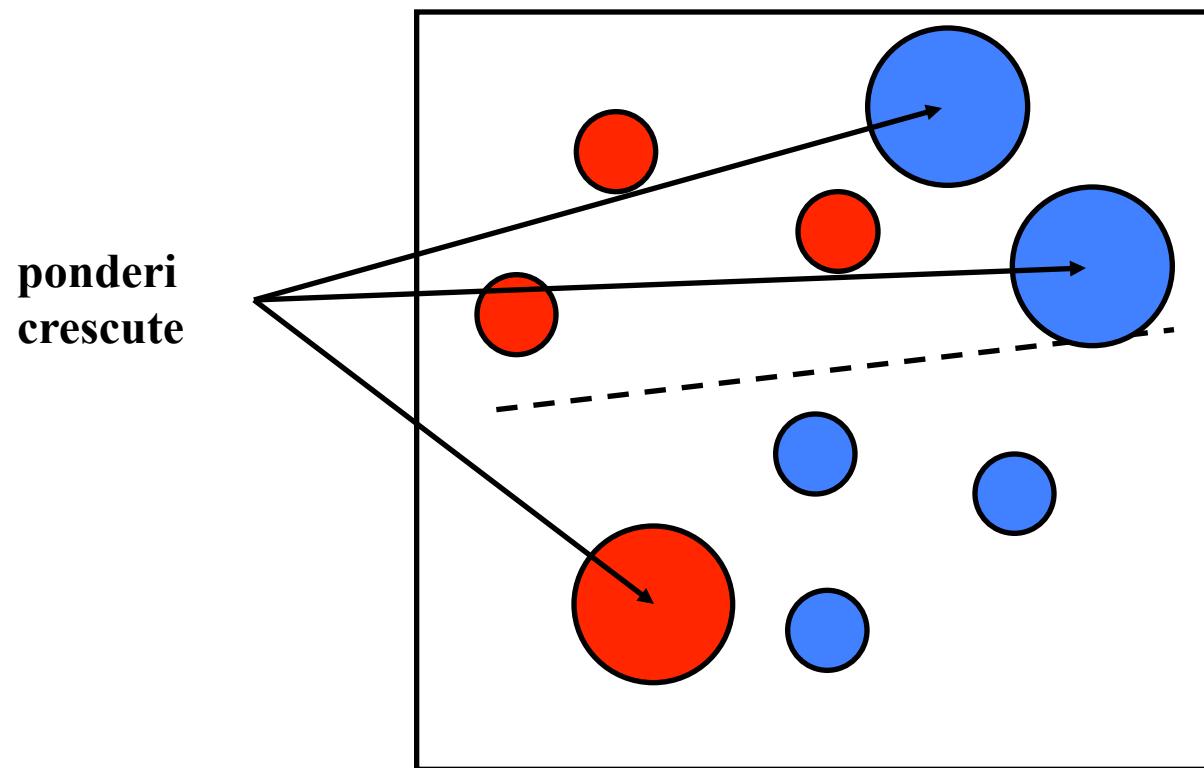
Pentru pasul următor:

- ponderează exemplele în funcție de erori
- alege un alt filtru/prag

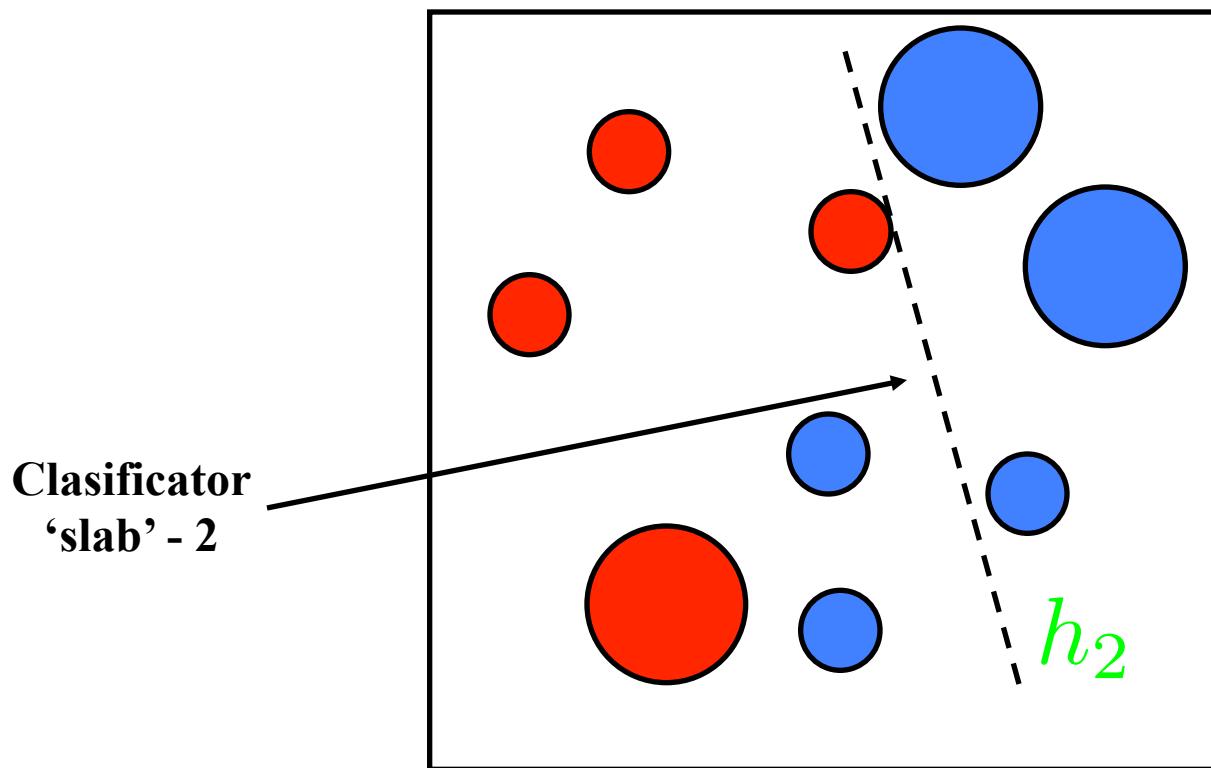
Boosting - intuiție 2D



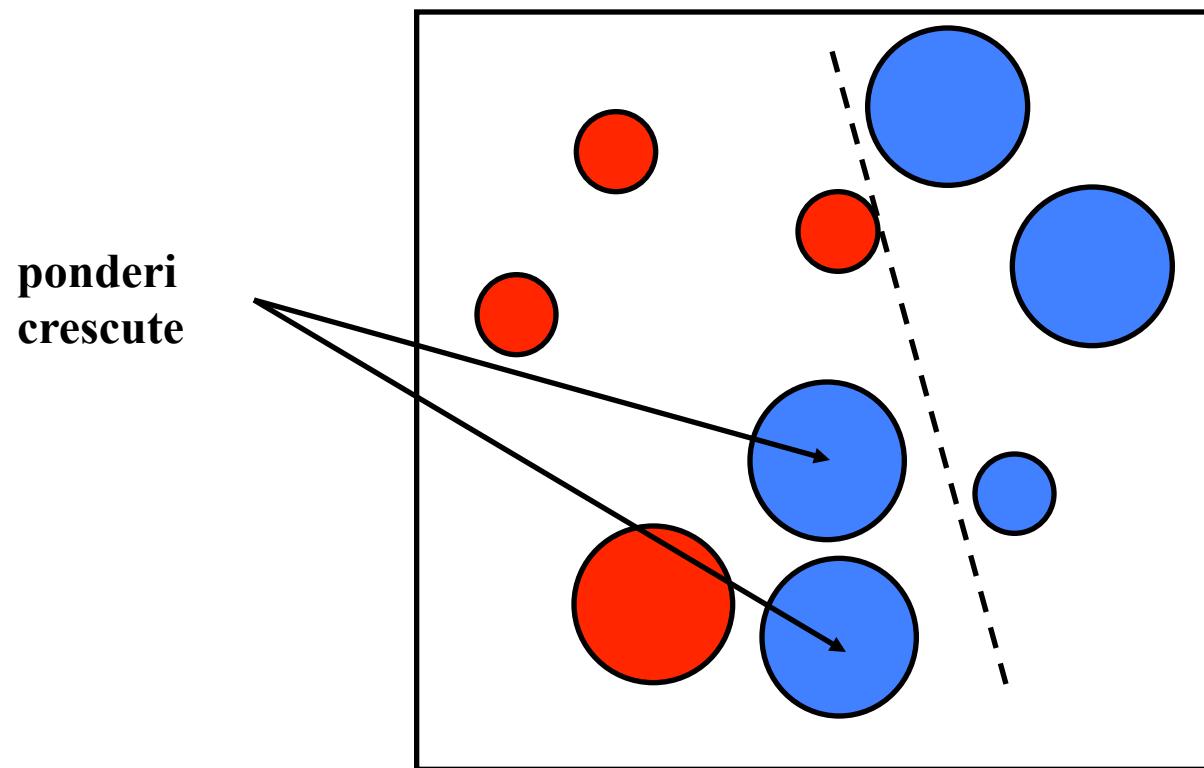
Boosting - intuiție 2D



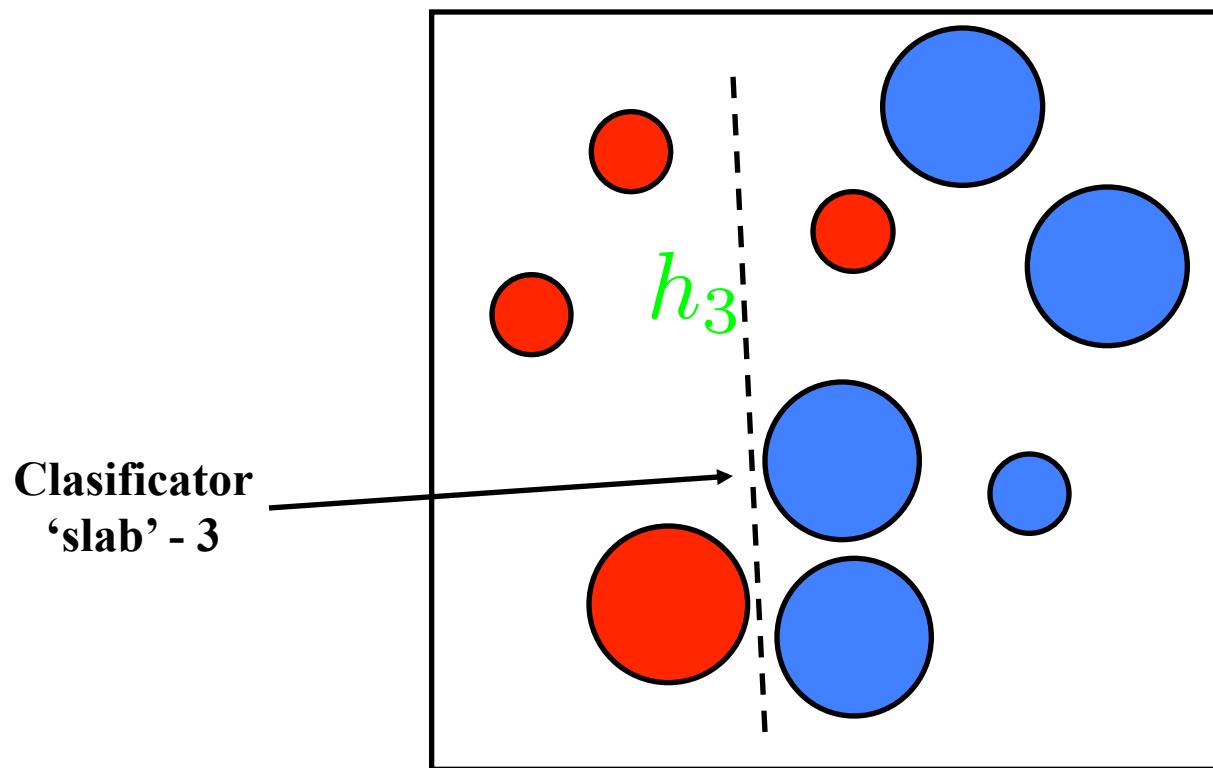
Boosting - intuiție 2D



Boosting - intuiție 2D

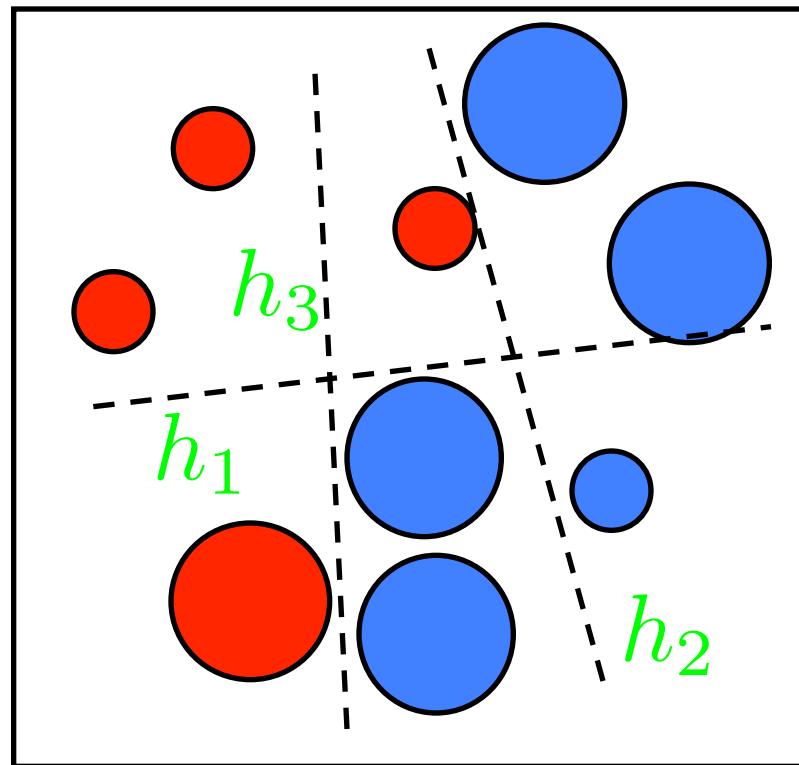


Boosting - intuiție 2D



Boosting - intuiție 2D

**Clasificatorul final =
combinație de clasificatorii
mai slabî**



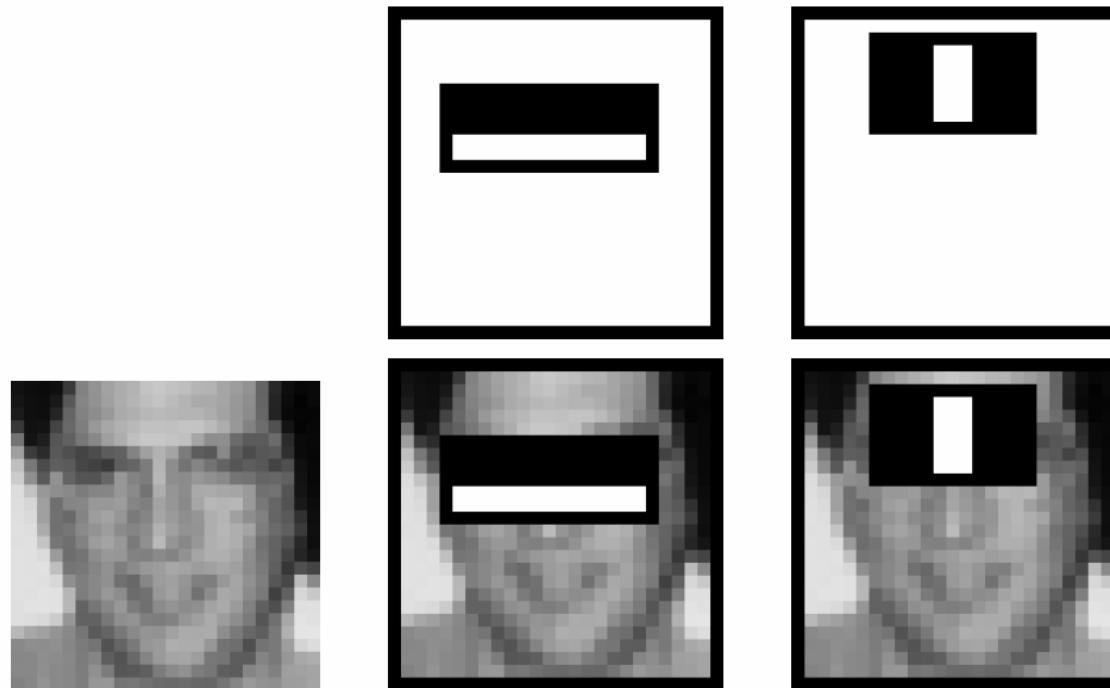
$$h_{final} = \alpha_1 h_1 + \alpha_2 h_2 + \alpha_3 h_3$$

Etapa de învățare/antrenare

- inițial, toate exemplele de antrenare au aceeași pondere (la fel de ‘greu’ de clasificat)
- la fiecare pas:
 - găsește clasificatorul ‘slab’ care are cea mai mică rată de eroare (ponderată) pentru clasificarea exemplelor de învățare – alege combinație filtru/prag cu cea mai bună performanță de clasificare
 - mărește ponderea exemplelor de învățare clasificate greșit de actualul clasificator ‘slab’
- calculează clasificatorul final ca o combinație liniară de toți clasificatorii slabi (ponderea fiecărui clasificator ‘slab’ este direct proporțională cu performanța lui de clasificare)

Caracteristici selectate

- primele două caracteristici selectate:



această combinație de caracteristici conduce la

- rată de detectare de 100% (nu ratăm nicio față)
- 50% rată de detectii fals pozitive (jumătate din ferestrele returnate ca detectii nu sunt fețe)

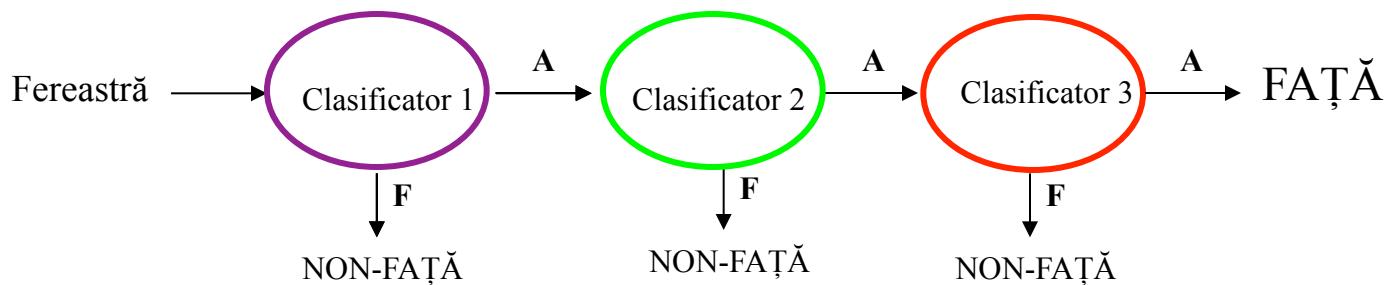
Performanță

- un clasificator cu 200 de caracteristici:
 - 95% rată de detectare (recall = câte fețe găsesc din cele existente în imagini)
 - rată de detecții fals pozitive 1 din 14084



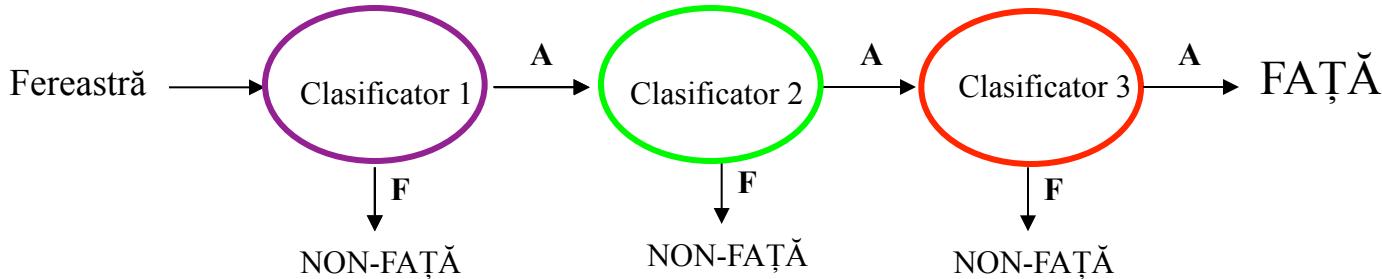
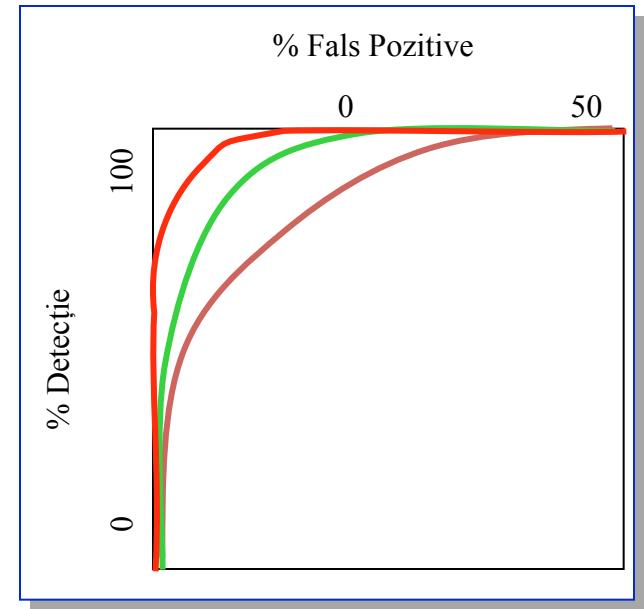
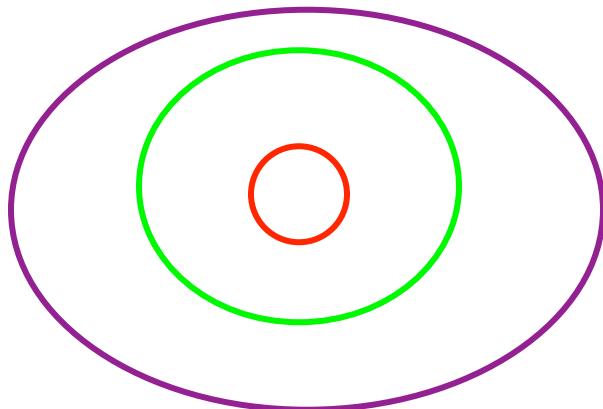
Cascadă de clasificatori

- începem cu clasificatori simpli care elimină multe dintre ferestrele non-fete, păstrând pe cele care conțin fețe
- un răspuns pozitiv pentru primul clasificator ($\text{output} > \Theta$) conduce la evaluarea unui clasificator (mai complex), și tot așa
- un răspuns negativ la orice pas conduce la eliminarea imediată a ferestrei (clasificarea ei ca un exemplu negativ – non-față)



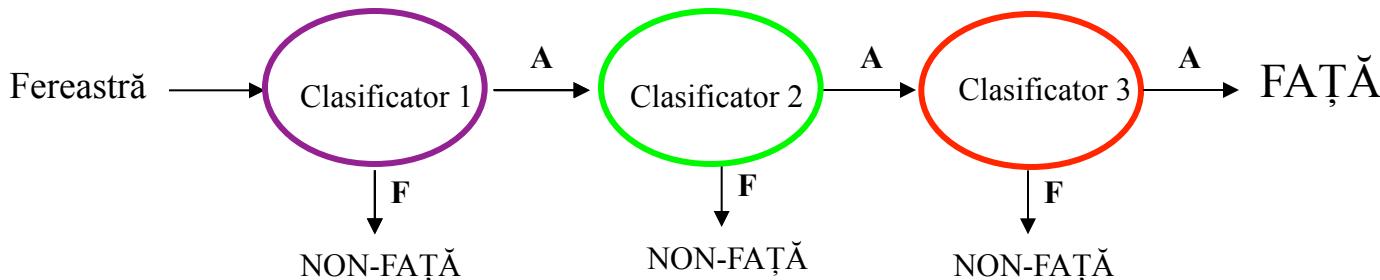
Cascadă de clasificatori

- evaluăm clasificatori în lanț care sunt din ce în mai complecși și au rată de detecție mare + rată de detectii fals pozitive mică



Cascadă de clasificatori

- rata de detectare și rata de detecții false pozitive le aflăm prin a multiplica valorile acestea de la fiecare clasificator încă din partea
- o rată de detectare de 0.9 și o rată de detecții false pozitive de ordin 10^{-6} pot fi obținute printr-o cascadă de 10 clasificatori astfel încât:
 - fiecare clasificator are o rată de detectare de 0.99 ($0.99^{10} \approx 0.9$)
 - fiecare clasificator are o rată de detecții false pozitive de 0.30 ($0.3^{10} \approx 6 \times 10^{-6}$)



Învățarea unei cascade de clasificatori

- fixăm rata de detectare și rata de detectii fals pozitive pentru fiecare etapă/clasificator
- adăugăm caracteristici (filtre/prag) la etapa curentă/ clasificatorul curent până când atingem performanța dorită (detectare + fals pozitive)
 - modificăm AdaBoost astfel încât să maximizăm rata de detectare (alegem alt prag Θ) (în loc să minimizăm eroare de clasificare)
 - testează pe un set de validare (în loc de set de învățare + test, avem set de învățare + set de validare + test)
- folosește detectiile fals pozitive de la etapa curentă ca exemple de învățare negative la etapa viitoare

Detalii de implementare

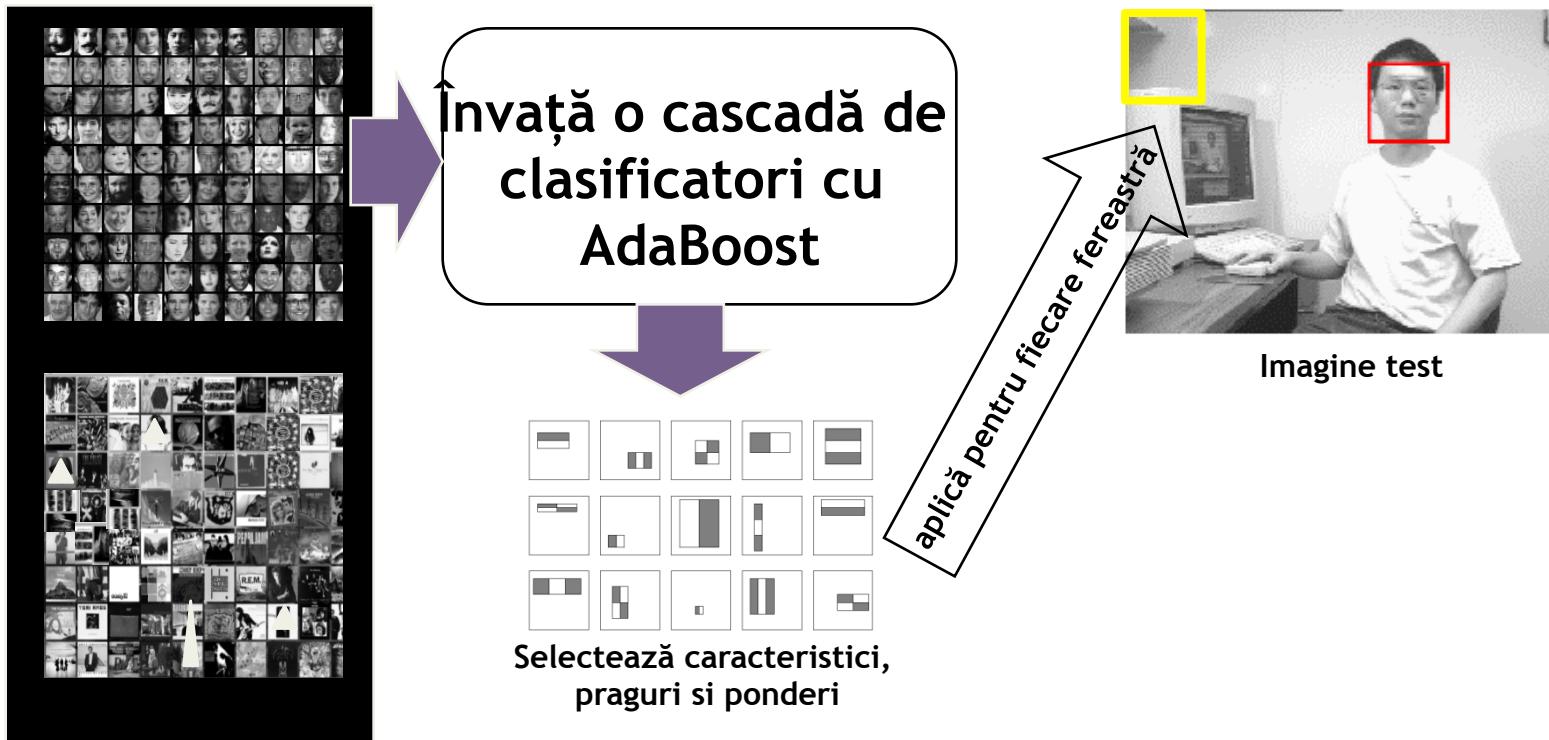
- Setul de învățare
 - 5000 fețe
 - frontal
 - 24x24 pixeli
 - 300 milioane non-fețe
 - 9500 imagini fără fețe
 - fețe normalize
 - scală, centrate
- Variabilitate mare
 - iluminare
 - expresii faciale
 - ochelari
 - mustață



Detalii de implementare

- timp de învățare: “săptămâni” pe o mașină Sun 466 MHz
- 38 de etape/clasificatori, în total 6061 caracteristici
- în medie 10 caracteristici evaluate /fereastră pentru o imagine test
- detectarea facială pentru o imagine 383 x 288 pixeli în 0.067 secunde

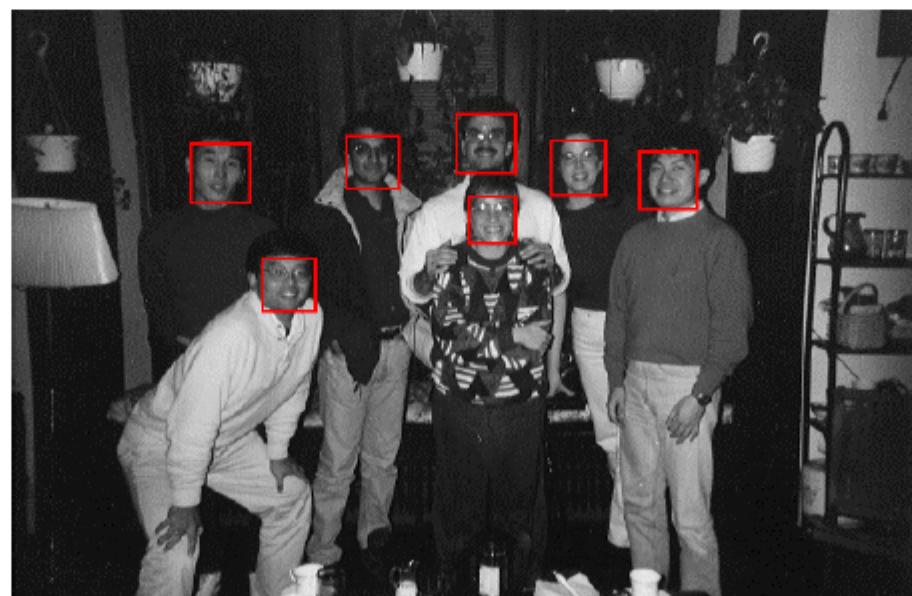
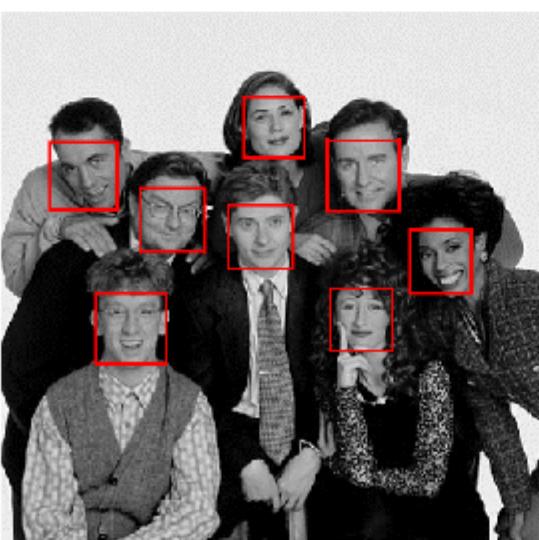
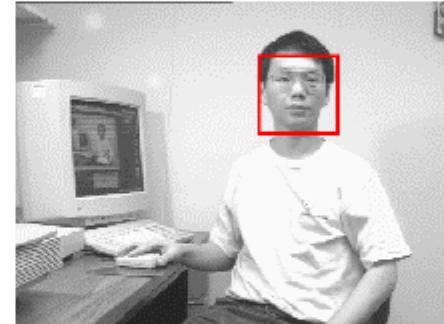
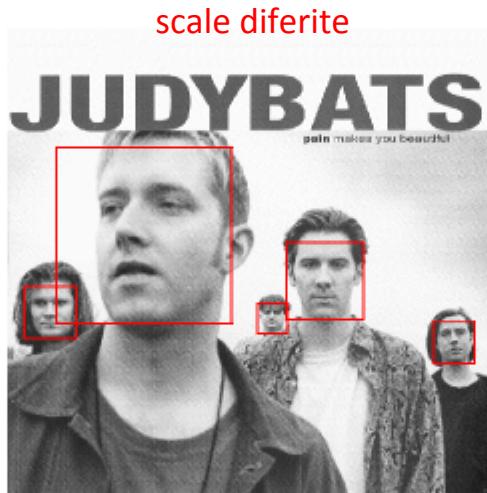
Detectorul facial Viola-Jones: sumar



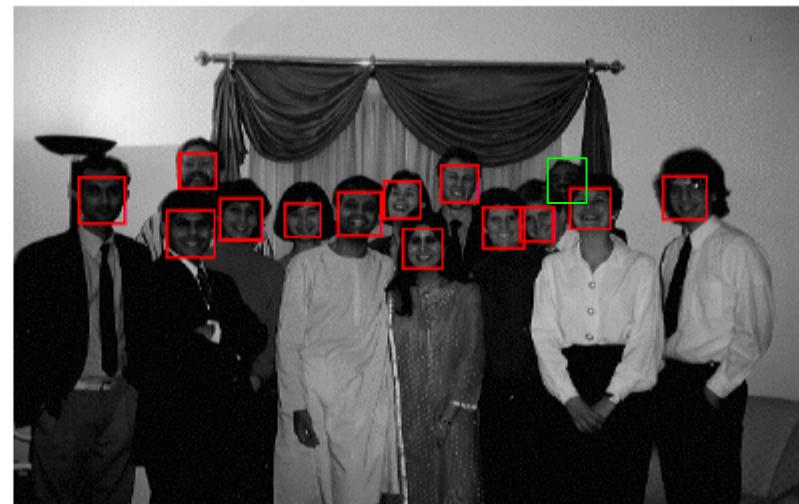
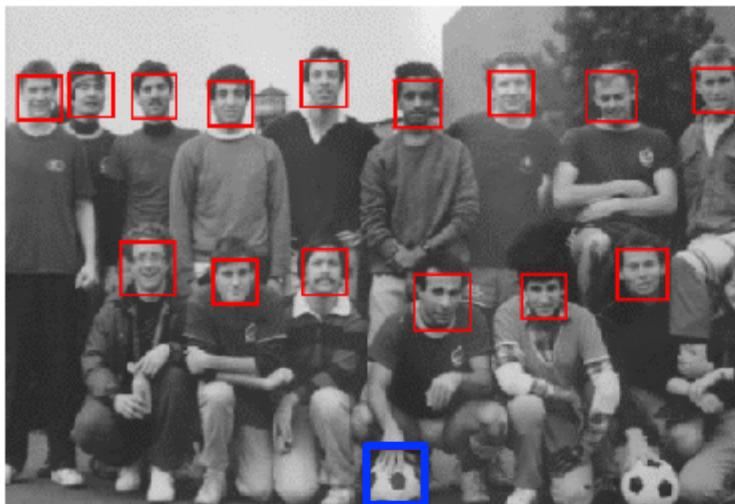
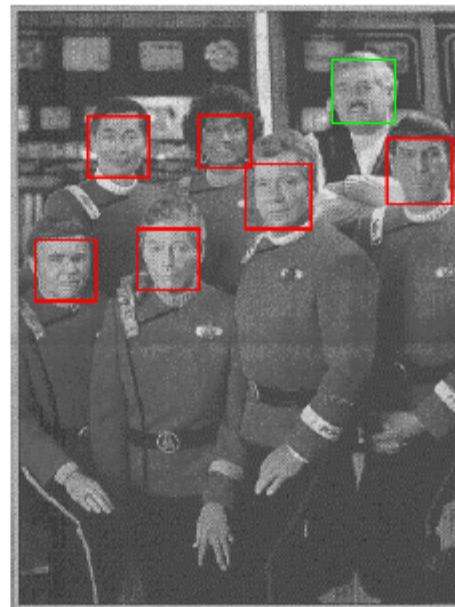
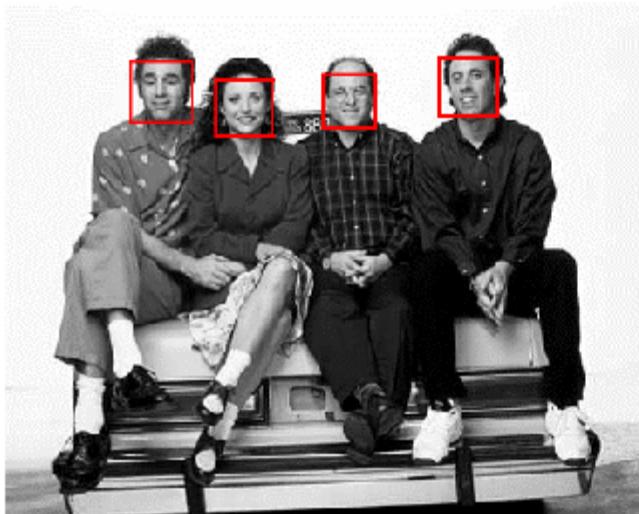
- î învață cu 5K exemple pozitive, 300M exemple negative
- detector facial în timp real - cascadă de 38 de clasificatori
- 6061 de caracteristici în total

Detectorul facial Viola-Jones: rezultate

Recall = $2/5 = 40\%$



Detectorul facial Viola-Jones: rezultate



1 exemplu fals pozitiv pe imagine

Detectorul facial Viola-Jones: rezultate

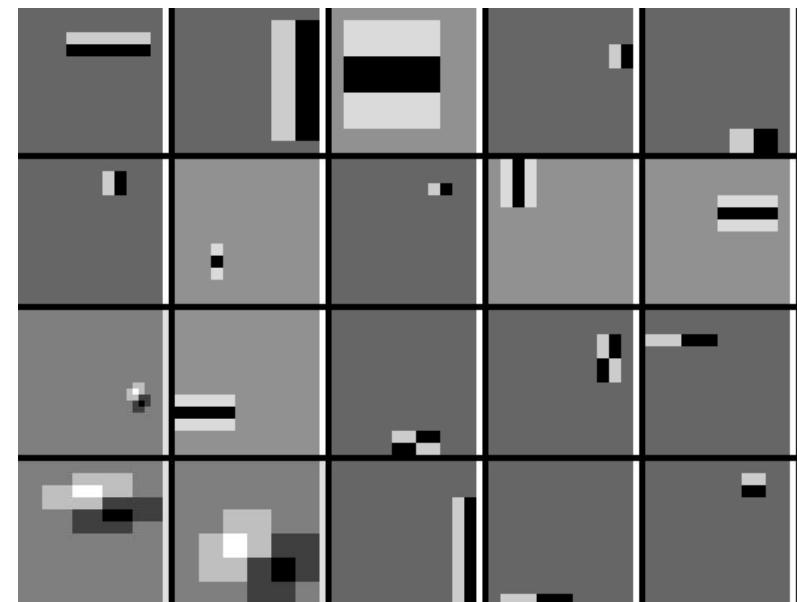


Detectorul facial Viola-Jones: rezultate

**cs341 sample video
face detection
Viola-Jones method**

Detectarea feței din profil

Detectarea fețelor din profil necesită învățarea unui detector separat cu exemple de învățare = fețe din profil



Detectorul facial Viola-Jones: rezultate

