

Vedere Artificială (Computer Vision)

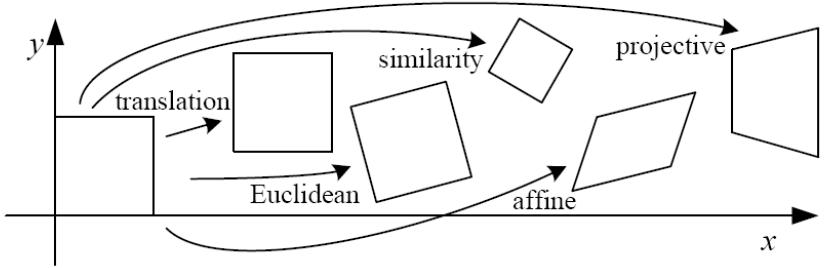
Bogdan Alexe

bogdan.alexe@fmi.unibuc.ro

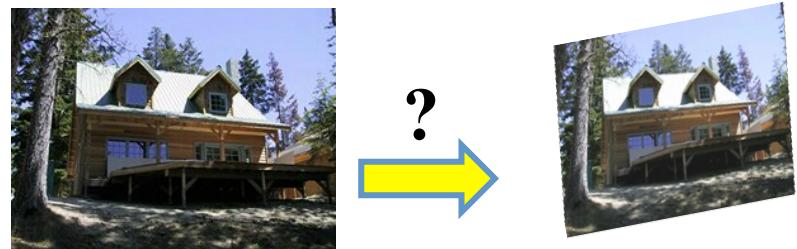
anul 2, master Informatică, semestrul I, 2019-2020

Recapitulare – cursul trecut

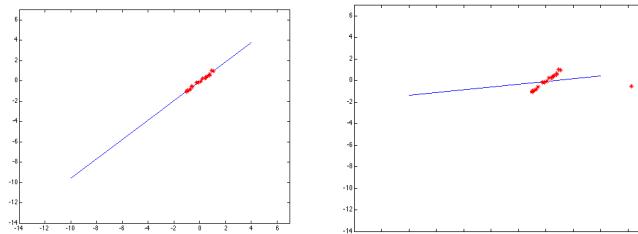
- Transformări geometrice 2D



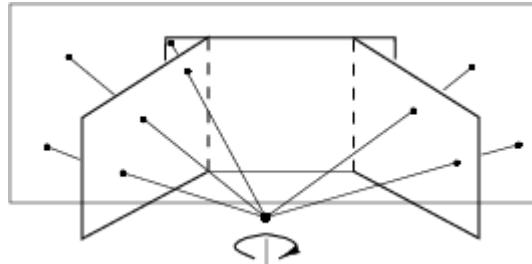
- Calculul unei transformări



- RANSAC pentru transformări

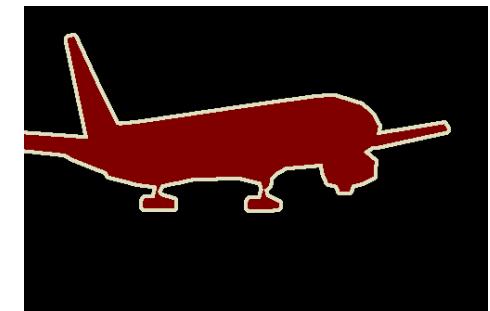
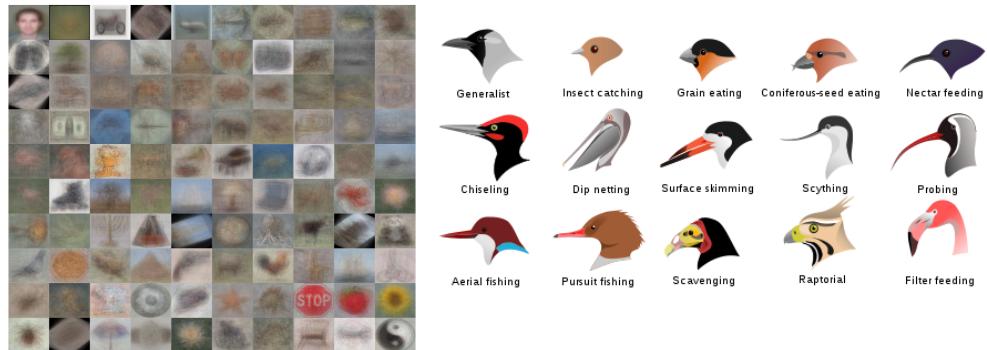
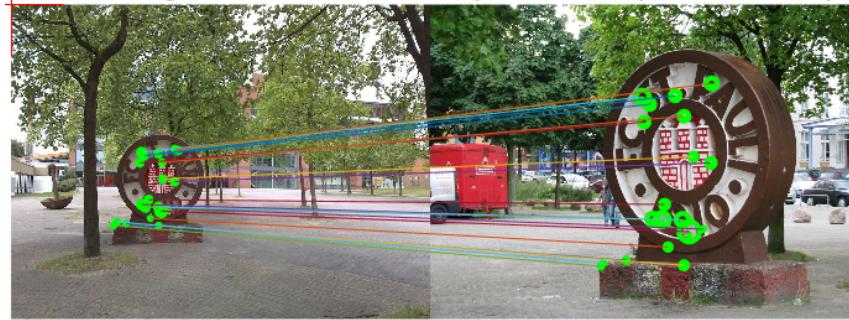


- Construcția de panorame



Cursul de azi

- Prezentare tema 2
- Clasificare în Vederea Artificială
- Detectarea de obiecte în imagini



Tema 2 – Regăsirea imaginilor folosind modelul bag of visual words

- set de date cu 50 de clase
- fiecare clasă conține 10 imagini cu aceeași scenă
- în total sunt 500 de imagini de antrenare
- scopul este de a construi un sistem care regăsește imaginile similare cu o imagine query
- sistem bazat pe:
 - modelul bag of visual words
 - descriptori SIFT
 - verificare geometrică cu RANSAC
 - inverted file index
- imagini test – o imagine/clasă

Clasa 1



Clasa 2



Clasa 45



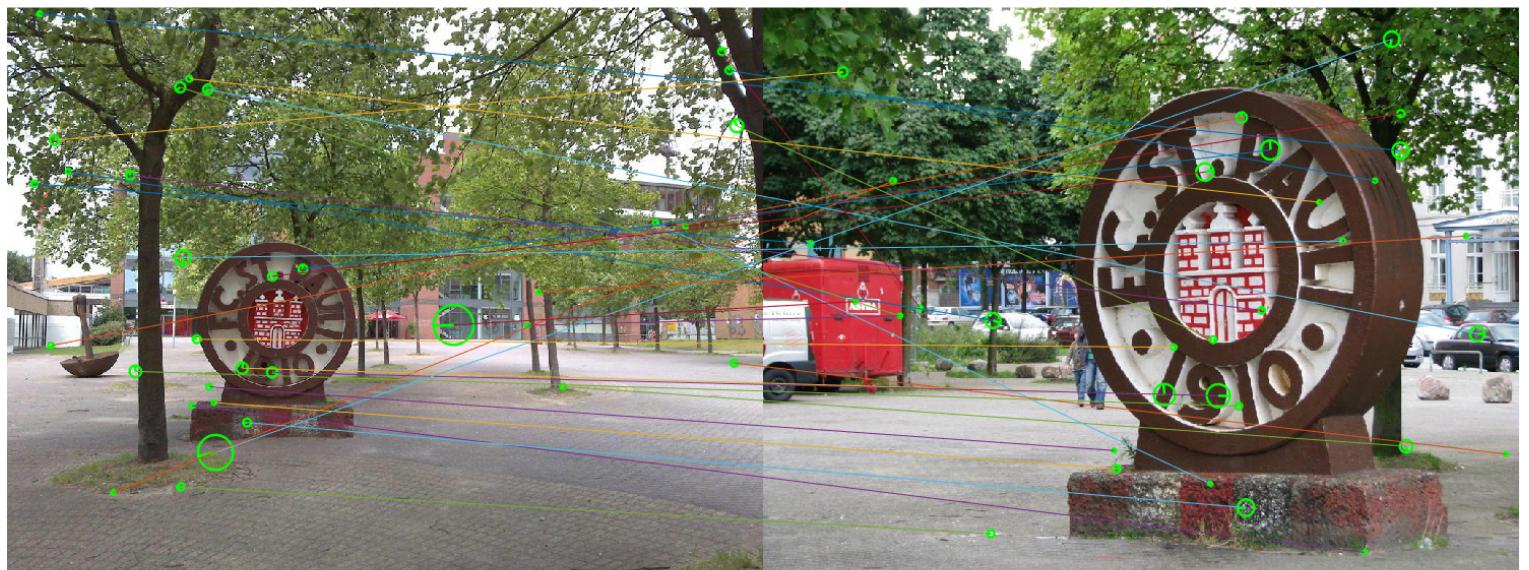
Extragere de SIFT-uri



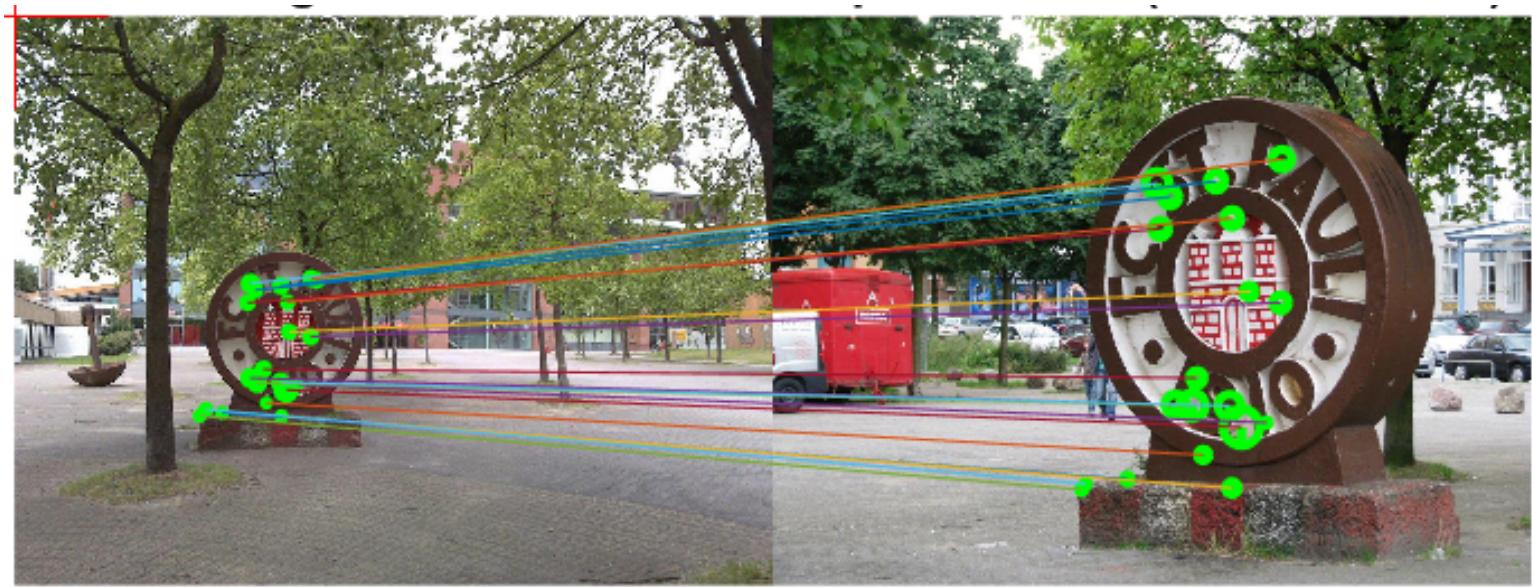
Găsire de corespondențe între descriptori



Găsire de corespondențe între descriptori folosind raportul dintre 1stNN/2ndNN



Găsire de corespondențe între descriptori folosind raportul dintre 1stNN/2ndNN + verificare geometrică cu RANSAC

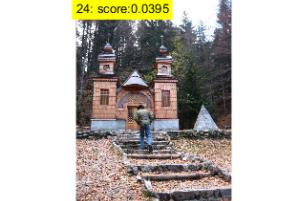
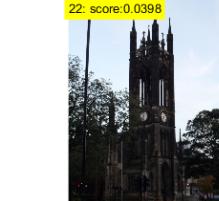


Găsire de corespondențe între cuvintele vizuale asociate descriptorilor cu verificare geometrică cu RANSAC



Regăsirea imaginilor pe bază de inverted file index

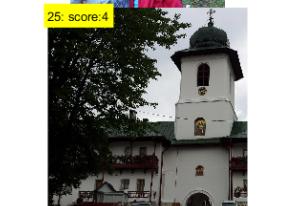
Imagine query



Regăsirea imaginilor pe bază de inverted file index + verificare geometrică



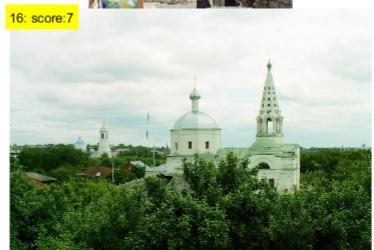
Imagine query



Regăsirea imaginilor pe bază de inverted file index + verificare geometrică



Imagine query



Home

The VLFeat [open source](#) library implements popular computer vision algorithms specializing in image understanding and local features extraction and matching. Algorithms include Fisher Vector, VLAD, SIFT, MSER, k-means, hierarchical k-means, agglomerative information bottleneck, SLIC superpixels, quick shift superpixels, large scale SVM training, and many others. It is written in C for efficiency and compatibility, with interfaces in MATLAB for ease of use, and detailed documentation throughout. It supports Windows, Mac OS X, and Linux. The latest version of VLFeat is [0.9.21](#).

ACM OpenSource
Award

Download

- [VLFeat 0.9.21](#) (Windows, Mac, Linux)
- [Source code and installation](#)
- [git repository, bug tracking.](#)

Tutorials

- Features: [Covariant detectors](#), [HOG](#), [SIFT](#), [MSER](#), [Quick shift](#), [SLIC](#)
- Clustering: [IKM](#), [HIKM](#), [AIB](#)
- Matching: [Randomized kd-trees](#)
- [All tutorials](#)

Example applications

- [Caltech-101 classification](#)
- [SIFT matching for auto-stitching](#)
- [All example applications](#)

Documentation

- [MATLAB commands](#)
- [CAPI](#) with algorithm descriptions
- [Command line tools](#)

Citing

```
@misc{vedaldi08vlfeat,  
Author = {A. Vedaldi and B. Fulkerson},  
Title = {{VLFeat}: An Open and Portable Library  
of Computer Vision Algorithms},  
Year = {2008},  
Howpublished = {\url{http://www.vlfeat.org/}}  
}
```

Acknowledgments



[UCLA Vision Lab](#) [Oxford VGG](#).

Detalii de implementare

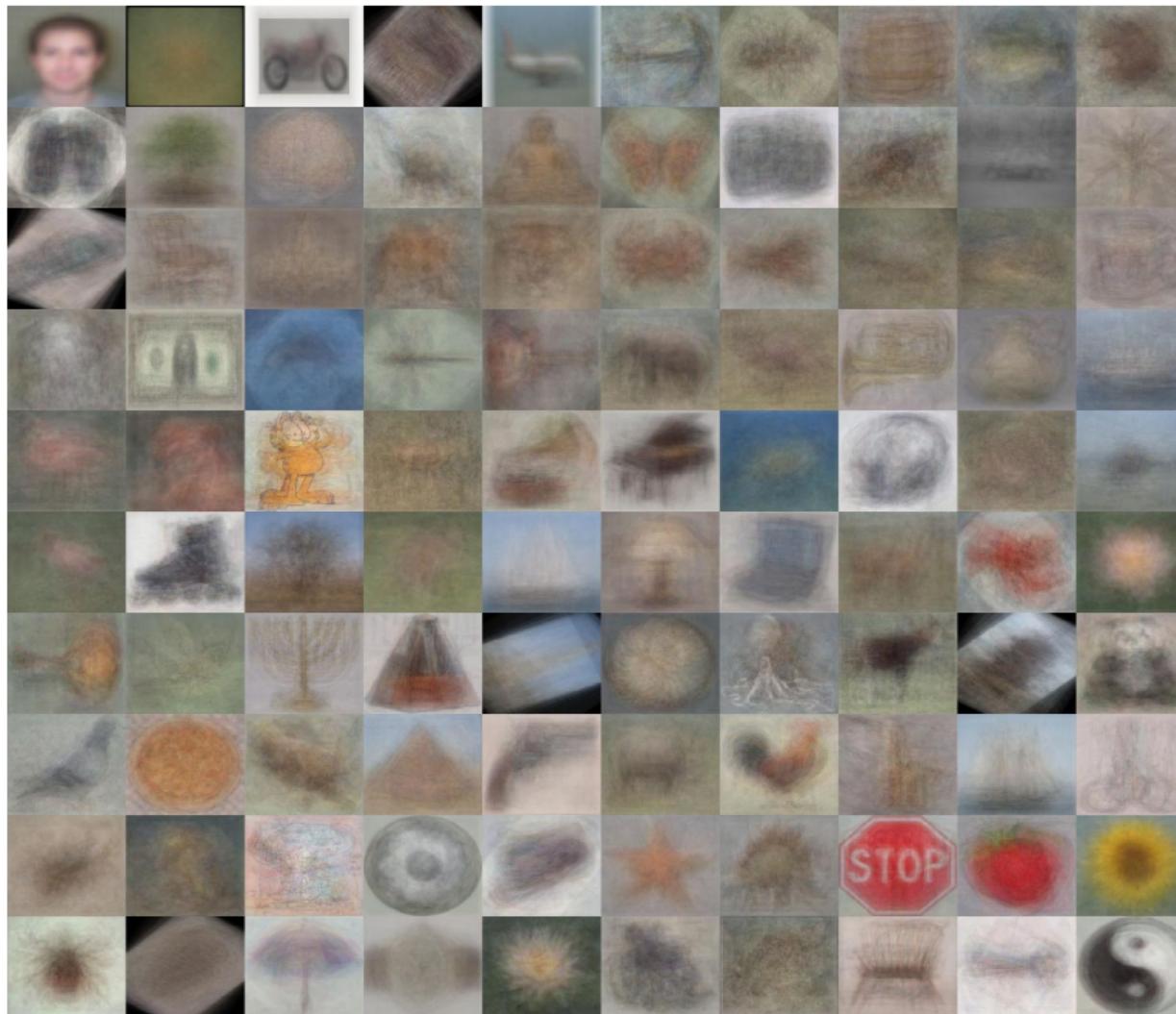
- Biblioteca vlfeat
- Detector de puncte de interes invariant la transformări afine
- Descriptori SIFT
- Dimensiunea vocabularului : 100.000 de cuvinte
- K-means++
- Găsirea de corespondențe cu BOVW
- Construcția inverted file index, matrice rară 500×100000
- TF-IDF
- Verificare geometrică cu RANSAC

Clasificare în Vederea Artificială

Probleme de clasificare în Vederea Artificială

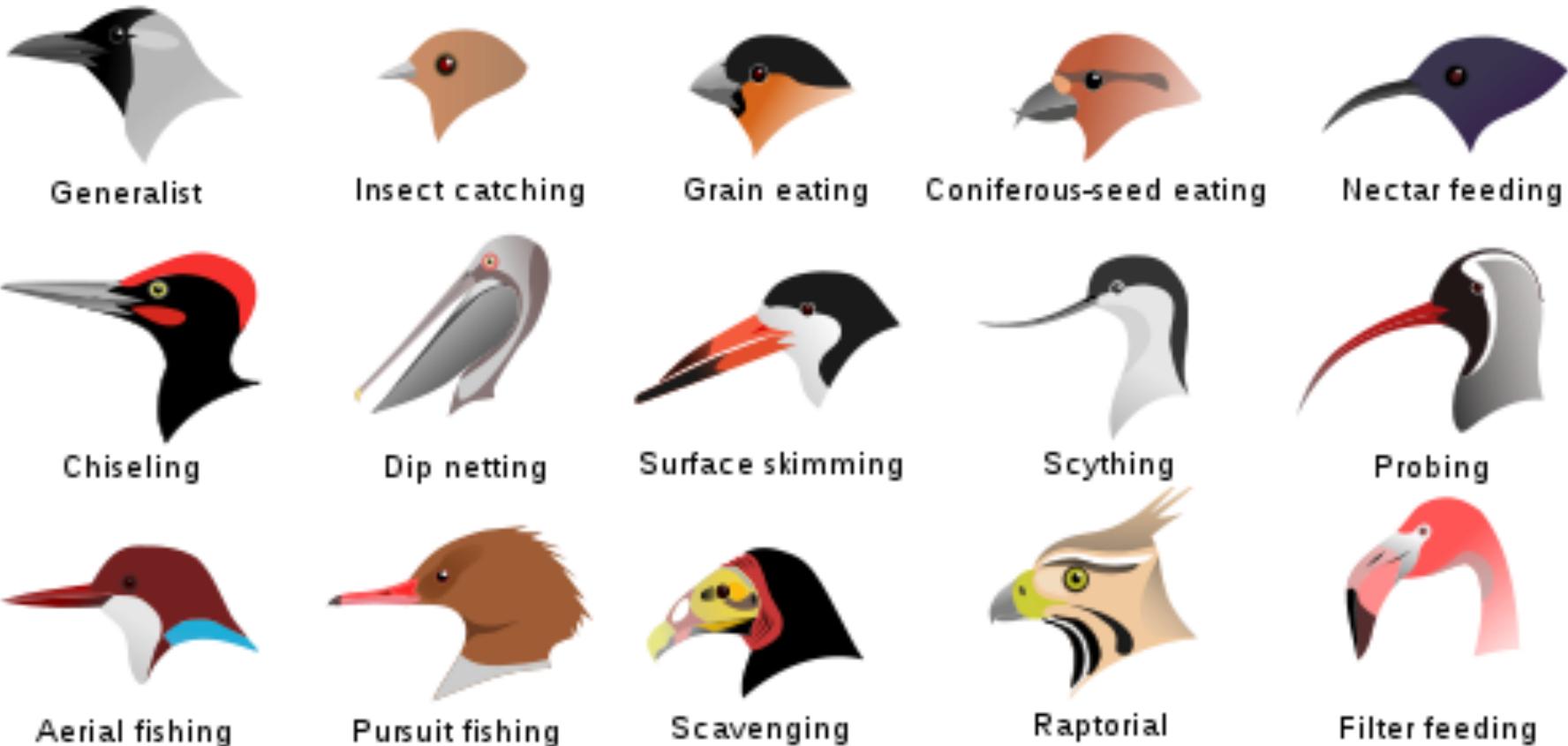
- Clasificarea imaginilor: indoor/outdoor, la nivel de scenă, ce obiecte sunt într-o imagine, etc.
 - Detectarea de obiecte: classificarea de regiuni (de formă generală sau dreptunghiulară – ferestre) dacă ele conțin sau nu un obiect
 - Segmentare Semantică: asignarea unei etichete fiecărui pixel dintr-o imagine dintr-o mulțime finită de clase
 - Recunoașterea Acțiunilor: clasifică un frame sau o secvență de frame-uri dintr-o video într-o anume acțiune
- ...

Clasificarea imaginilor



Recunoaștere de obiecte –Caltech 101: imaginea medie a unei clase de obiecte

Clasificarea imaginilor



Clasificarea la nivel de specie
(Fine grained object recognition)

[Visipedia Project](#)

Clasificarea imaginilor



spare bedroom

teenage bedroom

romantic bedroom



wooded kitchen

messy kitchen

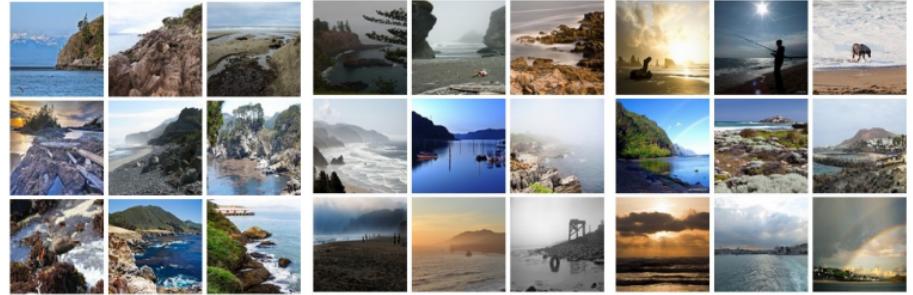
stylish kitchen



darkest forest path

wintering forest path

greener forest path



rocky coast

misty coast

sunny coast

Recunoaștere de scene: [[Zhou et al. NIPS 2014](#)]

Clasificarea imaginilor



Space Coast

Top Ranked Fonts

Adobe Caslon Pro Bold

Space Coast

Rotation LT Std Bold

Space Coast

Gazette LT Std Bold

Space Coast

Baskerville Cyr LT Std Bold

Space Coast

Joanna MT Std Bold

Space Coast



Saturday

Top Ranked Fonts

Hypatia Sans Pro Black

Saturday

Gill Sans Std Bold

Saturday

Montara Bold Gothic

Saturday

IT Ckabel Std Bold

Saturday

Myriad Arabic Black

Saturday

Clasificarea imaginilor



HDR



Macro



Baroque



Roccoco



Vintage



Noir



Northern Renaissance



Cubism



Minimal



Hazy



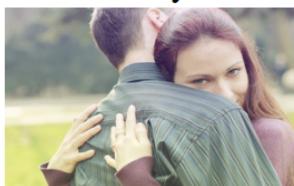
Impressionism



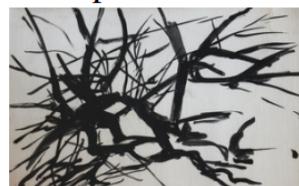
Post-Impressionism



Long Exposure



Romantic



Abs. Expressionism



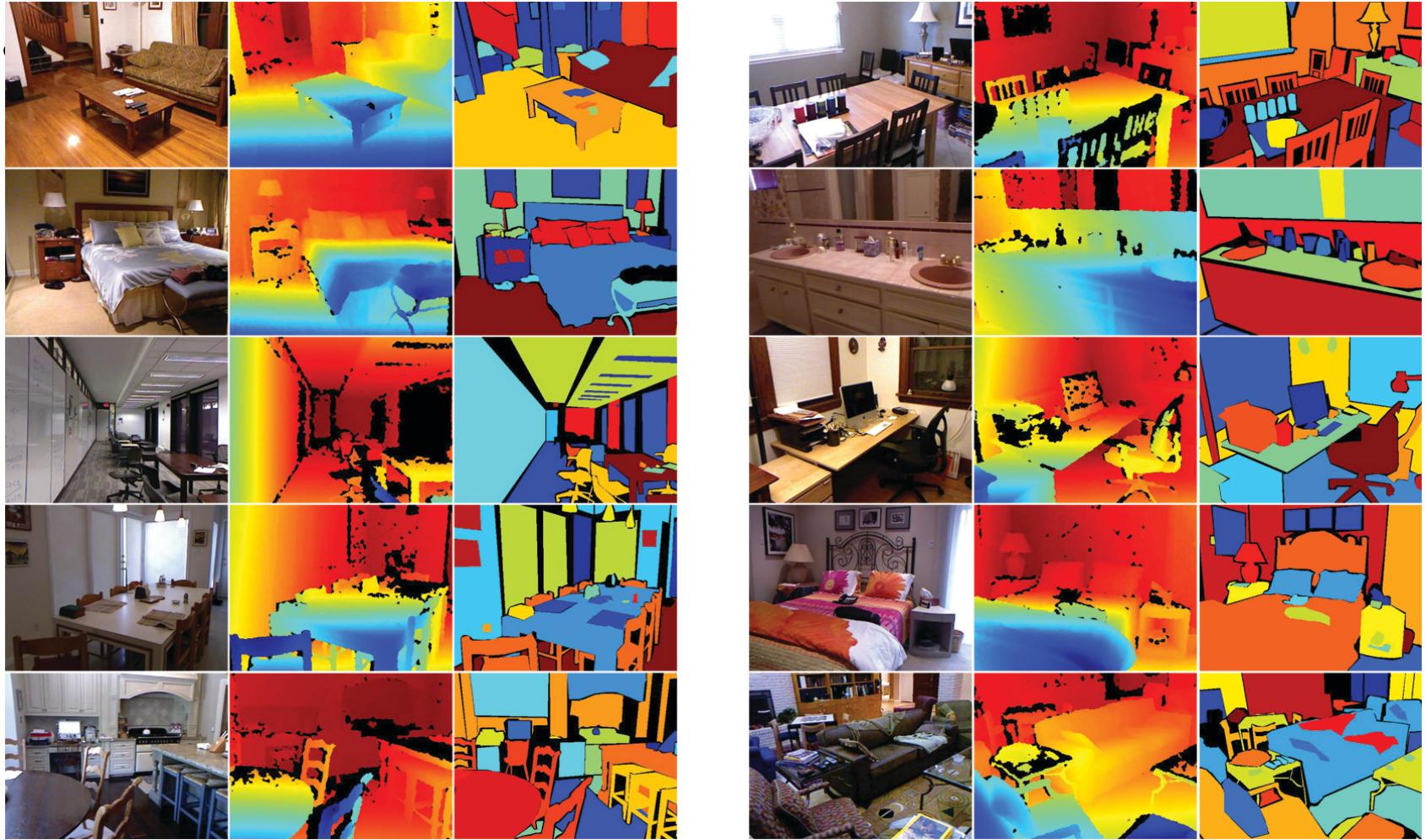
Color Field Painting

Flickr Style: 80K images covering 20 styles.

Wikipaintings: 85K images for 25 art genres.

Recunoaștere de stil [[Karayev et al. BMVC 2014](#)]

Clasificarea regiunilor



Segmentare semantică din imagini RGBD [[Silberman et al. ECCV 2012](#)]

Clasificarea regiunilor

brick	food	painted	tile
carpet	glass	paper	stone
ceramic	hair	plastic	water
fabric	leather	polishedstone	wood
foliage	metal	skin	



Segmentare semantică - [Bell et al. CVPR 2015]

Clasificarea imaginilor

- Conține o imagine test instantieri ale unei clase de obiecte/categorii ?
 - răspuns binar: DA/NU
- Exemplu: clasificator de imagini pentru ‘câine’

DA

NU

DA



Clasificarea imaginilor

Învățăm un clasificator al imaginilor pe bază de exemple

Exemple pozitive: imagini care conțin câini



Exemple negative: imagini care NU conțin câini



Alt exemplu: 2 clase - indoor/outdoor

Indoor – exemple



Outdoor - exemple



Clasificare = asocierea unei etichete (indoor, outdoor) pentru exemple noi

outdoor



outdoor

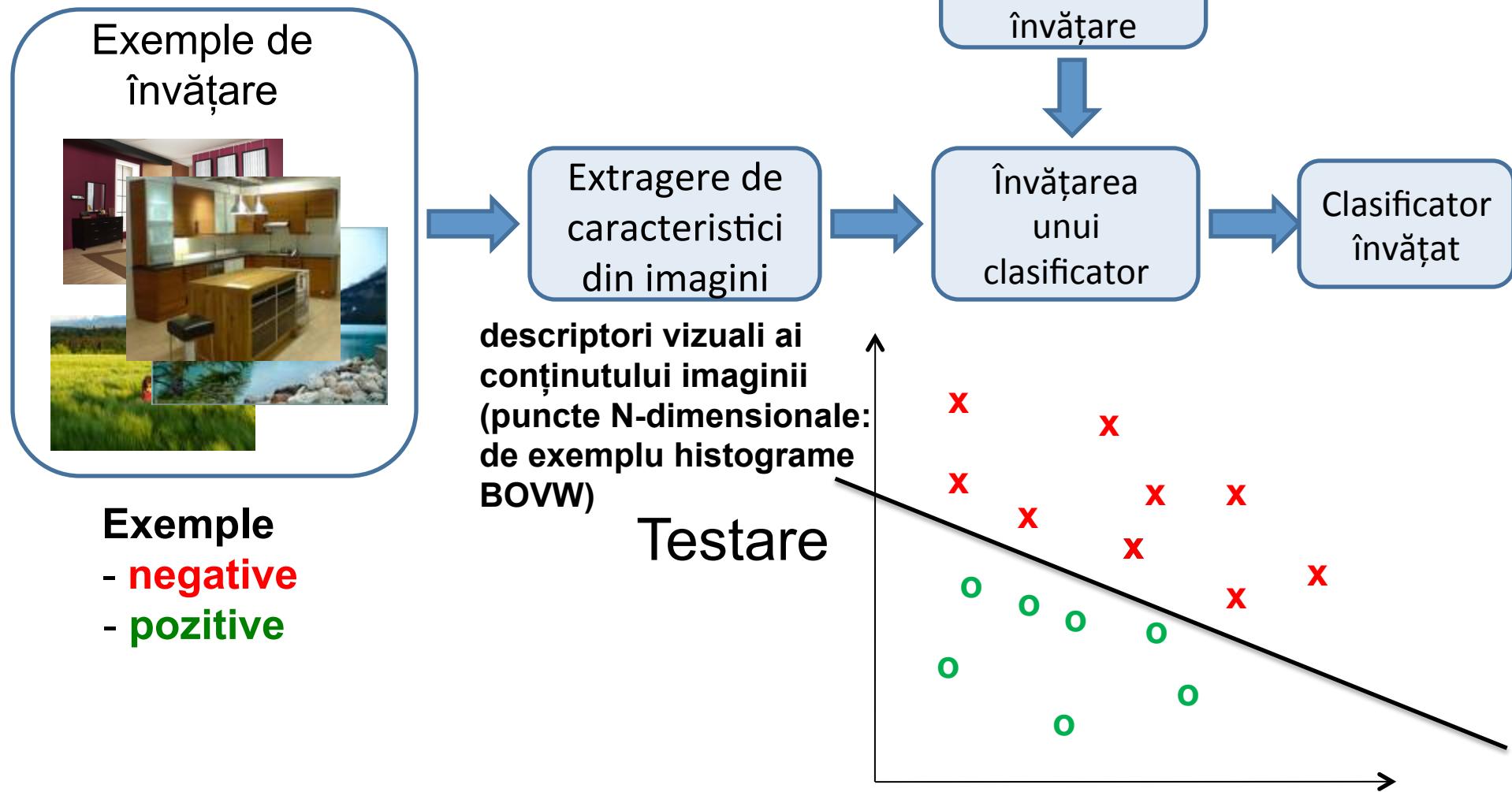


indoor

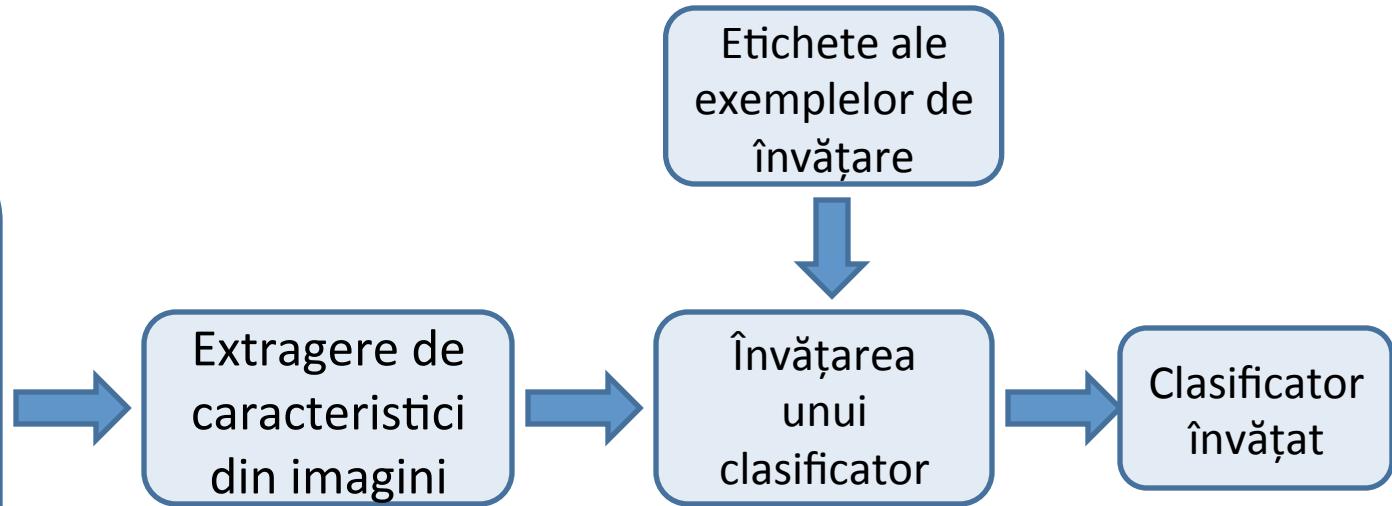


Imagini test:

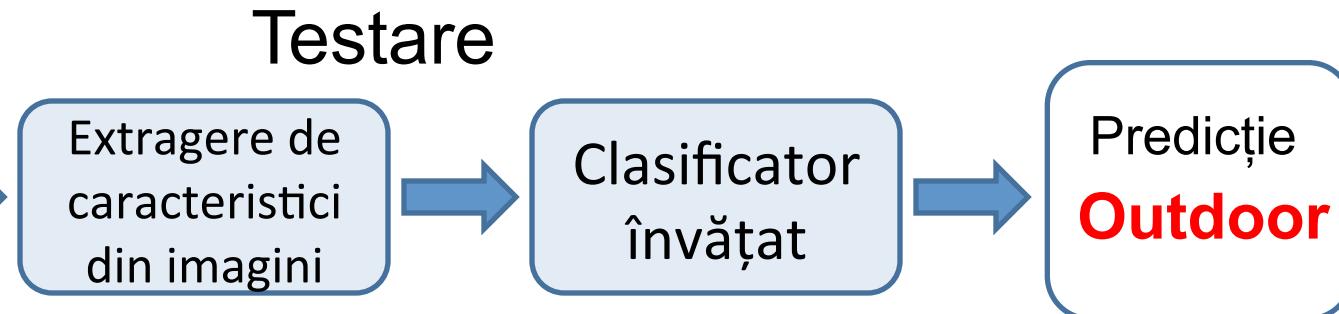
Învățarea supervizată



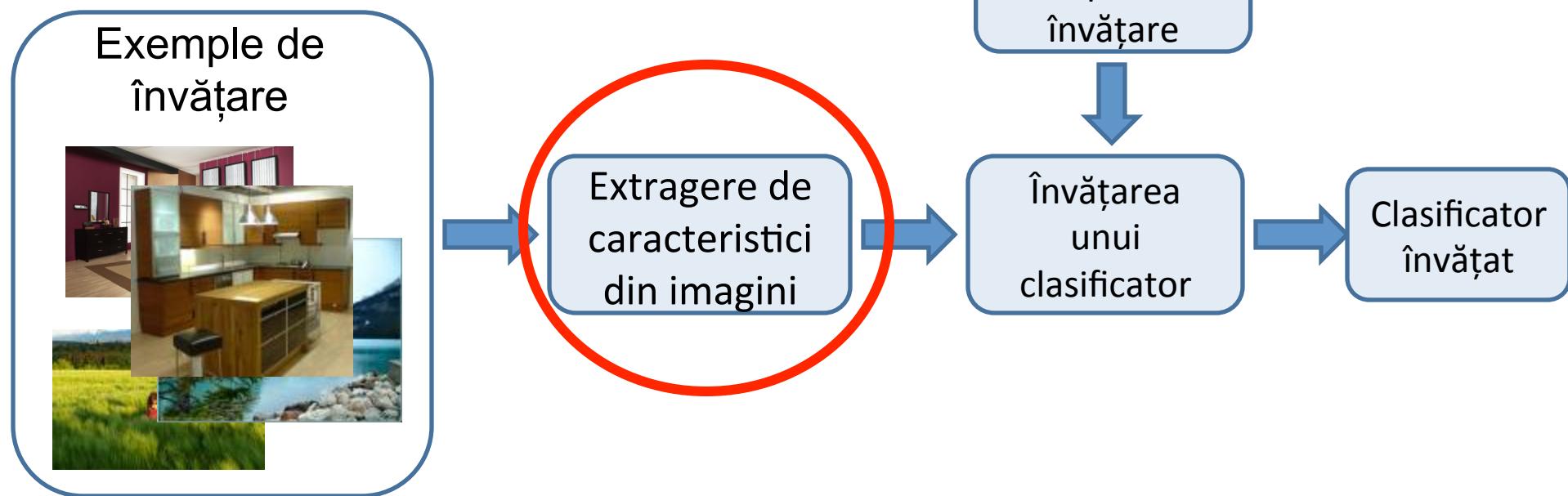
Etapa de testare



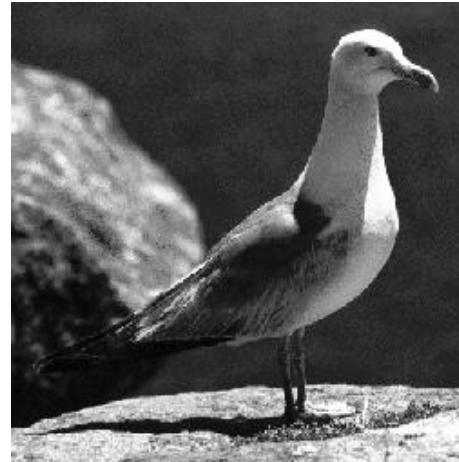
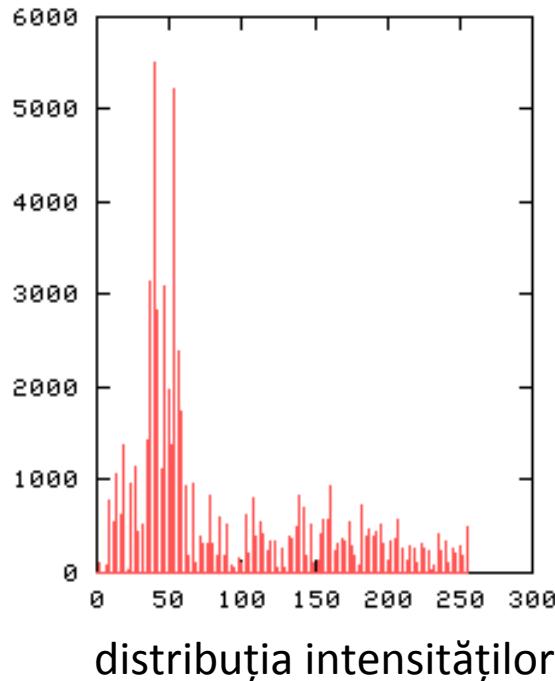
Imagine test



Extragere de caracteristici din imagini



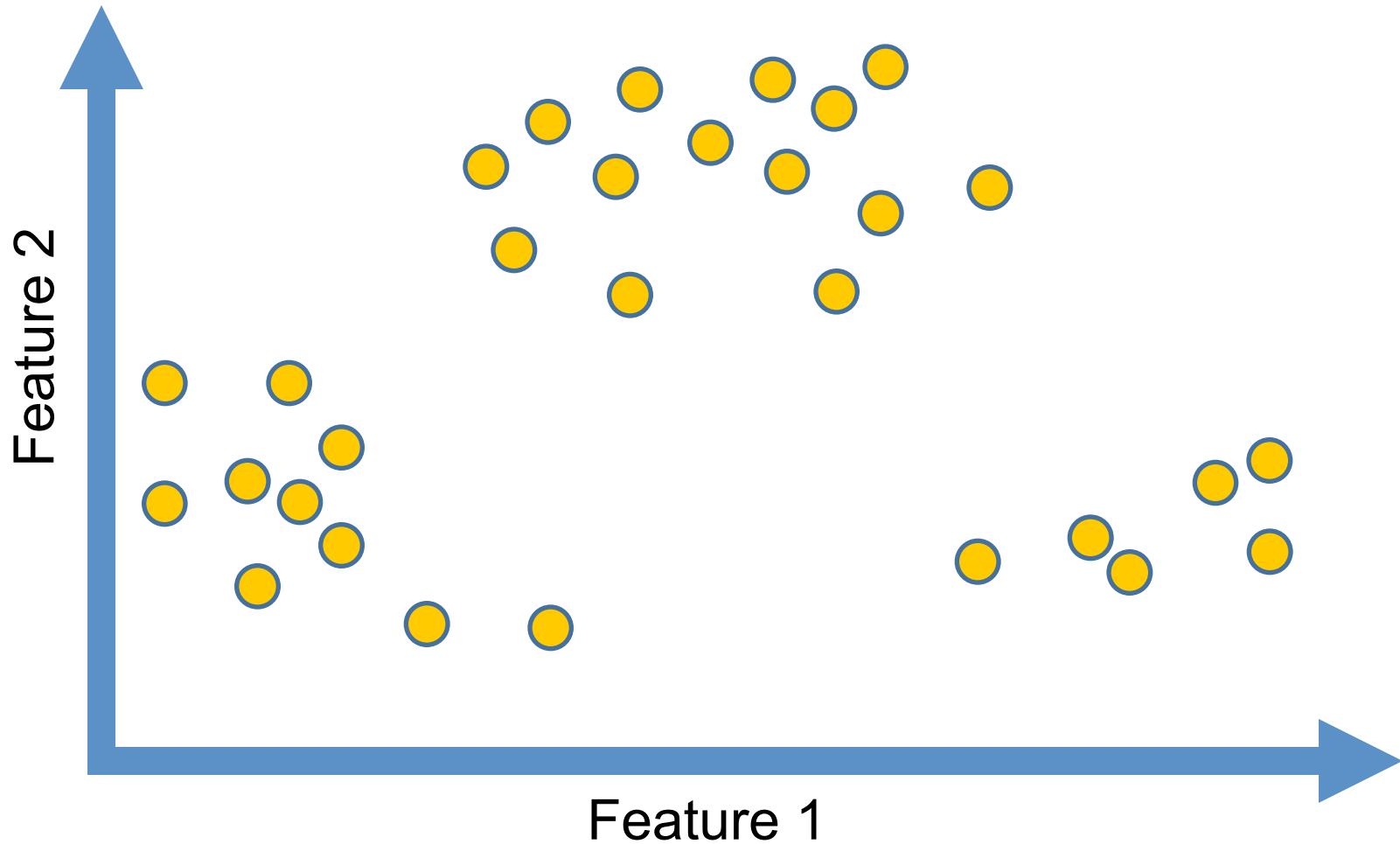
Histograme 1D



Măsoară distribuția caracteristicilor (intensitate, culoare , textură):

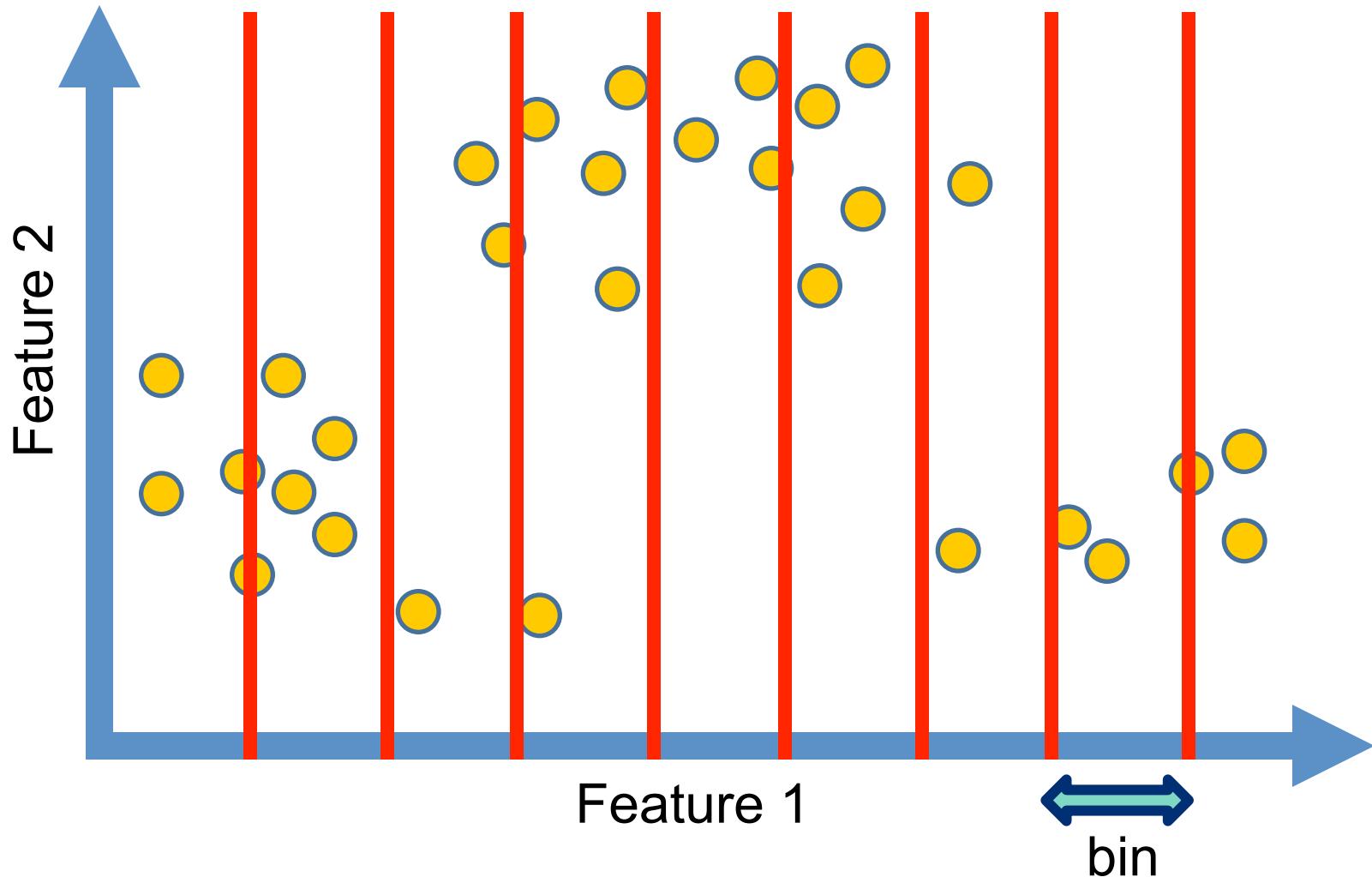
- grupează datele în intervale și numără câte puncte “pică” în fiecare interval
- normalizează histograma (suma elementelor = 1)

Histogramme 2D



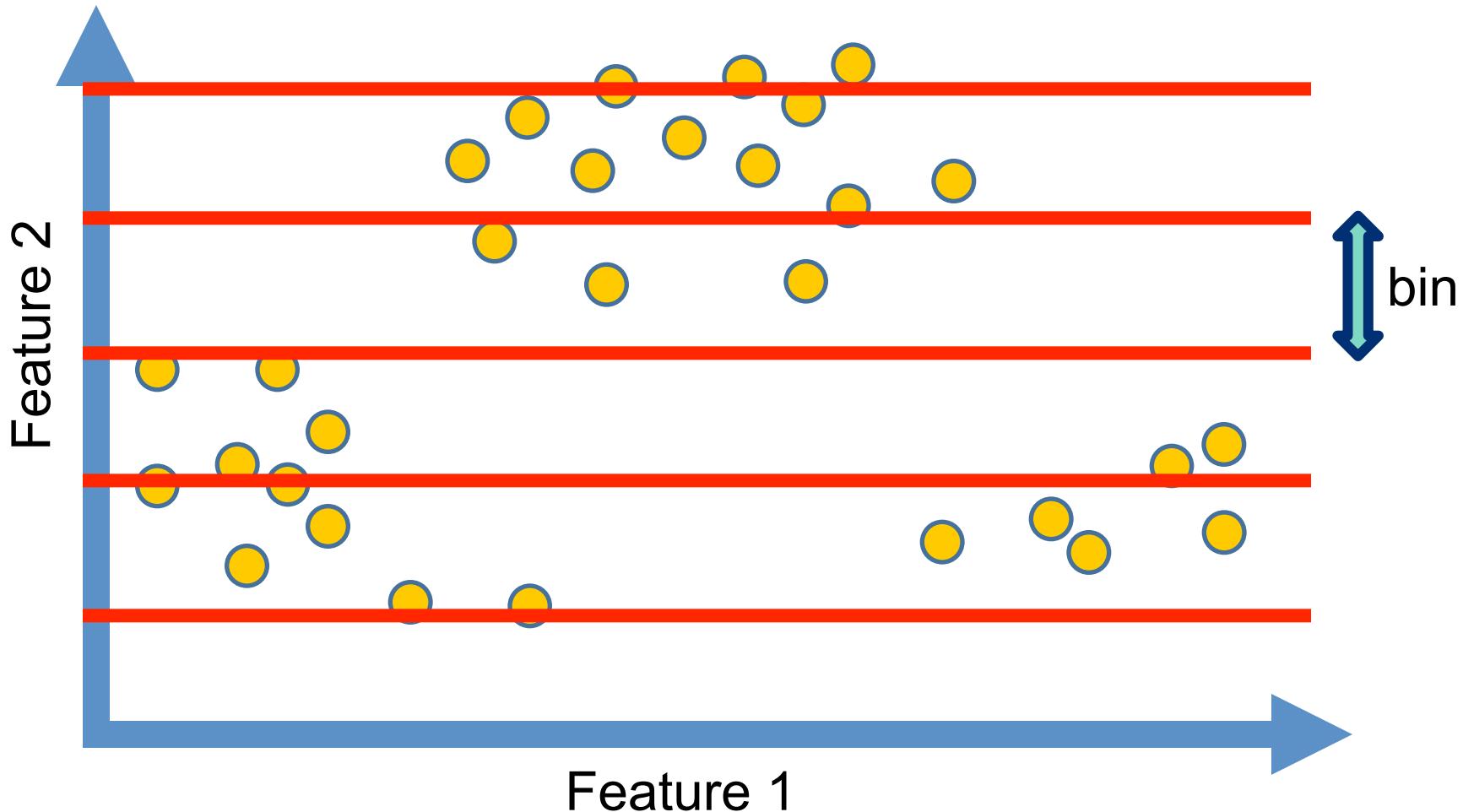
Histogramă individuală

- histogramă marginală pentru caracteristica 1



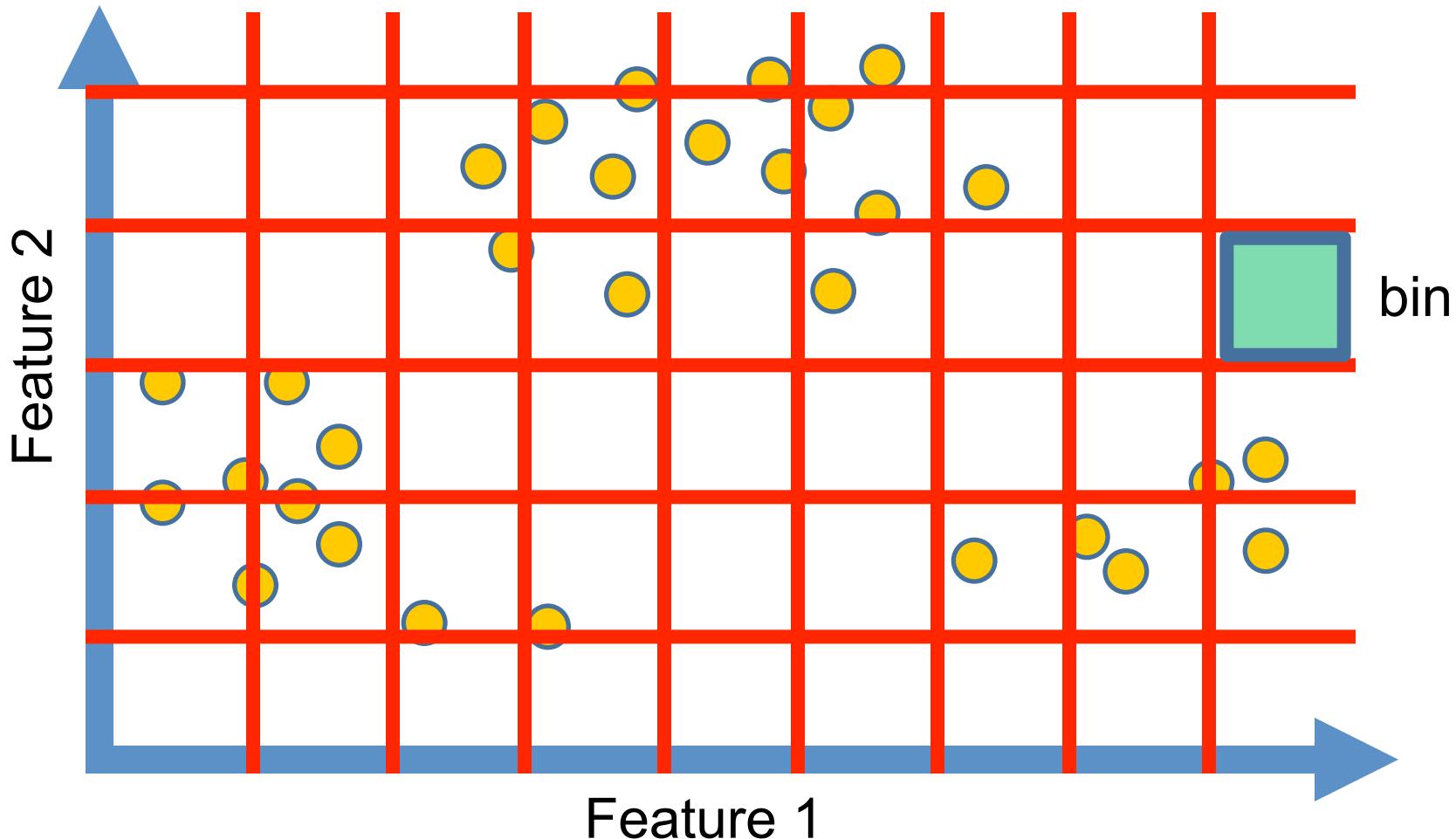
Histogramă individuală

- histogramă marginală pentru caracteristica 2

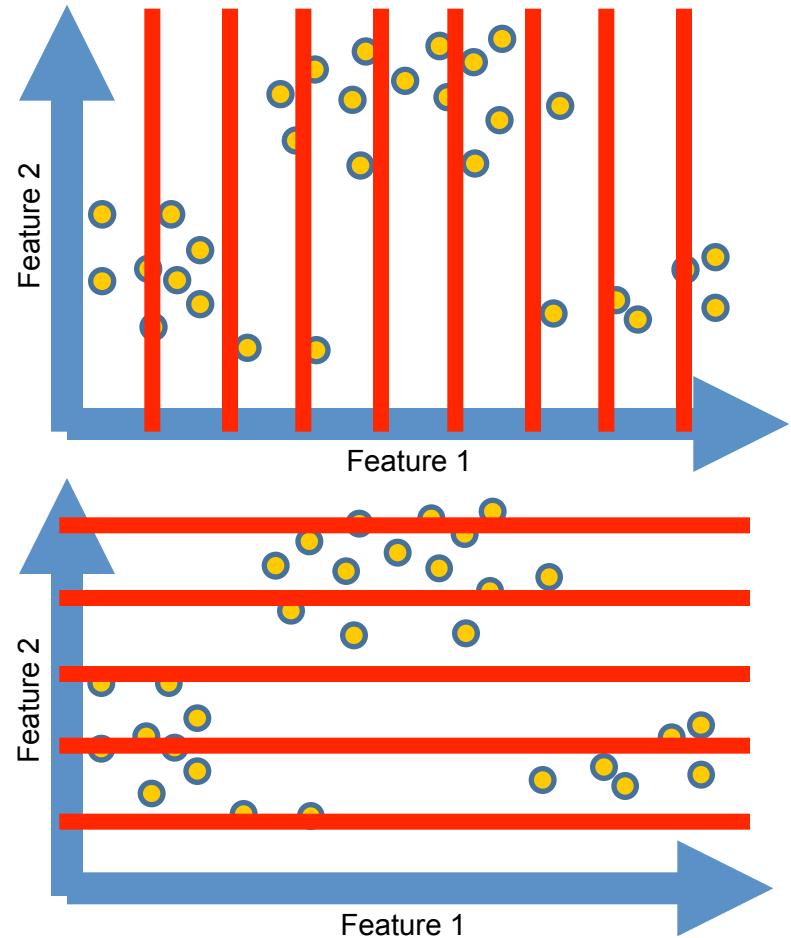
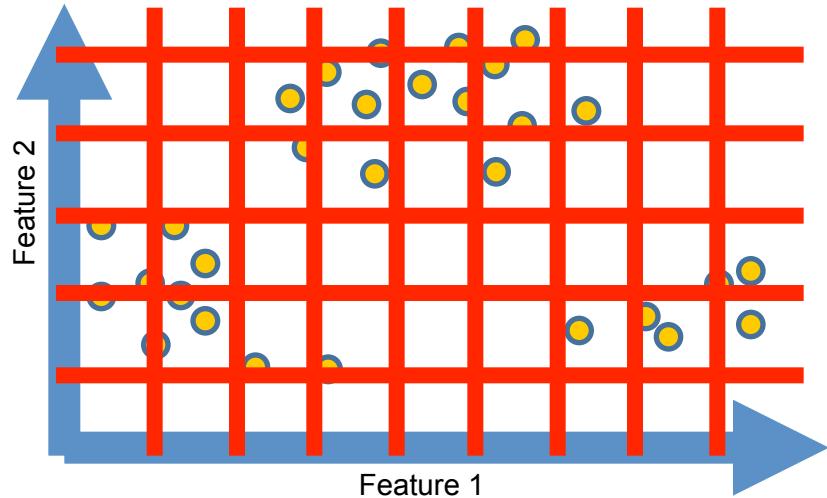


Histograme 2D

- histogramă comună pentru ambele caracteristici



Histograme individuale vs comune



Histograma comună

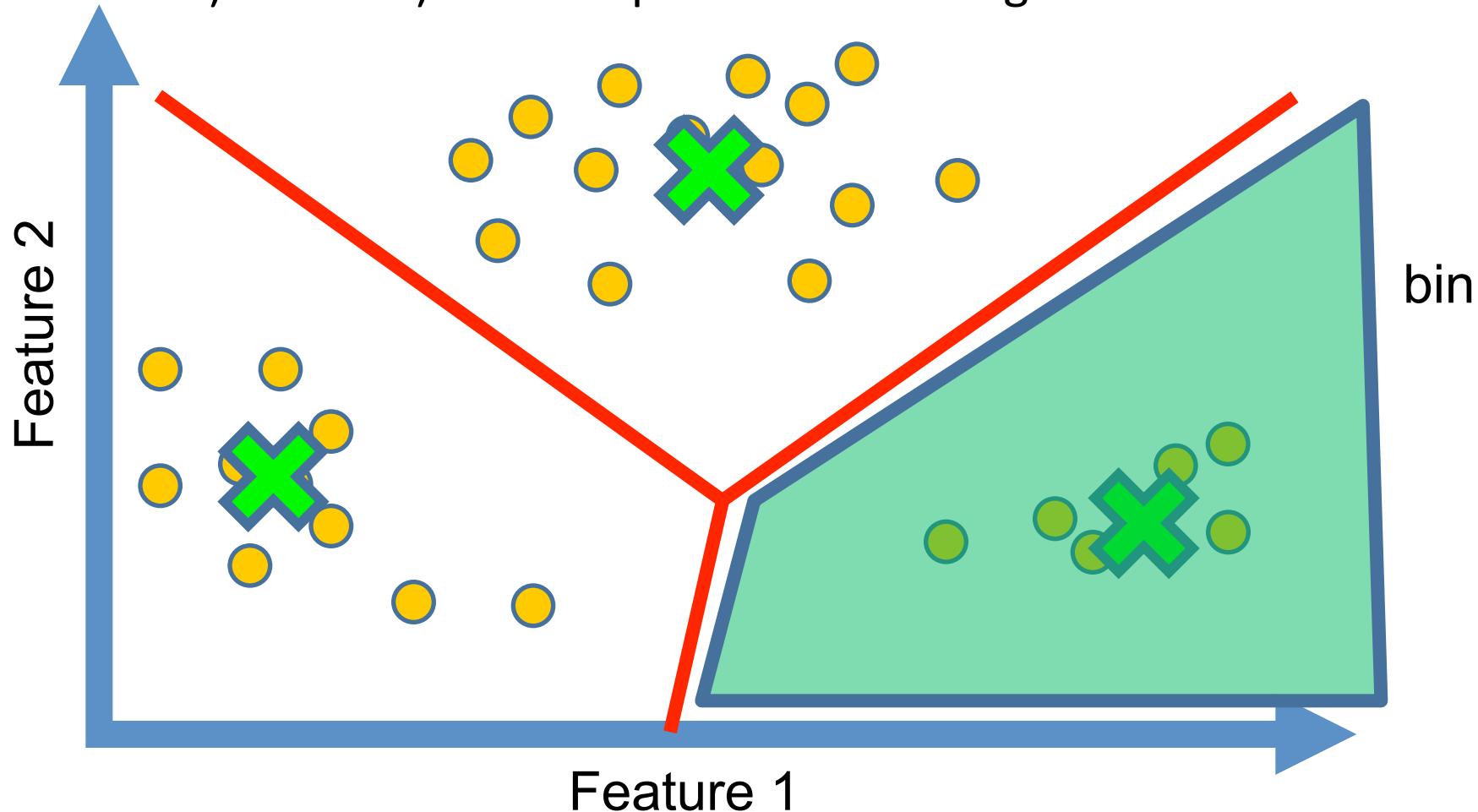
- necesită multe date pentru o aproximare bună a distribuției
- $\#intervale1 * \#intervale2$

Histograma individuală

- necesită caracteristici independente
- mai multe puncte/interval decât în cazul histogramei comune

Histograme pe baza clusterizării

- clusterizează datele în K clustere, numără câte puncte pică în fiecare cluster, obține o histogramă de dimensiune K
- folosește aceeași clusteri pentru toate imaginile



Calculul distanței între histograme

- Distanța Euclidină (L2)

$$d^2(h_i, h_j) = \sum_{m=1}^K (h_i(m) - h_j(m))^2$$

- Distanța Chi-pătrat

$$\chi^2(h_i, h_j) = \frac{1}{2} \sum_{m=1}^K \frac{(h_i(m) - h_j(m))^2}{h_i(m) + h_j(m)}$$

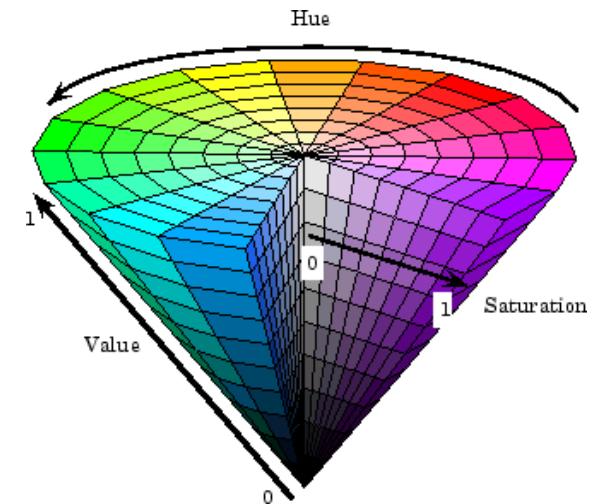
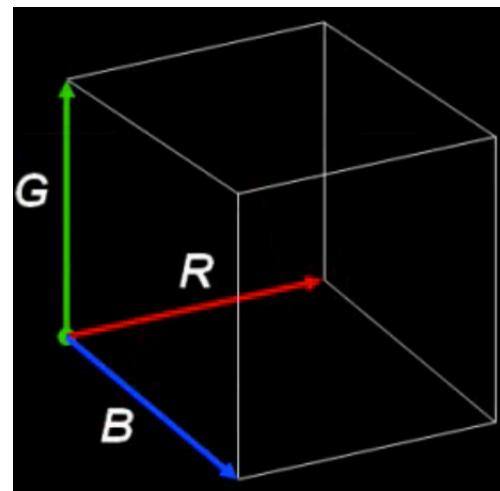
- Distanța dată de intersecția histogramelor

$$histint(h_i, h_j) = 1 - \sum_{m=1}^K \min(h_i(m), h_j(m))$$

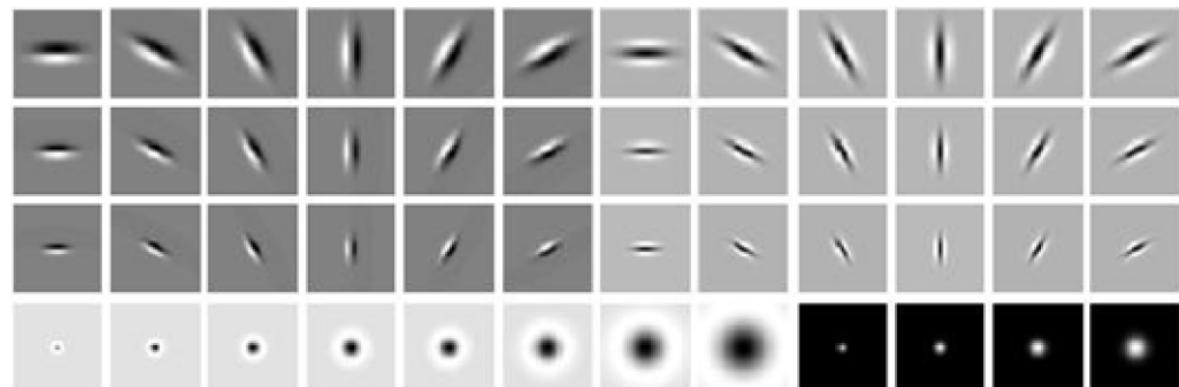
- Distanța cosinus, distanța Earth-mover's

Ce caracteristici folosim pentru histograme?

- Culoare

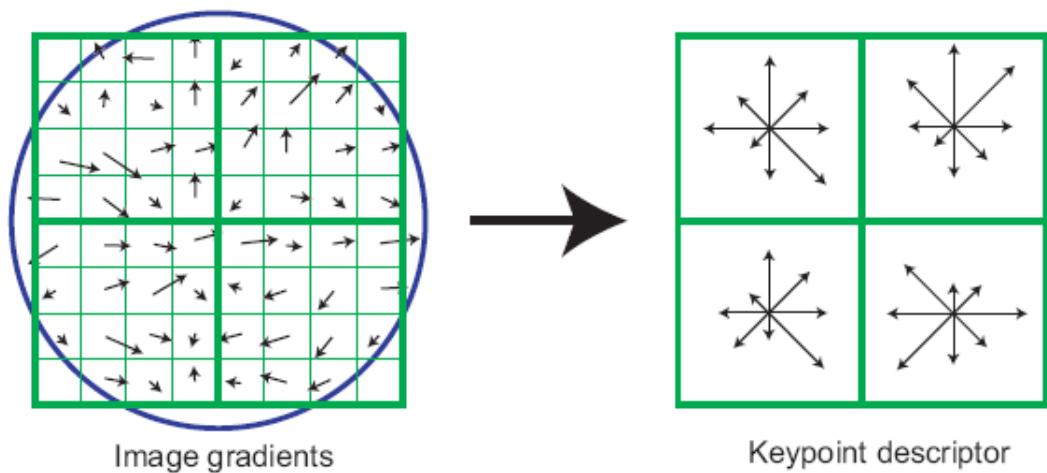
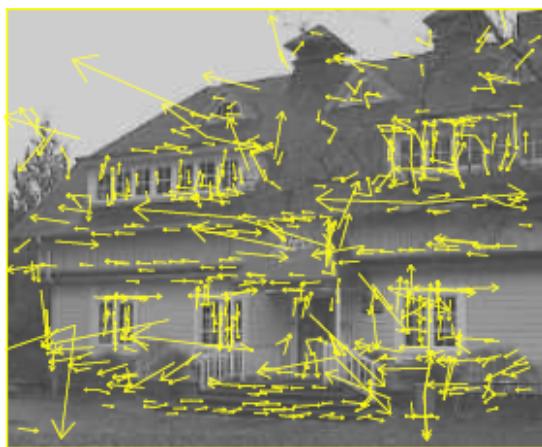


- Textură



Ce caracteristici folosim pentru histograme?

- Descriptori vizuali



SIFT – [Lowe IJCV 2004]

- Modelul “bag of visual words”

Clasificarea imaginilor cu modelul bag of visual words

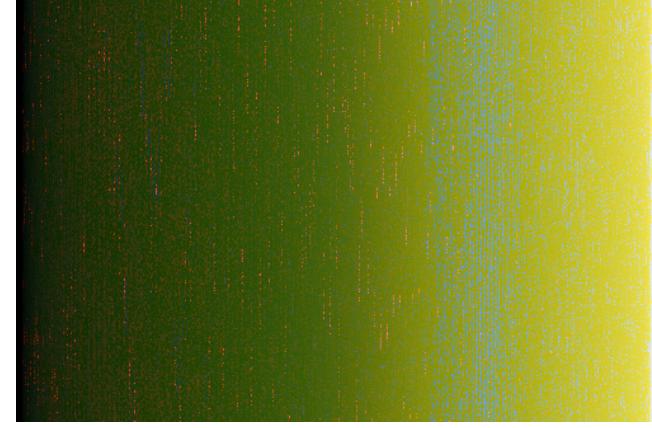
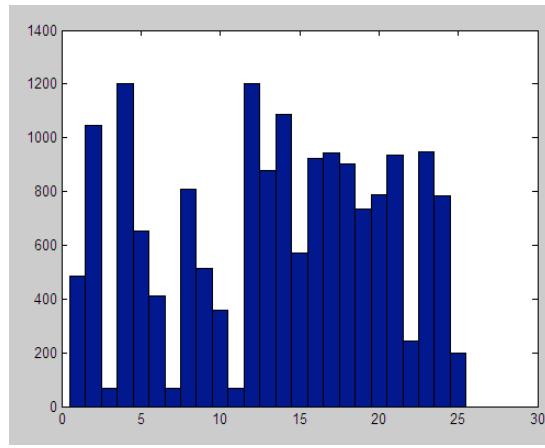
Învățare

1. Extrage caracteristici (descriptori) pentru toate imaginile de învățare
2. Clusterizează descriptorii, centri clusterilor = visual words = ‘cuvinte vizuale’
3. Asignează fiecarui descriptor eticheta/index-ul unui ‘cuvânt vizual’
4. Reprezintă fiecare imagine printr-o histogramă normalizată de ‘cuvinte vizuale’
5. Învață un clasificator pe exemplele de învățare folosind histogramele normalize

Testare

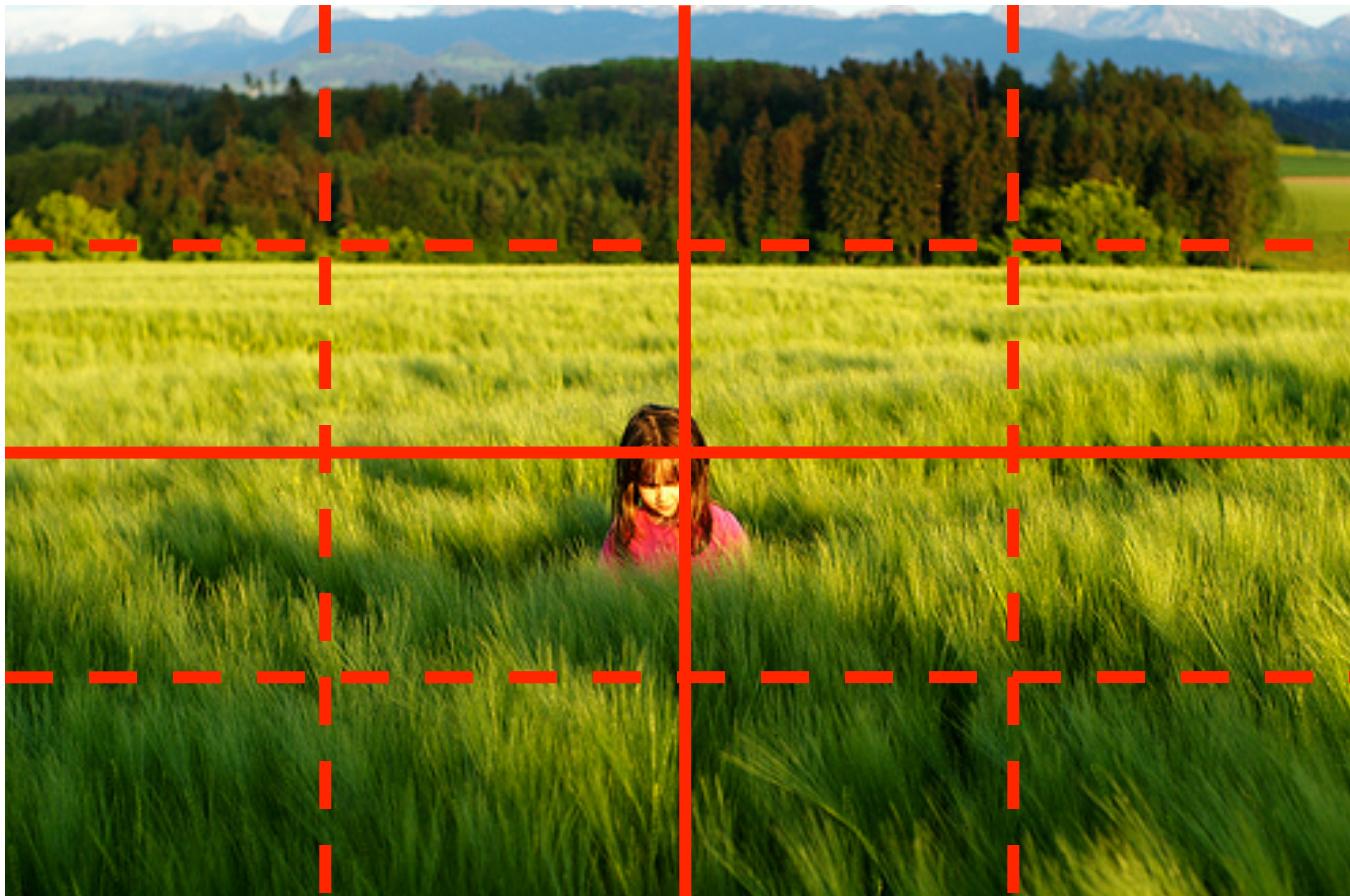
1. Extrage descriptori din imaginea test, asignează fiecarui descriptor eticheta unui ‘cuvânt vizual’
2. Calculează histograma normalizată de ‘cuvinte vizuale’
3. Calculează eticheta corespunzătoare imaginii test sau un scor folosind clasificatorul învățat.

Poziția în imagine a caracteristicilor



Cele trei imagini au aceeași histogramă de culori.

Piramidă spațială



Calculează histograma pentru fiecare subimagine

Exemplu: Clasificator BoW bazat pe piramidă spațială



Figure 4. Retrieval from the scene category database. The query images are on the left, and the eight images giving the highest values of the spatial pyramid kernel (for $L = 2, M = 200$) are on the right. The actual class of incorrectly retrieved images is listed below them.

- Histograme de descriptori SIFT clusterizați + SVM

Recunoașterea claselor de obiecte

Recunoașterea claselor de obiecte la nivel de instanță



mașina lui Mihai

Recunoașterea claselor de obiecte la nivel generic



Vrem să recunoștem toate instanțierile unei clase de obiecte

Recunoașterea claselor de obiecte

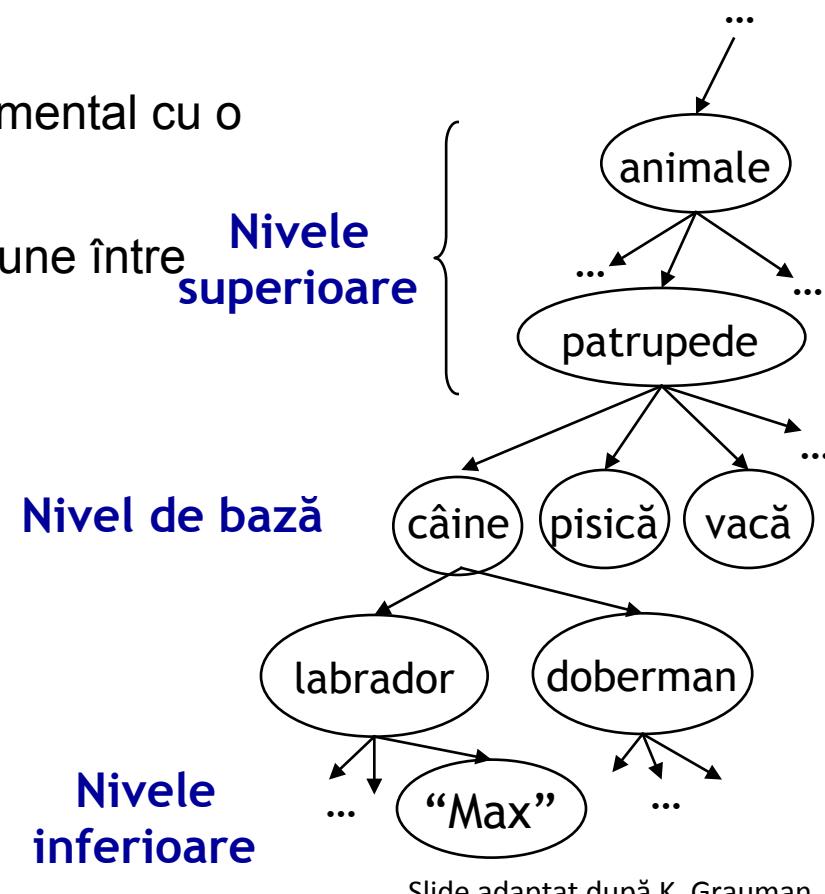
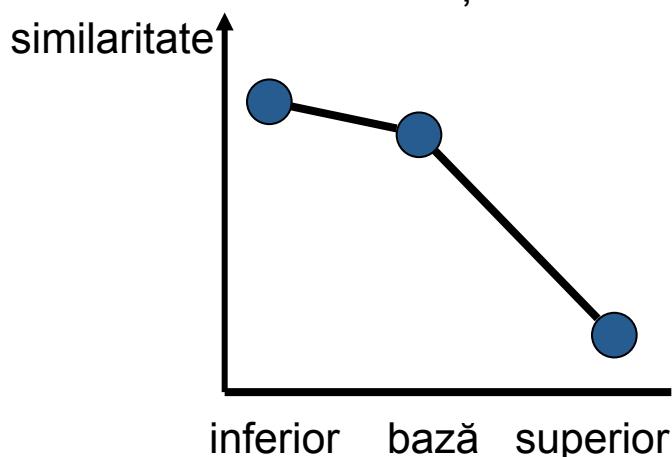
- Descrierea problemei:
 - “Fiind dat un număr mic de imagini de antrenare ale unei clase de obiecte, recunoașteți instanțieri ale clasei de obiecte asignând eticheta corespunzătoare clasei de obiecte.”
- Pentru care clasă de obiecte se poate asocia repede o imagine?



Organizarea ierarhică a claselor de obiecte (Rosch, 1976)

Nivelul de bază:

- clasele de obiecte la nivel de bază sunt clasele de nivel cel mai înalt pentru care membrii lor au forme percepute similare.
- nivelul cel mai înalt pentru care reprezentăm mental cu o singură imagine întreaga clasă de obiecte
- există un număr seminificativ de atribute comune între perechi de membri
- primul nivel numit și înțeles de copii

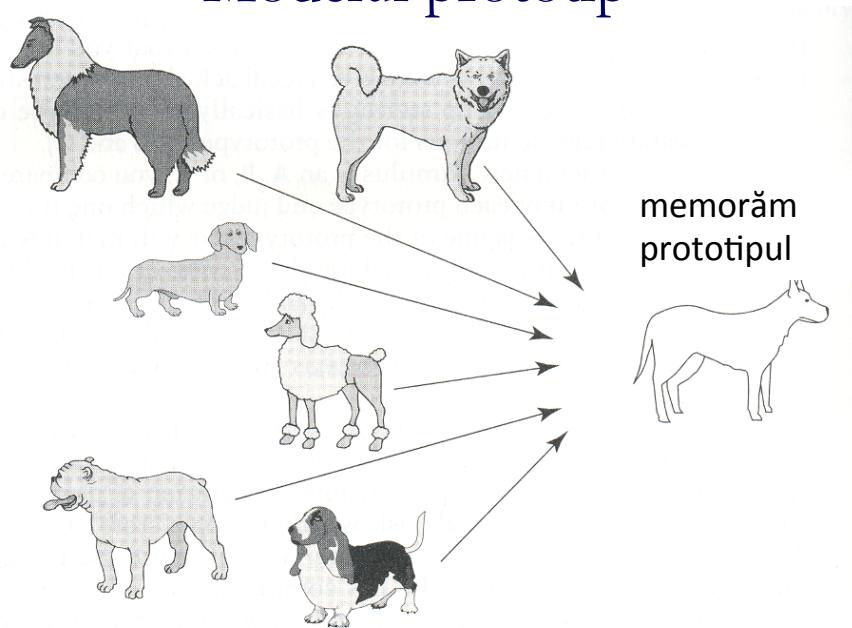


Organizarea ierarhică a claselor de obiecte (Rosch, 1976)

- Rosch a găsit următoarele:
 - oamenii clasifică clasele de obiecte mult mai repede la nivel de bază
 - tendință de a clasifica după nivelul de bază ('câine') înainte de nivelul superior ('animal') sau nivelul inferior ("golden retriever")
 - mai întâi clasificare la nivel de bază iar apoi recunoaștere
 - acord la nivel de bază (de cele mai multe ori)
 - desktop, laptop, tablete, pc, mac – nivel de bază
 - calculator – nivel de bază
 - desktop, laptop, tablete, pc, mac – nivel inferior
 - ține de nivelul de expertiză

Recunoașterea claselor de obiecte

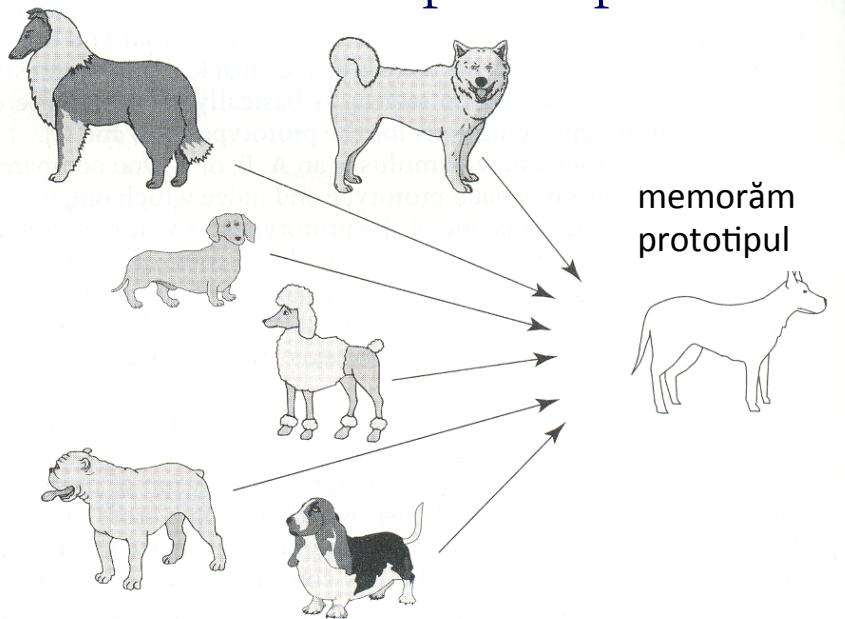
Modelul prototip



Realizăm clasificarea într-o clasă de obiecte prin compararea unui nou exemplu cu prototipul

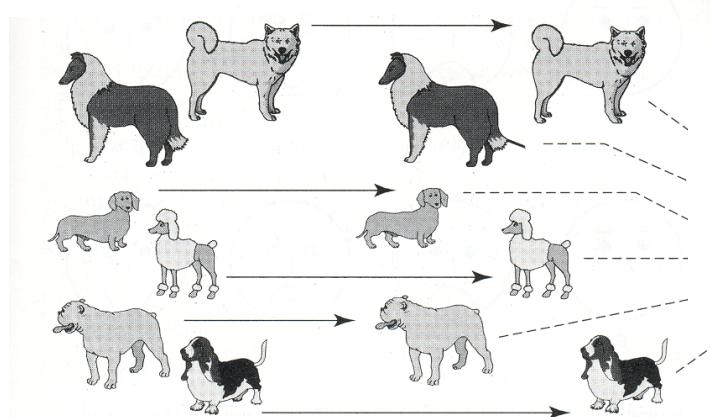
Recunoașterea claselor de obiecte

Modelul prototip



Realizăm clasificarea într-o clasă de obiecte prin compararea unui nou exemplu cu prototipul

Modelul cu exemple particulare

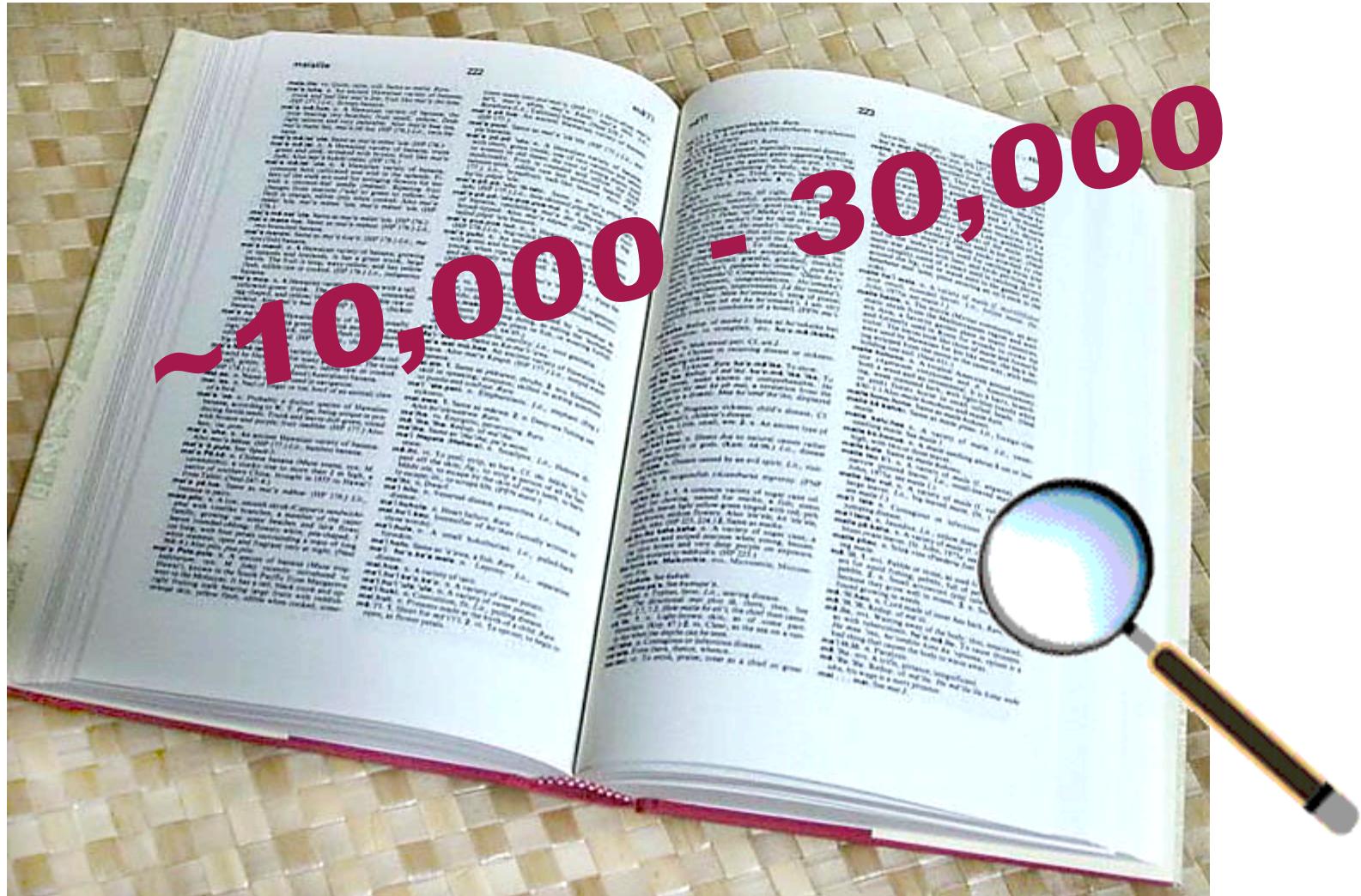


Realizăm clasificarea într-o clasă de obiecte prin compararea unui nou exemplu cu toate exemplele particulare memorate

Clase de obiecte - exemple

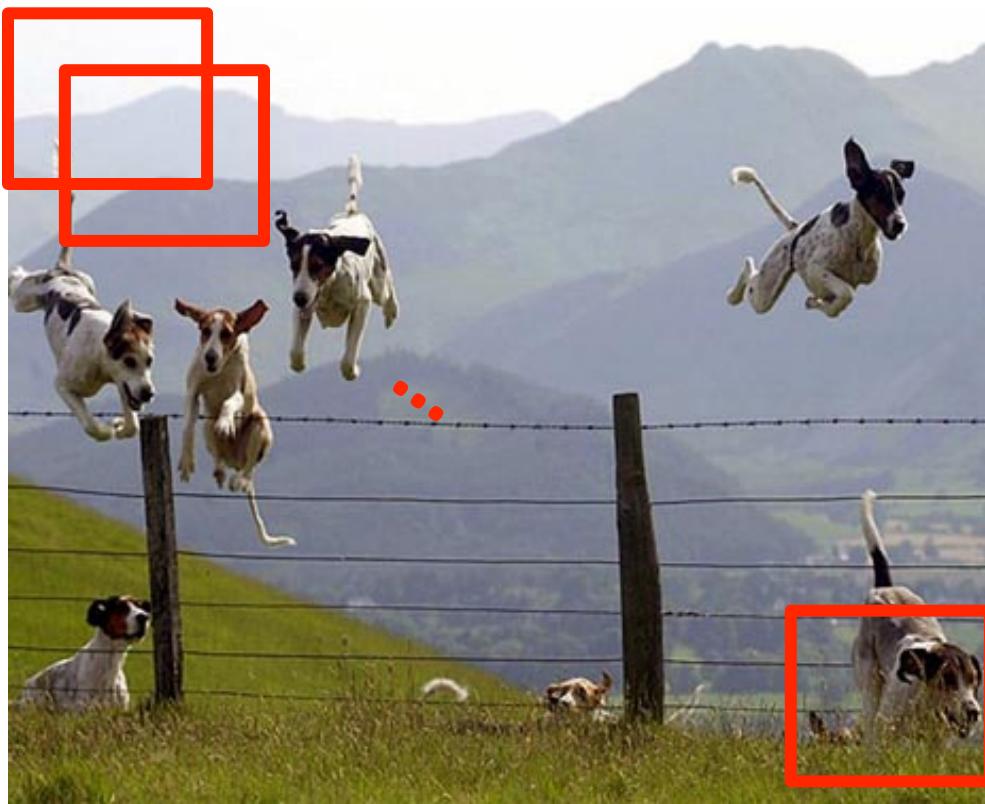


Cât de multe clase de obiecte există?

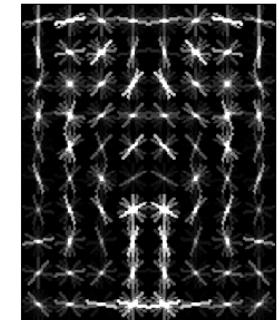


Detectarea (recunoașterea) claselor de obiecte

- localizarea instanțelor unei clase de obiecte: “Unde este?”
- modele discriminative care clasifică obiecte vs background la nivel de patch (fereastră)



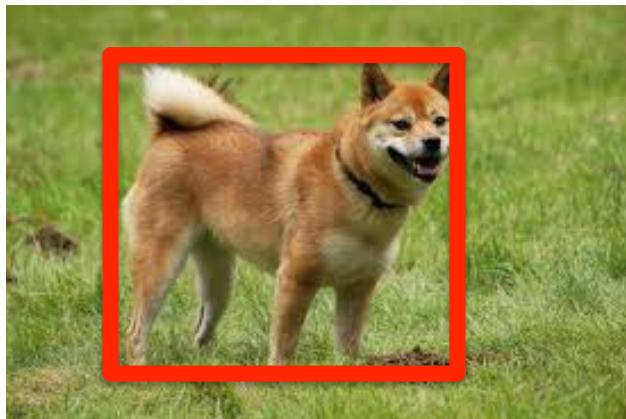
Dog Model



**Object or
Non-Object?**

Recunoașterea claselor de obiecte la nivel generic

- Recunoaște orice câine în imagine



Recunoașterea claselor de obiecte la nivel generic

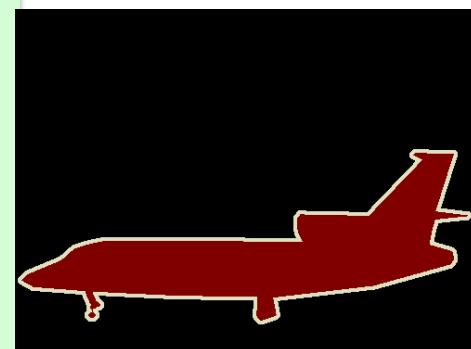
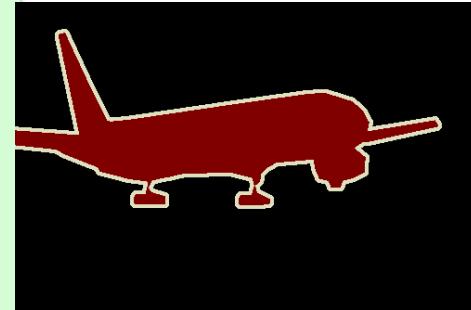
imagini inițiale



localizare - fereastră



localizare - contur



Dificultăți în modelarea claselor de obiecte



Iluminare



Mărime



Mascarea
obiectului



Variabilitate
înfățisare intra-
clasă



Poziționarea
camerei

Detectarea de obiecte – rețetă generală

Model al clasei de obiecte

Care sunt parametri modelului clasei de obiecte?



Generează ipoteze



Asignează un scor ipotezelor



Obține detectii

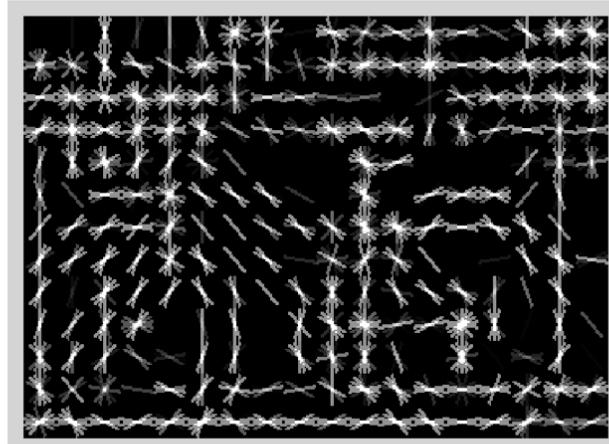
Specificarea unui model pentru o clasă de obiecte

1. Template de caracteristici la nivel de fereastră

- un obiect este modelat de o fereastră (x,y,w,h)
- caracteristici definite relativ la coordonatele ferestrei



Imagine

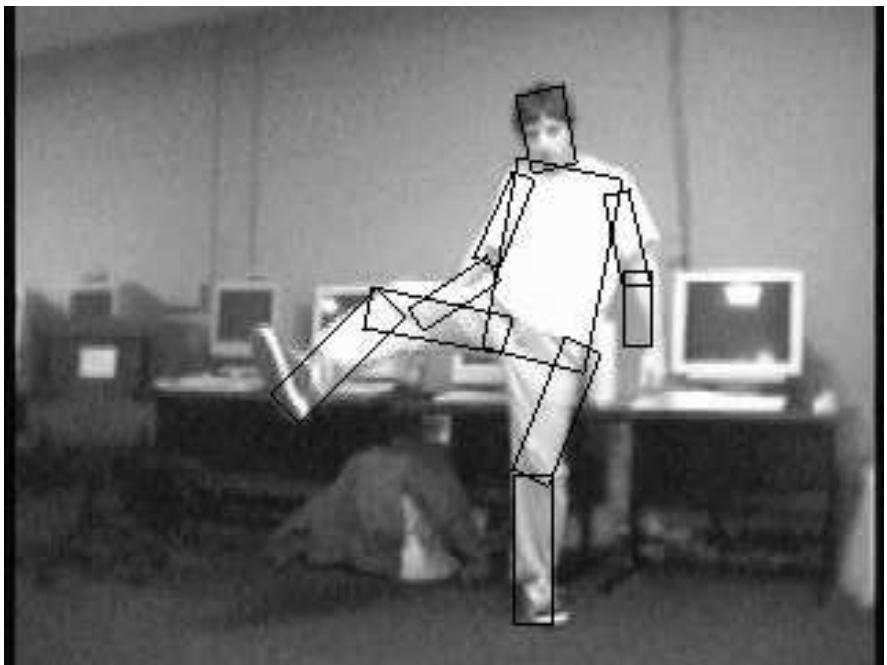
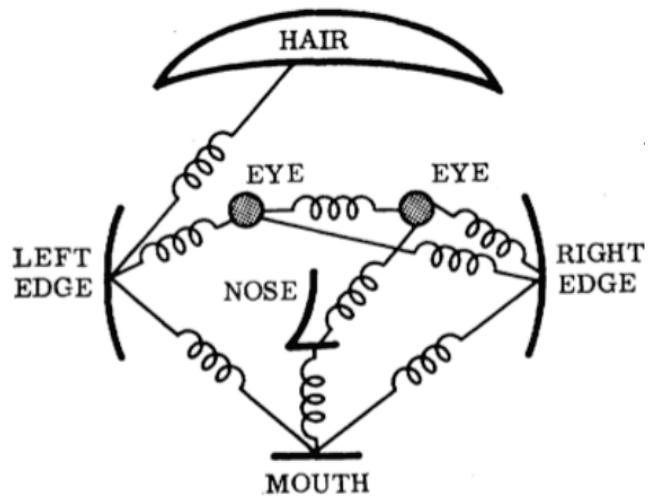


Vizualizare a template-ului

Specificarea unui model pentru o clasă de obiecte

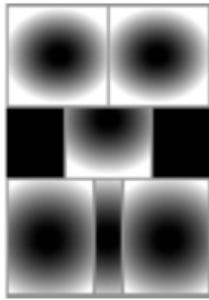
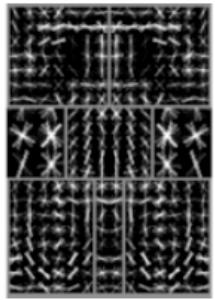
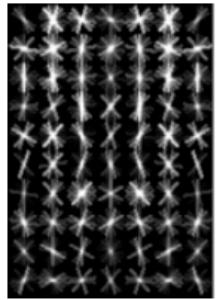
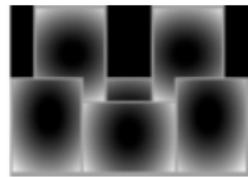
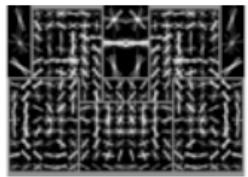
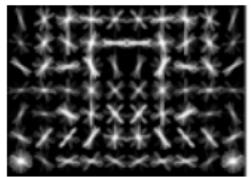
2. Model de părți articulate

- un obiect este o configurație de părți
- fiecare parte poate fi detectată



Specificarea unui model pentru o clasă de obiecte

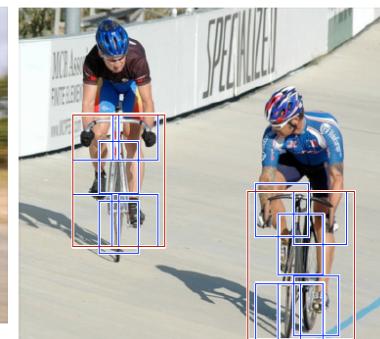
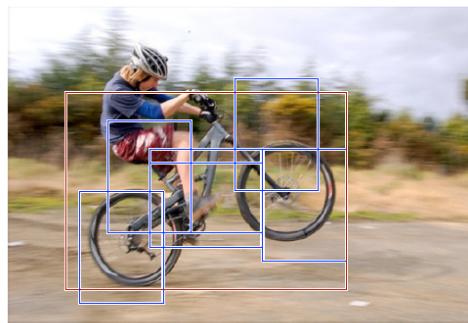
3. Model hidrid template/părți



root filters
coarse resolution

part filters
finer resolution

deformation
models



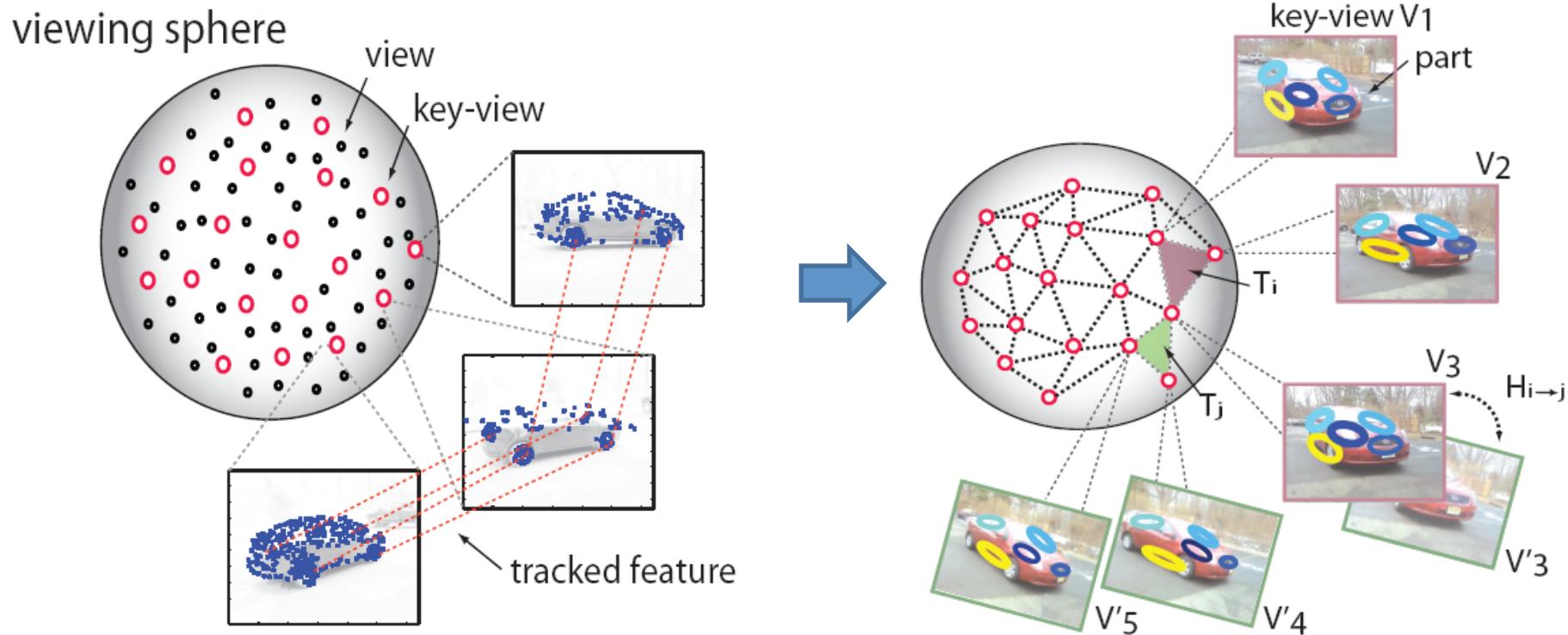
Detectii

Vizualizare template-uri

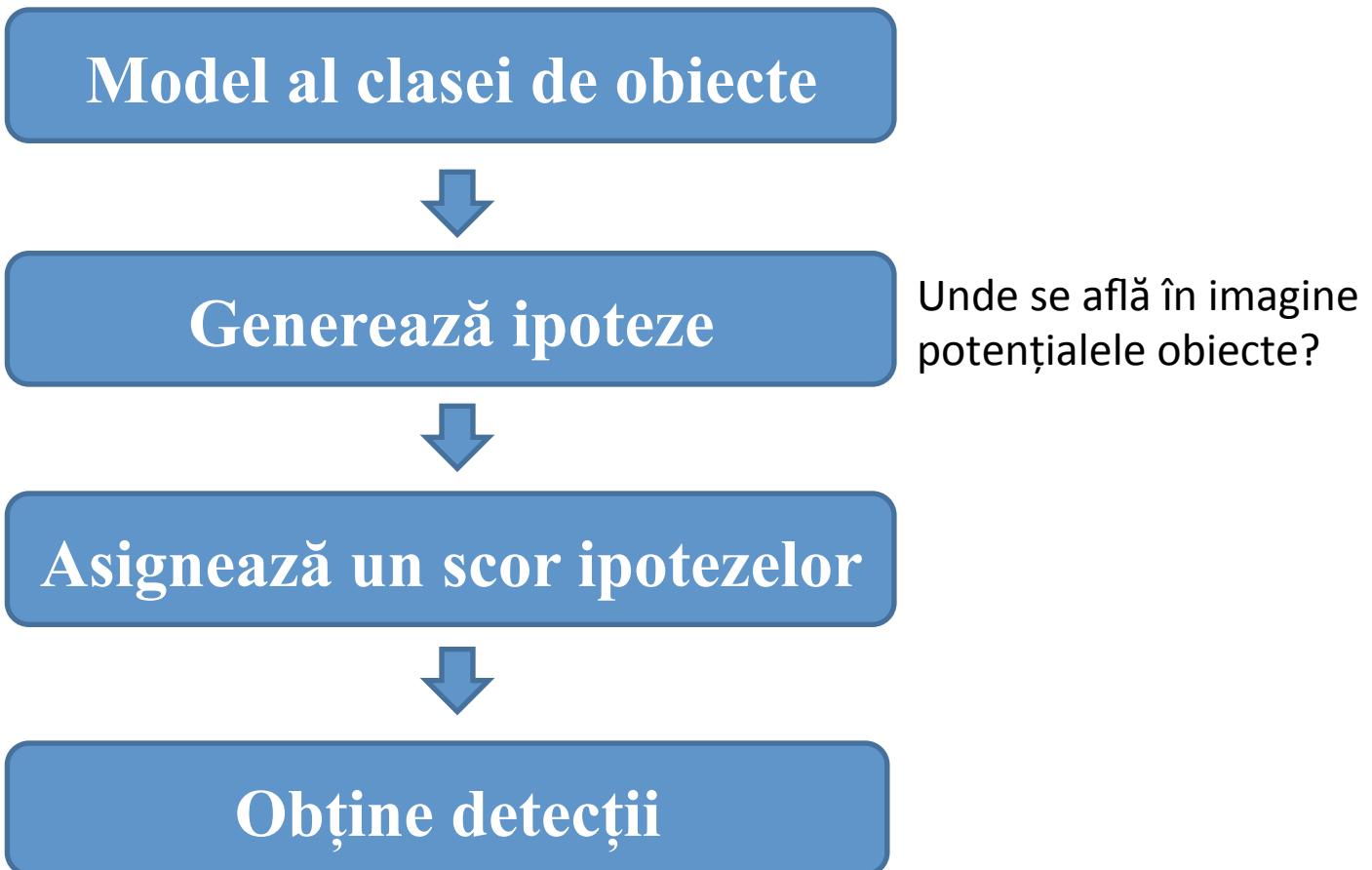
Specificarea unui model pentru o clasă de obiecte

3. Model 3D

- un obiect este o colecție de părți în 3D care se proiectează în 2D în planul imaginii



Detectarea de obiecte – rețetă generală



Generarea ipotezelor

1. Glisarea unei ferestre (sliding window)
 - clasificarea unei ferestre la fiecare poziție și mărime



Generarea ipotezelor

1. Glisarea unei ferestre (sliding window)
 - clasificarea unei ferestre la fiecare poziție și mărime



Generarea ipotezelor

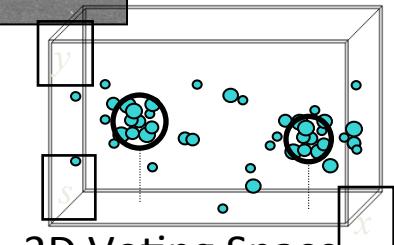
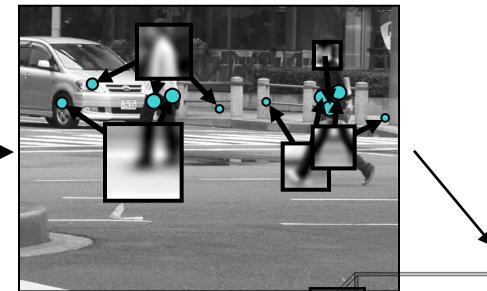
2. Voturi acumulate de la patch-uri/puncte de interes



Cuvinte vizuale asociate descriptorilor punctelor de interes

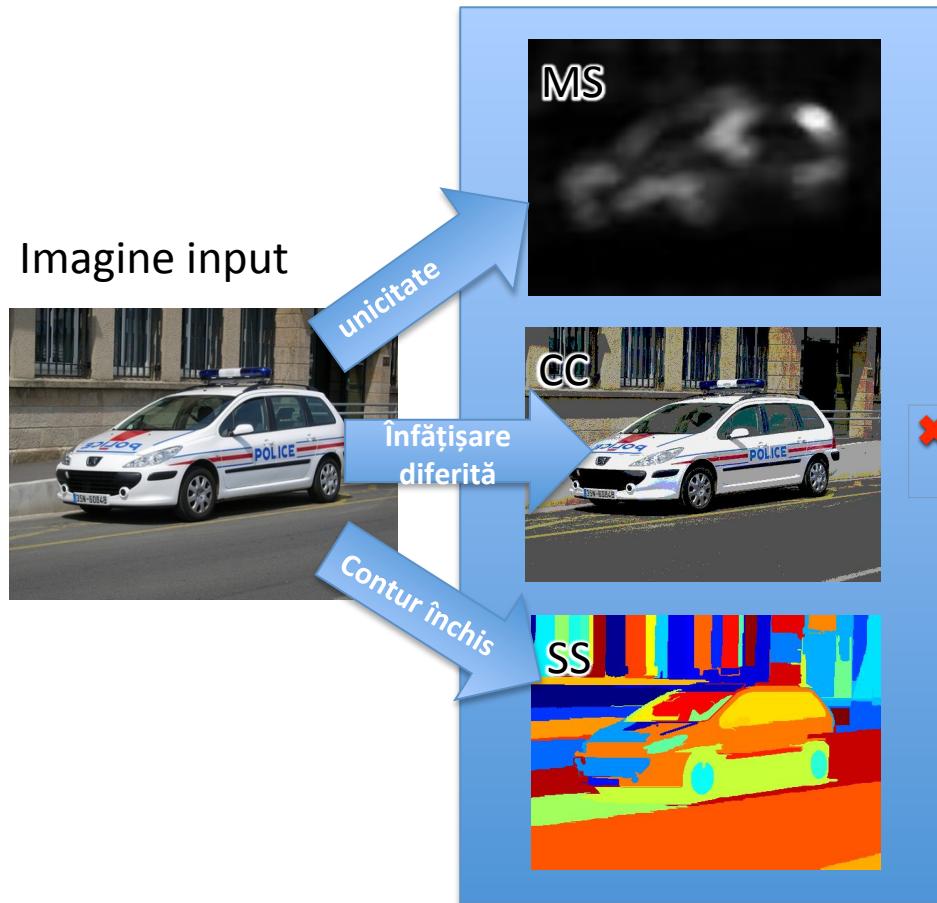


Acumulare de voturi

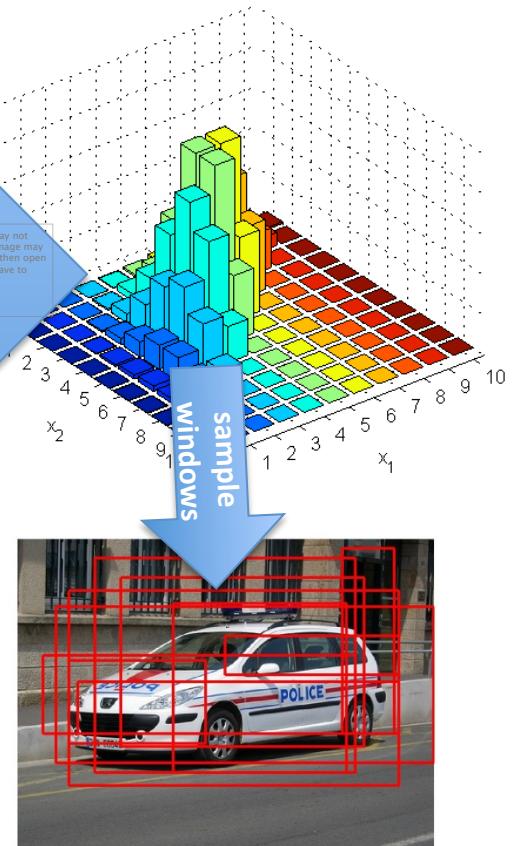


Generarea ipotezelor

3. Propuneri candidat la nivel de fereastă

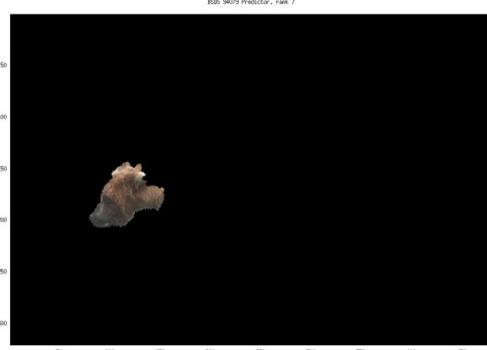
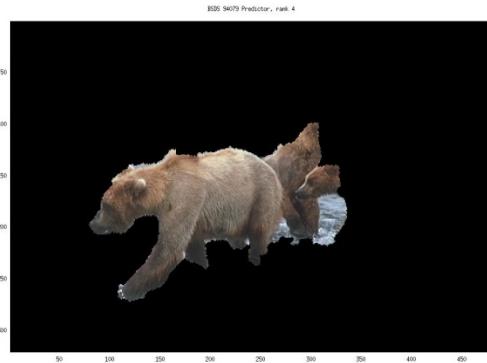


Output – distribuție obiectness

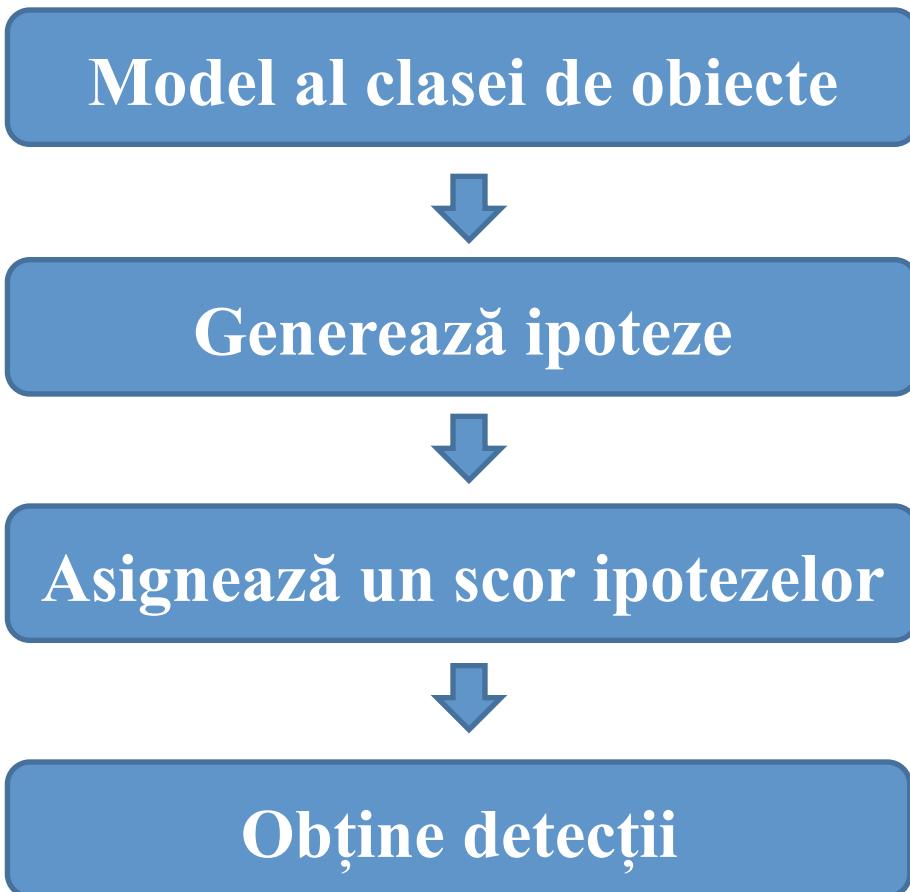


Generarea ipotezelor

3. Propuneri candidat la nivel de regiune

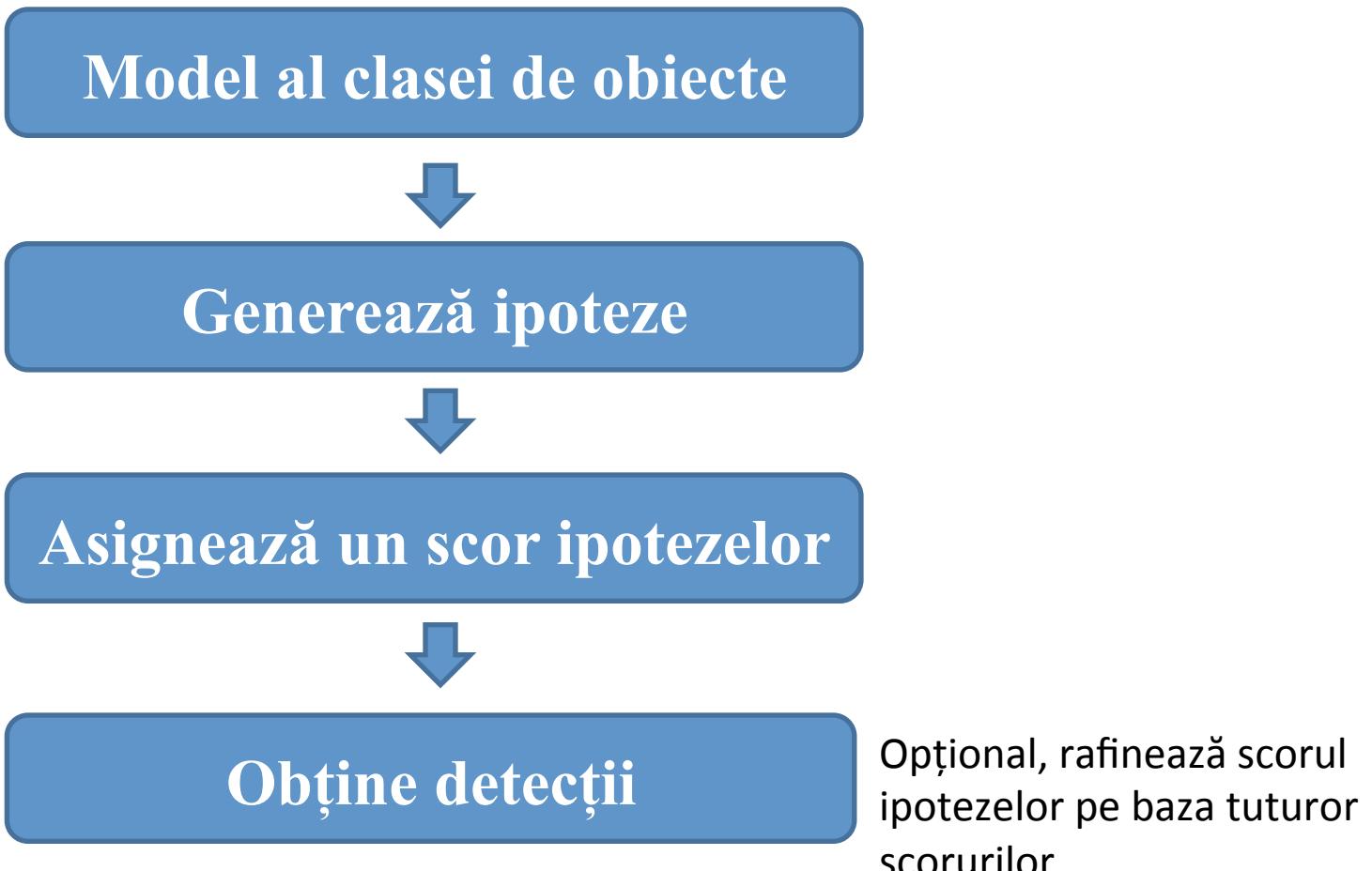


Detectarea de obiecte – rețetă generală



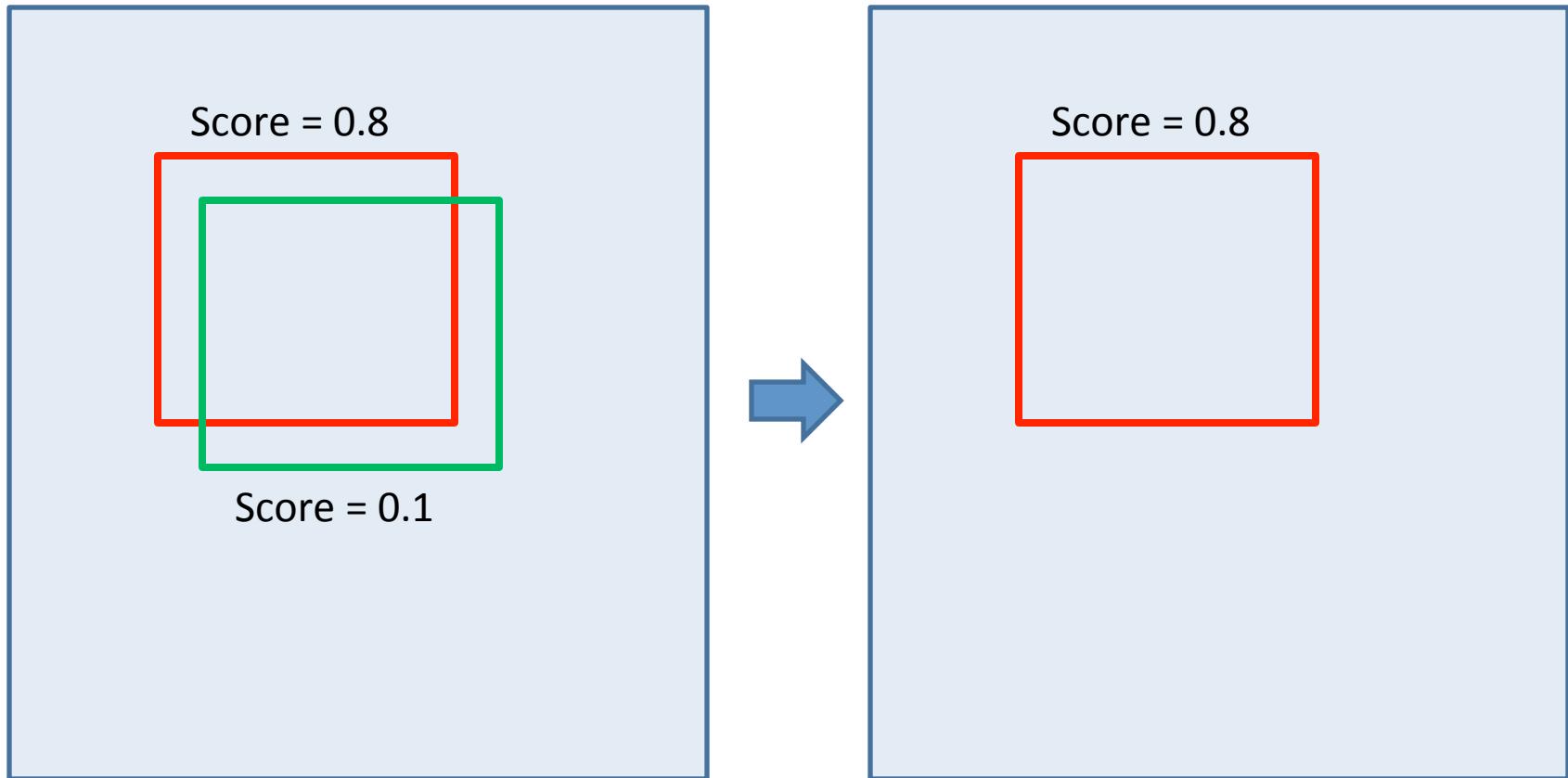
Clasificatori aplicați pe
reprezentări de caracterstici

Detectarea de obiecte – rețetă generală



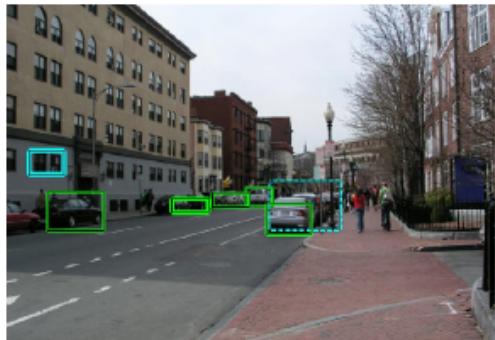
Obținerea detecțiilor

1. Suprimarea non-maximelor



Obținerea detectiilor

2. Folosirea contextului pentru eliminarea detectiilor fals pozitive



(g) Car Detections: Local



(h) Ped Detections: Local

