Advanced Computer Vision

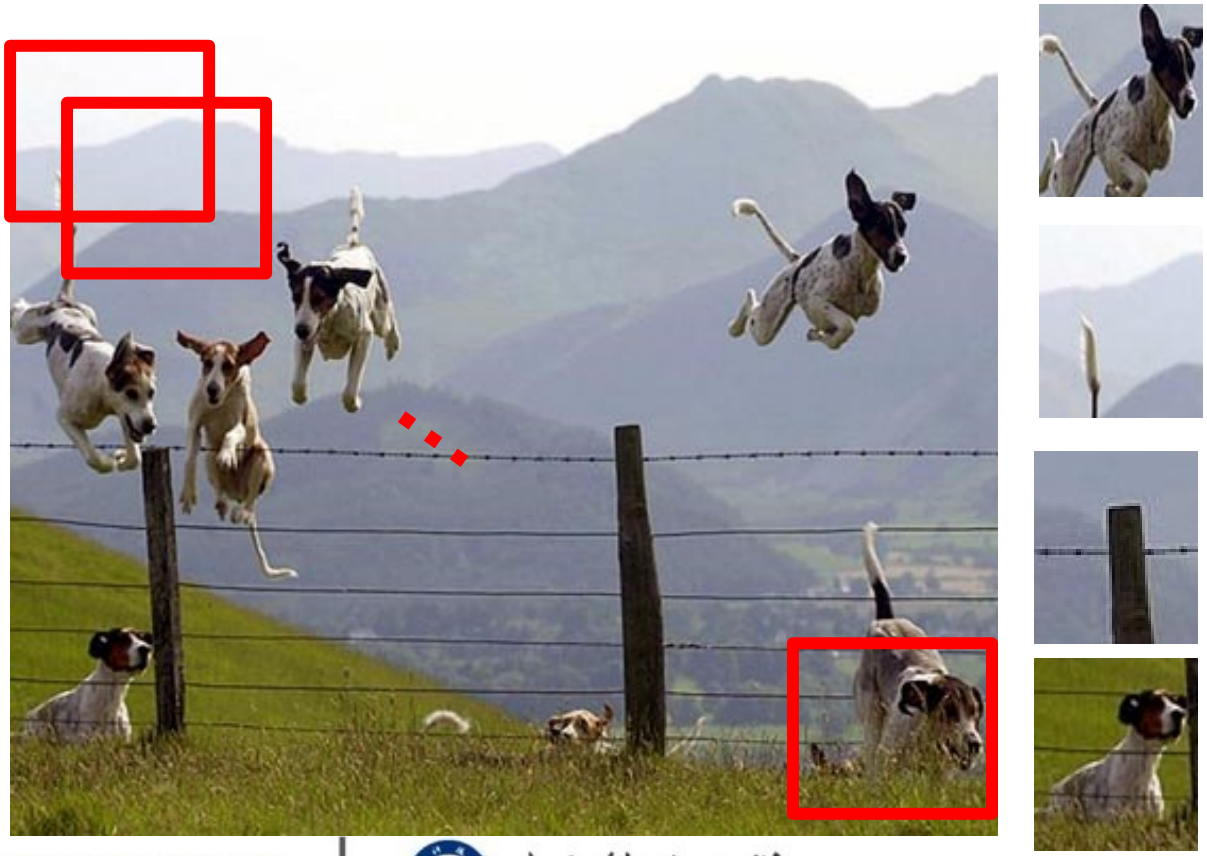
Detection with Sliding Windows

Object detection vs. Scene Recognition

- Scenes can be defined by distribution of “stuff” – materials and surfaces with arbitrary shape.
- Objects are “things” that own their boundaries
- Bag of words models are less popular for object detection because they throw away shape info.

Object Category Detection

- Focus on object search: “Where is it?”
- Build templates that quickly differentiate object patch from background patch



Object or
Non-Object?

Challenges in modeling the object class

Illumination

Object pose

'Clutter'

Occlusions

Intra-class appearance

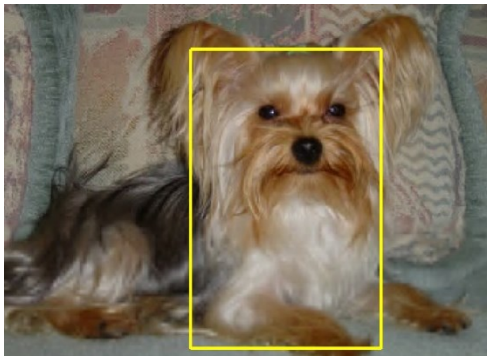
Viewpoint

Slide from K. Grauman, B. Leibe

Challenges in modeling the non-object class

True Detections

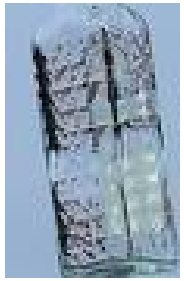
Bad Localization



Confused with Similar Object

Misc. Background

Confused with Dissimilar Objects



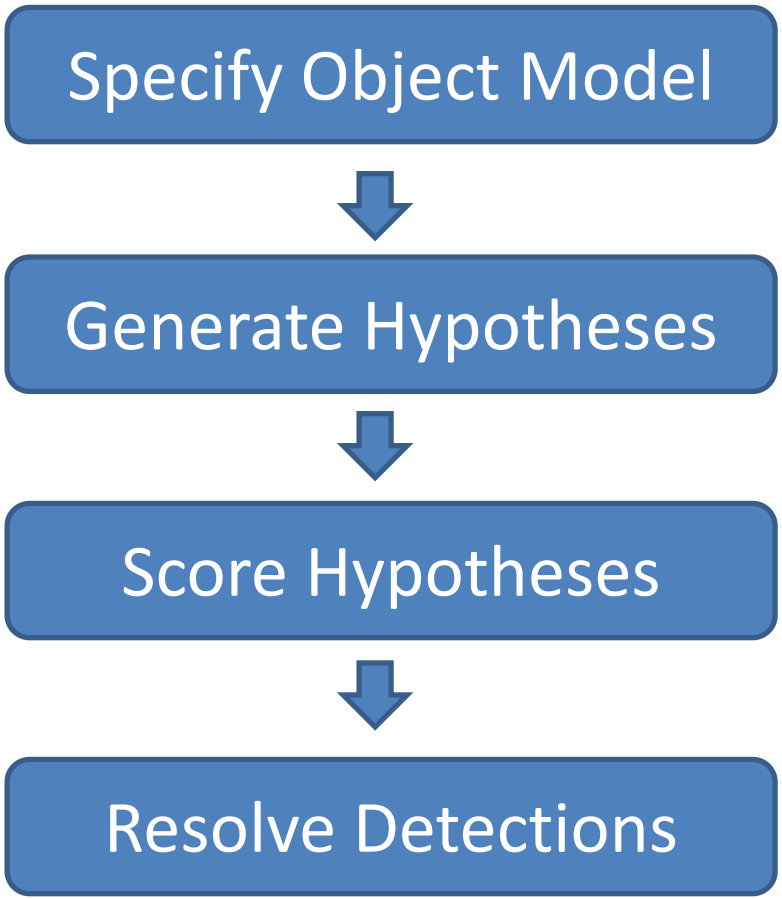
Object Detection Design challenges

- How to efficiently search for likely objects
  - Even simple models require searching hundreds of thousands of positions and scales.

- Feature design and scoring
  - How should appearance be modeled?
    What features correspond to the object?

- How to deal with different viewpoints?
  - Often train different models for a few different viewpoints

General Process of Object Recognition

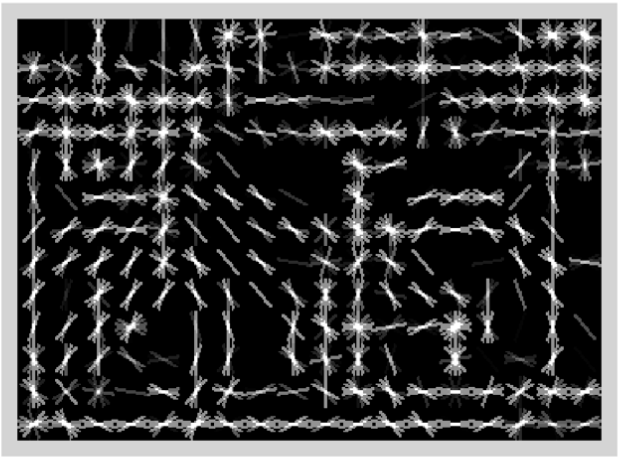


What are the object parameters?

Specifying an object model

1. Statistical Template in Bounding Box
   - Object is some (x,y,w,h) in image
   - Features defined wrt bounding box coordinates

Image

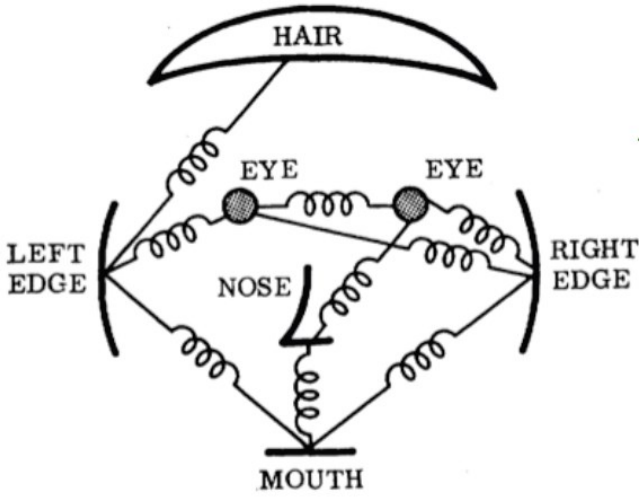


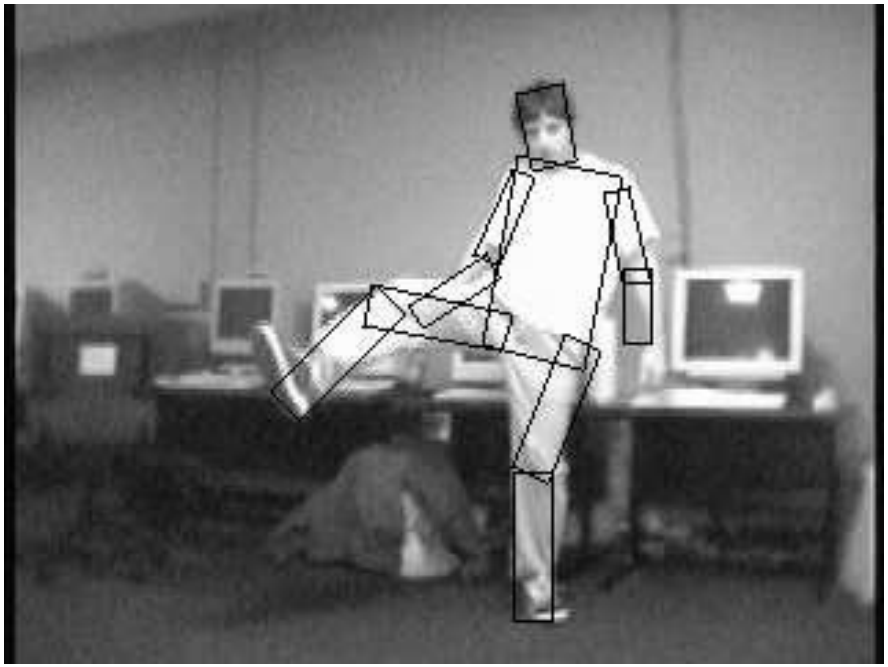
Template Visualization

Images from Felzenszwalb

Specifying an object model

2. Parts model
   - Object is configuration of parts
   - Each part is detectable



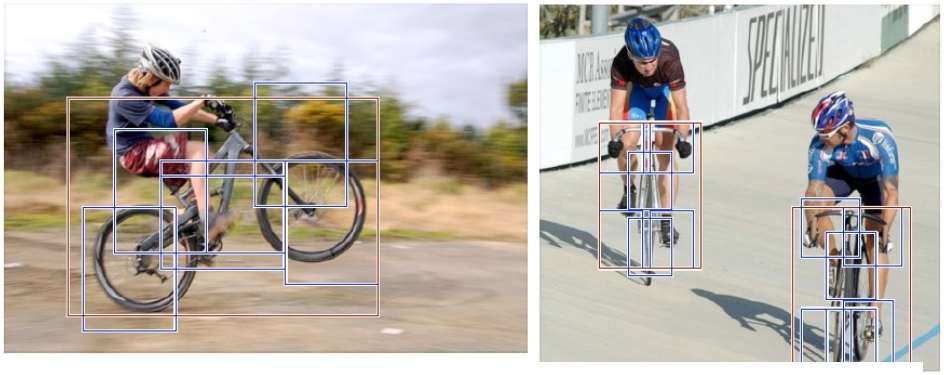


Images from Felzenszwalb

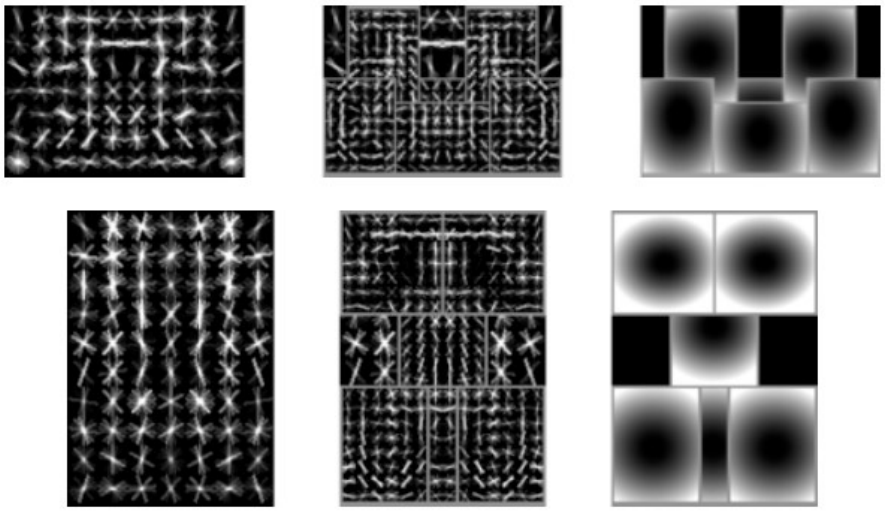
Specifying an object model

3. Hybrid template/parts model

Detections



Template Visualization



root filters
coarse resolution

part filters
finer resolution

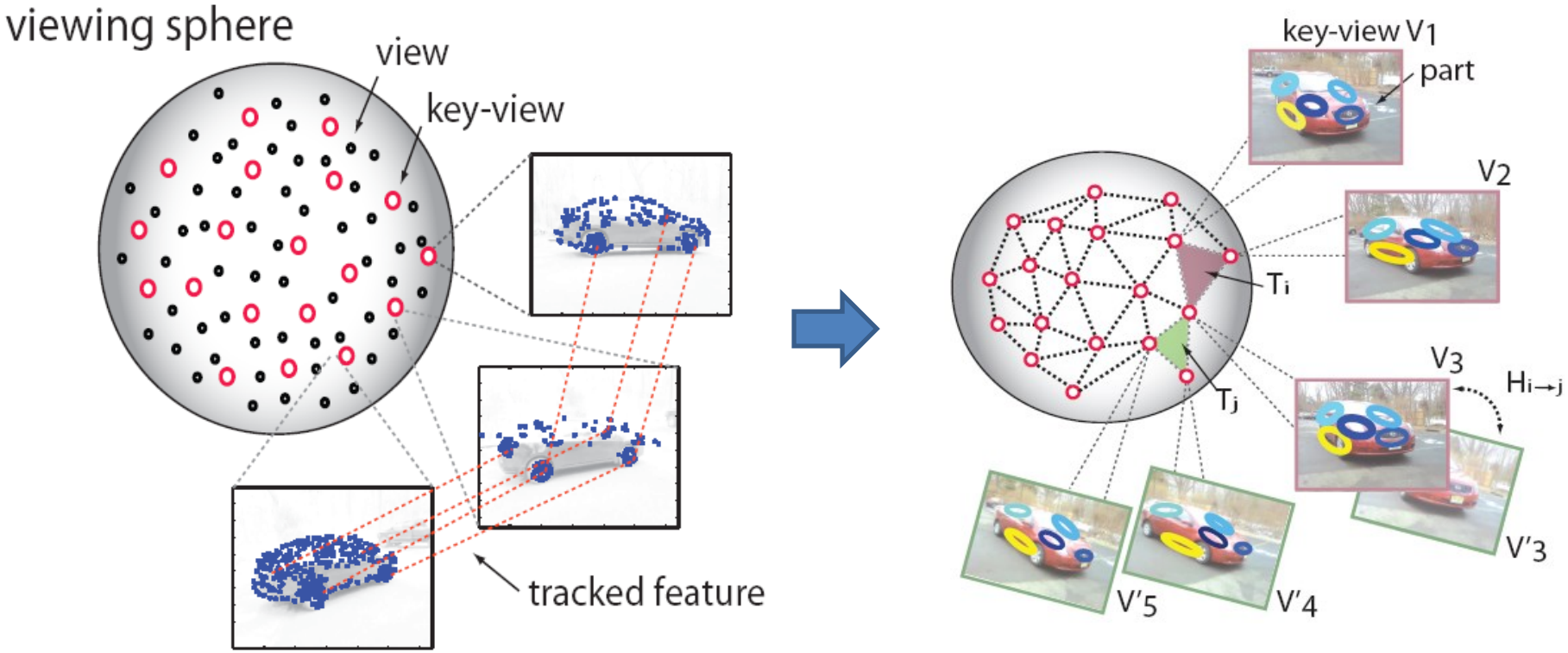
deformation
models

Felzenszwalb et al. 2008

Specifying an object model

4. 3D-ish model

- Object is collection of 3D planar patches under affine transformation

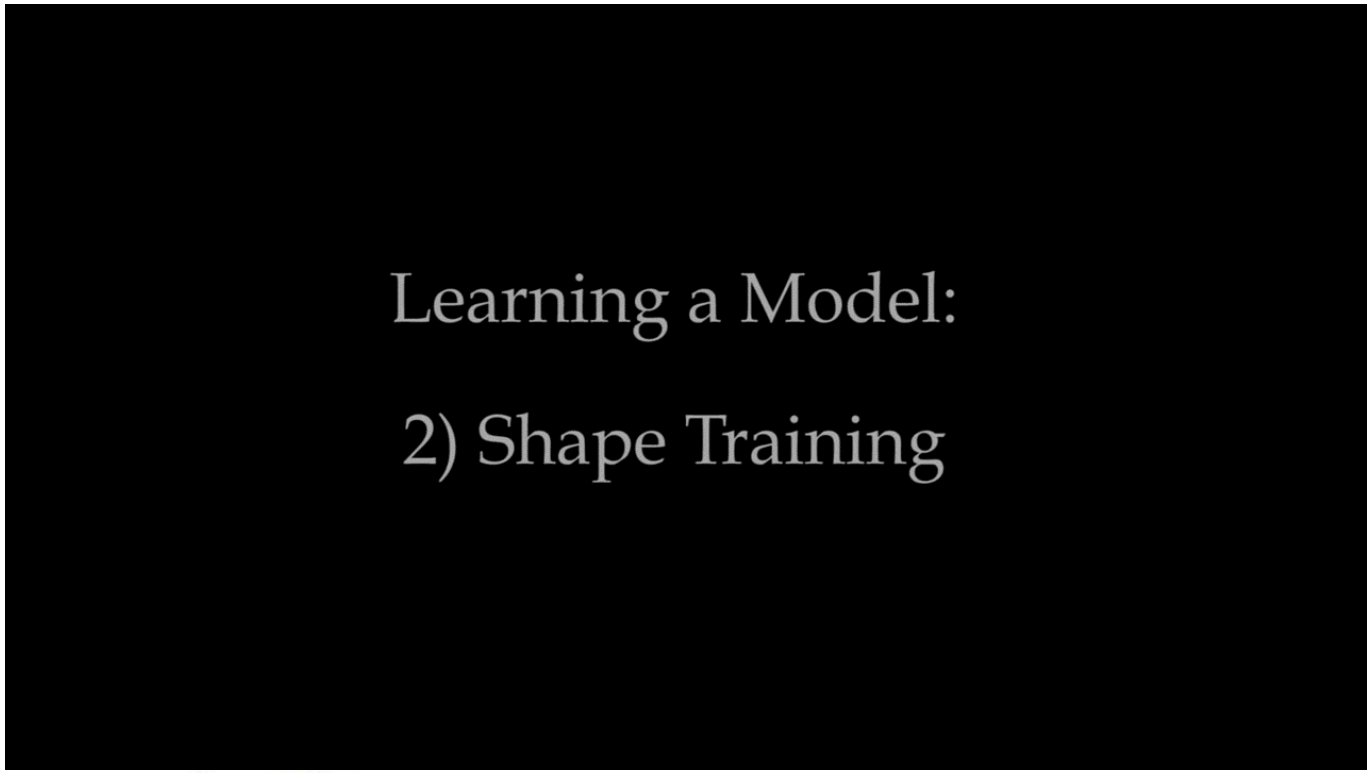


Loper et al. 2015

Specifying an object model

5. Deformable 3D model

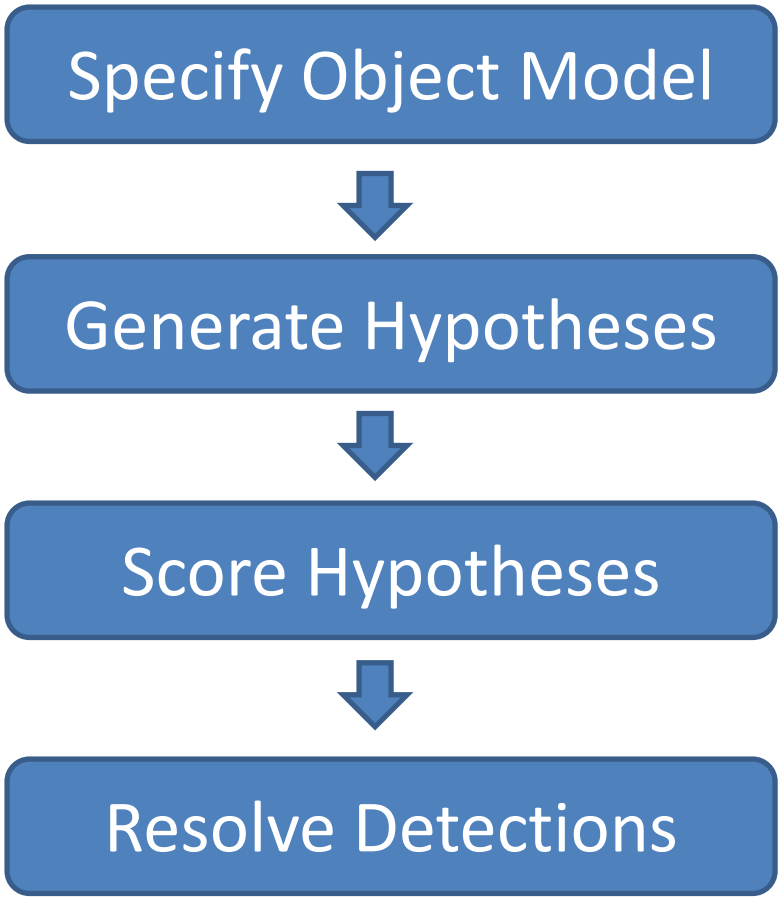
- Object is a parameterized space of shape/pose/deformation of class of 3D object



Why not just pick the most complex model?

- Inference is harder
  - More parameters
  - Harder to ‘fit’ (infer / optimize fit)
  - Longer computation

General Process of Object Recognition



Propose an alignment of the model to the image

Generating hypotheses

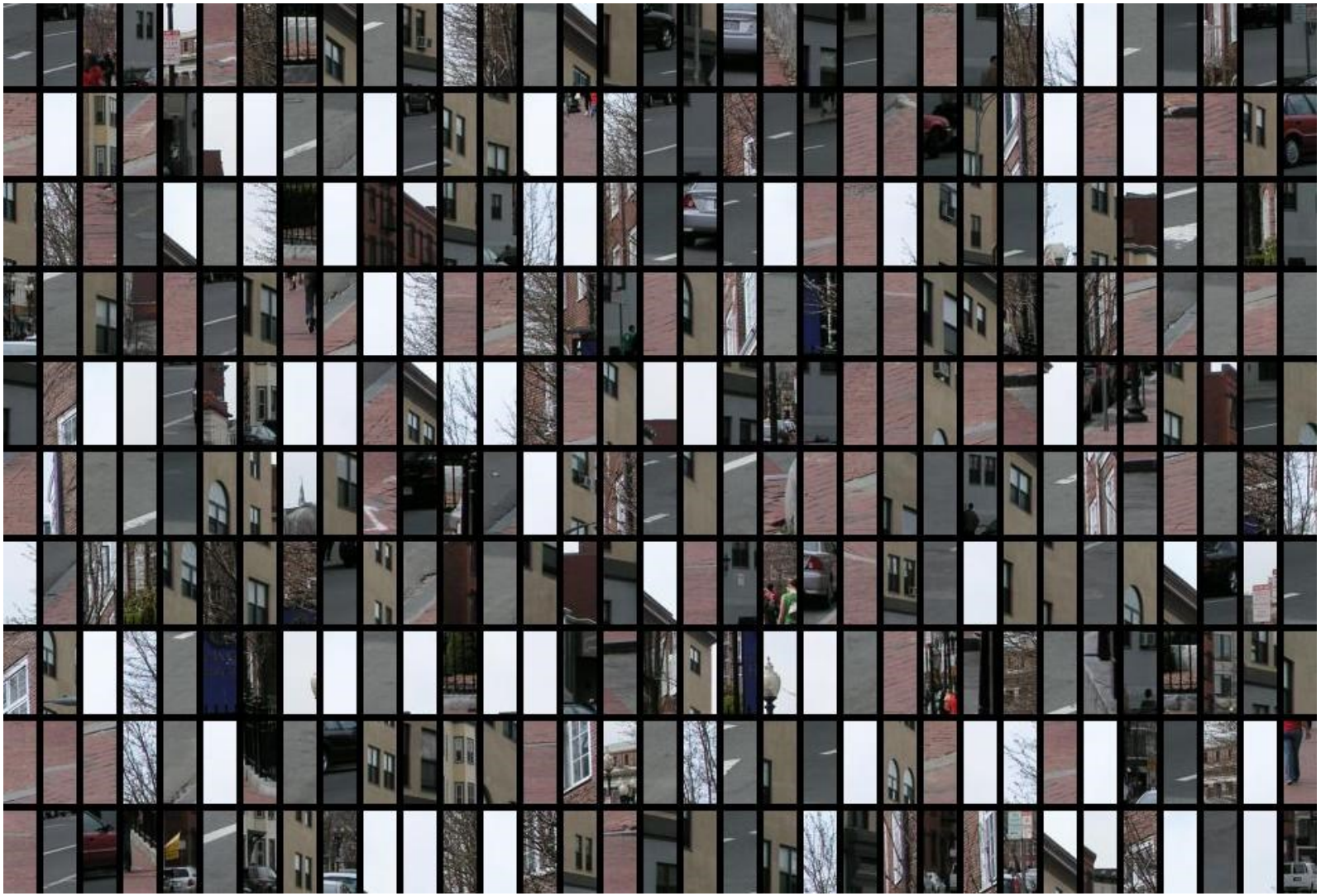
1. Sliding window
   - Test patch at each location and scale

Generating hypotheses

1. Sliding window
   - Test patch at each location and scale

Note – Template did not change size

Each window is separately classified



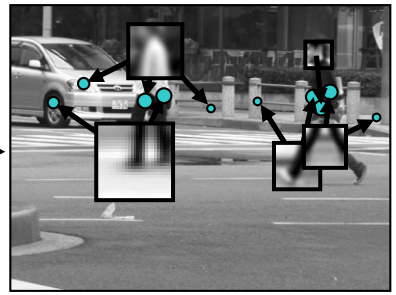
Generating hypotheses

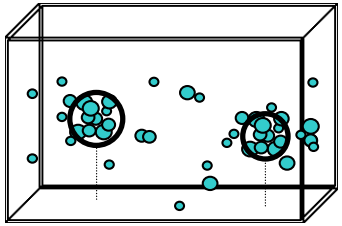
2. Voting from patches/keypoints

Interest Points

Matched Codebook Entries

Probabilistic Voting

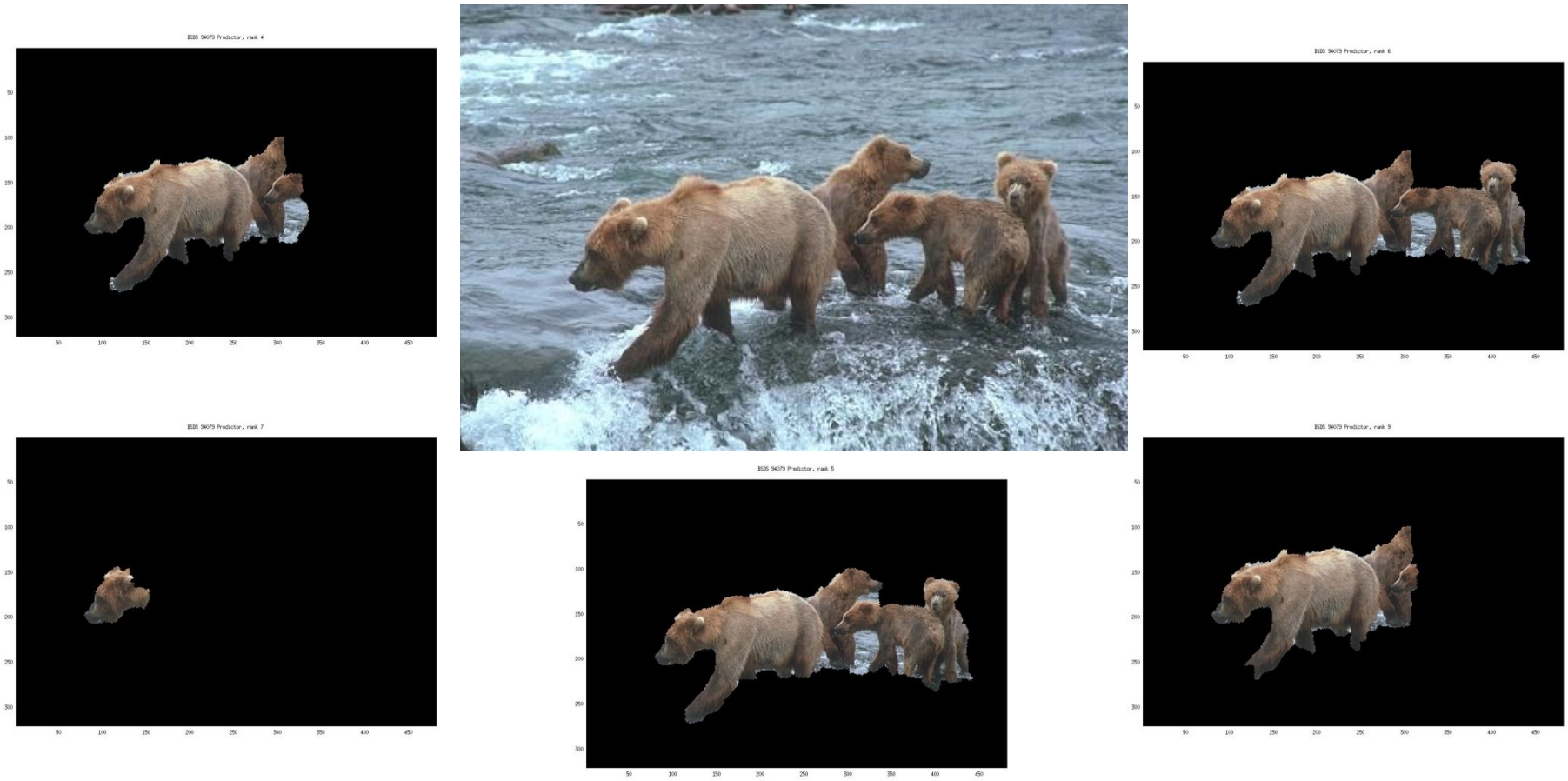




3D Voting Space (continuous)

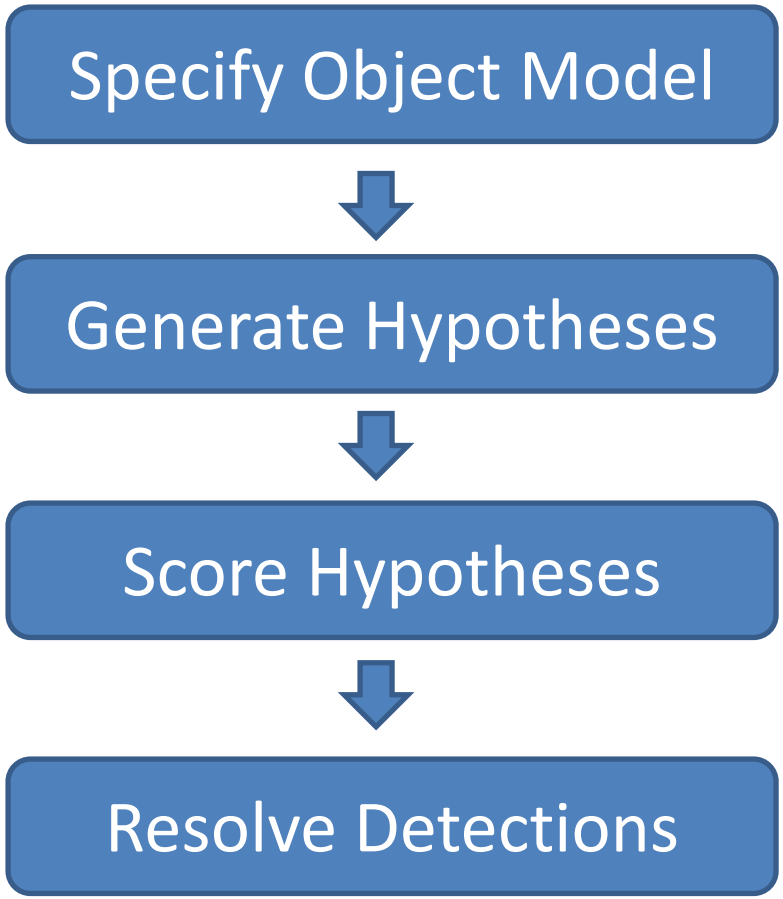
Generating hypotheses

3. Region-based proposal



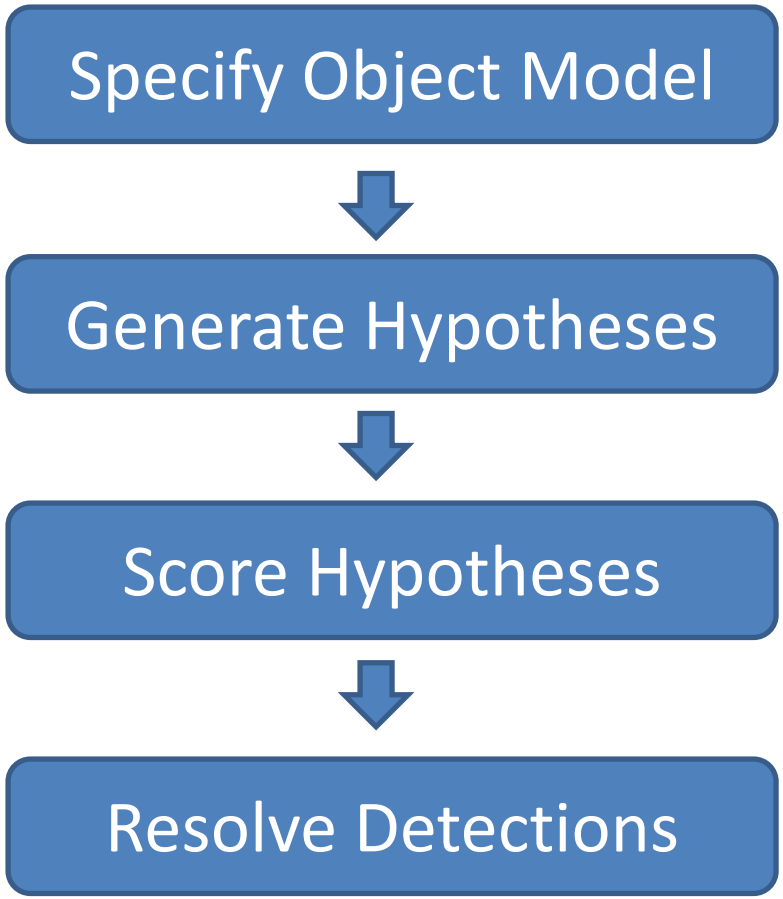
Endres Hoiem 2010

General Process of Object Recognition



Mainly-gradient based features, usually based on summary representation, many classifiers

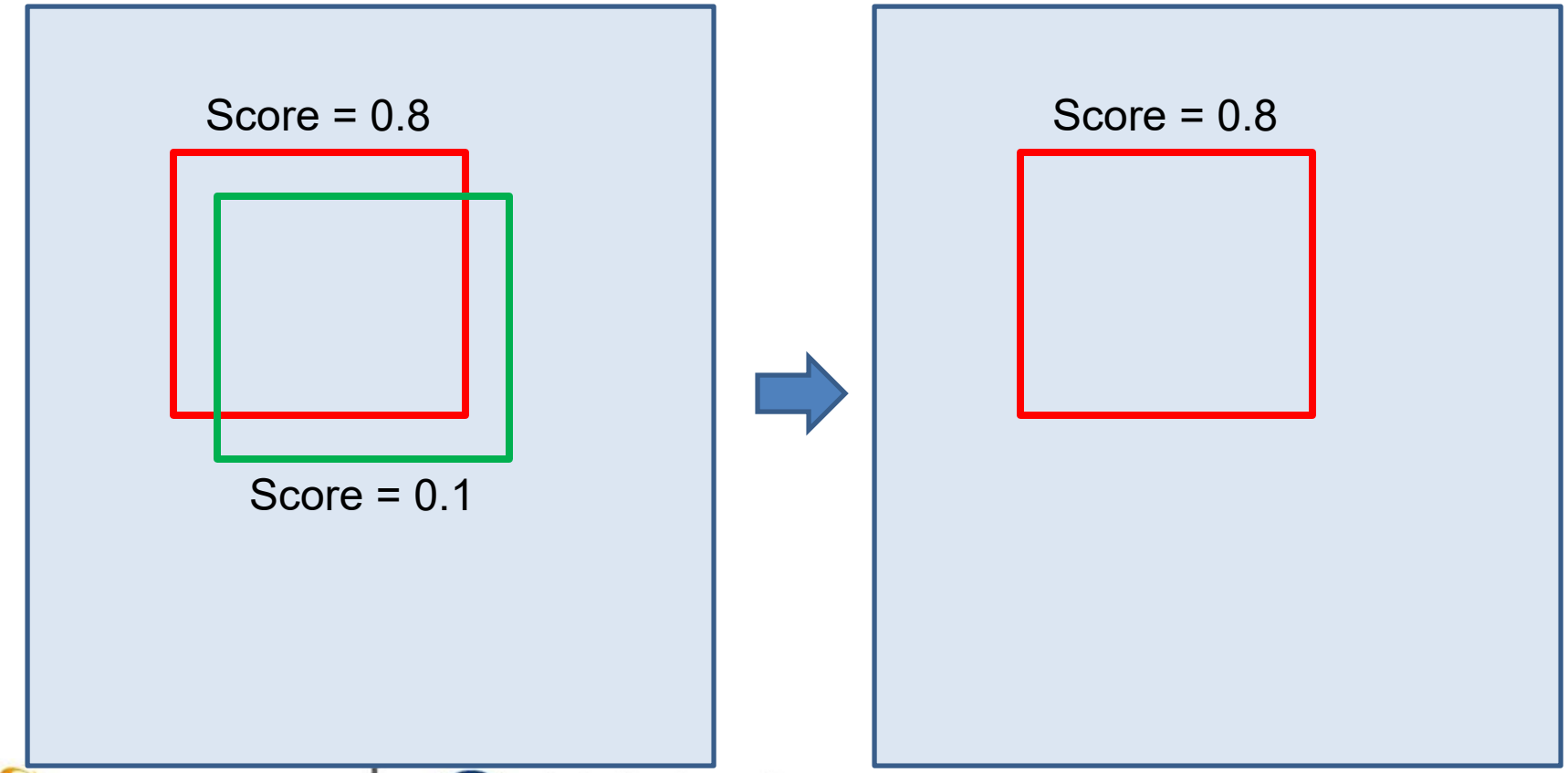
General Process of Object Recognition



Rescore each proposed object based on whole set

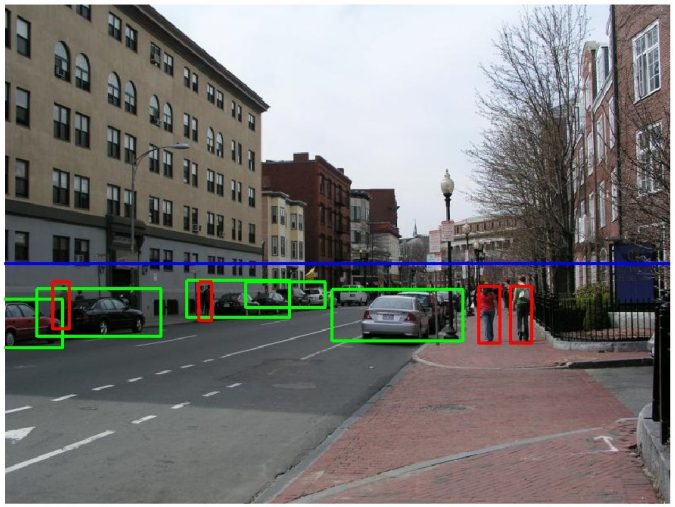
Resolving detection scores

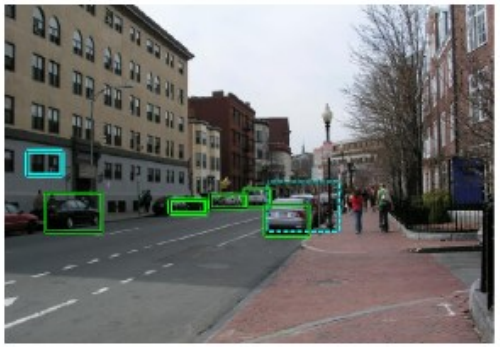
1. Non-max suppression



Resolving detection scores

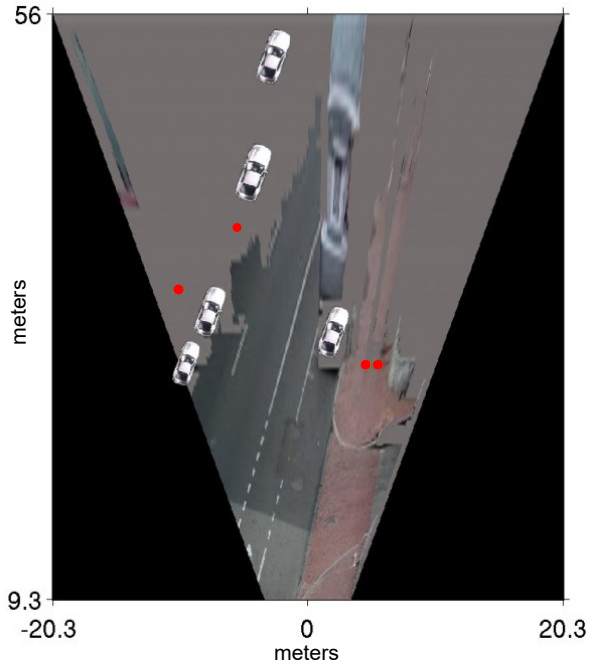
2. Context/reasoning





(g) Car Detections: Local

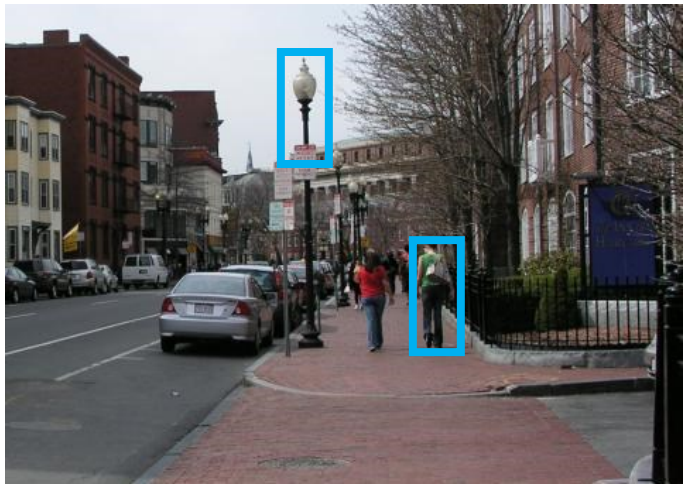
(h) Ped Detections: Local



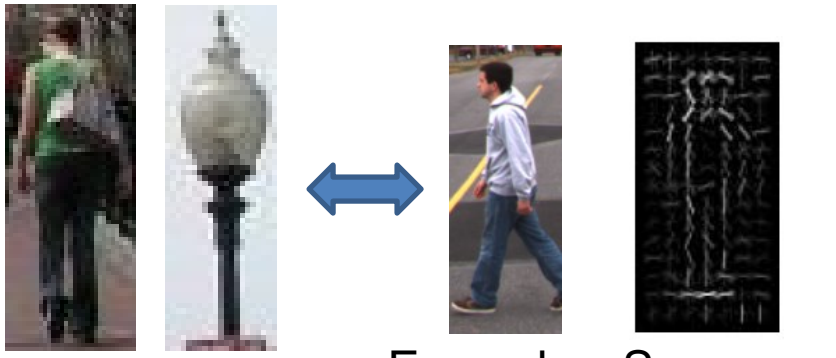
Hoiem et al. 2006

Basic Steps of Category Detection

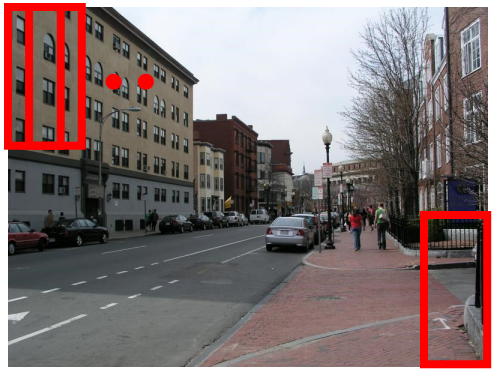
1. Align
   - E.g., choose position, scale orientation
   - How to make this tractable?



2. Compare
   - Compute similarity to an example object or to a summary representation
   - Which differences in appearance are important?



Sliding window: a simple alignment solution



Influential Works in Detection

- Sung-Poggio (1994, 1998) : ~2000 citations
  - Basic idea of statistical template detection, bootstrapping to get "face-like" negative examples, multiple whole-face prototypes (in 1994)
- Rowley-Baluja-Kanade (1996-1998) : ~3600
  - "Parts" at fixed position, non-maxima suppression, simple cascade, rotation, pretty good accuracy, fast
- Schneiderman-Kanade (1998-2000,2004) : ~1700
  - Careful feature engineering, excellent results, cascade
- Viola-Jones (2001, 2004) : ~13,000
  - Haar-like features, Adaboost as feature selection, hyper-cascade, very fast
- Dalal-Triggs (2005) : ~16,000 citations
  - Careful feature engineering, excellent results, HOG feature, online code
- Felzenszwalb-McAllester-Ramanan (2008): ~4,600 citations
  - Template/parts-based blend
- Girshick et al. (2013): ~2000 citations
  - R-CNN / Fast R-CNN / Faster R-CNN. Deep learned models on object proposals.

Dalal Triggs: Person detection with HOG & linear SVM

- Histograms of Oriented Gradients for Human Detection, Navneet Dalal, Bill Triggs, International Conference on Computer Vision & Pattern Recognition - June 2005
- http://lear.inrialpes.fr/pubs/2005/DT05/

Statistical Template

Object model =

sum of scores of features at fixed positions

+3 +2 -2 -1 -2.5 = -0.5 ? > 7.5

Non-object

+4 +1 +0.5 +3 +0.5 = 10.5 ? > 7.5

Object

Example: Dalal-Triggs pedestrian detector

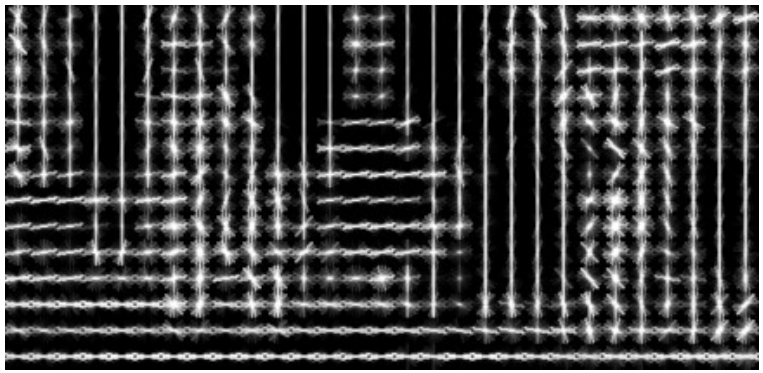
1. Extract fixed-sized (64x128 pixel) window at each position and scale
2. Compute HOG (histogram of gradient) features within each window
3. Score the window with a linear SVM classifier
4. Perform non-maxima suppression to remove overlapping detections with lower scores

Navneet Dalal and Bill Triggs, Histograms of Oriented Gradients for Human Detection, CVPR05

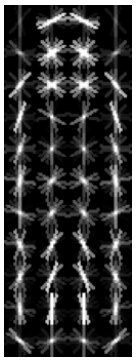
Pedestrian detection with HOG

- Learn a pedestrian template using a support vector machine
- At test time, compare feature map with template over sliding windows.
- Find local maxima of response
- *Multi-scale*: repeat over multiple levels of a HOG pyramid

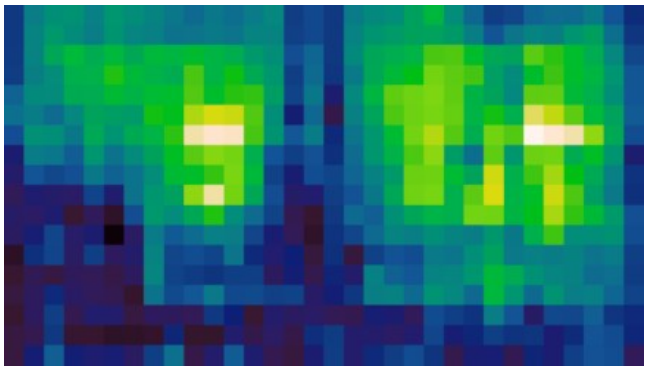
HOG feature map



Template



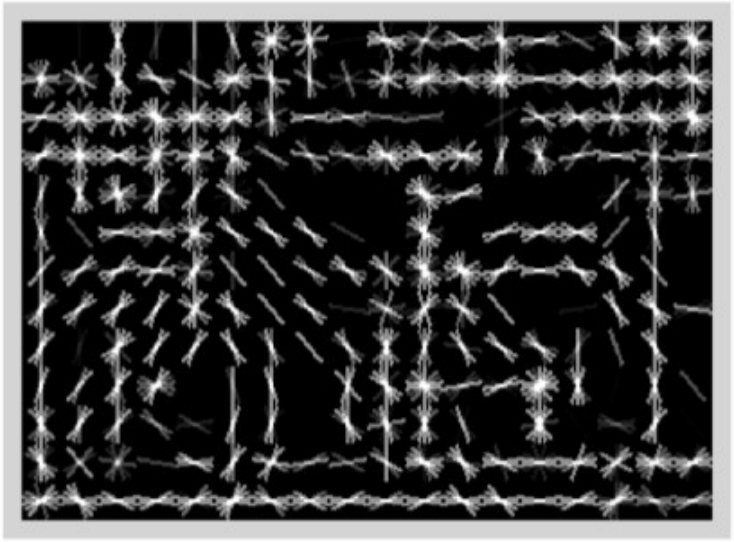
Detector response map



Can be continuous for more sophisticated maxima finding

N. Dalal and B. Triggs, Histograms of Oriented Gradients for Human Detection, CVPR 2005

Histogram of Gradient (HOG) features

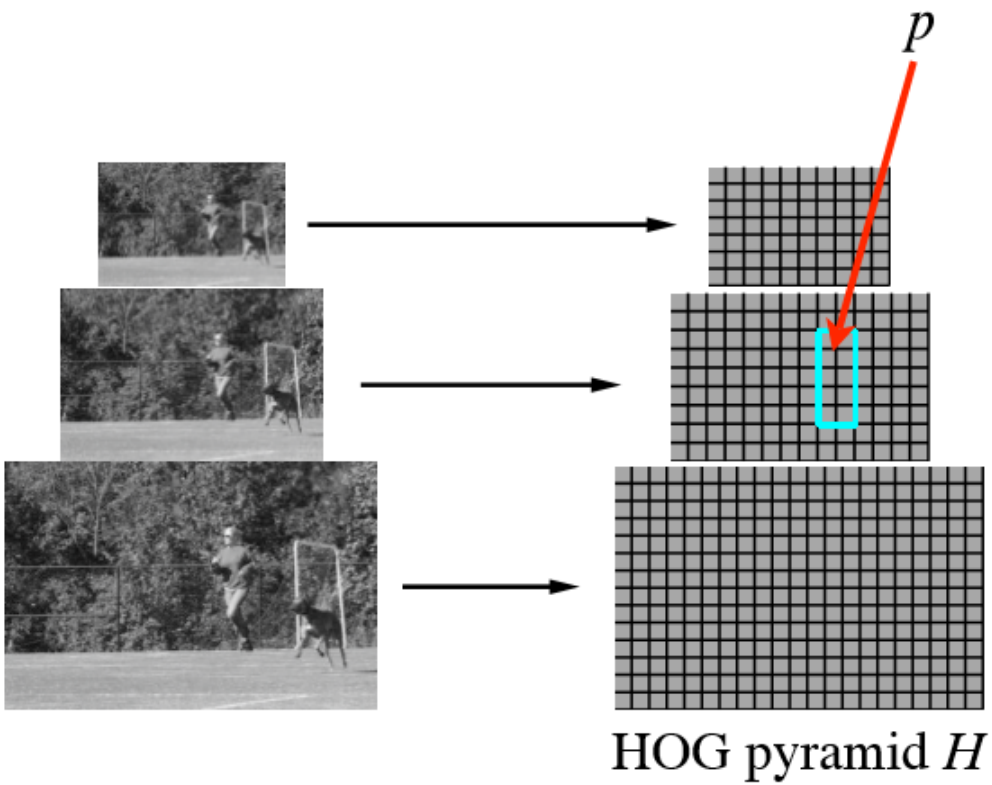


- Image is partitioned into 8x8 pixel blocks
- In each block we compute a histogram of gradient orientations
  - Invariant to changes in lighting, small deformations, etc.
- Compute features at different resolutions (pyramid)

Felzenszwalb

HOG Filters

- Array of weights for features in subwindow of HOG pyramid
- Score is dot product of filter and feature vector

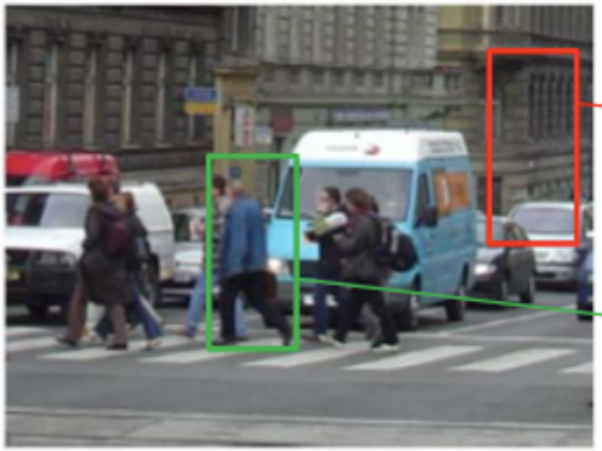


Filter F

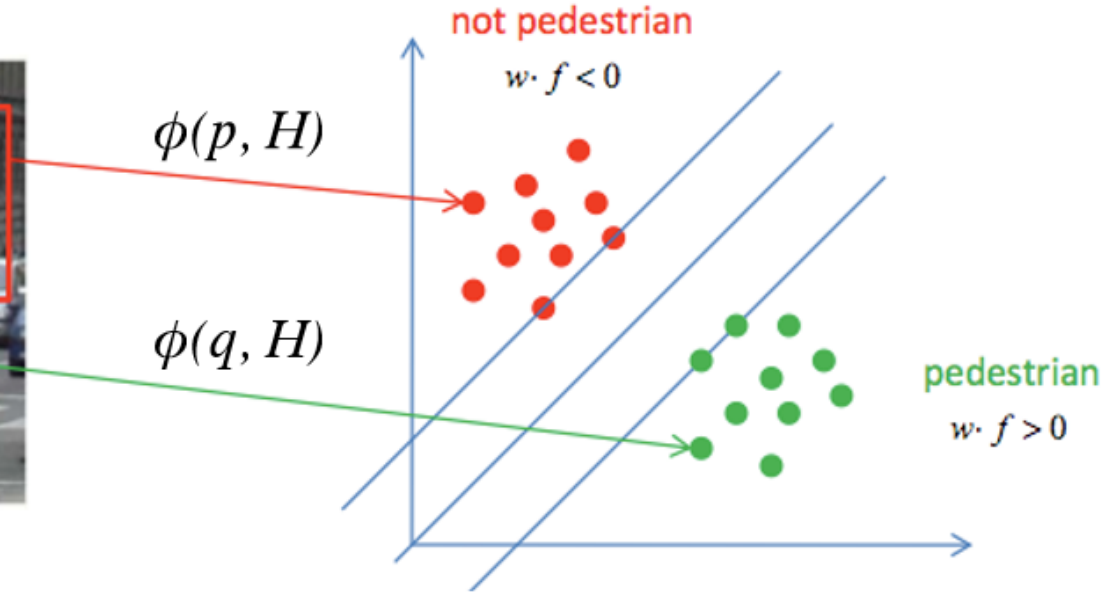
Score of F at position p is $F \cdot \phi(p, H)$

$\phi(p, H)$ = concatenation of HOG features from subwindow specified by p

Dalal & Triggs: HOG + linear SVMs



Typical form of a model



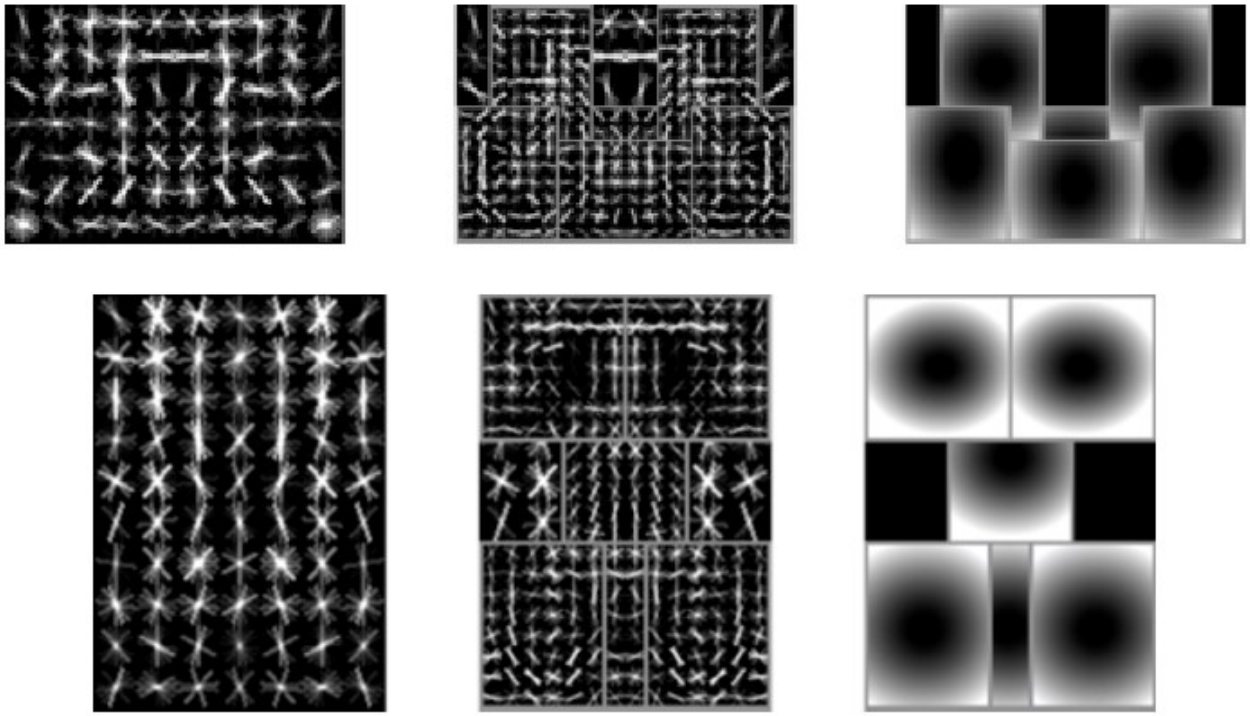
There is much more background than objects

Start with random negatives and repeat:

1) Train a model

2) Harvest false positives to define "hard negatives"

Discriminative part-based models

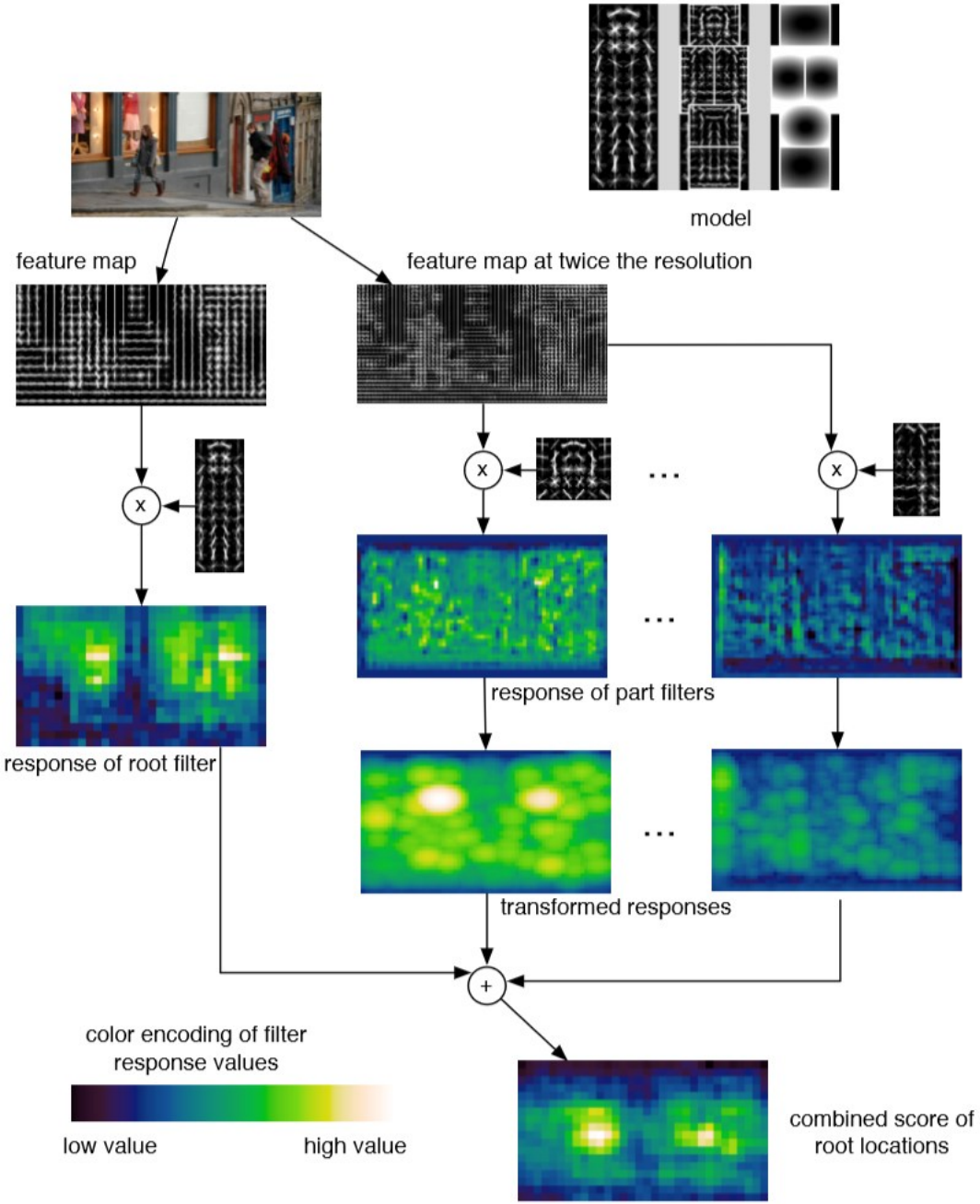


root filters
coarse resolution

part filters
finer resolution

deformation
models

Each component has a root filter F_0
and n part models (F_i, v_i, d_i)


model
feature map
feature map at twice the resolution
x
x
x
response of part filters
response of root filter
transformed responses
+
color encoding of filter
response values
low value
high value
combined score of
root locations

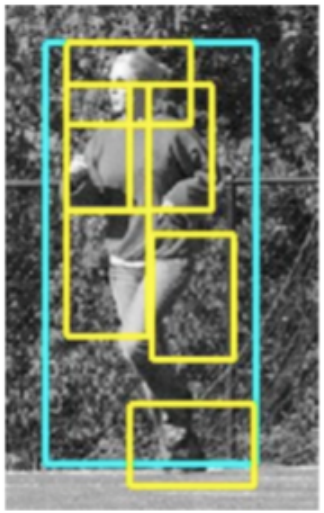
Score of a hypothesis

$$\text{score}(p_0, \ldots, p_n) = \underbrace{\sum_{i=0}^{n} F_i \cdot \phi(H, p_i)}_{\text{“data term”}} - \underbrace{\sum_{i=1}^{n} d_i \cdot (dx_i^2, dy_i^2)}_{\text{“spatial prior”}}$$

filters

displacements

deformation parameters



$$\text{score}(z) = \beta \cdot \Psi(H, z)$$

concatenation filters and deformation parameters

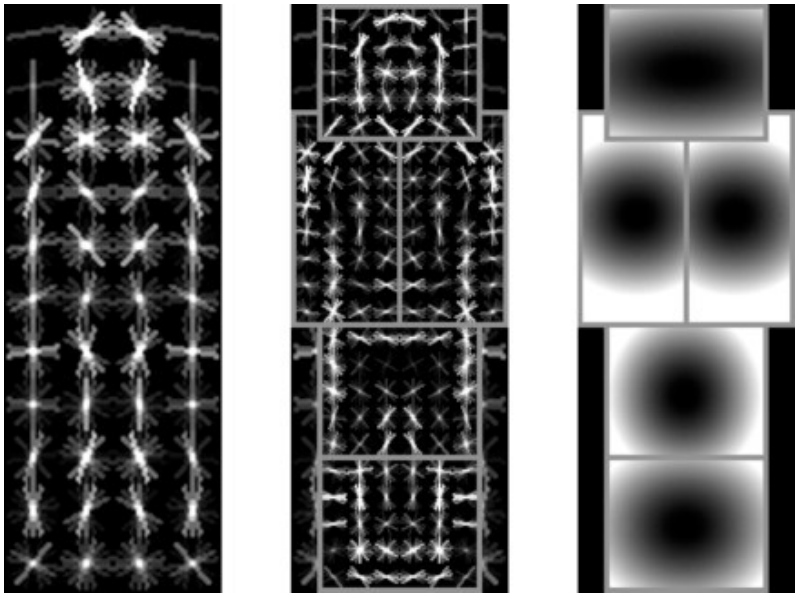
concatenation of HOG features and part displacement features

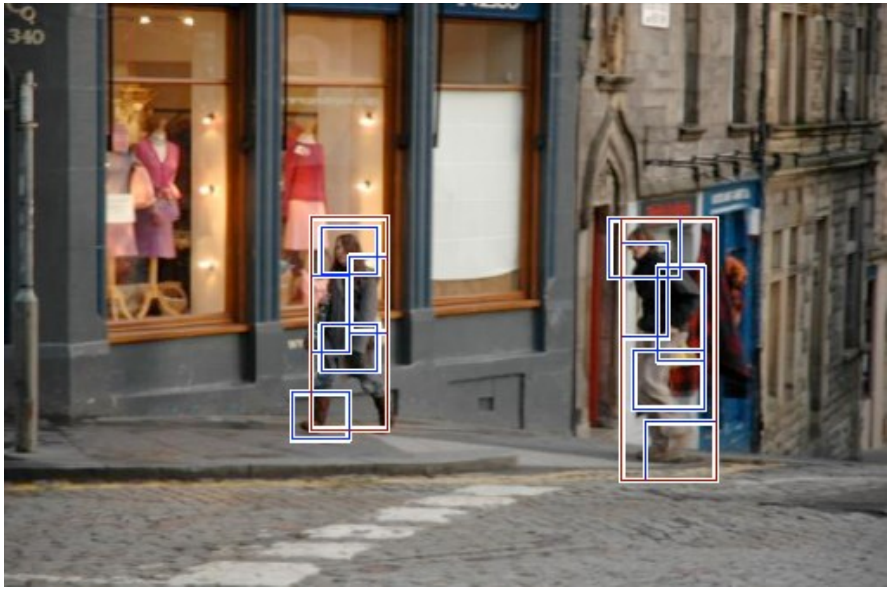
Discriminative part-based models

Root filter

Part filters

Deformation weights

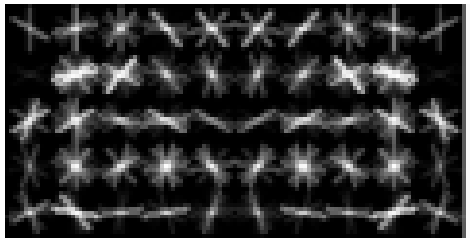
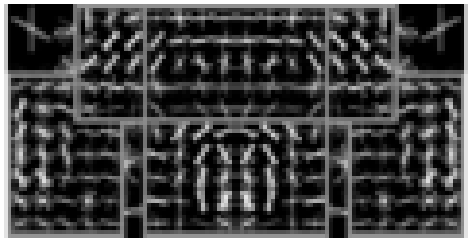
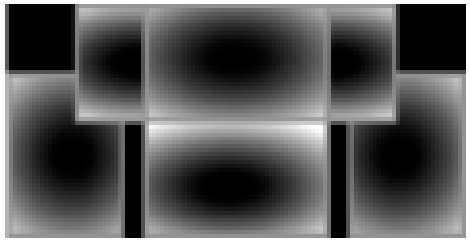




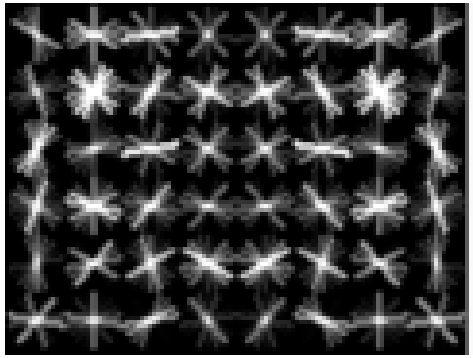
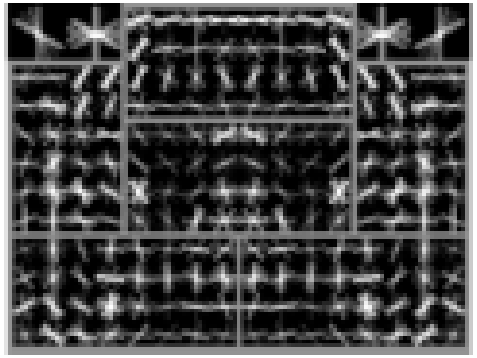
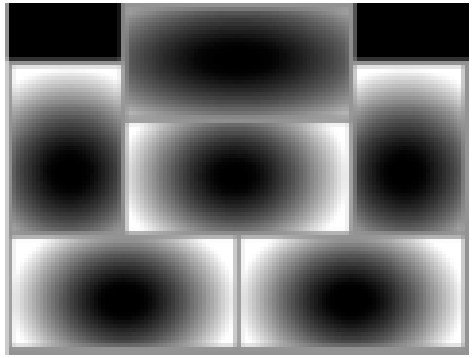
P. Felzenszwalb, R. Girshick, D. McAllester, D. Ramanan, Object Detection with Discriminatively Trained Part Based Models, PAMI 32(9), 2010

Car model

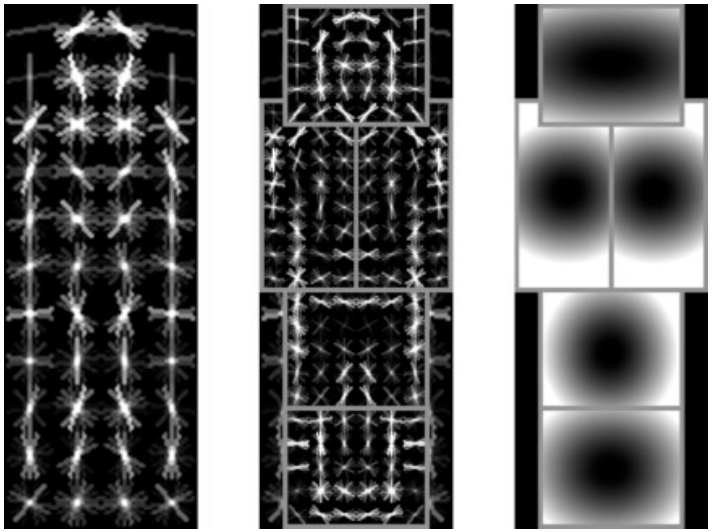
Component 1

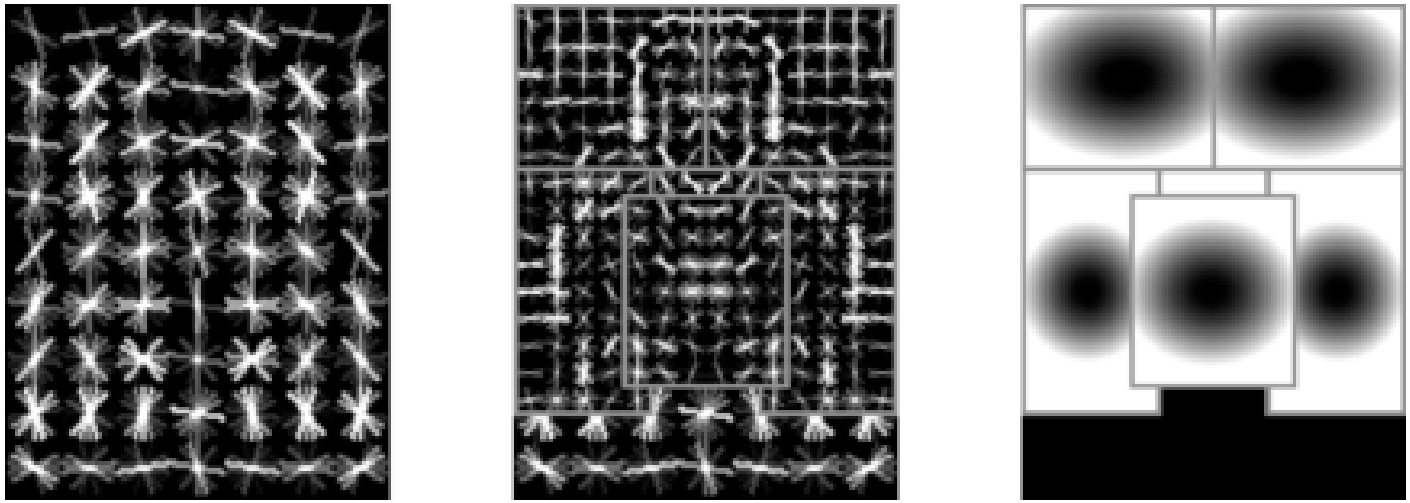




Component 2

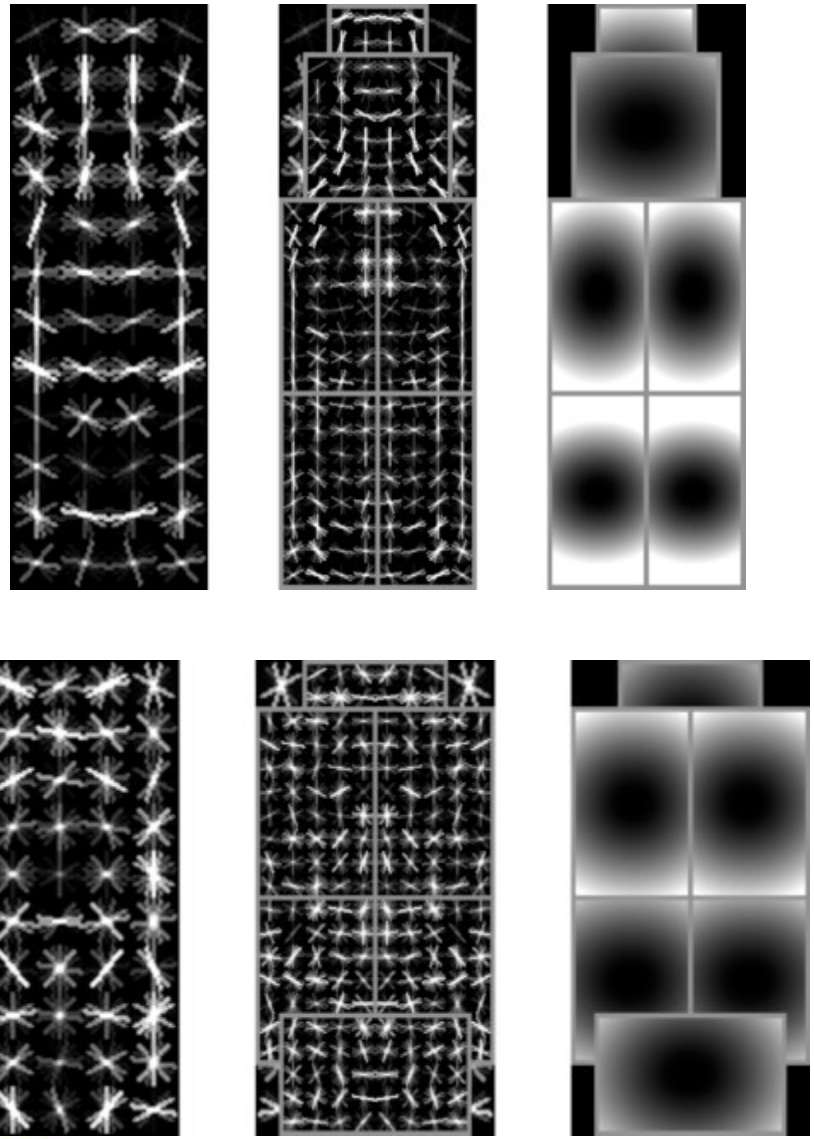




Person model



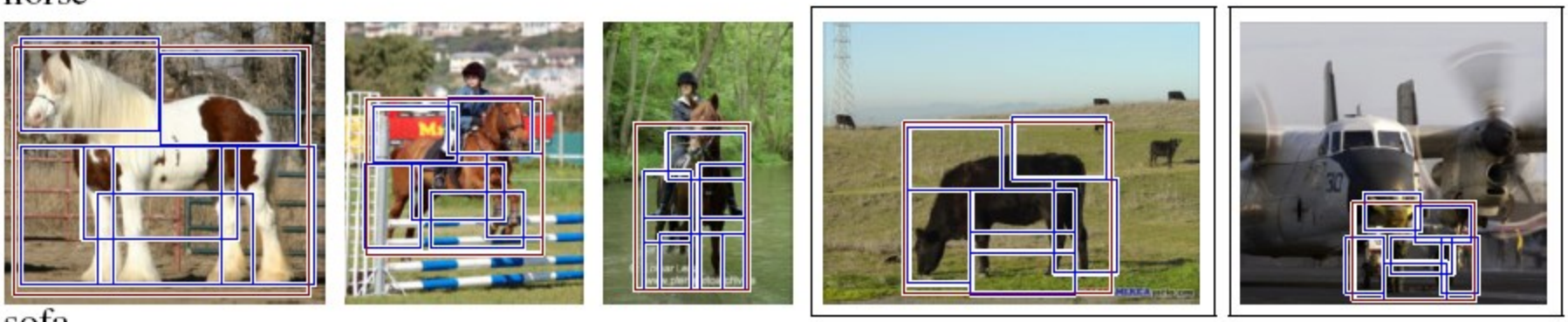


Bottle model

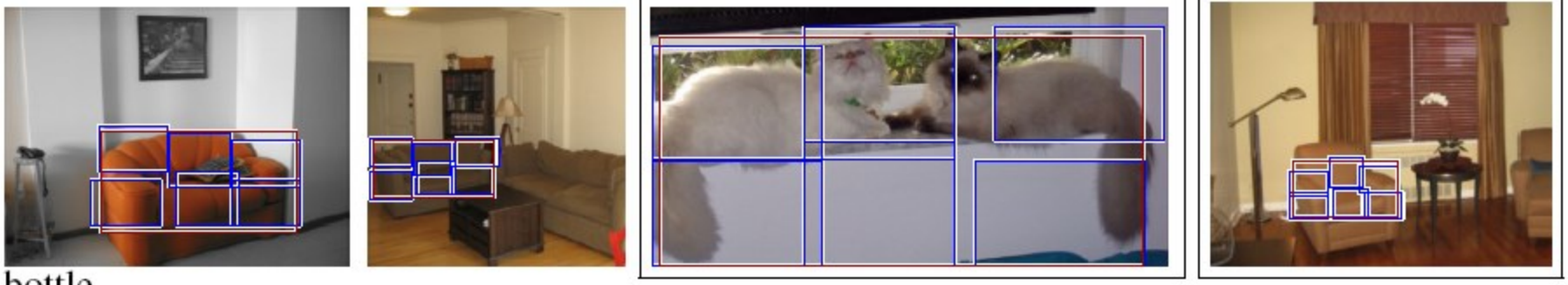


Good detections?

horse

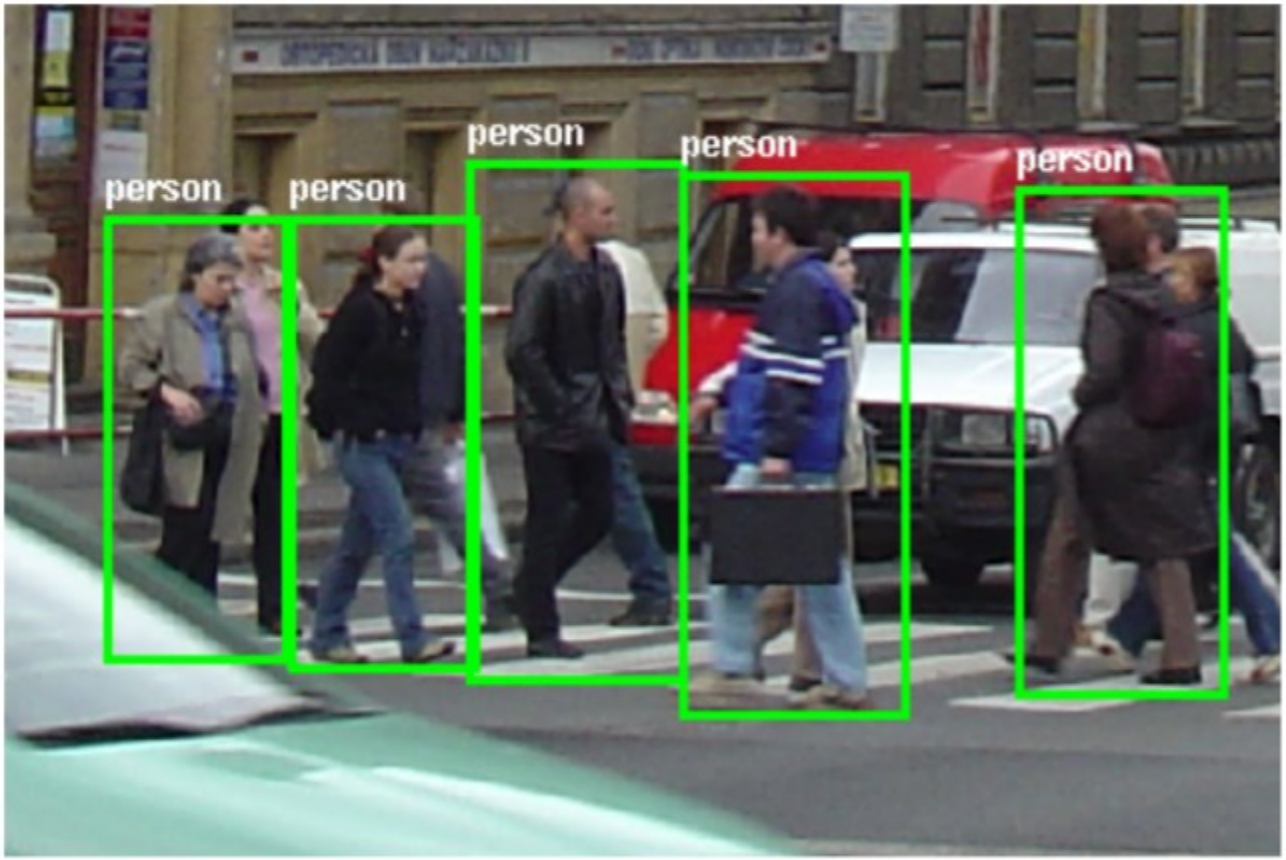


sofa



bottle

INRIA pedestrian database

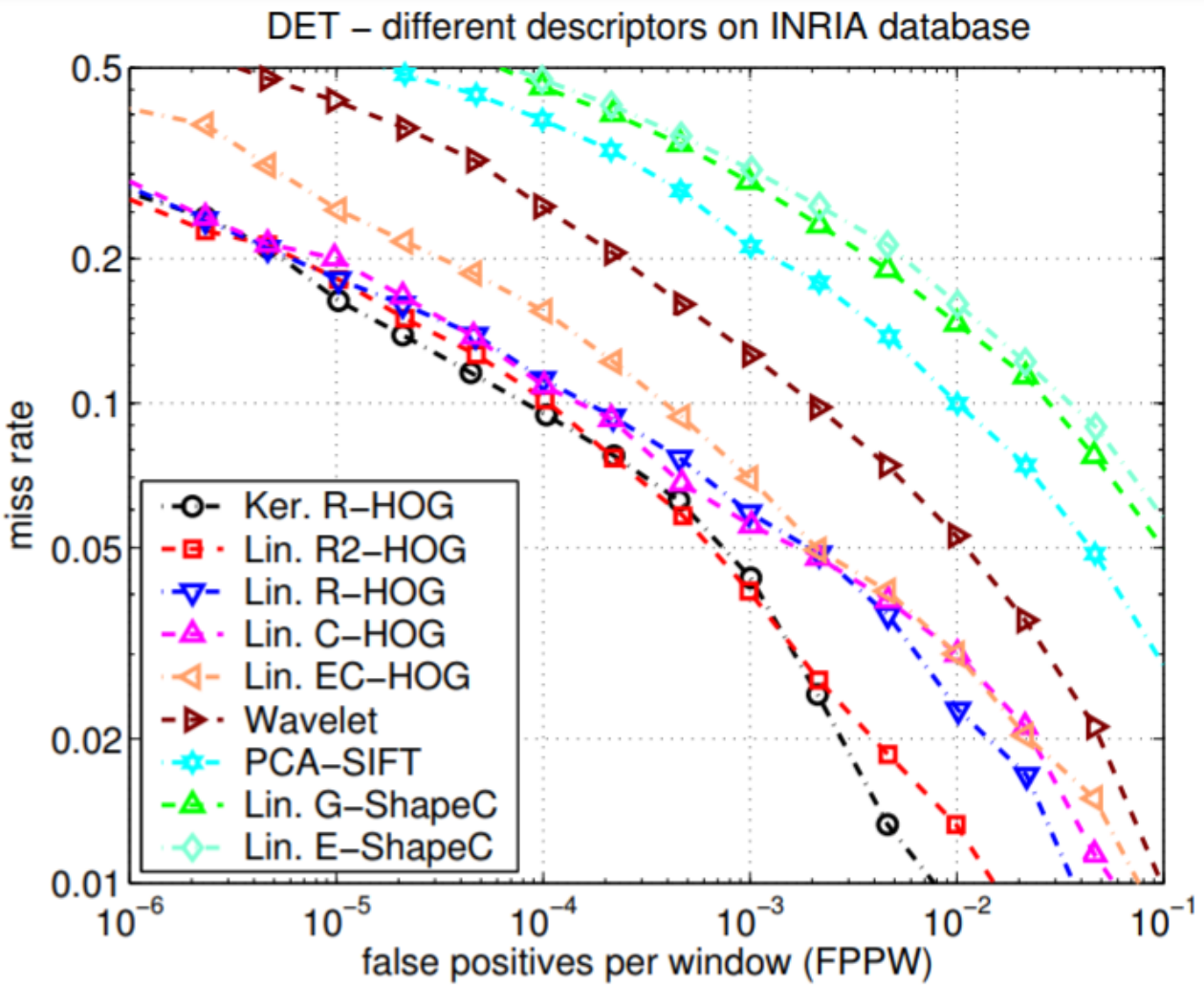


INRIA pedestrian database issues

Figure 1. Details from the INRIA test set highlighting some limitations. (a–d) Unlabelled persons. (e–h) Ambiguous cases. (e) Reflections of persons on a shop window, not labelled. (f) Some persons drawn on a wall, only one of them is labelled. (g) Some mannequins, all labelled. (h) A poster depicting a man, not labelled.

How good is HOG at person detection?

Miss rate =
1 - recall



Something to think about…

- Sliding window detectors work
  - *very well* for faces
  - *fairly well* for cars and pedestrians
  - *badly* for cats and dogs
- Why are some classes easier than others?

Strengths/Weaknesses of Statistical Template Approach

Strengths

- Works very well for non-deformable objects with canonical orientations: faces, cars, pedestrians
- Fast detection

Weaknesses

- Not so well for highly deformable objects or “stuff”
- Not robust to occlusion
- Requires lots of training data