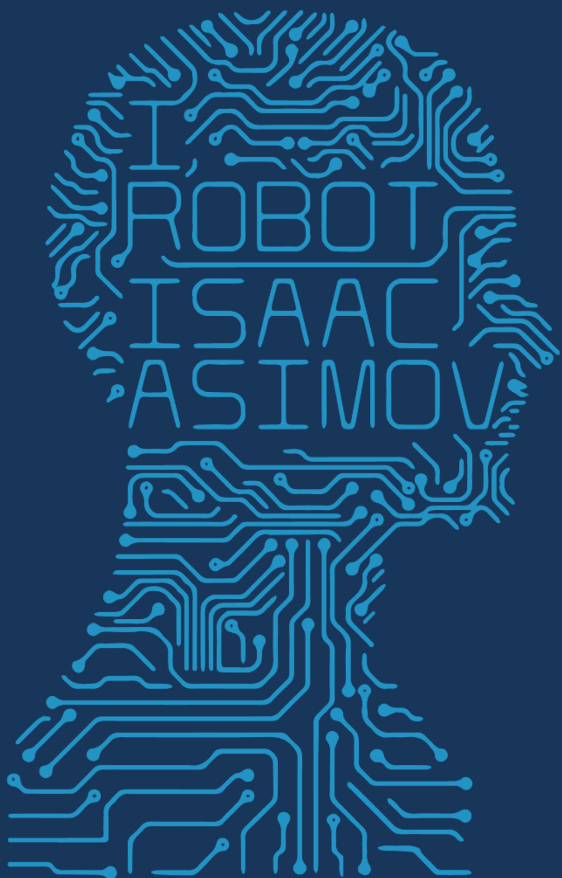
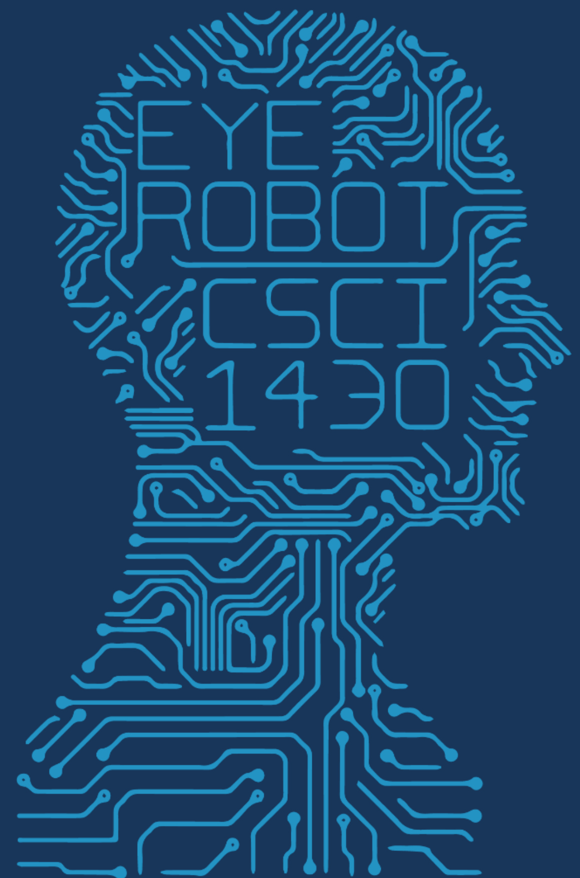


1950

Future Vision



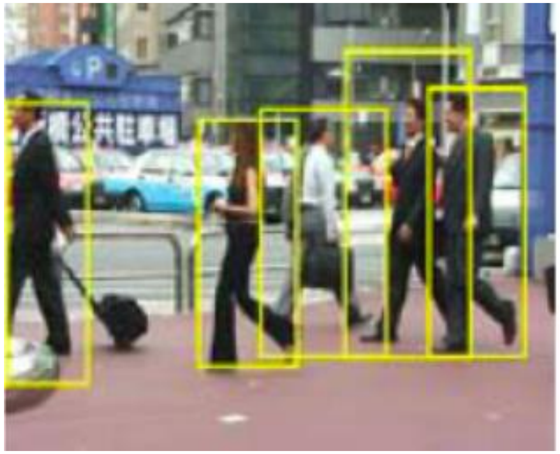
2017 MWF 1pm

Computer Vision

Category vs. instance recognition

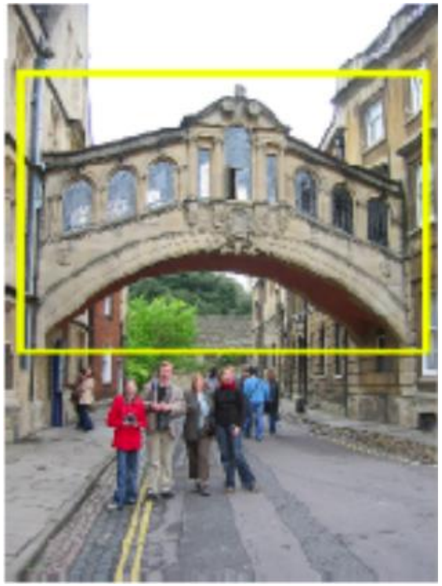
Category:

- Find all the people
- Find all the buildings
- Often within a single image
- Often 'sliding window'



Instance:

- Is this face James?
- Find this specific famous building
- Often within a database of images



Object detection vs. Scene Recognition

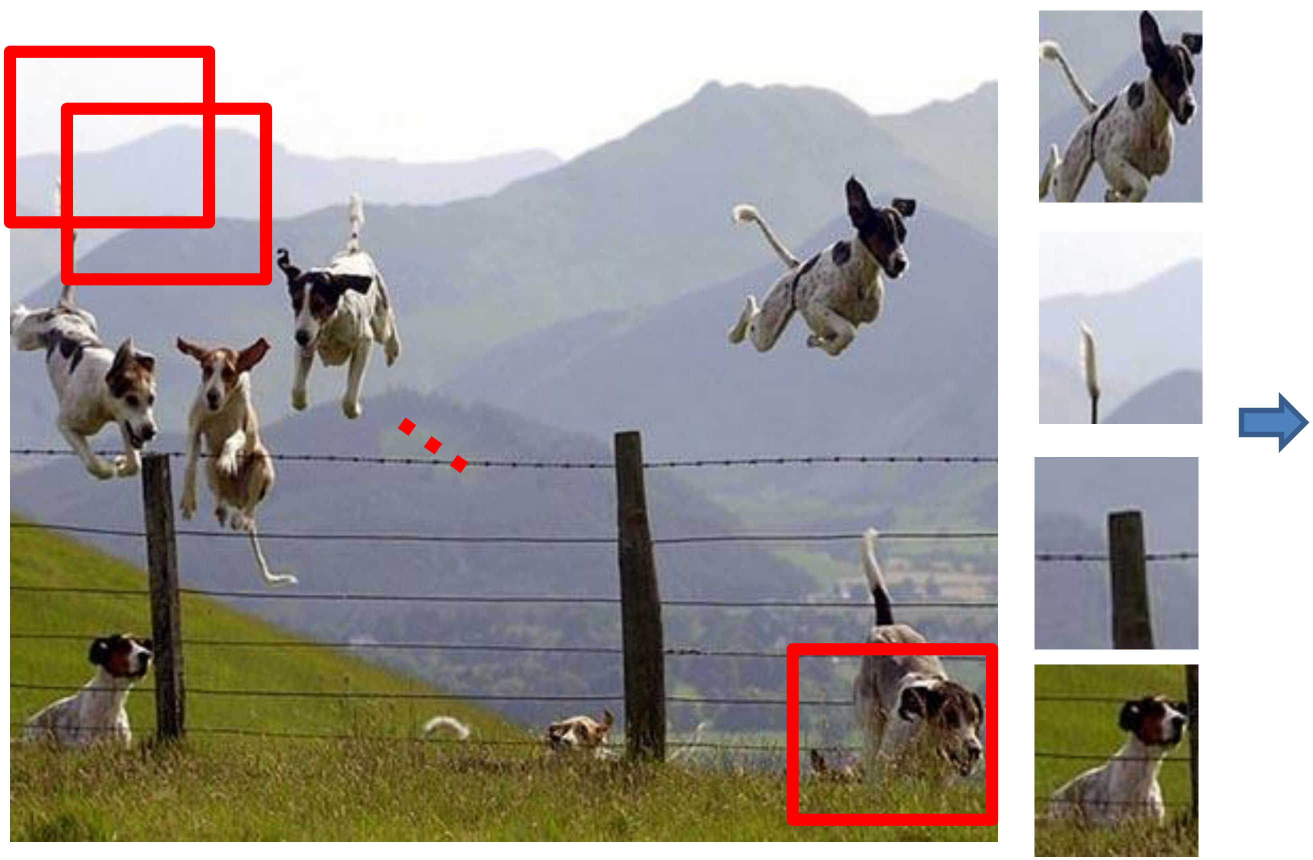
- Scenes can be defined by distribution of “stuff” – materials and surfaces with arbitrary shape.

- 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



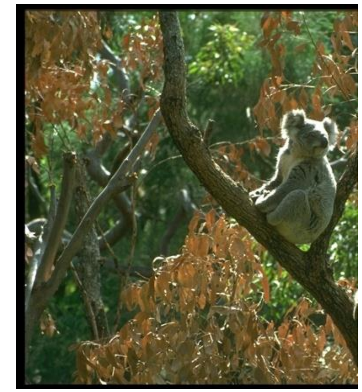
Object or **Non-Object?**

James Hays

Challenges in modeling the object class

Illumination

Object pose



'Clutter'

Occlusions

Intra-class appearance

Viewpoint

Slide from K. Grauman, B. Leibe

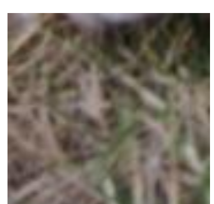
Challenges in modeling the non-object class

True Detections

Bad Localization

Confused with Similar Object

Misc. Background



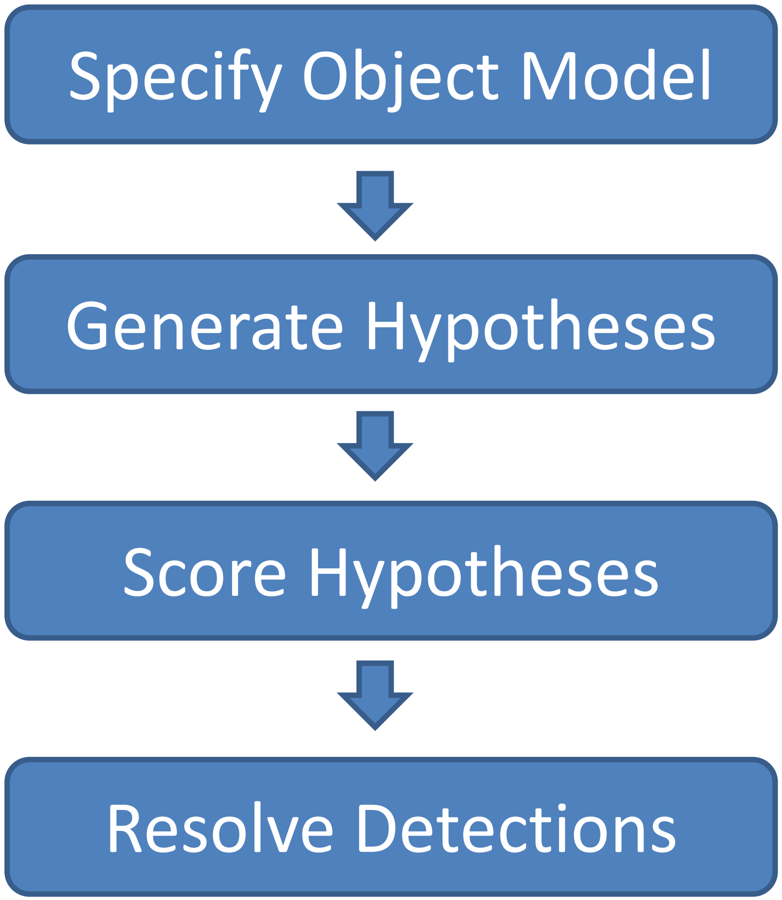
Confused with Dissimilar Objects

James Hays

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



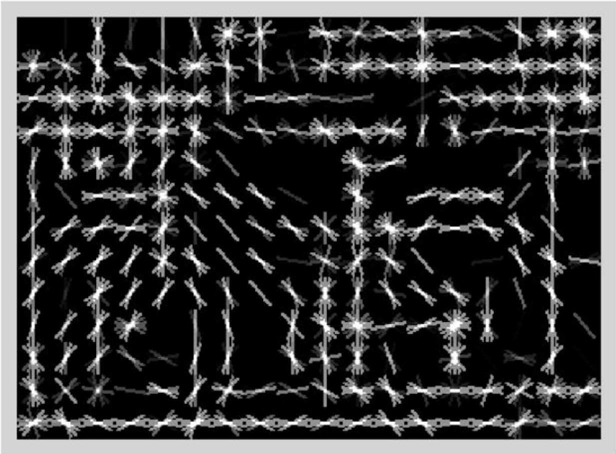
What are the object parameters?

James Hays

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

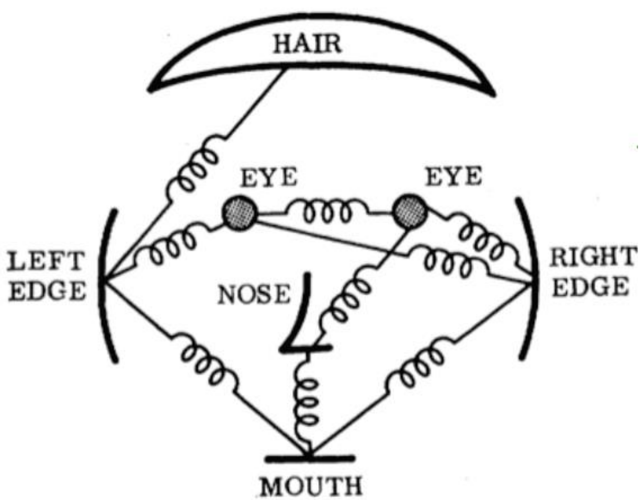


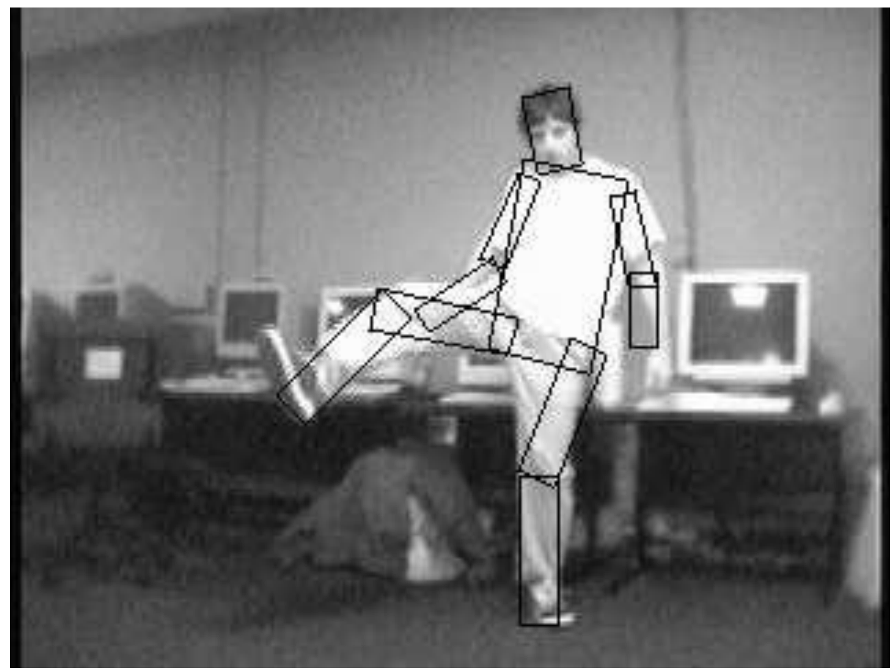
Template Visualization

Images from Felzenszwalb

Specifying an object model

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



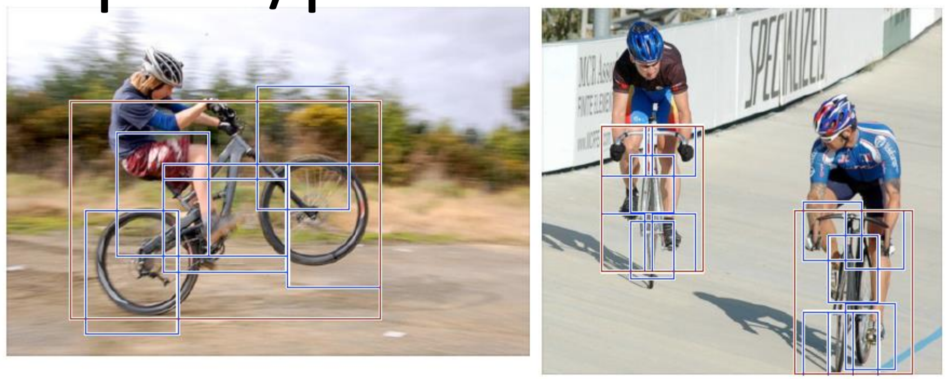


Images from Felzenszwalb

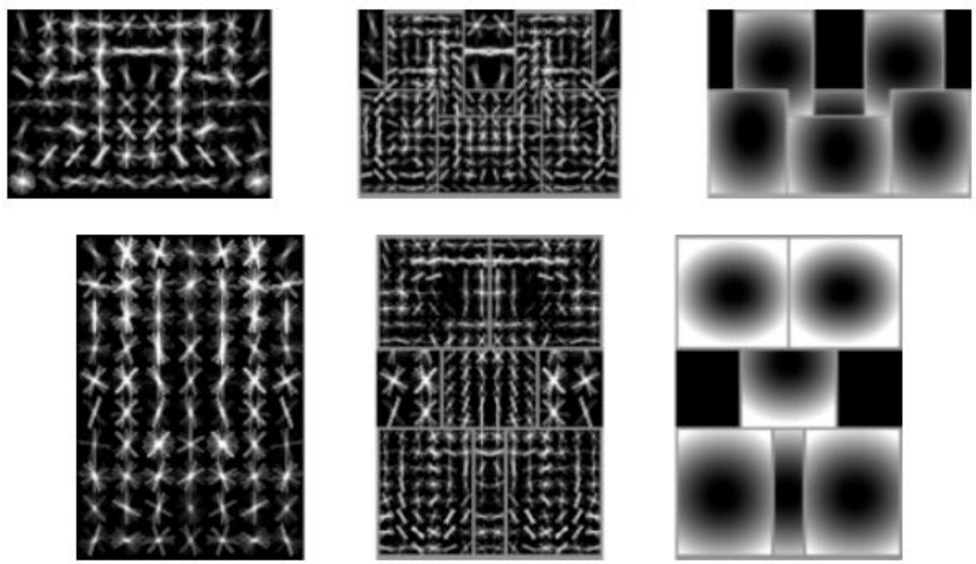
Specifying an object model

3. Hybrid template/parts model

Detections



Template Visualization



root filters
coarse resolution

part filters
finer resolution

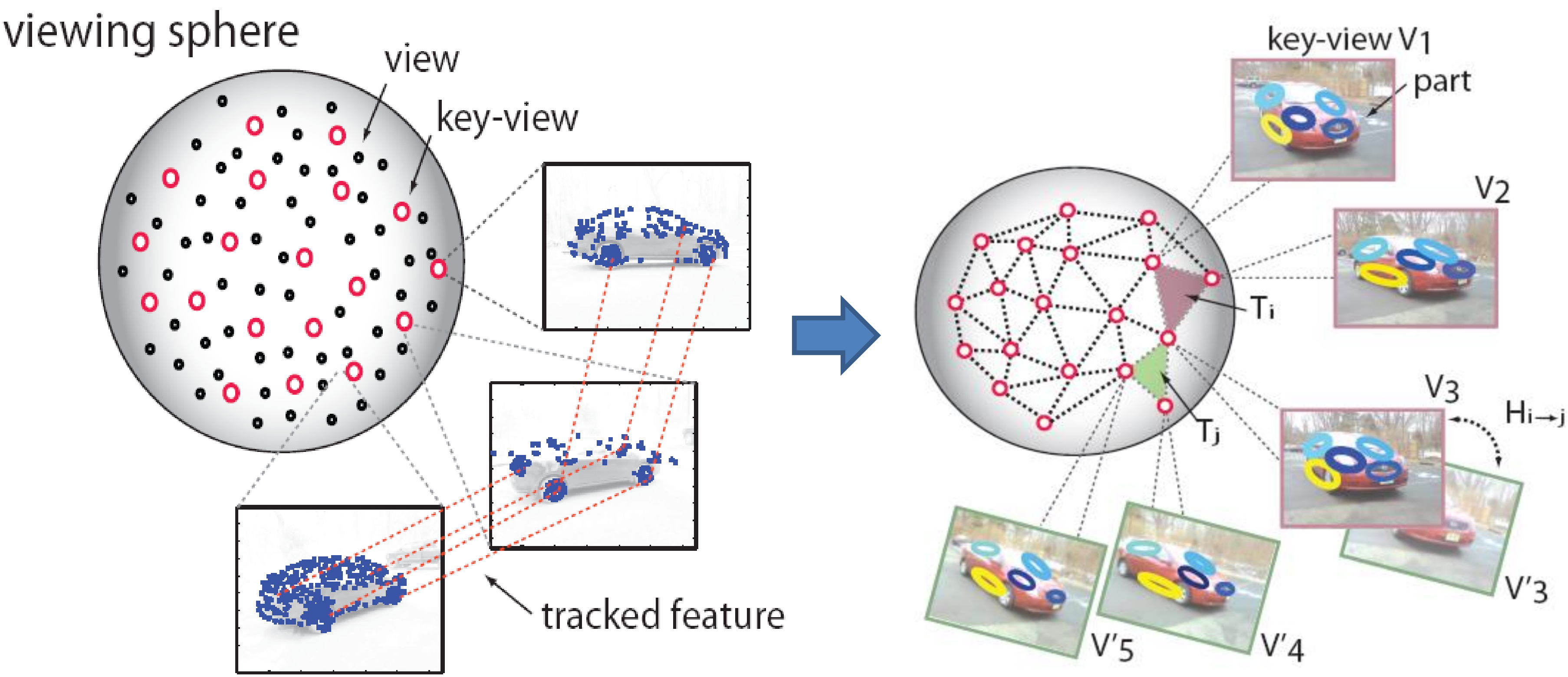
deformation
models

Felzenszwalb et al. 2008

Specifying an object model

4. 3D-ish model

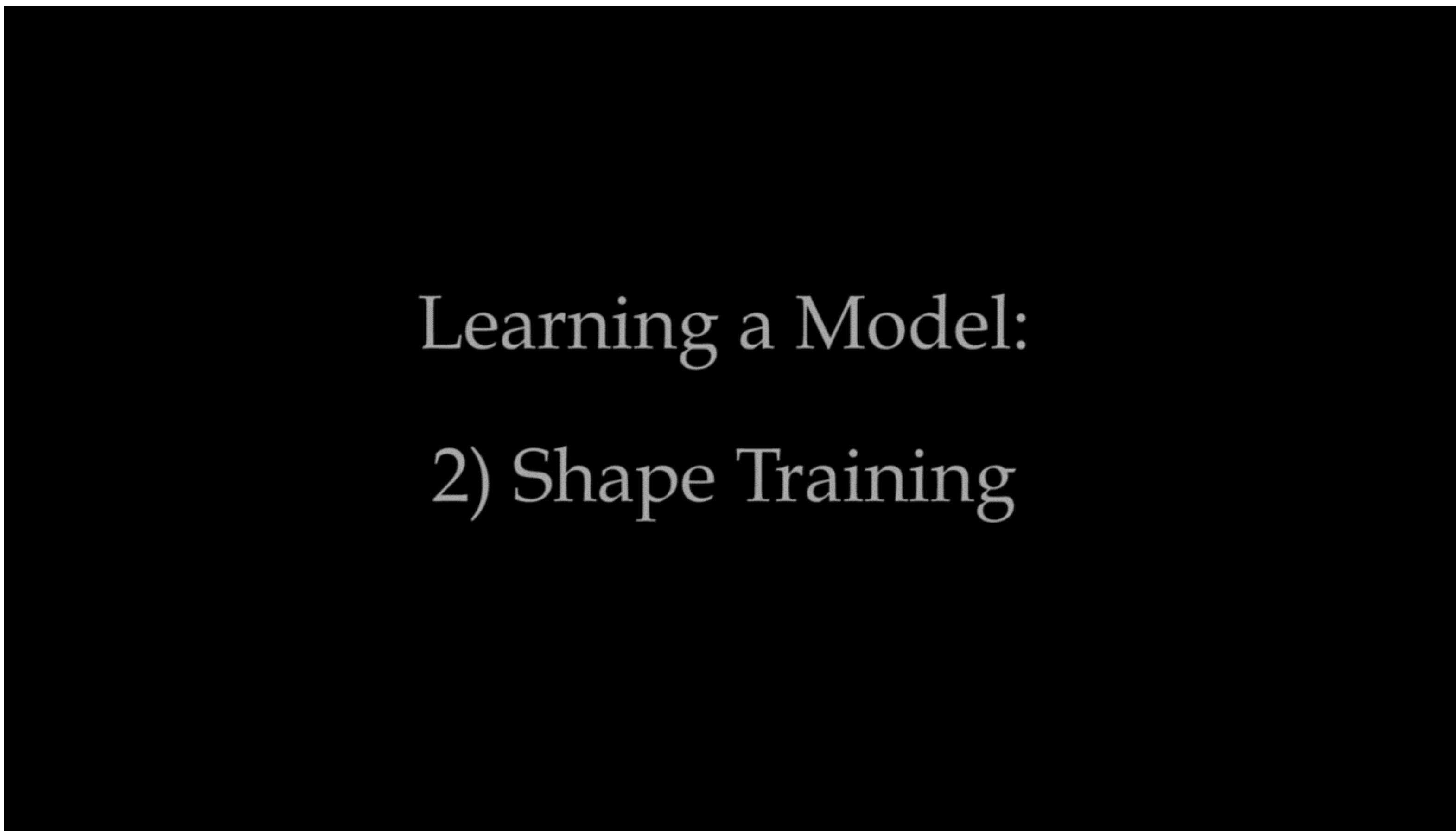
- Object is collection of 3D planar patches under affine transformation



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

Specify Object Model

↓

Generate Hypotheses

↓

Score Hypotheses

↓

Resolve Detections

Propose an alignment of the model to the image

James Hays

Generating hypotheses

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

James Hays

Generating hypotheses

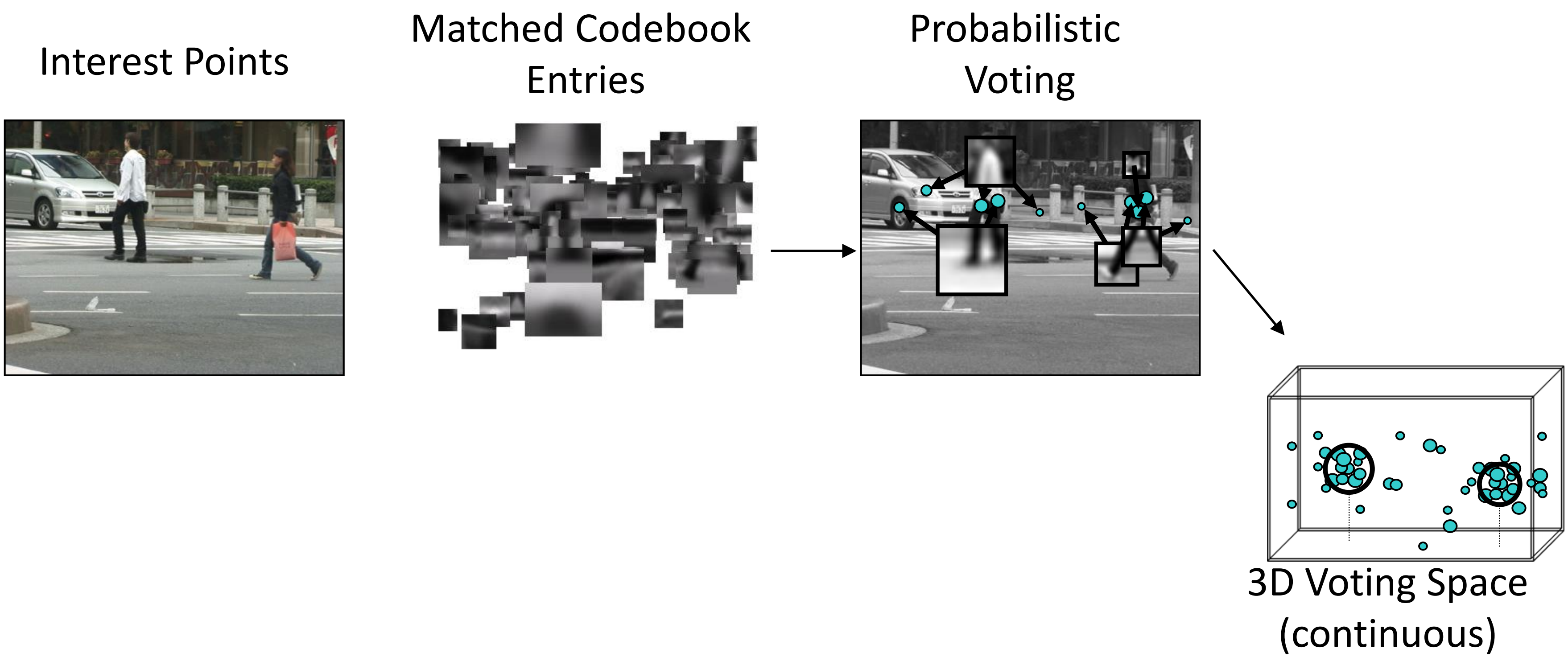
1. Sliding window
   – Test patch at each location and scale

Note – Template did not change size

Each window is separately classified

Generating hypotheses

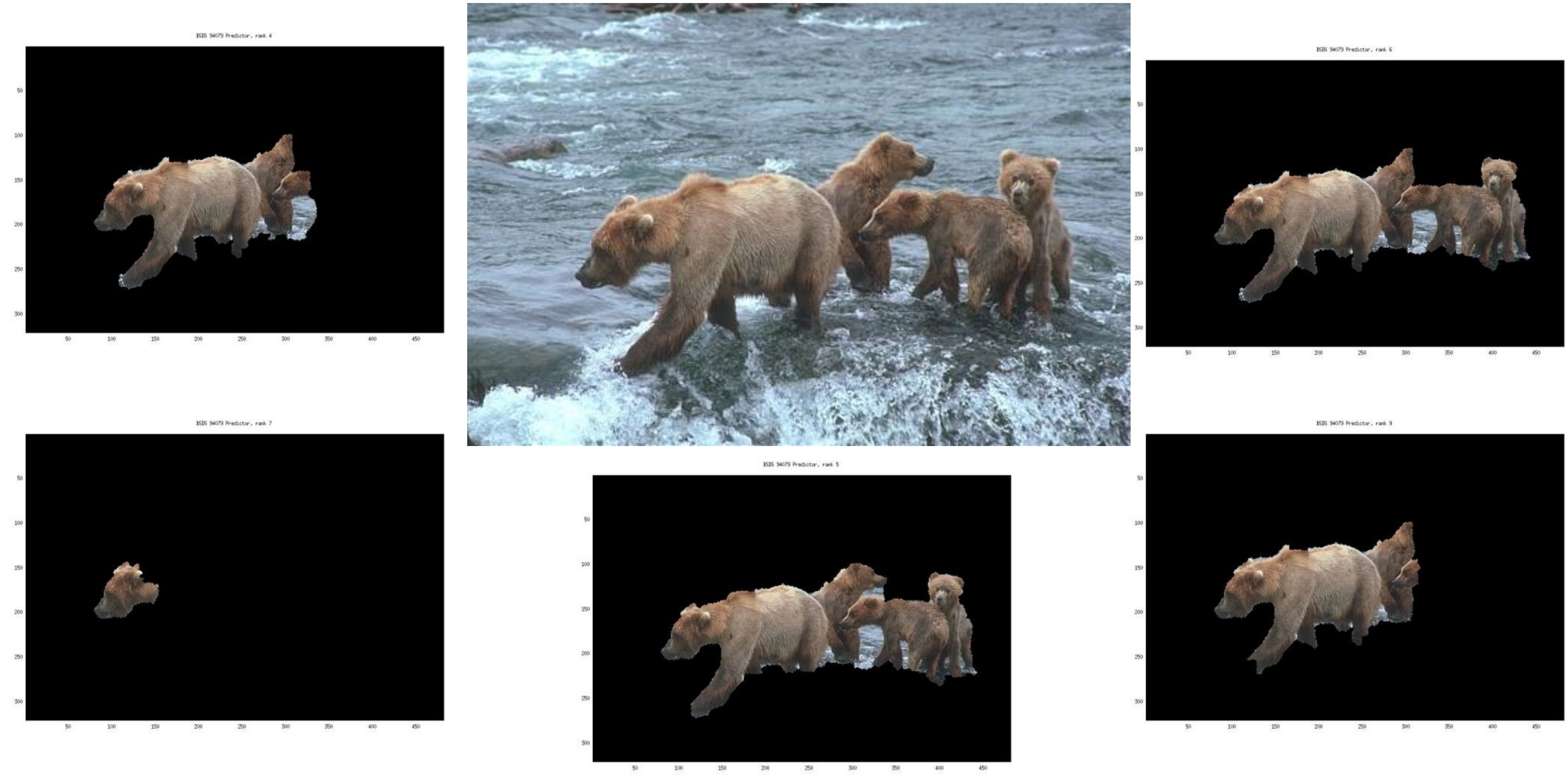
2. Voting from patches/keypoints



Implicit Shape Model by Leibe et al.

Generating hypotheses

3. Region-based proposal



Endres Hoiem 2010

General Process of Object Recognition

Specify Object Model

↓

Generate Hypotheses

↓

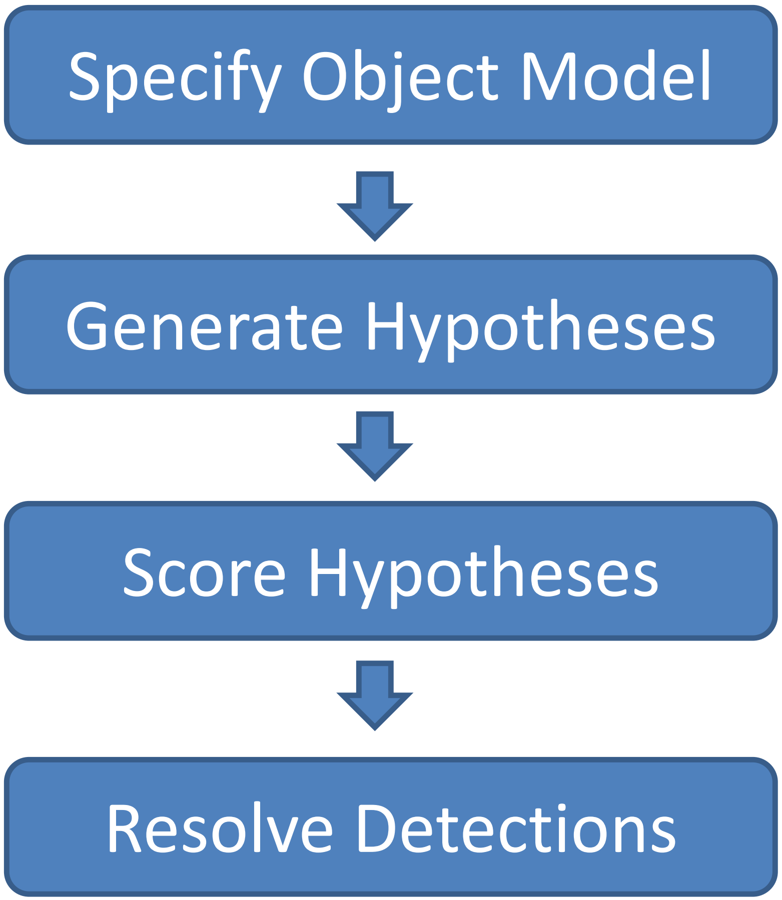
Score Hypotheses

↓

Resolve Detections

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

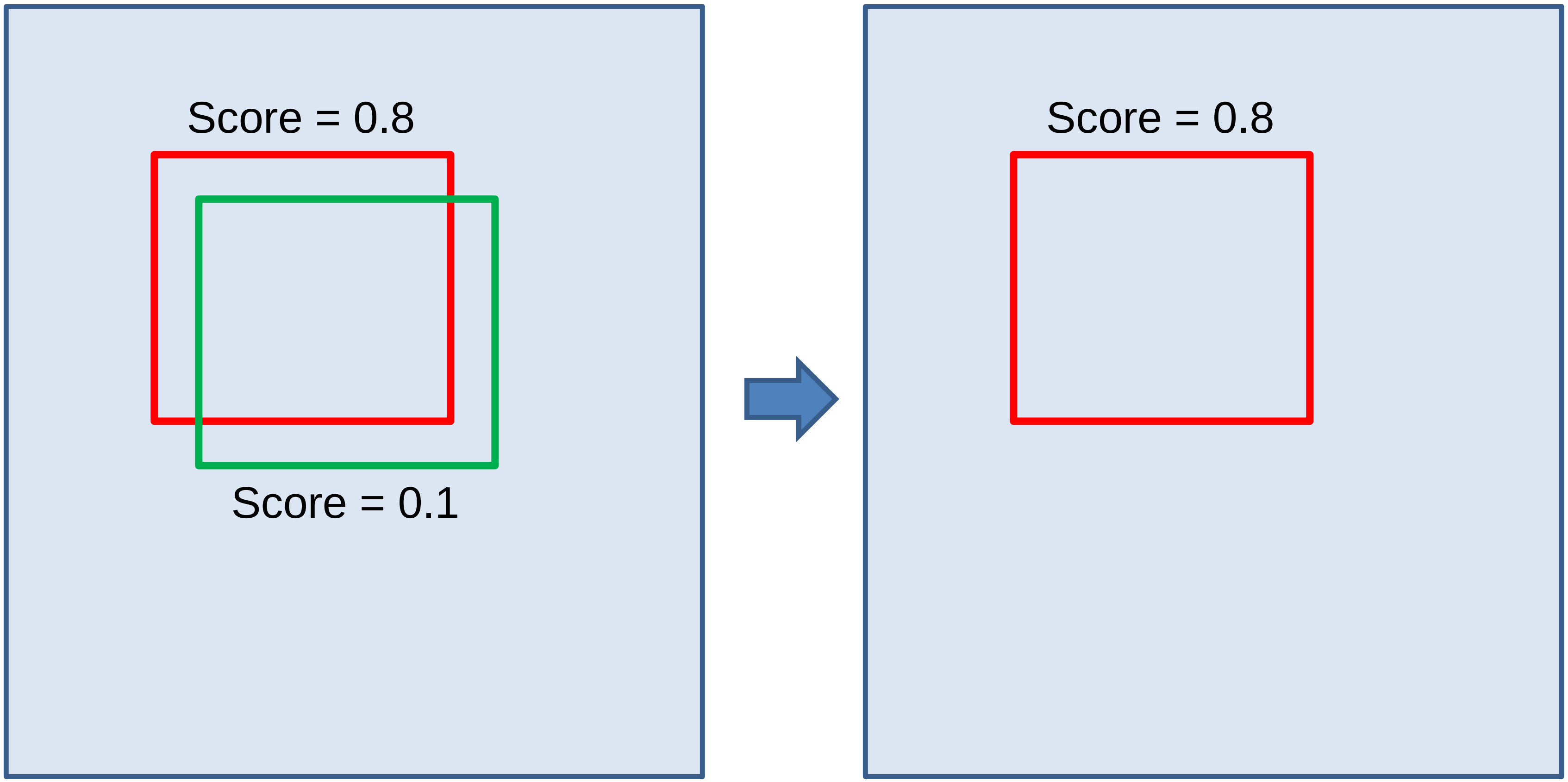
General Process of Object Recognition



Rescore each proposed object based on whole set

Resolving detection scores

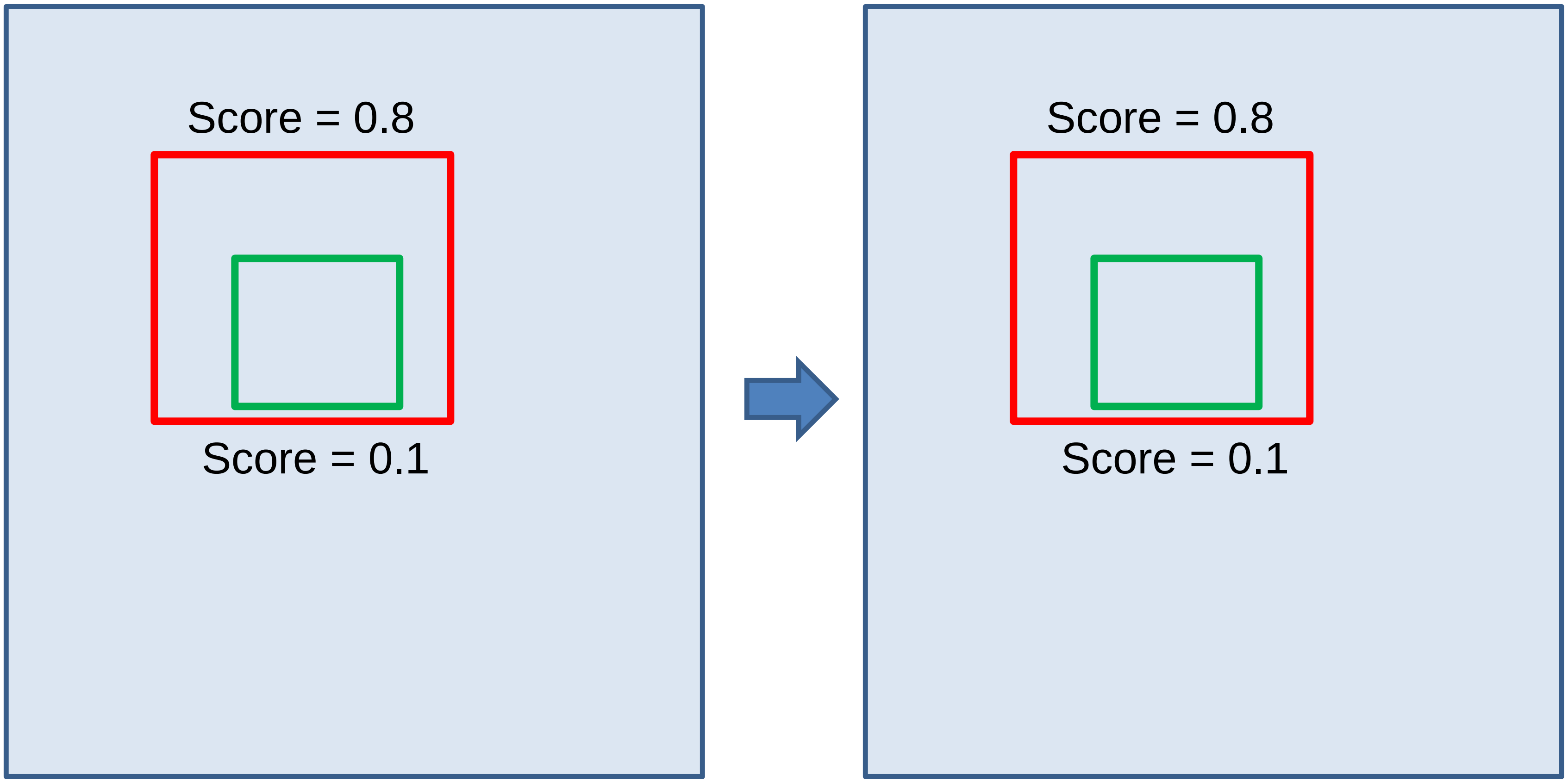
1. Non-max suppression



James Hays

Resolving detection scores

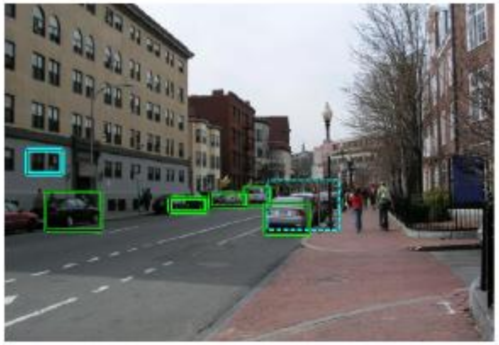
1. Non-max suppression



"Overlap" score is below some threshold

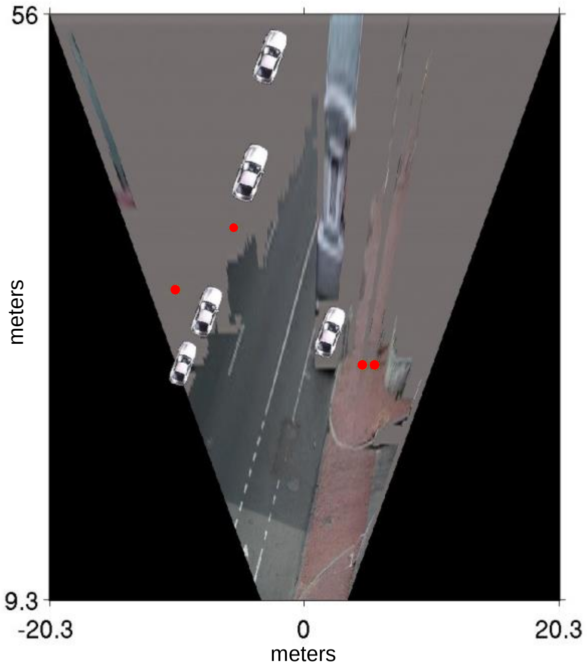
Resolving detection scores

2. Context/reasoning



(g) Car Detections: Local

(h) Ped Detections: Local



Hoiem et al. 2006

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/

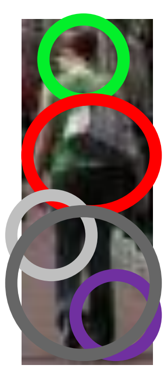
Statistical Template

Object model =

sum of scores of features at fixed positions

$$+3 +2 -2 -1 -2.5 = -0.5 \overset{?}{>} 7.5$$

Non-object



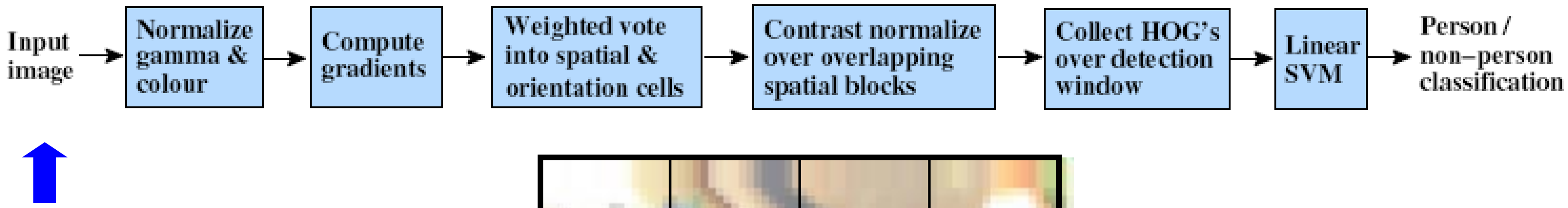
$$+4 +1 +0.5 +3 +0.5 = 10.5 \overset{?}{>} 7.5$$

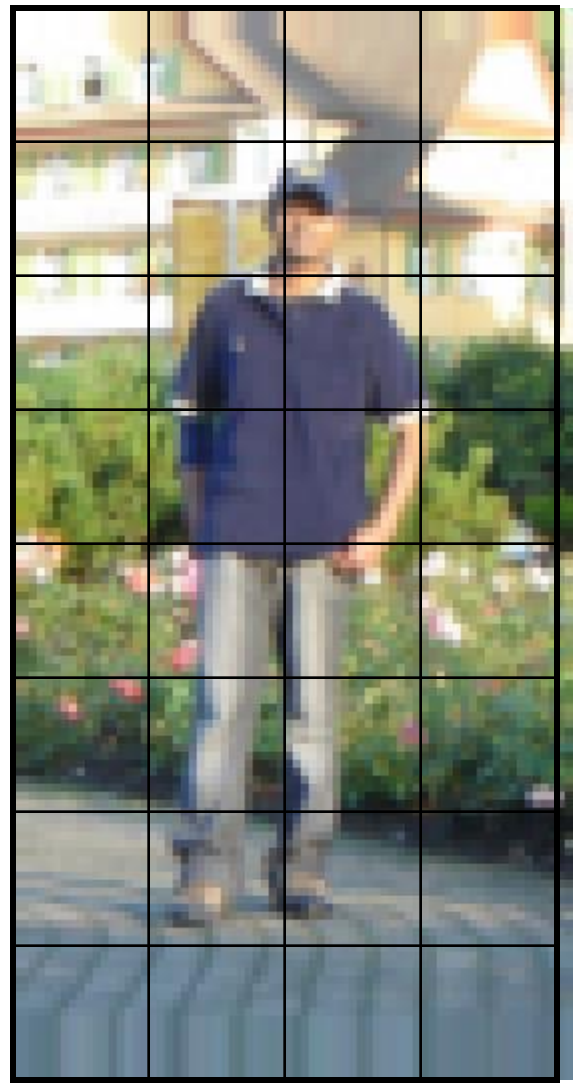
Object

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

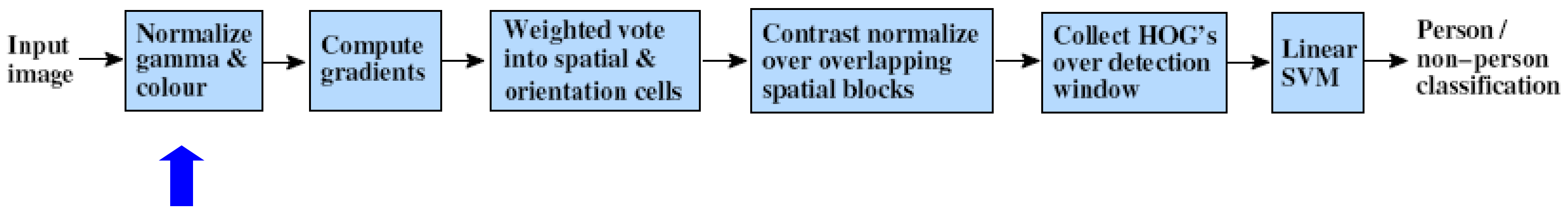
Navneet Dalal and Bill Triggs, Histograms of Oriented Gradients for Human Detection, CVPR05





Slides by Pete Barnum

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

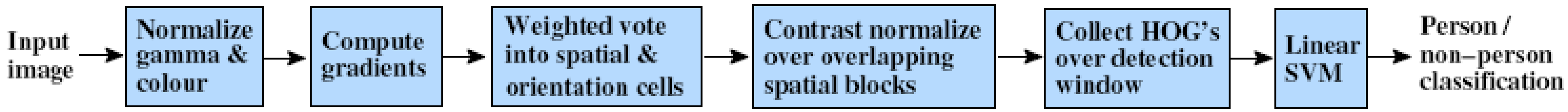


- Tested with
  - RGB
  - LAB
  - Grayscale

Slightly better performance vs. grayscale

- Gamma Normalization and Compression
  - Square root
  - Log

Very slightly better performance vs. no adjustment



Outperforms

-1	0	1

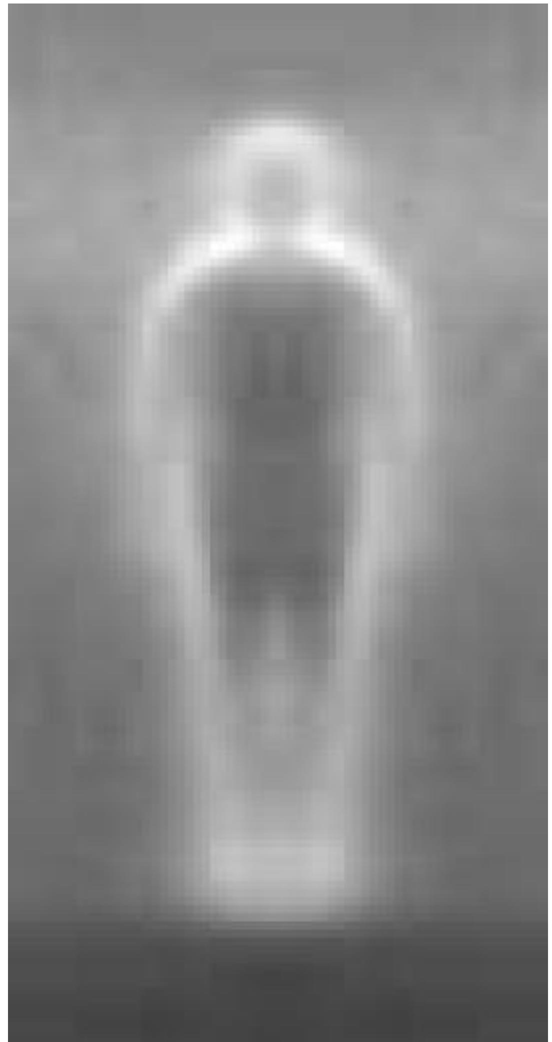
centered

-1	1

uncentered

1	-8	0	8	-1

cubic-corrected



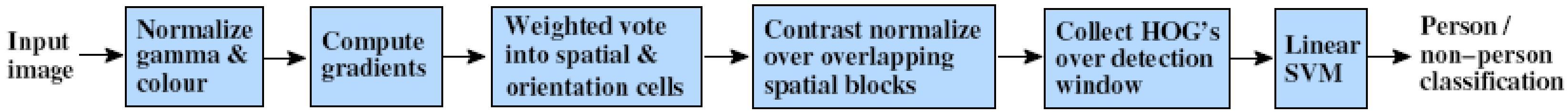
0	1
-1	0

diagonal

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

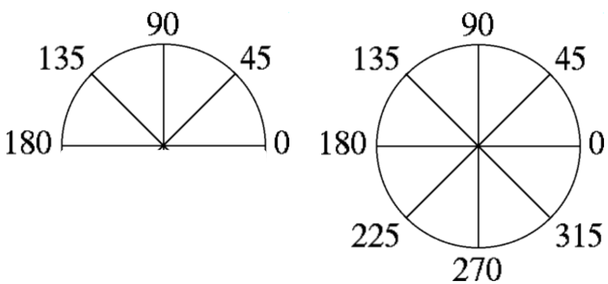
Sobel

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

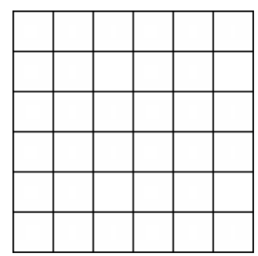


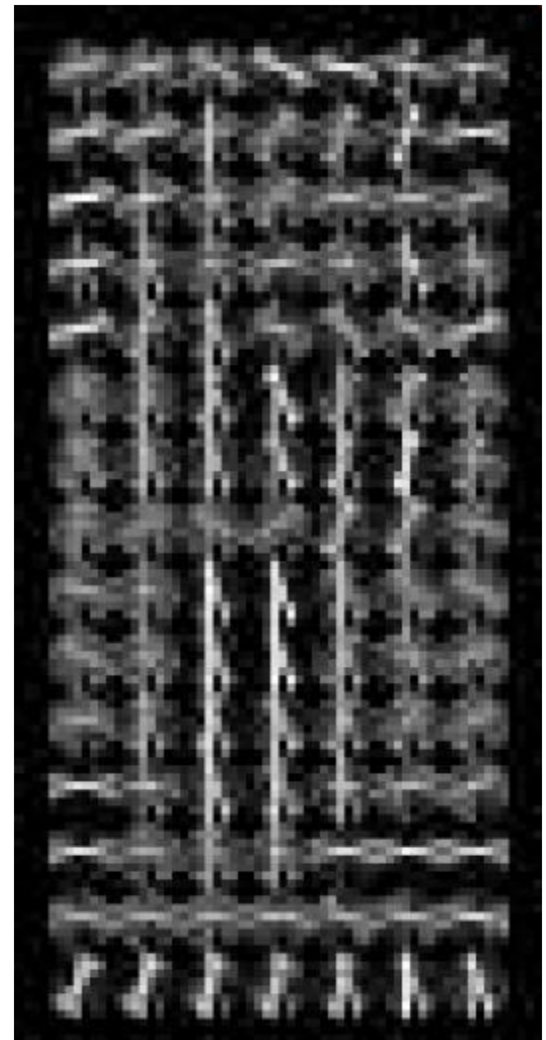
Histogram of Oriented Gradients

Orientation: 9 bins (for unsigned angles 0 -180)



Histograms in k x k pixel cells



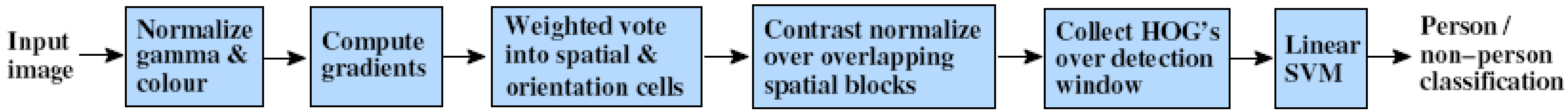


– Votes weighted by magnitude

– Bilinear interpolation between cells

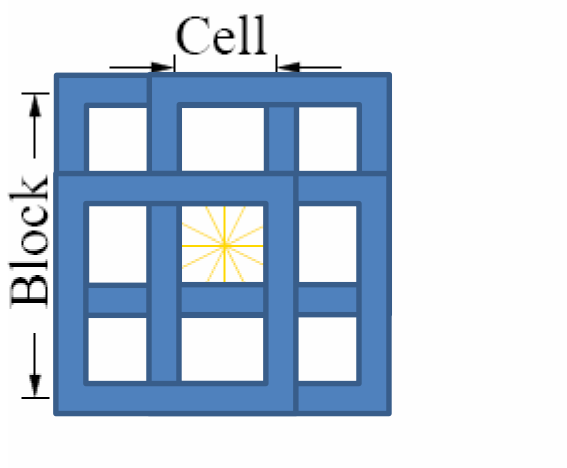
Slides by Pete Barnum

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



Normalize with respect to surrounding cells

Rectangular HOG (R-HOG)



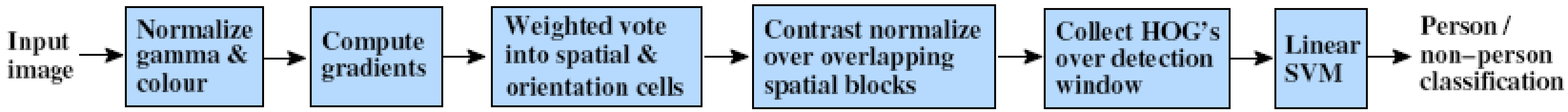
How to normalize?

- Concatenate all cell responses from block into vector.
- Normalize vector.
- Extract responses from cell of interest.

$$f = \frac{v}{\sqrt{\|v\|_2^2 + e^2}}$$

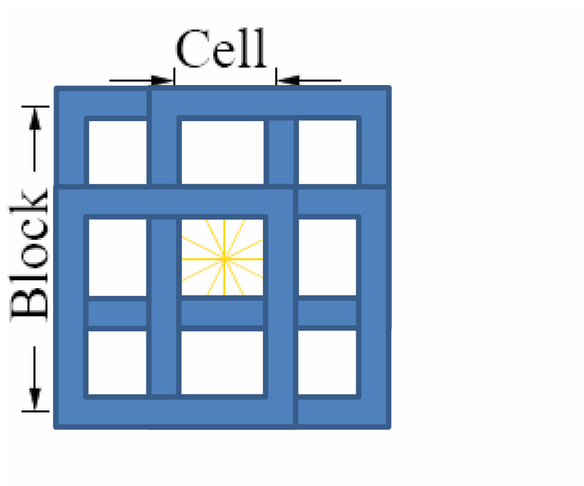
e is a small constant
(for empty bins)

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



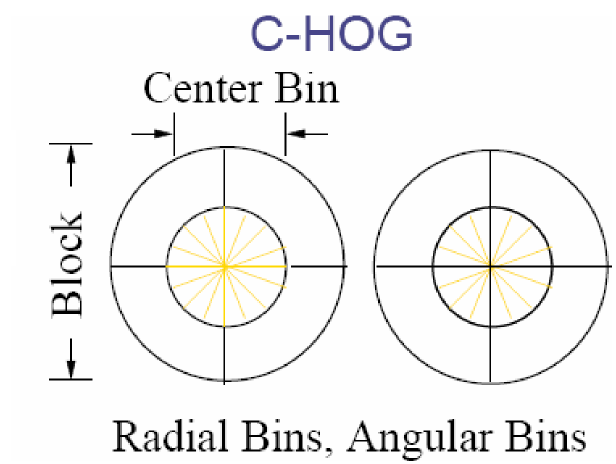
Normalize with respect to surrounding cells

Rectangular HOG (R-HOG)



$$f = \frac{v}{\sqrt{\|v\|_2^2 + e^2}}$$

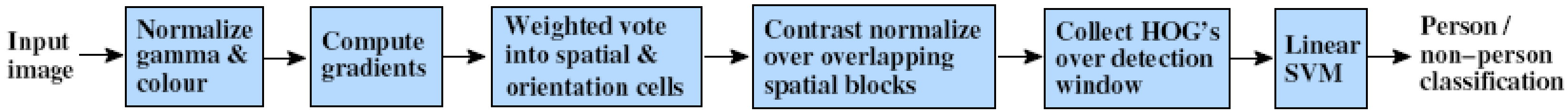
Circular HOG also exist, but trickier implementation

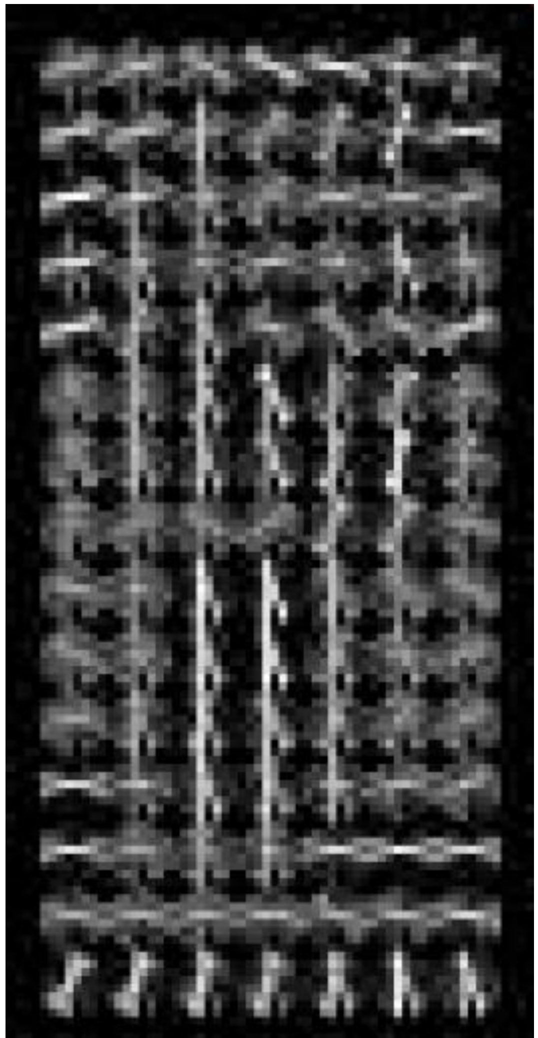


e is a small constant
(for empty bins)

Slides by Pete Barnum

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



X= 

orientations

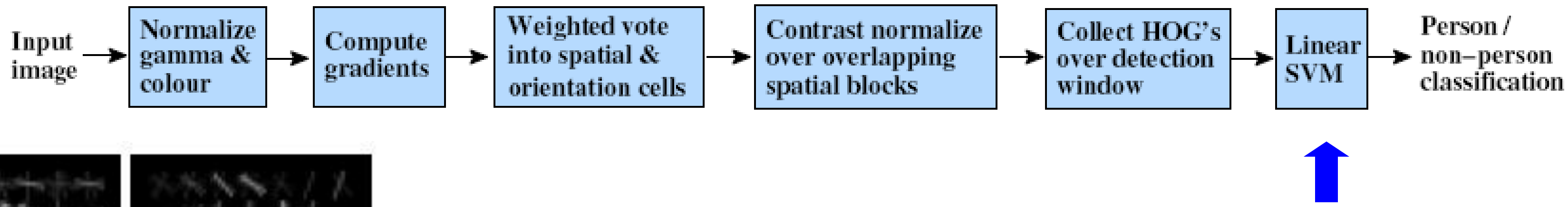
features = 15 x 7 x 9 x 4 = 3780

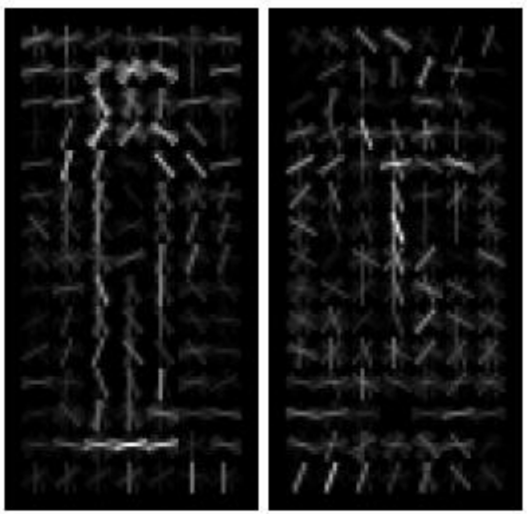
cells

normalizations by neighboring cells

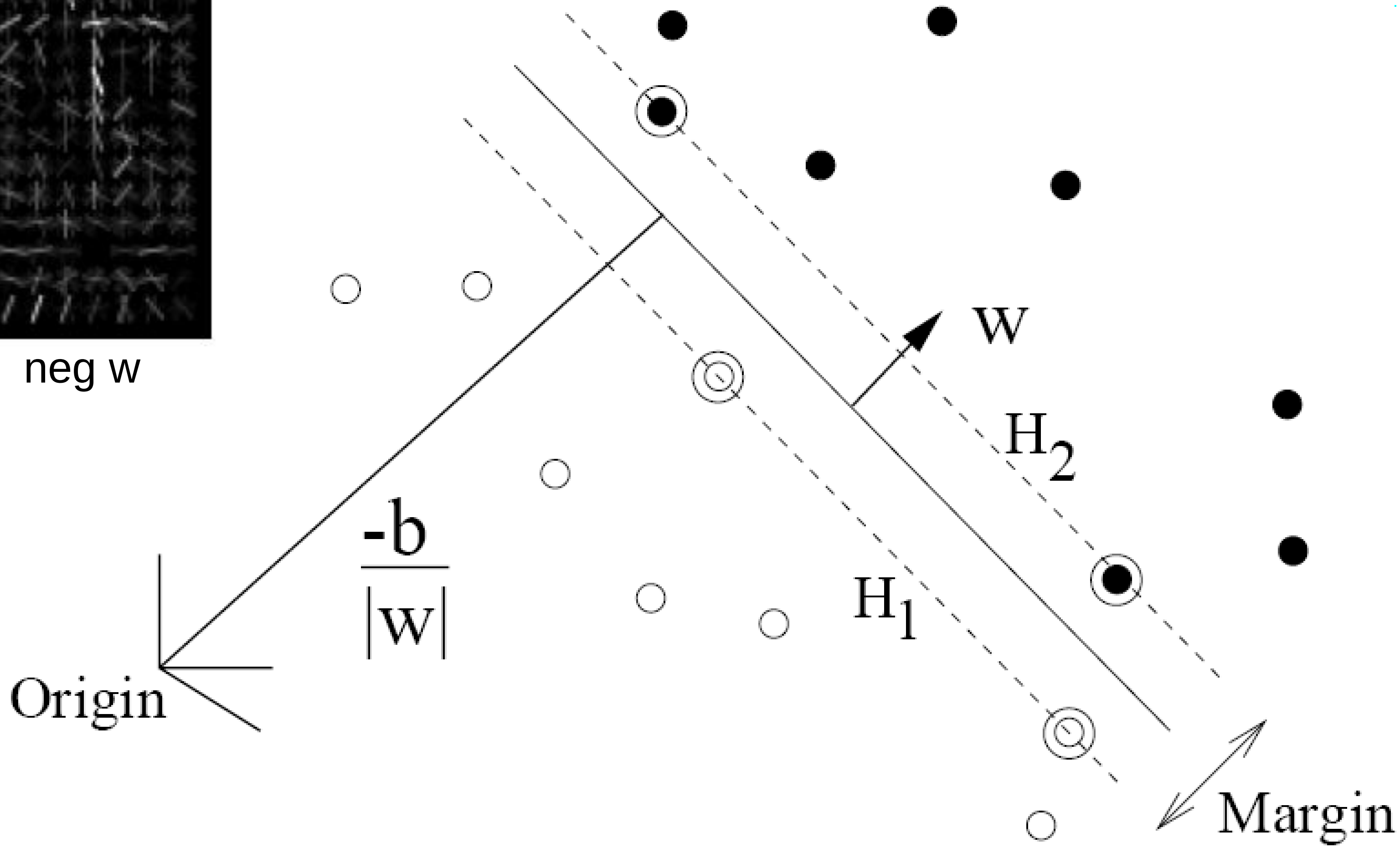
Slides by Pete Barnum

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

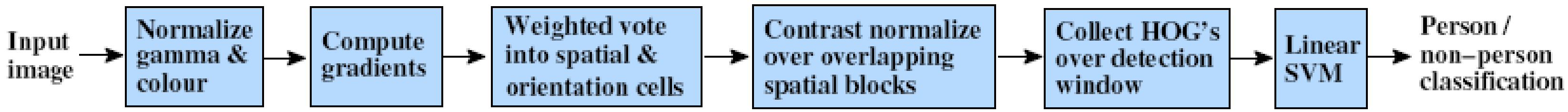




pos w   neg w



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



$$0.16 = w^T x - b$$

$$sign(0.16) = 1$$

$=>$ pedestrian

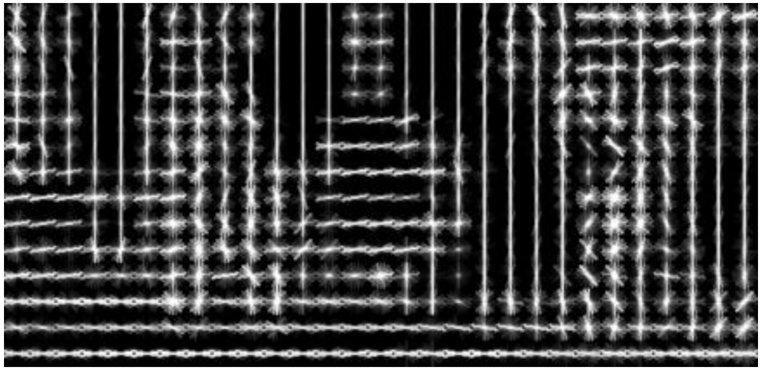
Slides by Pete Barnum

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

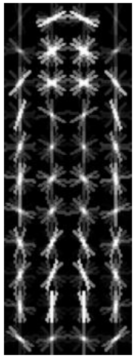
Pedestrian detection with HOG

- 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

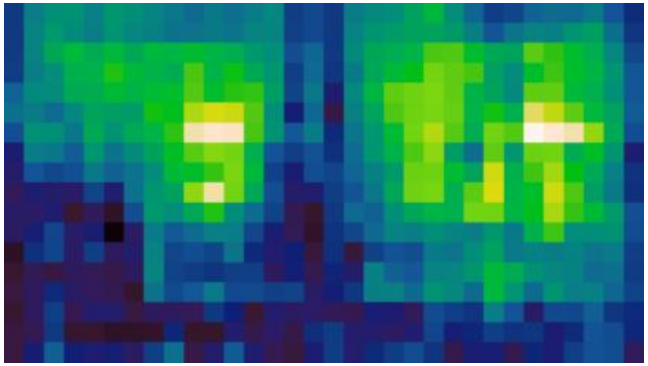
HOG feature map



Template



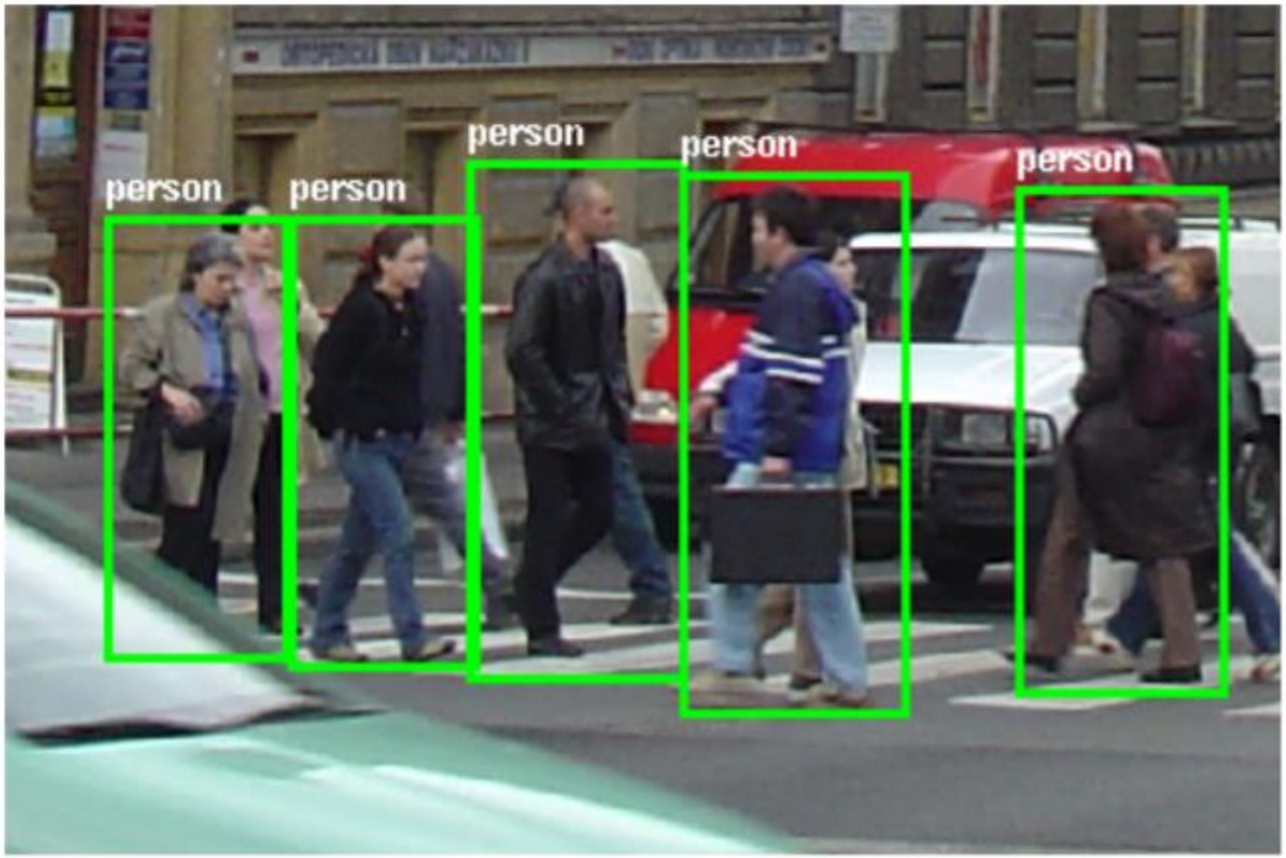
Detector response map



Can be continuous for more sophisticated maxima finding

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

INRIA pedestrian database

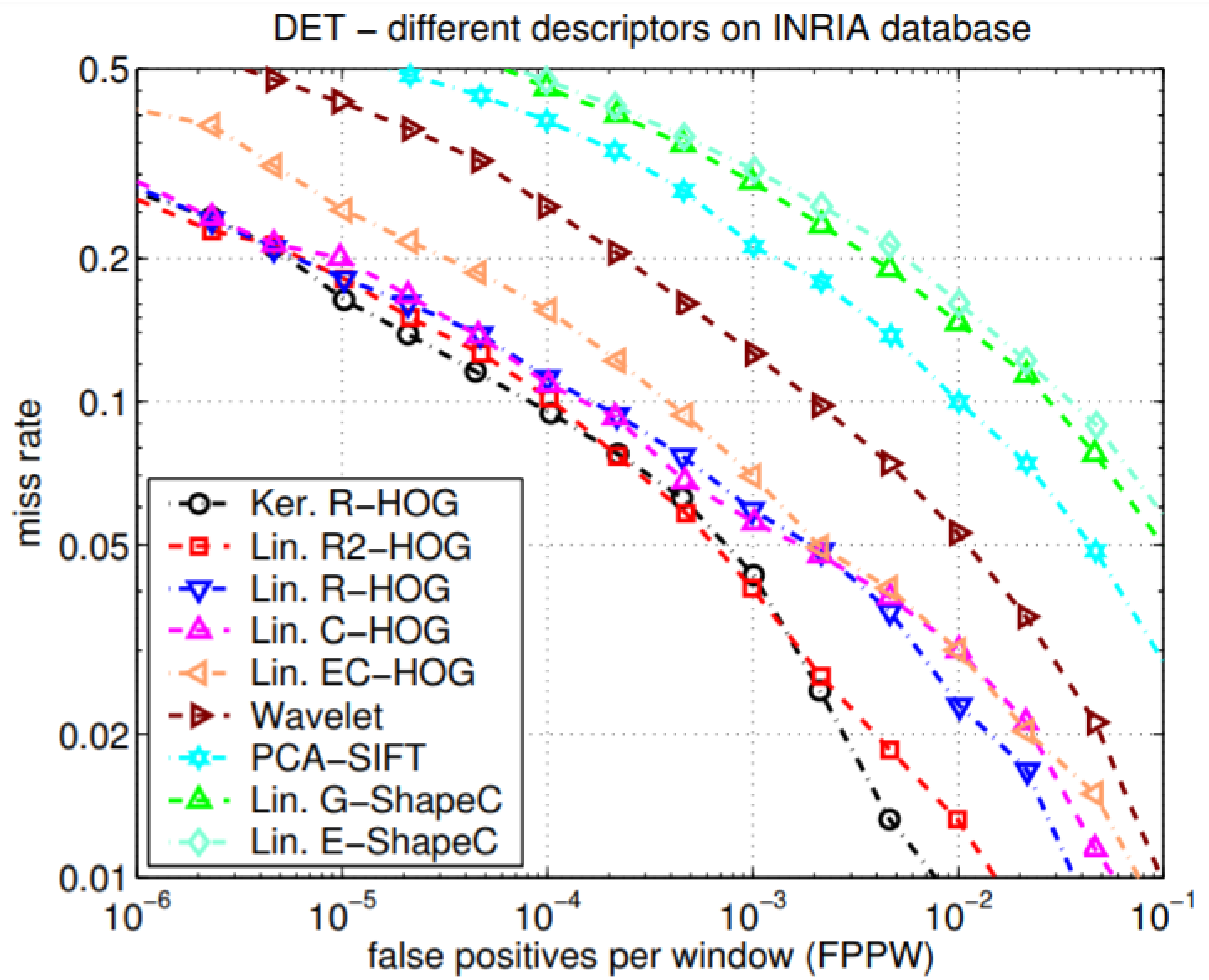


INRIA pedestrian database issues

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

Tricks of the trade

- Details in feature computation really matter
  - E.g., normalization in Dalal-Triggs improves detection rate by 27% at fixed false positive rate
- Template size
  - Typical choice is size of smallest expected detectable object
- “Jittering” or “augmenting” to create synthetic positive examples
  - Create slightly rotated, translated, scaled, mirrored versions as extra positive examples.
- Bootstrapping to get hard negative examples
  1. Randomly sample negative examples
  2. Train detector
  3. Sample negative examples that score > -1
  4. Repeat until all high-scoring negative examples fit in memory