Pedestrian Detection

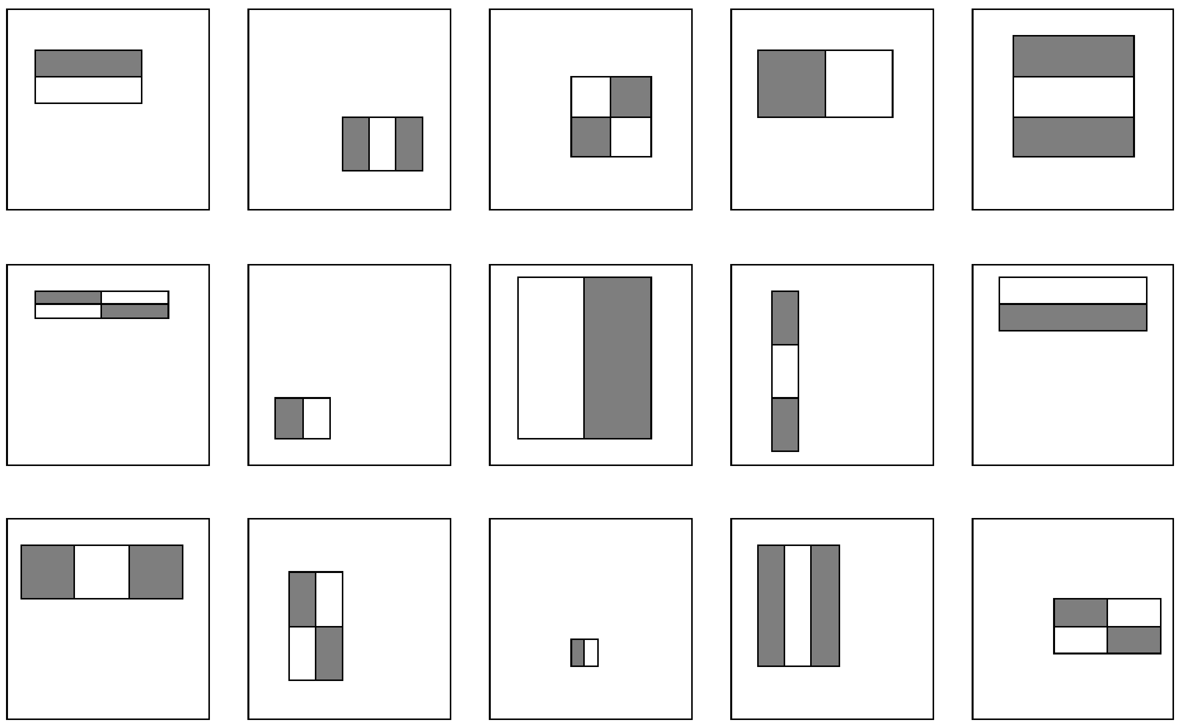
Vinay P. Namboodiri

- Slide credits to Navneet Dalal

Last Class

Feature selection

- For a 24x24 detection region, the number of possible rectangle features is ~160,000!

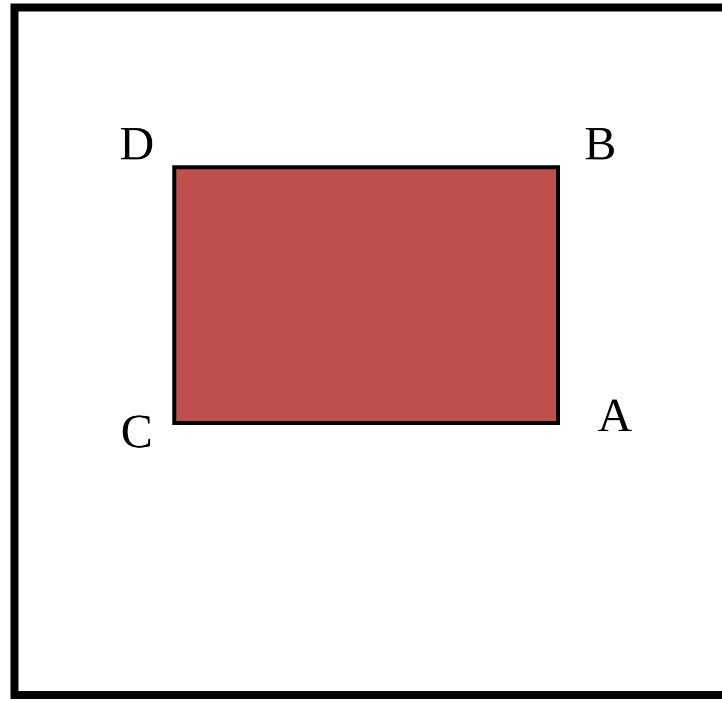


Last Class

Computing sum within a rectangle

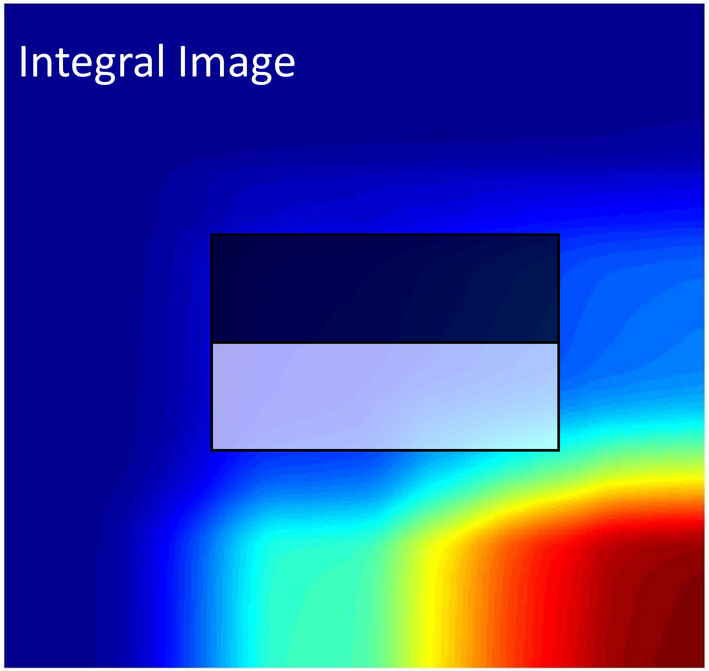
- Let A,B,C,D be the values of the integral image at the corners of a rectangle
- Then the sum of original image values within the rectangle can be computed as:

  sum = A – B – C + D
- Only 3 additions are required for any size of rectangle!



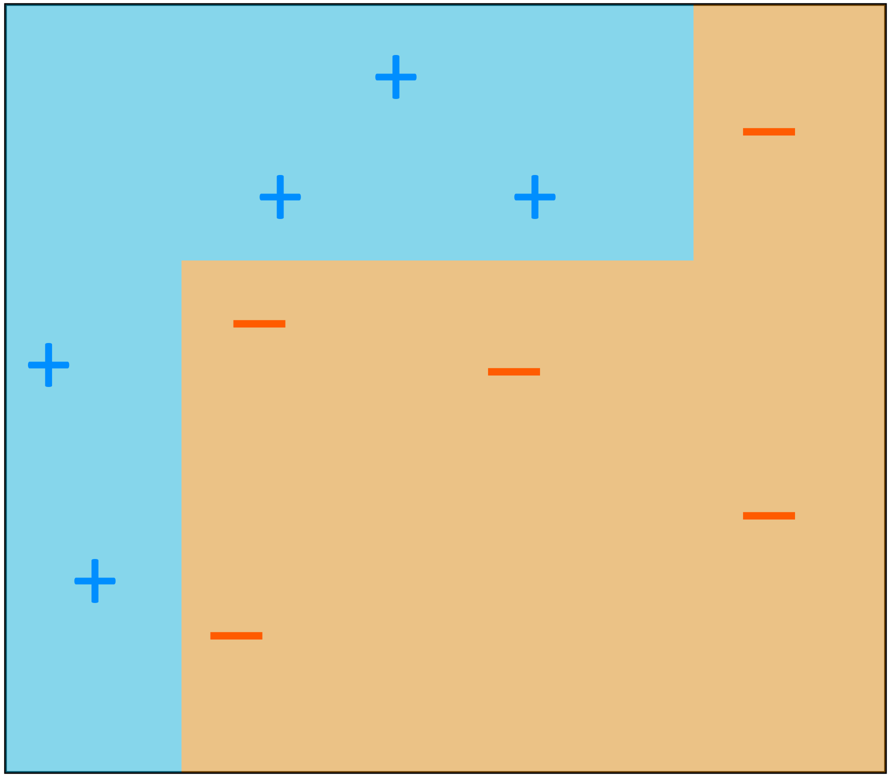
Last Class

Example



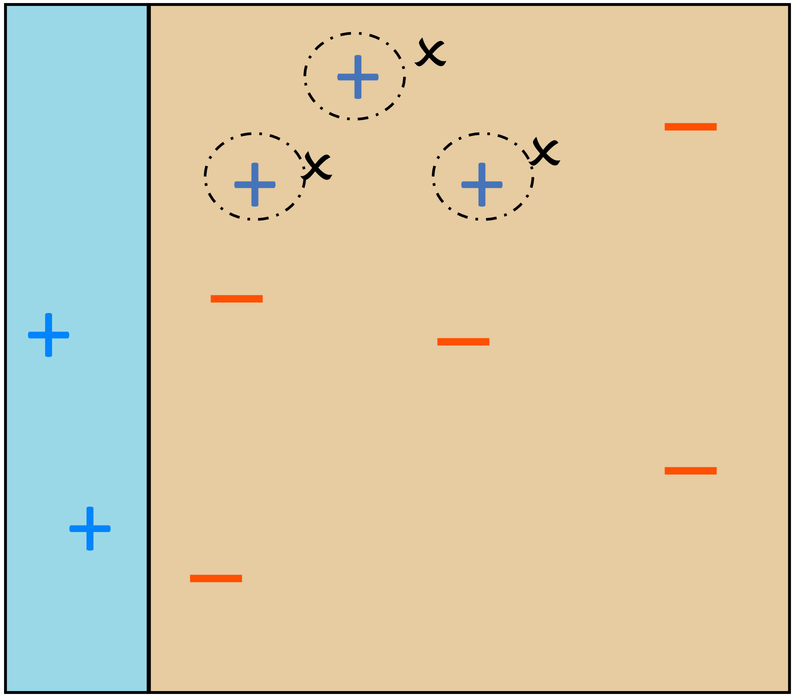
Last Class

Example of a Good Classifier



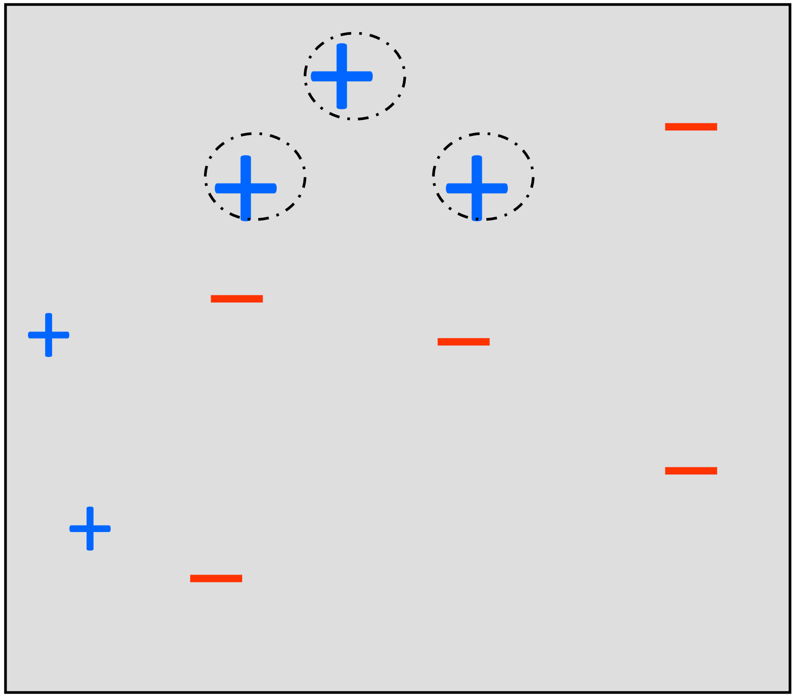
Last Class

Round 1 of 3



h_1 $\varepsilon_1 = 0.300$

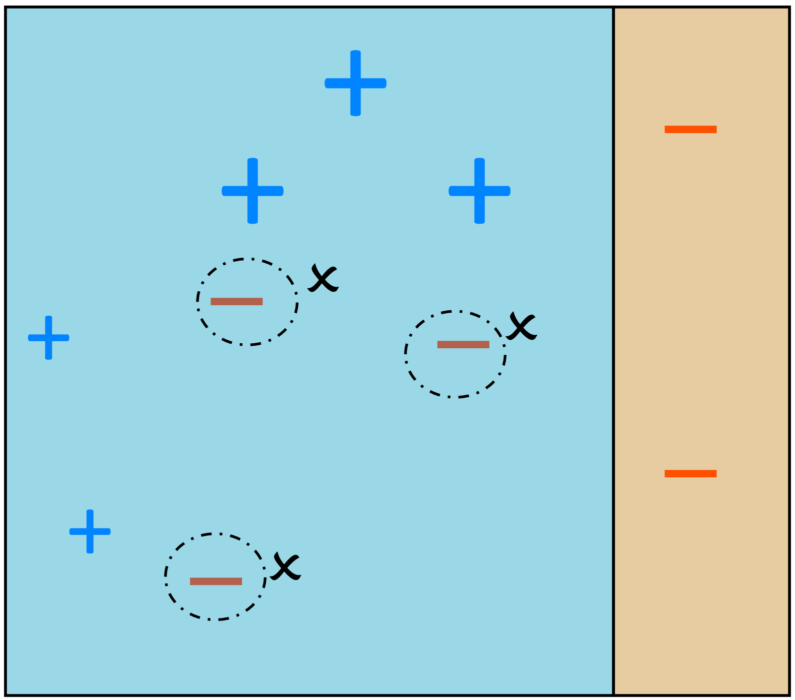
$\alpha_1 = 0.424$



D_2

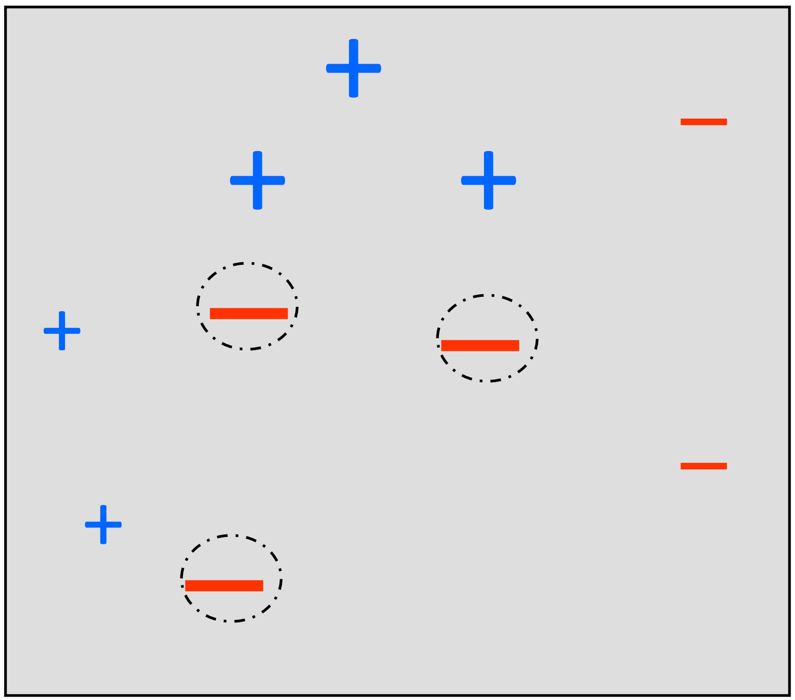
Last Class

Round 2 of 3



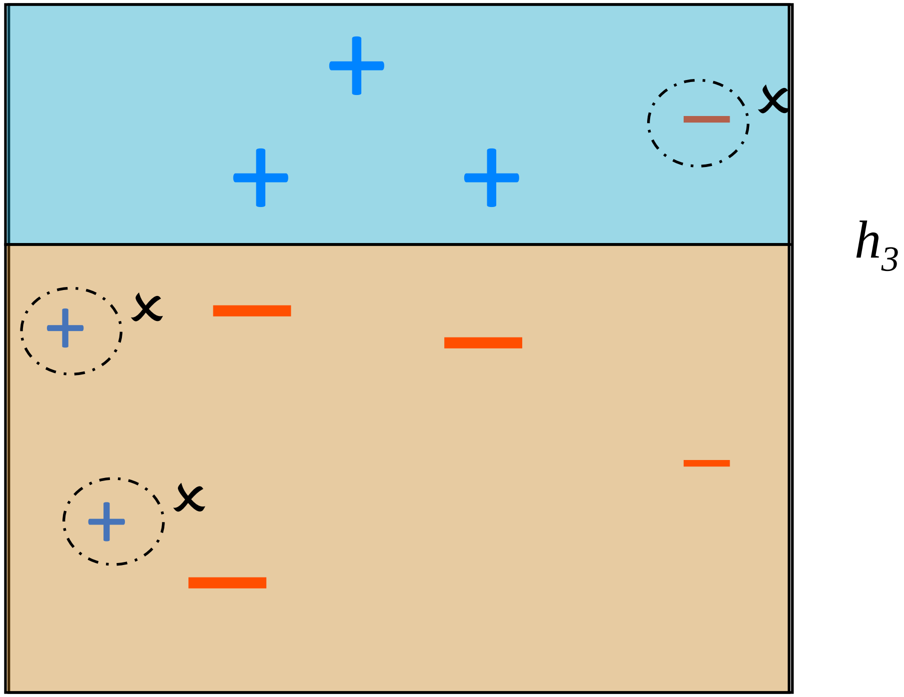
$\varepsilon_2 = 0.196$ h_2

$\alpha_2 = 0.704$



D_2

Round 3 of 3



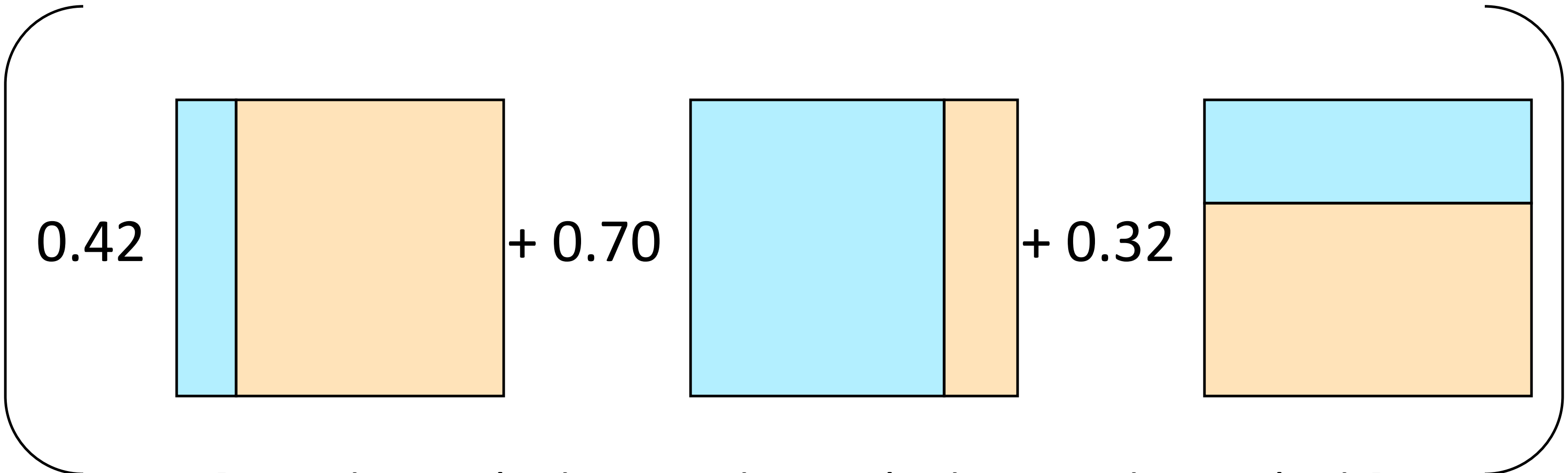
ε_3 = 0.344

α_2=0.323

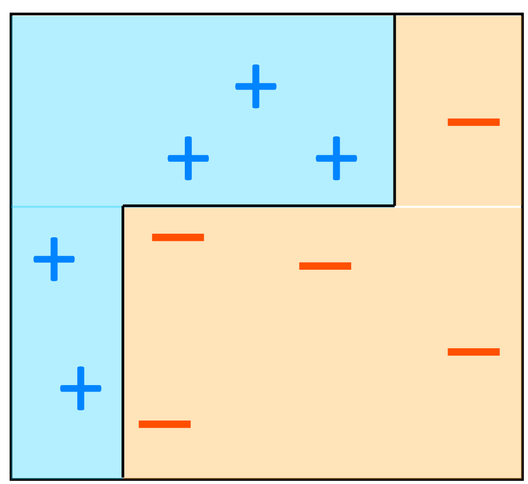
STOP

Last Class

Final Hypothesis



H_{final} = sign[0.42(h1? 1|-1) + 0.70(h2? 1|-1) + 0.32(h3? 1|-1)]



Last Class

AdaBoost

Given: m examples $(x_1, y_1), \ldots, (x_m, y_m)$ where $x_i \in X, y_i \in Y = \{-1, +1\}$

Initialize $D_1(i) = 1/m$

For t = 1 to T

1. Train learner $\boldsymbol{h_t}$ with min error $\varepsilon_t = \Pr_{i \sim D_t}[h_t(x_i) \neq y_i]$

2. Compute the hypothesis weight $\alpha_t = \frac{1}{2} \ln\left(\frac{1-\varepsilon_t}{\varepsilon_t}\right)$

3. For each example *i* = 1 to m

$$D_{t+1}(i) = \frac{D_t(i)}{Z_t} \times \begin{cases} e^{-\alpha_t} & \text{if } h_t(x_i) = y_i \\ e^{\alpha_t} & \text{if } h_t(x_i) \neq y_i \end{cases}$$

Output

$$H(x) = \operatorname{sign}\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right)$$

The goodness of h_t is calculated over D_t and the bad guesses.

The weight **Ada**pts. The bigger ε_t becomes the smaller α_t becomes.

Boost example if incorrectly predicted.

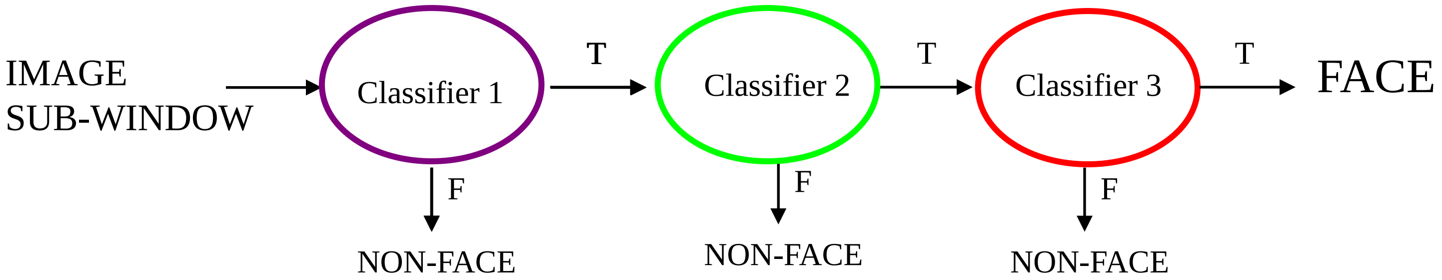
Z_t is a normalization factor.

Linear combination of models.

Last Class

Attentional cascade

- We start with simple classifiers which reject many of the negative sub-windows while detecting almost all positive sub-windows
- Positive response from the first classifier triggers the evaluation of a second (more complex) classifier, and so on
- A negative outcome at any point leads to the immediate rejection of the sub-window

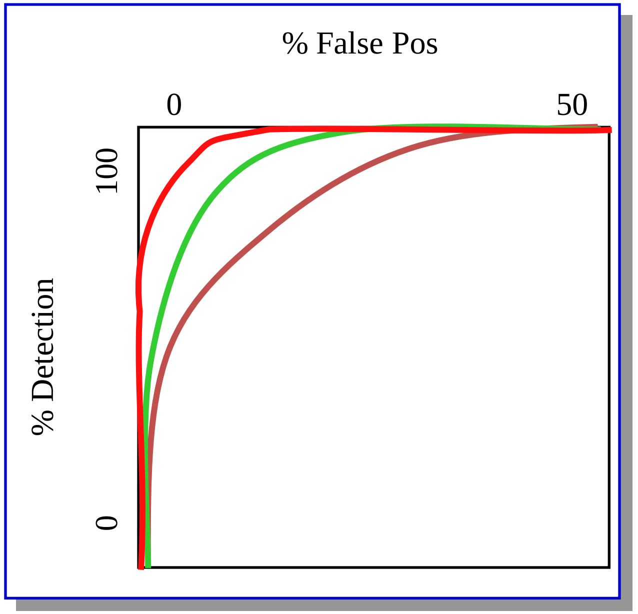


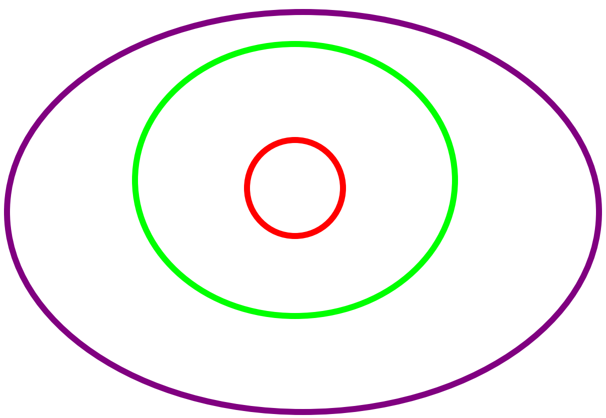
Last Class

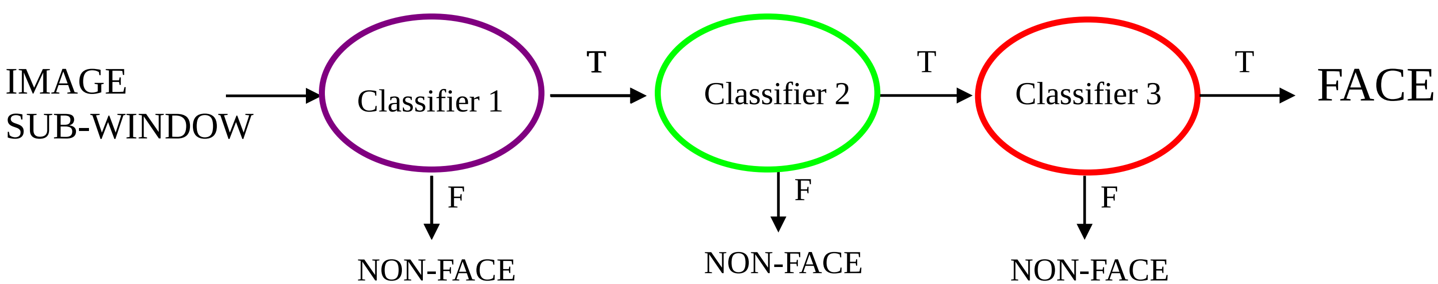
Attentional cascade

- Chain classifiers that are progressively more complex and have lower false positive rates:

Receiver operating characteristic

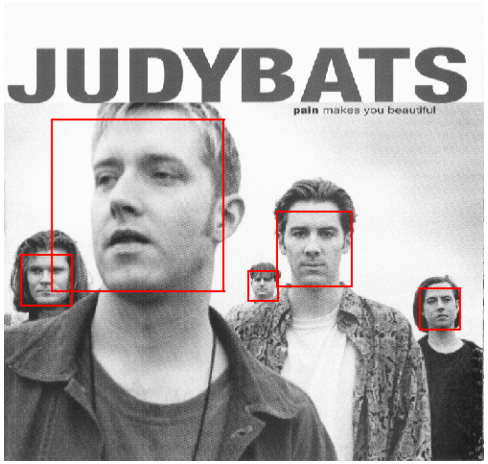


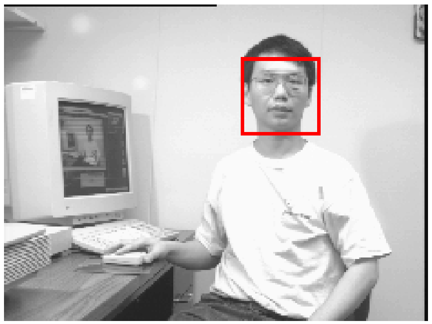


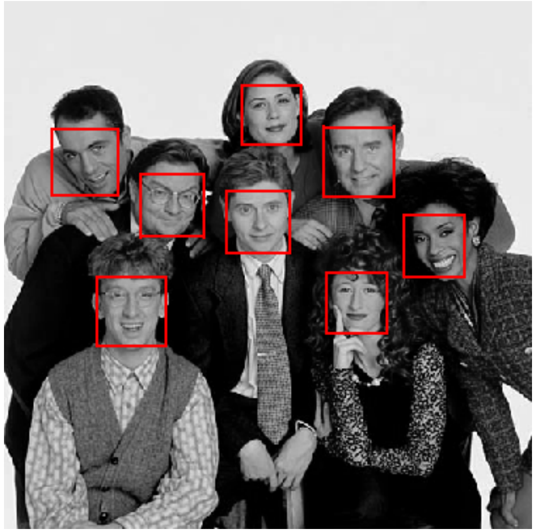


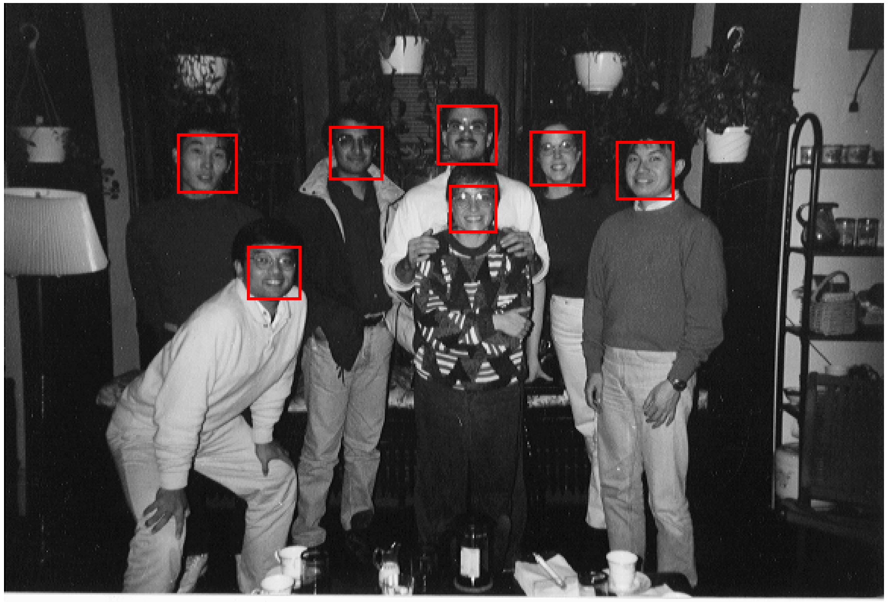
Output of Face Detector on Test Images

Last Class









Last Class

Summary: Viola/Jones detector

- Rectangle features
- Integral images for fast computation
- Boosting for feature selection
- Attentional cascade for fast rejection of negative windows

Finding People in Images

This Class

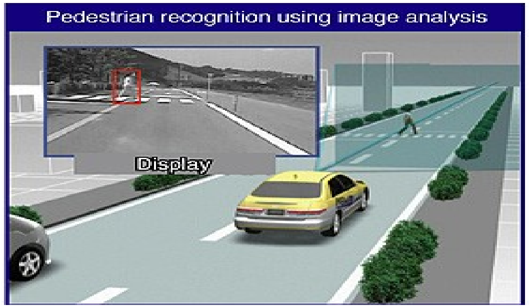
Method by Dalal and Triggs, CVPR 2005

Goals & Applications

Goal: Detect and localise people in images and videos

Applications:

- Images, films & multi-media analysis
- Pedestrian detection for smart cars
- Visual surveillance, behavior analysis



Difficulties

Wide variety of articulated poses

Variable appearance and clothing

Complex backgrounds

Unconstrained illumination

Occlusions, different scales

Videos sequences involves motion of the subject, the camera and the objects in the background

Main assumption: upright fully visible people

Overview of Methodology

Detection Phase

Scan image(s) at all scales and locations

↓

Extract features over windows

↓

Run linear SVM classifier on all locations

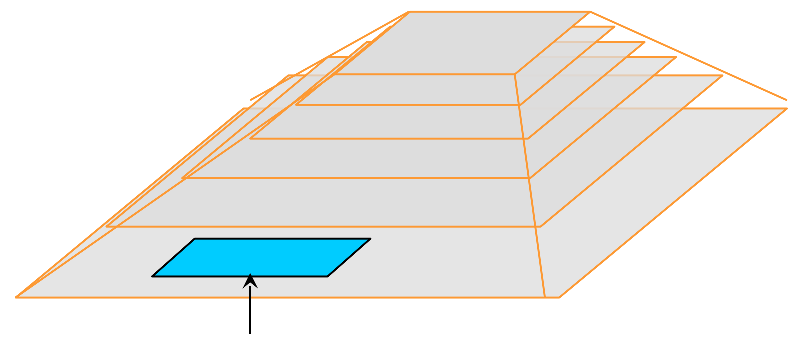
↓

Fuse multiple detections in 3-D position & scale space

↓

Object detections with bounding boxes

Scale-space pyramid

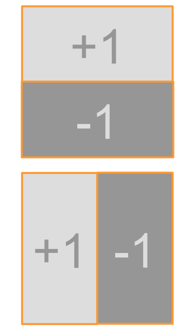


Detection window

Focus on building robust feature sets (static & motion)

Finding People in Images

Existing Person Detectors/Feature Sets



Current Approaches

Haar wavelets + SVM:

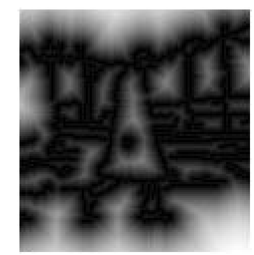
- Papageorgiou & Poggio, 2000; Mohan et al 2000

Rectangular differential features + adaBoost:

- Viola & Jones, 2001

Edge templates + nearest neighbour:

- Gavrila & Philomen, 1999



Model based methods

- Felzenszwalb & Huttenlocher, 2000; Ioffe & Forsyth, 1999

Other works

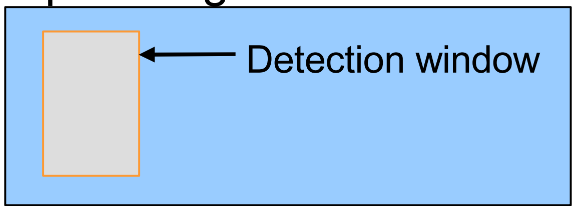
- Leibe et al, 2005; Mikolajczyk et al, 2004

Orientation histograms

Freeman et al, 1996; Lowe, 1999 (SIFT); Belongie et al, 2002 (Shape contexts)

Static Feature Extraction

Input image



Normalise gamma

↓

Compute gradients

↓

Weighted vote in spatial & orientation cells

↓

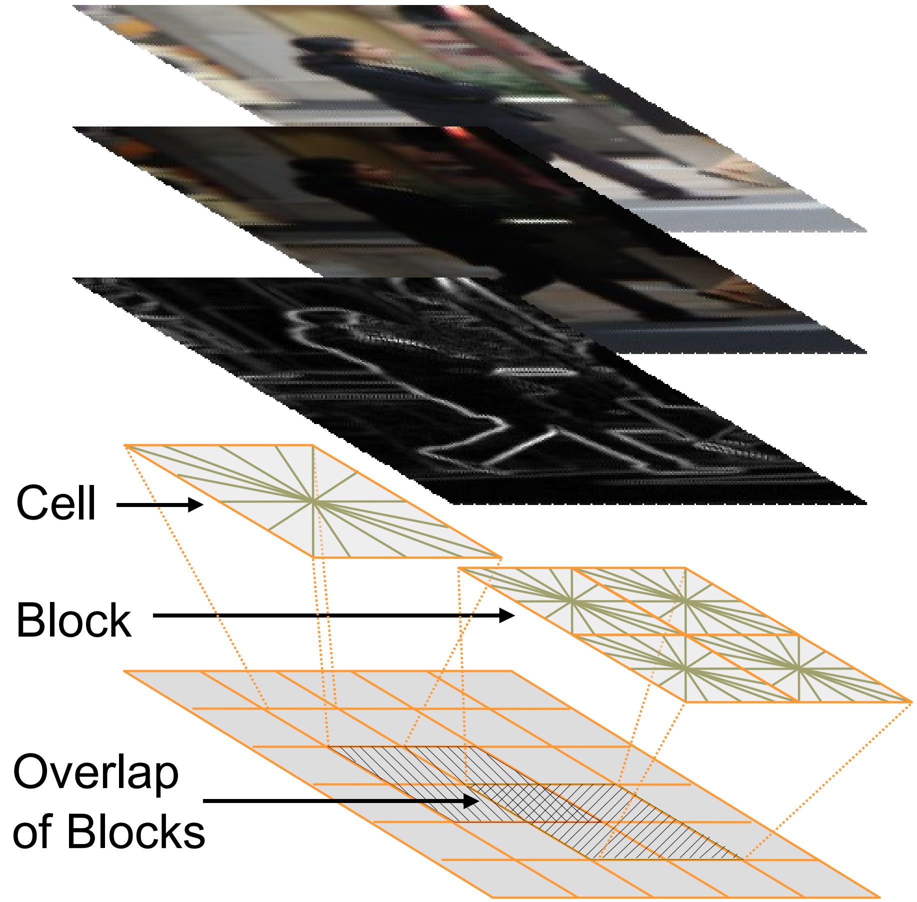
Contrast normalise over overlapping spatial cells

↓

Collect HOGs over detection window

↓

Linear SVM



Feature vector f = [..., ..., ...]

Overview of Learning Phase

Learning phase

Input: Annotations on training images

Create fixed-resolution normalised training image data set

Encode images into feature spaces

Learn binary classifier

Resample negative training images to create hard examples

Encode images into feature spaces

Learn binary classifier

Object/Non-object decision

Retraining reduces false positives by an order of magnitude!

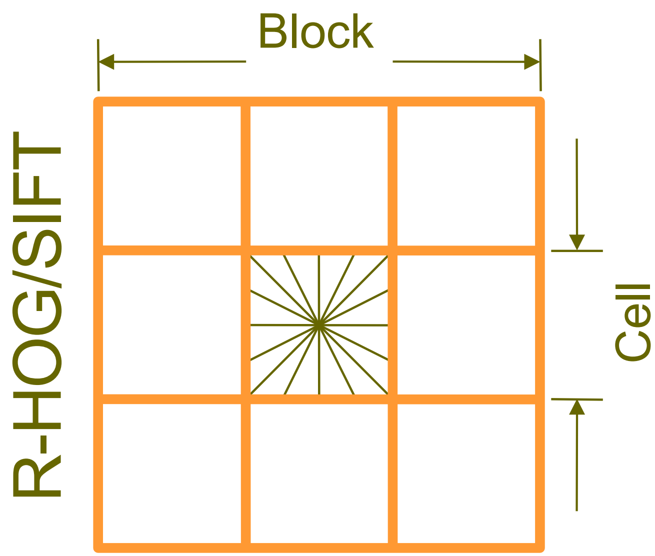
HOG Descriptors

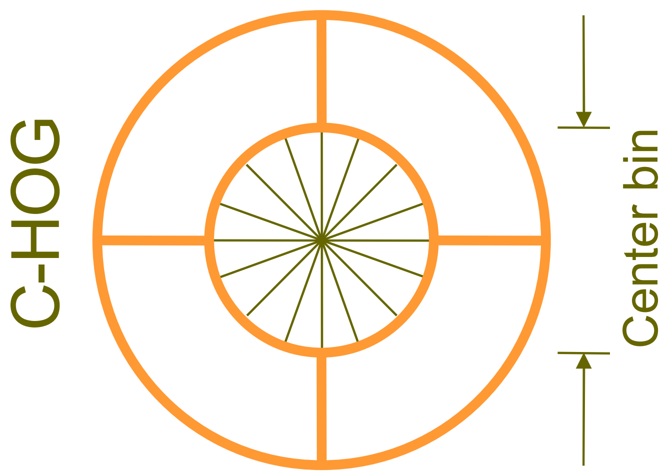
Parameters

- Gradient scale
- Orientation bins
- Percentage of block overlap

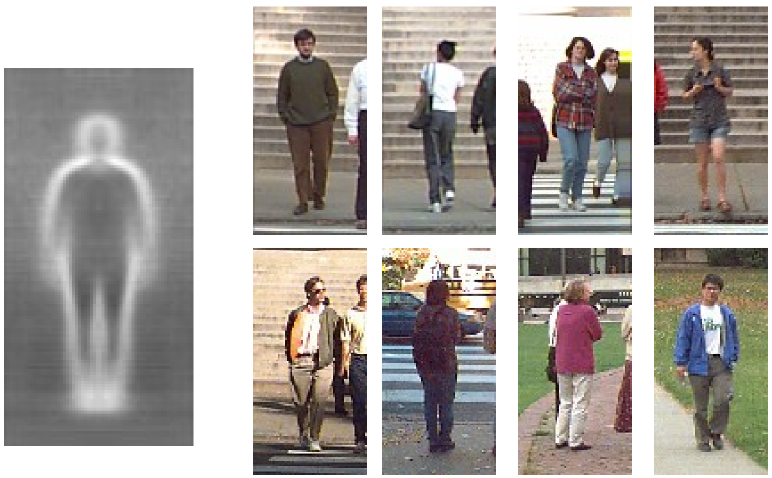
Schemes

- RGB or Lab, colour/gray-space
- Block normalisation
  - *L2*-norm, $v \leftarrow v / \sqrt{\|v\|_2^2 + \varepsilon}$
  - or
  - *L1*-norm, $v \leftarrow \sqrt{v / (\|v\|_1 + \varepsilon)}$



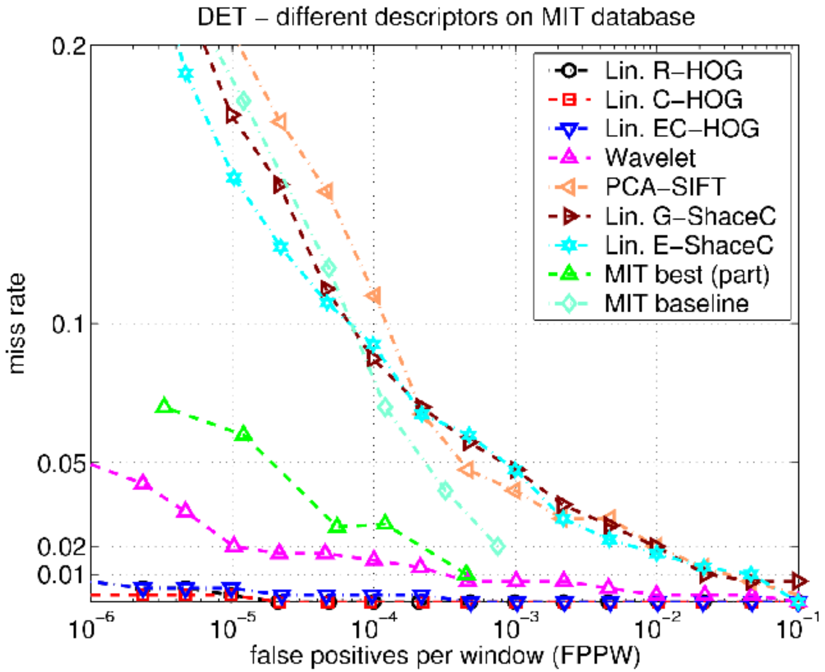


Evaluation Data Sets

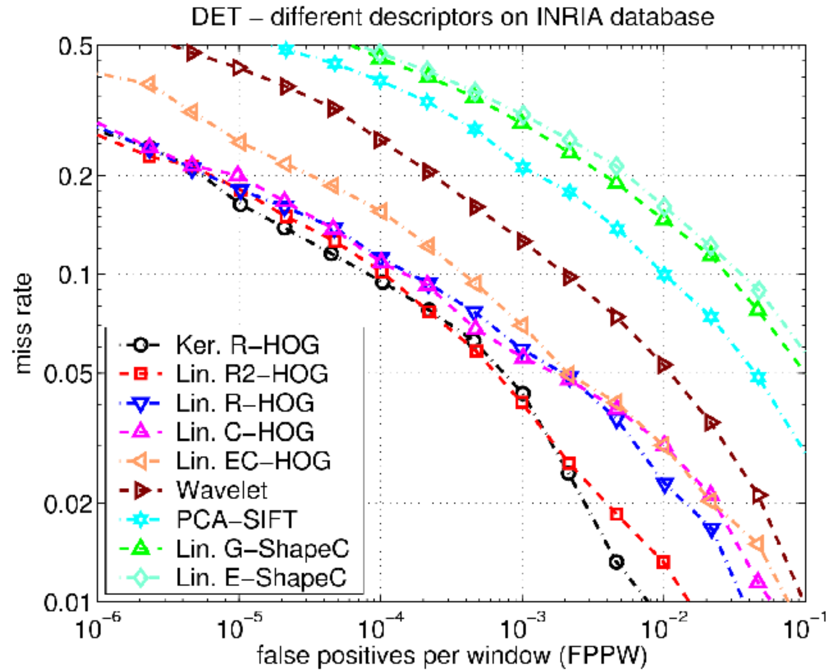
MIT pedestrian database	INRIA person database
	
Train 507 positive windows Negative data unavailable	Train 1208 positive windows 1218 negative images
Test 200 positive windows Negative data unavailable	Test 566 positive windows 453 negative images
Overall 709 annotations+ reflections	Overall 1774 annotations+ reflections

Overall Performance

MIT pedestrian database



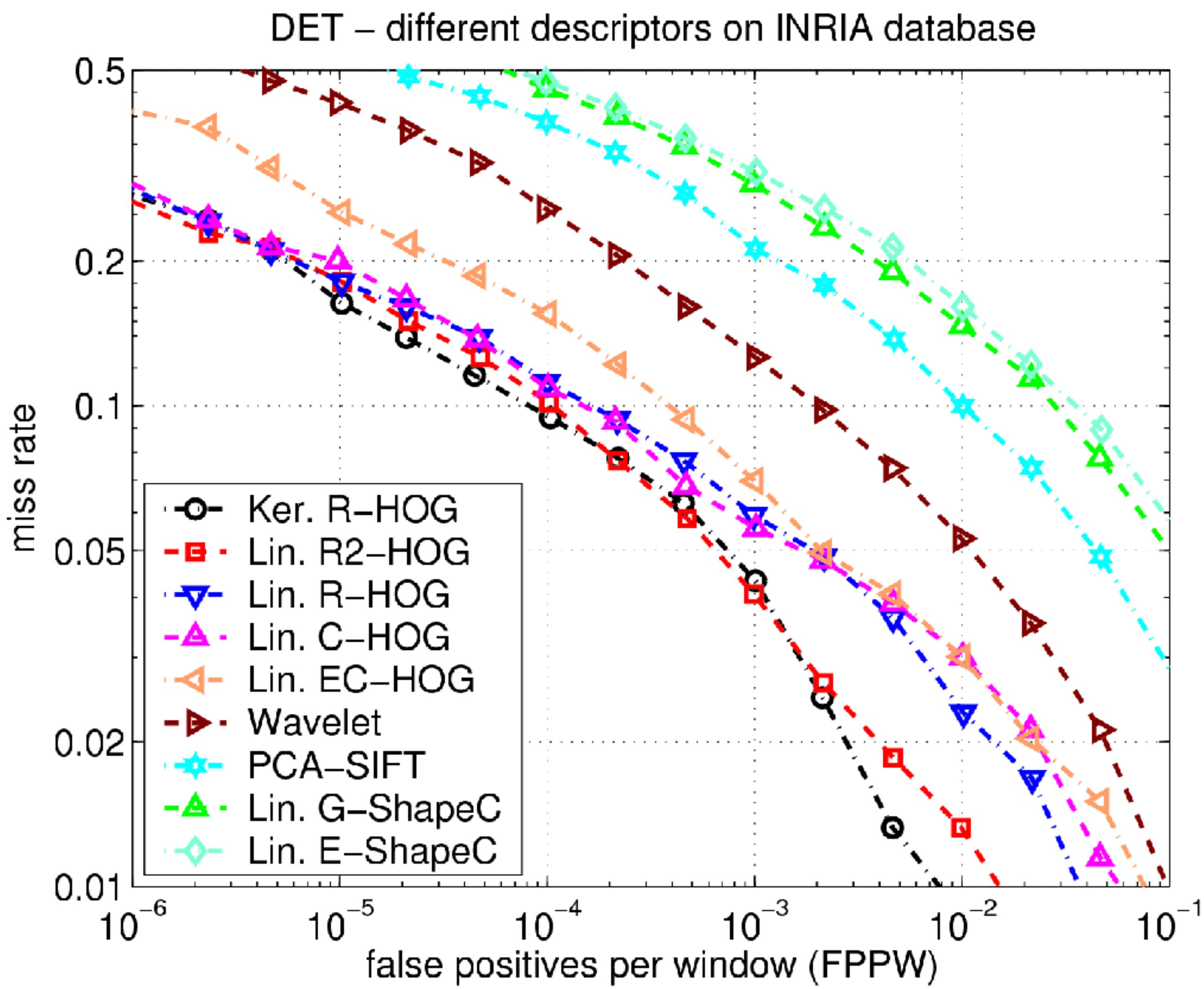
INRIA person database



R/C-HOG give near perfect separation on MIT database

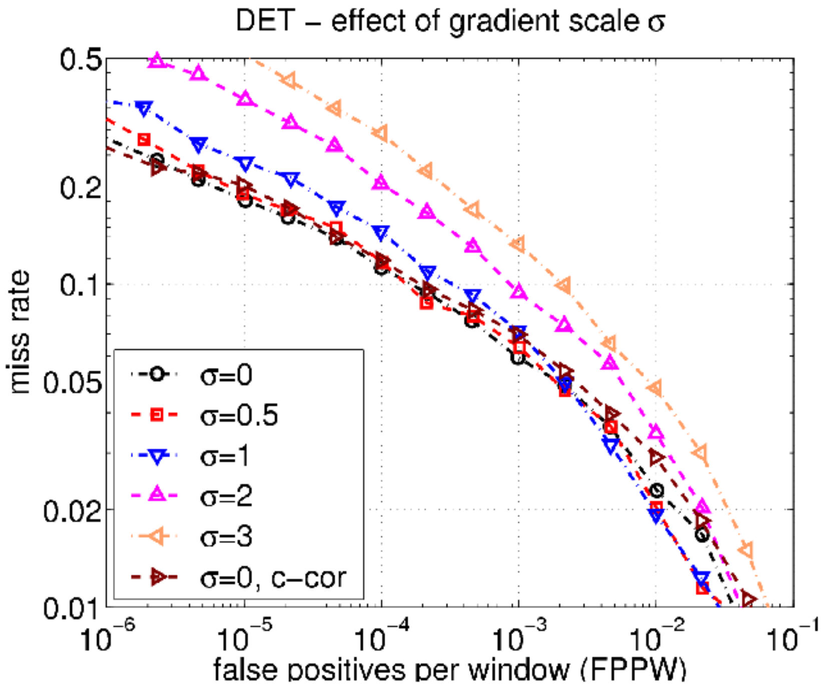
Have 1-2 order lower false positives than other descriptors

Performance on INRIA Database



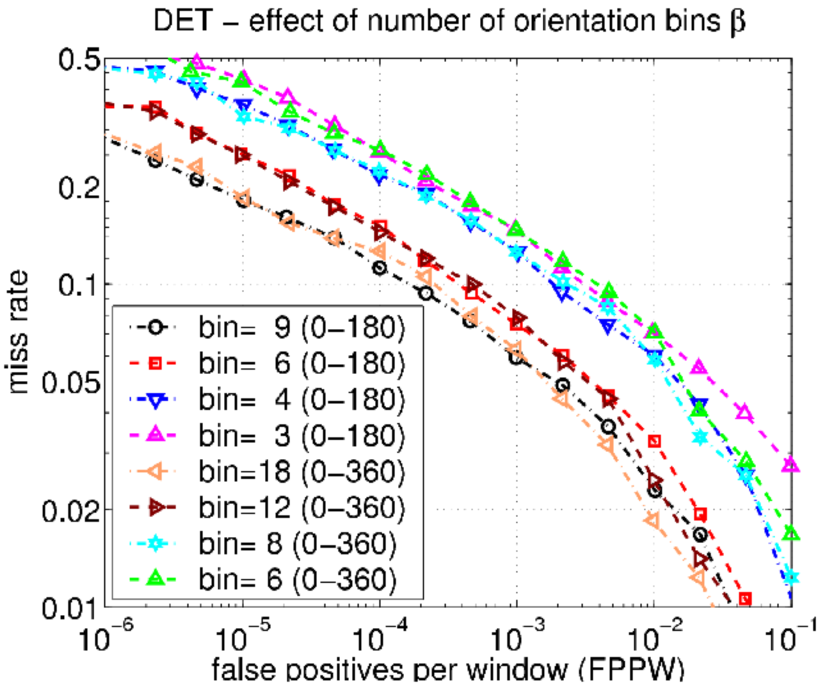
Effect of Parameters

Gradient smoothing, σ



Reducing gradient scale from 3 to 0 decreases false positives by 10 times

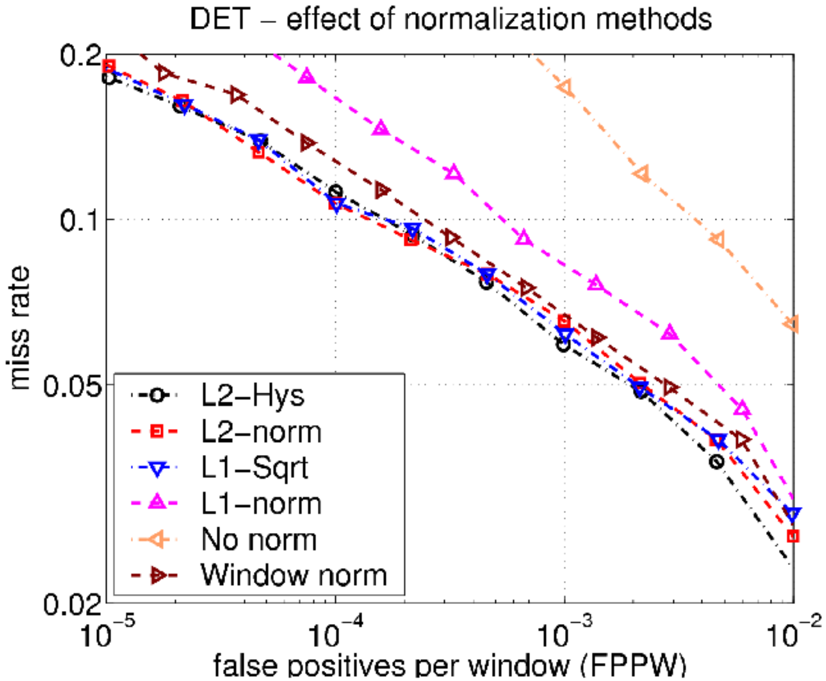
Orientation bins, β



Increasing orientation bins from 4 to 9 decreases false positives by 10 times

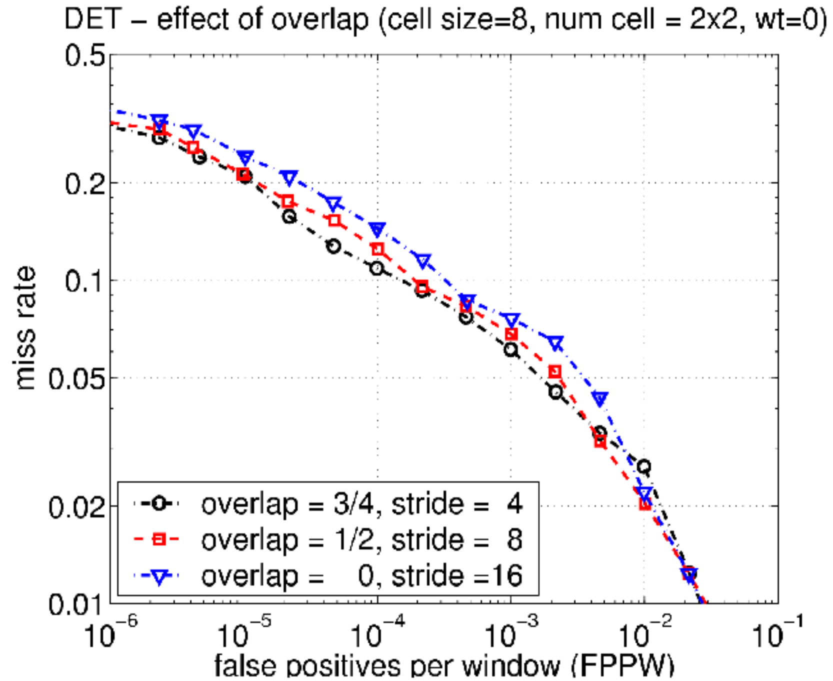
Normalisation Method & Block Overlap

Normalisation method



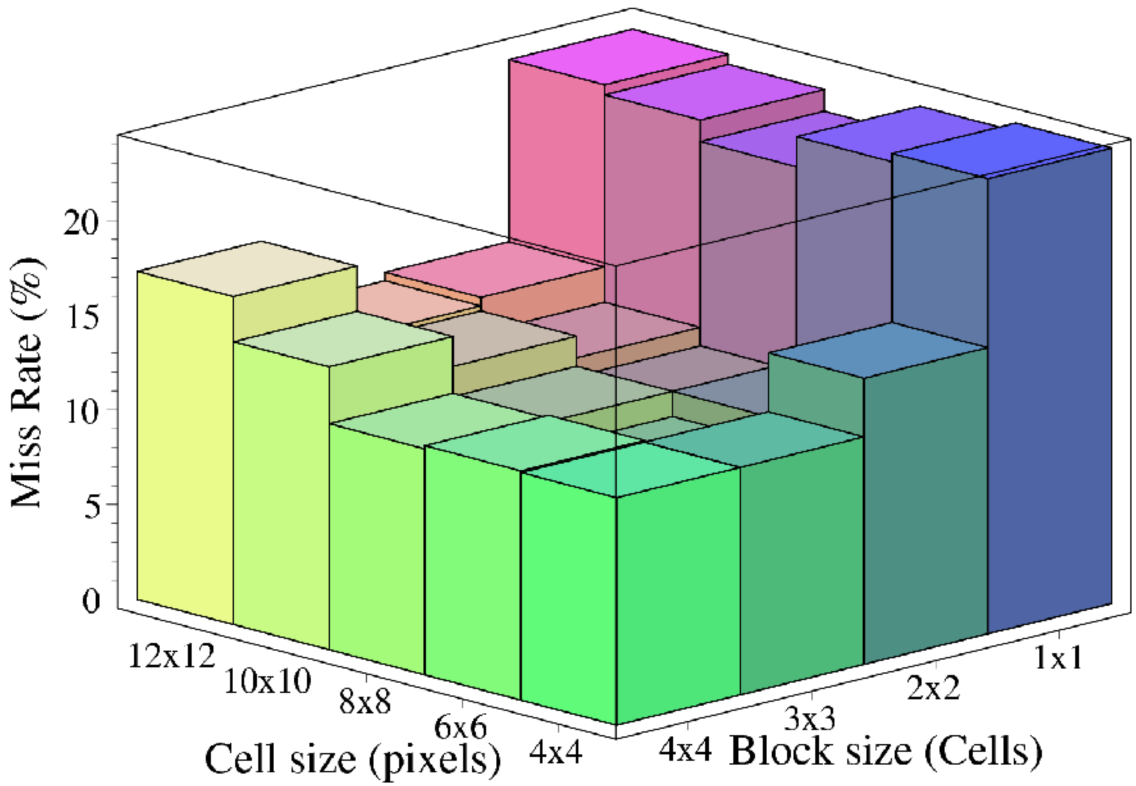
Strong local normalisation is essential

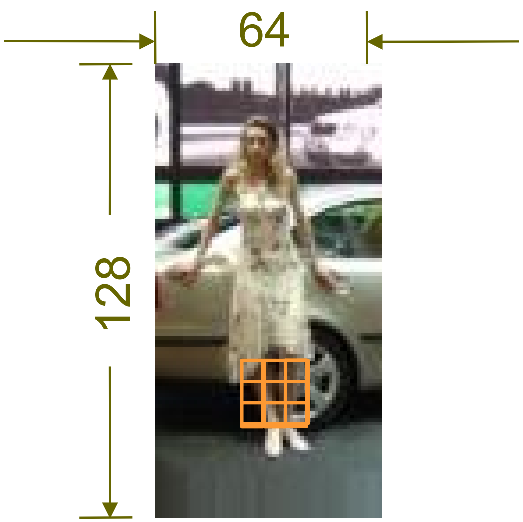
Block overlap



Overlapping blocks improve performance, but descriptor size increases

Effect of Block and Cell Size

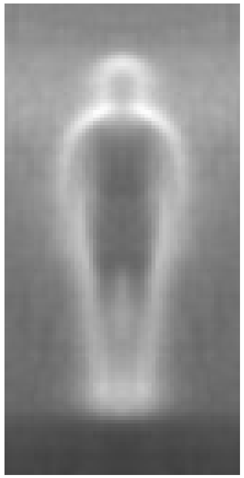




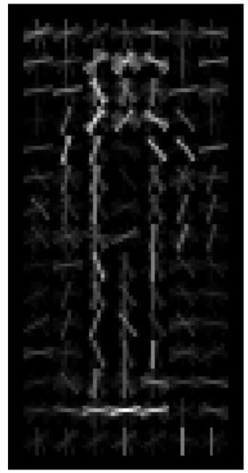
Trade off between need for local spatial invariance and need for finer spatial resolution

Descriptor Cues

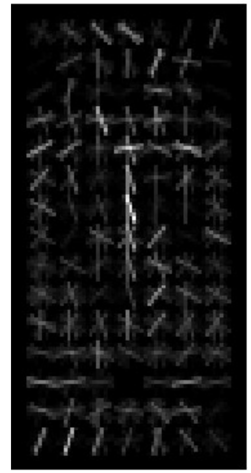
Input example



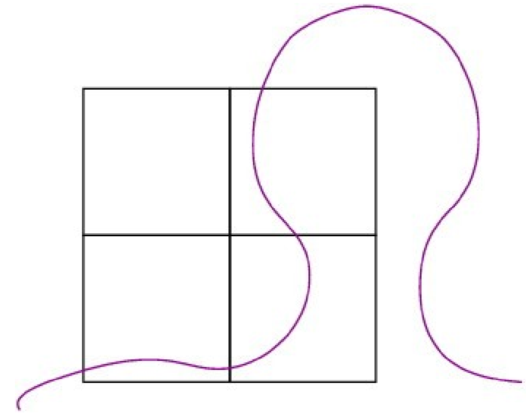
Average gradients



Weighted pos wts



Weighted neg wts



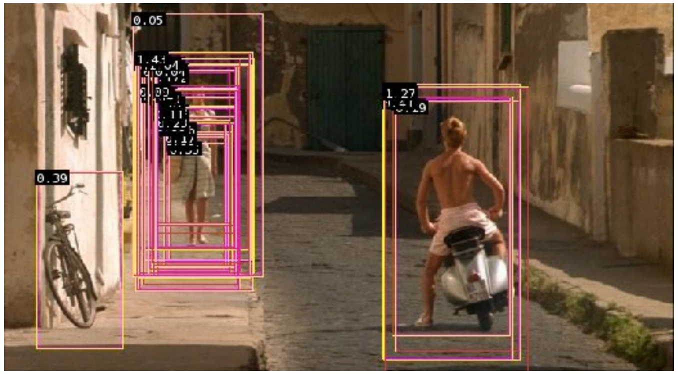
Outside-in weights

Most important cues are head, shoulder, leg silhouettes

Vertical gradients inside a person are counted as negative

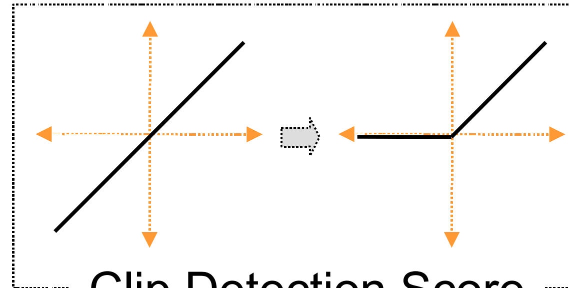
Overlapping blocks just outside the contour are most important

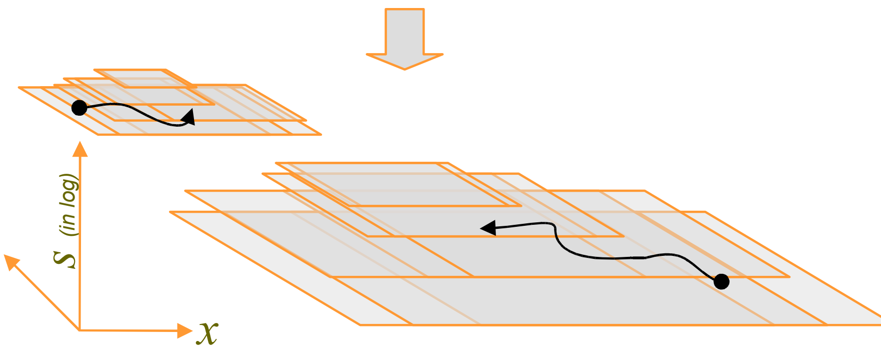
Multi-Scale Object Localisation



Multi-scale dense scan of detection window

Bias



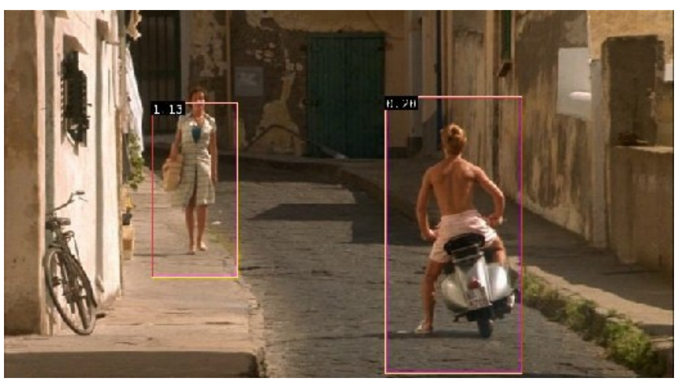


Threshold

$$H_i = [\exp(s_i)\sigma_x, \exp(s_i)\sigma_y, \sigma_s]$$

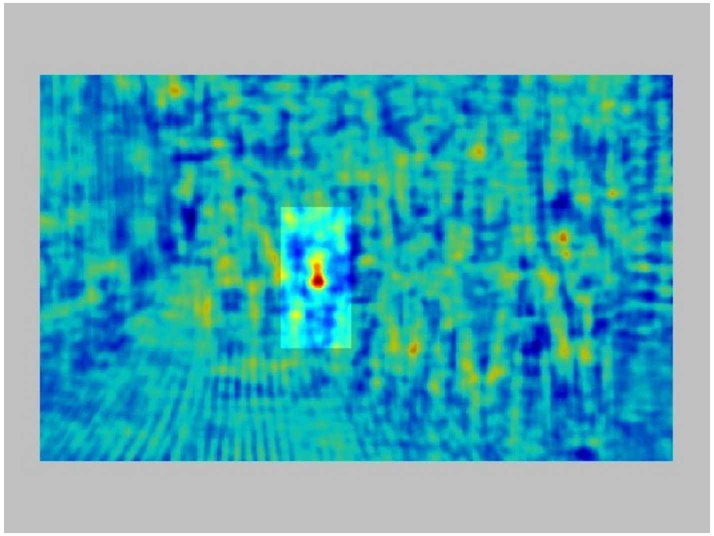
$$f(\mathbf{x}) = \sum_i^n w_i \exp\left(-\left\|(\mathbf{x}-\mathbf{x}_i)/H_i^{-1}\right\|^2/2\right)$$

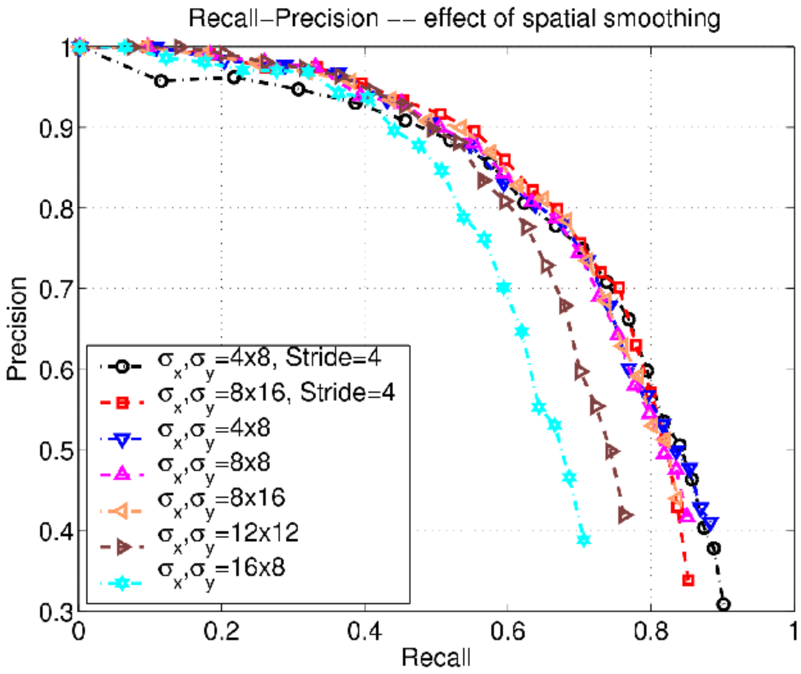
Apply robust mode detection, like mean shift



Final detections

Effect of Spatial Smoothing



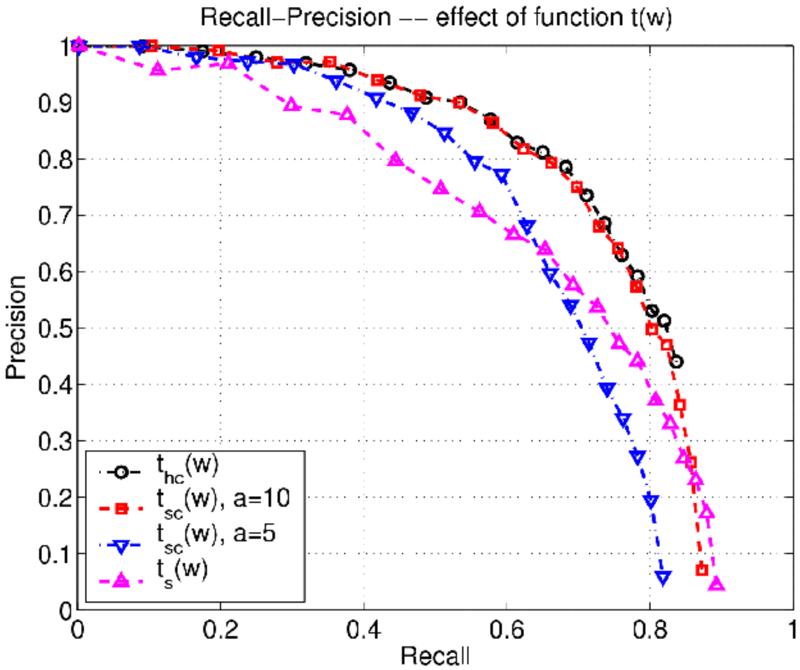


Spatial smoothing aspect ratio as per window shape, smallest sigma approx. equal to stride/cell size

Relatively independent of scale smoothing, sigma equal to 0.4 to 0.7 octaves gives good results

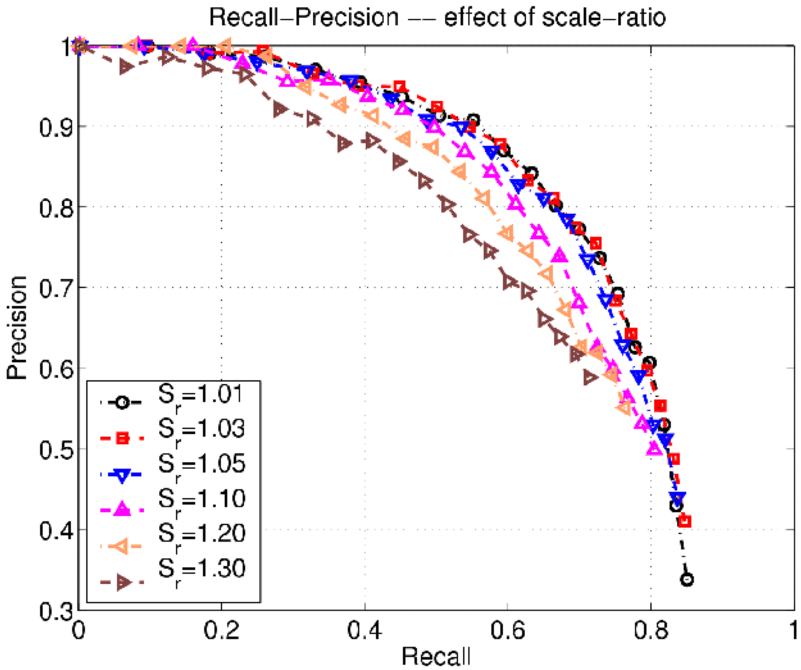
Effect of Other Parameters

Different mappings



Hard clipping of SVM scores gives the best results than simple probabilistic mapping of these scores

Effect of scale-ratio



Fine scale sampling helps improve recall

Results Using Static HOG

No temporal smoothing of detections

Conclusions for Static Case

Fine grained features improve performance

Rectify fine gradients then pool spatially

- No gradient smoothing, [1 0 -1] derivative mask
- Orientation voting into fine bins
- Spatial voting into coarser bins

Use gradient magnitude (no thresholding)

Strong local normalization

Use overlapping blocks

Robust non-maximum suppression

- Fine scale sampling, hard clipping & anisotropic kernel

Human detection rate of 90% at 10^{-4} false positives per window

Slower than integral images of Viola & Jones, 2001

Applications to Other Classes

M. Everingham et al. *The 2005 PASCAL Visual Object Classes Challenge*. Proceedings of the PASCAL Challenge

Parameter Settings

- Most HOG parameters are stable across different classes
- Parameters that change
  - Gamma compression
  - Normalisation methods
  - Signed/un-signed gradients

Results from Pascal VOC 2006

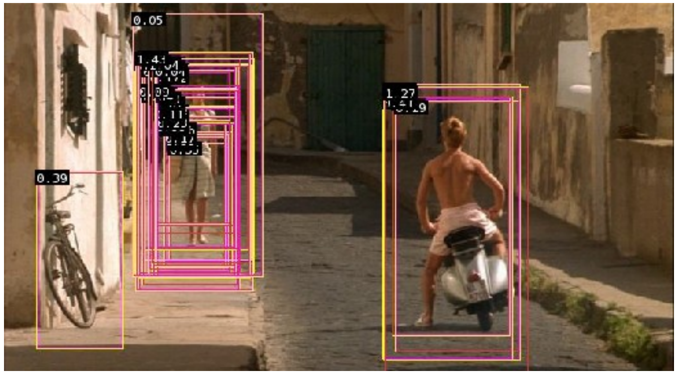
	Person	Car	Motorbike	Bicycle	Bus	Sheep	Horse	Cow	Cat	Dog
Cambridge	0.030	0.254	0.178	0.249	0.138	0.131	0.091	0.149	0.151	**0.118**
ENSMP	-	0.398	-	-	-	-	-	0.159	-	-
HOG	**0.164**	**0.444**	**0.390**	0.414	0.117	**0.251**	-	0.212	-	-
Laptev= HOG+ Ada-boost	0.114	-	0.318	**0.440**	-	-	**0.140**	0.224	-	-
TUD	0.074	-	0.153	-	-	-	-	-	-	-
TKK	0.039	0.222	0.265	0.303	**0.169**	0.227	0.137	**0.252**	**0.160**	0.113

HOG outperformed other methods for 4 out of 10 classes

Its adaBoost variant outperformed other methods for 2 out of 10 classes

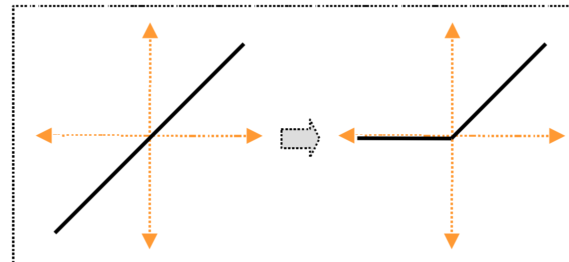
Thank You

Multi-Scale Object Localisation

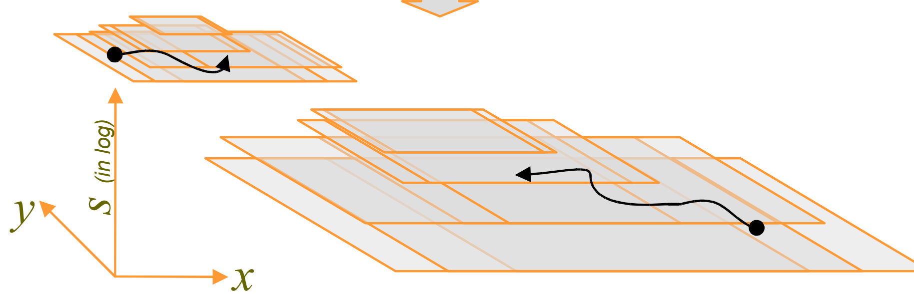


Multi-scale dense scan of detection window

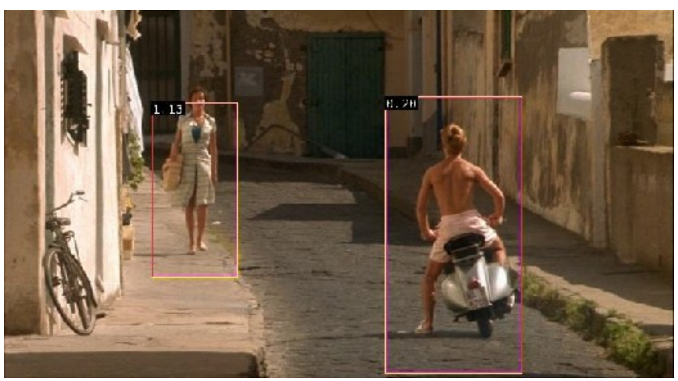
Bias



Clip Detection Score



Goal



Final detections

Threshold

$$\mathrm{H}_i = [\exp(s_i)\sigma_x, \exp(s_i)\sigma_y, \sigma_s]$$

$$f(\mathbf{x}) = \sum_i^n w_i \exp\left(-\left\| (\mathbf{x} - \mathbf{x}_i) / \mathrm{H}_i^{-1} \right\|^2 / 2 \right)$$

Apply robust mode detection, like mean shift