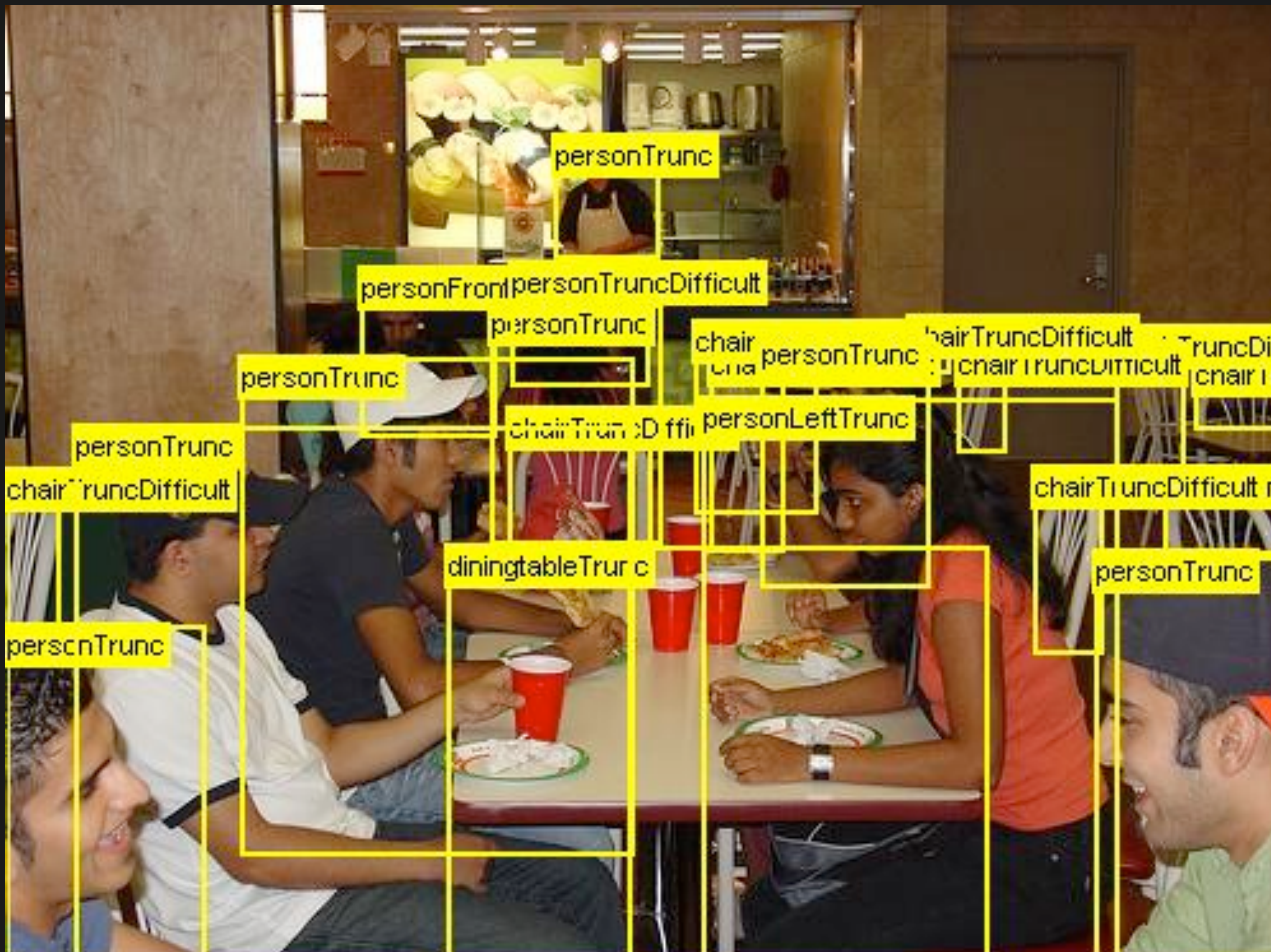
A discriminative parts-based model

Deva Ramanan
UC Irvine

Joint work with
Pedro Felzenszwalb (UChicago)
David McAllester (TTI-C)

PASCAL07 Challenge



'Difficult' objects aren't scored, but 'truncated' ones are

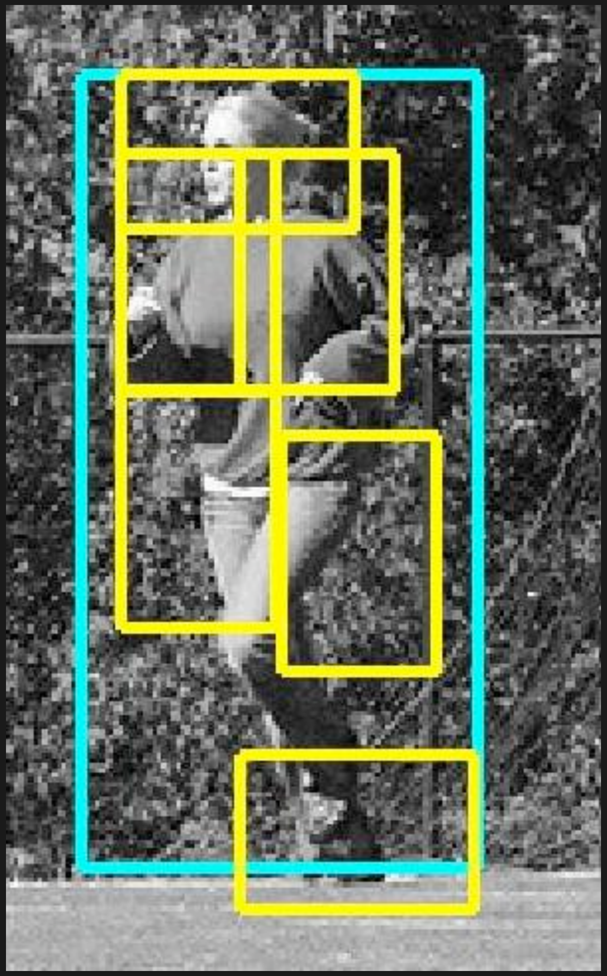
Preview of results

The rat race for medals

	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv
Our rank	3	1	2	1	1	2	2	4	1	1	1	4	2	2	1	1	2	1	4	1
Our score	.180	**.411**	.092	**.098**	**.249**	.349	.396	.110	**.155**	**.165**	**.110**	.062	.301	.337	**.267**	**.140**	.141	**.156**	.206	**.336**
Darmstadt							.301													
INRIA Normal	.092	.246	.012	.002	.068	.197	.265	.018	.097	.039	.017	.016	.225	.153	.121	.093	.002	.102	.157	.242
INRIA Plus	.136	.287	.041	.025	.077	.279	.294	.132	.106	.127	.067	.071	**.335**	.249	.092	.072	.011	.092	.242	.275
IRISA		.281					.318	.026	.097	.119			.289	.227	.221		**.175**			.253
MPI Center	.060	.110	.028	.031	.000	.164	.172	.208	.002	.044	.049	.141	.198	.170	.091	.004	.091	.034	.237	.051
MPI ESSOL	.152	.157	**.098**	.016	.001	.186	.120	**.240**	.007	.061	.098	**.162**	.034	.208	.117	.002	.046	.147	.110	.054
Oxford	**.262**	.409				**.393**	**.432**							**.375**					**.334**	
TKK	.186	.078	.043	.072	.002	.116	.184	.050	.028	.100	.086	.126	.186	.135	.061	.019	.036	.058	.067	.090

- Out of 20 classes, we currently get 10 golds & 6 silvers
- New Oxford/MSR results very impressive, but we still win on some categories (person)
- Fast matlab code (2 sec/image) available online

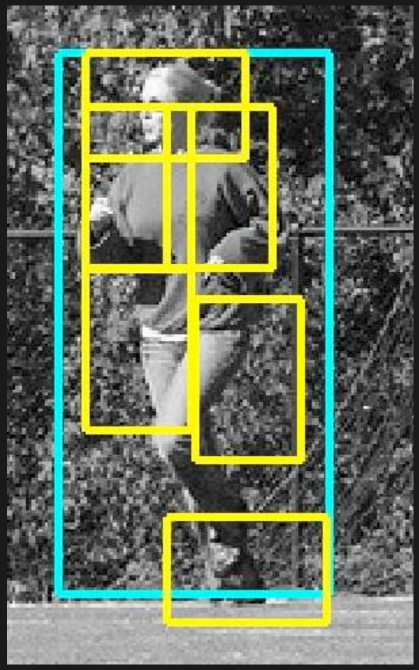
Model overview



-Model consists of root filter plus deformable parts

-Training data consists of bounding boxes (part structure learned automatically)

Rich related work



Fischler & Elschlager 73, Burl et al 98, Ioffe & Forsyth 01, Mohan et al 01, Belongie et al 02, Fergus et al 03, Felzenszwalb & Huttenlocher 05, Crandall et al 05, Berg et al 05, Liebe et al 05, Sudderth et al 05, Amit & Trouve 07....

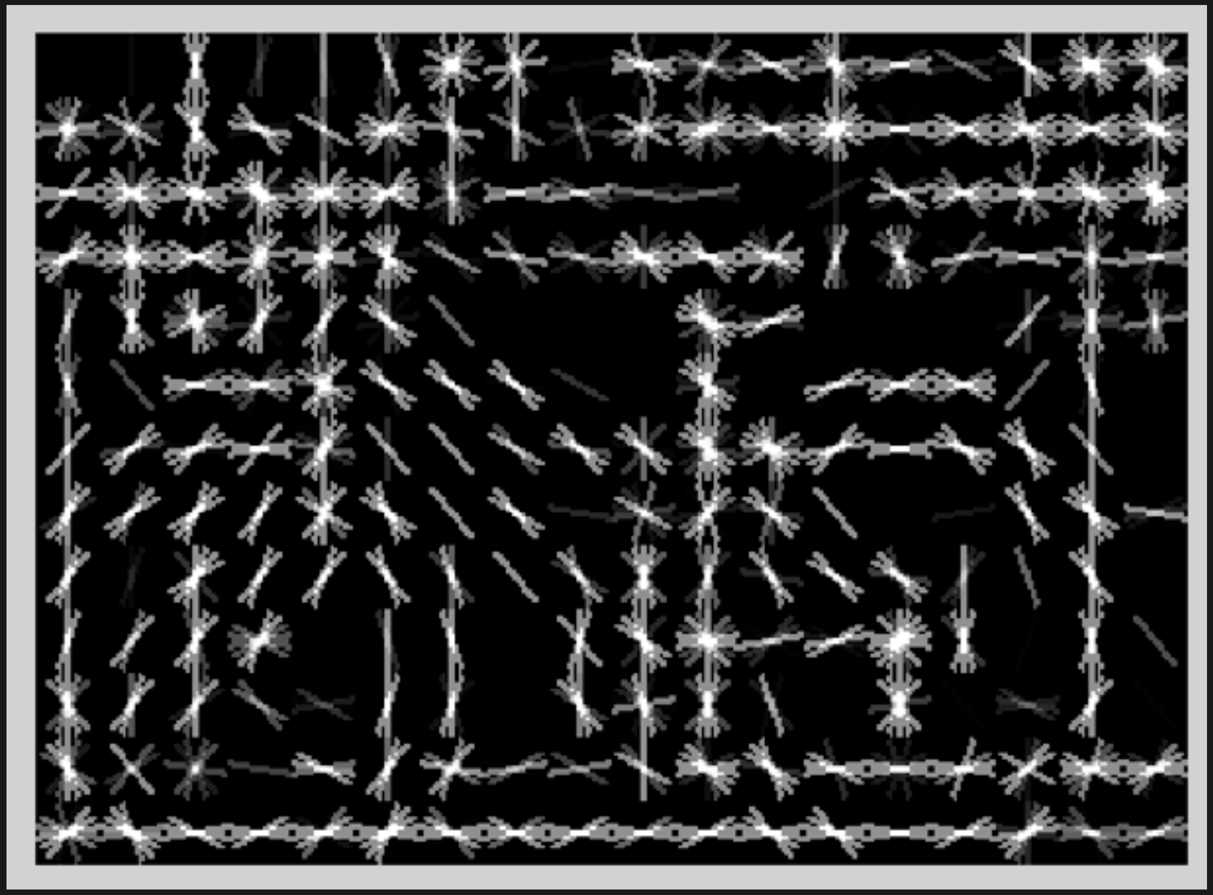
Our flavor:

Dense window scanning (no feature detection)

Multiscale histogram-of-gradient features

Discriminative (SVM) training with weakly-labeled data

Image features - histograms of gradients

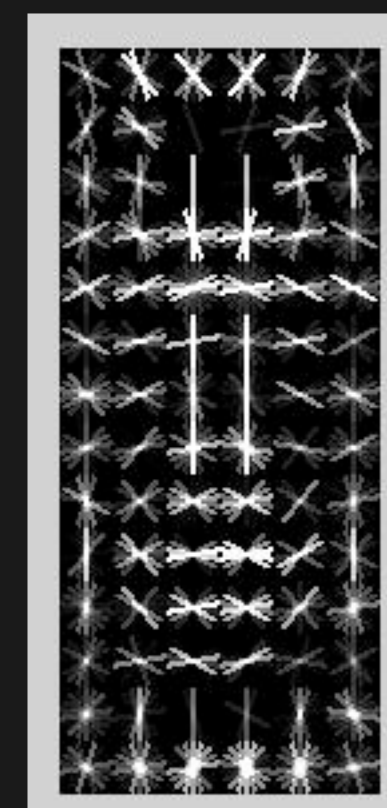


- Our implementation of DalalTriggs HOG features

Learned model

$$f_w(x) = w \cdot \Phi(x)$$

positive weights



negative weights

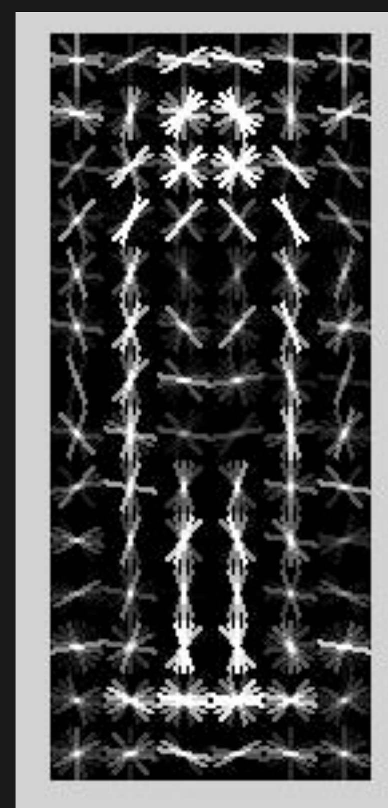
What do negative weights mean?

$$wx > 0$$

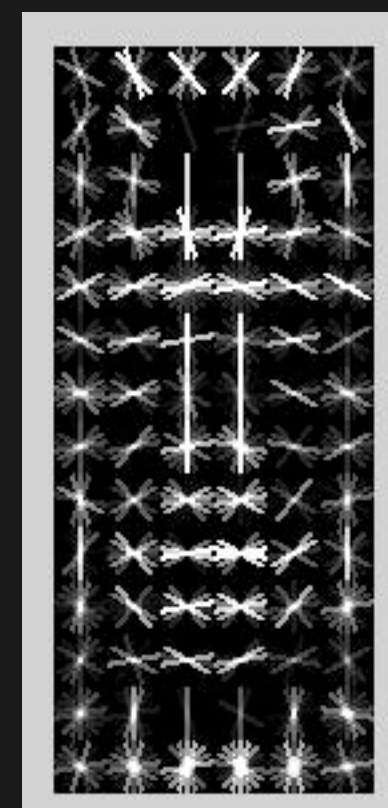
$$(w_+ - w_-)x > 0$$

$$w_+ > w_-x$$

pedestrian
model



>



pedestrian
background
model

Complete system should compete pedestrian/pillar/doorway models

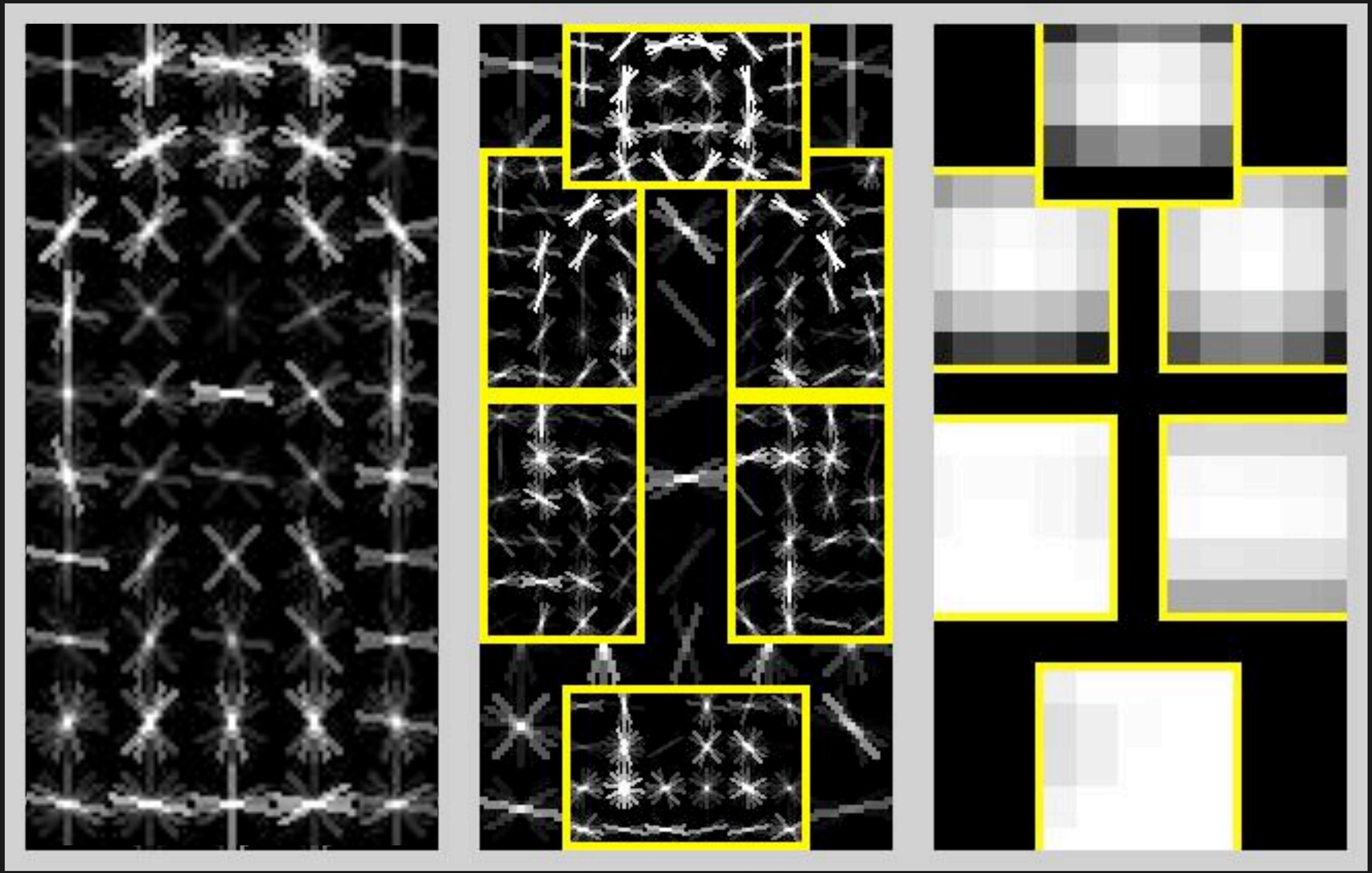
Discriminative models come equipped with own bg

(avoid firing on doorways by penalizing vertical edges)

Multi-scale star model

root filter
8x8
resolution

Multi-scale star model

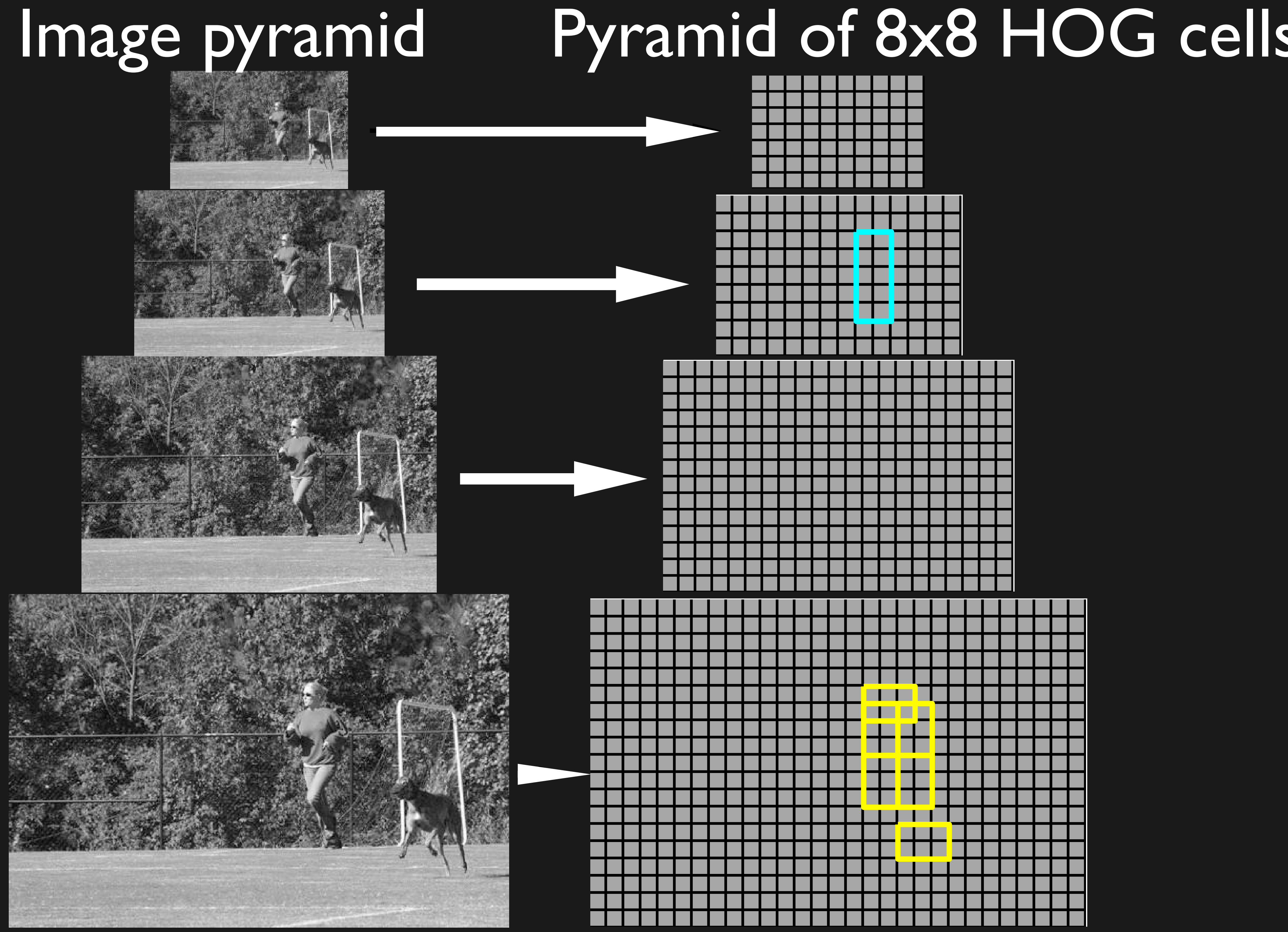


root filter
8x8
resolution

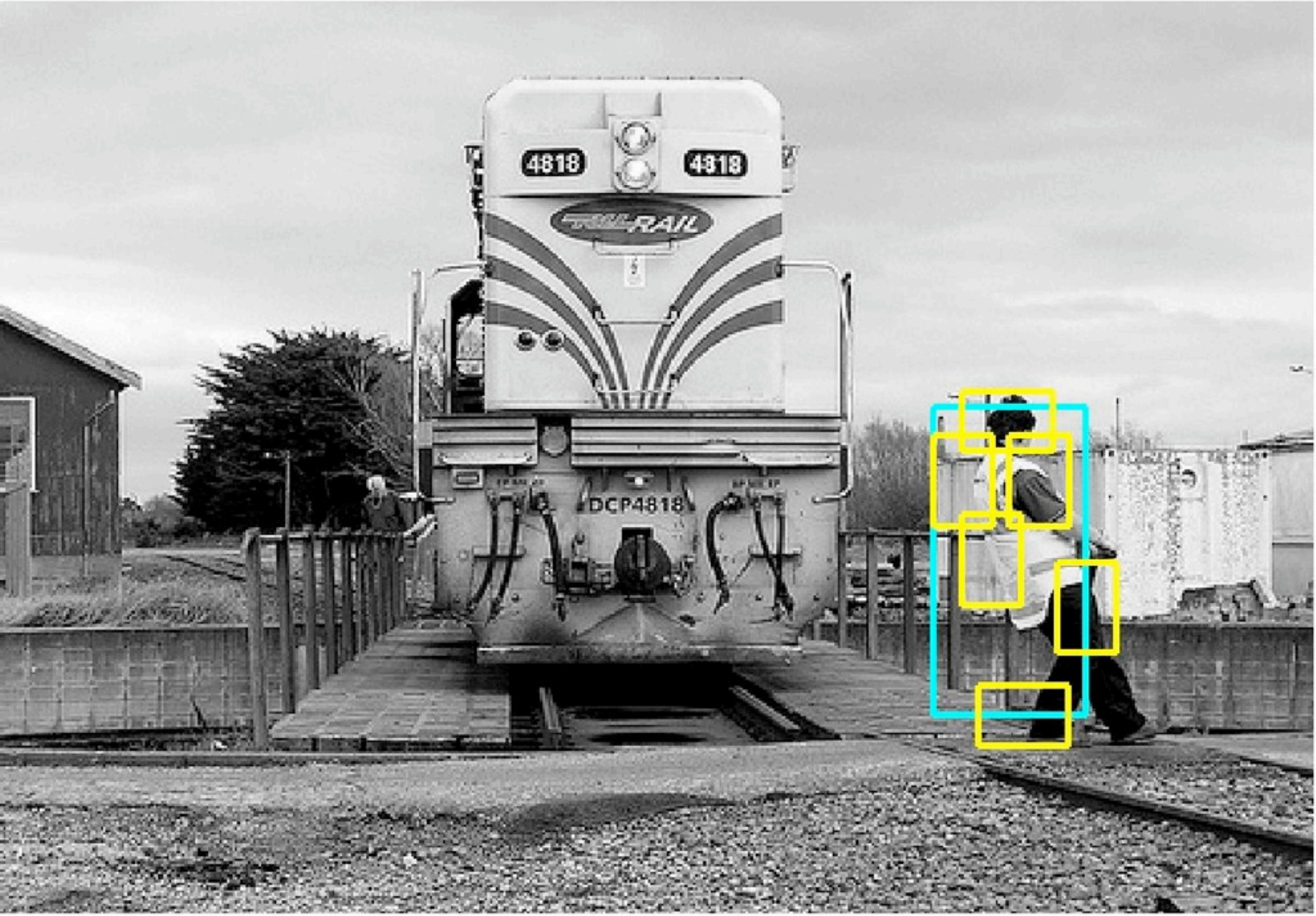
part filters
4x4
resolution

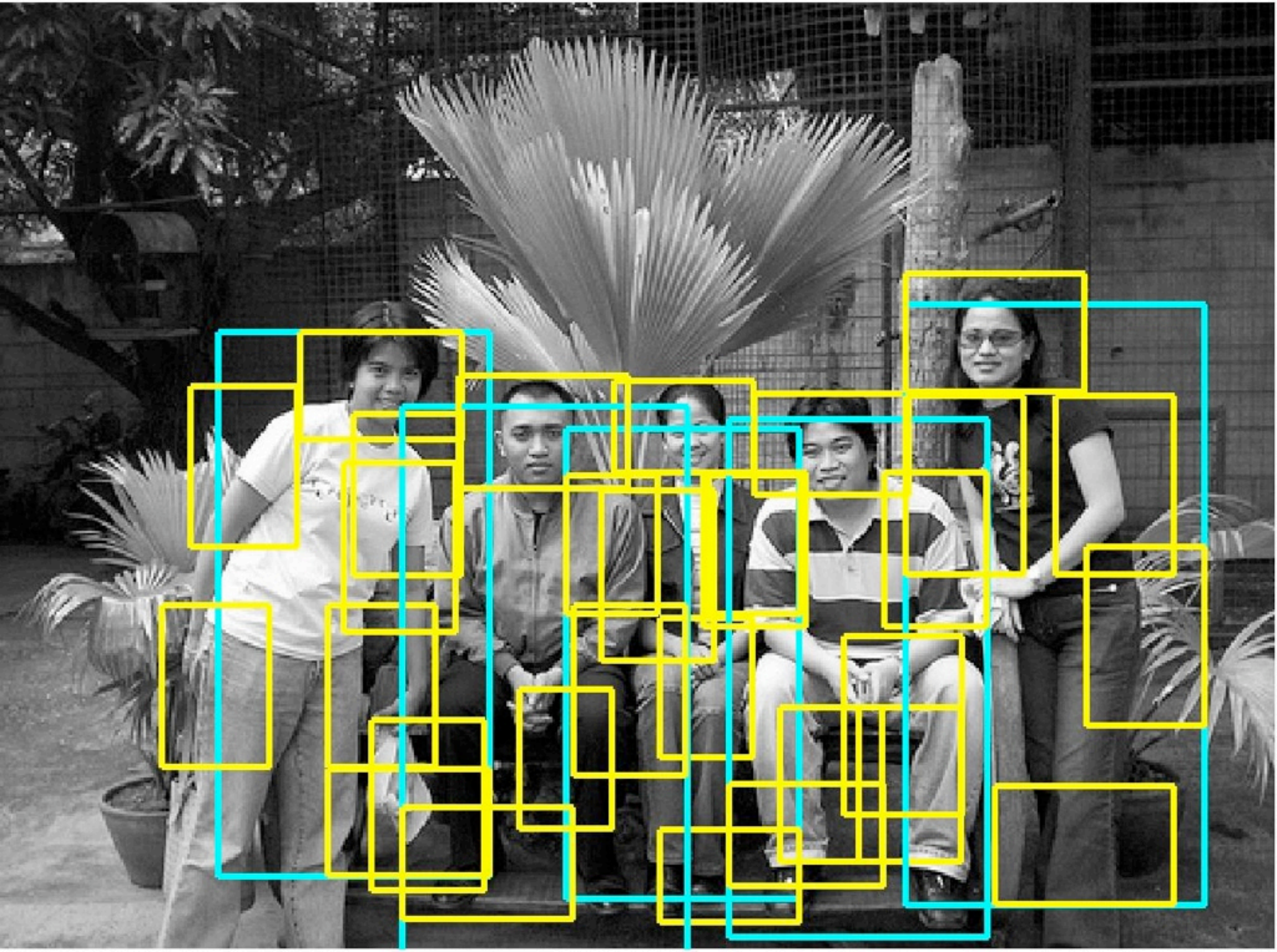
bounded
quadratic
spatial model

‘Cleaner’ multiscale



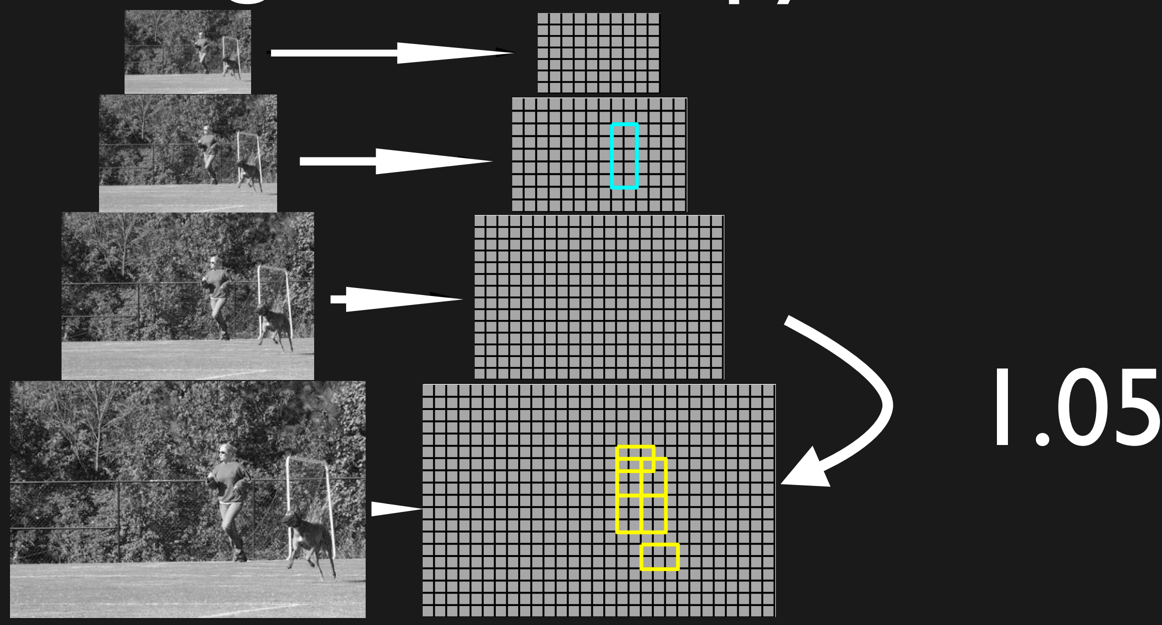
Part filters are not 4x4, but 8x8 at a finer image resolution


4818
4818
RAIL
DCP4818



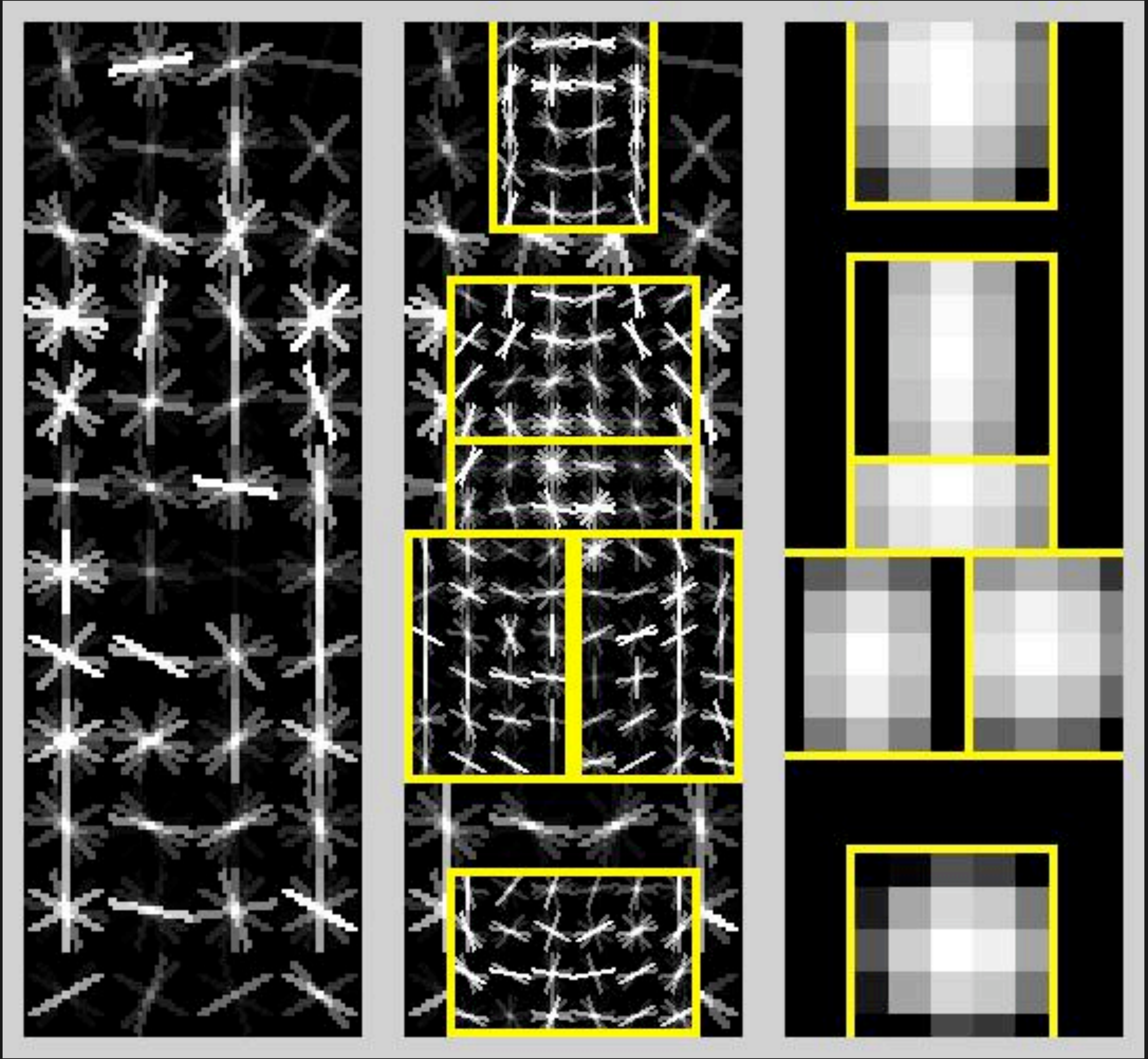
Some stats

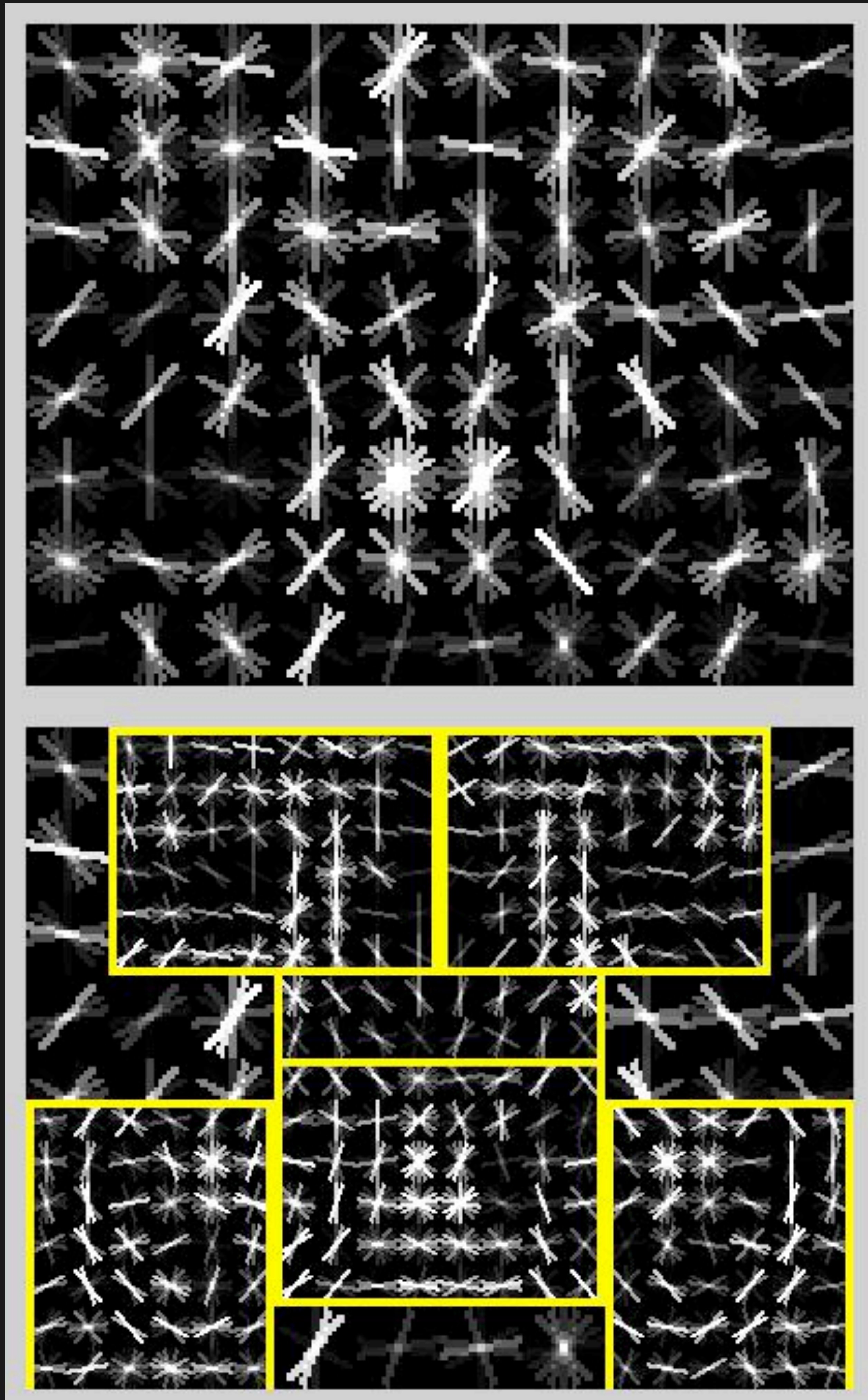
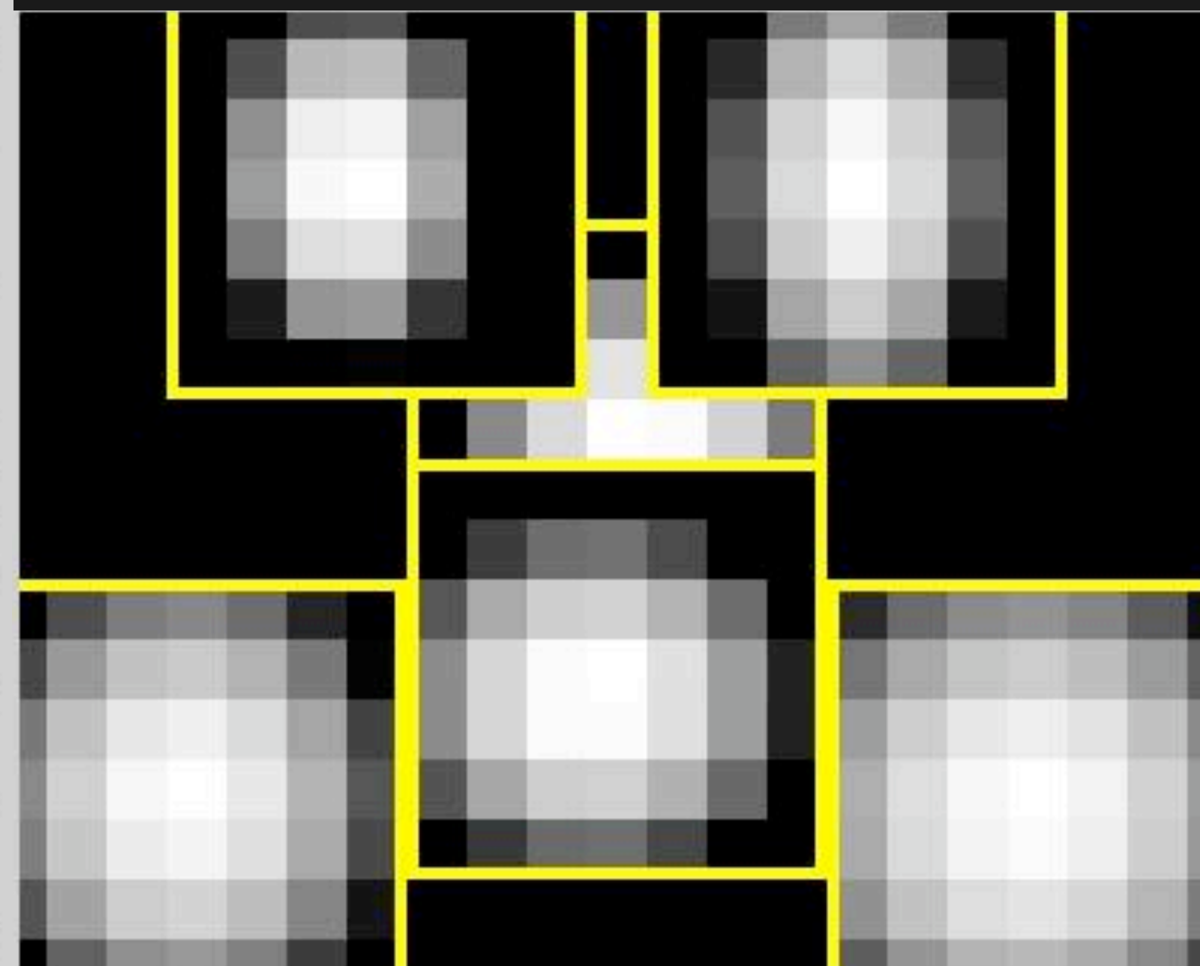
- We use 1.05 scaling between pyramid levels

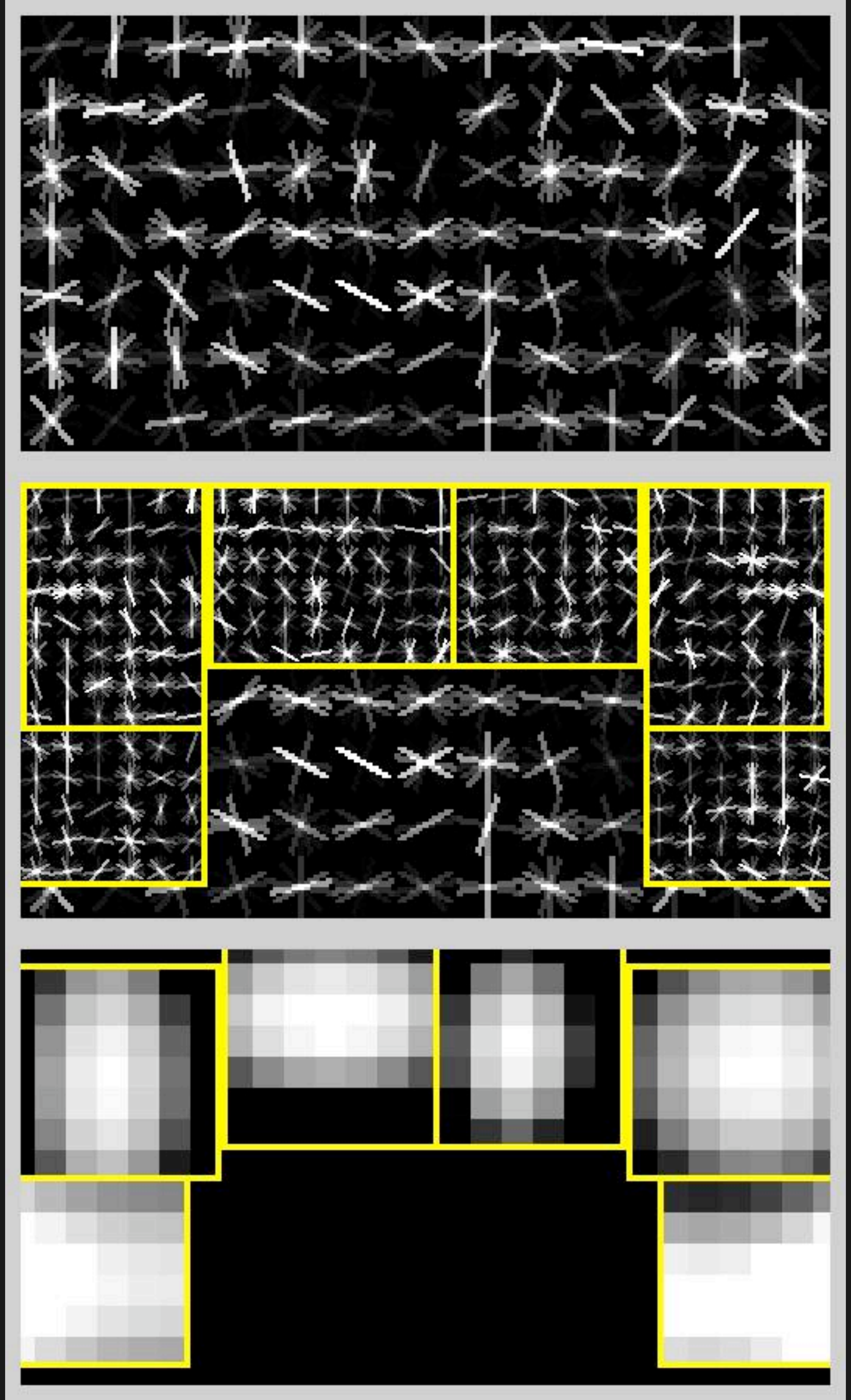


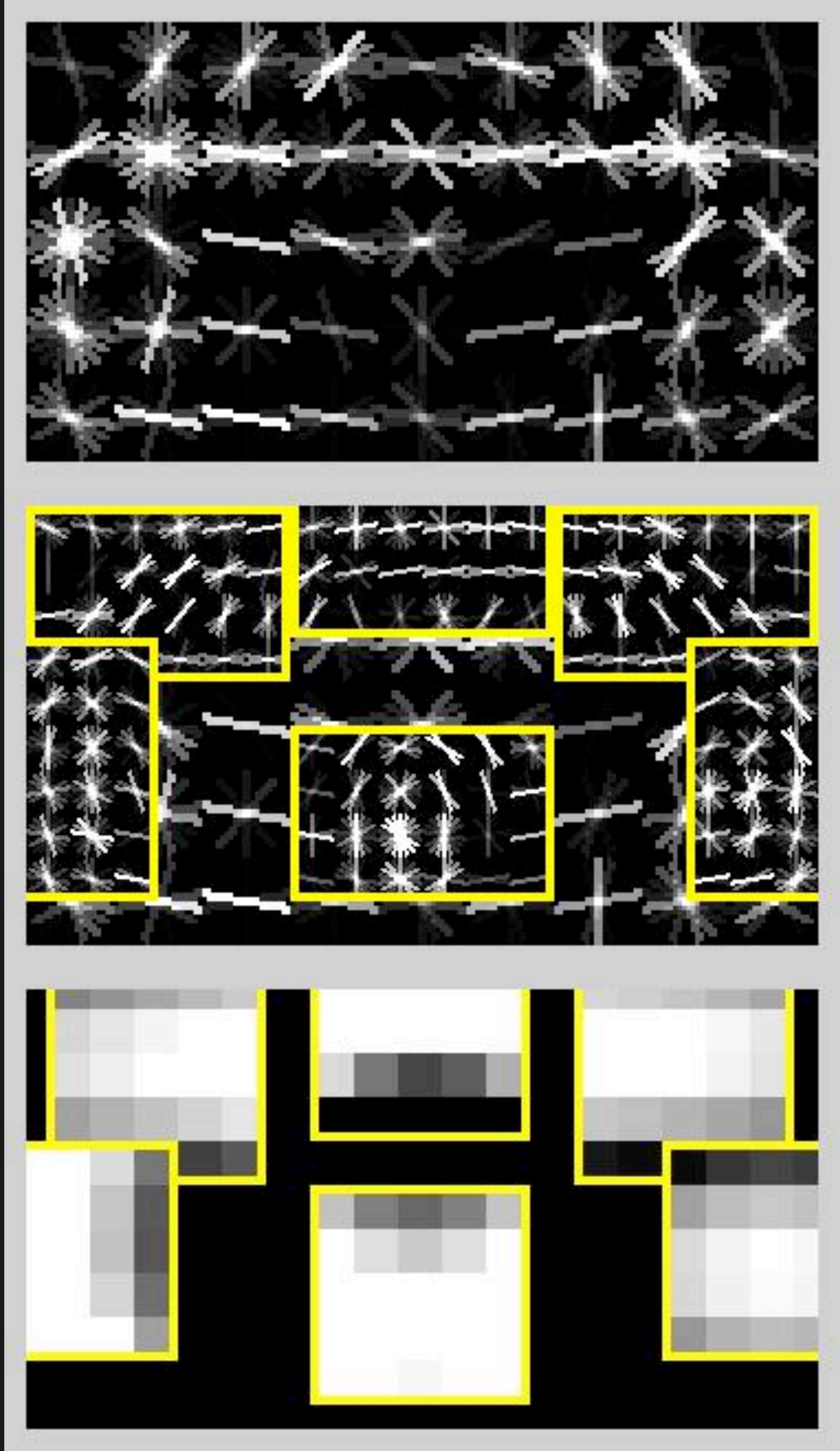
- Training time: 3-4 hours per class using 1 cpu, including learning part models automatically

- Testing time: 2 seconds per image per model



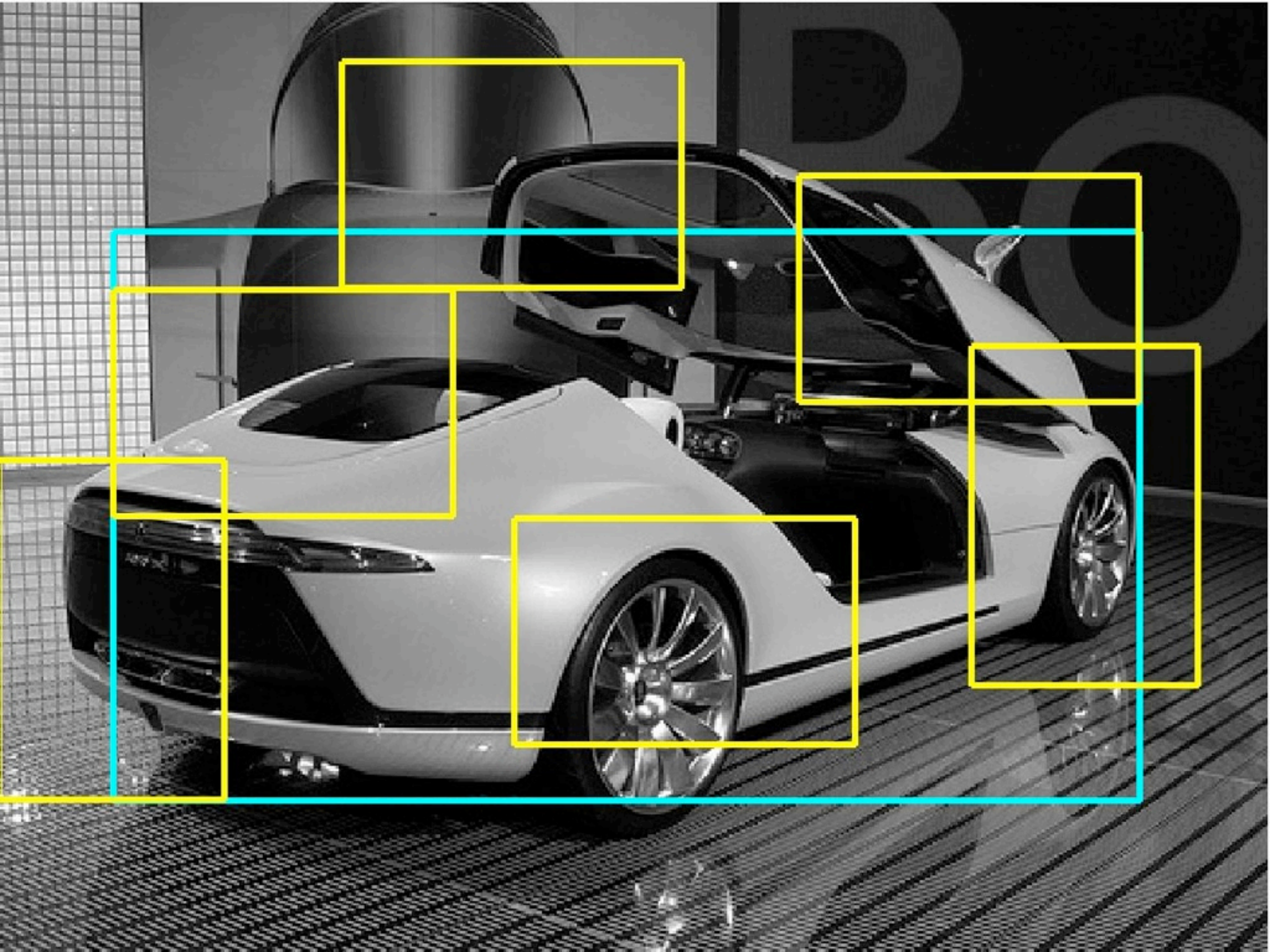







3 ‘wheels’?
We need 3D representations

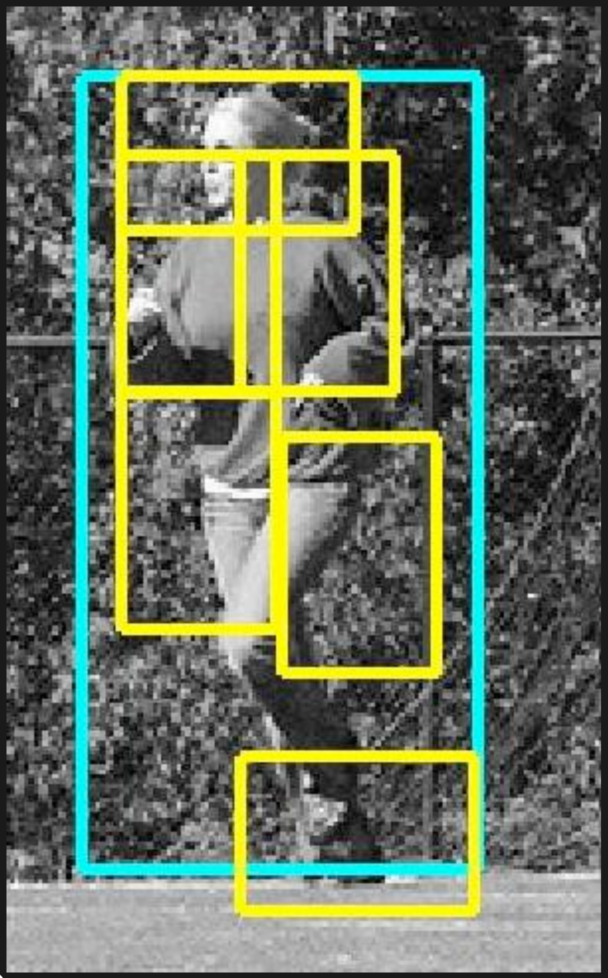
non-gaussian shape models




SUPPLY CO.

Formal model

$$f_w(x) = w \cdot \Phi(x)$$



$$f_w(x) = \max_z w \cdot \Phi(x, z)$$

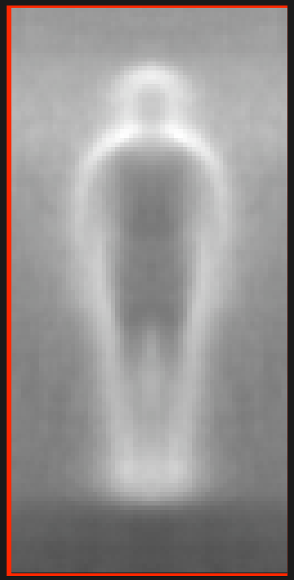
z = vector of part offsets

w = concatenation of filters & deformation parameters

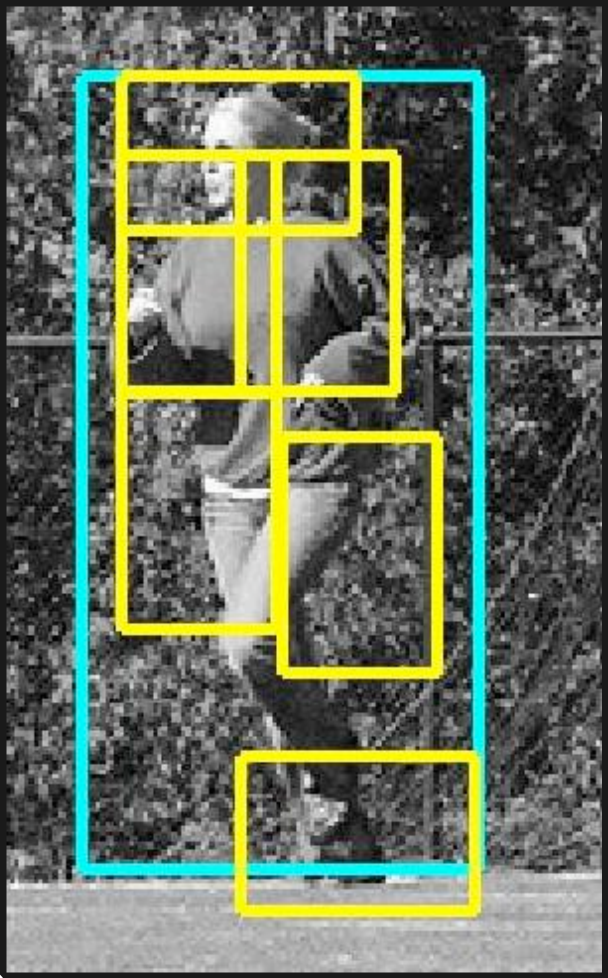
$\Phi(x, z)$= concatenation of HOG features & part offsets

Linear vs convex models

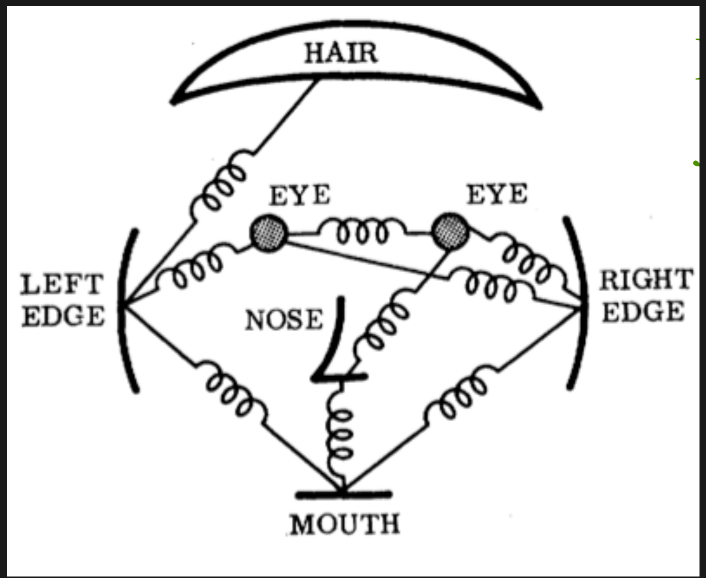
$$f_w(x) = w \cdot \Phi(x)$$



vs



$$f_w(x) = \max_z w \cdot \Phi(x, z)$$



Latent SVMs

$$f_w(x) = \max_z w \cdot \Phi(x, z)$$

Assume we are given positive and negative training windows $\{x_i\}$

$$w^* = \arg\min_w \lambda ||w||^2 + \ \ldots$$

$$\sum_{i \in pos} \max(0, 1 - f_w(x_i)) + \sum_{i \in neg} \max(0, 1 + f_w(x_i))$$

If f() is linear in w, this is a standard SVM (convex)

If f() is arbitrary, this (in general) is not convex

If f() is convex in w, the training objective is 'semi-convex'

(Instance of LeCun's Energy Based Model)

Latent SVMs

$$f_w(x) = \max_z w \cdot \Phi(x, z)$$

Assume we are given positive and negative training windows $\{x_i\}$

$$\hat{w} = \arg\min_w \lambda ||w||^2 + \ \ldots$$

$$\sum_{i \in pos} \max(0, 1 - w \cdot \phi(x_i, z_i)) + \sum_{i \in neg} \max(0, 1 + f_w(x_i))$$

Optimization is convex if we fix the z_i for positive x_i

(ie, if we know part locations on positives)

Train with coordinate descent

1) Given w, for each positive x_i find z_i that maximizes

$$w \cdot \Phi(x_i, z_i)$$

(optimize location of parts on positives)

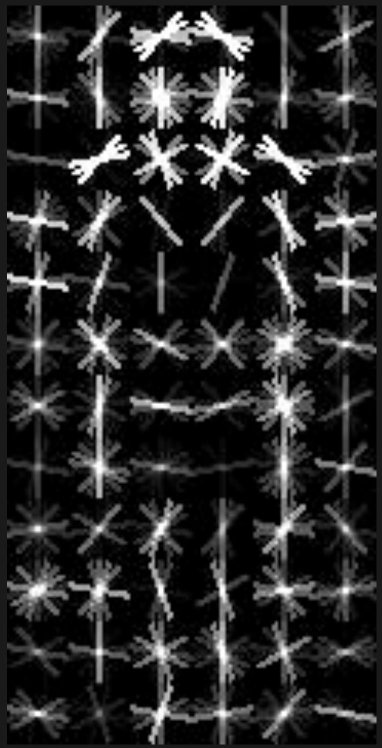
2) Given positive z_i, find w that optimizes convex objective

It can be show that this reduces the overall (nonconvex) objective on each iteration so we converge to a local minimum.

Root filter initialization

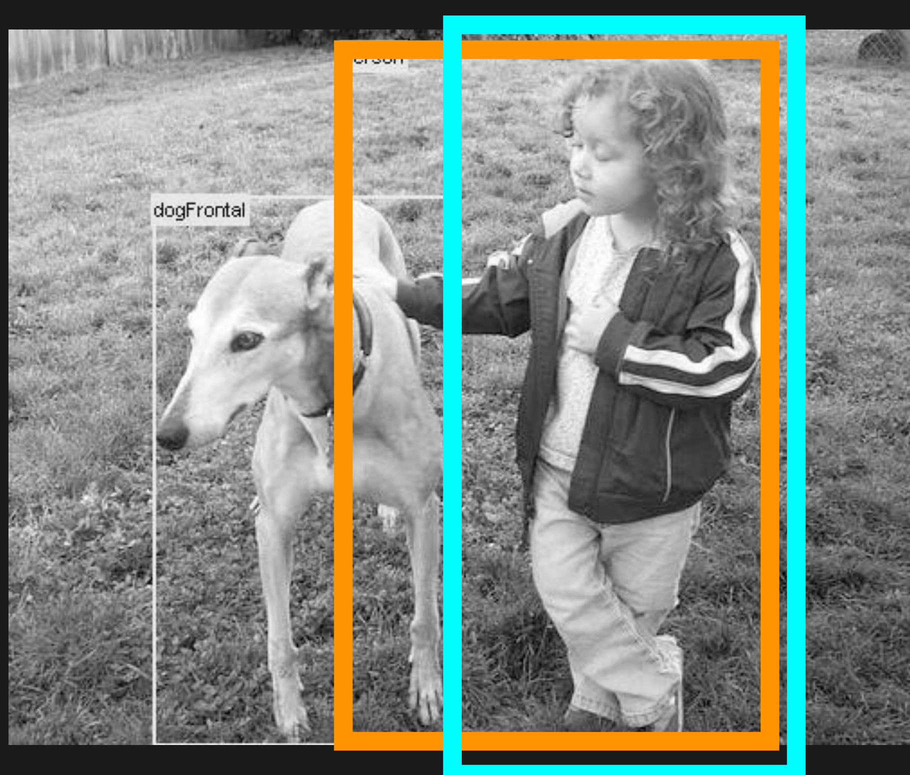
- We select the aspect and size by a heuristic tuned on 2006 data (use most common aspect and smallest area > 80% of training bounding boxes)

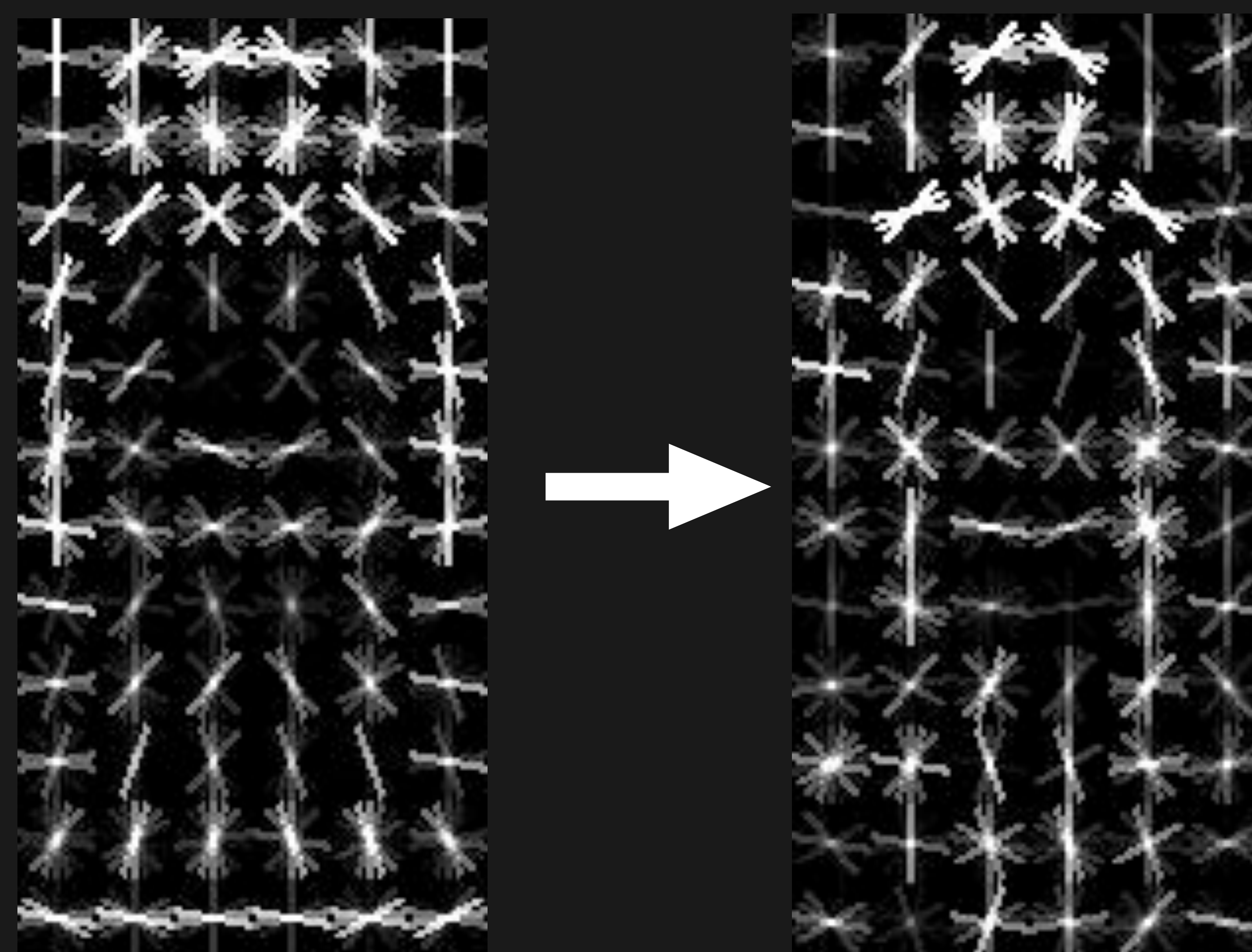
- Train a root filter with SVM-light: use non-truncated positives (warped to fixed aspect & size) and random negatives



Root filter refinement

- For each positive training example, estimate a latent box that overlaps original box > 50%
- Automatically adjust bounding boxes with a LSVM



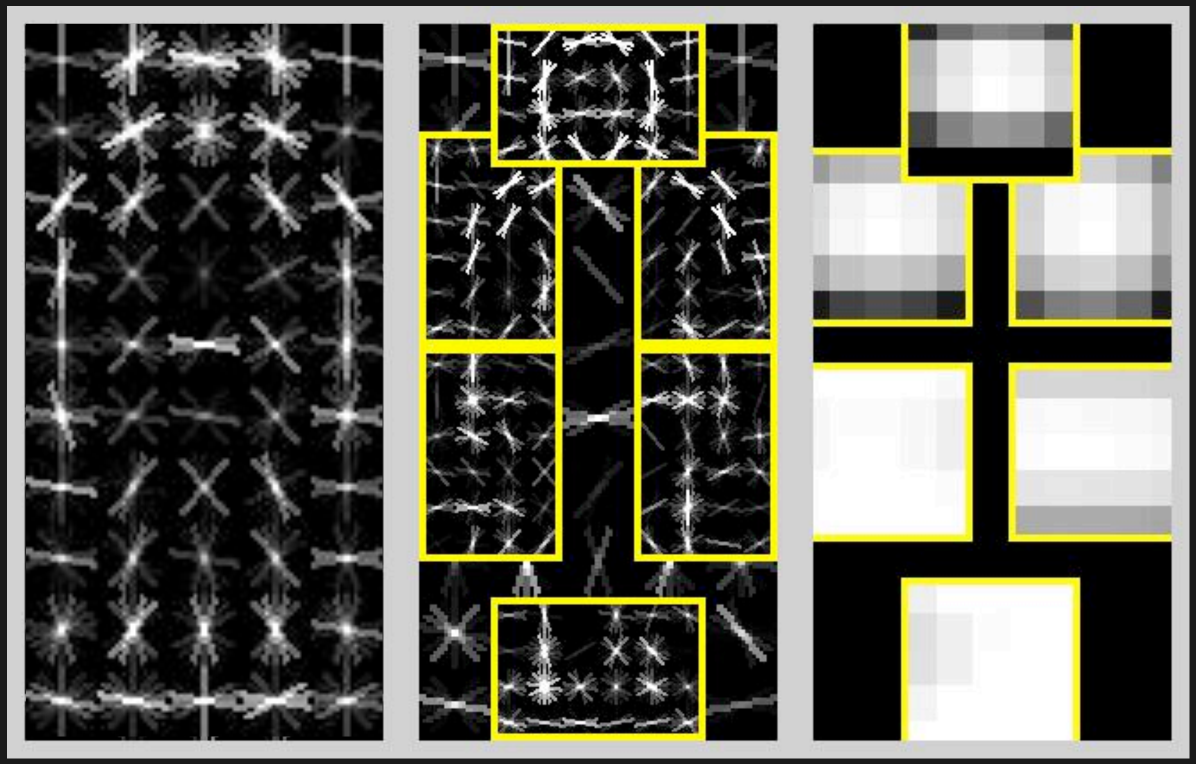


‘Tightens’ head weights

Part filter initialization

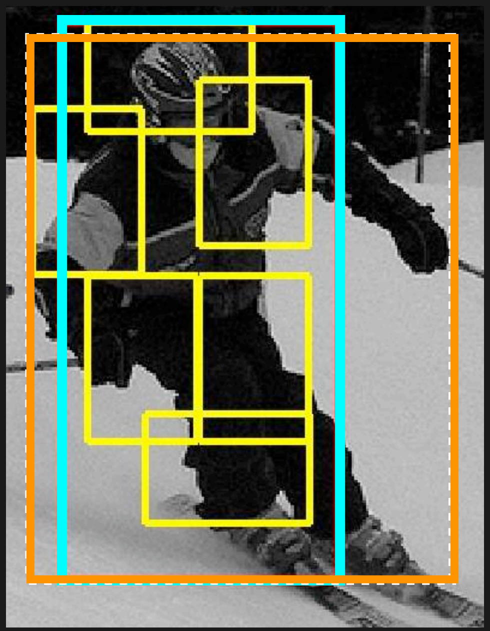
- Look for regions in root filter with lots of energy - part filter initialized to subwindow doubled in resolution

- Spatial model allows for a bounded offset from original anchor point - quadratic deformation cost initialized to weak gaussian

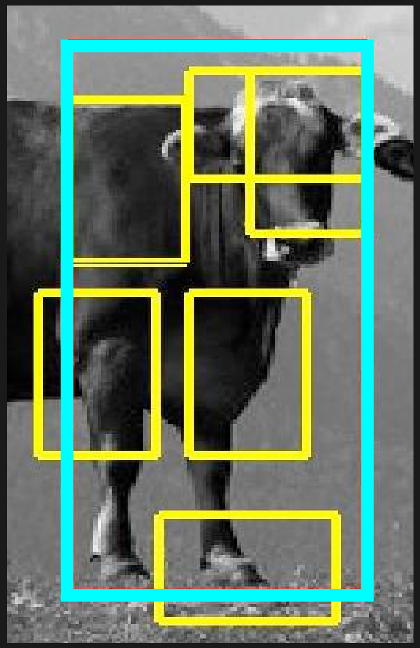


Model update

•Update each positive with best-scoring $\Phi(x_i, z_i)$ with >50% overlap of original box



•Collect negative $\Phi(x_i, z_i)$`s by finding margin violations on negative images



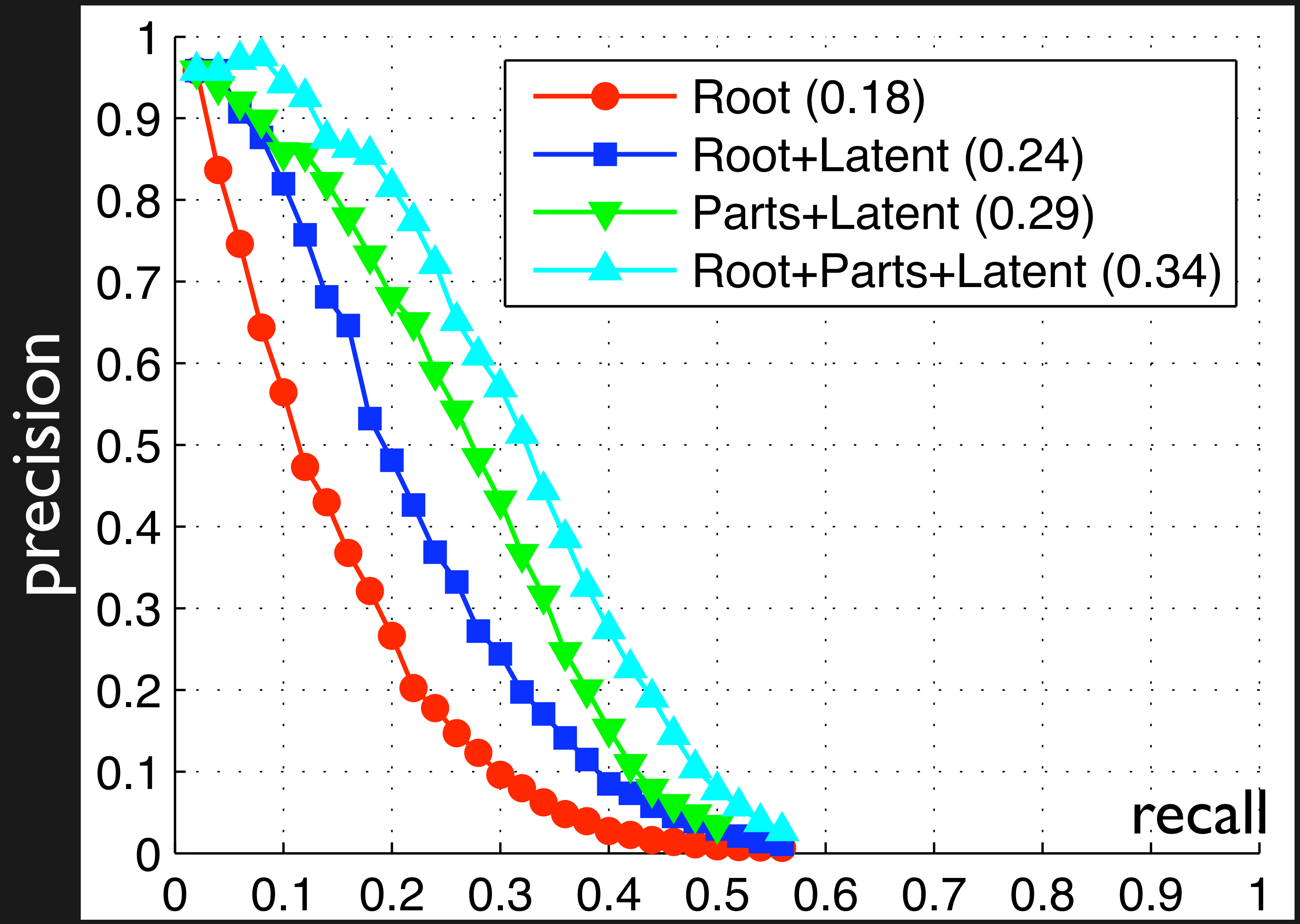
•Use $\Phi(x_i, z_i)$`s to train a new detector (w) with SVM-light (Joachims)

•Repeat update 10 times

Tried online updates; couldn't get it to work (Yan?)

Component analysis

PASCAL Person2006



- Factor of 2 improvement over ’06 winner - DalalTriggs (.16)
- Adjustment of b.box helps rigid template - blue
- Parts help - green
- Multiscale (parts + root together) helps - cyan

A look back

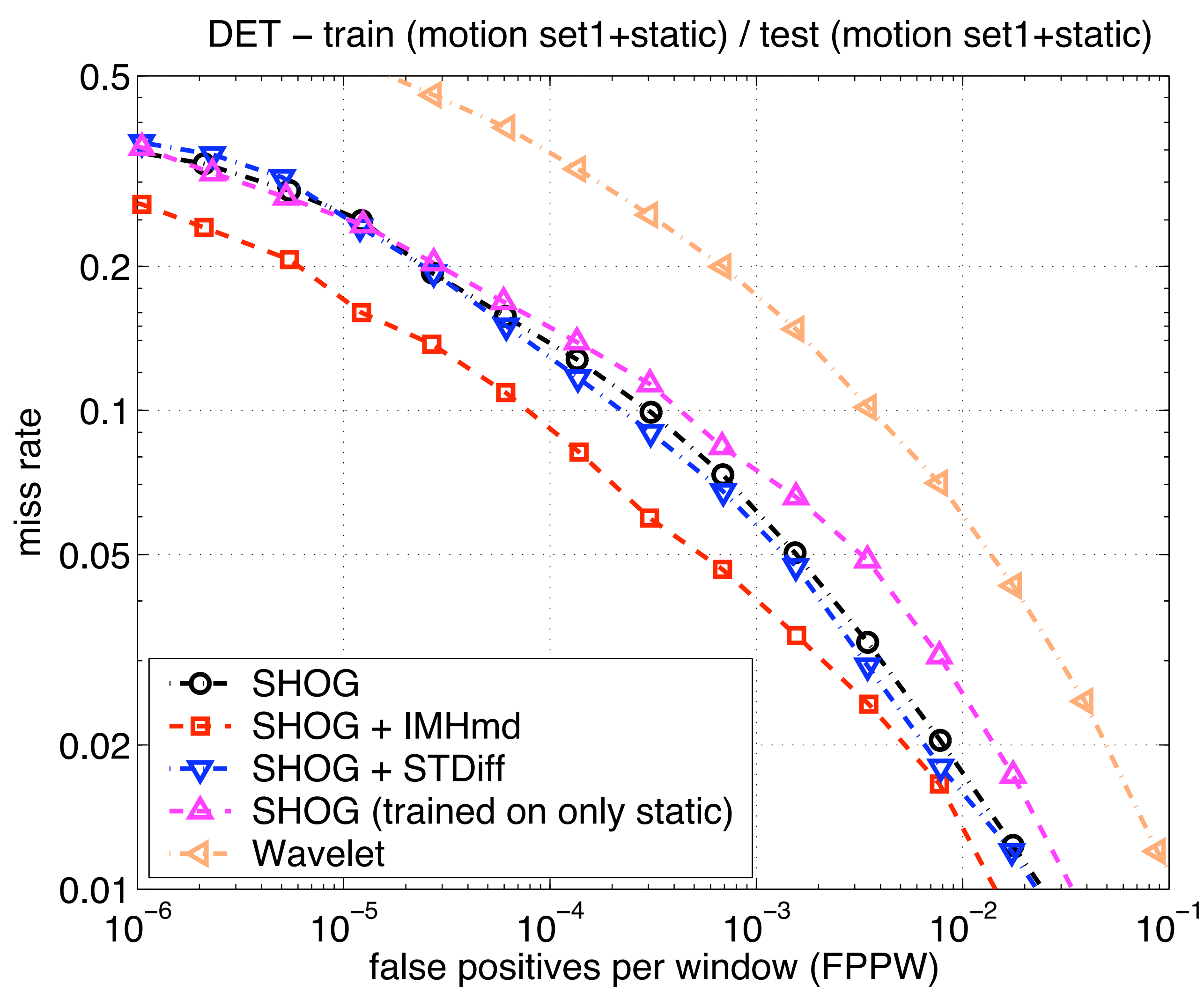
Training part-based models with classification machinery helps (cause of implicit bg model?)

Good classification <=> good object detection ?

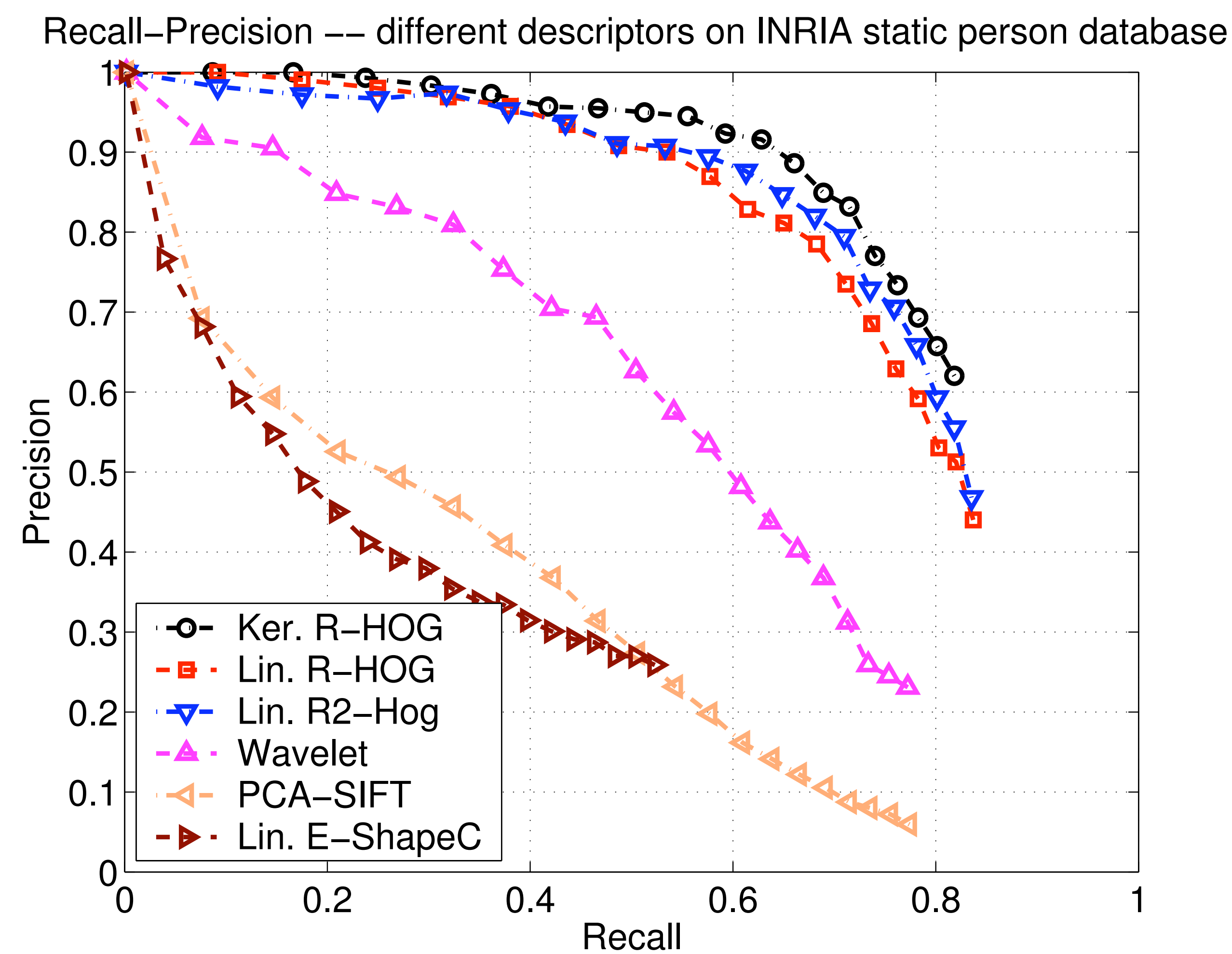
Oxford's results suggest so, but....

Classification vs Obj. Detection

False positives per window



fraction of detections that overlaps ground truth



(a)

HOG-based detectors again significantly outperform the wavelet based one, but surprisingly the combined static and motion HOG detector does not seem to offer a significant advantage over the static HOG one: The static detector gives an AP of 0.553 compared to 0.527 for the motion detector. These results are surprising and disappointing because Sect. 6.5.2, where we used DET curves (*c.f.* Sect. B.1) for evaluations, shows that for exactly the same data set, the individual window classifier for the motion detector gives significantly better performance than

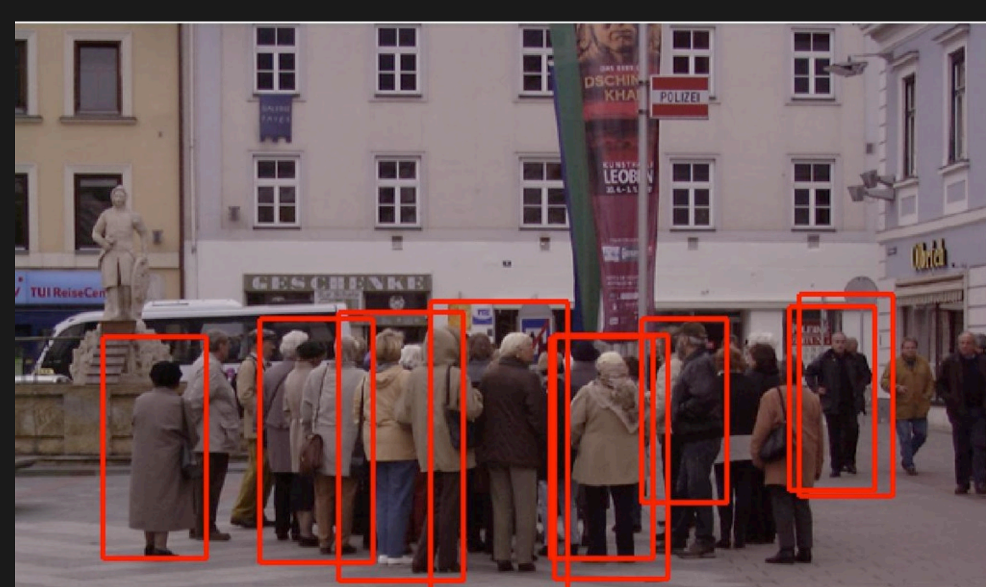
Dalal's thesis (p27): good classification does not imply good detection

Why not score FPPW?

1) Score is tied to resolution of scan
   (not valid for segmentation/pyramid-based search)

2) We can directly score the task we care about (DAF: Can we use it to avoid hitting pedestrians?)

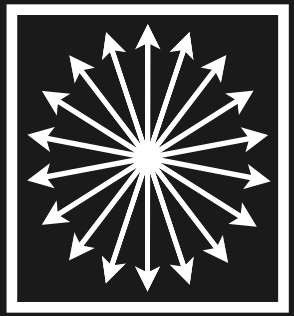
2) We need to account for non-max suppression (non-trivial: "auto-correlation" of detector response should be smooth and peaky)



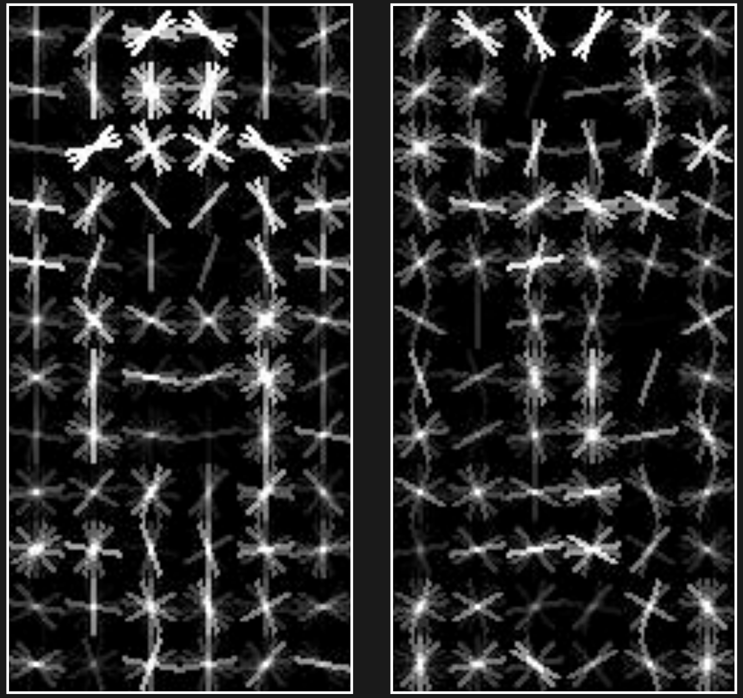
Conclusion

What makes our part model work?

-Histograms-of-gradient features



-Discriminatively-trained



-Multi-scale