Discriminatively Trained Mixtures of Deformable Part Models

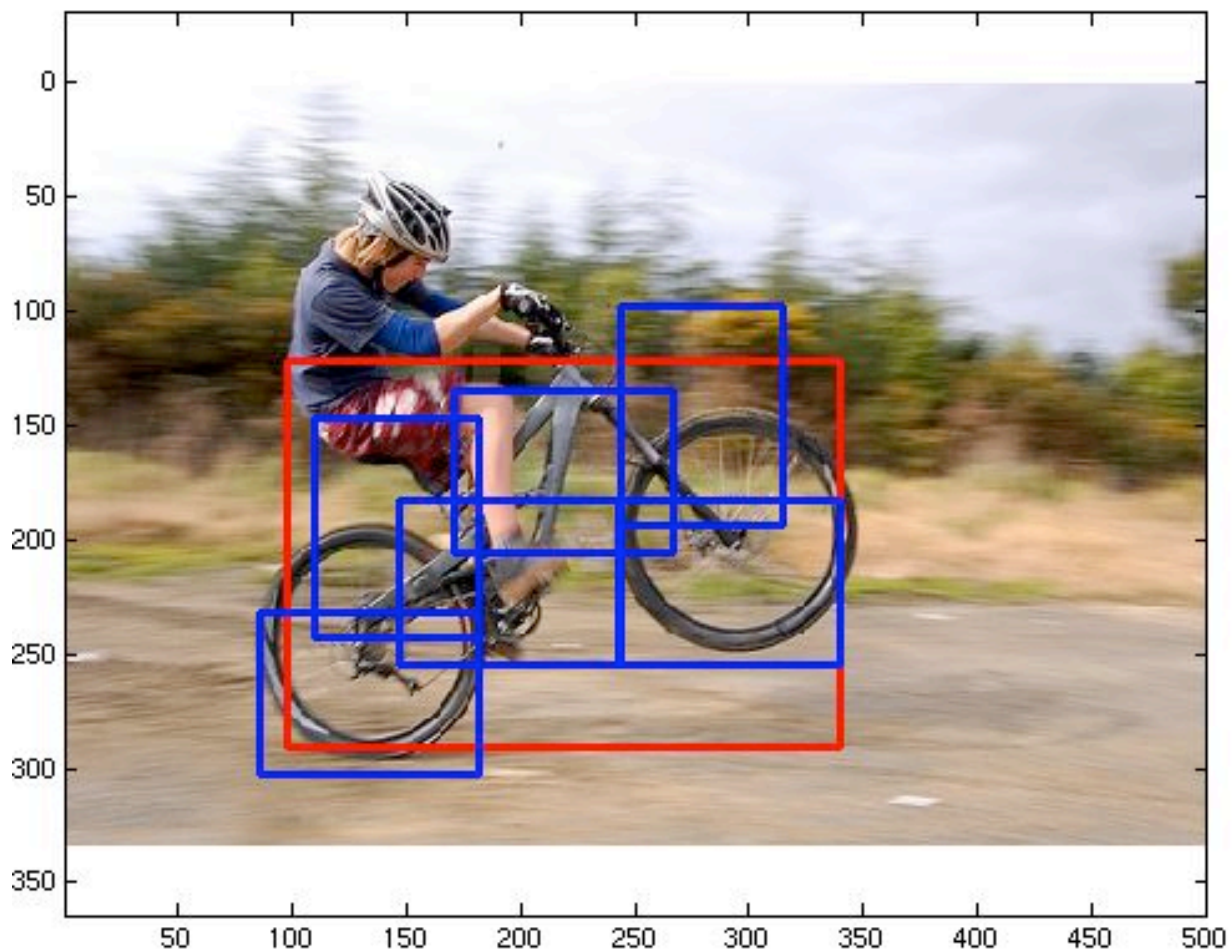
Pedro Felzenszwalb and Ross Girshick
University of Chicago

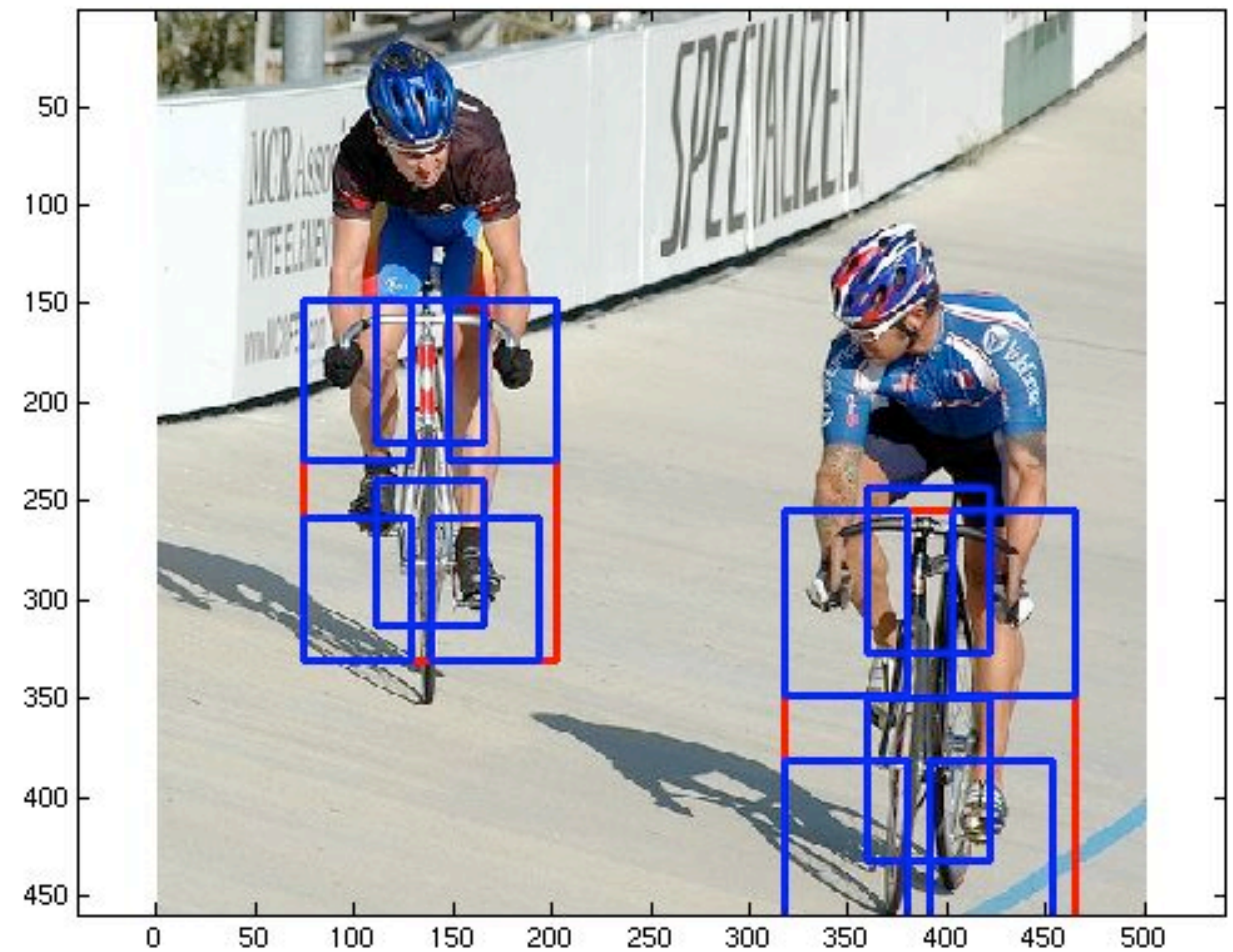
David McAllester
Toyota Technological Institute at Chicago

Deva Ramanan
UC Irvine

http://www.cs.uchicago.edu/~pff/latent

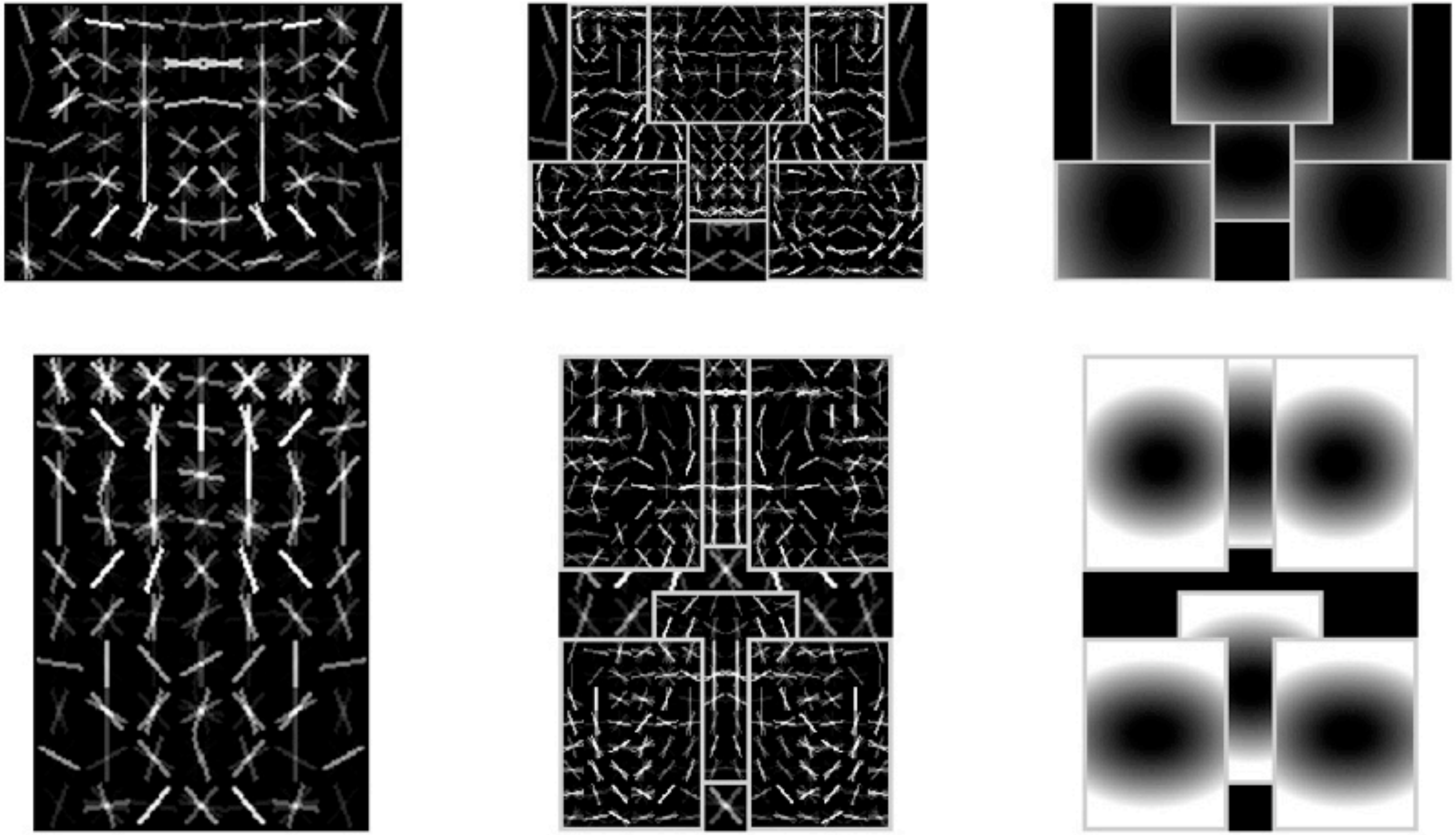
Model Overview





- Mixture of deformable part models (pictorial structures)
- Each component has global template + deformable parts
- Fully trained from bounding boxes alone

2 component bicycle model

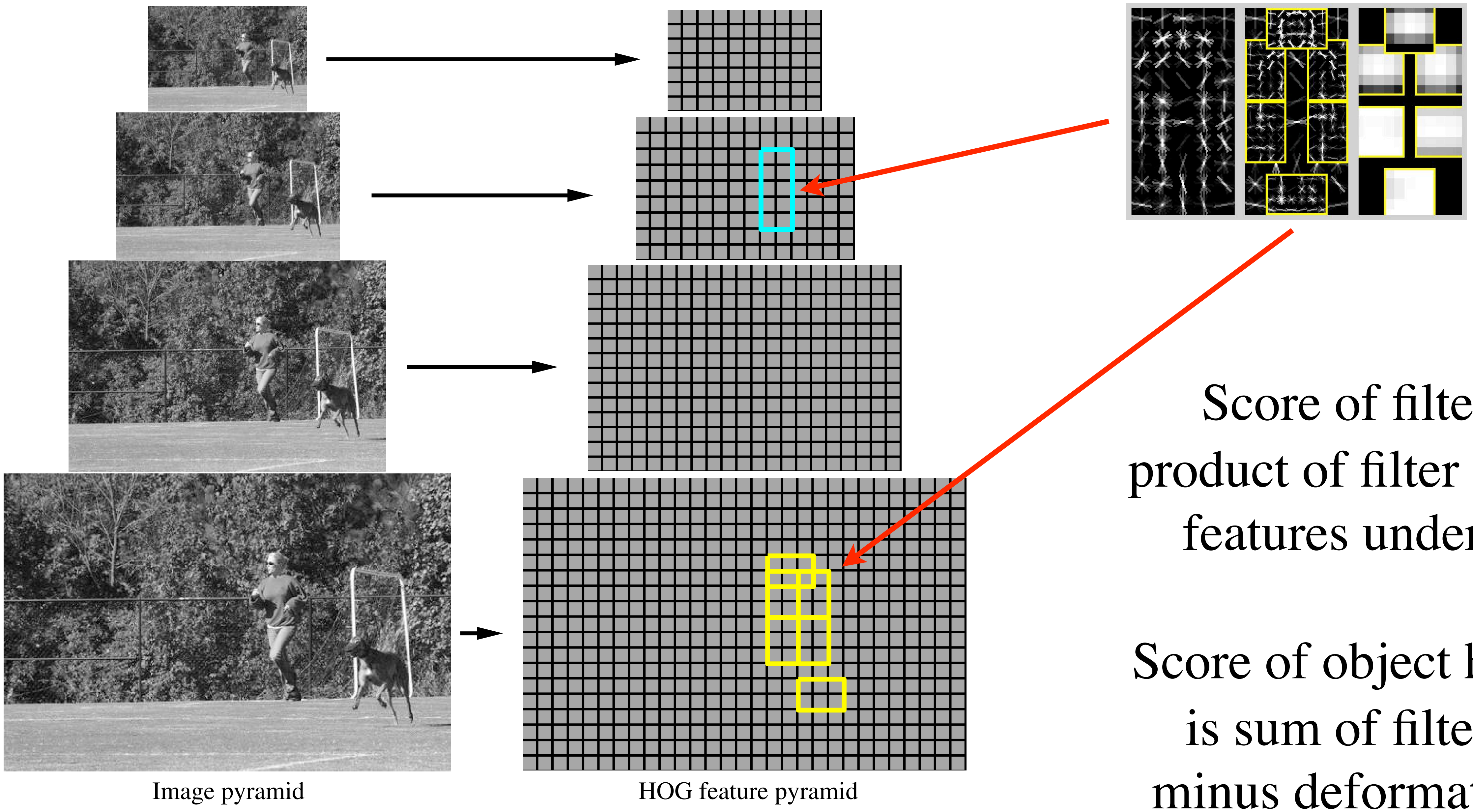


root filters
coarse resolution

part filters
finer resolution

deformation
models

Object Hypothesis



Score of filter is dot product of filter with HOG features underneath it

Score of object hypothesis is sum of filter scores minus deformation costs

Multiscale model captures features at two resolutions

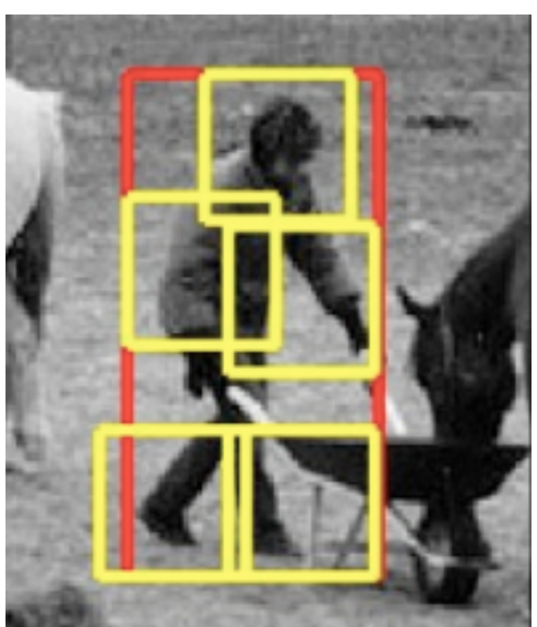
Connection with linear classifier

score on detection window x can be written as

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

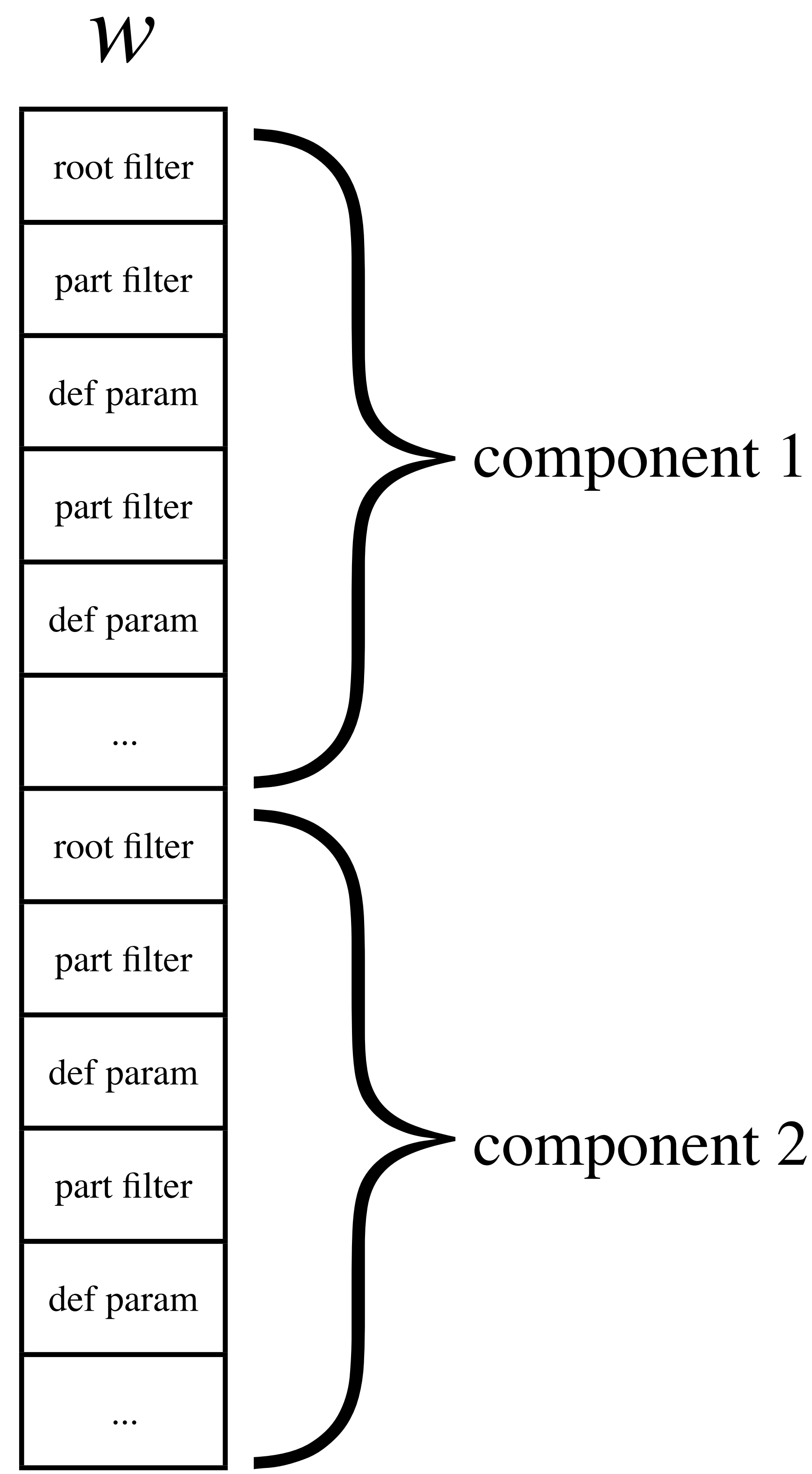
concatenation filters and deformation parameters

concatenation of HOG features and part displacements and 0's



w: model parameters

z: latent variables: component label and filter placements



Latent SVM

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

Linear in w if z is fixed

$$\text{Training data: } (x_1, y_1), \ldots, (x_n, y_n) \text{ with } y_i \in \{-1, 1\}$$

$$\text{Learning: find } w \text{ such that } y_i f_w(x_i) > 0$$

$$w^* = \underset{w}{\operatorname{argmin}}\, \lambda ||w||^2 + \sum_{i=1}^{n} \max(0, 1 - y_i f_w(x_i))$$

Regularization

Hinge loss

Latent SVM training

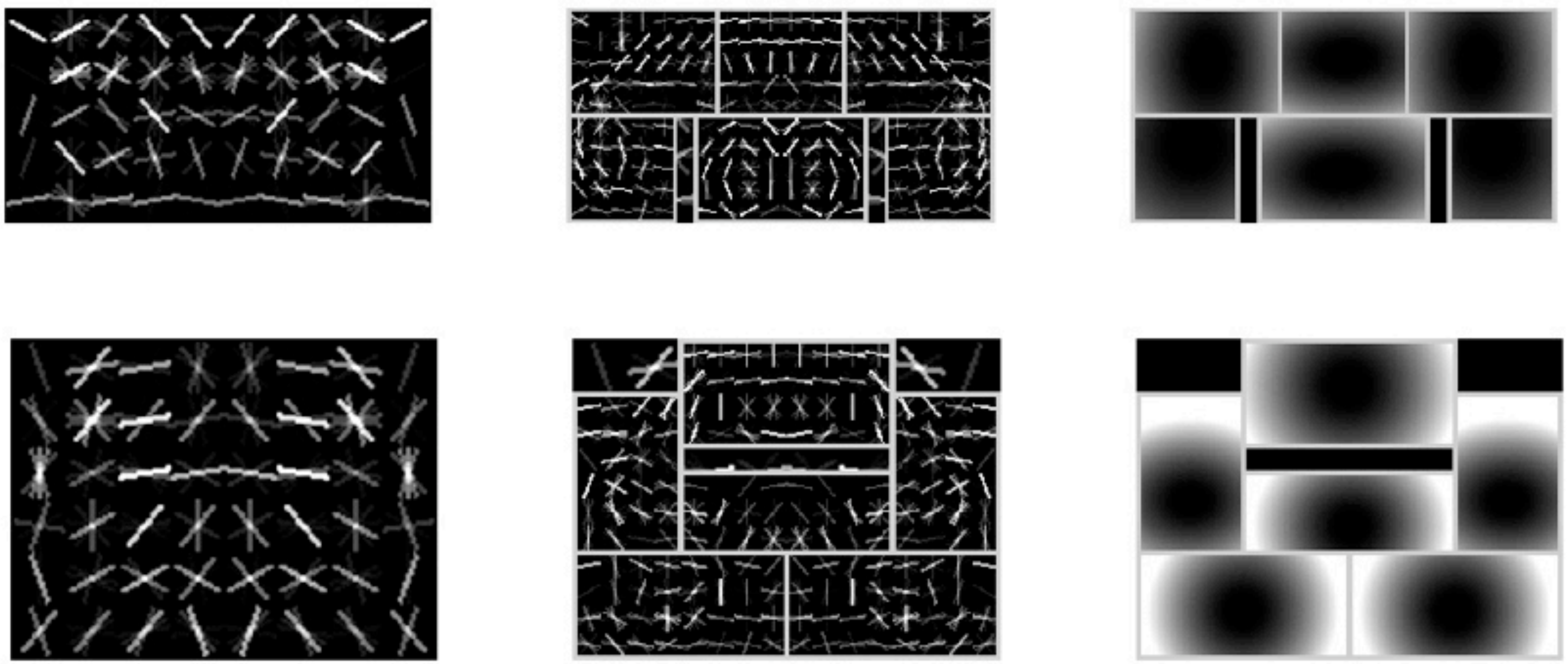
$$w^* = \underset{w}{\operatorname{argmin}}\, \lambda||w||^2 + \sum_{i=1}^{n} \max(0, 1 - y_i f_w(x_i))$$

- Non-convex optimization
- Huge number of negative examples
- Convex if we fix z for positive examples
- Optimization:
  - Initialize w and iterate:
    - Pick best z for each positive example
    - Optimize w via gradient descent with data mining

Initializing w

- For k component mixture model:
- Split examples into k sets based on bounding box aspect ratio
- Learn k root filters using standard SVM
  - Training data: warped positive examples and random windows from negative images (Dalal & Triggs)
- Initialize parts by selecting patches from root filters
  - Subwindows with strong coefficients
  - Interpolate to get higher resolution filters
  - Initialize spatial model using fixed spring constants

Car model

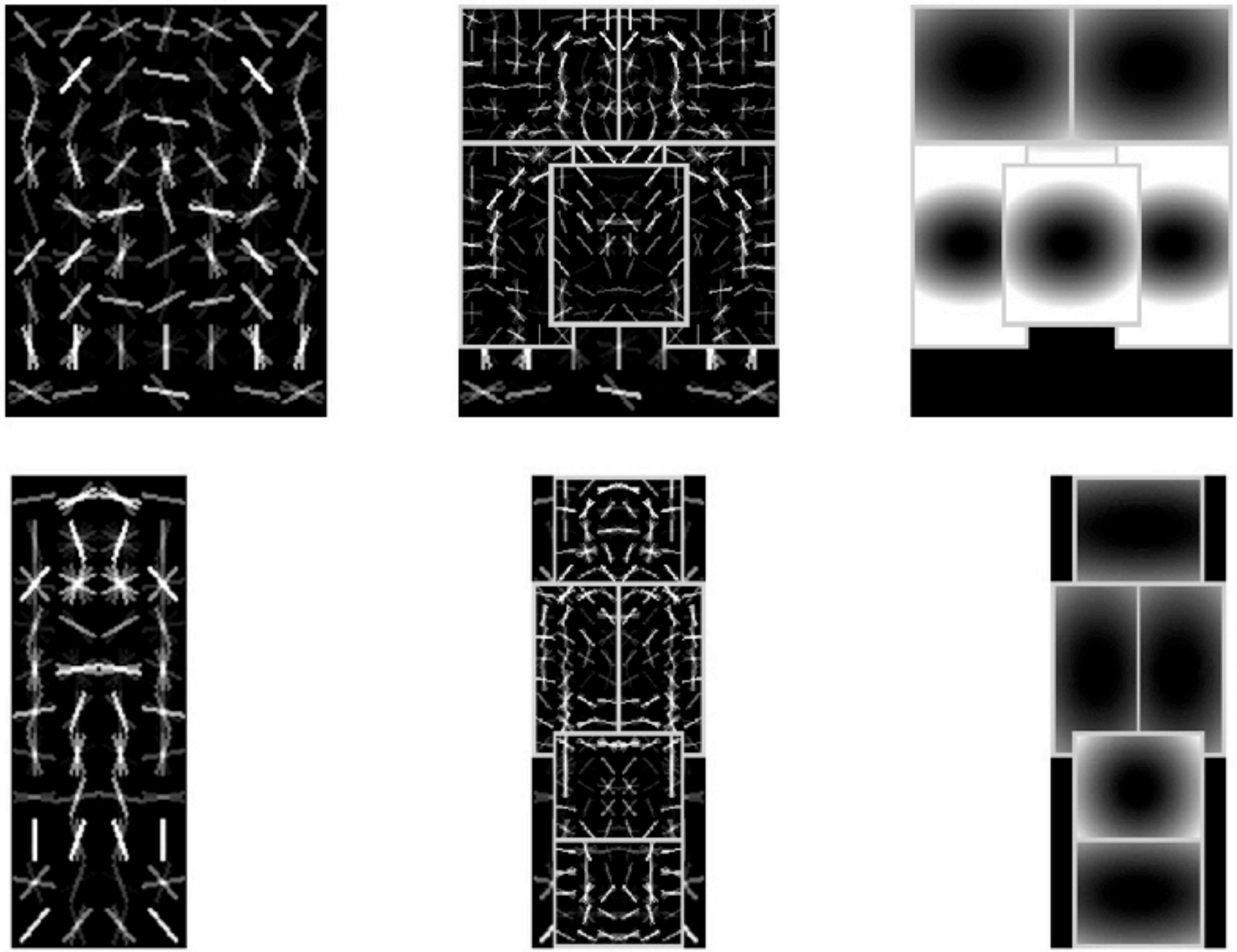


root filters
coarse resolution

part filters
finer resolution

deformation
models

Person model

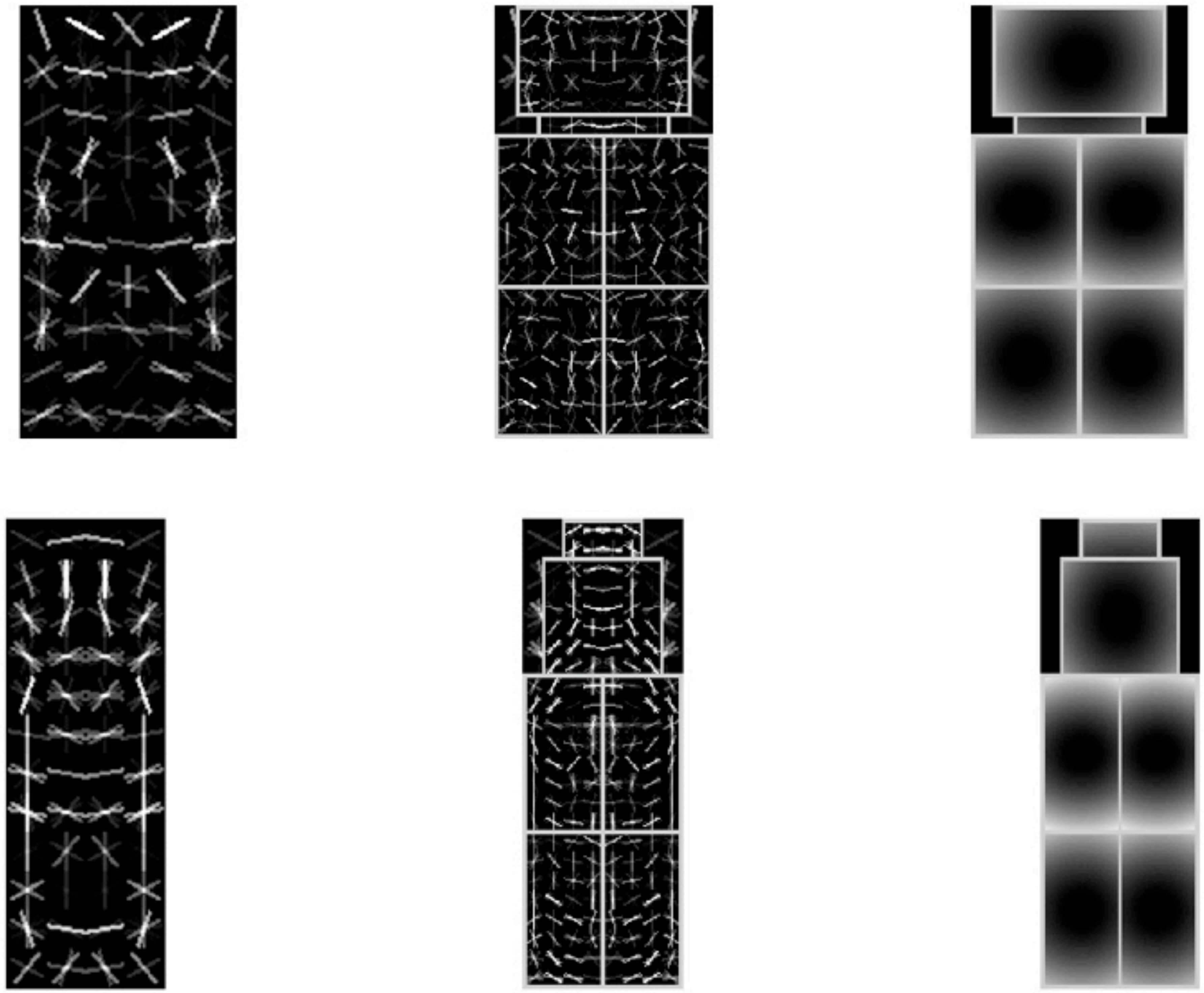


root filters
coarse resolution

part filters
finer resolution

deformation
models

Bottle model

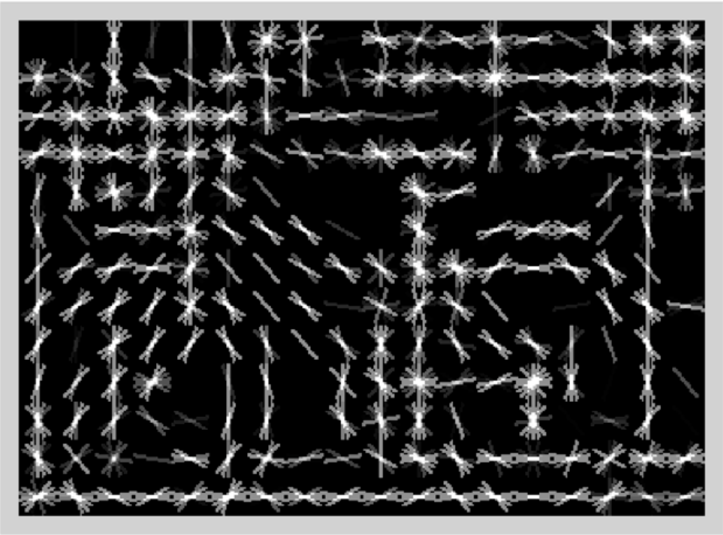


root filters
coarse resolution

part filters
finer resolution

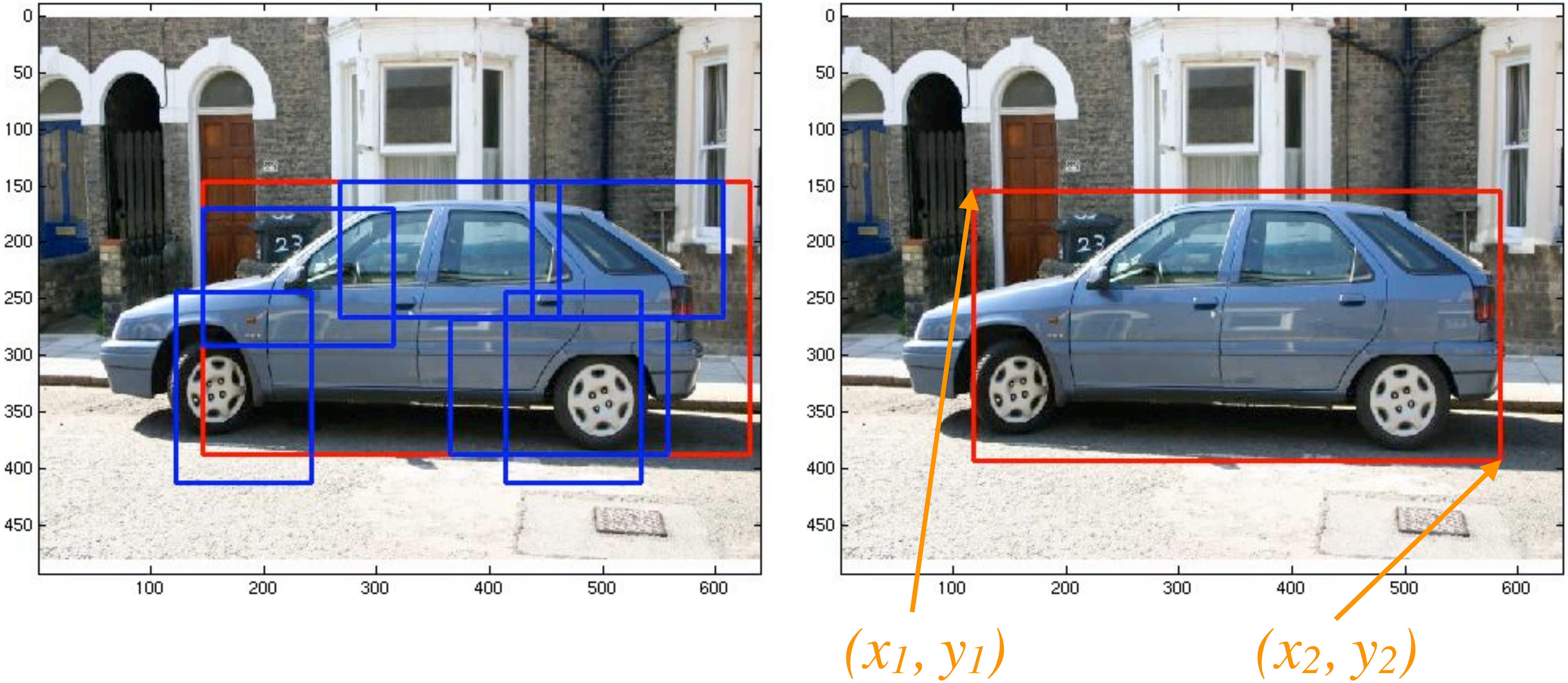
deformation
models

Histogram of Gradient (HOG) features



- Dalal & Triggs:
  - Histogram gradient orientations in 8x8 pixel blocks (9 bins)
  - Normalize with respect to 4 different neighborhoods and truncate
  - 9 orientations * 4 normalizations = 36 features per block
- PCA gives ~10 features that capture all information
  - Fewer parameters, speeds up convolution, but costly projection at runtime
- Analytic projection: spans PCA subspace and easy to compute
  - 9 orientations + 4 normalizations = 13 features
- We also use 2*9 contrast sensitive features for 31 features total

Bounding box prediction

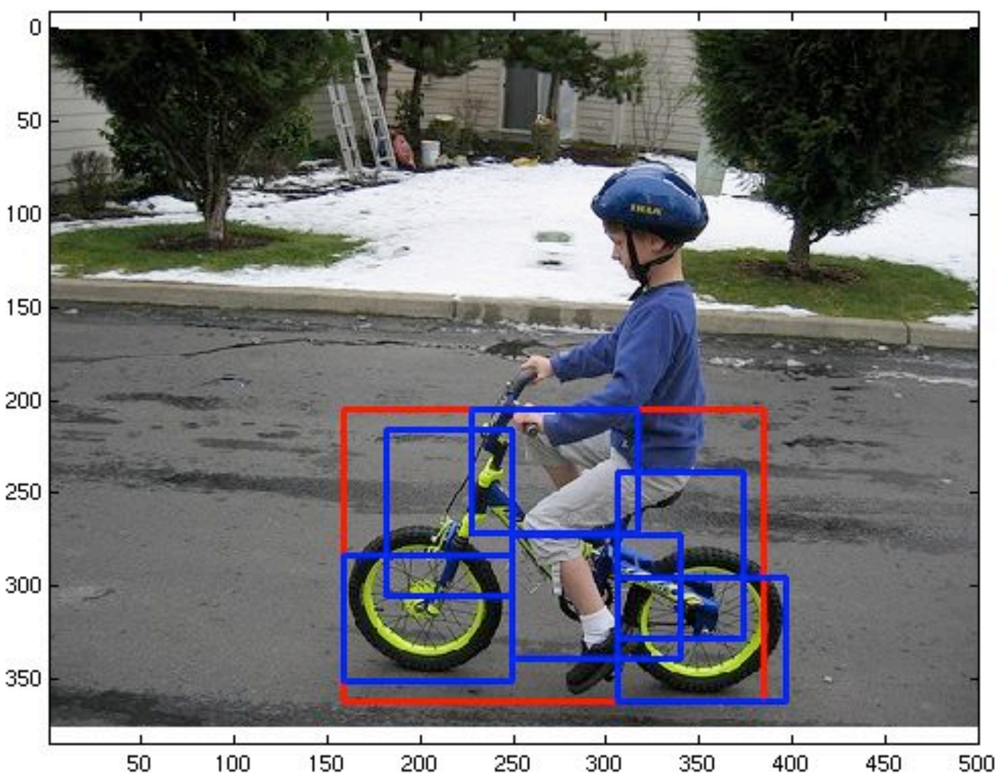


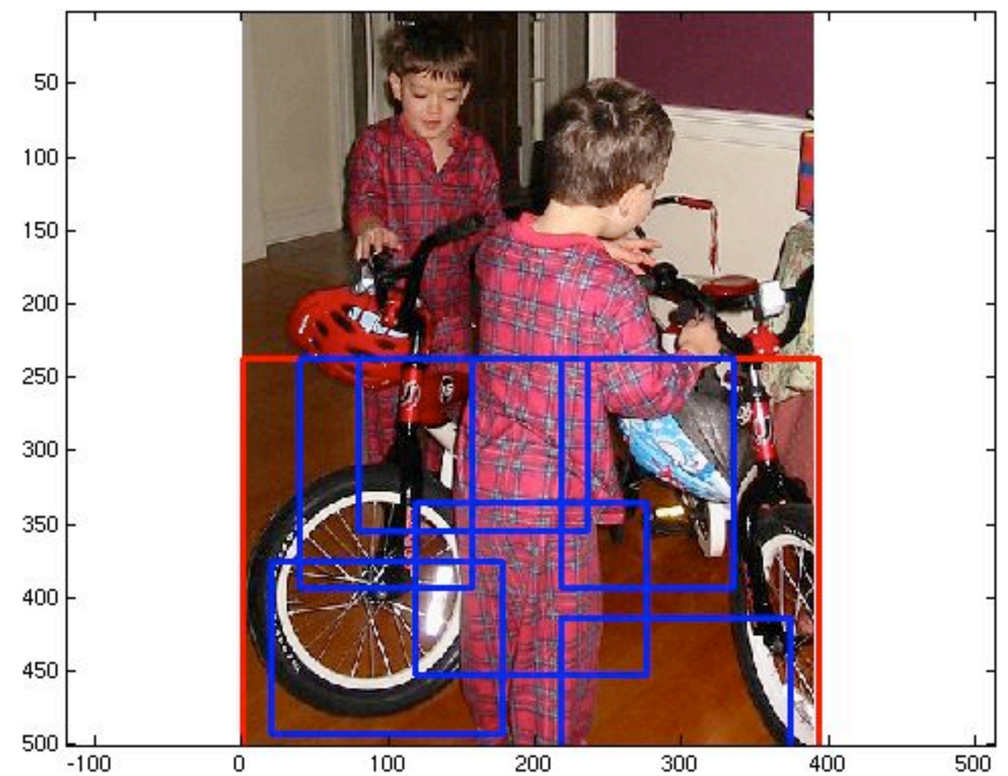
- predict (x_1, y_1) and (x_2, y_2) from part locations
- linear function trained using least-squares regression

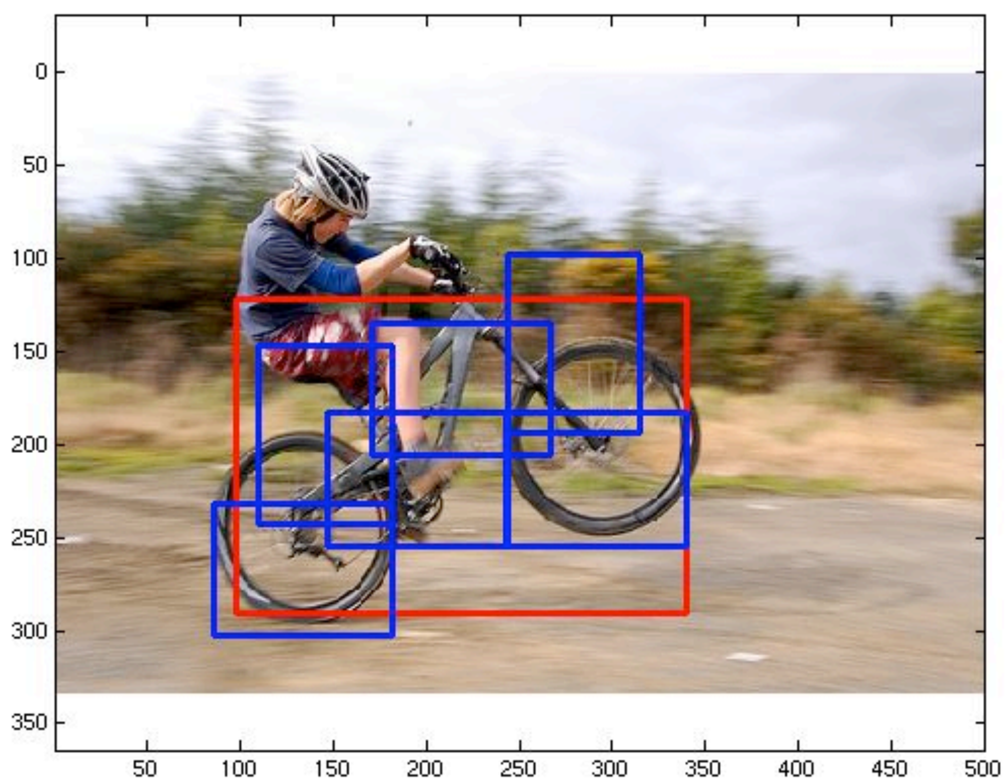
Context rescoring

- Rescore a detection using "context" defined by all detections
- Let v_i be the max score of detector for class i in the image
- Let s be the score of a particular detection
- Let (x_1,y_1), (x_2,y_2) be normalized bounding box coordinates
- $f = (s, x_1, y_1, x_2, y_2, v_1, v_2 \ldots , v_{20})$
- Train class specific classifier
    - f is positive example if true positive detection
    - f is negative example if false positive detection

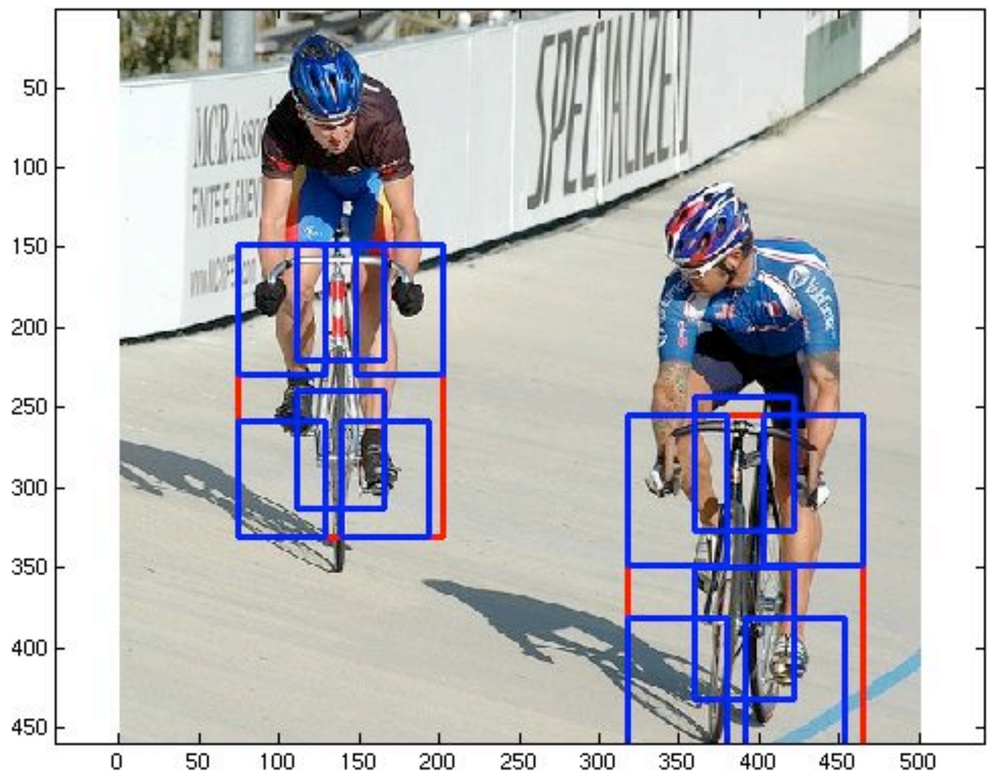
Bicycle detection

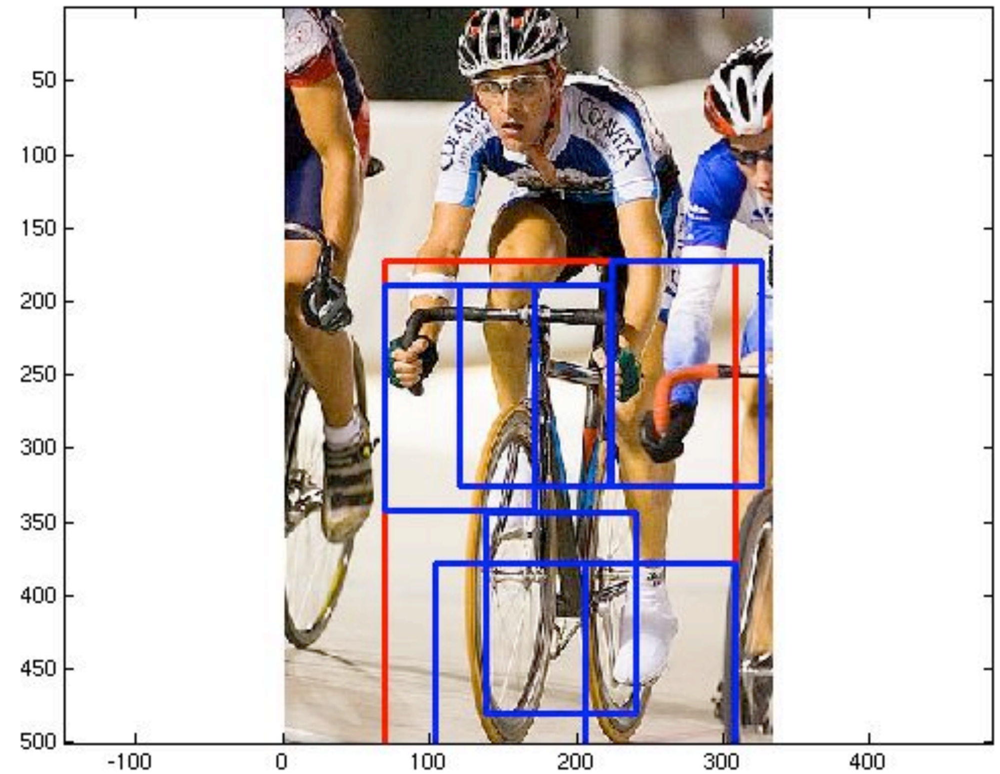




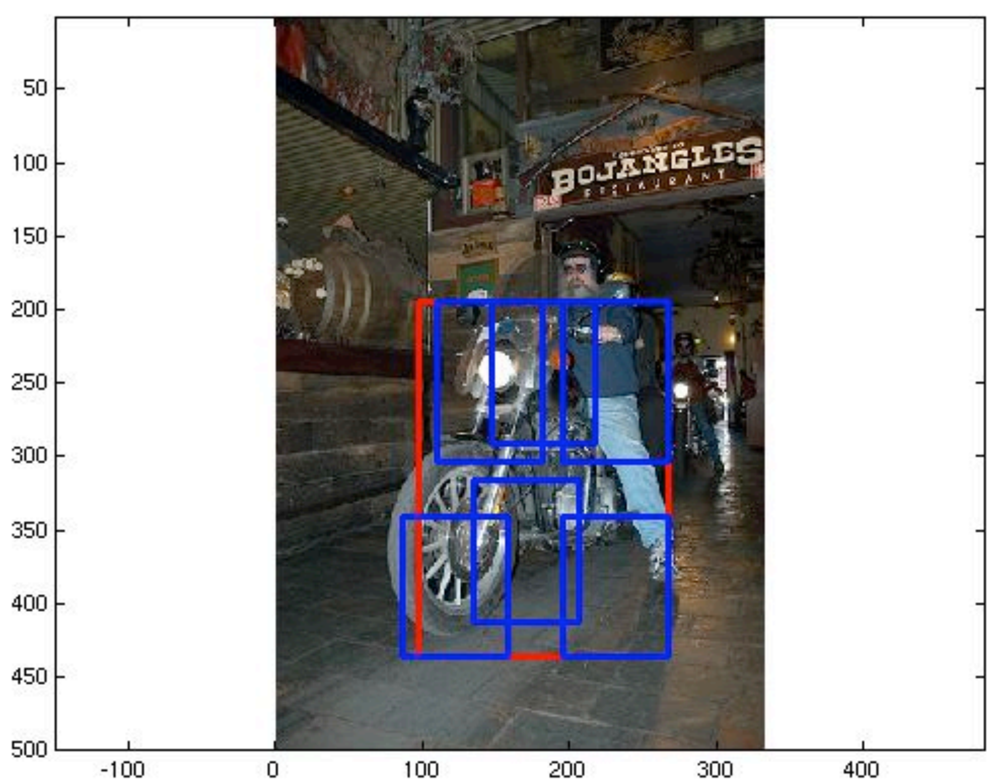


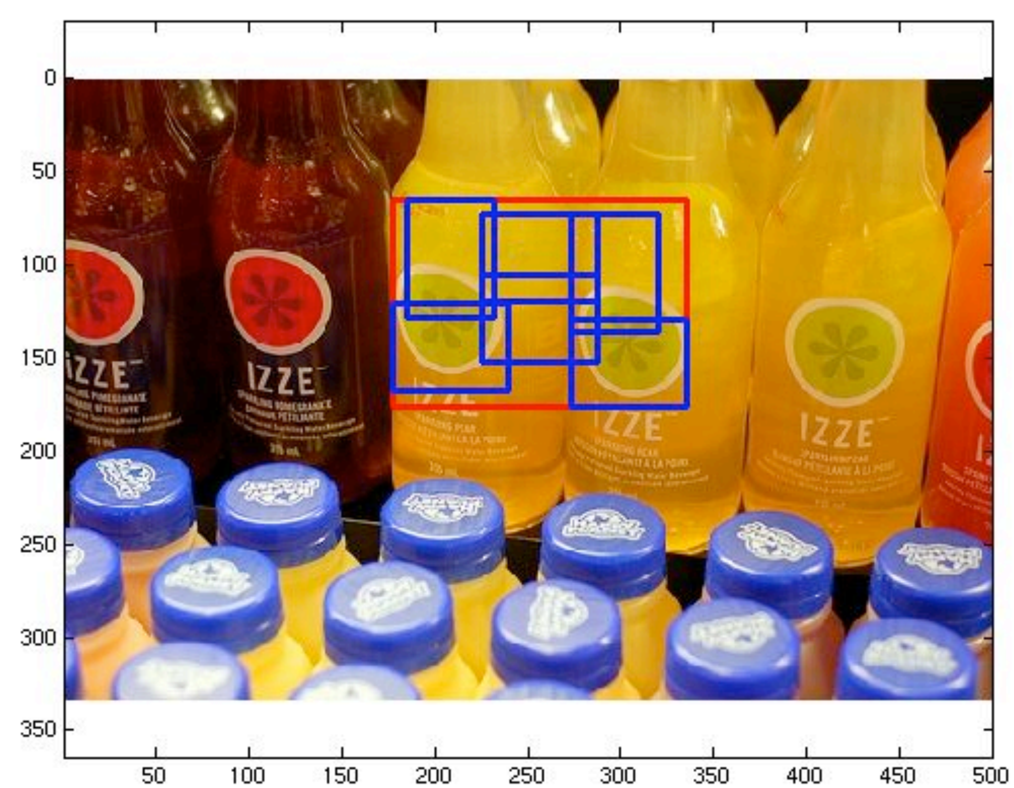
More bicycles



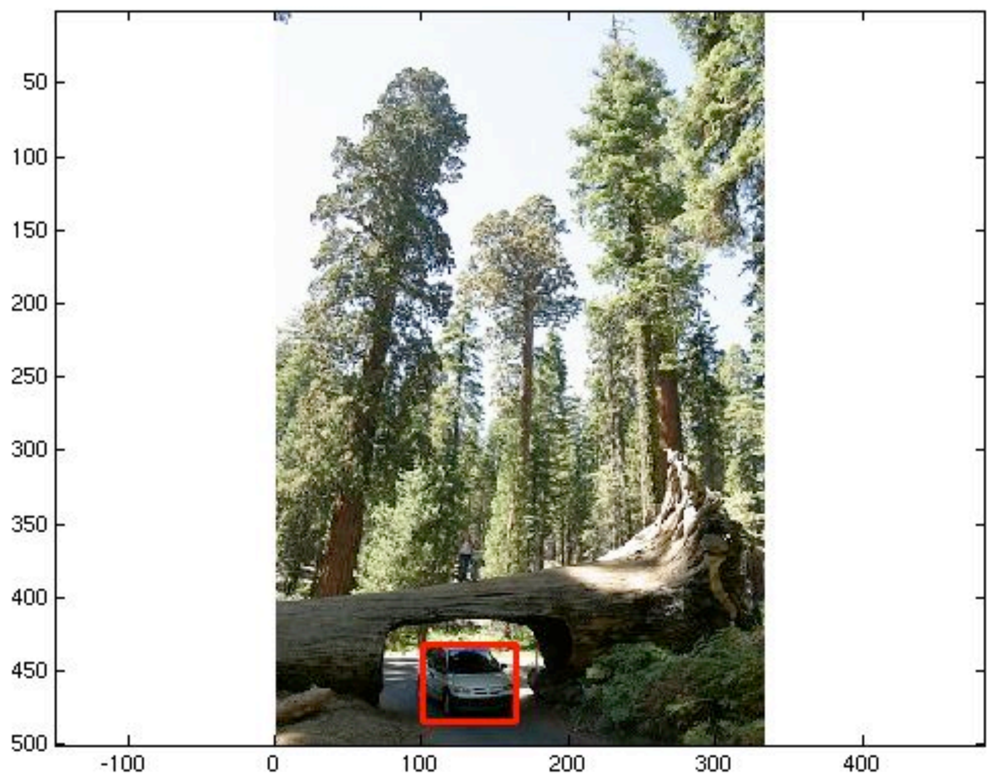


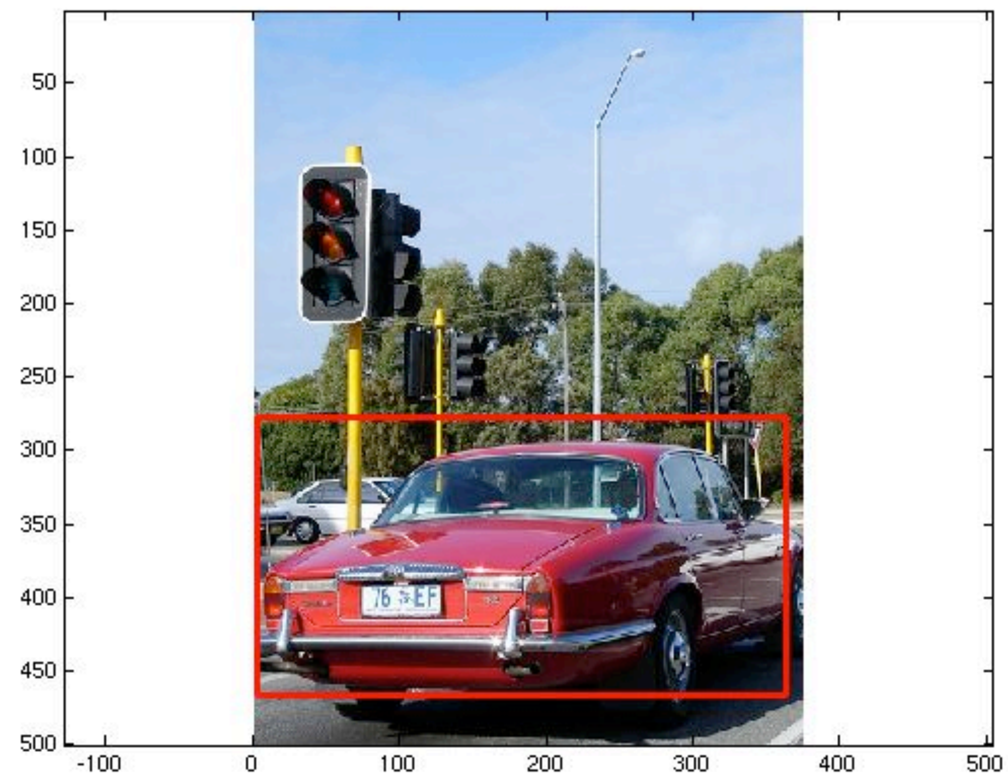
False positives

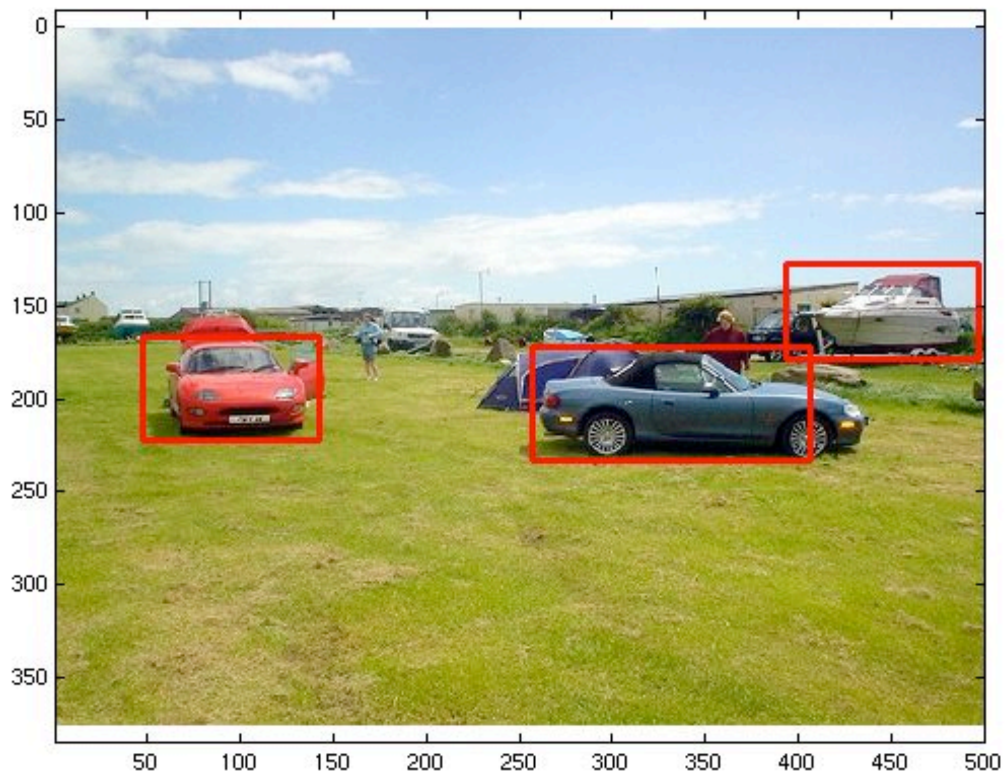


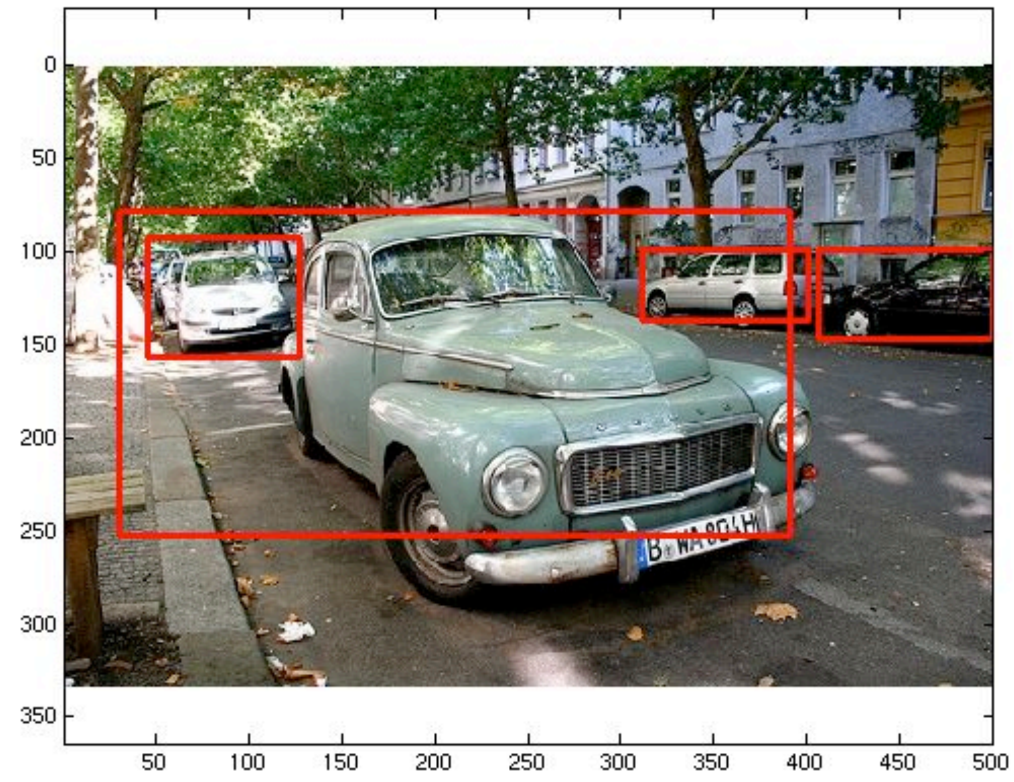


Car

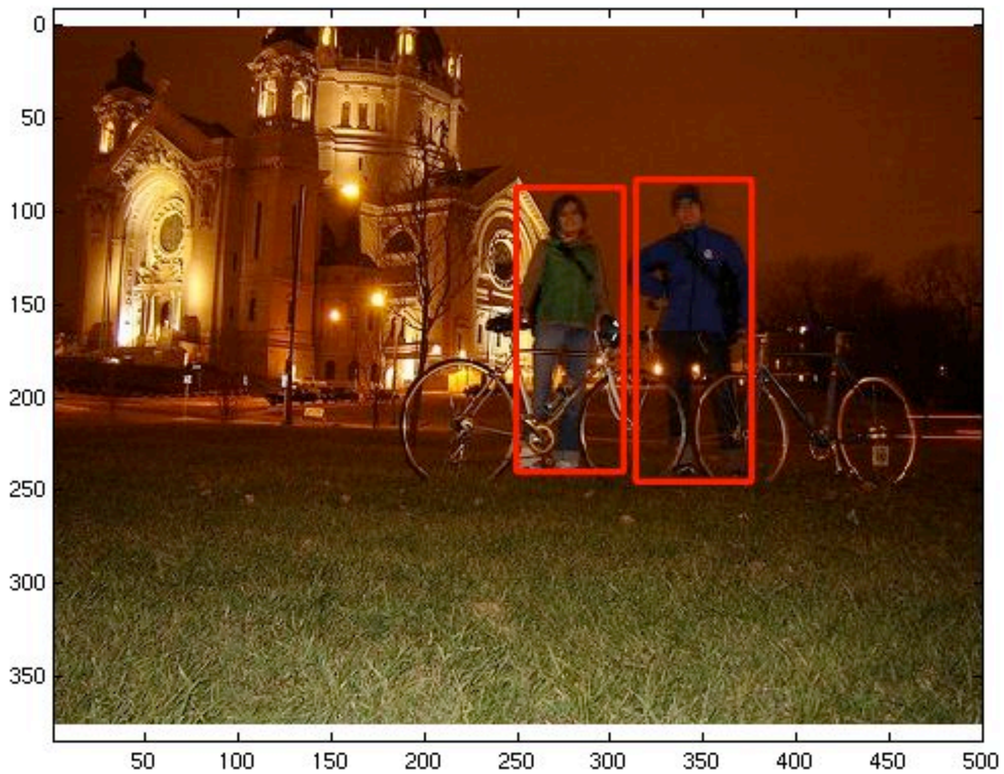

50
100
150
200
250
300
350
400
450
500
-100
0
100
200
300
400

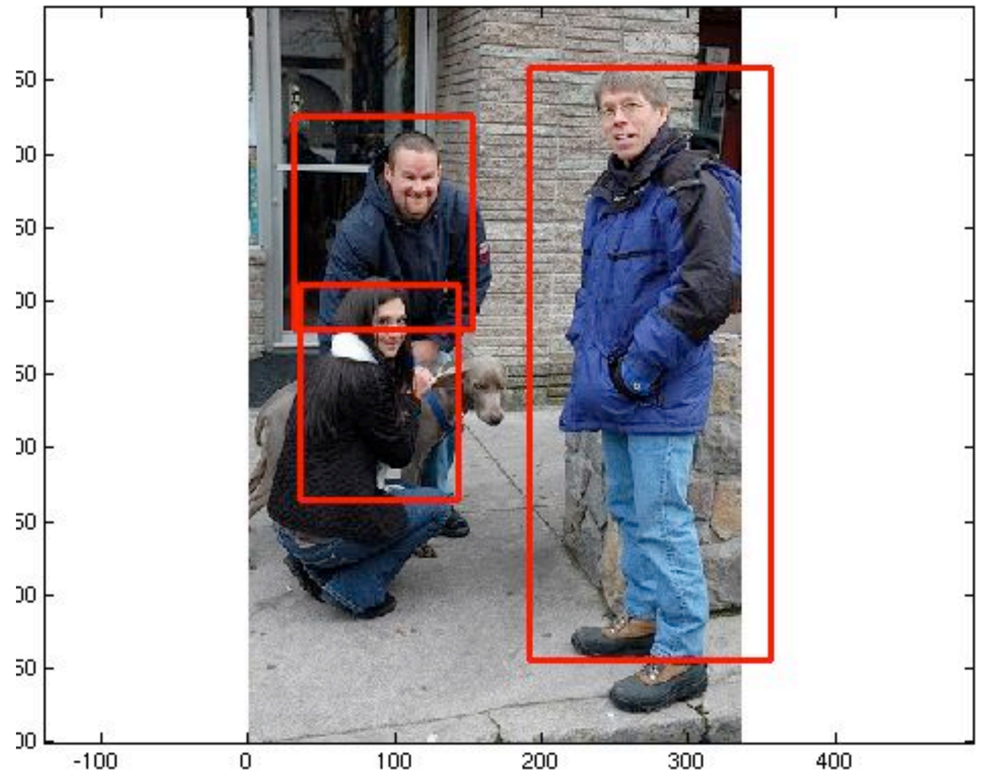

50
100
150
200
250
300
350
400
450
500
-100
0
100
200
300
400
500


0
50
100
150
200
250
300
350
50
100
150
200
250
300
350
400
450
500


0
50
100
150
200
250
300
350
50
100
150
200
250
300
350
400
450
500

Person





Bottle

Horse

Code

Source code for the system and models trained on PASCAL 2006, 2007 and 2008 data are available here:

http://www.cs.uchicago.edu/~pff/latent