Deformable Part Models

References:

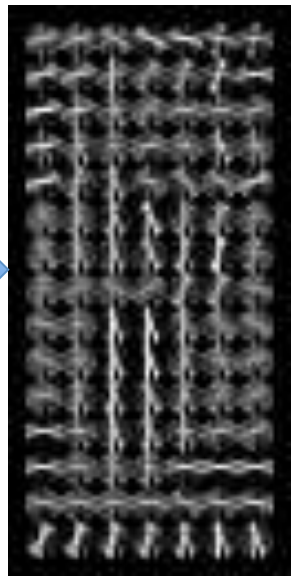
Felzenszwalb, Girshick, McAllester and Ramanan, “Object Detection with Discriminatively Trained Part Based Models,” PAMI 2010

Code available at http://www.cs.berkeley.edu/~rbg/latent/

Recall: Dalal and Triggs, 2005

- Detect upright pedestrians
- Histogram of oriented gradient feature vector
- Linear SVM classifier; sliding window detector

64X128



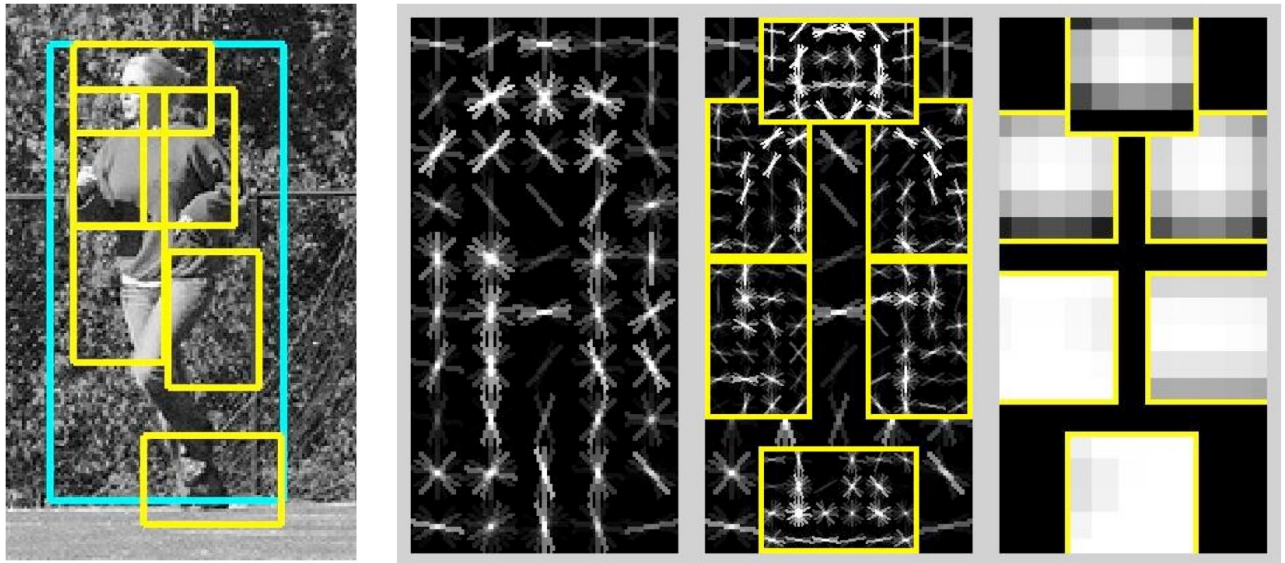
HoG descriptor

$\mathbf{x}_i \in \mathbb{R}^d, \text{ with } d = 1024$

Sliding window classifier

$$f(x) = \mathbf{w}^\top \mathbf{x} + b$$

Today: Deformable Part Models



detection root filter part filters deformation models

Model has a root filter plus deformable parts

Slide from Pedro Felzenswalb

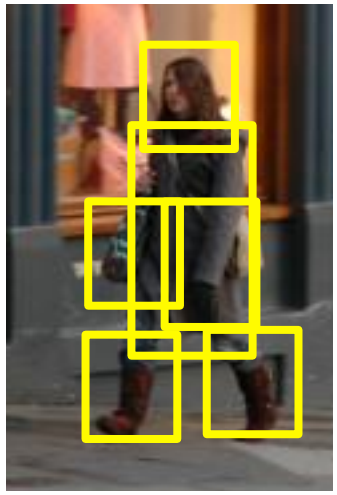
Motivation

- Simple template-based object models lack ability to handle geometric deformations due to articulation and pose.
- Simple bag of words models have no trouble with deformations, but are limited in their ability to finely localize objects.
- Want to make more expressive models based on geometric deformations of a canonical configuration of object parts.

Motivation

- Problem: More expressive object models are difficult to train because they often use latent (unobserved/unlabeled) information.
- Example: learning a part-based model from images where only bounding boxes are labeled.

What we have: bounding box



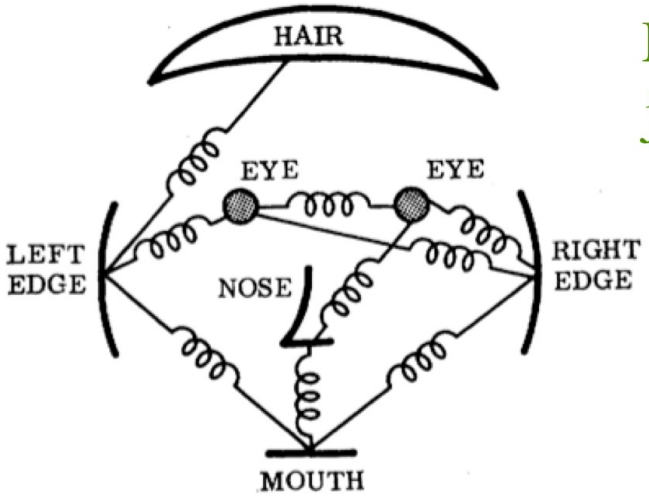
What we want: torso, head, arms and legs

DPM Innovations

- Star-structured graph model to represent body parts and their geometric relations
- Mixtures of models to handle large variations in viewpoint / pose
- Latent-SVM formulation
- Data mining of hard negative examples
- PCA on HOG features for dimension reduction
- Performance on PASCAL challenge dataset

History: Pictorial Structures

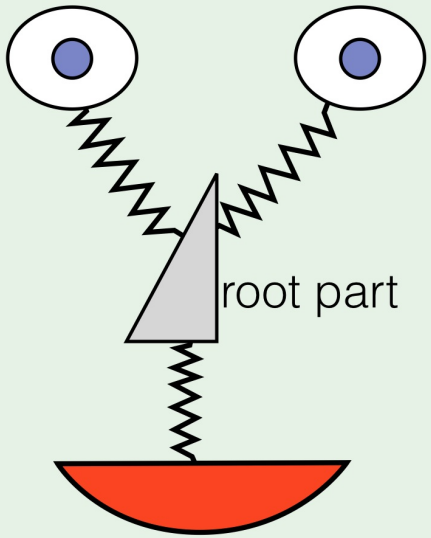
- Introduced by Fischler and Elschlager in 1973
- Part-based models:
  - Each part represents local visual properties
  - "Springs" capture spatial relationships



Matching model to image involves joint optimization of part locations "stretch and fit"

Slide from Pedro Felzenswalb

History: Pictorial Structures



test image

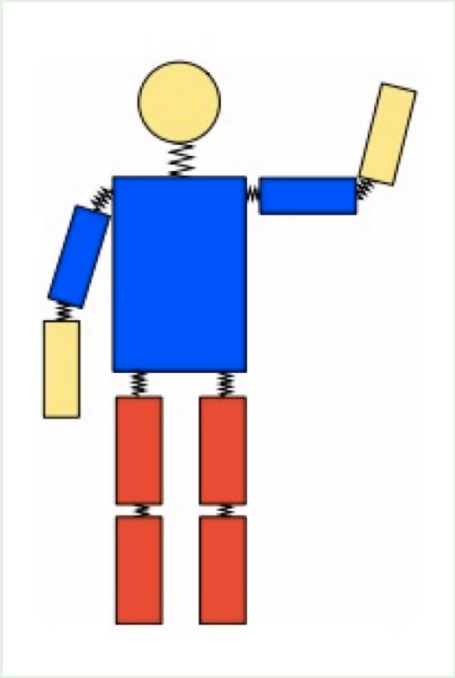
"star" model

detection

Goal: alignment of part model with features in an image.

Slide from Ross Girshick

History: Pictorial Structures



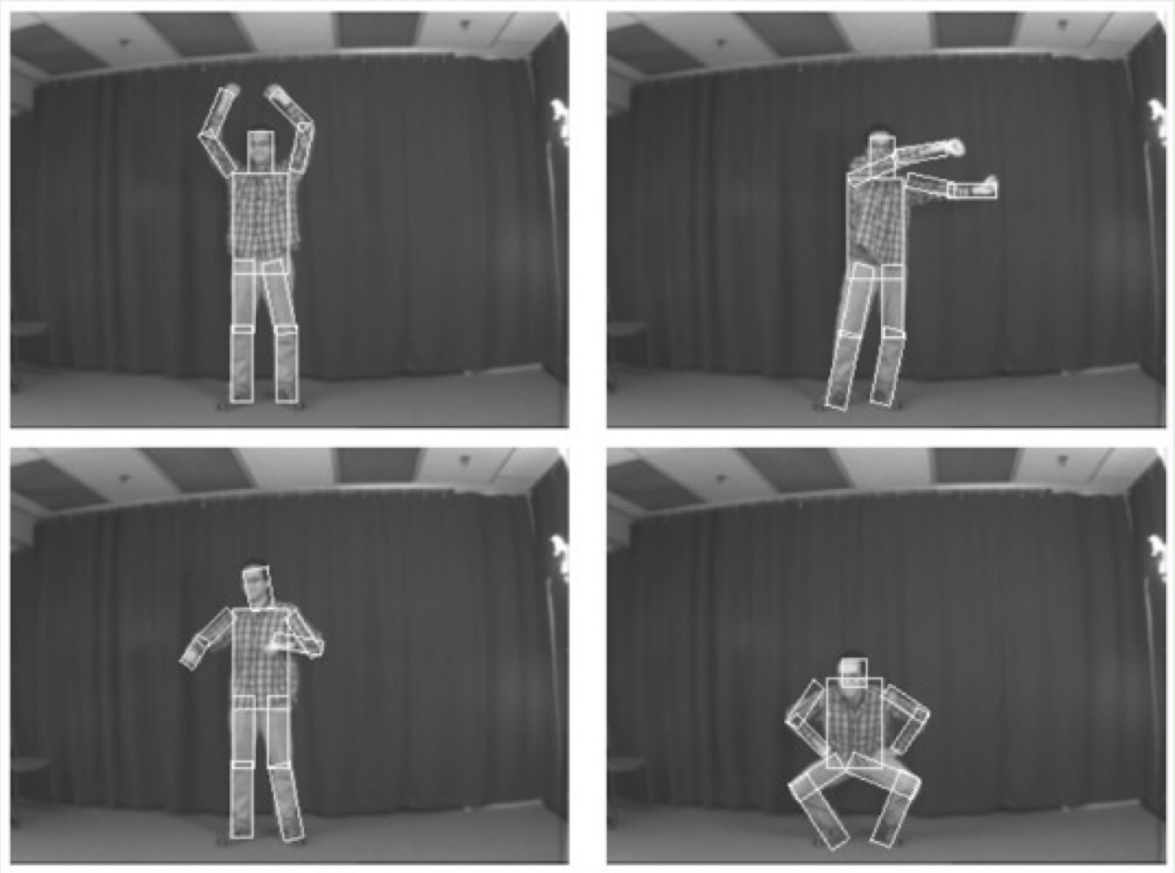


Image: [Felzenszwalb and Huttenlocher 05]

Felzenswalb showed that the best alignment of parts for a tree-structured model can be computed efficiently using dynamic programming.

Slide from Ross Girshick

Pictorial Structures: Matching

- Model is represented by a graph $G = (V, E)$
  - $V = \{v_1,...,v_n\}$ are the parts
  - $(v_i,v_j) \in E$ indicates a connection between parts
- $m_i(l_i)$ is score for placing part i at location l_i
- $d_{ij}(l_i,l_j)$ is a deformation cost
- Optimal configuration for the object is $L = (l_1,...,l_n)$ maximizing

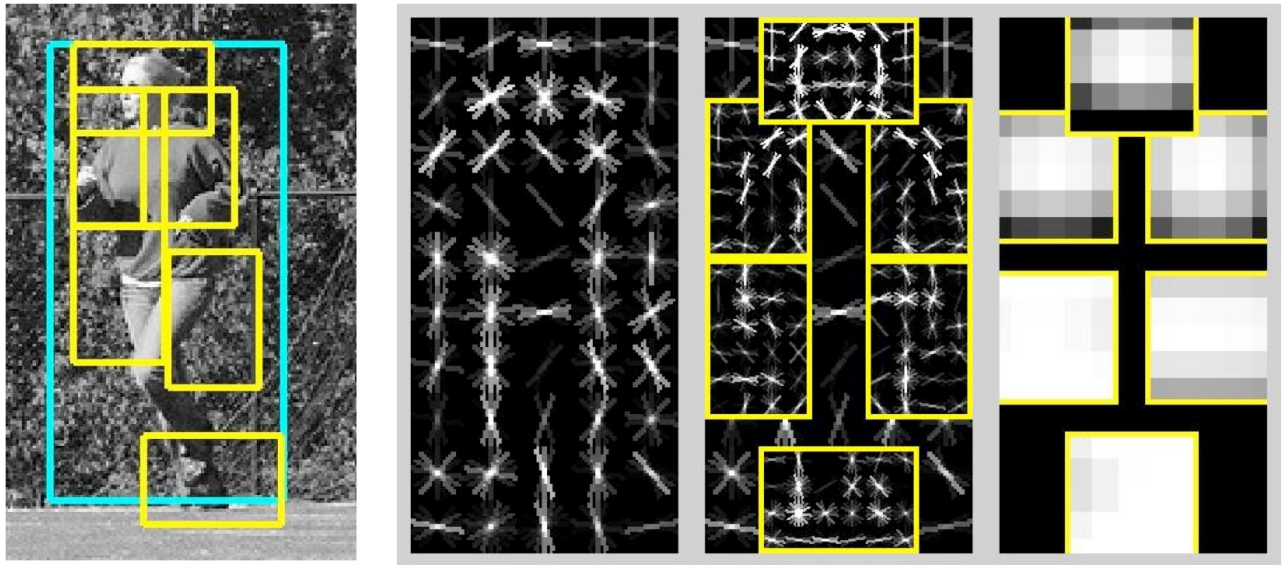
$$E(L) = \sum_{i=1}^{n} m_i(l_i) - \sum_{(v_i,v_j) \in E} d_{ij}(l_i,l_j)$$

Slide from Pedro Felzenswalb

From Pictorial Structures to DPM

Specifies unary costs.
Multiscale detector.

Pairwise costs.

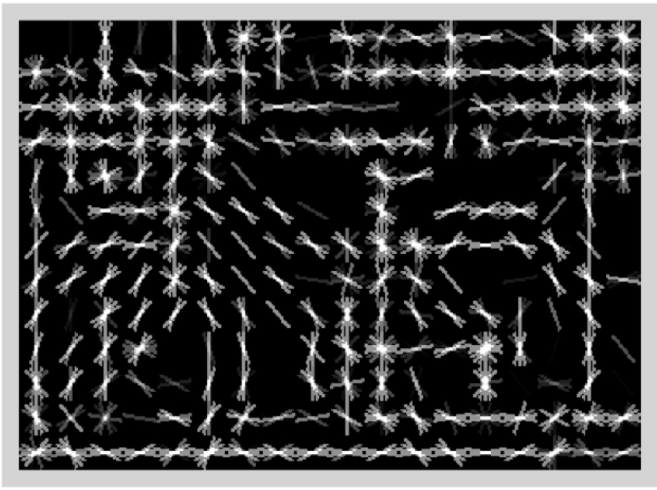


detection root filter part filters deformation models

Model has a root filter plus deformable parts

Slide from Pedro Felzenswalb

Histogram of Gradient (HOG) Features

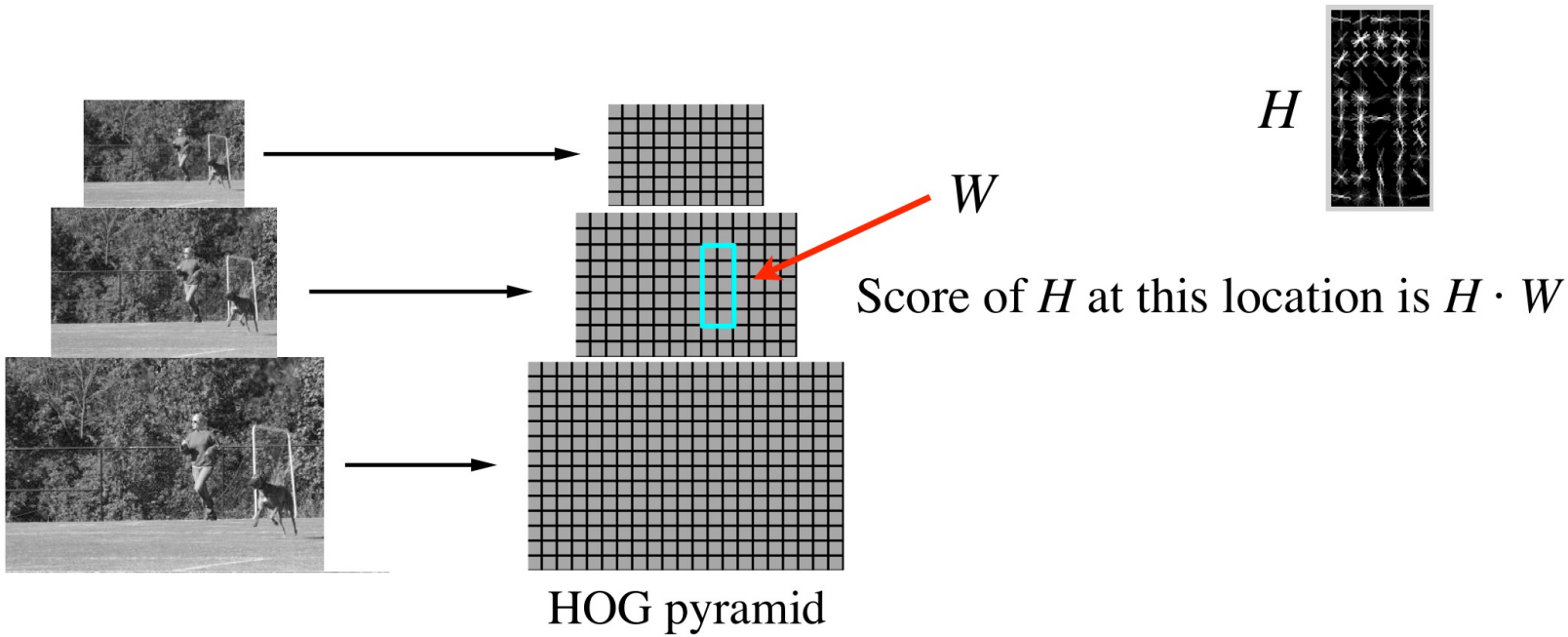


- Image is partitioned into 8x8 pixel blocks
- In each block we compute a histogram of gradient orientations
  - Invariant to changes in lighting, small deformations, etc.
- We compute features at different resolutions (pyramid)

Slide from Pedro Felzenswalb

Filters

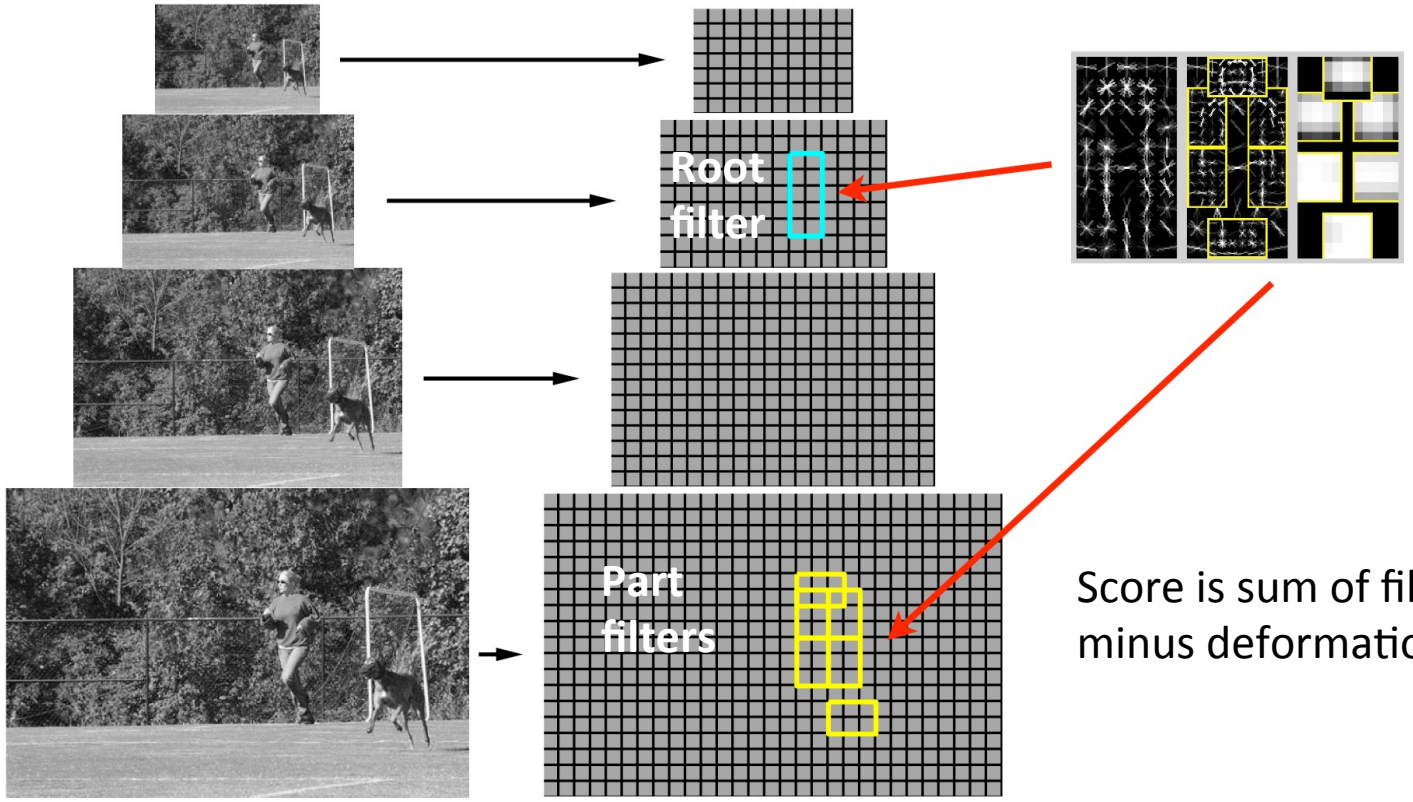
- Filters are rectangular templates defining weights for features
- Score is dot product of filter and subwindow of HOG pyramid



Dalal and Triggs, single template

Slide from Pedro Felzenswalb

Object Hypothesis



Score is sum of filter scores
minus deformation costs

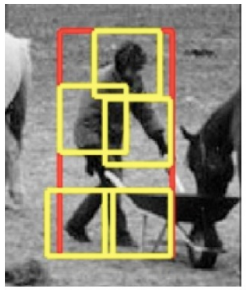
Multiscale model captures features at two-resolutions

Parts are represented at twice the resolution of the root filter.

Slide from Pedro Felzenswalb

Scoring with Linear Classifiers

- Score of model is sum of filter scores plus deformation scores
  - Bounding box in training data specifies that score should be high for some placement in a range



w is a model
x is a detection window
z are filter placements

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

concatenation of filters and deformation parameters

concatenation of features and part displacements

Slide from Pedro Felzenswalb

More Specifically...

$$z = (p_1, \ldots, p_n)$$

$$\text{score}(I, p_0) = \max_{p_1,\ldots,p_n} \sum_{i=0}^{n} m_i(I, p_i) - \sum_{i=1}^{n} d_i(p_0, p_i)$$

Filter scores Spring costs

Filter scores $m_i(I, p_i) = \mathbf{w}_i \cdot \phi(I, p_i)$

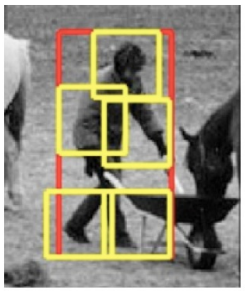
Like in Dalal&Triggs but for multiple parts

Spring costs $d_i(p_0, p_i) = \mathbf{d}_i \cdot (dx^2, dy^2, dx, dy)$

e.g. d_i = [1,1,0,0] yields squared distance error

Slide from Ross Girshick

Scoring with Linear Classifiers



w is a model
x is a detection window
z are filter placements

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

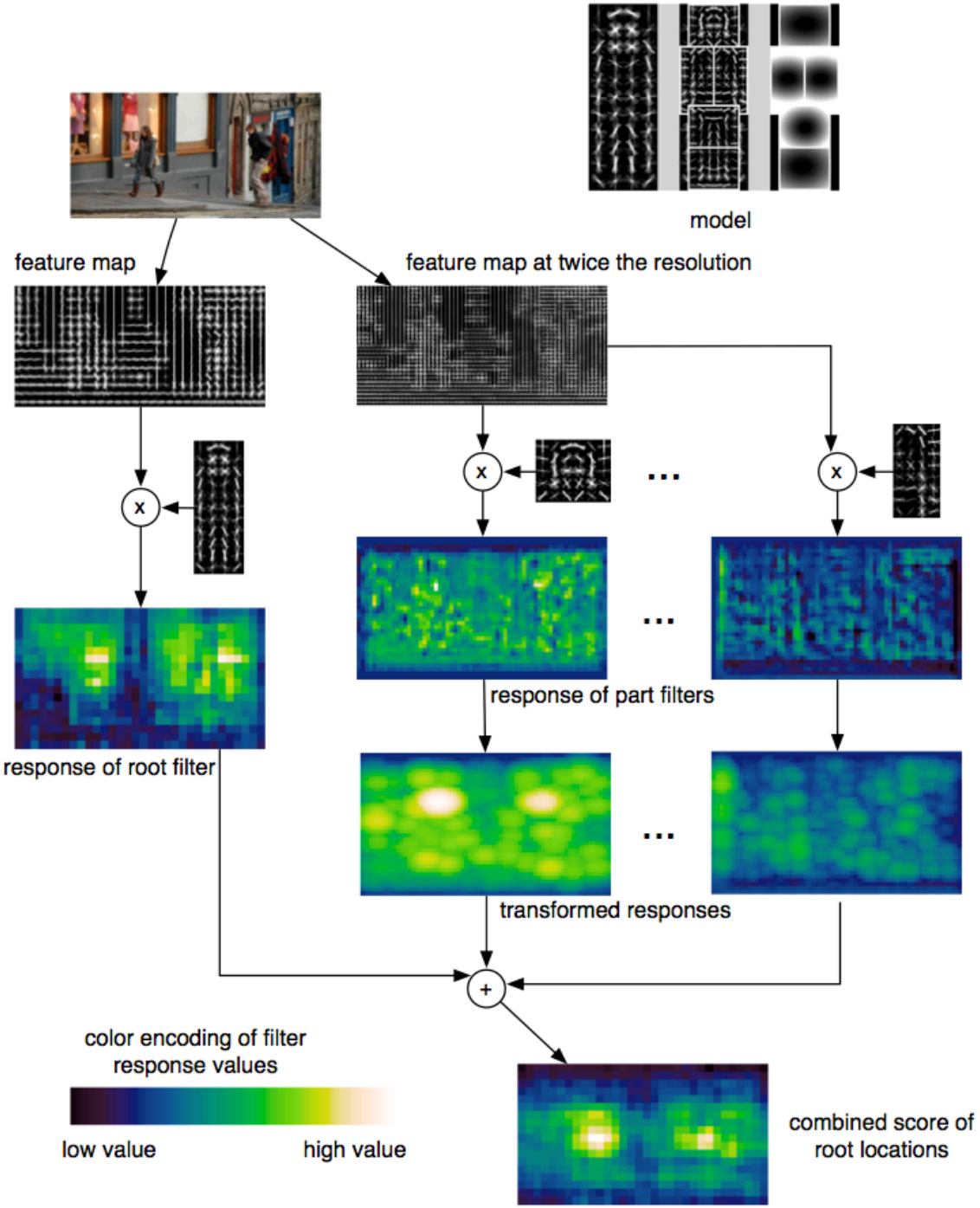
concatenation of filters and deformation parameters

[w0,w1,...,wn,
d1,d2,...,dn]

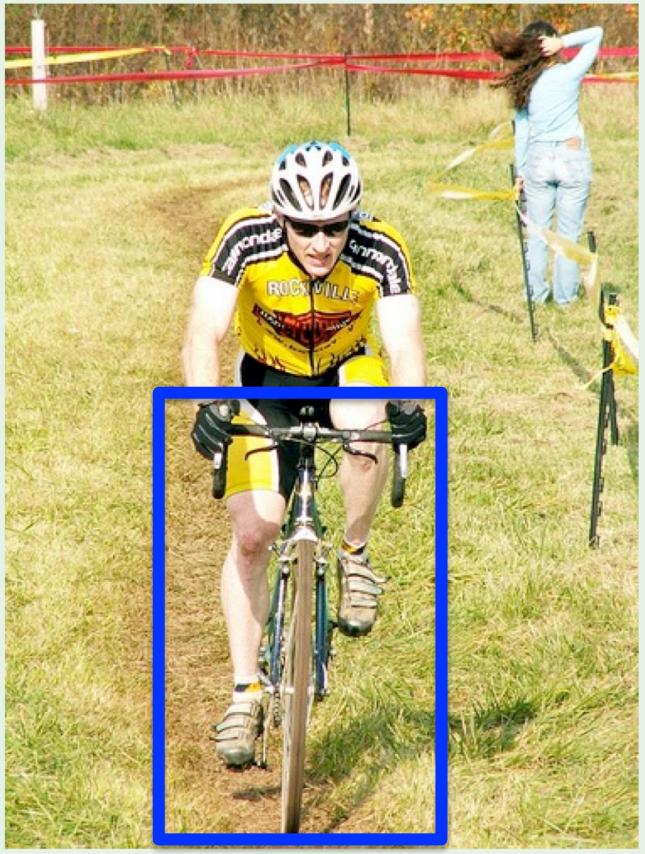
concatenation of features and part displacements

[Φ(I,p0) , ... , Φ(I,pn),
(dx_1^2,dy_1^2,dx_1,dy_1), ..., (dx_n^2,dy_n^2,dx_n,dy_n)]

Detection Overview



Handling Large Variation in Viewpoints

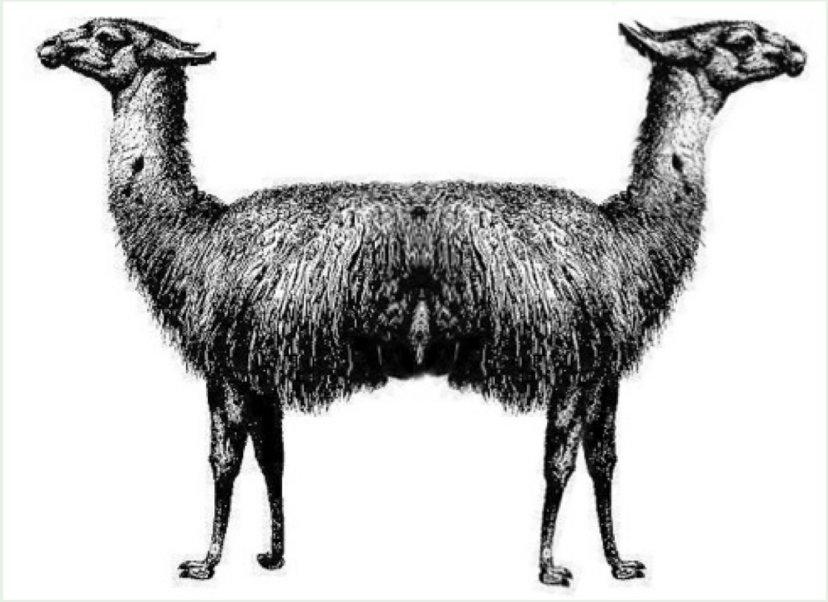


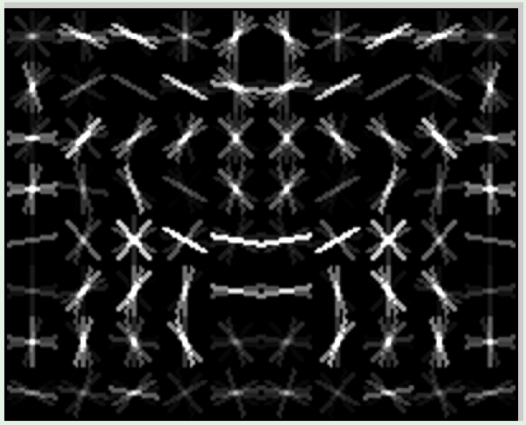
Slide from Ross Girshick

Handling Large Variation in Viewpoints

Good generalization properties on Doctor Dolittle's farm

( + ) / 2 =

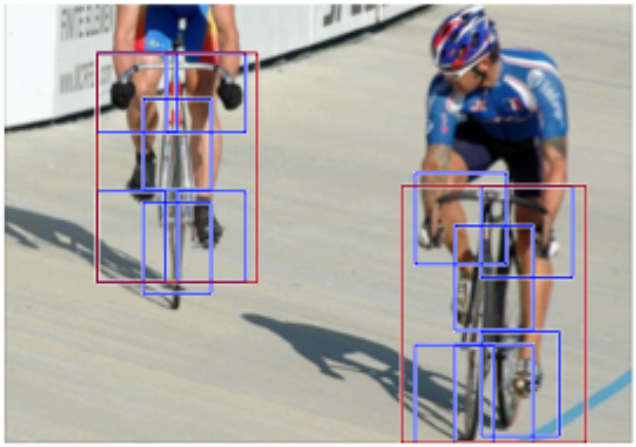


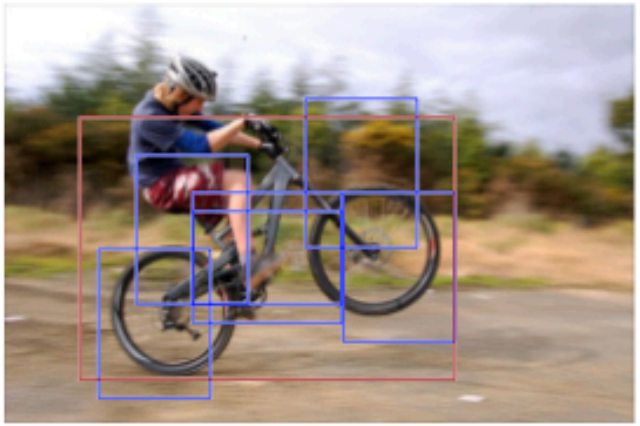


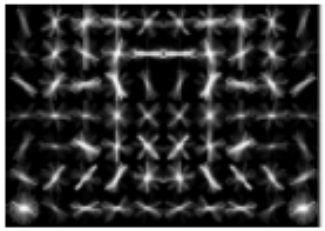
This was supposed to detect horses

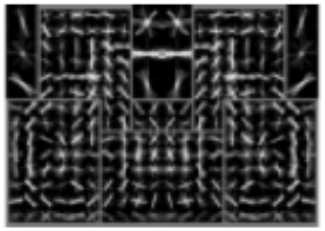
Slide from Ross Girshick

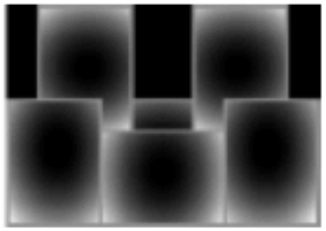
Solution: Use Mixture Model



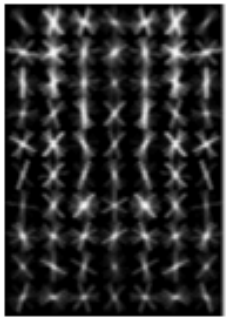


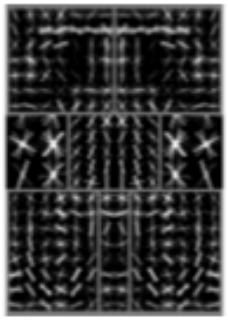


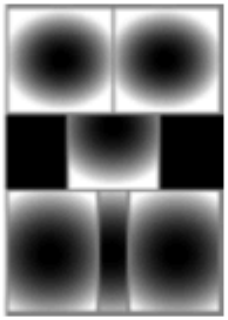




Mixture component 1:
Bicycles viewed from side





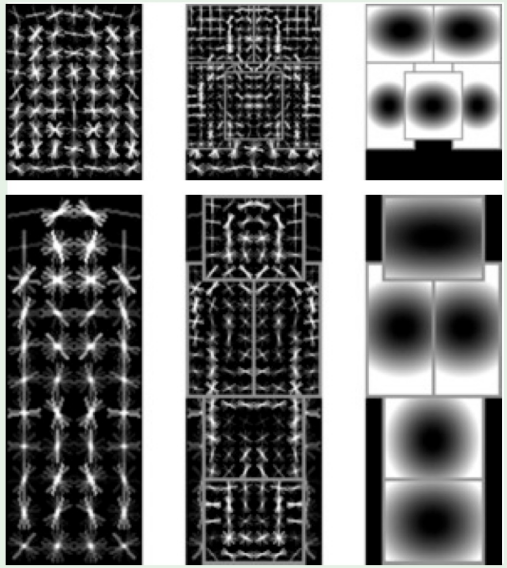


Mixture component 2:
Bicycles viewed from front/rear

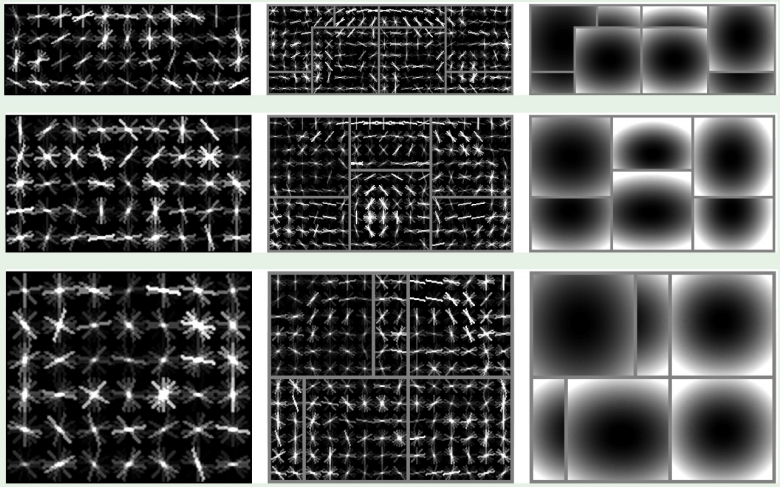
Solution: Use a Mixture Model

Data driven: aspect, occlusion modes, subclasses

Person mixture: 2 components



Car mixture: 3 components



One model for side, one for front and one for 45 degrees.

FMR CVPR '08: AP = 0.27 (person)

FGMR PAMI '10: AP = 0.36 (person)

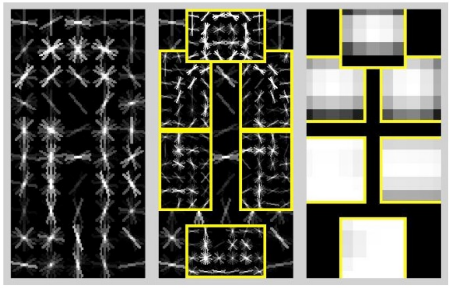
One model for head+torso, one for full body

Slide from Ross Girshick

Training

- Training data consists of images with labeled bounding boxes
- Need to learn the model structure, filters and deformation costs

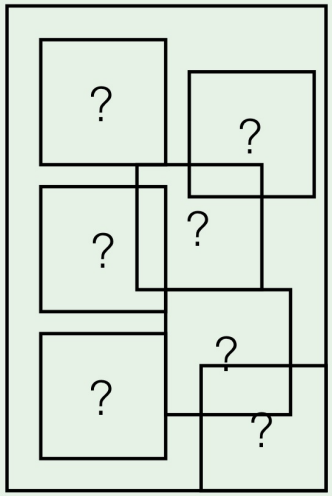
Training



Slide from Pedro Felzenswalb

Part 2: DPM parameter learning

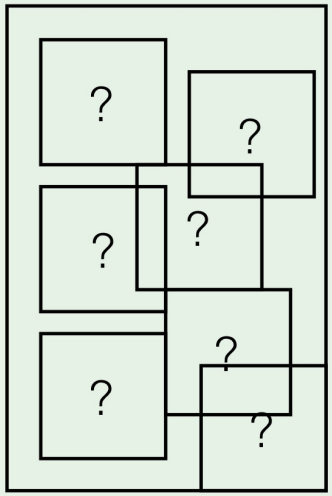
fixed model *structure*



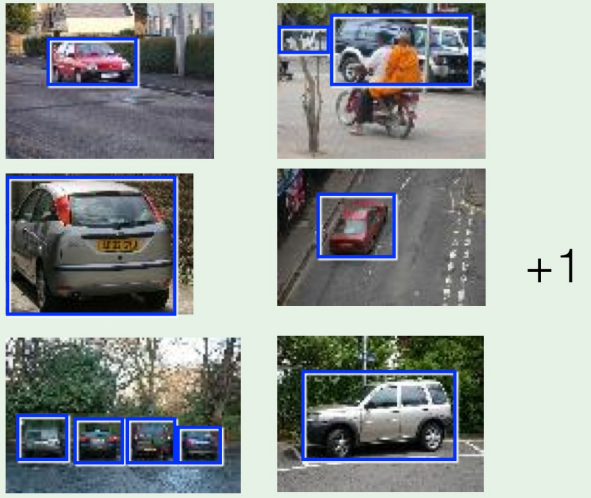
Slide from Ross Girshick

Part 2: DPM parameter learning

fixed model *structure*



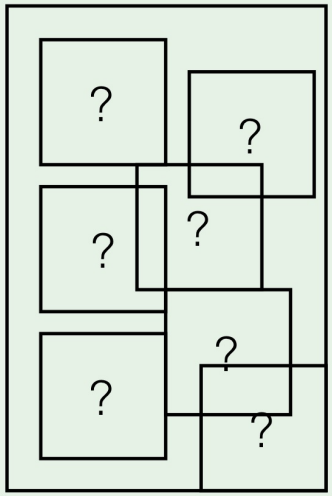
training images *y*



Slide from Ross Girshick

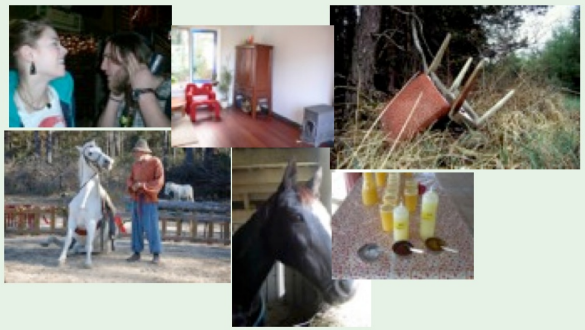
Part 2: DPM parameter learning

fixed model *structure*



training images *y*

+1

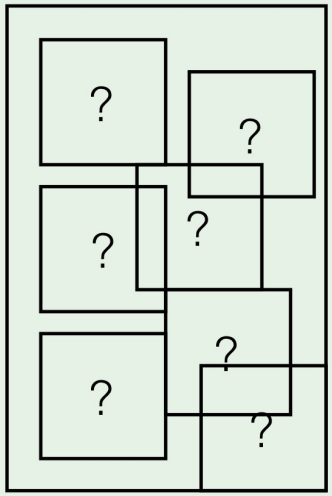


-1

Slide from Ross Girshick

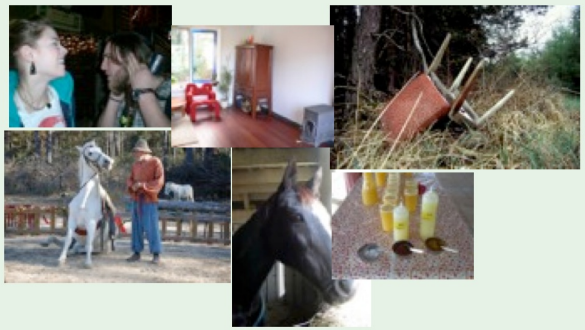
Part 2: DPM parameter learning

fixed model *structure*



training images *y*

+1



-1

Parameters to learn:
- biases (per component)
- deformation costs (per part)
- filter weights

Slide from Ross Girshick

Recall: SVM Training

1. Maximize margin $2/||\mathbf{w}||$
2. Correctly classify all training data points:

$$\mathbf{x}_i \text{ positive } (y_i = 1): \quad \mathbf{x}_i \cdot \mathbf{w} + b \geq 1$$

$$\mathbf{x}_i \text{ negative } (y_i = -1): \quad \mathbf{x}_i \cdot \mathbf{w} + b \leq -1$$

Quadratic optimization problem:

Minimize $\frac{1}{2}\mathbf{w}^T\mathbf{w}$

Subject to $y_i(\mathbf{w}\cdot\mathbf{x}_i+b) \geq 1$

One constraint for each training point.

rewrite

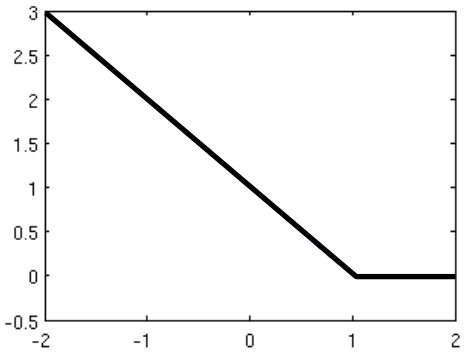
$$w^* = \underset{w}{\operatorname{argmin}}\, \lambda||w||^2 + \sum_{i=1}^{n} \max(0, 1 - y_i f_w(x_i))$$

Aside: SVM and Hinge Loss

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

To maximize the margin, which is 2 / ‖w‖

"hinge" loss



Intuition: the inequality constraints
$fw(x^+) > 1$ and $fw(x^-) < -1$
contribute a linear penalty when they are not satisfied, but are not penalized when the inequalities hold.

This is a consequence of encoding the constraints into the objective function using Lagrange multipliers. For constraints that are are satisfied (the appropriate inequality holds), the value of the corresponding Lagrange multiplier becomes 0.

SVM vs LSVM Training

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

SVM → $f_w(x_i) = w \cdot \Phi(x_i)$

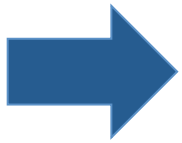
LSVM → $f_w(x_i) = \max_{z \text{ in } Z(x_i)} w \cdot \Phi(x_i, z)$

Max over latent positions for part x_i

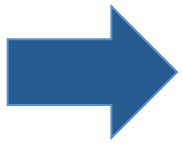
This introduces some difficulties during training

SVM vs LSVM Training

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

SVM  $f_w(x_i) = w \cdot \Phi(x_i)$

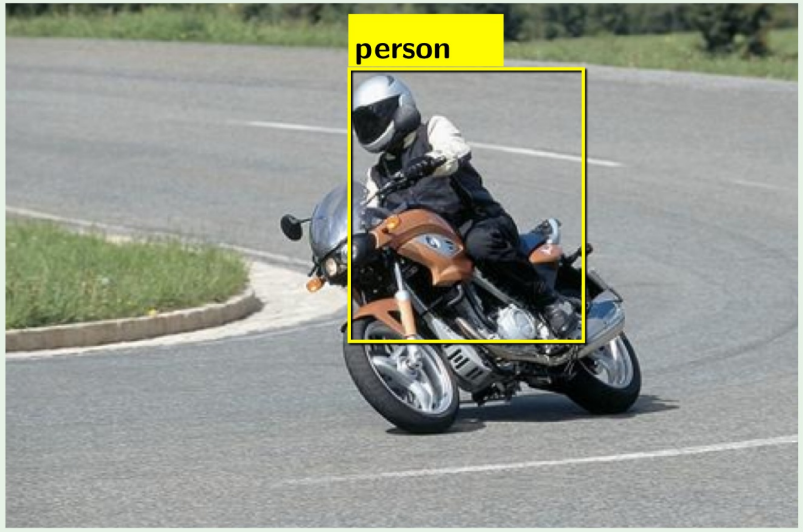
Convex problem, guaranteed optimal solution!

LSVM  $f_w(x_i) = \max_{z \text{ in } Z(x_i)} w \cdot \Phi(x_i, z)$

Only semi-convex, no guaranteed optimal solution

Positive examples ($y = +1$)

x specifies an image and bounding box



We want

$$f_{\mathbf{w}}(x) = \max_{z \in Z(x)} \mathbf{w} \cdot \Phi(x, z)$$

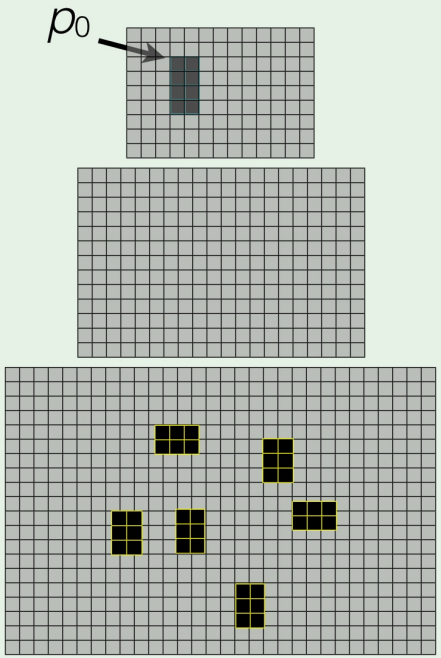
to score >= +1

$Z(x)$ includes all z with more than 70% overlap with ground truth

Slide from Ross Girshick

Negative examples ($y = -1$)

x specifies an image and a HOG pyramid location p_0



We want

$$f_{\mathbf{w}}(x) = \max_{z \in Z(x)} \mathbf{w} \cdot \Phi(x, z)$$

to score <= -1

$Z(x)$ restricts the root to p_0 and allows *any* placement of the other filters

Slide from Ross Girshick

How we learn parameters: latent SVM

$$E(\mathbf{w}) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_i \max\{0, 1 - y_i f_{\mathbf{w}}(x_i)\}$$

Slide from Ross Girshick

How we learn parameters: latent SVM

$$E(\mathbf{w}) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_i \max\{0, 1 - y_i f_{\mathbf{w}}(x_i)\}$$

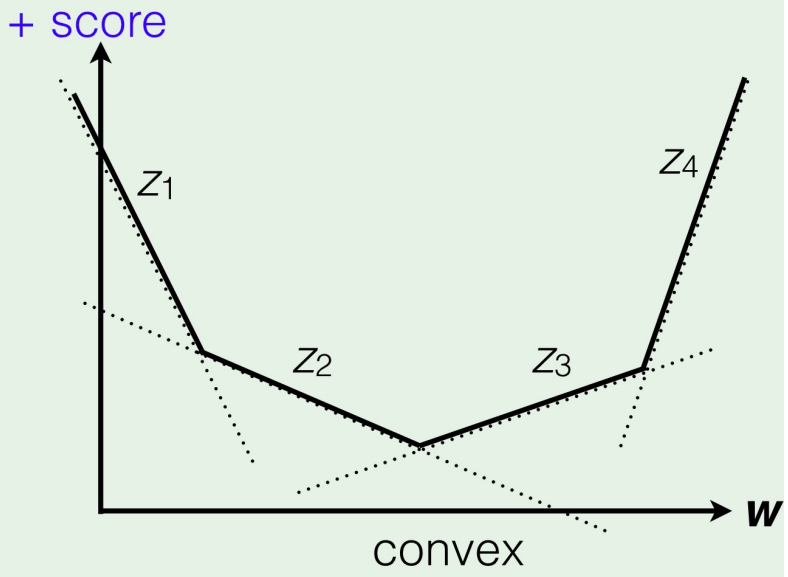
$$E(\mathbf{w}) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i\in P} \max\{0, 1 - \max_{z\in Z(x)} \mathbf{w}\cdot\Phi(x_i, z)\} + C\sum_{i\in N} \max\{0, 1 + \max_{z\in Z(x)} \mathbf{w}\cdot\Phi(x_i, z)\}$$

Slide from Ross Girshick

How we learn parameters: latent SVM

$$E(\mathbf{w}) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_i \max\{0, 1 - y_i f_{\mathbf{w}}(x_i)\}$$

$$E(\mathbf{w}) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i\in P} \max\{0, 1 - \max_{z\in Z(x)} \mathbf{w}\cdot\Phi(x_i, z)\}$$
$$+ C\sum_{i\in N} \max\{0, 1 + \max_{z\in Z(x)} \mathbf{w}\cdot\Phi(x_i, z)\}$$

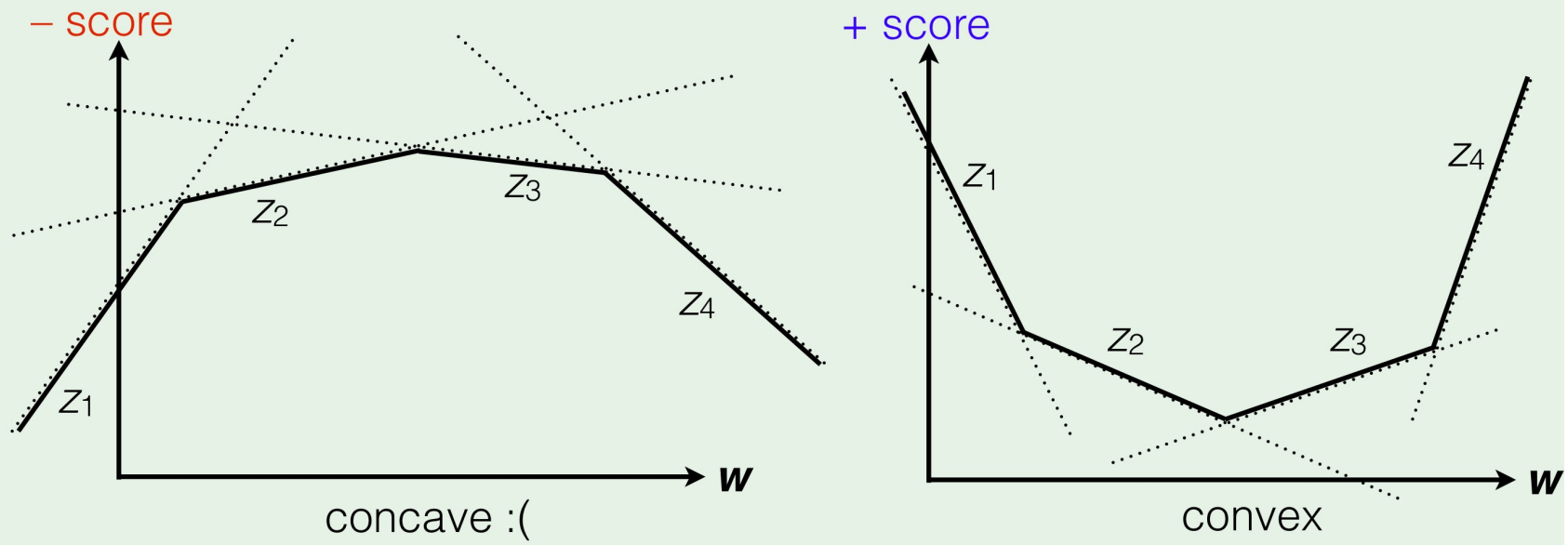


Slide from Ross Girshick

How we learn parameters: latent SVM

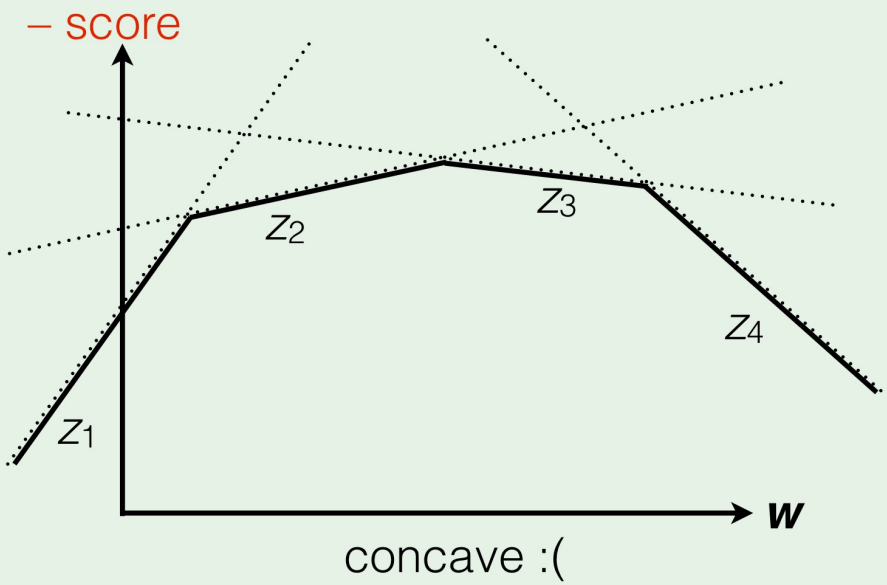
$$E(\mathbf{w}) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_i \max\{0, 1 - y_i f_{\mathbf{w}}(x_i)\}$$

$$E(\mathbf{w}) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i\in P} \max\{0, 1 - \max_{z\in Z(x)} \mathbf{w}\cdot\Phi(x_i, z)\} + C\sum_{i\in N} \max\{0, 1 + \max_{z\in Z(x)} \mathbf{w}\cdot\Phi(x_i, z)\}$$

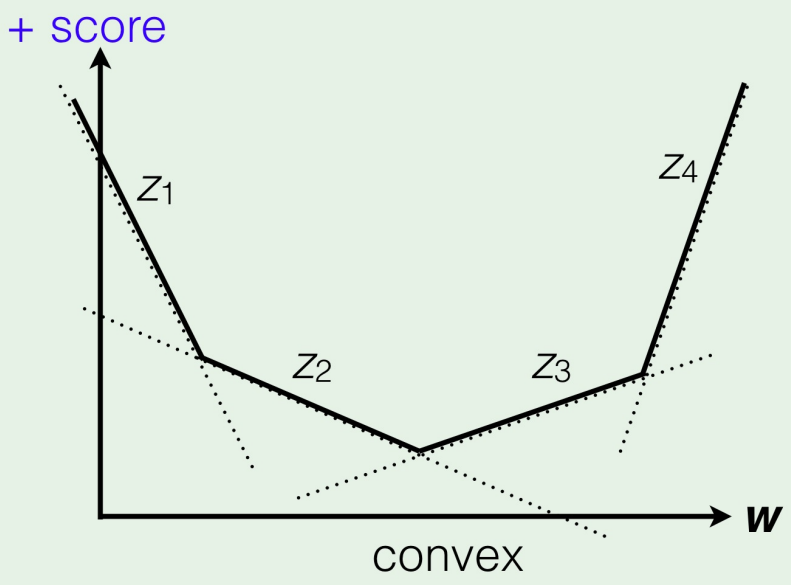


Slide from Ross Girshick

Observations



concave :(



convex

Latent SVM objective is convex in the negatives

but not in the positives

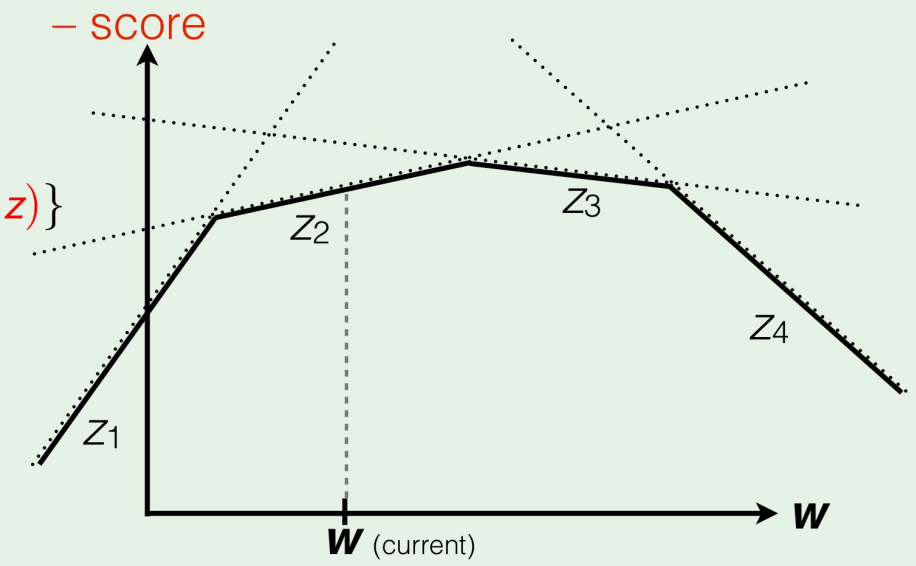
>> “semi-convex”

Since whole objective function is no longer convex, there is no longer a single global optimum, so no easy solution method.

Slide from Ross Girshick

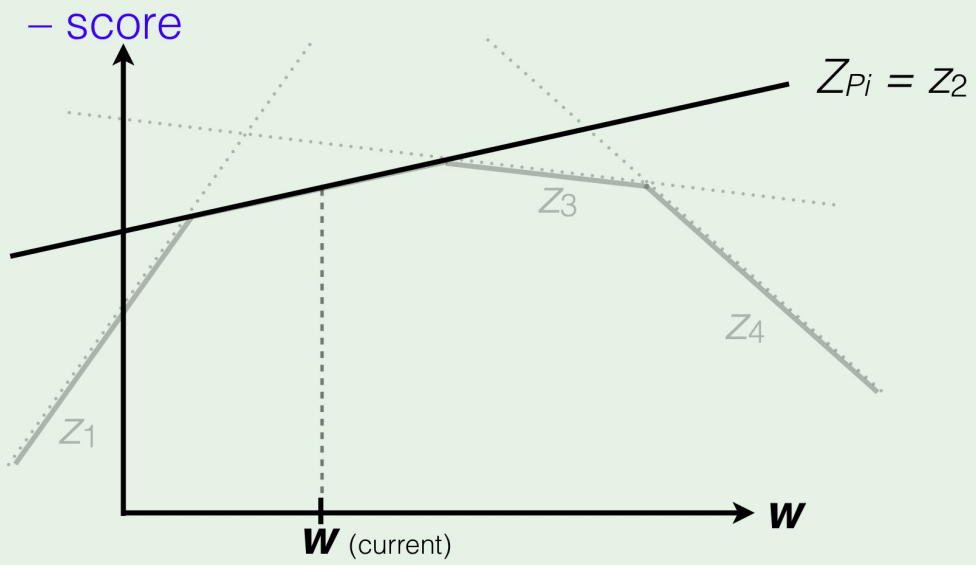
Convex upper bound on loss

$$\max\{0, 1 - \max_{z \in Z(x)} \mathbf{w} \cdot \Phi(x_i, z)\}$$



$$\max\{0, 1 - \mathbf{w} \cdot \Phi(x_i, Z_{Pi})\}$$

convex



Approximation by upper bound

Slide from Ross Girshick

Auxiliary objective

Let $Z_P = \{Z_{P1}, Z_{P2}, \ldots\}$

Current estimates of latent configurations that yield upper bound for each positive example.

$$E(\mathbf{w}, Z_P) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i \in P} \max\{0, 1 - \mathbf{w} \cdot \Phi(x_i, Z_{Pi})\}$$
$$+ C\sum_{i \in N} \max\{0, 1 + \max_{z \in Z(x)} \mathbf{w} \cdot \Phi(x_i, z)\}$$

Slide from Ross Girshick

Auxiliary objective

Let $Z_P = \{Z_{P1}, Z_{P2}, \ldots\}$

Current estimates of latent configurations that yield upper bound for each positive example.

$$E(\mathbf{w}, Z_P) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i \in P} \max\{0, 1 - \mathbf{w} \cdot \Phi(x_i, Z_{Pi})\}$$
$$+ C\sum_{i \in N} \max\{0, 1 + \max_{z \in Z(x)} \mathbf{w} \cdot \Phi(x_i, z)\}$$

Yields upper bound for whole objective function.

Note that $E(\mathbf{w}, Z_p) \geq \min_{Z_P} E(\mathbf{w}, Z_P) = E(\mathbf{w})$

Slide from Ross Girshick

Auxiliary objective

Current estimates of latent configurations that yield upper bound for each positive example.

Let $Z_P = \{Z_{P1}, Z_{P2}, \ldots\}$

$$E(\mathbf{w}, Z_P) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i \in P} \max\{0, 1 - \mathbf{w} \cdot \Phi(x_i, Z_{Pi})\} + C\sum_{i \in N} \max\{0, 1 + \max_{z \in Z(x)} \mathbf{w} \cdot \Phi(x_i, z)\}$$

Yields upper bound for whole objective function.

Note that $E(\mathbf{w}, Z_p) \geq \min_{Z_P} E(\mathbf{w}, Z_P) = E(\mathbf{w})$

And it is a tight upper bound

and $\mathbf{w}^* = \min_{\mathbf{w}, Z_P} E(\mathbf{w}, Z_P) = \min_{\mathbf{w}} E(\mathbf{w})$

Slide from Ross Girshick

Auxiliary objective

$$\mathbf{w}^* = \min_{\mathbf{w}, Z_P} E(\mathbf{w}, Z_P) = \min_{\mathbf{w}} E(\mathbf{w})$$

Unfortunately, this isn't easy to optimize either, not to mention it is based on estimates of upper bound latent configurations, which could be wrong.

Slide from Ross Girshick

Auxiliary objective

$$\mathbf{w}^* = \min_{\mathbf{w}, Z_P} E(\mathbf{w}, Z_P) = \min_{\mathbf{w}} E(\mathbf{w})$$

Iterative approach to solving this.
NOT guaranteed to find optimal solution
unless you start with good initial estimates.

Find stationary point by coordinate descent on $E(\mathbf{w}, Z_P)$

Slide from Ross Girshick

Auxiliary objective

$$\mathbf{w}^* = \min_{\mathbf{w}, Z_P} E(\mathbf{w}, Z_P) = \min_{\mathbf{w}} E(\mathbf{w})$$

Iterative approach to solving this.
NOT guaranteed to find optimal solution
unless you start with good initial estimates.

Find stationary point by coordinate descent on $E(\mathbf{w}, Z_P)$

Initialization: either by picking a $\boldsymbol{w}_{(0)}$ (or Z_P)

Slide from Ross Girshick

Auxiliary objective

$$\mathbf{w}^* = \min_{\mathbf{w}, Z_P} E(\mathbf{w}, Z_P) = \min_{\mathbf{w}} E(\mathbf{w})$$

Iterative approach to solving this.
NOT guaranteed to find optimal solution
unless you start with good initial estimates.

Find stationary point by coordinate descent on $E(\mathbf{w}, Z_P)$

Initialization: either by picking a $\boldsymbol{w}_{(0)}$ (or Z_P)

Step 1:

$$Z_{Pi} = \underset{z \in Z(x_i)}{\operatorname{argmax}} \mathbf{w}_{(t)} \cdot \Phi(x_i, z) \quad \forall i \in P$$

Slide from Ross Girshick

Auxiliary objective

$$\mathbf{w}^* = \min_{\mathbf{w}, Z_P} E(\mathbf{w}, Z_P) = \min_{\mathbf{w}} E(\mathbf{w})$$

Iterative approach to solving this.
NOT guaranteed to find optimal solution
unless you start with good initial estimates.

Find stationary point by coordinate descent on $E(\mathbf{w}, Z_P)$

Initialization: either by picking a $\boldsymbol{w}_{(0)}$ (or Z_P)

Step 1:

$$Z_{Pi} = \underset{z \in Z(x_i)}{\operatorname{argmax}} \mathbf{w}_{(t)} \cdot \Phi(x_i, z) \quad \forall i \in P$$

Step 2:

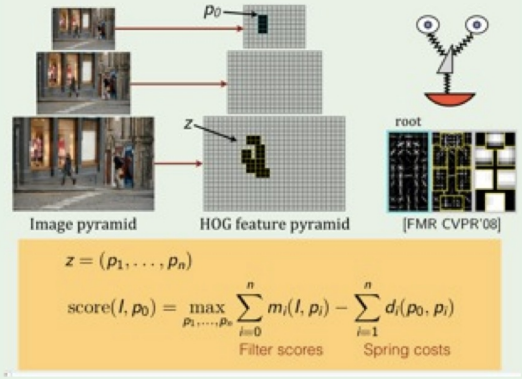
$$\mathbf{w}_{(t+1)} = \underset{\mathbf{w}}{\operatorname{argmin}} E(\mathbf{w}, Z_P)$$

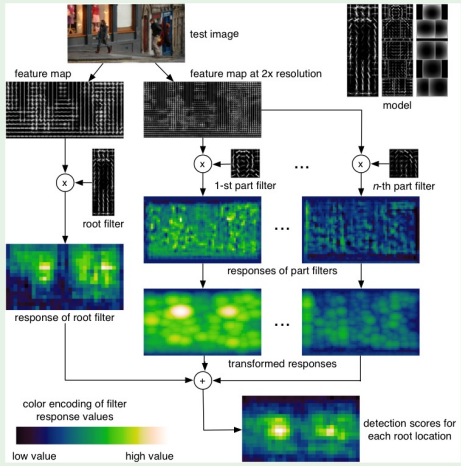
Slide from Ross Girshick

Step 1

$$Z_{Pi} = \underset{z \in Z(x_i)}{\operatorname{argmax}} \mathbf{w}_{(t)} \cdot \Phi(x_i, z) \quad \forall i \in P$$

This is just detection: Easy to compute





Slide from Ross Girshick

Step 2

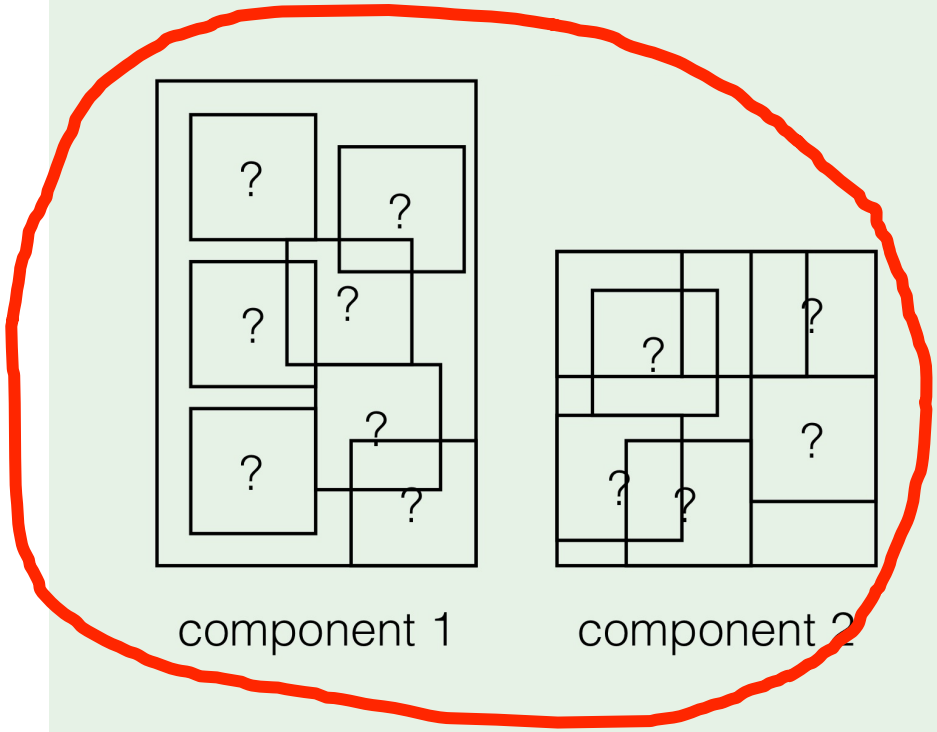
$$\min_{\mathbf{w}} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i\in P} \max\{0, 1-\mathbf{w}\cdot\Phi(x_i, Z_{Pi})\}$$
$$+ C\sum_{i\in N} \max\{0, 1+\max_{z\in Z(x)} \mathbf{w}\cdot\Phi(x_i, z)\}$$

Convex | Easy to solve

Slide from Ross Girshick

What about the model structure?

Can we learn that as well?



training images *y*

Model structure
- # components
- # parts per component
- root and part filter shapes
- part anchor locations

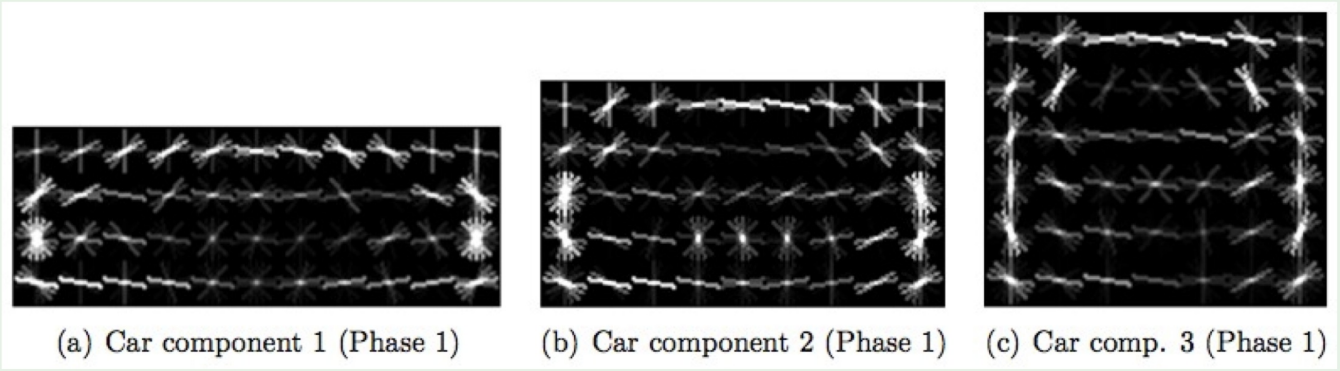
Slide from Ross Girshick

Learning model structure

Initialize a mixture model to handle different viewpoints

Initialization:

Split positives by aspect ratio



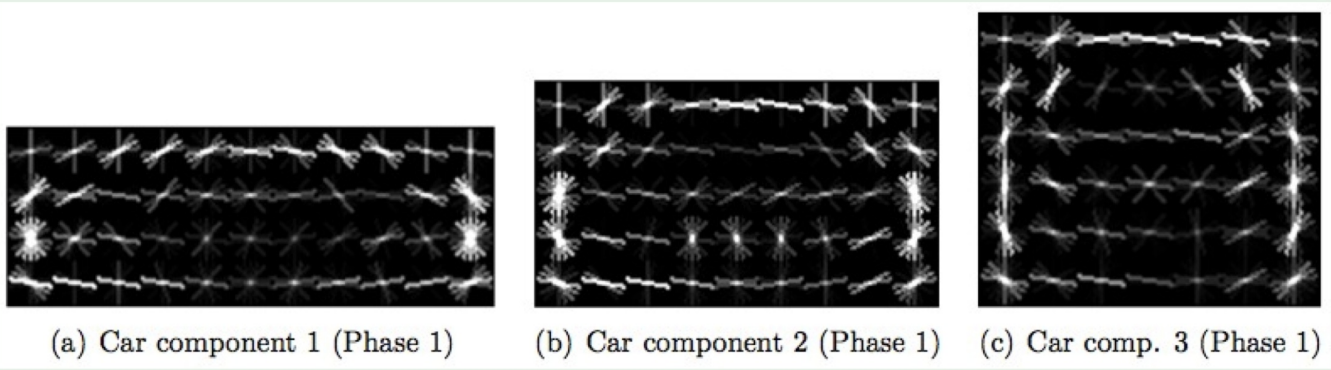
(a) Car component 1 (Phase 1) (b) Car component 2 (Phase 1) (c) Car comp. 3 (Phase 1)

Warp to common size

Train Dalal & Triggs model for each aspect ratio on its own

Slide from Ross Girshick

Learning model structure

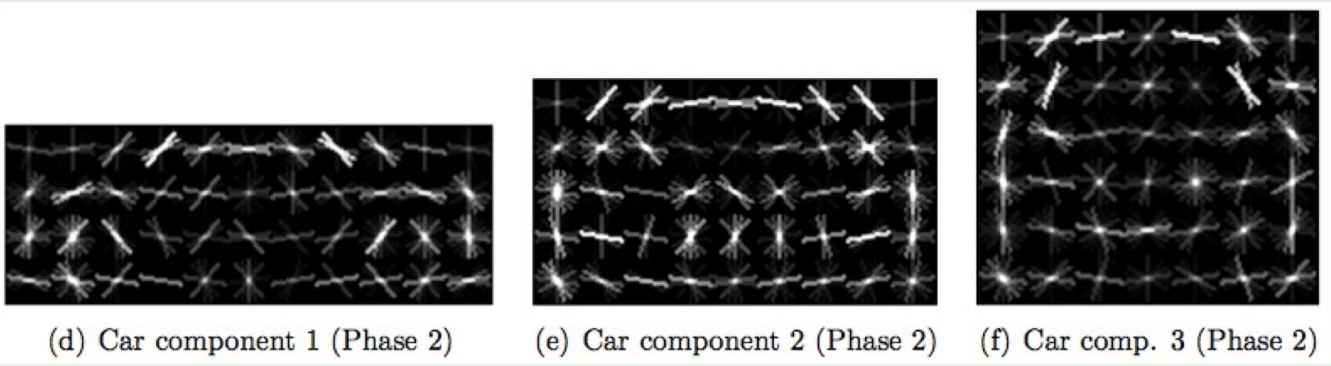


(a) Car component 1 (Phase 1) (b) Car component 2 (Phase 1) (c) Car comp. 3 (Phase 1)

Use D&T filters as initial **w** for LSVM training

Merge components

Root filter placement and component choice are latent



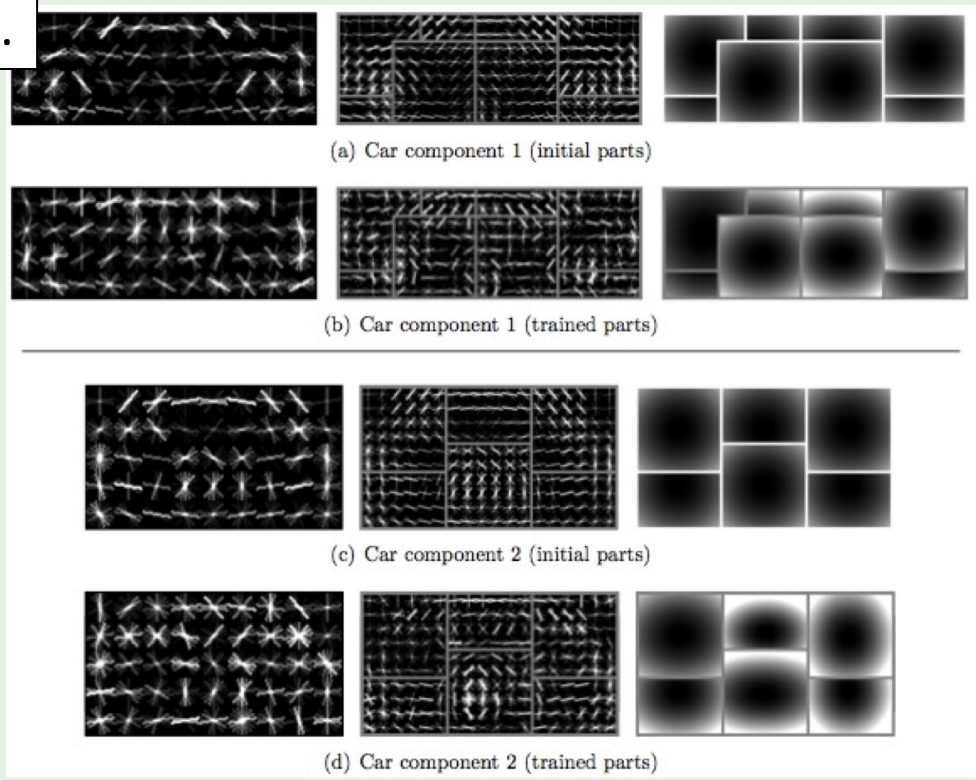
(d) Car component 1 (Phase 2) (e) Car component 2 (Phase 2) (f) Car comp. 3 (Phase 2)

Slide from Ross Girshick

Initialize parts for each component

Refine by training.

Learning model structure



(a) Car component 1 (initial parts)

(b) Car component 1 (trained parts)

(c) Car component 2 (initial parts)

(d) Car component 2 (trained parts)

Add parts to cover high-energy areas of root filters

Note: parts are constrained to be symmetrically placed

Continue training model with LSVM

Slide from Ross Girshick

Summary: Model Learning

- Learn mixture model (component root filters)
- Learn part model for each mixture component
- LSVM used during both phases

Mining Hard Negatives

Typical dataset

300 – 8,000 positive examples

500 million to 1 billion negative examples
(not including latent configurations!)

Detections problems are highly unbalanced (many more negatives than positives).
We REALLY don't want to consider all negative examples while training!

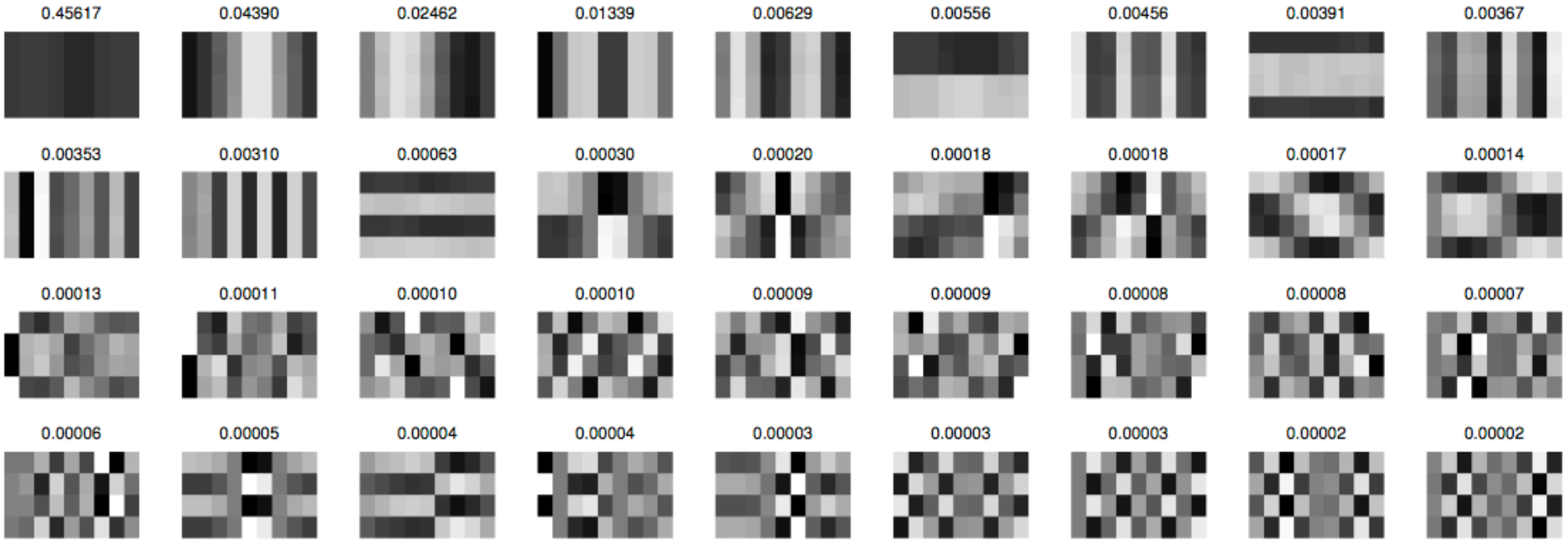
Slide from Ross Girshick

Mining "Hard Negative" Examples

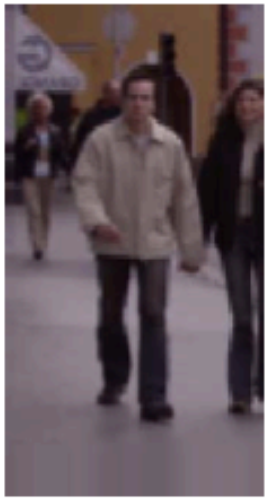
- Set of negative training examples is HUGE
  (a single image can yield 10^5 examples for a scanning window classifier).
- Construct training data from the positive instances and a selection of the "hard negative" instances, via bootstrapping
  - Train using subset of negative examples
  - Apply resulting classifier to all negative examples, and take those incorrectly classified as a new set of hard negatives
  - Retrain, and perhaps repeat

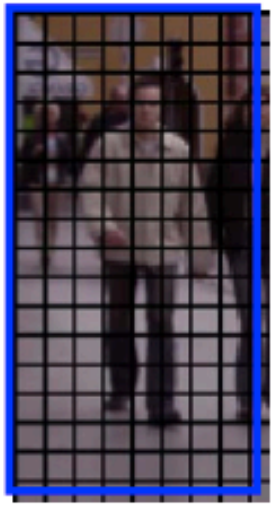
PCA Analysis of HOG

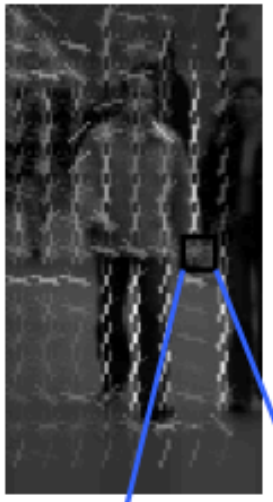
- Collected 36-dimensional HOG features at a variety of scales over large number of images
- PCA analysis to identify top eigenvectors



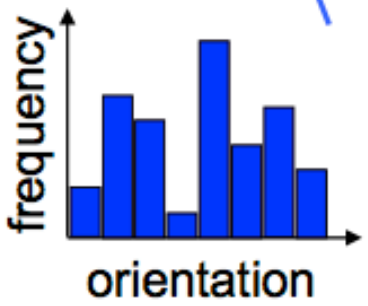
Recall: HOG Features





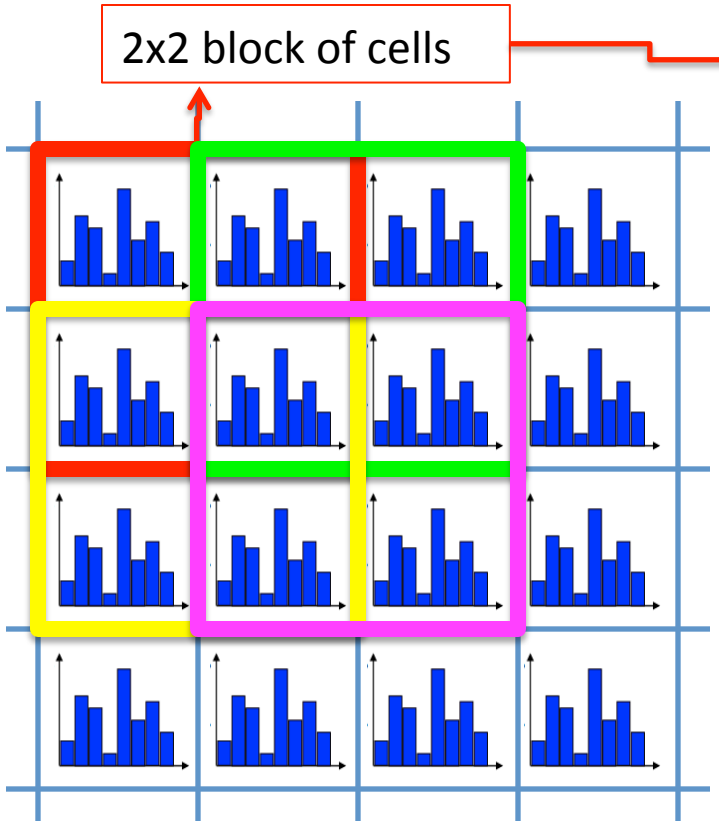


- tile window into 8 x 8 pixel cells
- each cell represented by HOG



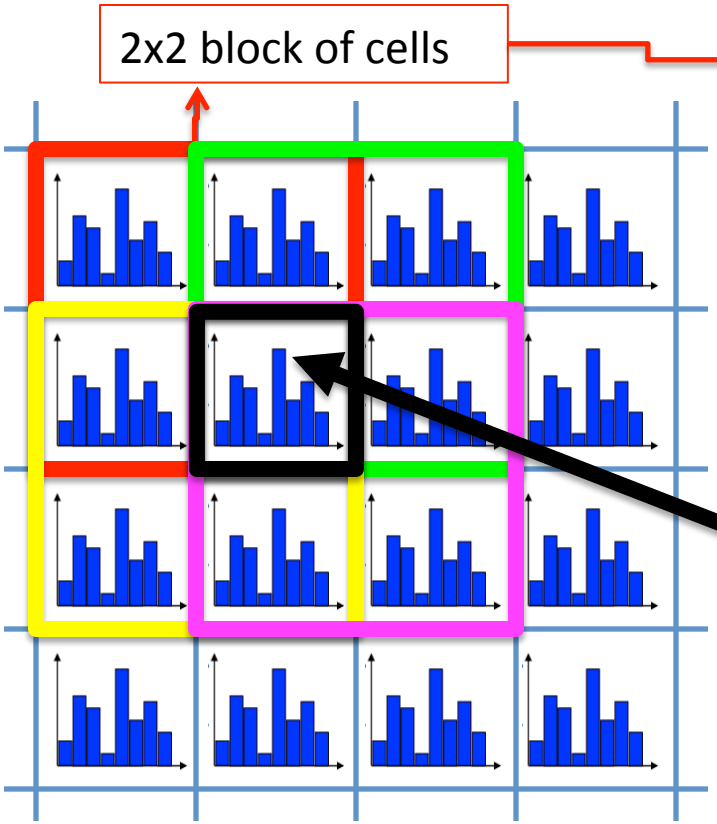
Each cell contains a histogram of gradient orientations, weighted by gradient magnitude

HoG Feature Extraction: Blocks



Independent contrast normalization over 2x2 blocks of cells.

HoG Feature Extraction: Blocks



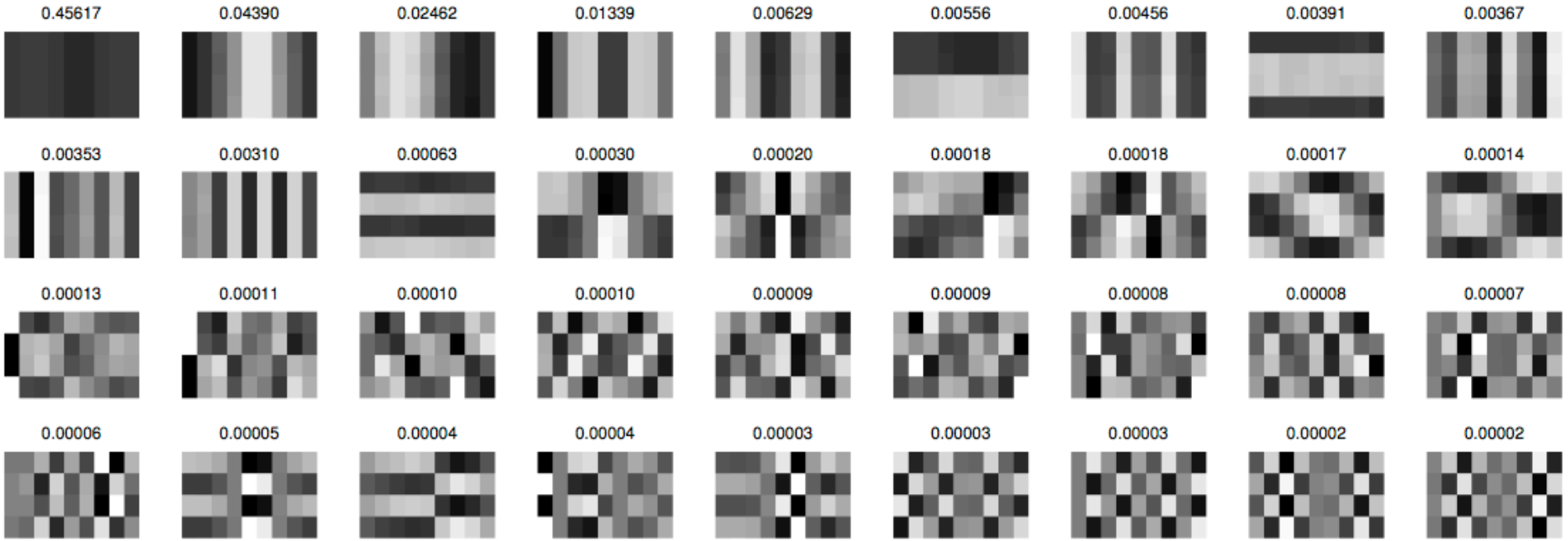
Independent contrast normalization over 2x2 blocks of cells.

Each histogram gets normalized 4 different times.

9 orientations x 4 orientations
= 36 feature values for each cell

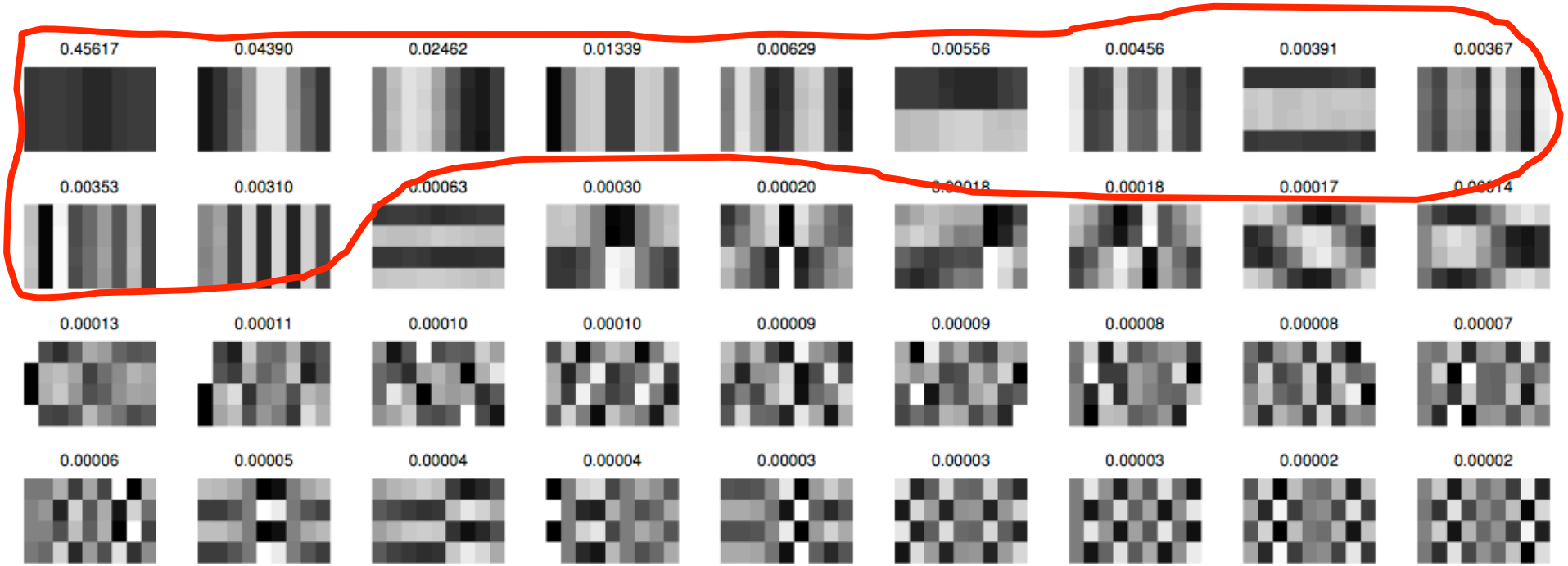
PCA Analysis of HOG

- Collected 36-dimensional HOG features at a variety of scales over large number of images
- PCA analysis to identify top eigenvectors



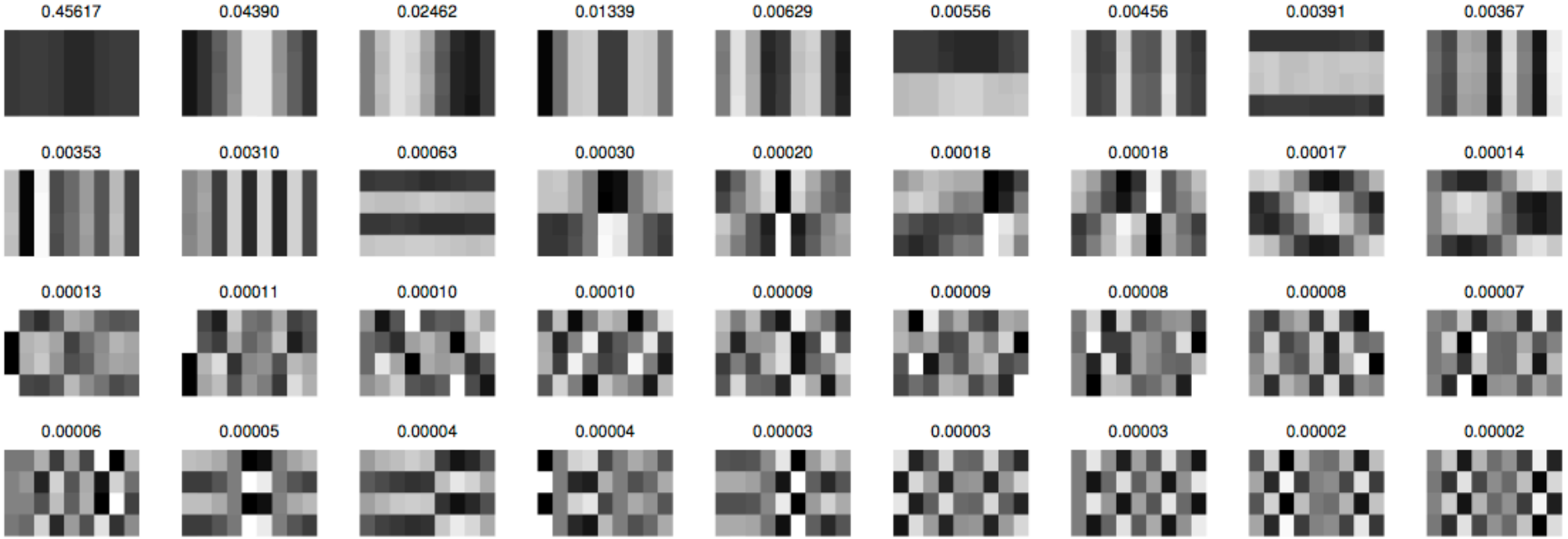
PCA Analysis of HOG

- Collected 36-dimensional HOG features at a variety of scales over large number of images
- PCA analysis to identify top eigenvectors
- Top 11 account for nearly all of the variation



PCA Analysis of HOG

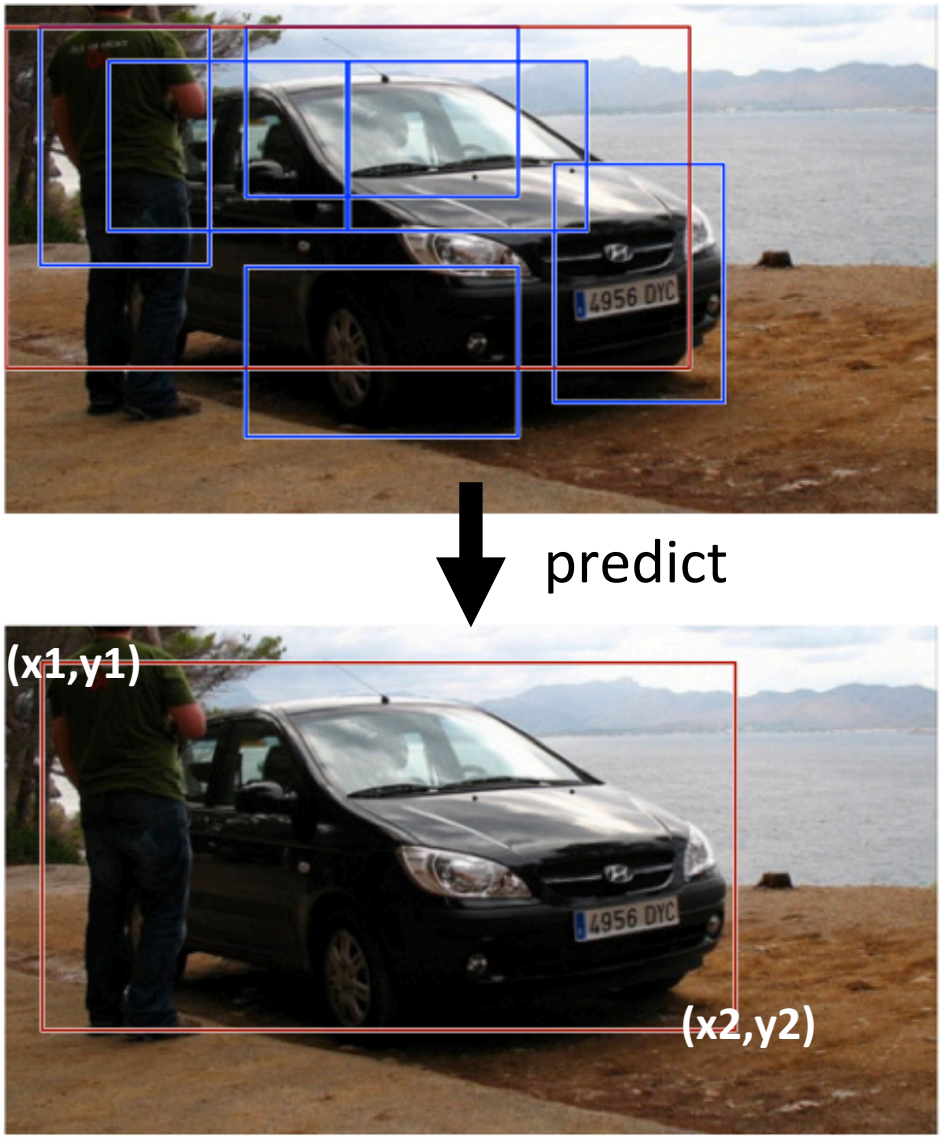
- Projection onto PCA bases is expensive
- However, note that top eigenvectors are constant over rows or cols
- Analytic projection to 13 dimensions: sum over each of 4 rows and 9 columns



Bounding Box Prediction

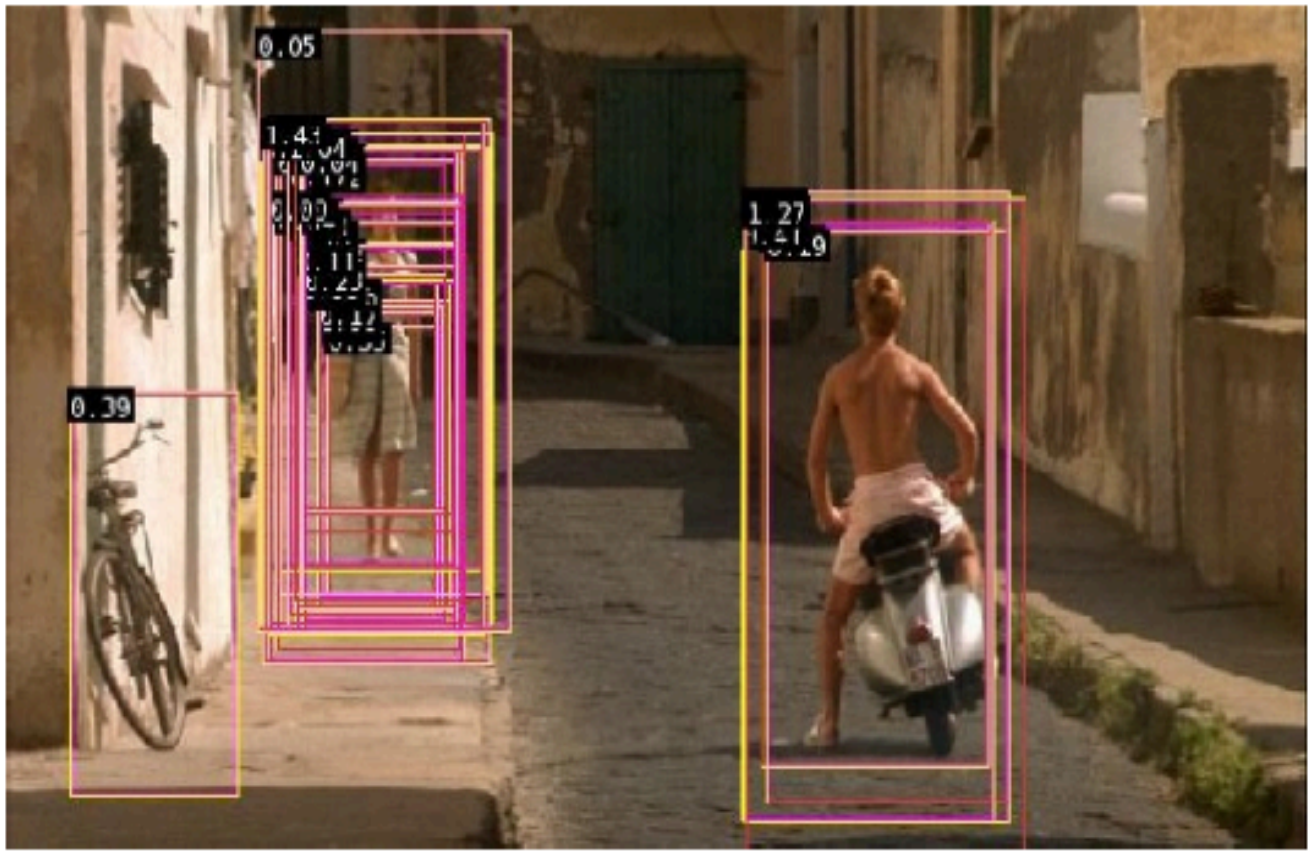
Problem: Location and size of root filter (red rectangle) may not correlate perfectly with bounding boxes in human-labeled data

Solution: Learn four linear regression functions to predict x1, y1, x2, y2 of bounding box from a vector containing locations of each part (including root) and the width (scale) of the root. (2n+3)→x1 (or y1, x2, y2)



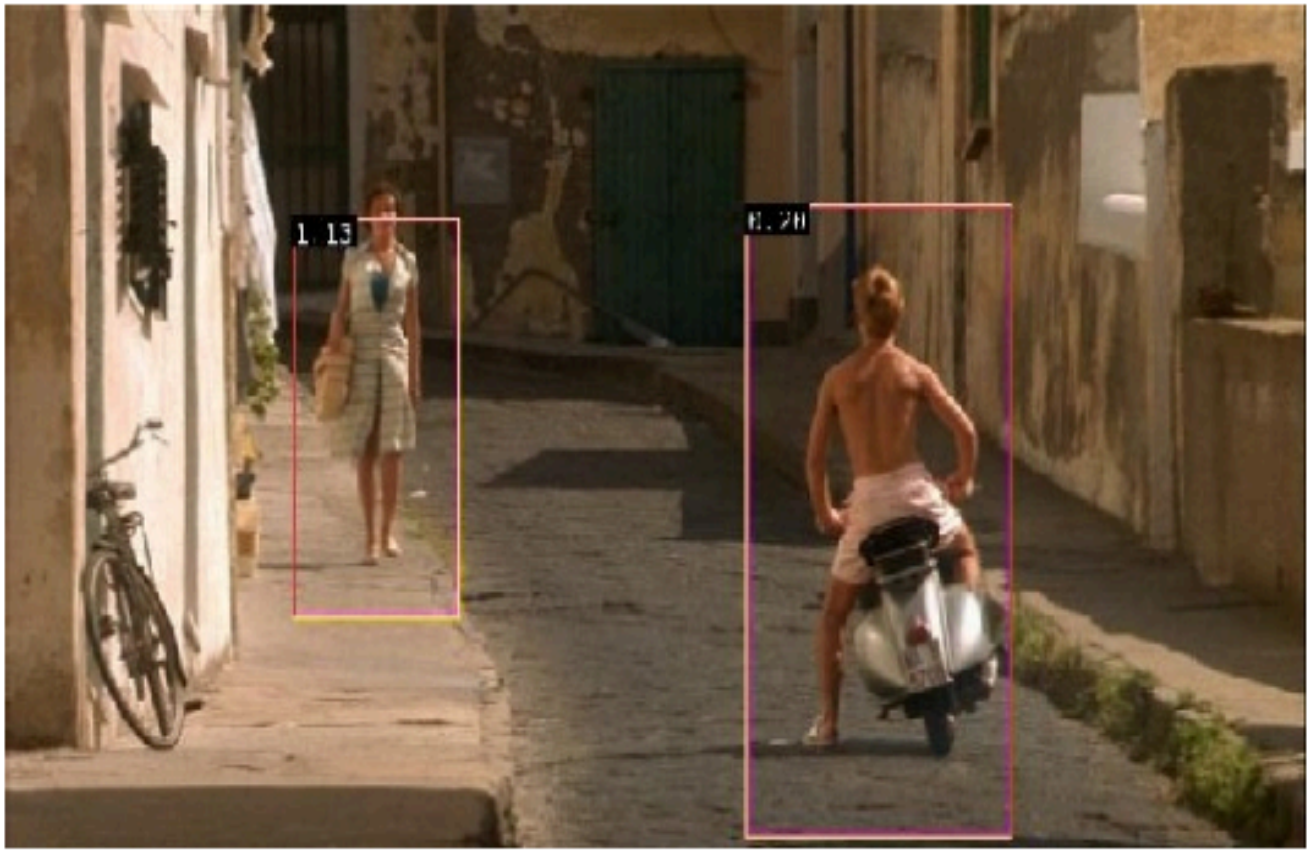
Non-Maximum Suppression

- Sliding window detectors tend to produce many overlapping detections on same object



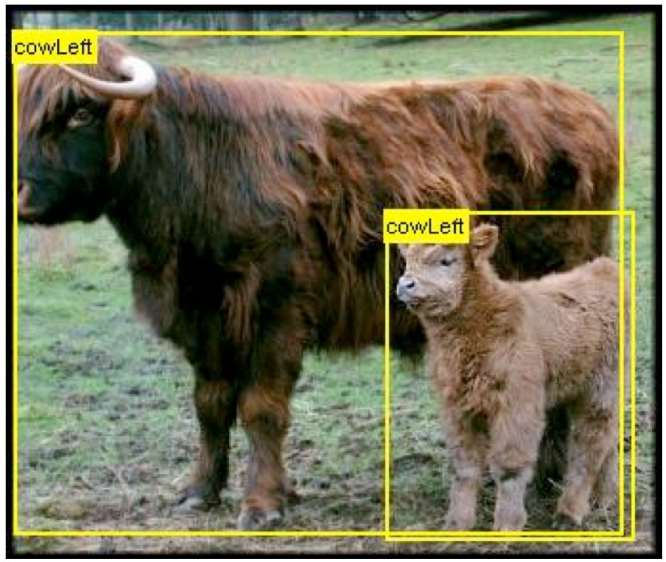
Non-Maximum Suppression

- Greedy solution: run through list of detections in descending order of confidence, removing ones that overlap more than 50% with a higher-scoring detection.



Part III: PASCAL Challenge

- ~10,000 images, with ~25,000 target objects
  - Objects from 20 categories (person, car, bicycle, cow, table...)
  - Objects are annotated with labeled bounding boxes



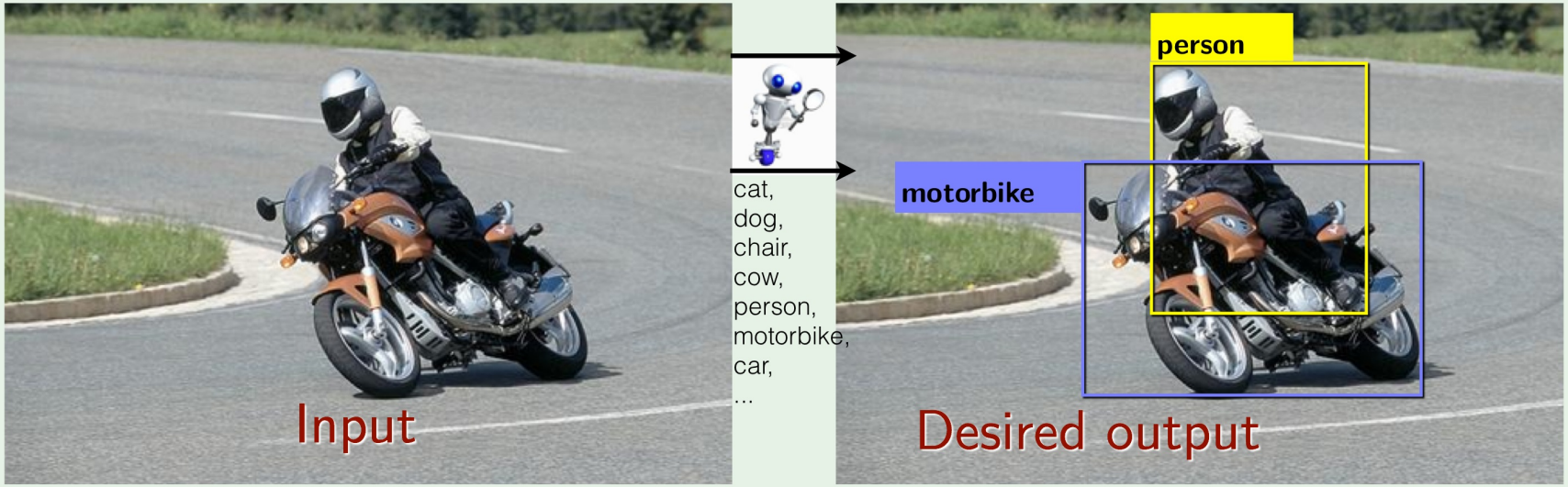
Slide from Pedro Felzenswalb

Example Annotations from the Dataset

Slide from Pedro Felzenswalb

Formalizing the object detection task

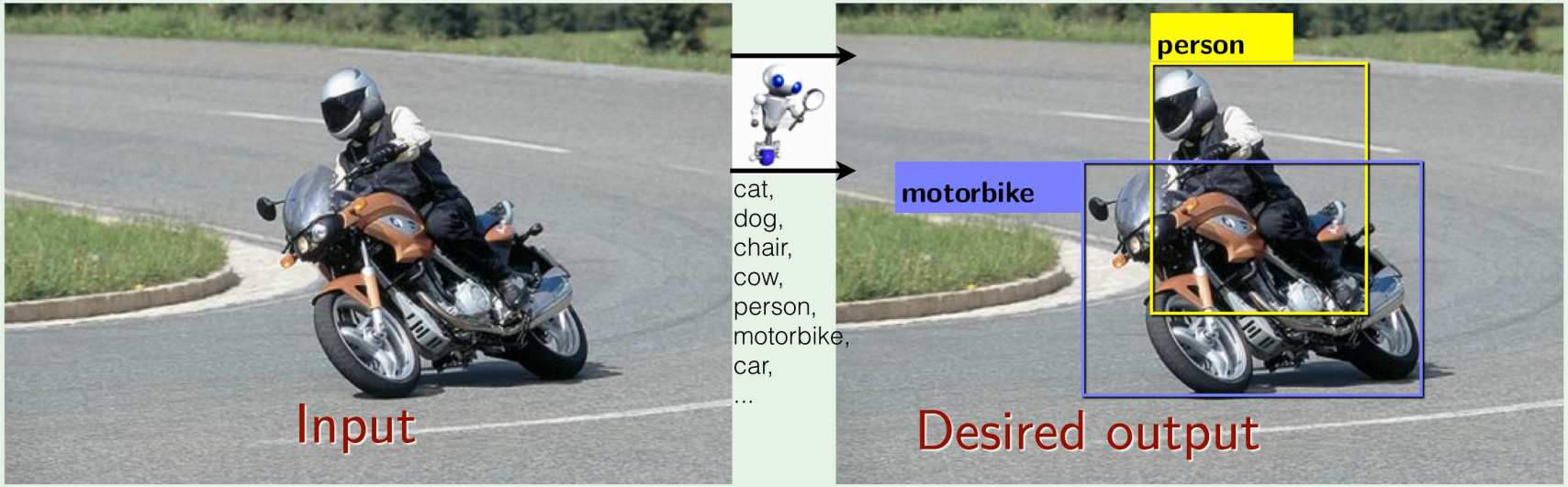
Many possible ways, this one is popular:

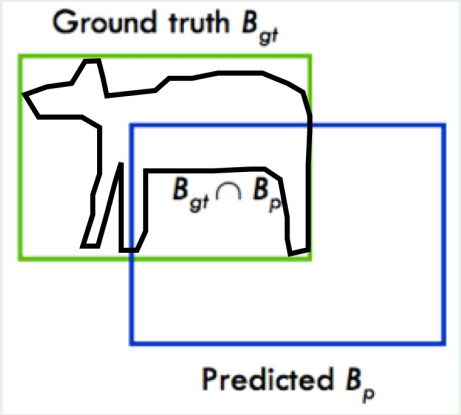


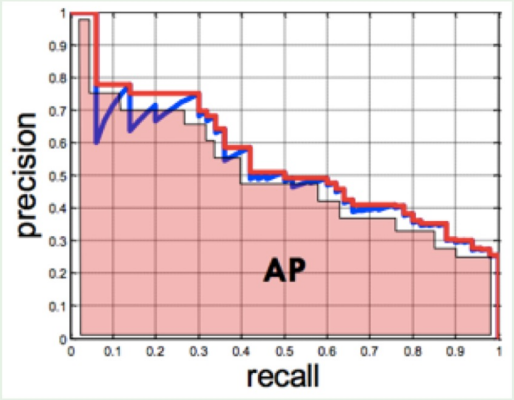
Slide from Ross Girshick

Formalizing the object detection task

Many possible ways, this one is popular:







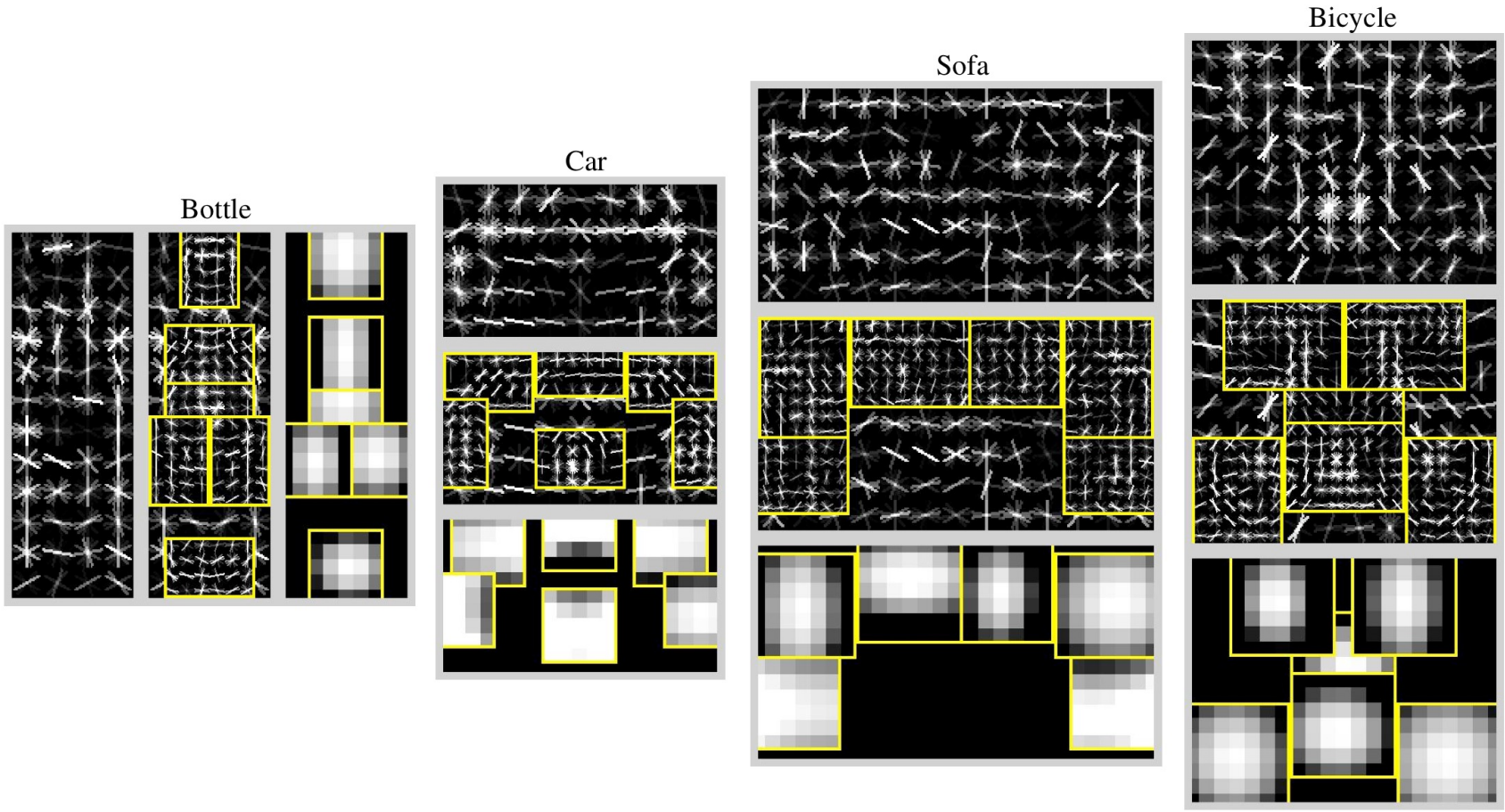
Performance summary:

Average Precision (AP)
0 is worst 1 is perfect

Bounding box overlap of 50% needed, to detect

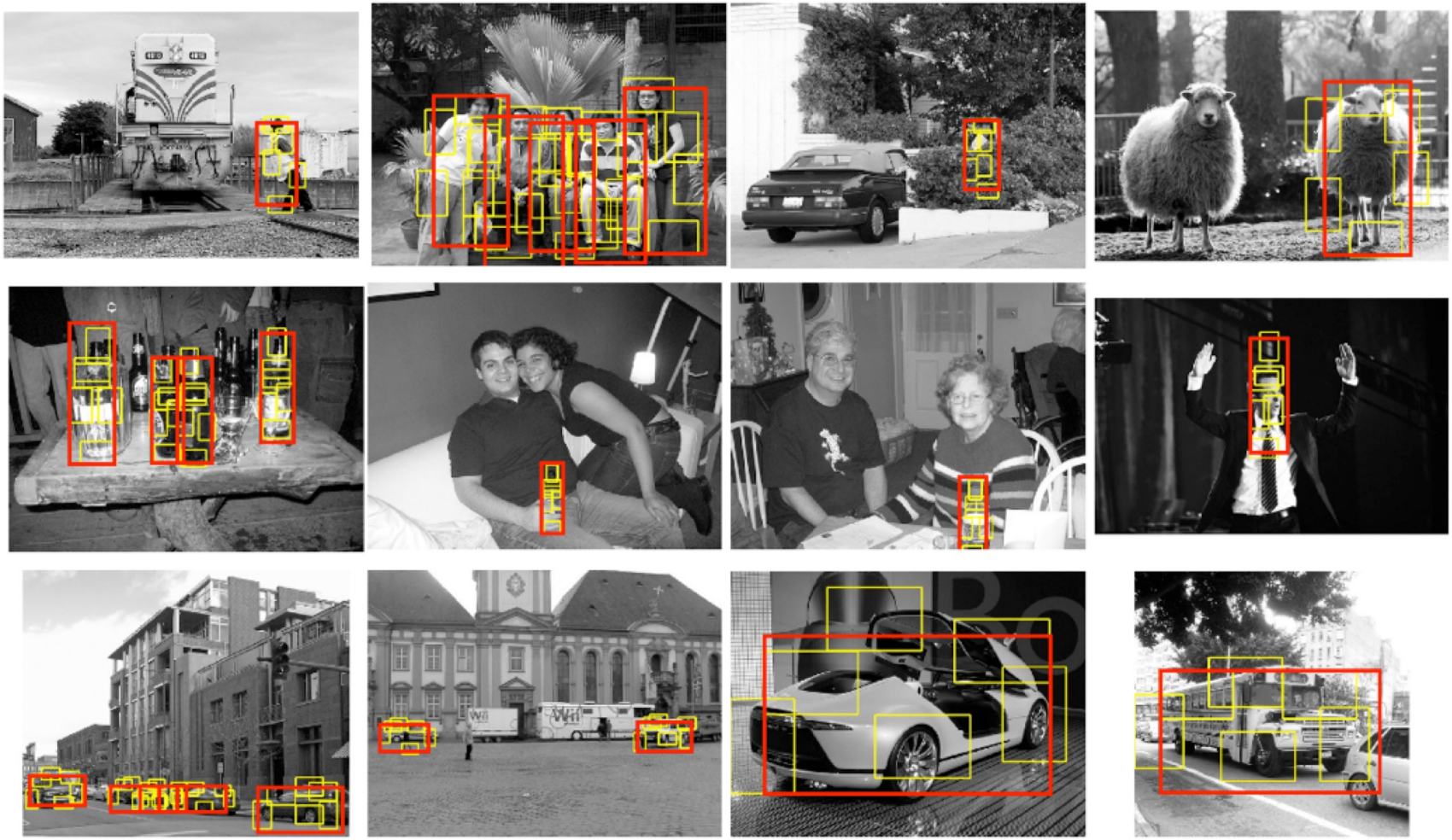
Slide from Ross Girshick

Learned Models



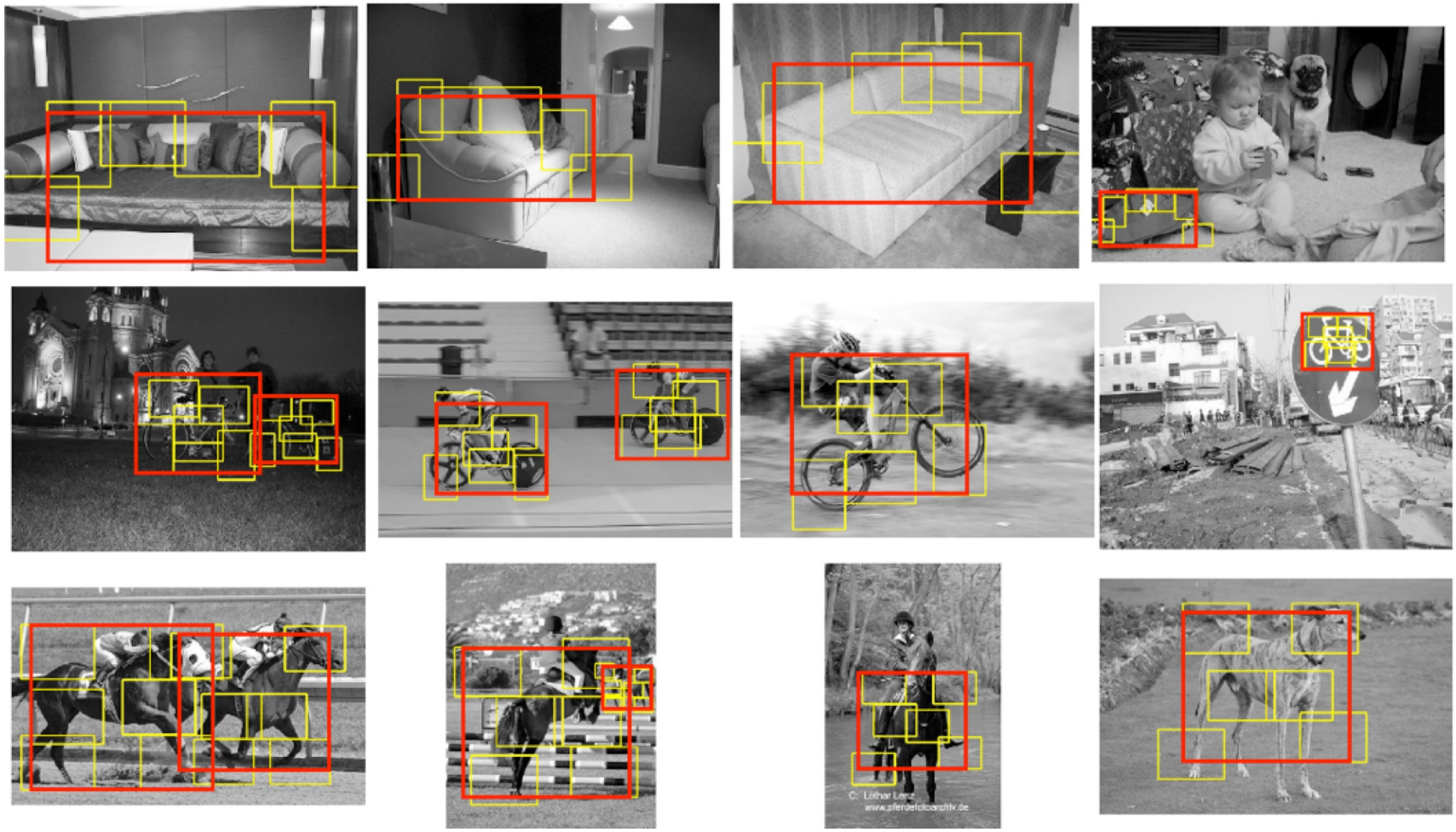
Slide from Pedro Felzenswalb

Example Results



Slide from Pedro Felzenswalb

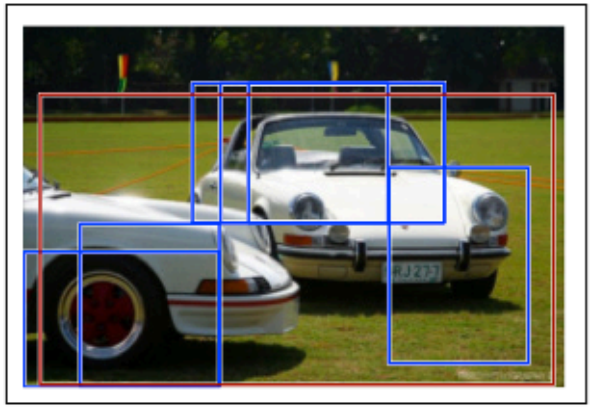
More Results

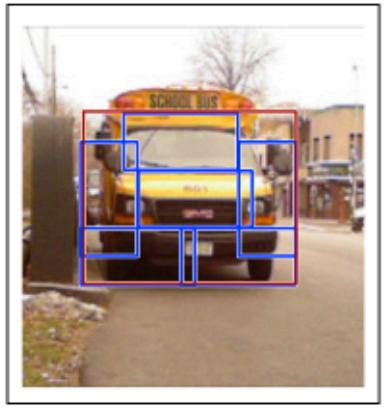


Slide from Pedro Felzenswalb

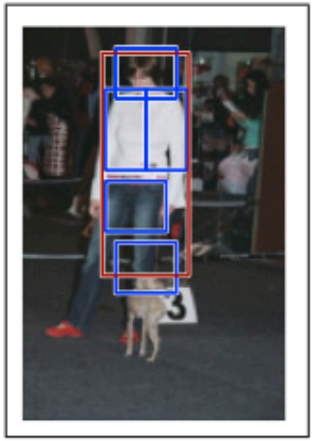
Some False Positives

Car

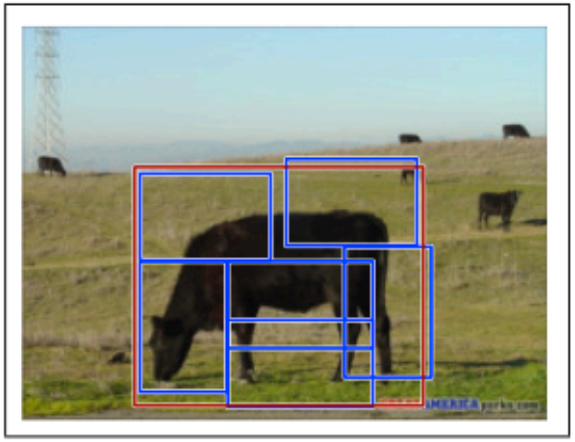


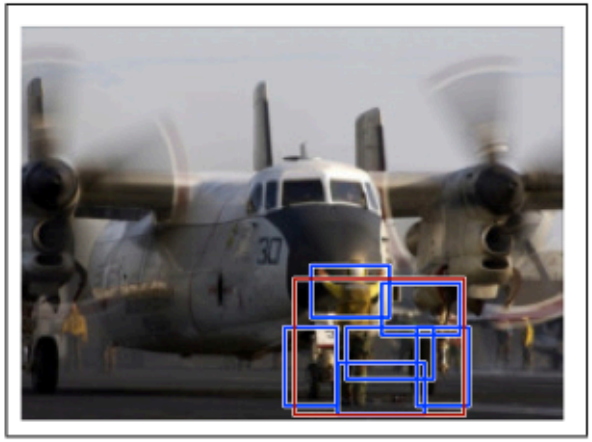


Person (Insufficient overlap)



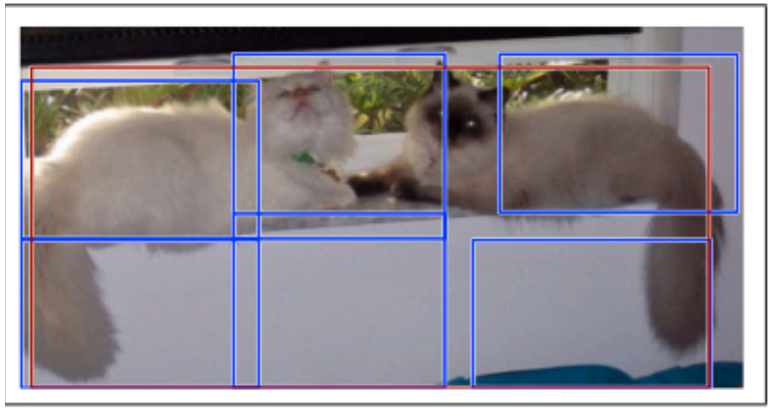
Horse

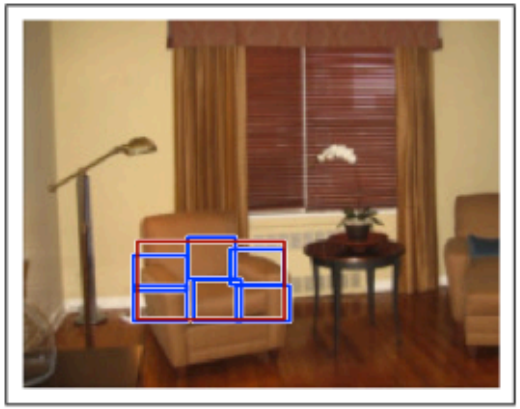




Some False Positives

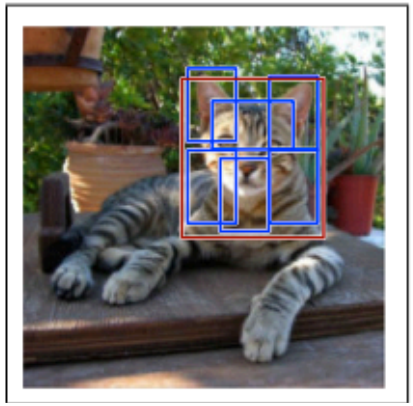
Sofa

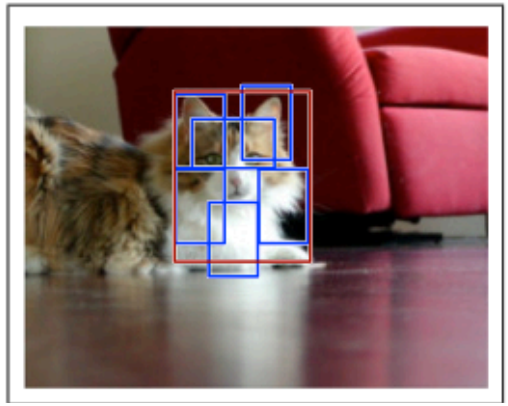




Bottle

Cat





(Due to insufficient overlap)

Overall Results

- 9 systems competed in the 2007 challenge
- Out of 20 classes we get:
  - First place in 10 classes
  - Second place in 6 classes
- Some statistics:
  - It takes ~2 seconds to evaluate a model in one image
  - It takes ~3 hours to train a model
  - MUCH faster than most systems

Slide from Pedro Felzenswalb

Summary: DPM Innovations

- Star-structured graph model to represent body parts and their geometric relations
- Mixtures of models to handle large variations in viewpoint / pose
- Latent-SVM formulation
- Data mining of hard negative examples
- PCA on HOG features for dimension reduction
- Performance on PASCAL challenge dataset