CMPSCI 670: Computer Vision

Object detection

University of Massachusetts, Amherst
November 5, 2014

Instructor: Subhransu Maji

Administrivia

- Homework 4 due today
- Homework 5 will be posted later today
  - *Recognition:* classify images into cats vs. dogs

vs

- DLS speaker:

Distinguished Lecturer Series

Maneesh Agrawala
University of California, Berkeley
EECS Department

Wednesday, November 5, 2014
4:00pm - 5:00pm
Computer Science Building, Room 151
Faculty Host: Evangelos Kalogerakis

"Storytelling Tools"

Storytelling is essential for communicating ideas. When they are well told, stories help us make sense of information, appreciate cultural or societal differences, and imagine living in entirely different worlds. Audio/visual stories in the form of radio programs, books-on-tape, podcasts, television, movies and animations, are especially powerful because they provide a rich multisensory experience. Technological advances have made it easy to capture stories using the microphones and cameras that are readily available in our mobile devices, But, the raw media rarely tells a compelling story.

Lecture outline

- Applications of object detection
- Approach and challenges
- Building an object detector
  - Multi-scale detection (Dalal & Triggs)
    - Detection as template matching
    - HOG feature pyramid
    - Non-maximum suppression
  - Training a detector — classifiers, hard negative mining
  - Evaluating a detector — some detection benchmarks
- Part-based models — poselets

Applications of detection

auto-focus based on faces

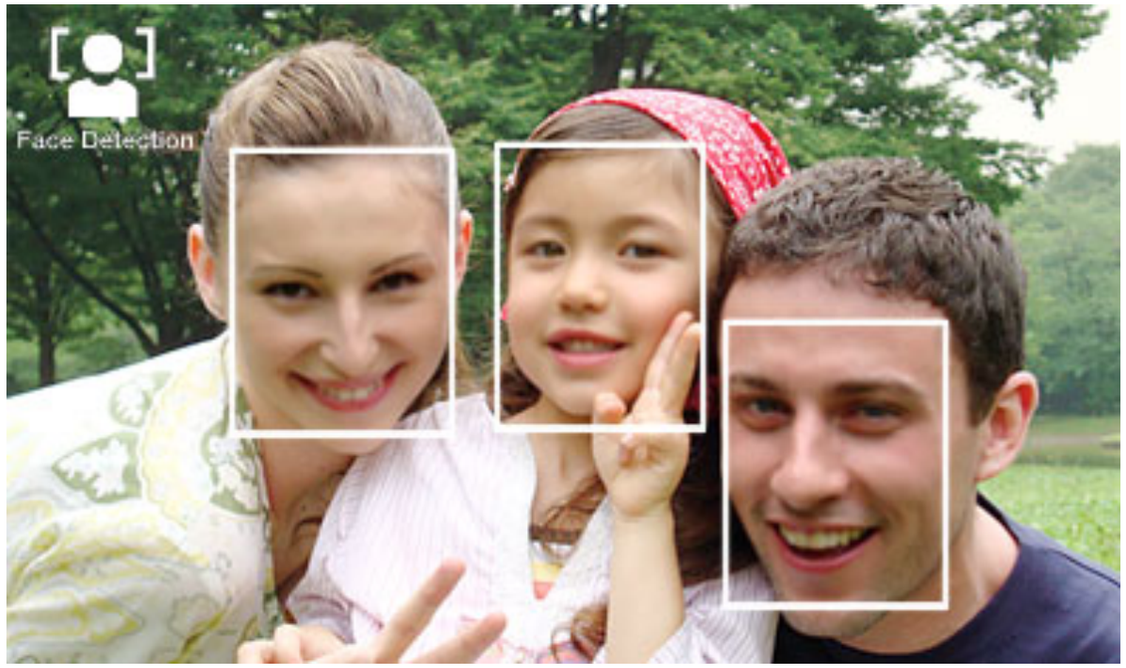
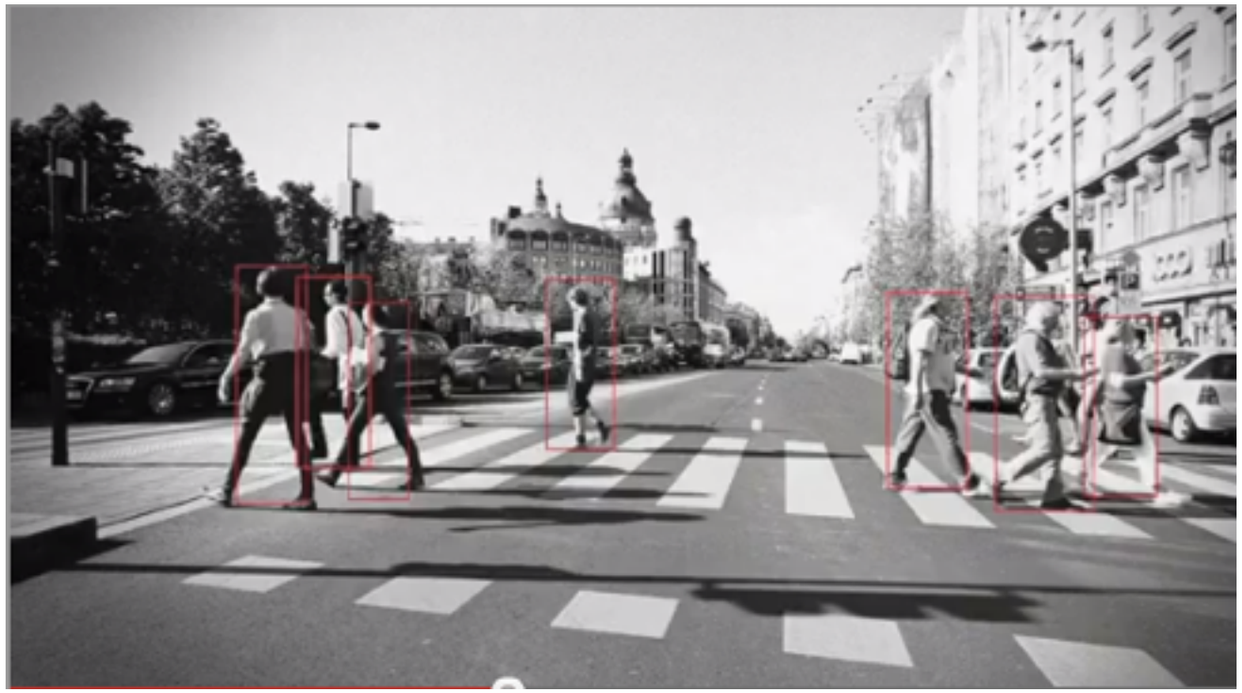
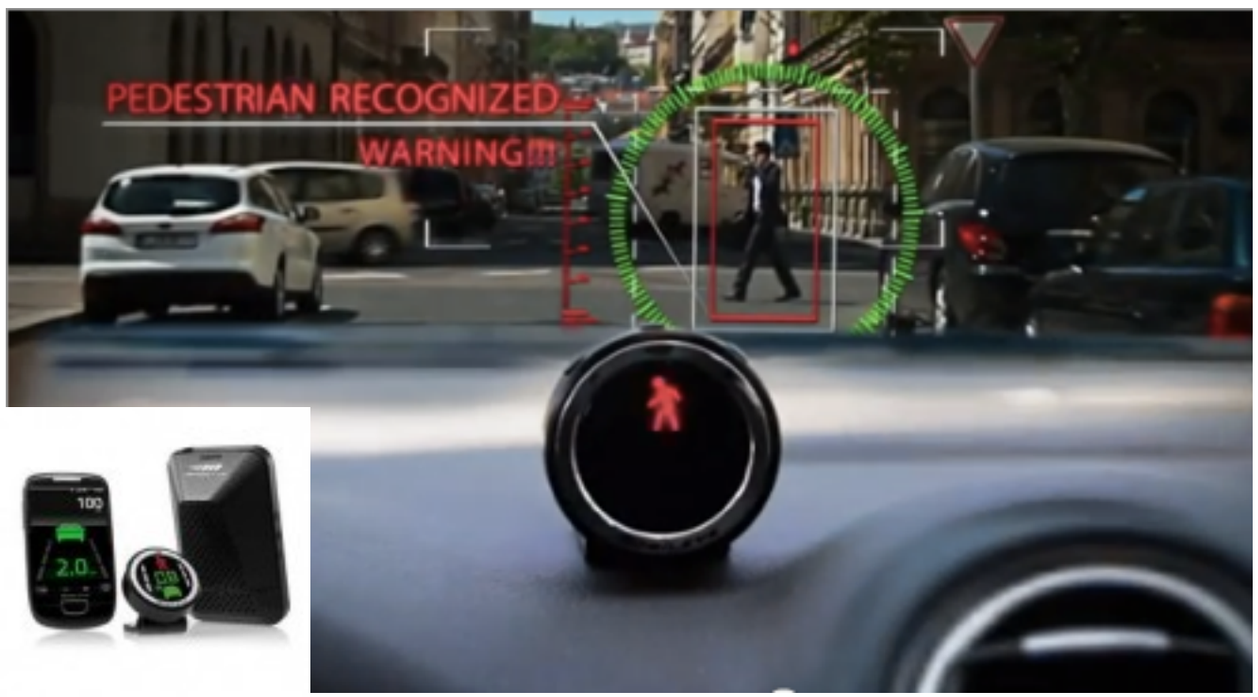


image credit : sony.co.in

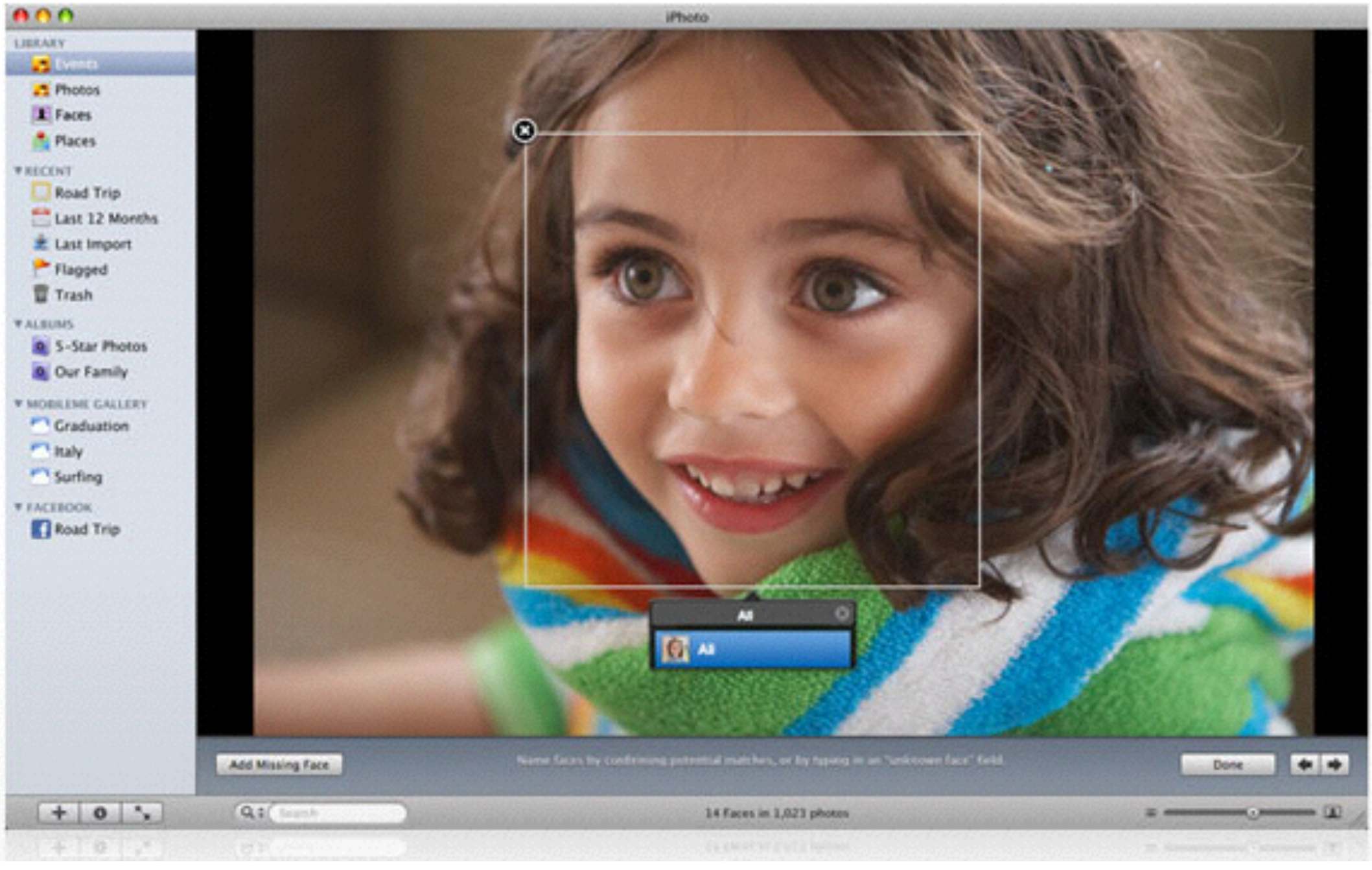
pedestrian collision warning





http://www.mobileye.com

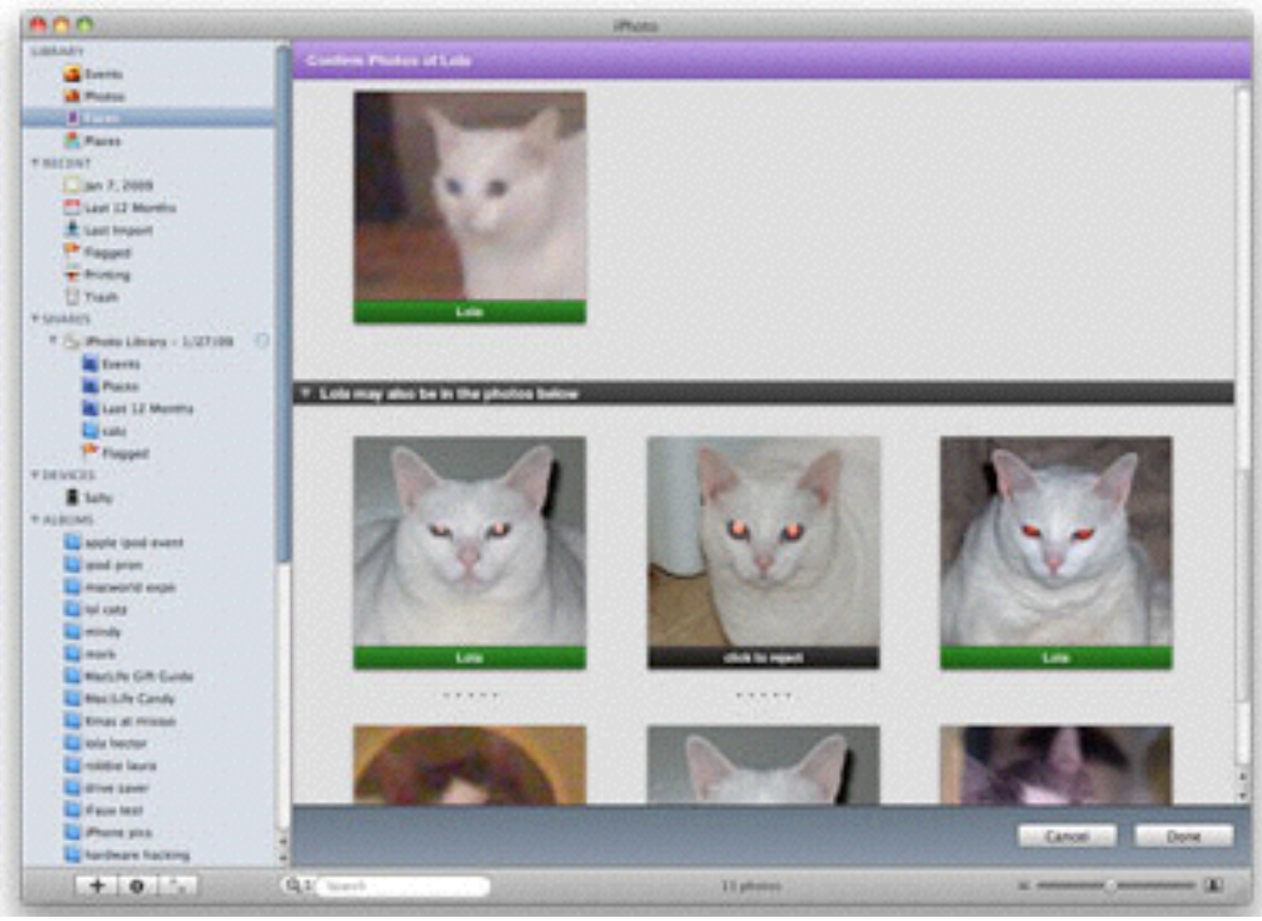
Consumer application: Apple iPhoto



http://www.apple.com/ilife/iphoto/

Consumer application: Apple iPhoto

Can be trained to recognize pets!



http://www.maclife.com/article/news/iphotos_faces_recognizes_cats

Consumer application: Apple iPhoto

Things iPhoto thinks are faces

Detection = repeated classification

face or not?

Detection

Challenges of object detection

- Sliding window detector must evaluate tens of thousands of location/scale combinations.
- Objects are rare
  - For example, there are on average 0–10 faces per image
    - A megapixel image has ~10^6 pixels and a comparable number of candidate face locations
    - For computational efficiency, we should try to spend as little time as possible on the non-face windows
    - To avoid having a false positive in every image, our false positive rate has to be less than 10^{-6}

Lecture outline

- Applications of object detection
- Challenges
- Building an object detector (Dalal & Triggs)
  - Detection as template matching
    - HOG feature pyramid
    - Non-maximum suppression
  - Learning a template — linear SVMs, hard negative mining
  - Evaluating a detector — some detection benchmarks
- Part-based models — poselets

Detection as template matching

- Consider matching with image patches
  - What could go wrong?

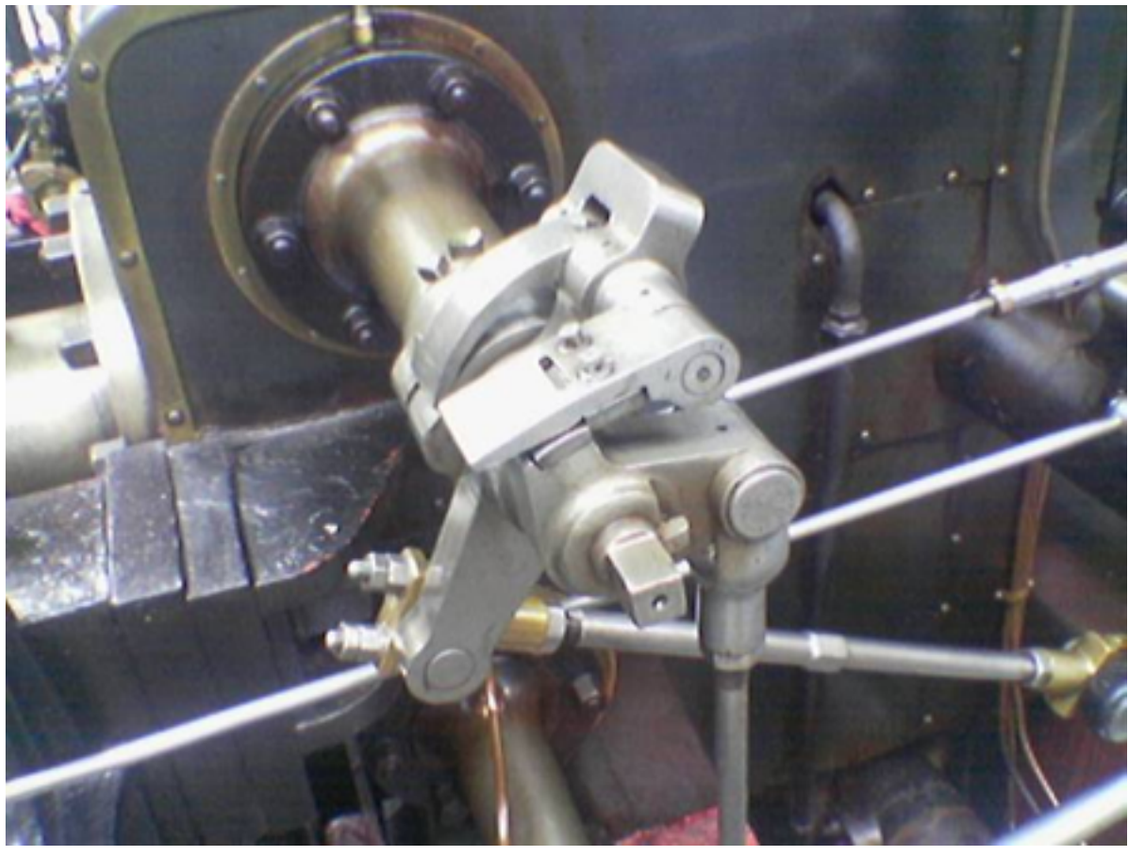
template

image

match quality
e.g., cross correlation

What is a feature map?

- Any transformation of an image into a new representation
- Example: transform an image into a binary edge map



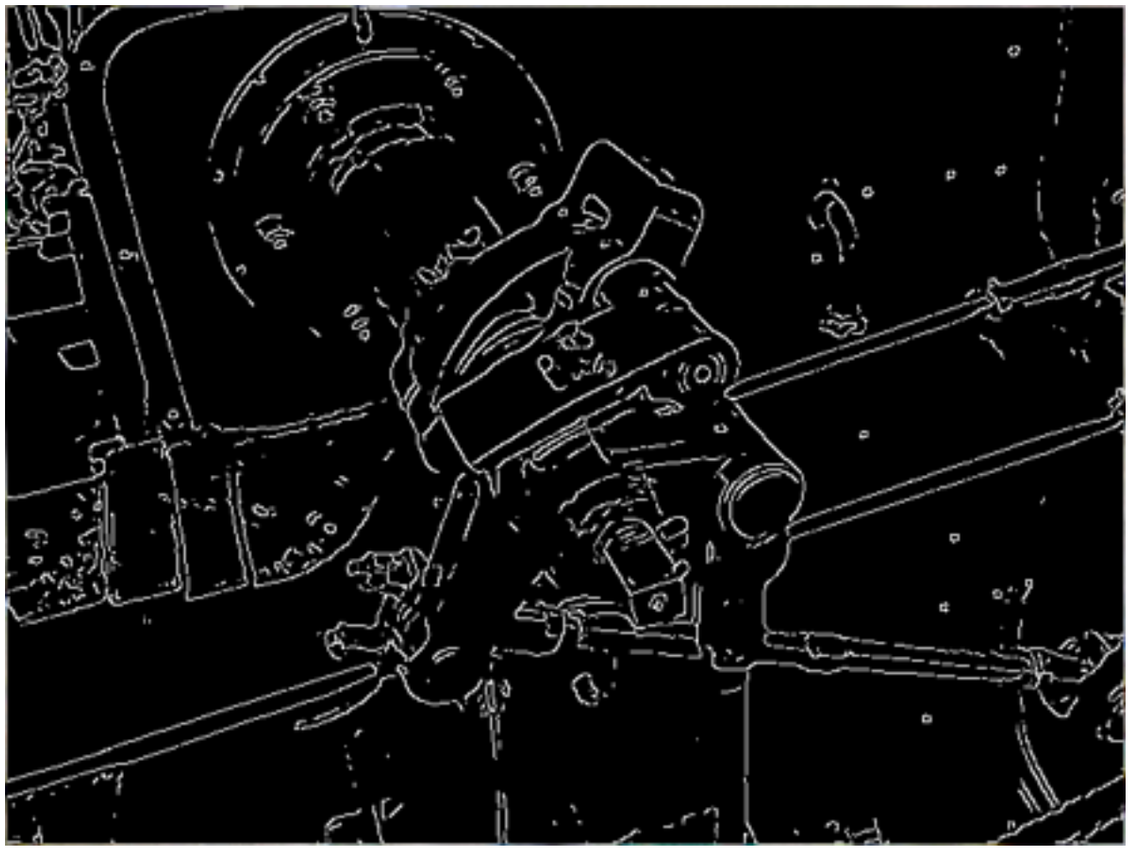


Image source: wikipedia

Feature map goals

- Introduce invariance
  - Bias, gain, nonlinear transformations
  - Small deformations

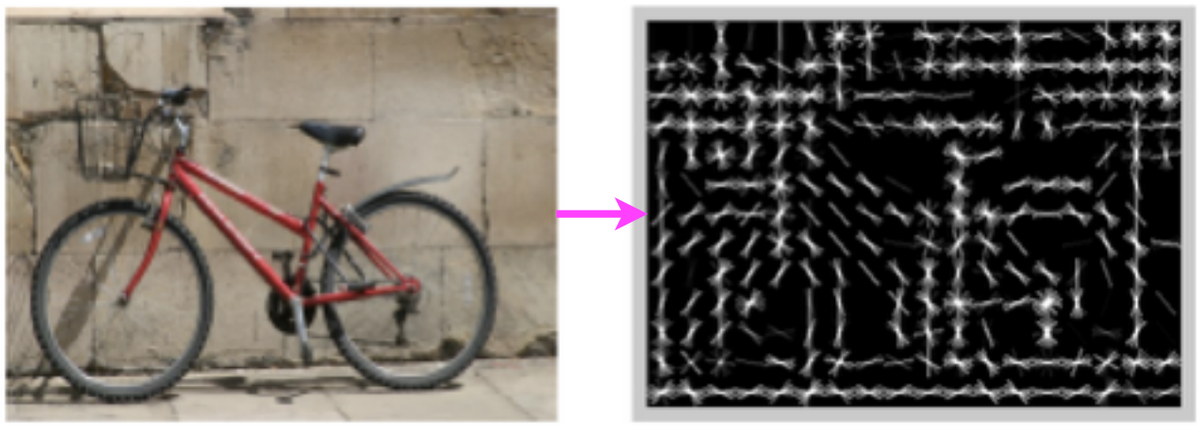
Figure 1.3: Variation in appearance due to a change in illumination

- Preserve larger scale spatial structure

Image: [Fergus05]

Histograms of oriented gradients (HOG)

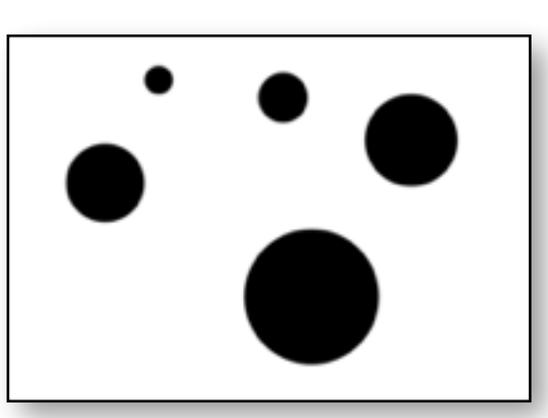
- Introduce invariance
  - Bias / gain / nonlinear transformations
    - bias: gradients / gain: local normalization
    - nonlinearity: clamping magnitude, orientations
  - Small deformations
    - spatial subsampling
    - local “bag” models

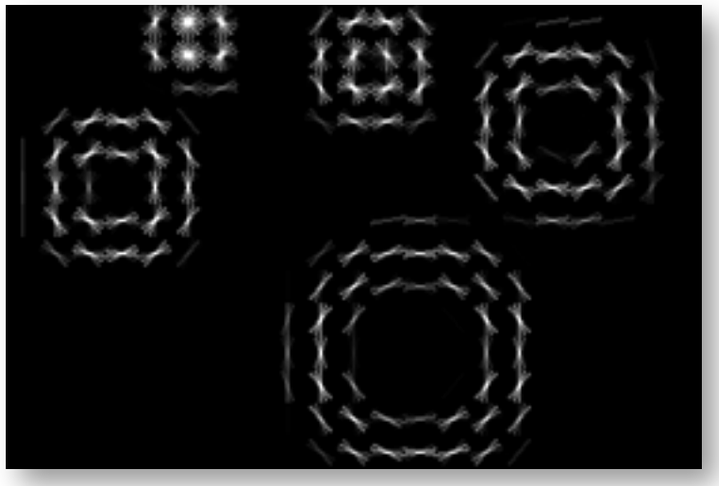


- References
  - “Histograms of oriented gradients for human detection.” N. Dalal and B. Triggs, CVPR 2005.
  - “Finding people in images and videos.” N. Dalal, Ph.D. Thesis, Institut National Polytechnique de Grenoble, 2006.

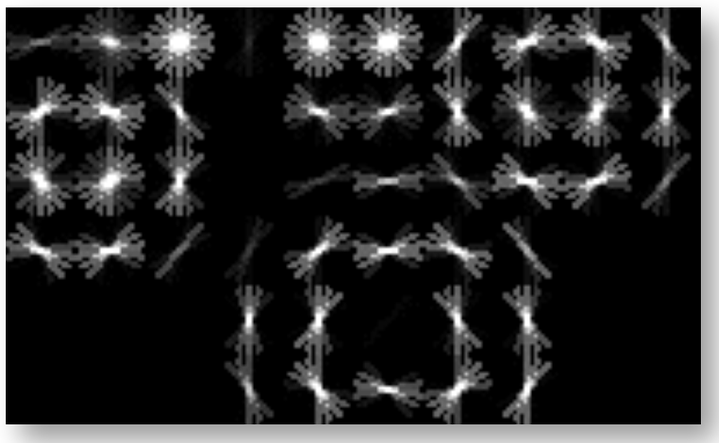
Histograms of oriented gradients (HOG)

- Partition image into blocks at multiple scales and compute histogram of gradient orientations in each block





10x10 cells



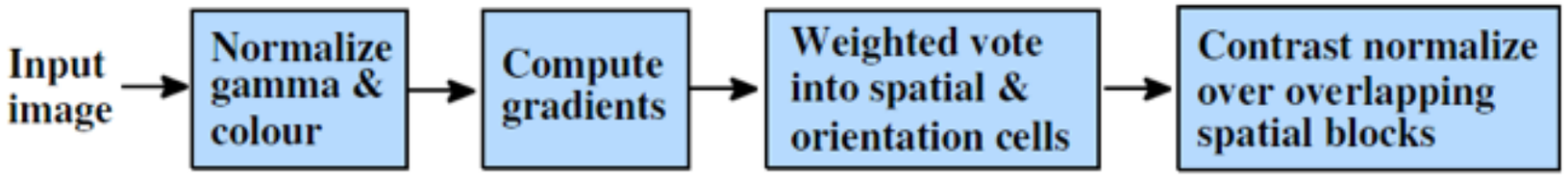
20x20 cells

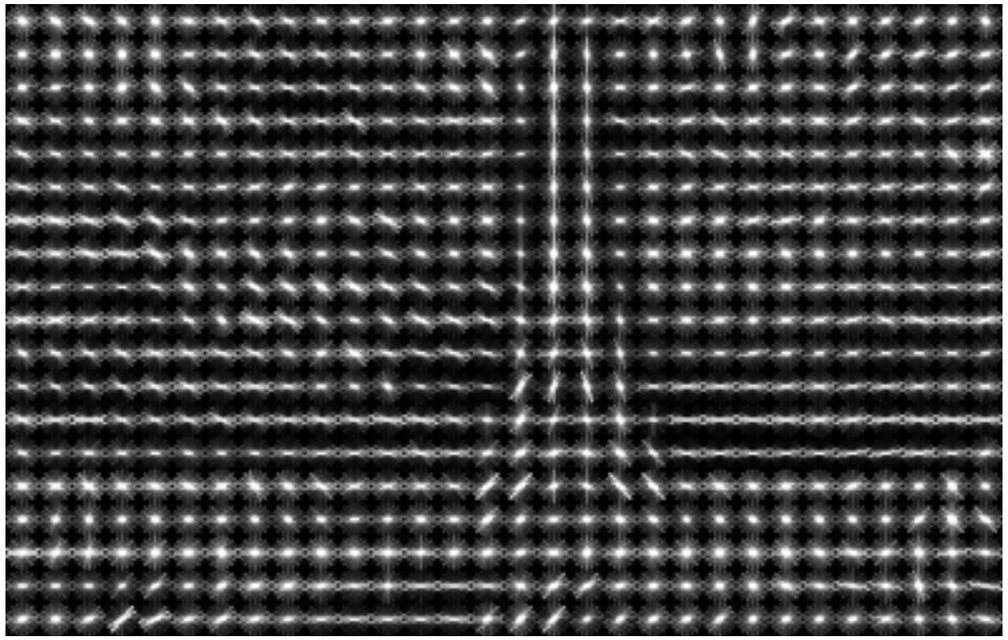
N. Dalal and B. Triggs, Histograms of Oriented Gradients for Human Detection, CVPR 2005

Image credit: N. Snavely

Histograms of oriented gradients (HOG)

- Partition image into blocks at multiple scales and compute histogram of gradient orientations in each block



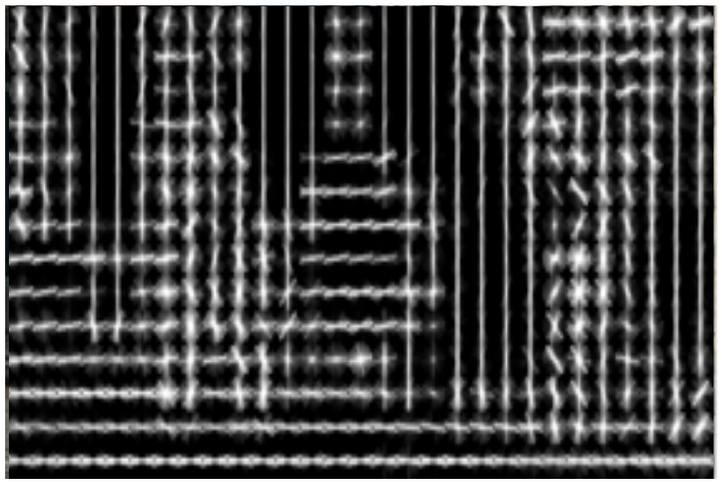


N. Dalal and B. Triggs, Histograms of Oriented Gradients for Human Detection, CVPR 2005

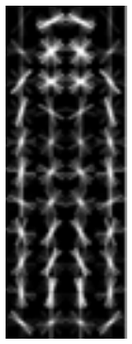
Image credit: N. Snavely

Template matching with HOG

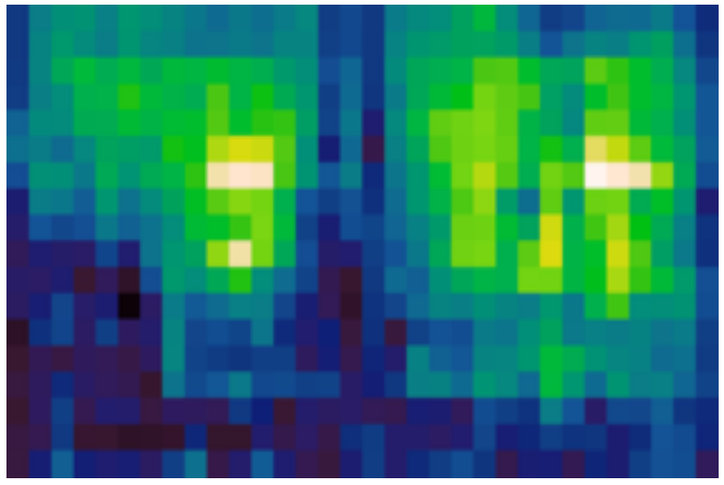
HOG feature map



Template

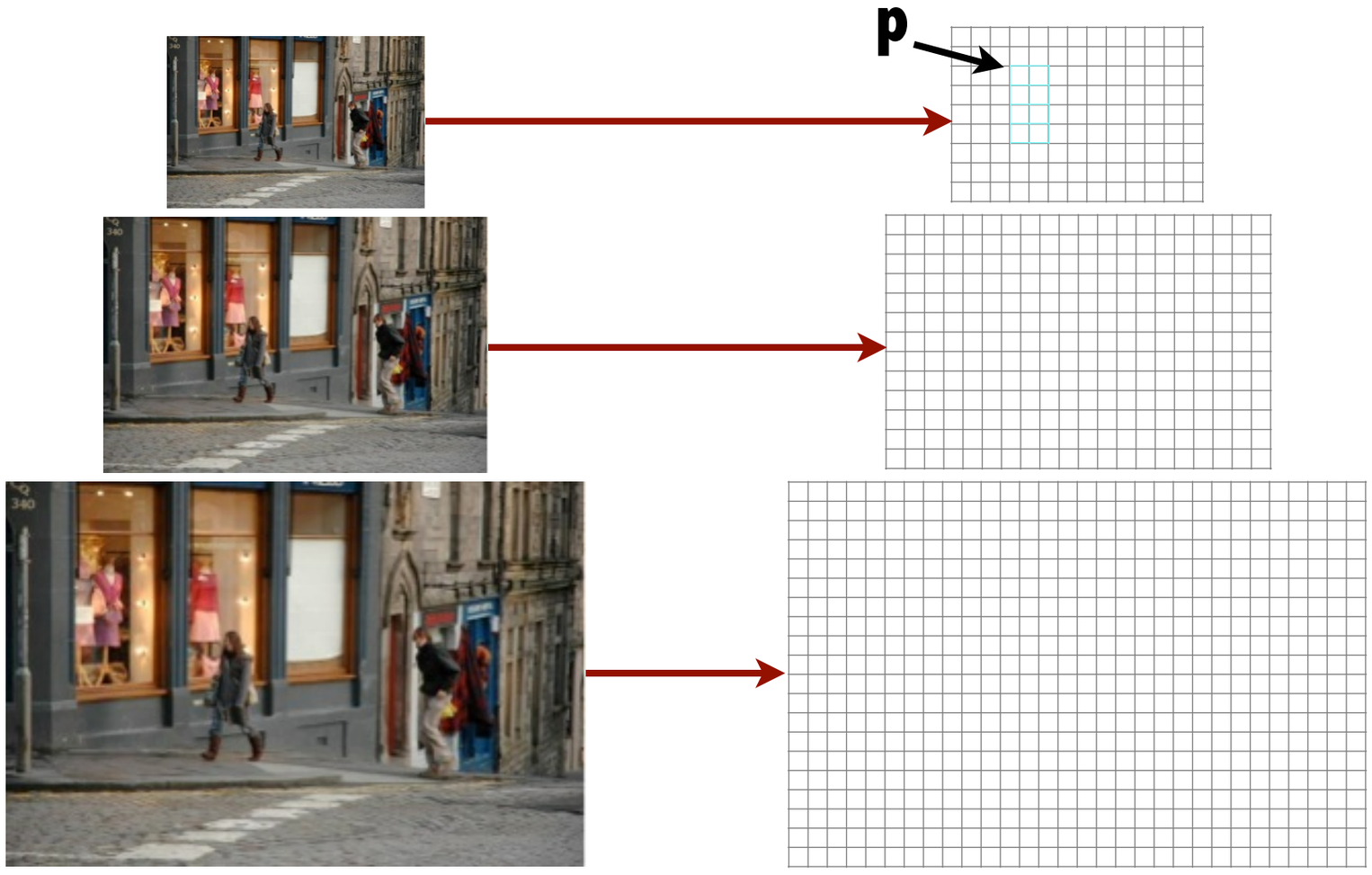


Detector response map

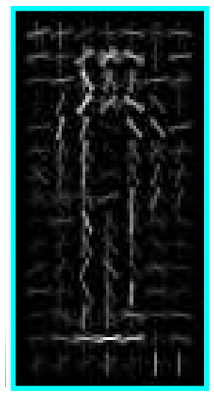


- Compute the HOG feature map for the image
- Convolve the template with the feature map to get score
- Find peaks of the response map (non-max suppression)
- What about multi-scale?

Multi-scale template matching

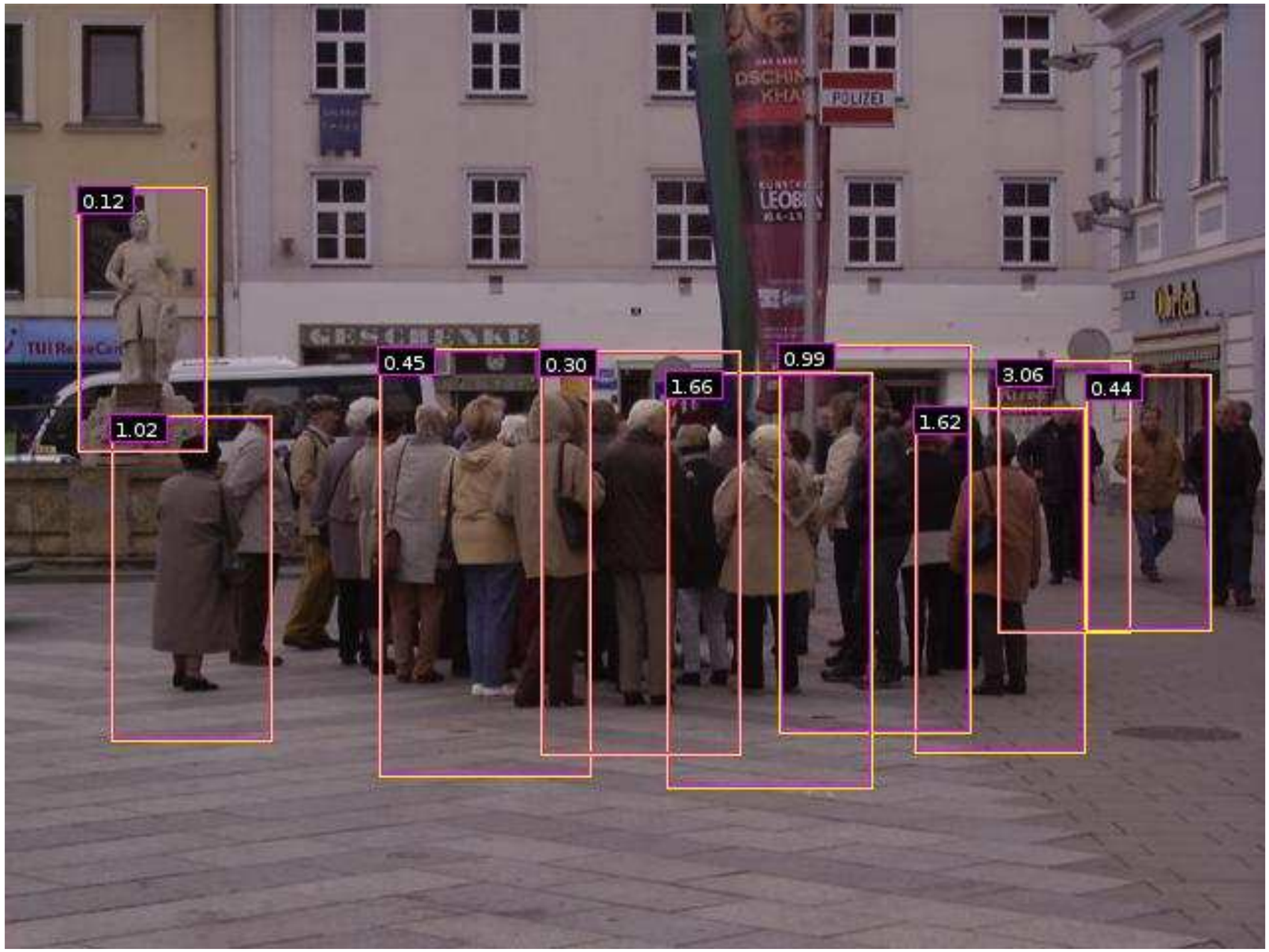


$$score(I, p) = \mathbf{w} \cdot \phi(I, p)$$



- Compute HOG of the whole image at multiple resolutions
- Score each sub-windows of the feature pyramid

Example pedestrian detections



[Dalal06]

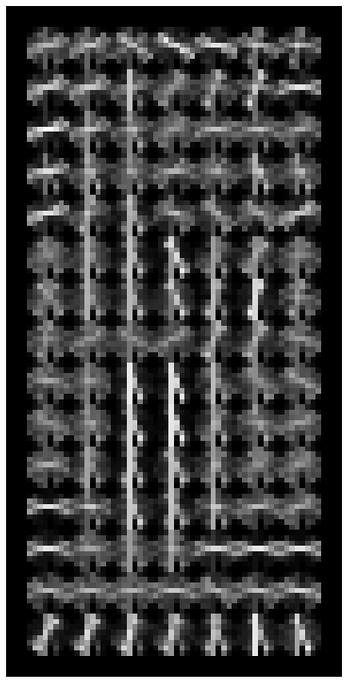
Lecture outline

- Applications of object detection
- Challenges
- Building an object detector (Dalal & Triggs)
  - Detection as template matching
    - HOG feature pyramid
    - Non-maximum suppression
  - Learning a template — linear SVMs, hard negative mining
  - Evaluating a detector — some detection benchmarks
- Part-based models — poselets

Learning a template

Pos = {…  …}

Annotations



is this template good?

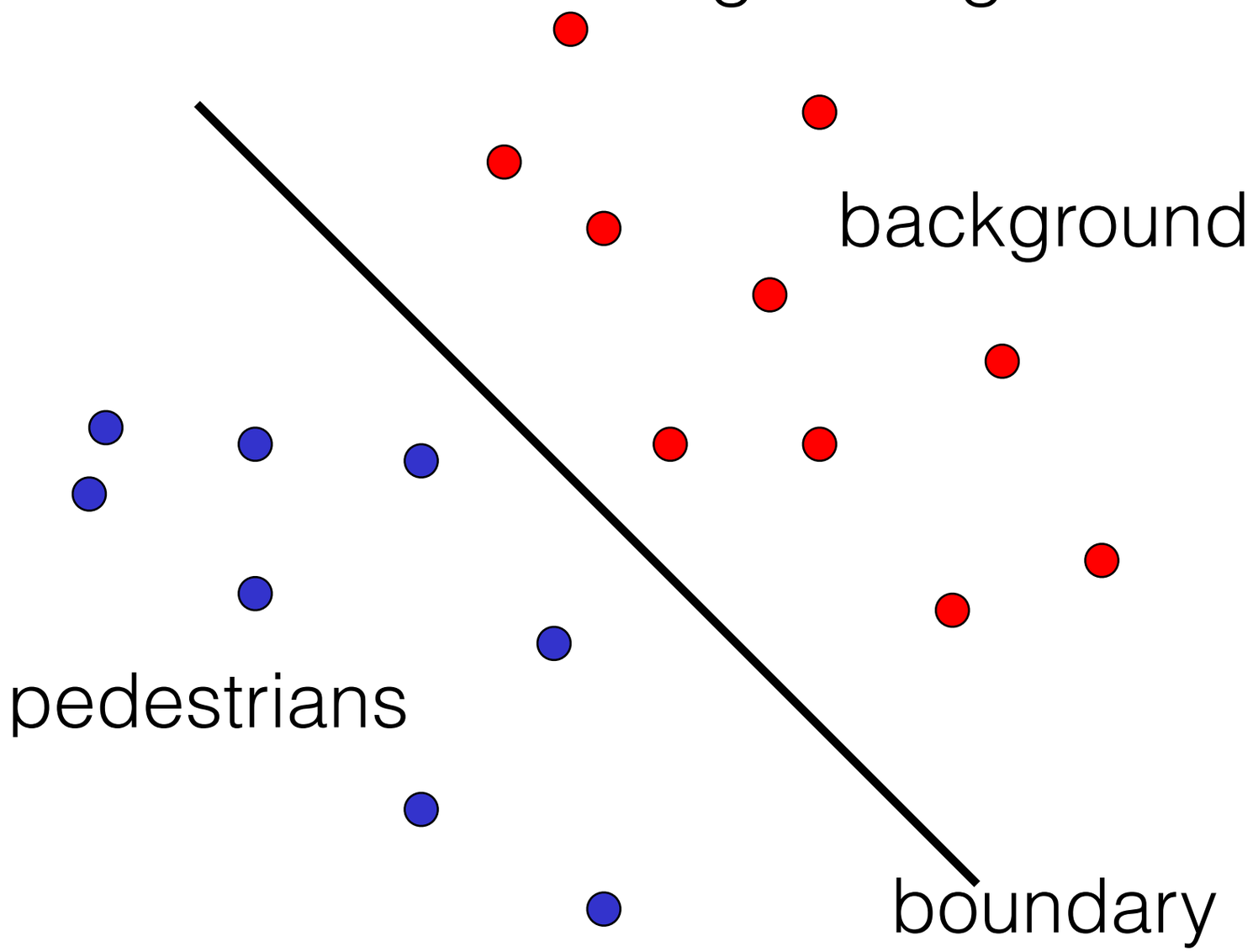
Cropped positive

HOG

[Dalal06]

Learning a template

- Score high on pedestrians and low on background patches
- Discriminative learning setting — lets use linear SVMs!

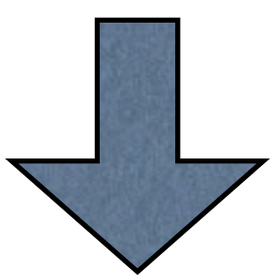


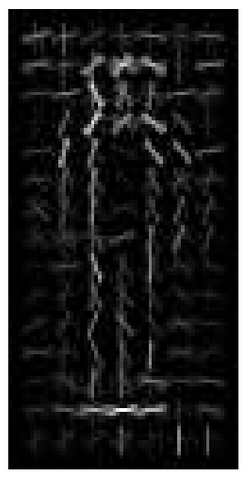
Issue: too many background patches

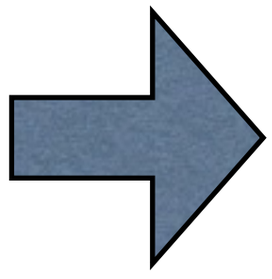
Initial training

Pos = {…  …}

Neg = {… random background patches …}







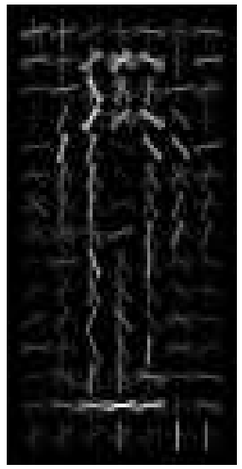
Test on cropped windows

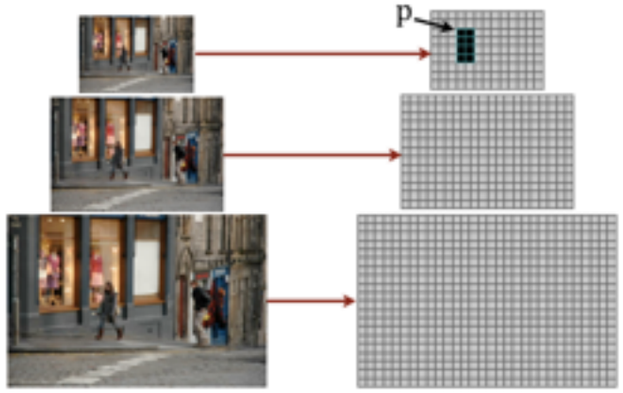
SVM

Mining hard negatives

Pos ={…  …}

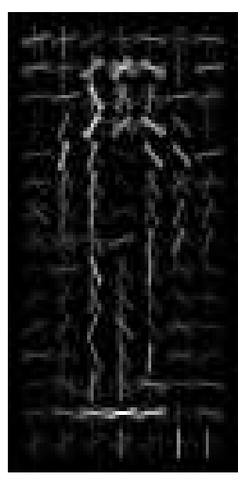
Neg_{rand} = {... random background patches ...}





SVM “Hard” negatives

+ Neg_{hard} = {... windows with score >= -1 ...}



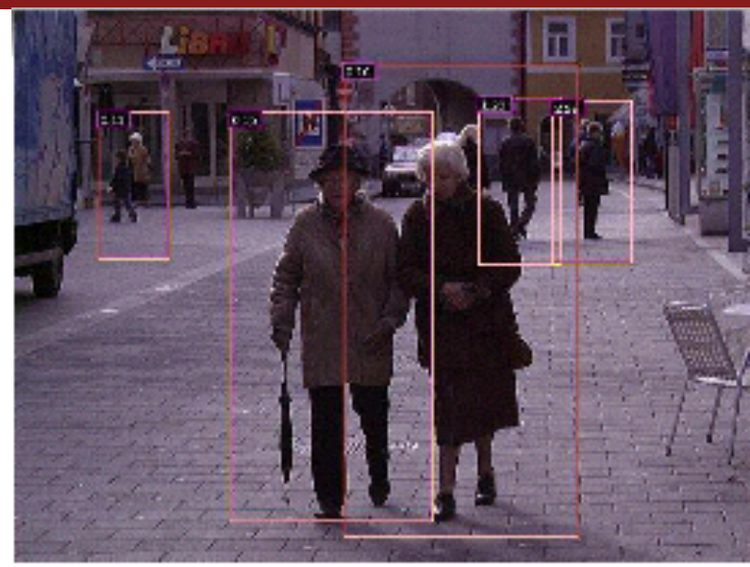
SVM

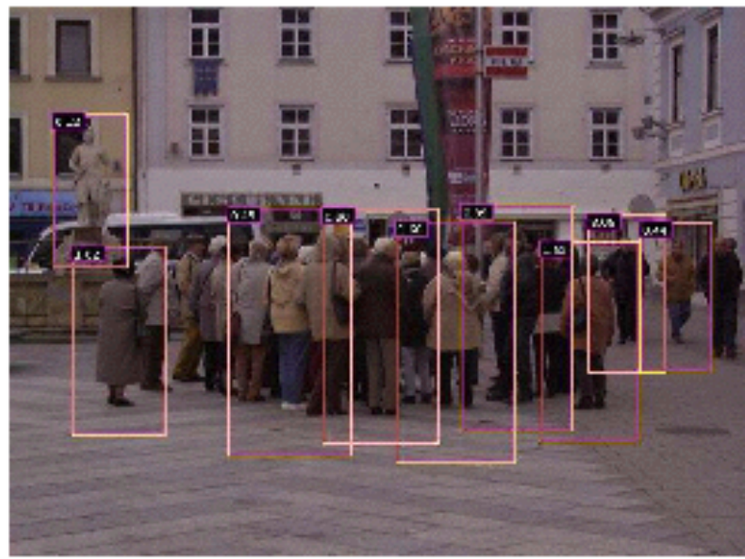
Lecture outline

- Applications of object detection
- Challenges
- Building an object detector (Dalal & Triggs)
  - Detection as template matching
    - HOG feature pyramid
    - Non-maximum suppression
  - Learning a template — linear SVMs, hard negative mining
  - Evaluating a detector — some detection benchmarks
- Part-based models — poselets

INRIA person dataset

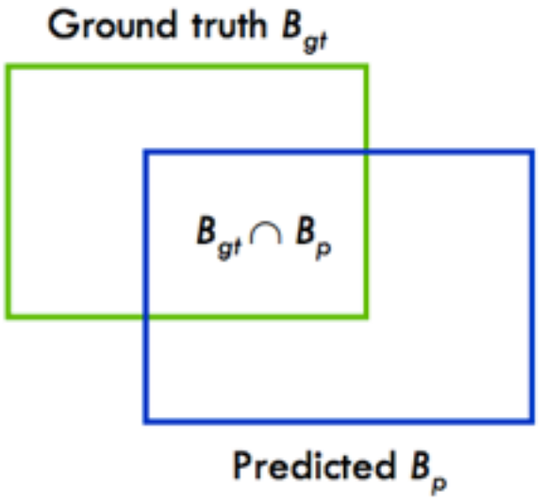
- N. Dalal and B. Triggs, CVPR 2005
- One of the first realistic datasets
  - Wide variety of articulated poses
  - Variable appearance/clothing
  - Complex backgrounds
  - Unconstrained illumination
  - Occlusions, different scales



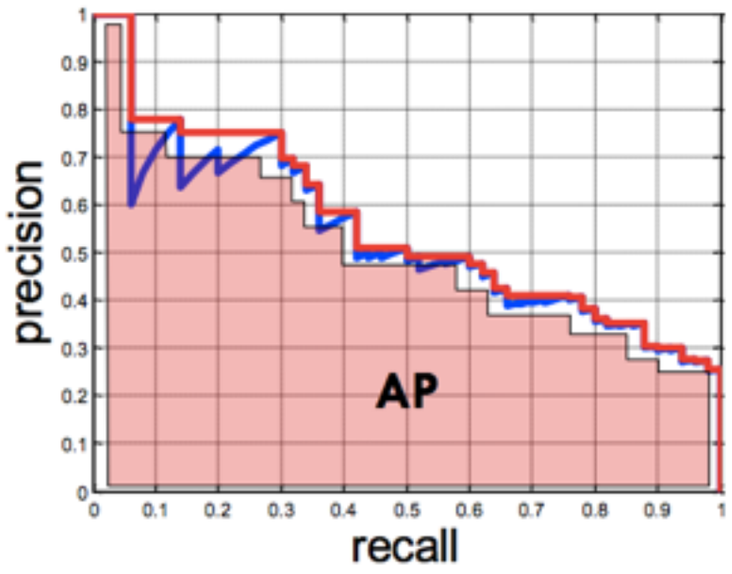


http://pascal.inrialpes.fr/data/human/

Detection evaluation

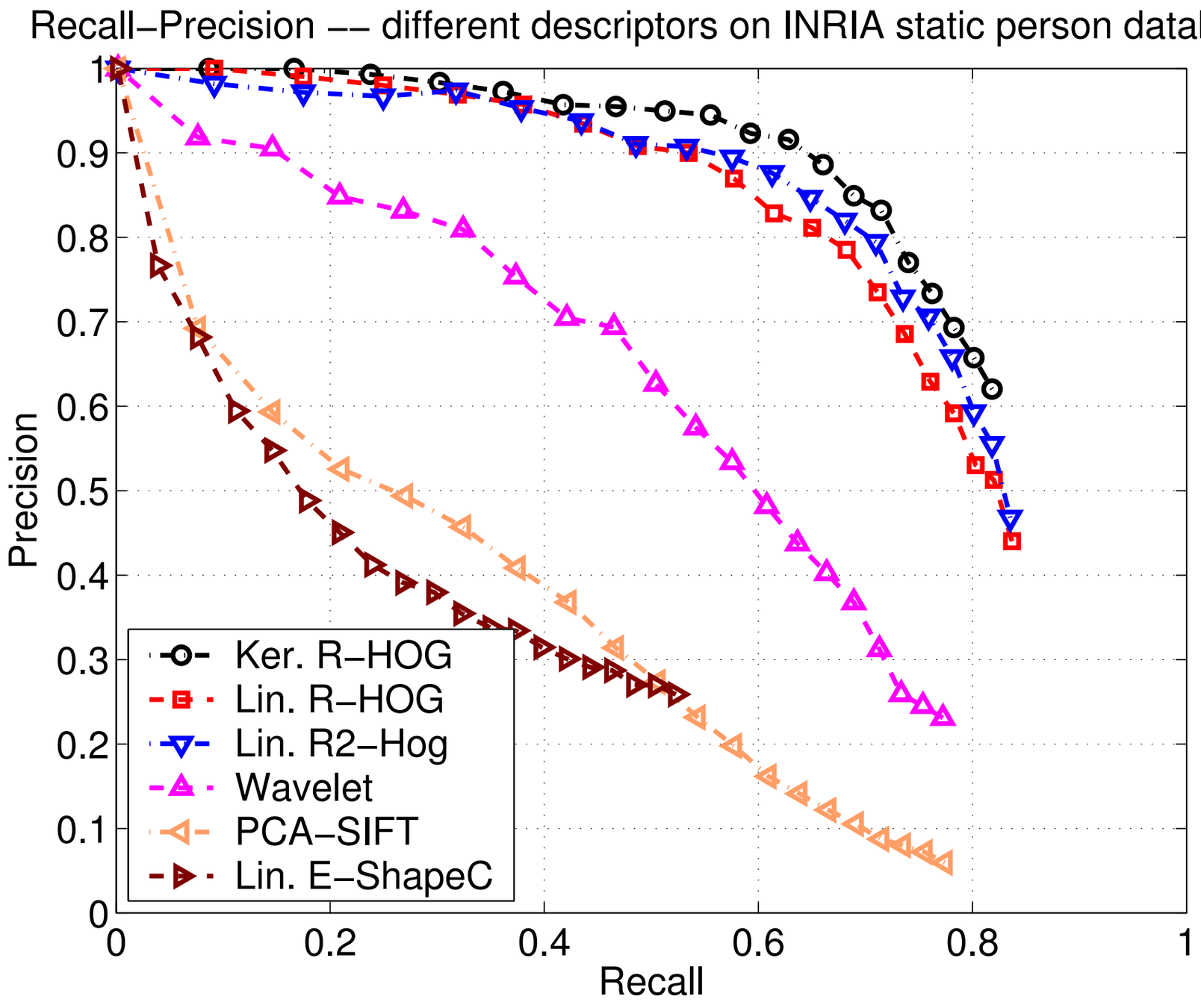


$$\text{overlap}(B_{\text{gt}}, B_{\text{p}}) = \frac{|B_{\text{gt}} \cap B_{\text{p}}|}{|B_{\text{gt}} \cup B_{\text{p}}|}$$



- Assign each prediction to
  - true positive (TP) or false positive (FP)
- $\text{Precision}_{@k} = \#\text{TP}_{@k} / (\#\text{TP}_{@k} + \#\text{FP}_{@k})$
- $\text{Recall}_{@k} = \#\text{TP}_{@k} / \#\text{TotalPositives}$
- Average Precision (AP)

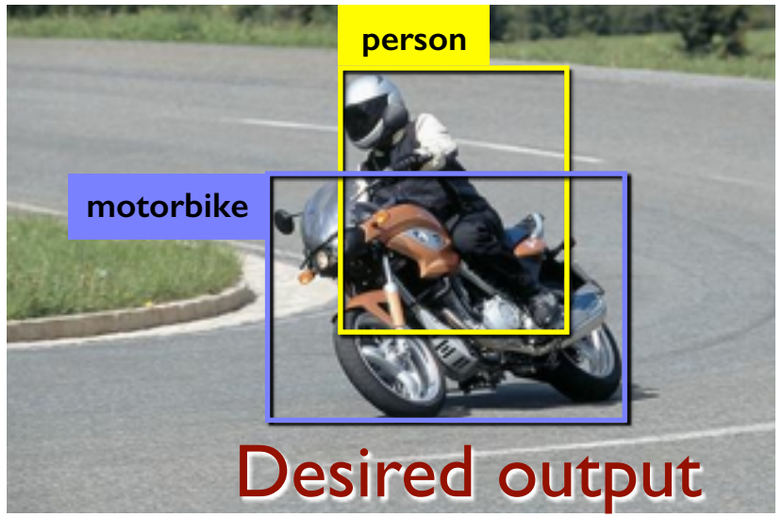
Pedestrian detection on INRIA dataset



- AP = 0.75 with a linear SVM
- Very good, right?

PASCAL VOC Challenge

- Localize & name (*detect*) 20 basic-level object categories
  - Airplane, bicycle, bus, cat, car, dog, person, sheep, sofa, monitor, *etc.*



- Run from 2005 - 2012
- 11k training images with 500 to 8000 instances / category
- Substantially more challenging images
- Dalal and Triggs detector AP on ‘person’ category: **12%**

PASCAL examples

PASCAL examples

- Viewpoint

Image credits: PASCAL VOC

PASCAL examples

- Subcategory — “airplane” images

Image credits: PASCAL VOC

PASCAL examples

- Subcategory — “car” images

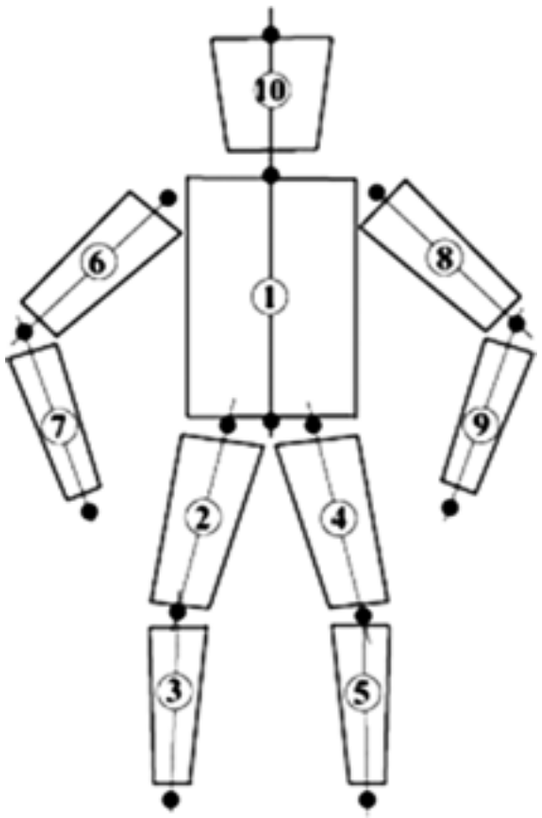
Image credits: PASCAL VOC

Part-based models

- A single template is not enough to explain the variability
  - “person” detection AP = 12% using a single template
- Lets focus on the person category
  - viewpoint, articulation, clothing, etc cause wide appearance change. How can we model this?
- **Idea:** lets try to detect parts and stitch them together
- But what should the parts be?

Parts based on human anatomy

pictorial structures



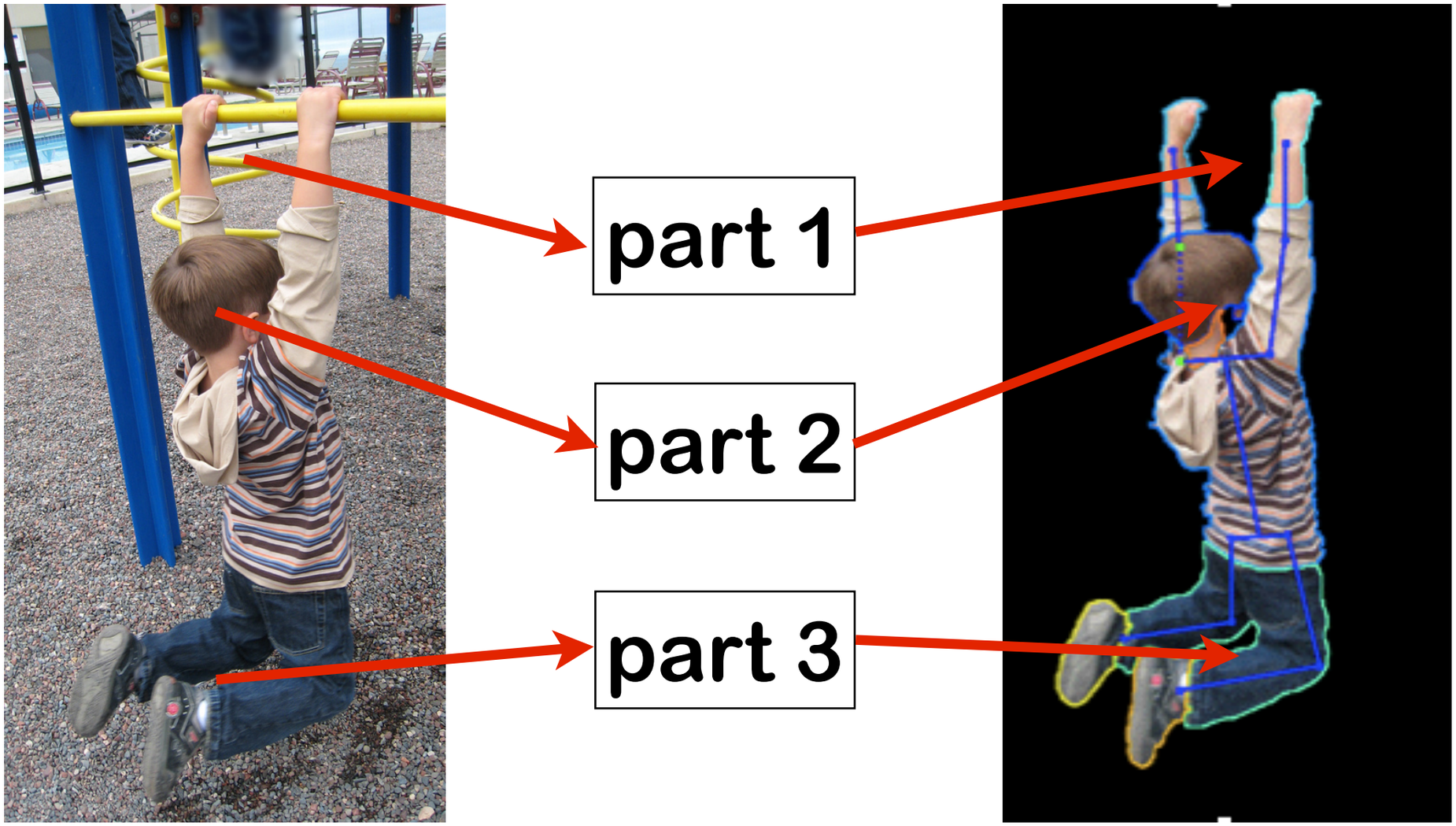
“stick-figure models”

Fisher & Elchlager 73, Nevatia & Binford 77,
Felzenszwalb et al. 05, Ren et al. 05,
Andriluka et al. 09, Ferrari et al. 08,
Ramanan 06

it is hard to detect limbs

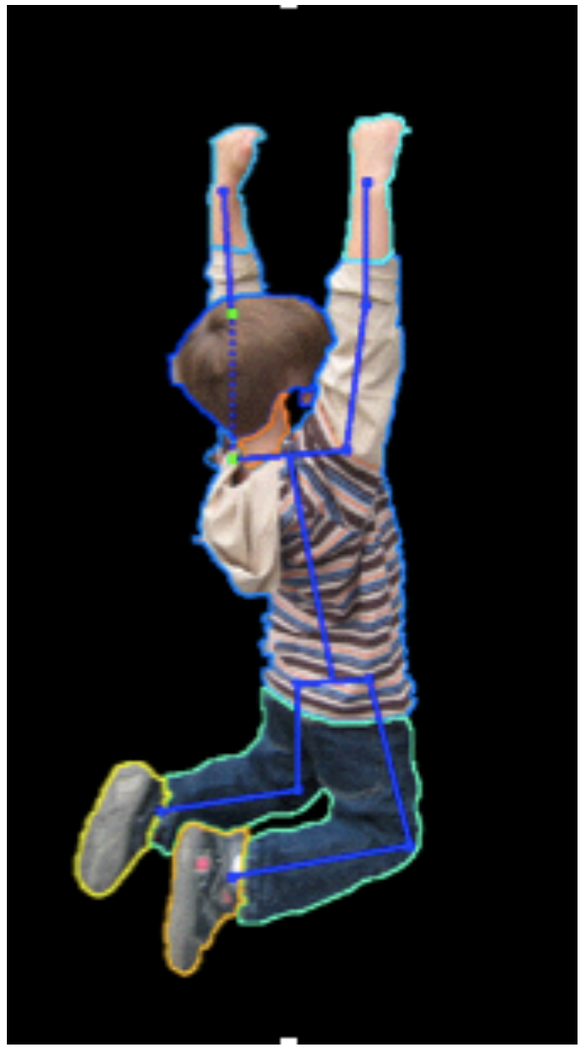
Can we leverage the success of *face* and *pedestrian* detectors?

Properties of good parts



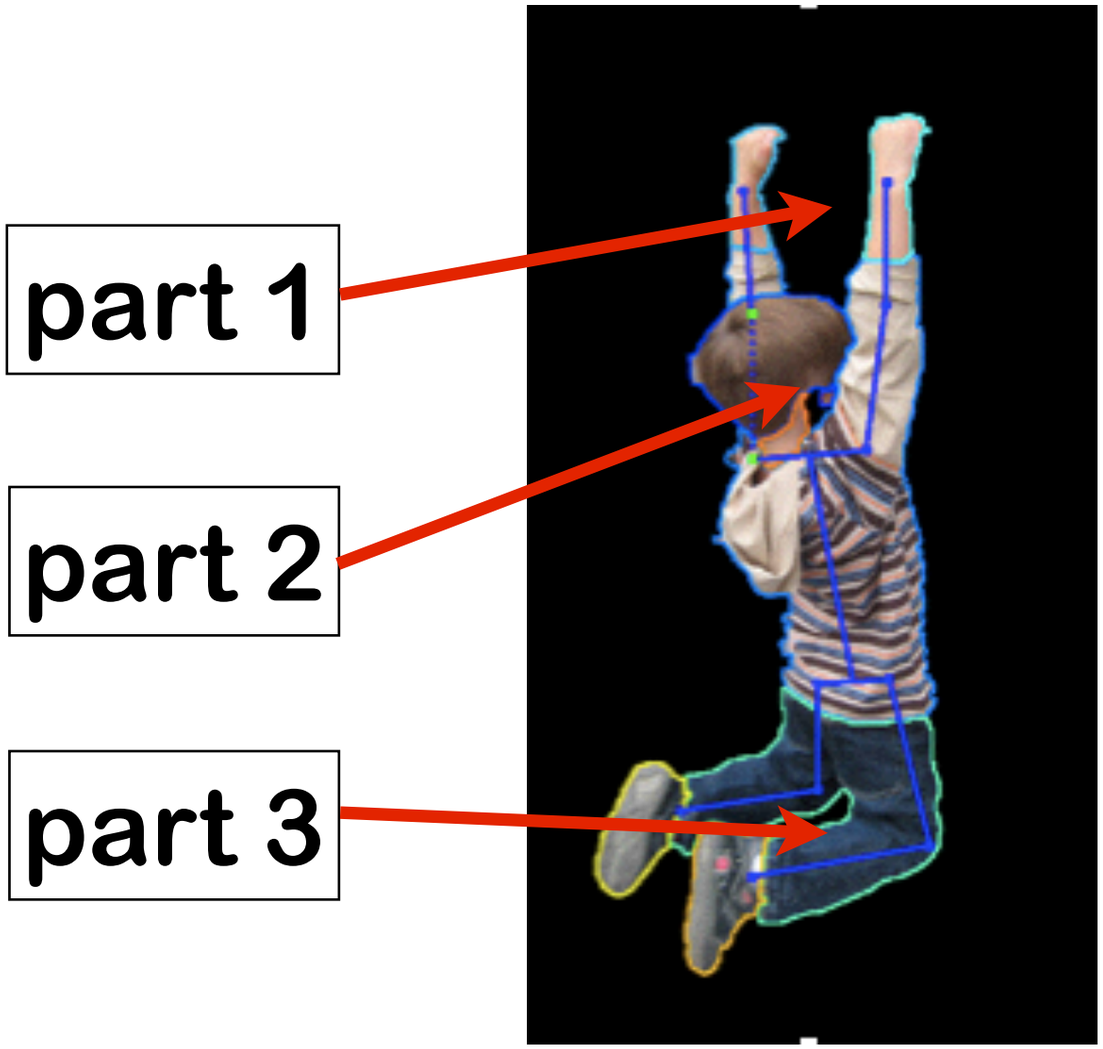
Parts should be useful beyond detection:
pose, gender, clothing, age, action, hair-style, etc.

Properties of good parts



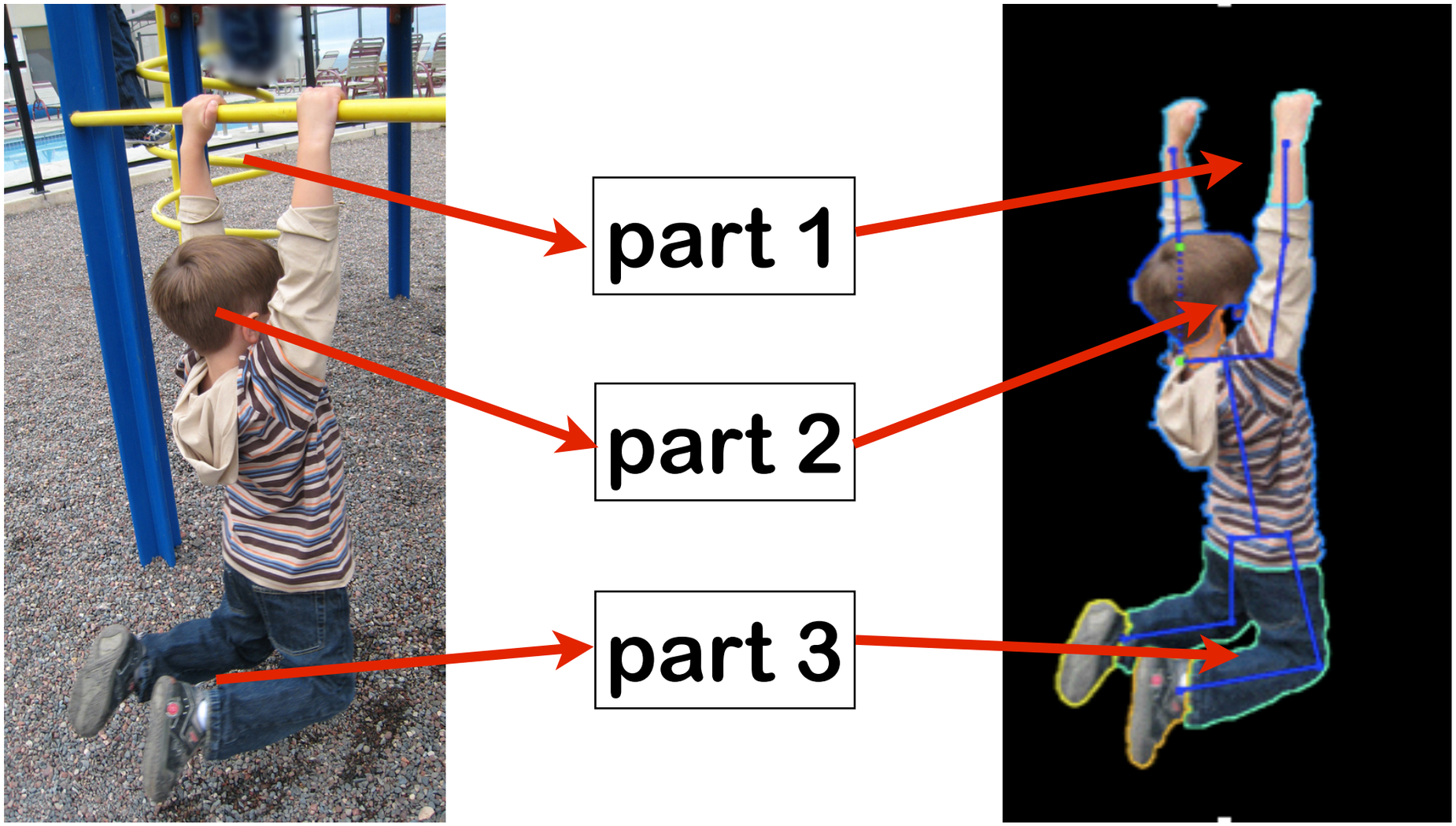
It should be easy to detect the part from the image
i.e., want discriminative parts such as faces

Properties of good parts



It should be easy to predict the pose given the part
i.e., want parts tightly clustered in pose space

Properties of good parts



Want parts that are (1) visually discriminative and are (2) semantically meaningful

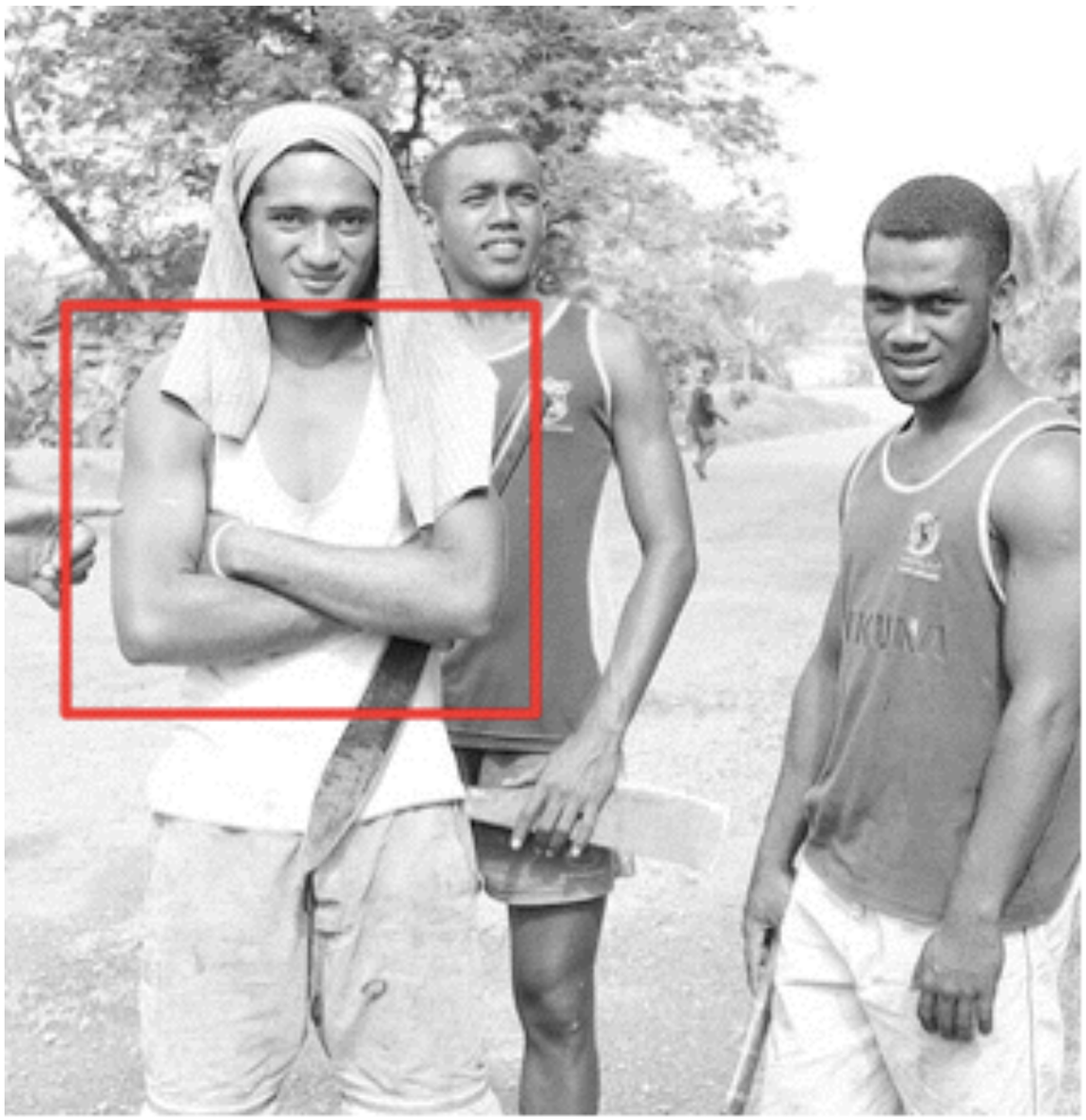
Examples of good parts

parts are often far visually, but they are close semantically

We call such parts **poselets**

Bourdev, Maji, Brox and Malik,
Detecting people using mutually consistent poselet activations, ECCV 2010

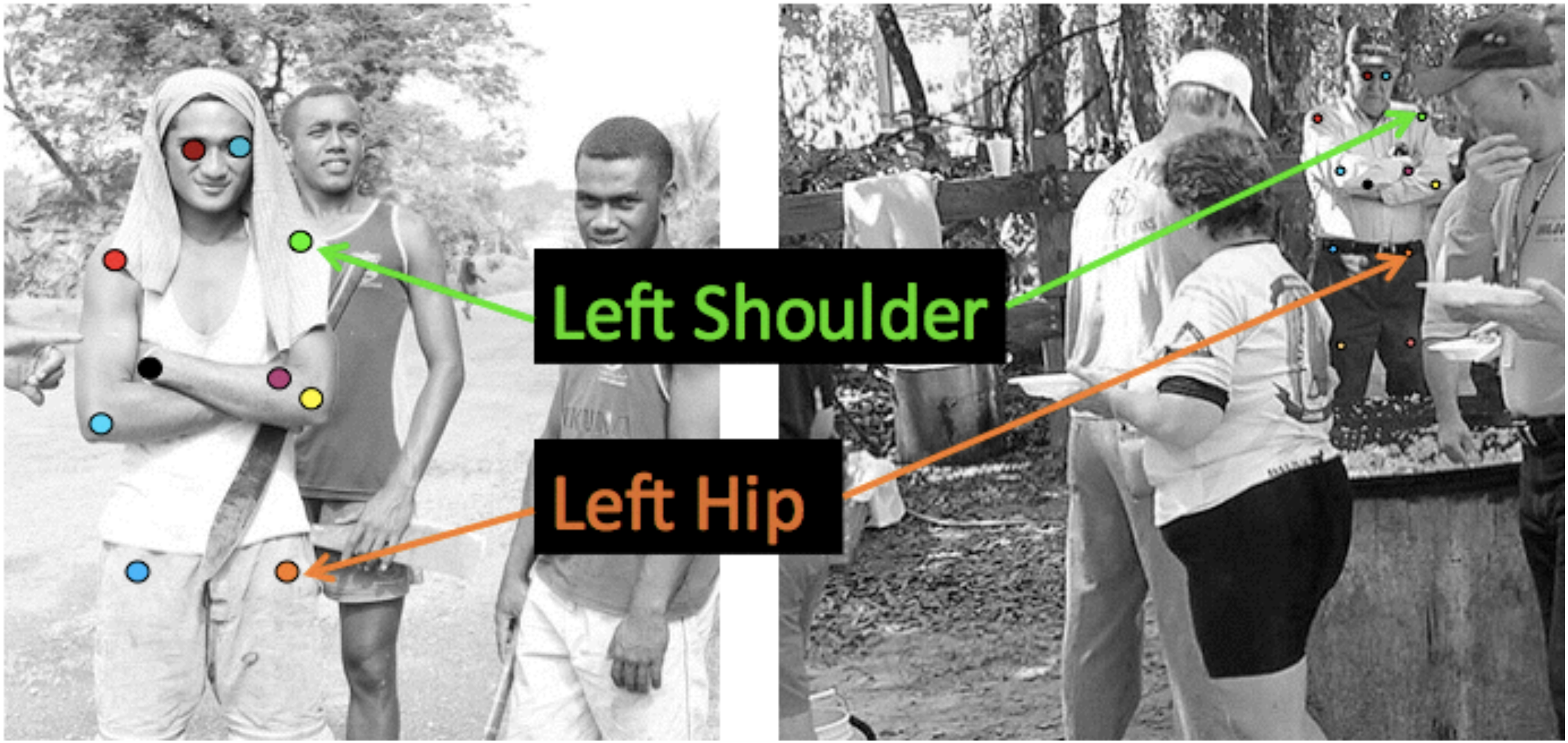
How to find *semantically* similar patches?



Given a part of the human pose

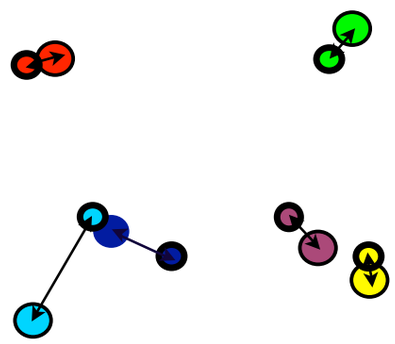
How do we find a similar pose configuration in another image?

How to find *semantically* similar patches?



We annotated the locations of various joints such as eyes, nose, shoulders and limbs for each training instances

How to find *semantically* similar patches?



residual error

Training poselet classifiers

residual error

0.15 0.20 0.15 0.85 0.35 0.10

- Given a source patch
- Find the closest patch in every other instance
- Sort them by residual error
- Threshold the list

Training poselet classifiers

Pos = {…       …}

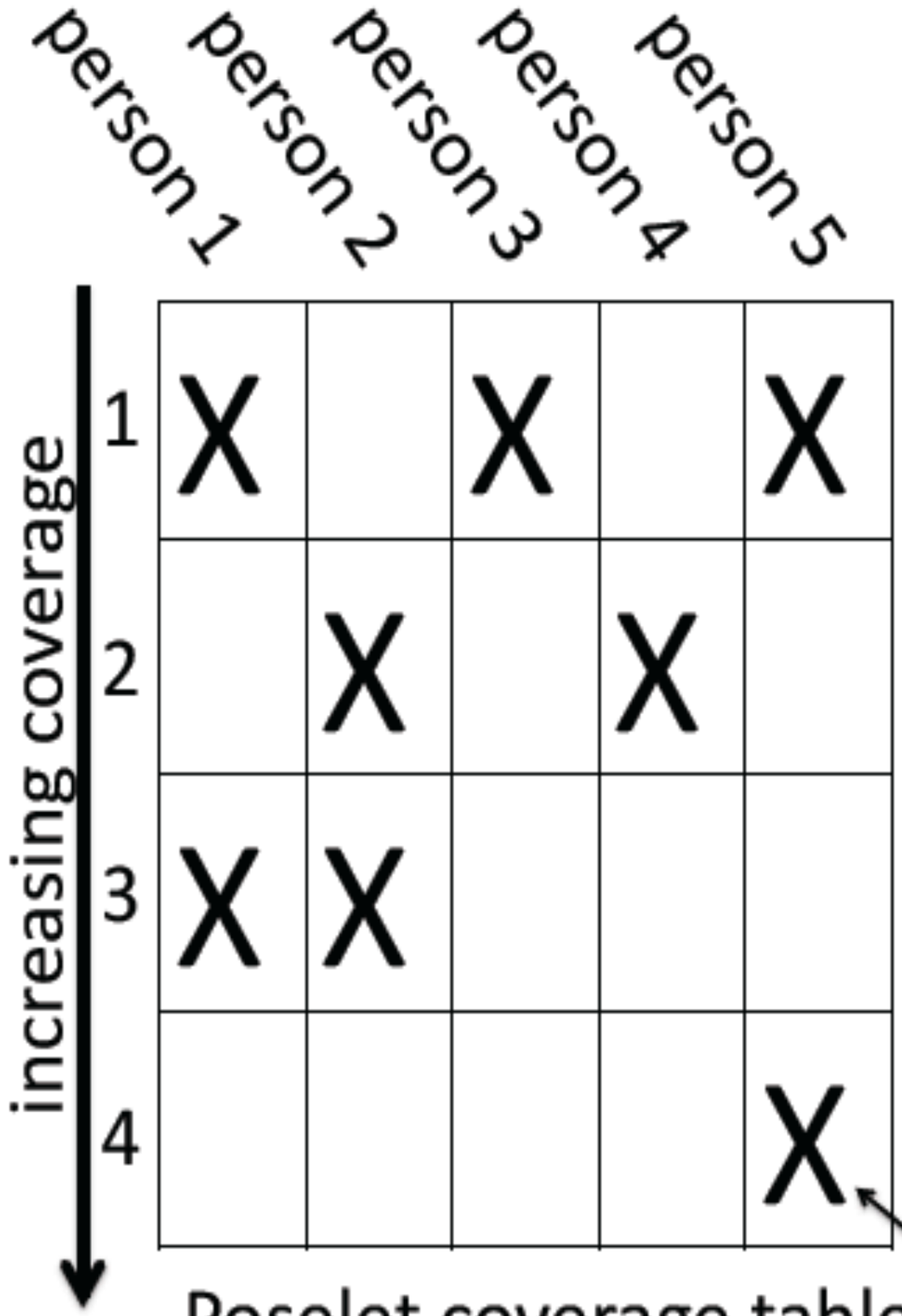
- Given a source patch
- Find the closest patch in every other instance
- Sort them by residual error
- Threshold the list
- Use these patches to train a standard Dalal & Triggs detector, i.e. HOG + linear SVMs with data mining

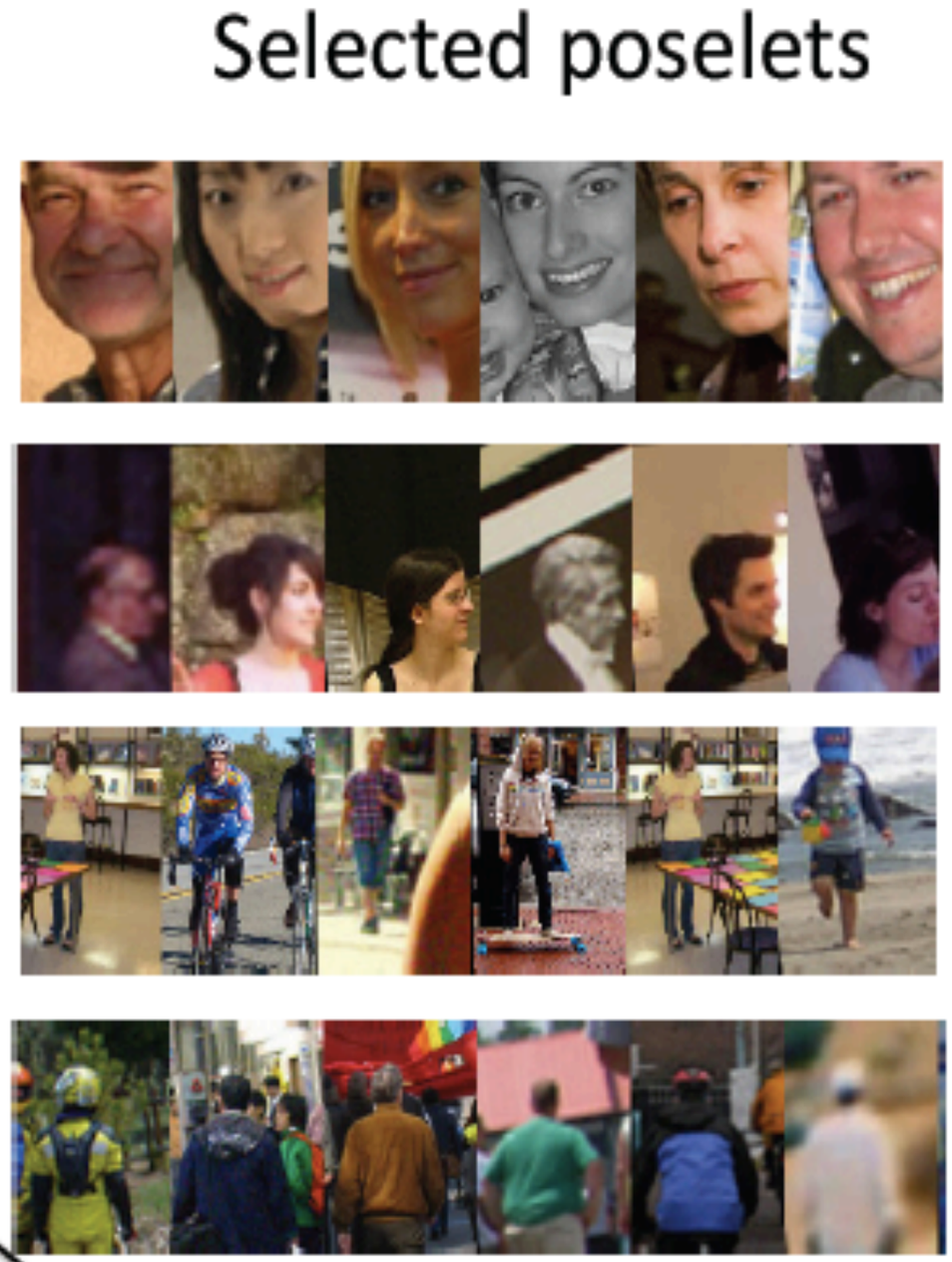
Which poselets should we train?

- Train a large number of poselets and select a subset
- Generate thousands of *random windows*, generate poselet candidates using the earlier method and train detectors using HOG + linear SVMs (Dalal & Triggs)

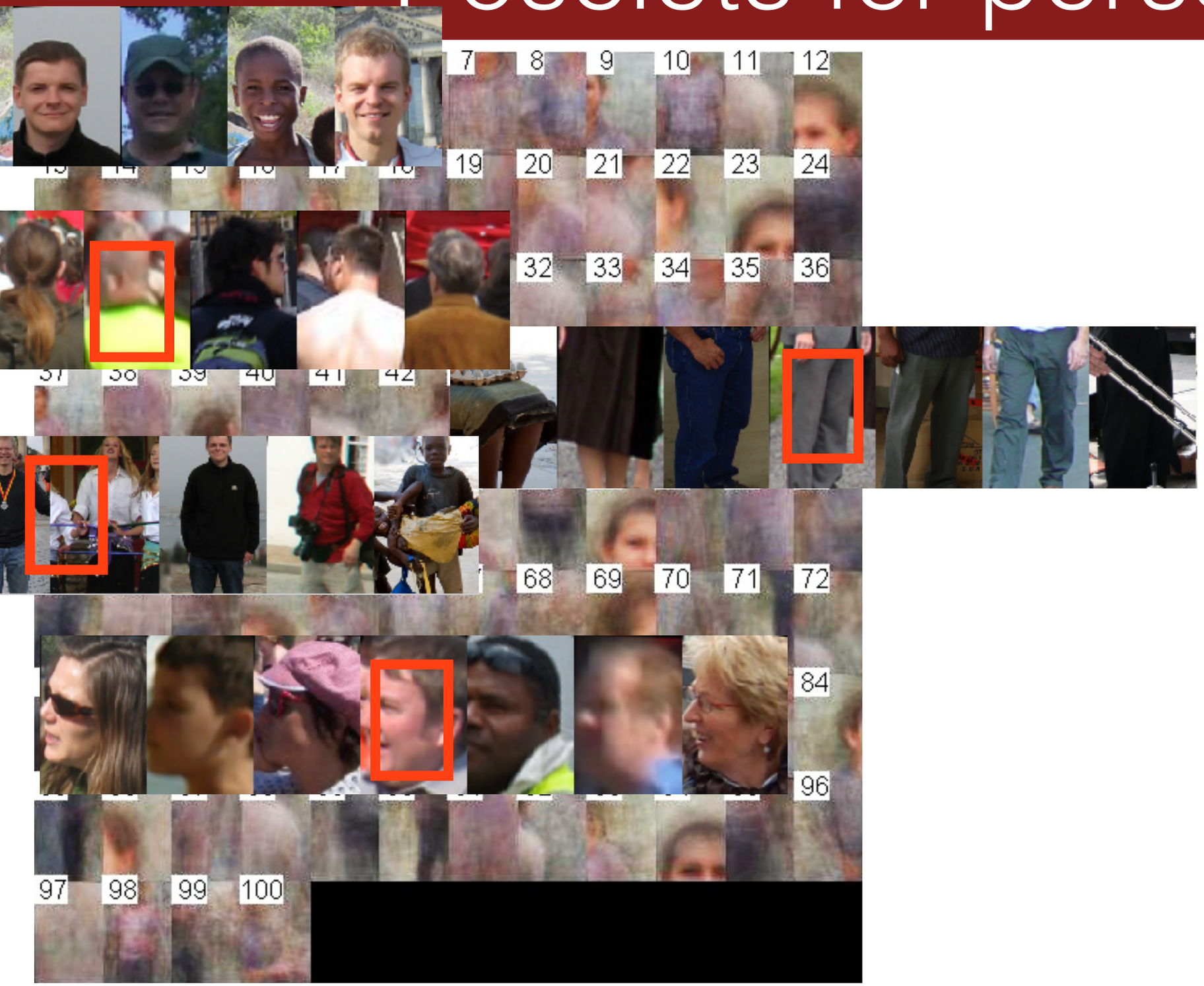
- Select a set of poselets that are
  - individually effective
  - complimentary

Selecting poselets for detection





Poselets for person



Bourdev et al. 10

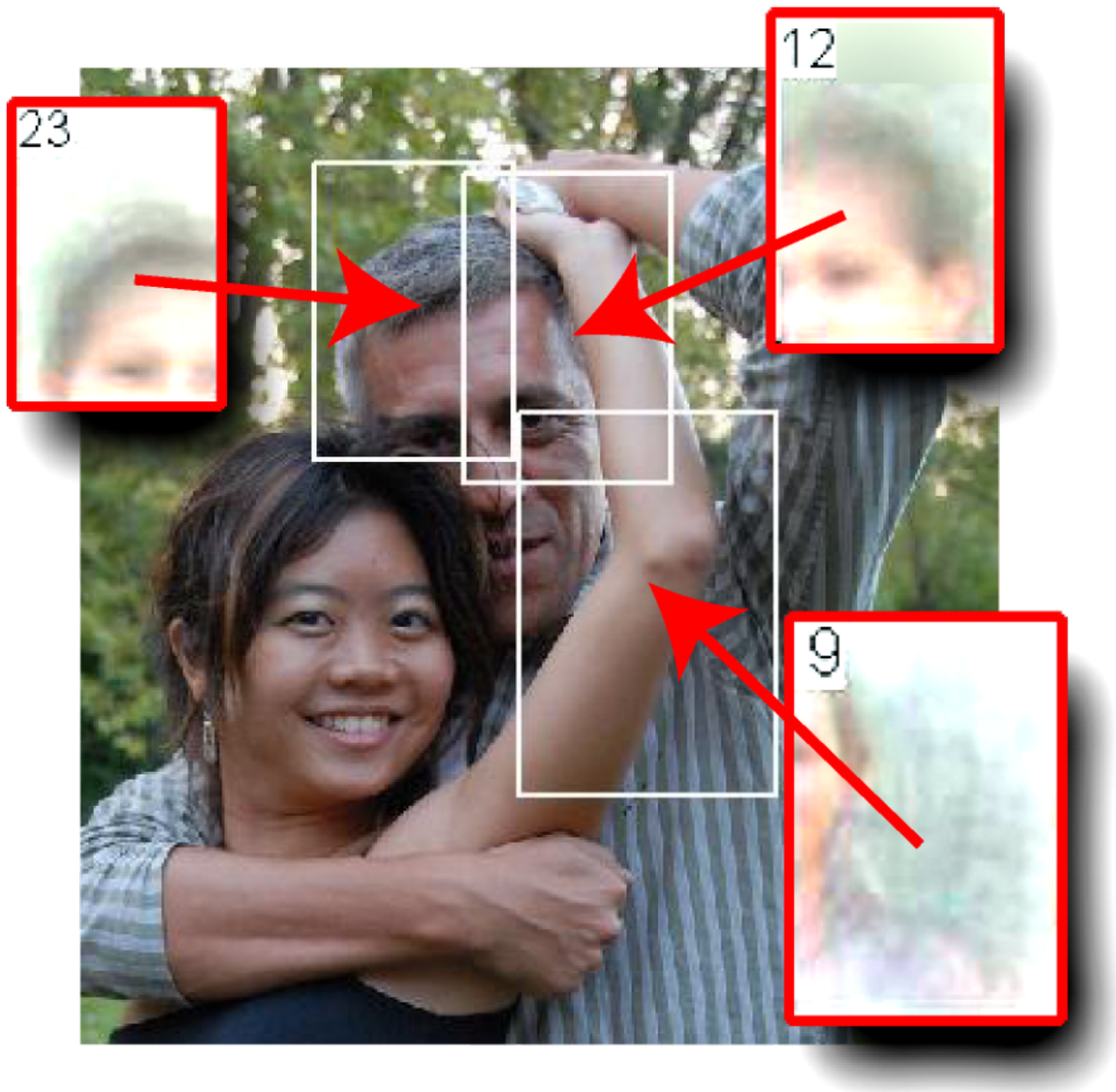
Person detection using poselets

- Example of a poselet

- Estimate relative bounding box on the training set

expected bounds

Person detection using poselets



- Detect each poselet in an image
- Vote for the person bounding box
- Find non-overlapping clusters
- Score each cluster using a weighted combination of poselet detection scores

$$s_i = \sum_{p \in C_i} w_p a_p$$

person detection score

weight of each poselet

poselet detection score

Bourdev & Malik 09, Bourdev et al.10, Maji & Malik 10

PASCAL VOC detection challenge

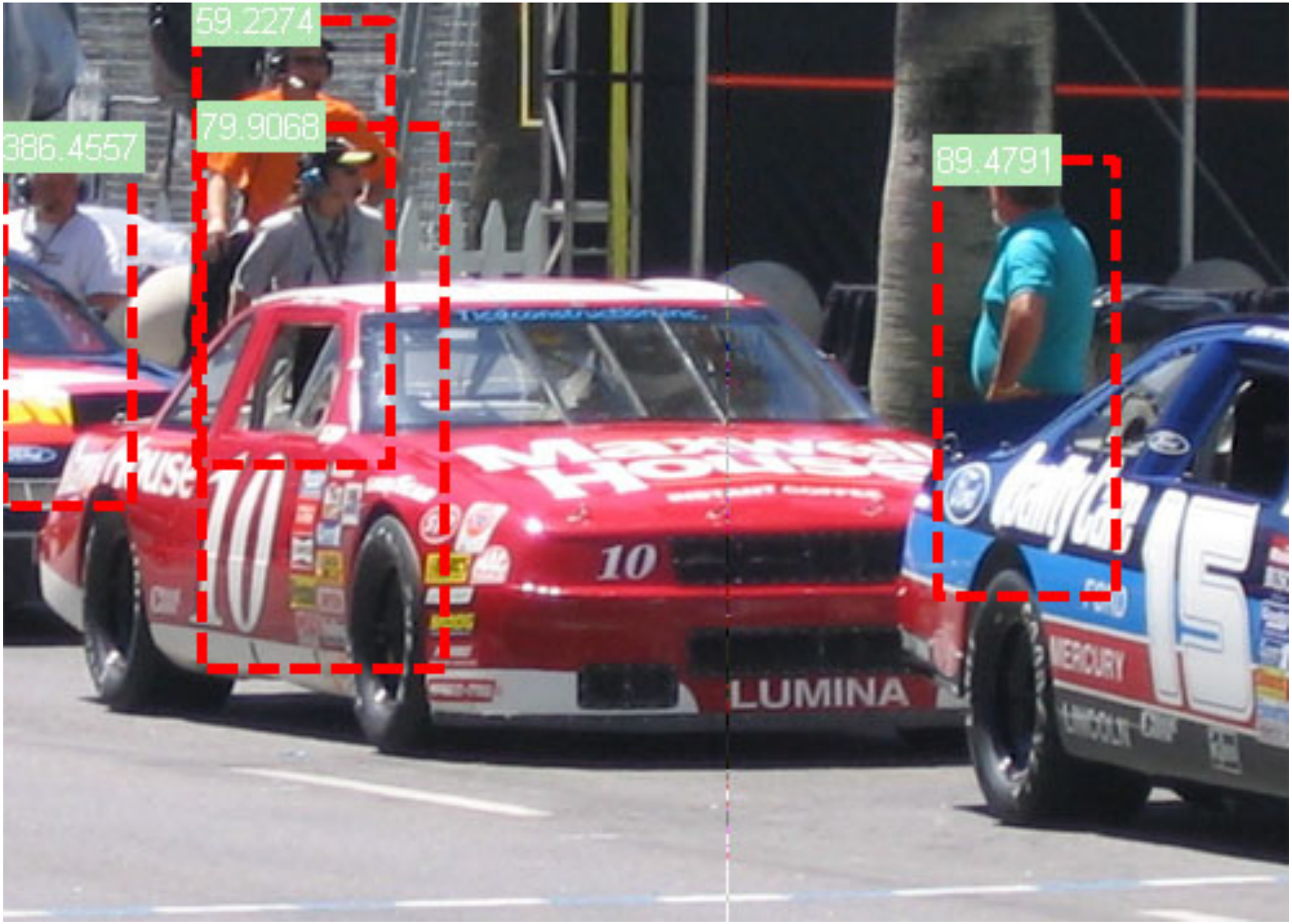
"person" category VOC 2010 test set

Method	Detection AP
Poselets	48.5%
Dalal & Triggs	12.0%

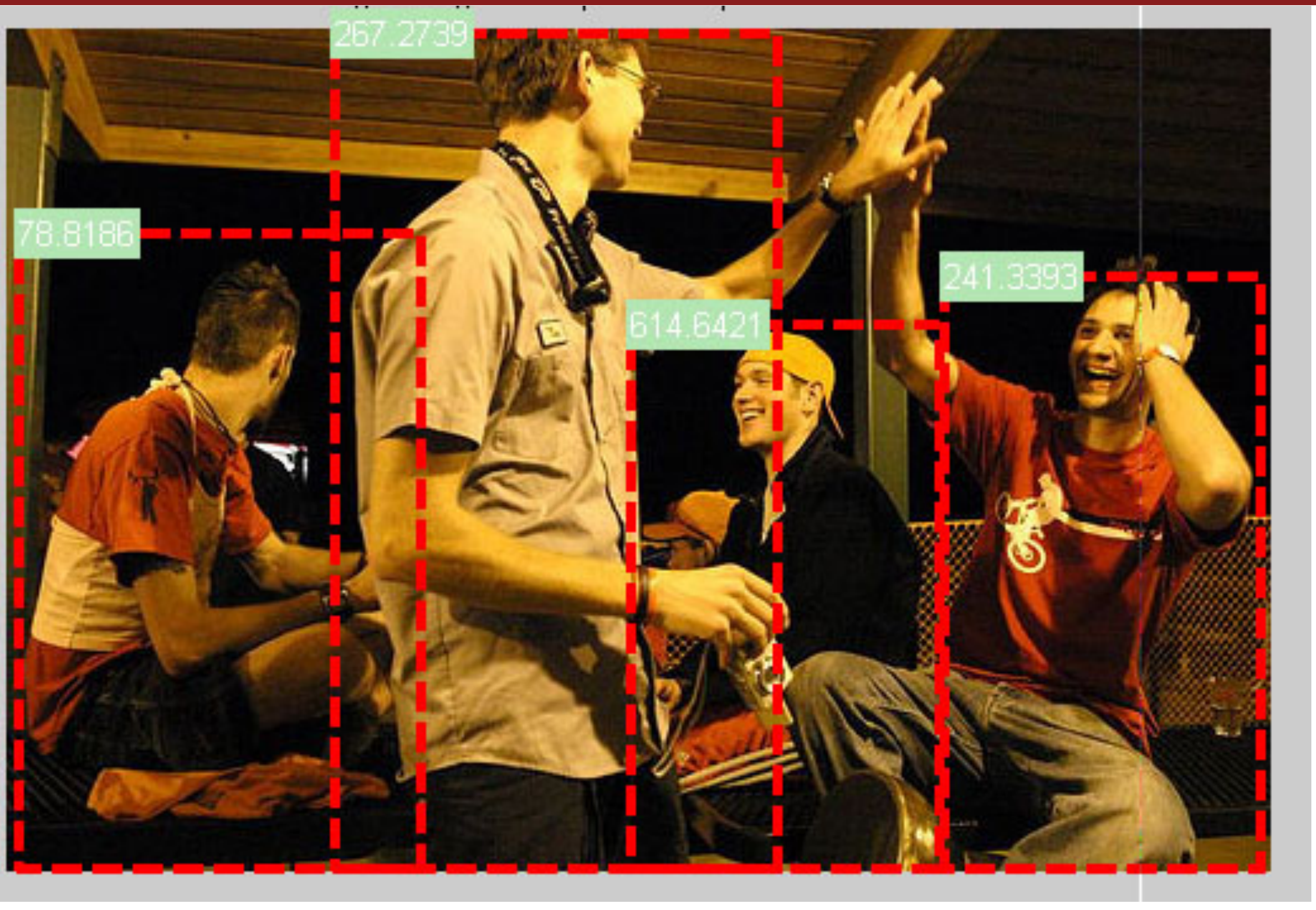
poselet detector — same features, 100x templates

L. Bourdev, S. Maji, T. Brox, J. Malik
Detecting people using mutually consistent poselet activations, ECCV 2010
http://www.cs.berkeley.edu/~lbourdev/poselets/

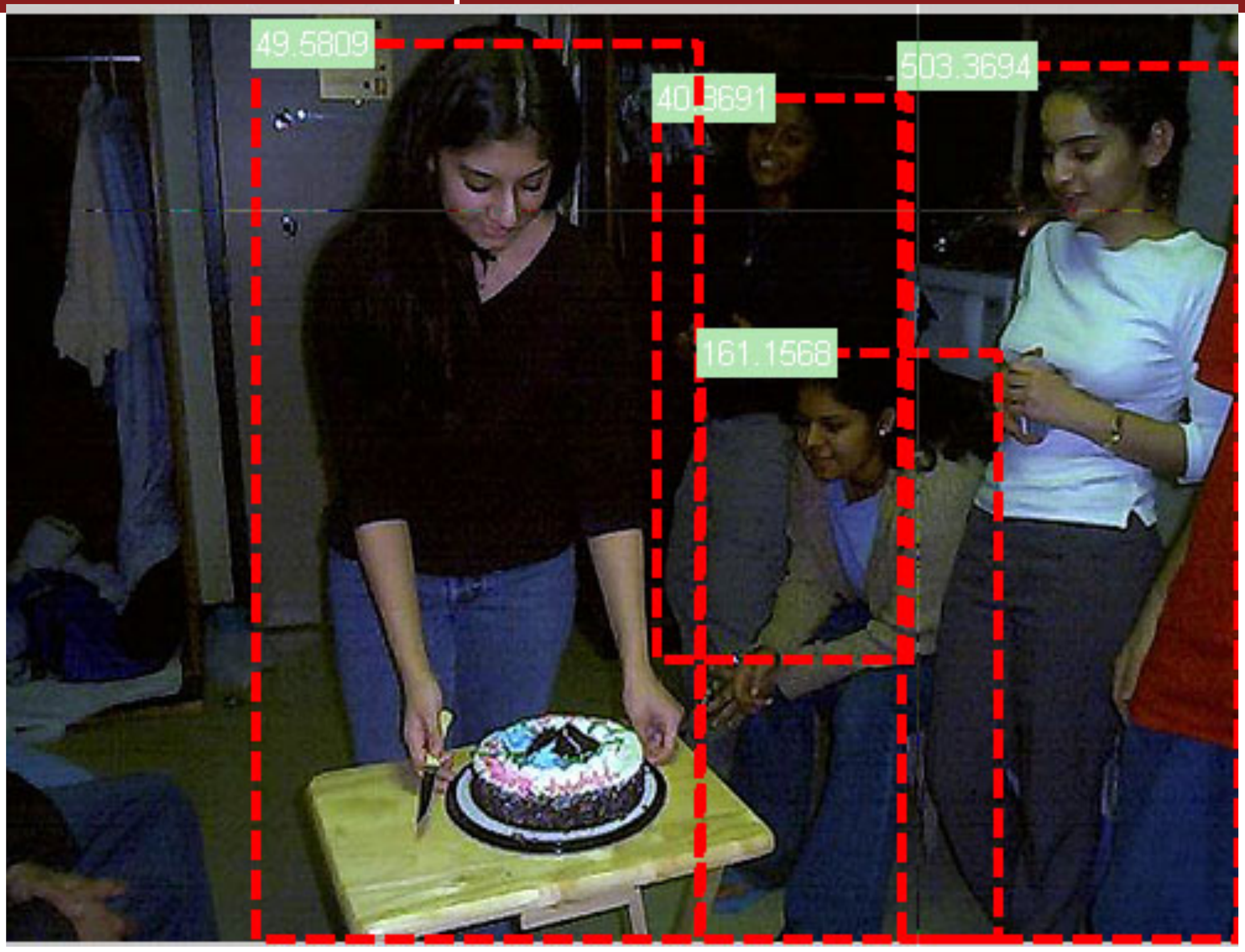
Example detections



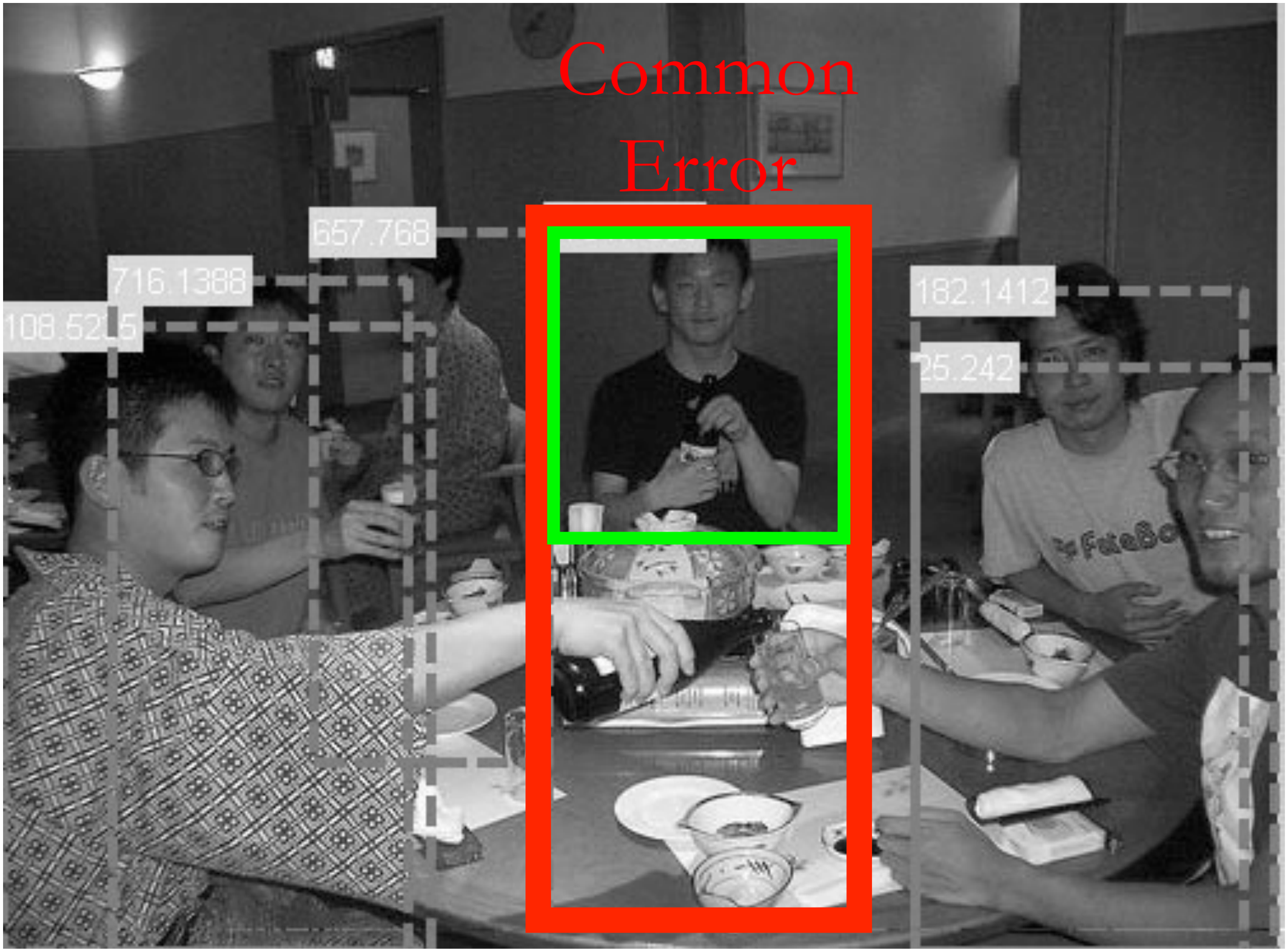
Example detections



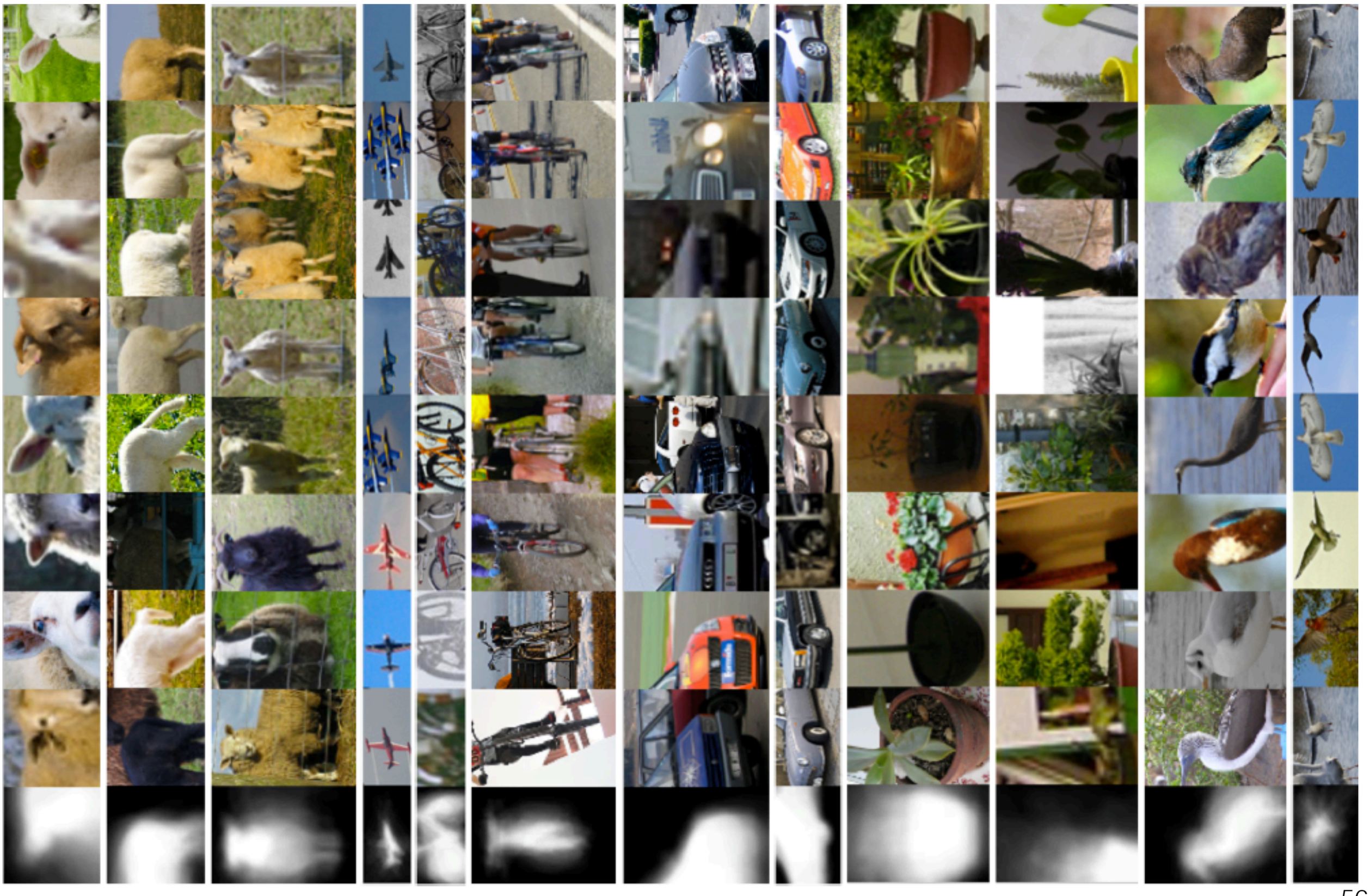
Example detections



Example detections



Poselets for other categories

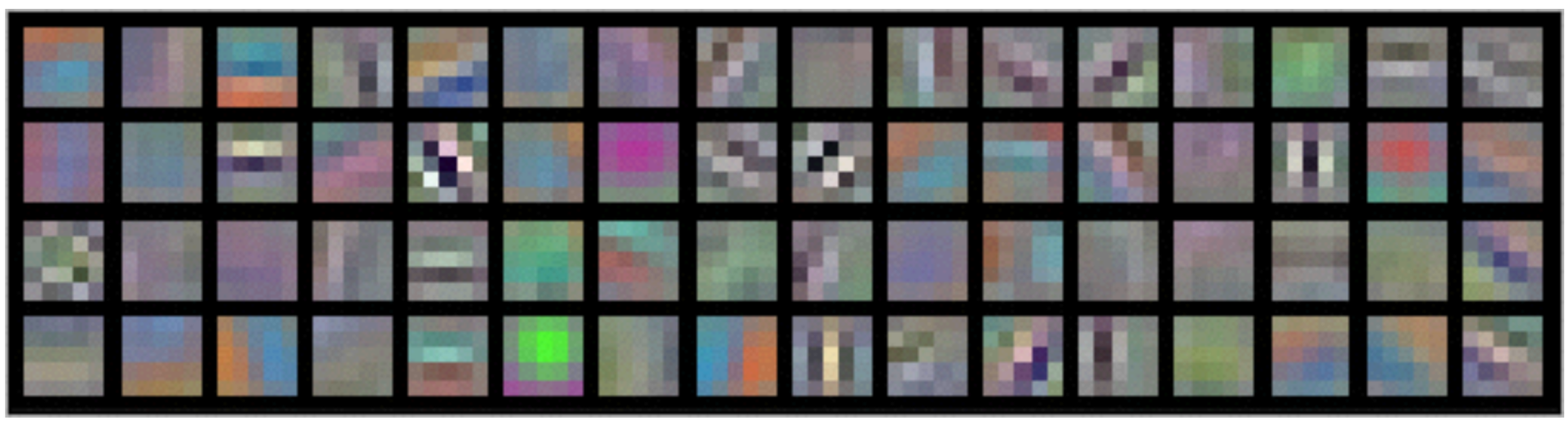


Conclusion

- Detection as multi-scale template matching
  - However to model complex categories such a “person” we need many templates
- Key challenges:
  - What features to use — HOG is a simple one, but not the most accurate (we are throwing away a lot of information). Current state of the art methods use complex features.
  - How to learn templates?
    - poselets used extra annotations, but these are expensive to collect
    - unsupervised methods — deformable part models, Felzenszwalb et al.
  - Efficiency is an issue — there are many tens of thousands of classifications per image

Further thoughts and readings …

- There are many approaches for detection. Here are some seminal ones you can learn about
  - Viola and Jones face detector — still widely in use
  - HOG + linear SVM pedestrian detector — Dalal & Triggs
  - Deformable part-based models, Felzenszwalb et al.
  - Poselets and their applications — Bourdev et al.
- Current state of the art — deep convolutional neural network features as a replacement of HOG



CNN filterbank (Krishevsky et al. 2010)