Computer Vision

Bogdan Alexe

bogdan.alexe@fmi.unibuc.ro

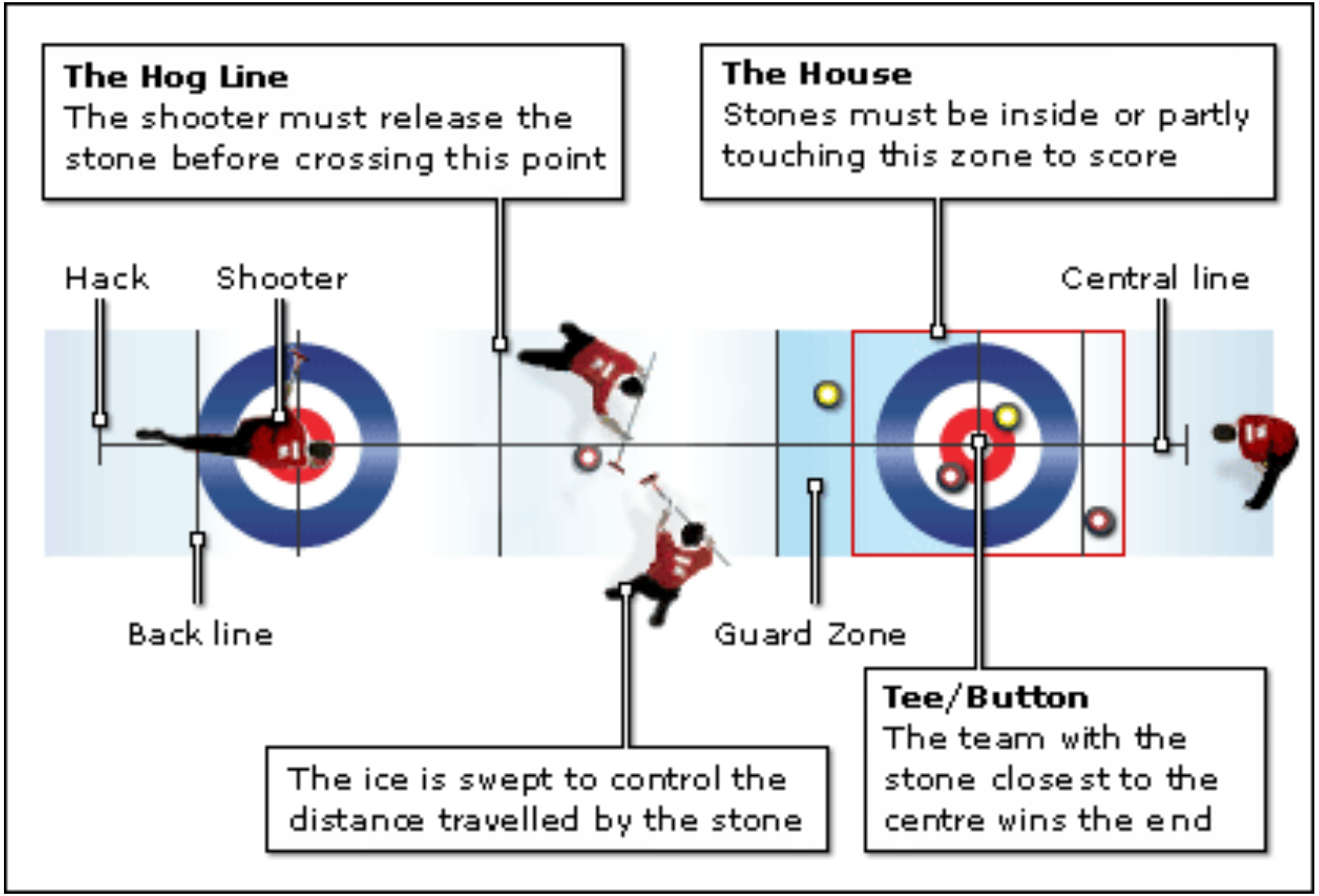
University of Bucharest, 2nd semester, 2020-2021

Project 2 – presentation

tinyurl.com/CV-2021-Project2

Project 2 – Video analysis of footages from the sport of curling

Project 2 – Video analysis of footages from the sport of curling



Task 1 – count the stones in a frame

- the task is to count number of stones in an image and specify how many are red and how many are yellow

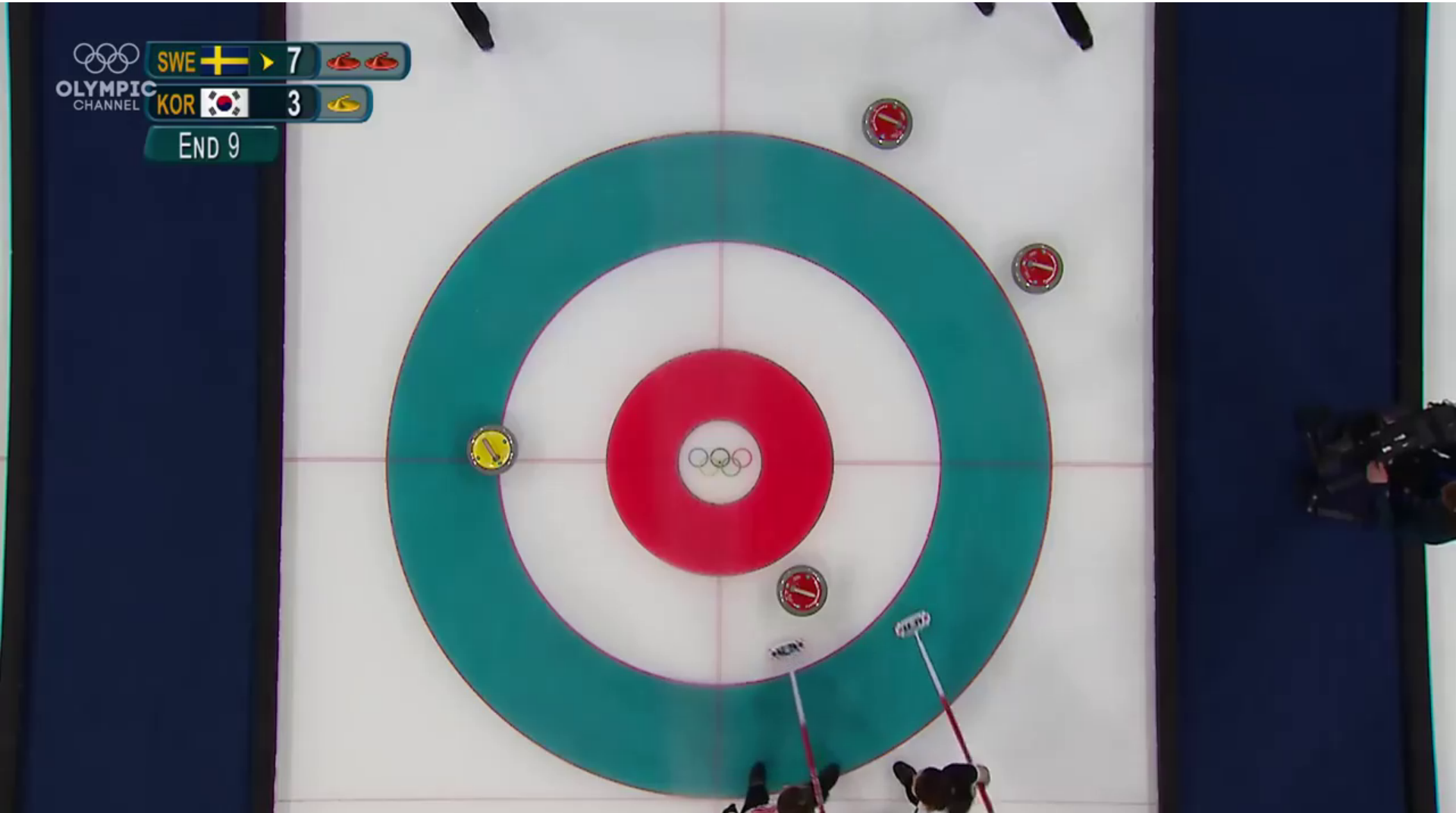
- you are given 25 images for training (with annotated ground-truth)

- at test time you have to do the prediction on 25 test images

- 25 * 0.06 points/image = 1.5 points

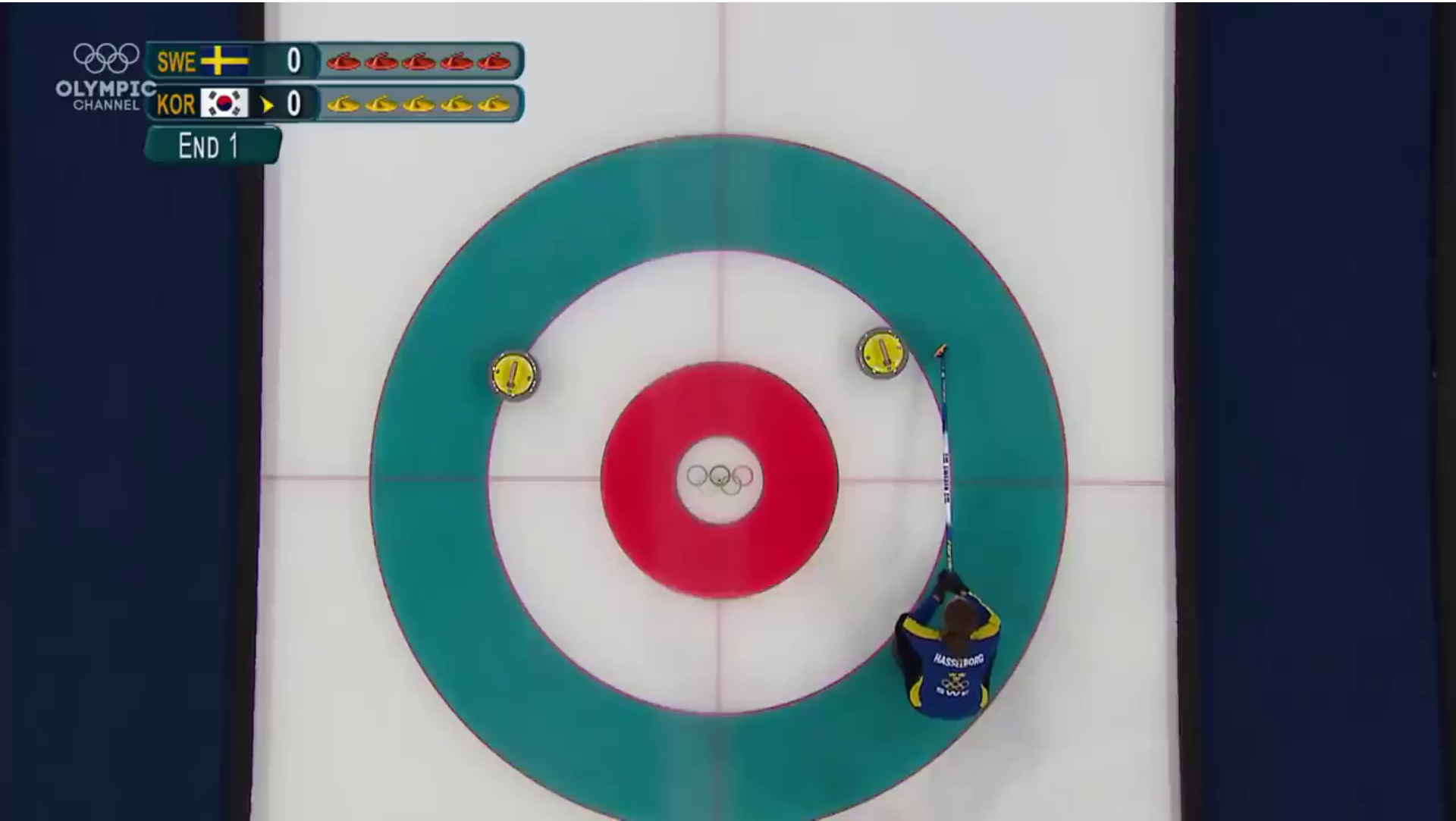
Task 1 – count the stones in a frame

Training example 1: 4 stones (3 red + 1 yellow)



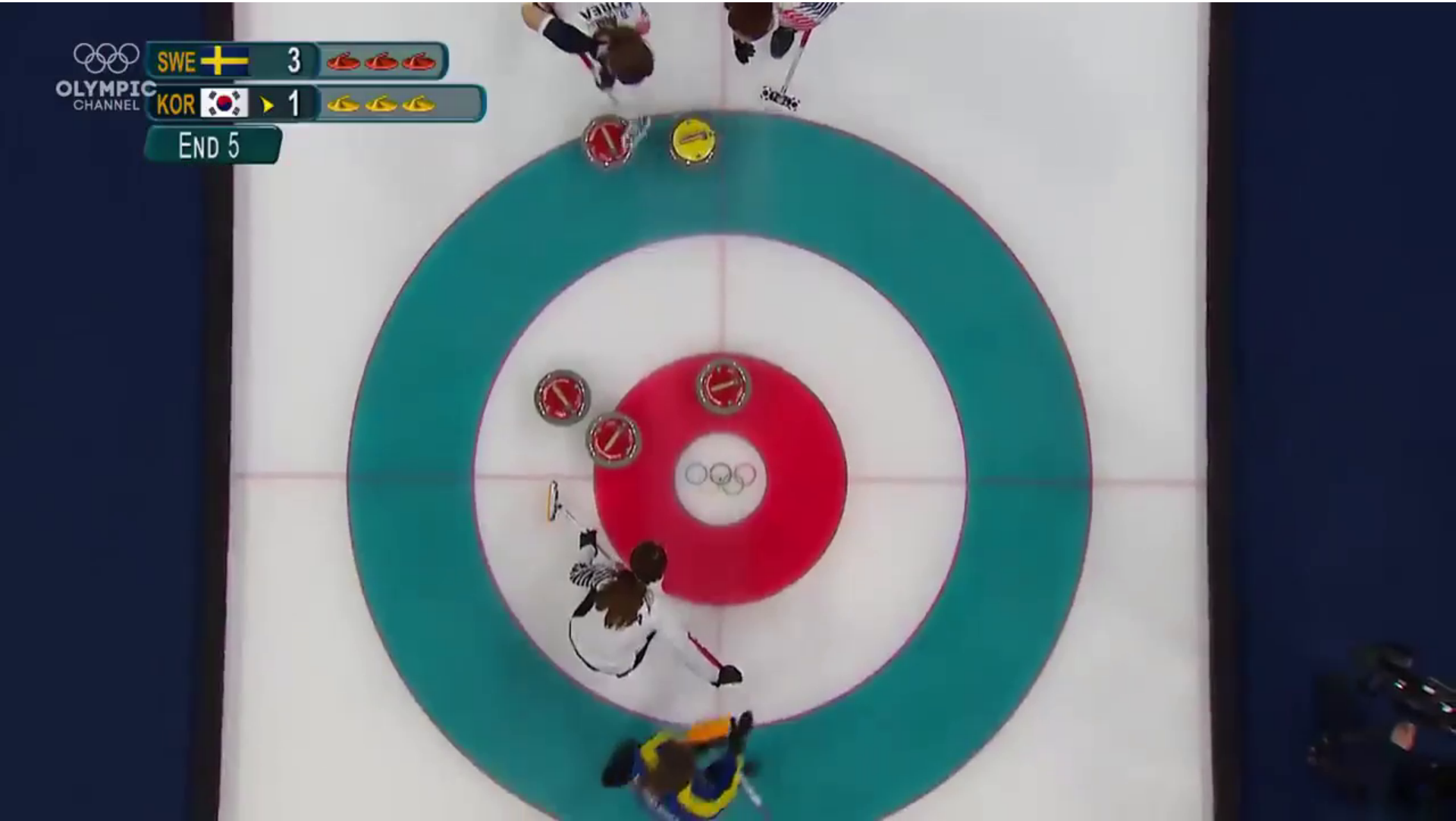
Task 1 – count the stones in a frame

Training example 2: 2 stones (0 red + 2 yellow)



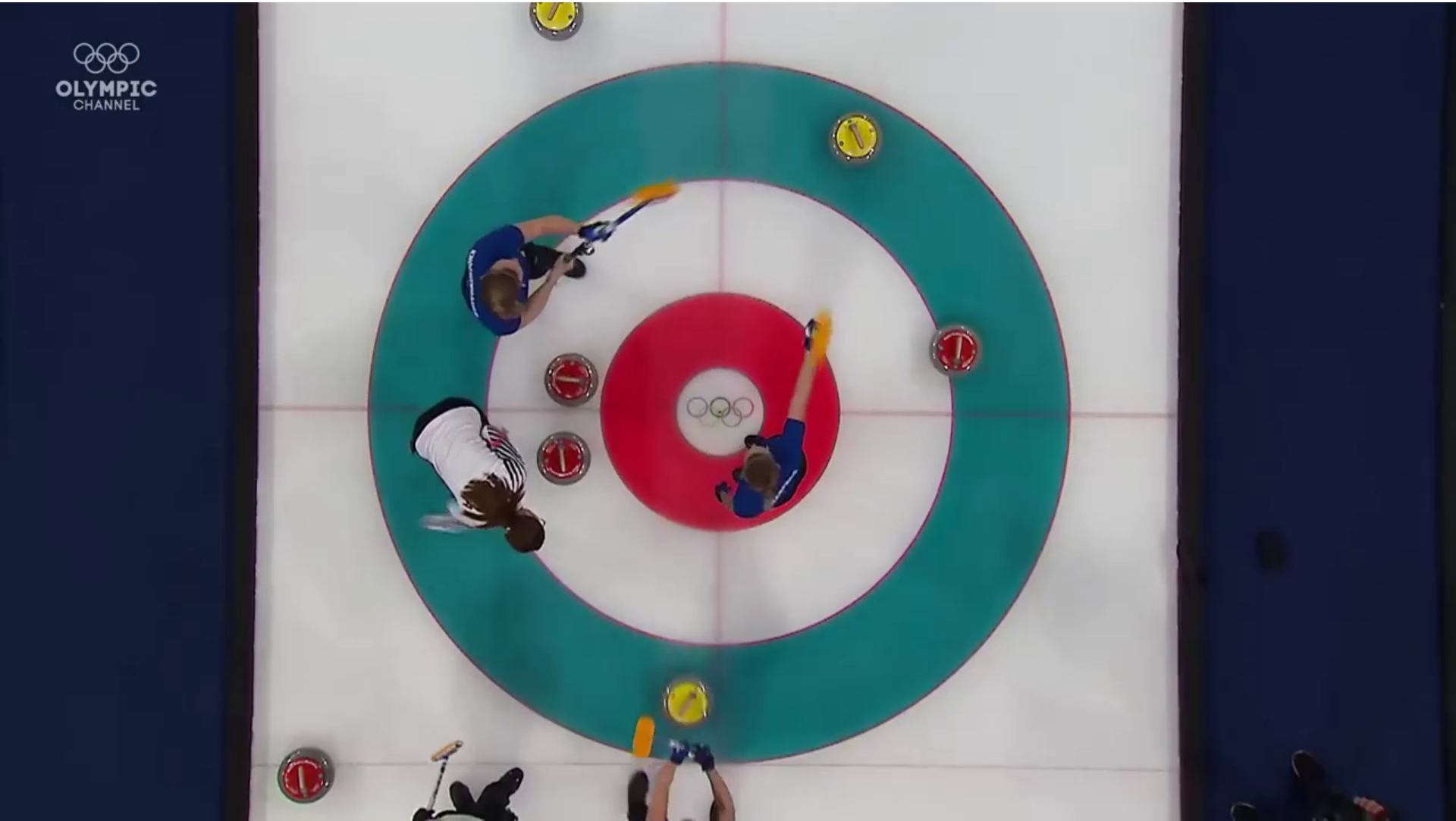
Task 1 – count the stones in a frame

Training example 4: 5 stones (4 red + 1 yellow)



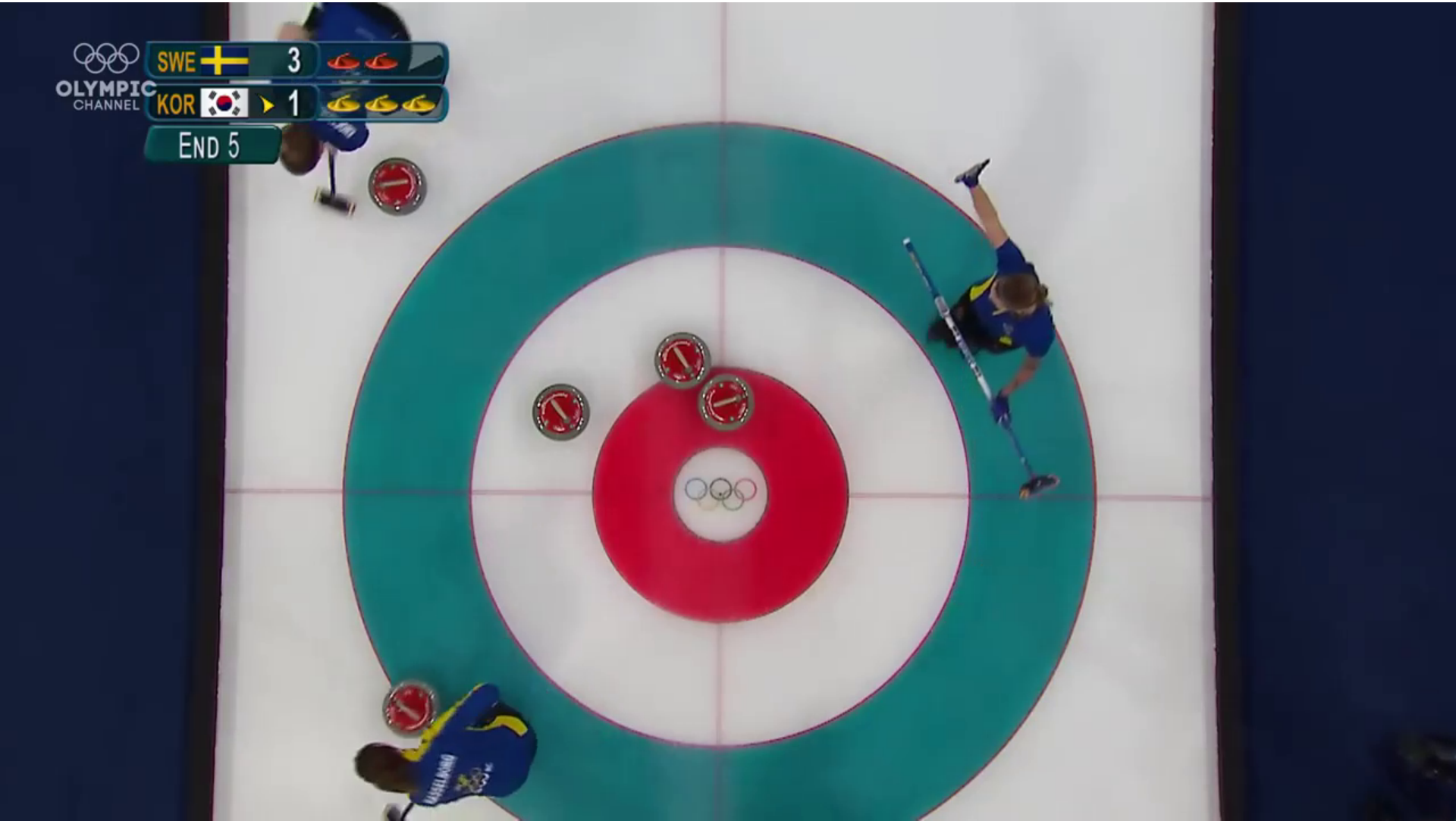
Task 1 – count the stones in a frame

Training example 15: 7 stones (4 red + 3 yellow)



Task 1 – count the stones in a frame

Training example 22: 5 stones (5 red + 0 yellow)



Task 2 – infer the score after a play

- the task is to infer the score after a play (what is the score at the end of the video)
- you are given 15 videos for training (with annotated ground-truth)
- at test time you have to do the prediction on 15 test images
- 15*0.1 points/video = 1.5 points

Task 2 – infer the score after a play

- the task is to infer the score after a play (what is the score at the end of the video)

- how to compute the score:
  - determine how many red and yellow stones are in the House (green circle). If there is no stone in the House the score is 0-0
  - only one color makes points, the one having the closest stones to the center (number of points = # of this stones)

Task 2 – infer the score after a play

Training example 1: 0 red vs 1 yellow

Task 2 – infer the score after a play

Training example 10: 3 red vs 0 yellow

Task 2 – infer the score after a play

Training example 14: 1 red vs 0 yellow

Task 3 – track a released stone

- the task is to track the released stone (you are given the initial bounding box in the first frame)

- the scene is constrained, the ice surface is seen always from above

- compute at each frame the detection, compare it for evaluation with the ground truth

Task 3 – track a released stone

- the task is to track the released stone (you are given the initial bounding box in the first frame)
- you are given 15 videos for training (with annotated ground-truth)
- at test time you have to do the prediction on 15 test images
- 15*0.1 points/video = 1.5 points

Task 3 – track a released stone

Training example 3

Task 4 – track a released stone (unconstrained)

- the task is to track the released stone (you are not given anything)

- the scene is unconstrained, the ice surface could be filmed from different viewpoints

- compute at each frame the detection, compare it for evaluation with the ground truth

Task 4 – track a released stone (unconstrained)

- the task is to track the released stone (you are not given anything)

- you are given 10 videos for training (with annotated ground-truth)

- at test time you have to do the prediction on 10 test images

- 10*0.1 points/video = 1 points

Task 4 – track a released stone (unconstrained)

Project 2 details

- all data will be released this week here: **tinyurl.com/CV-2021-Project2**

- read carefully the pdf (format, etc)

- deadline:
  - Saturday 26th of June (code submission)
  - Sunday 27th of June (results submission)

Course structure

1. **Features and filters: low-level vision**

   Linear filters, color, texture, edge detection

2. **Grouping and fitting: mid-level vision**

   Fitting curves and lines, robust fitting, RANSAC, Hough transform, segmentation

3. **Multiple views**

   Local invariant feature and description, epipolar geometry and stereo, object instance recognition

4. **Object Recognition: high – level vision**

   Object classification, object detection, part based models, bovw models

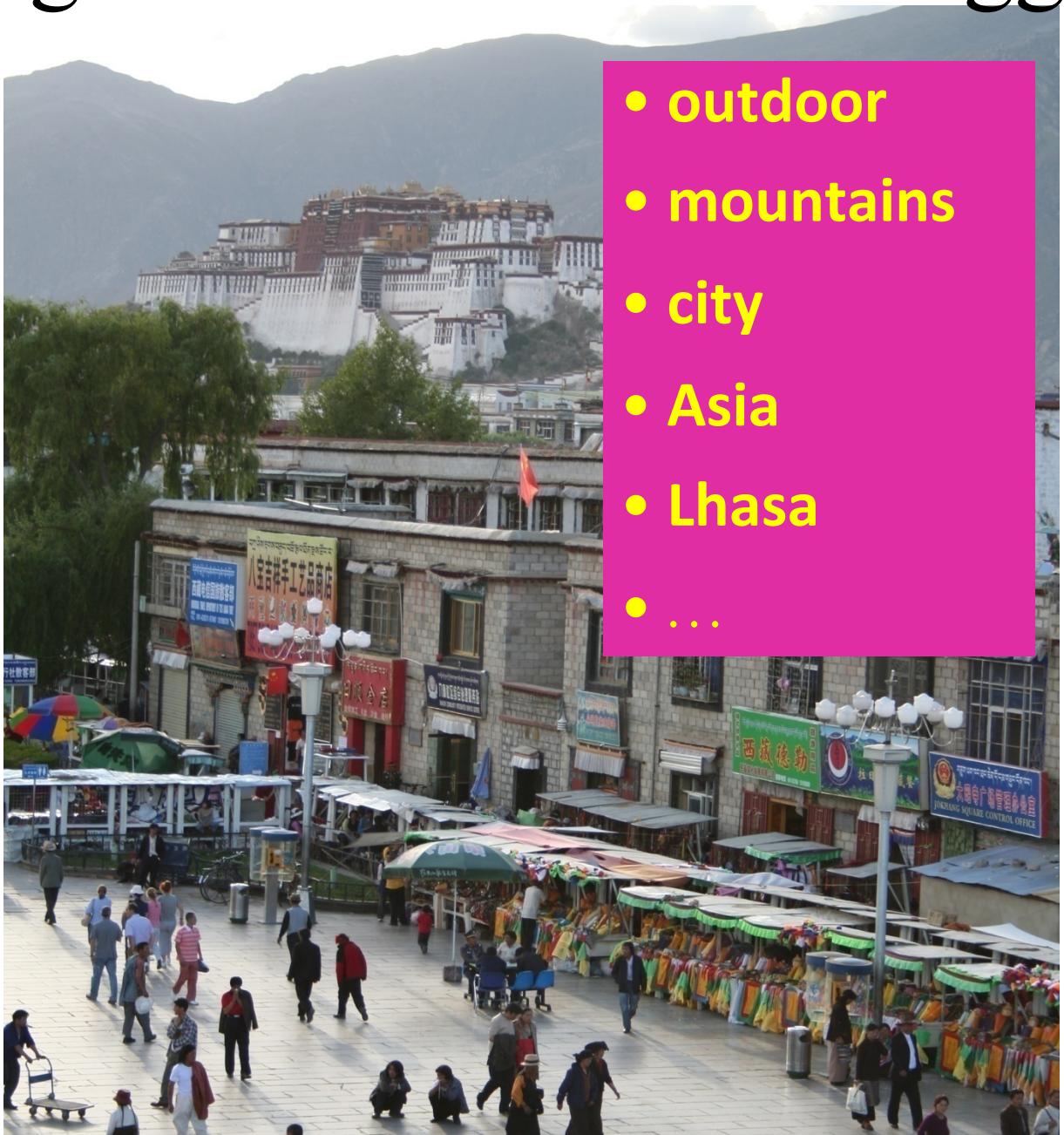
5. **Video understanding**

   Object tracking, background subtraction, motion descriptors, optical flow

Common recognition tasks

Slide credit
Fei-Fei Li

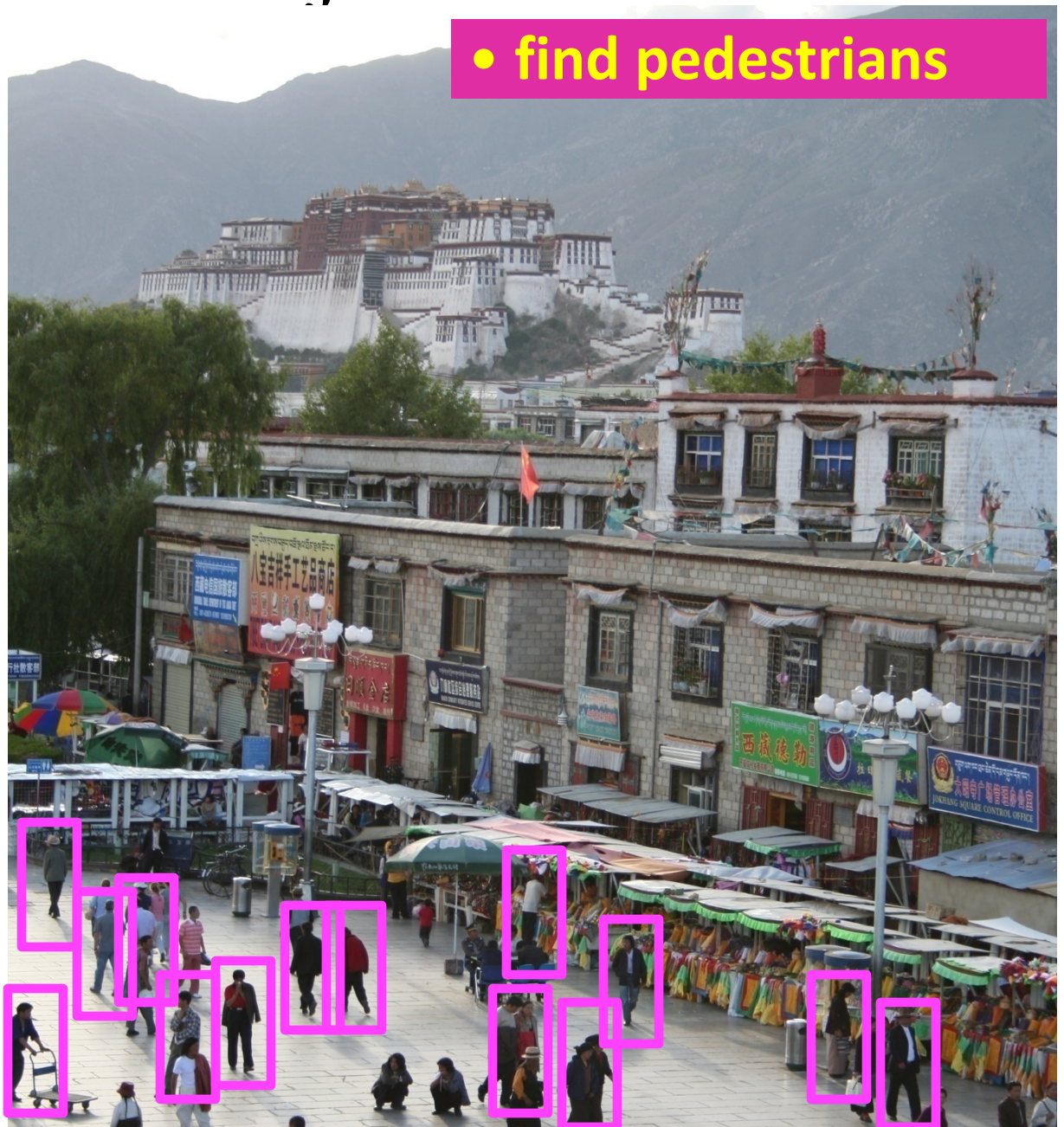
Image classification and tagging



- outdoor
- mountains
- city
- Asia
- Lhasa
- …

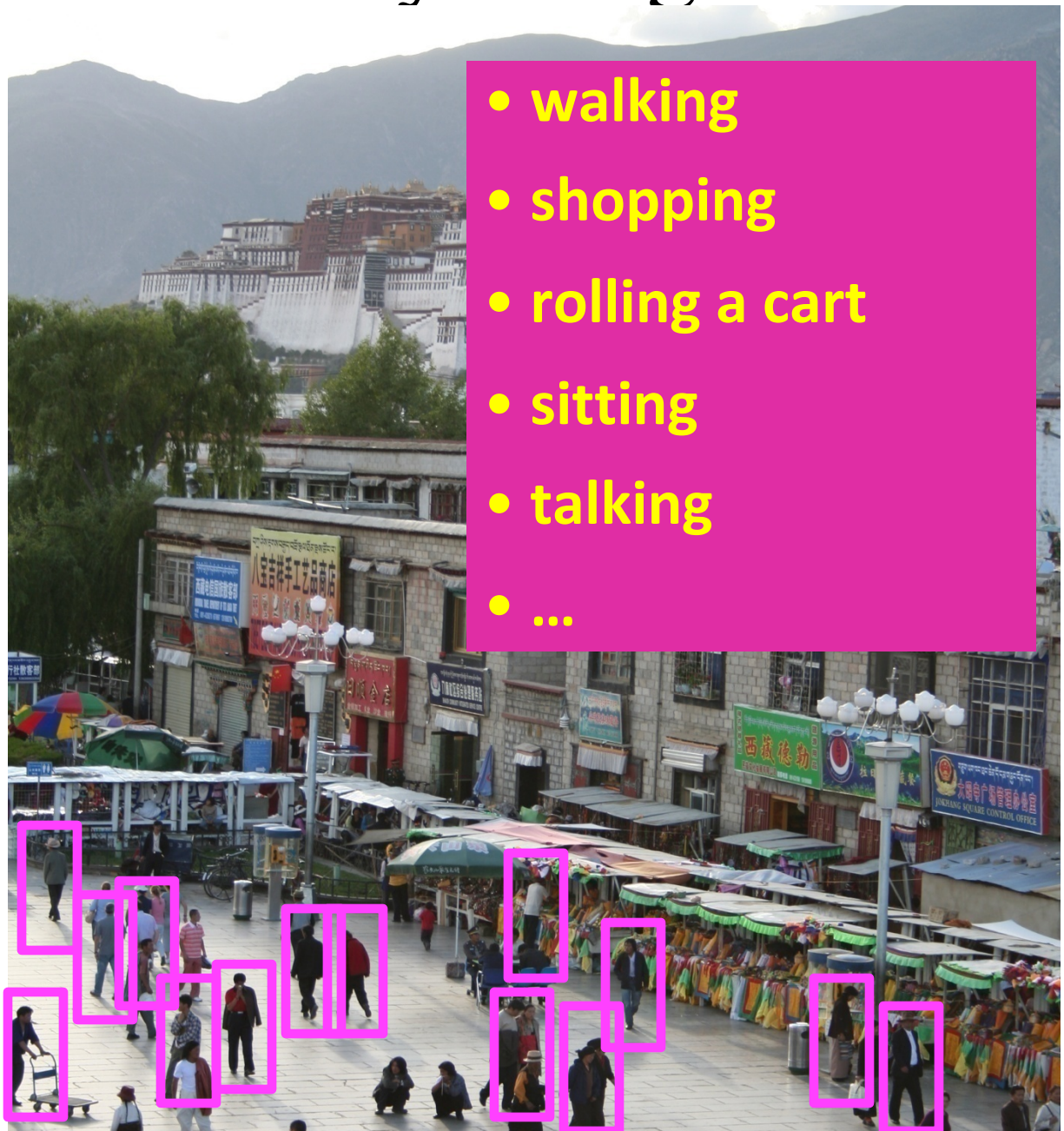
Slide credit
Fei-Fei Li

Object detection



Slide credit
Fei-Fei Li

Activity recognition



Slide credit
Fei-Fei Li

Semantic segmentation

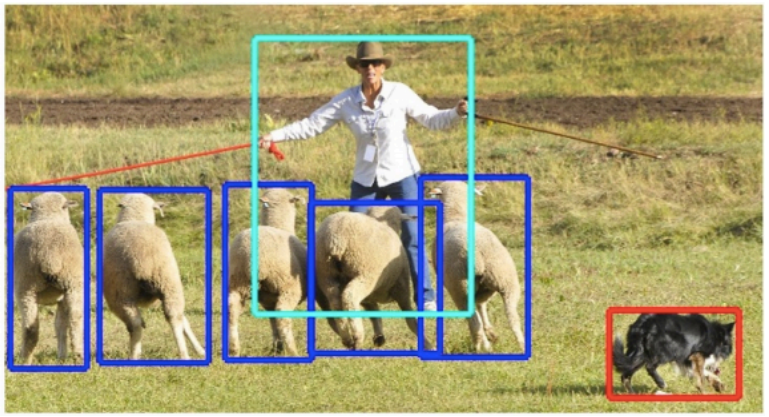
Slide credit
Fei-Fei Li

Semantic segmentation

Slide credit
Fei-Fei Li

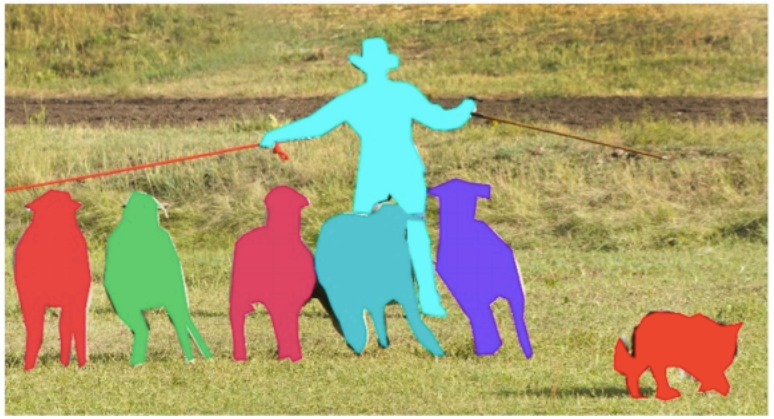
Classification, detection, semantic segmentation, instance segmentation

image classification



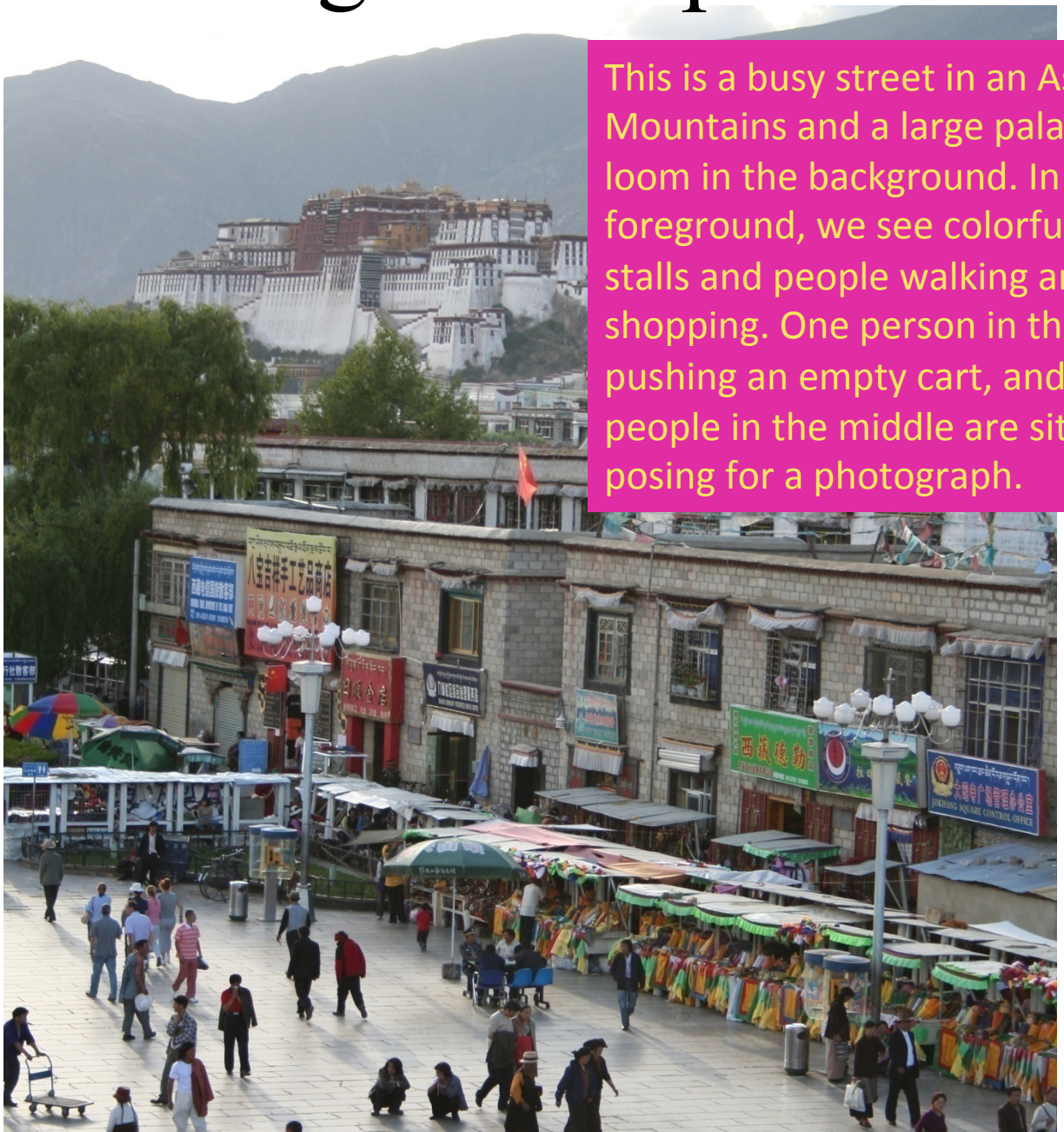
object detection

semantic segmentation



instance segmentation

Image description



This is a busy street in an Asian city. Mountains and a large palace or fortress loom in the background. In the foreground, we see colorful souvenir stalls and people walking around and shopping. One person in the lower left is pushing an empty cart, and a couple of people in the middle are sitting, possibly posing for a photograph.

Slide credit
Fei-Fei Li

Image classification

The statistical learning framework

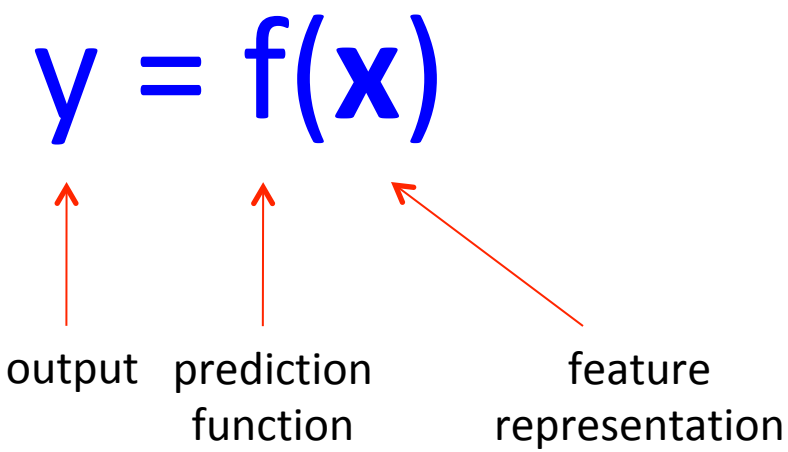
- Apply a prediction function to a feature representation of the image to get the desired output:

f() = “apple”

f() = “tomato”

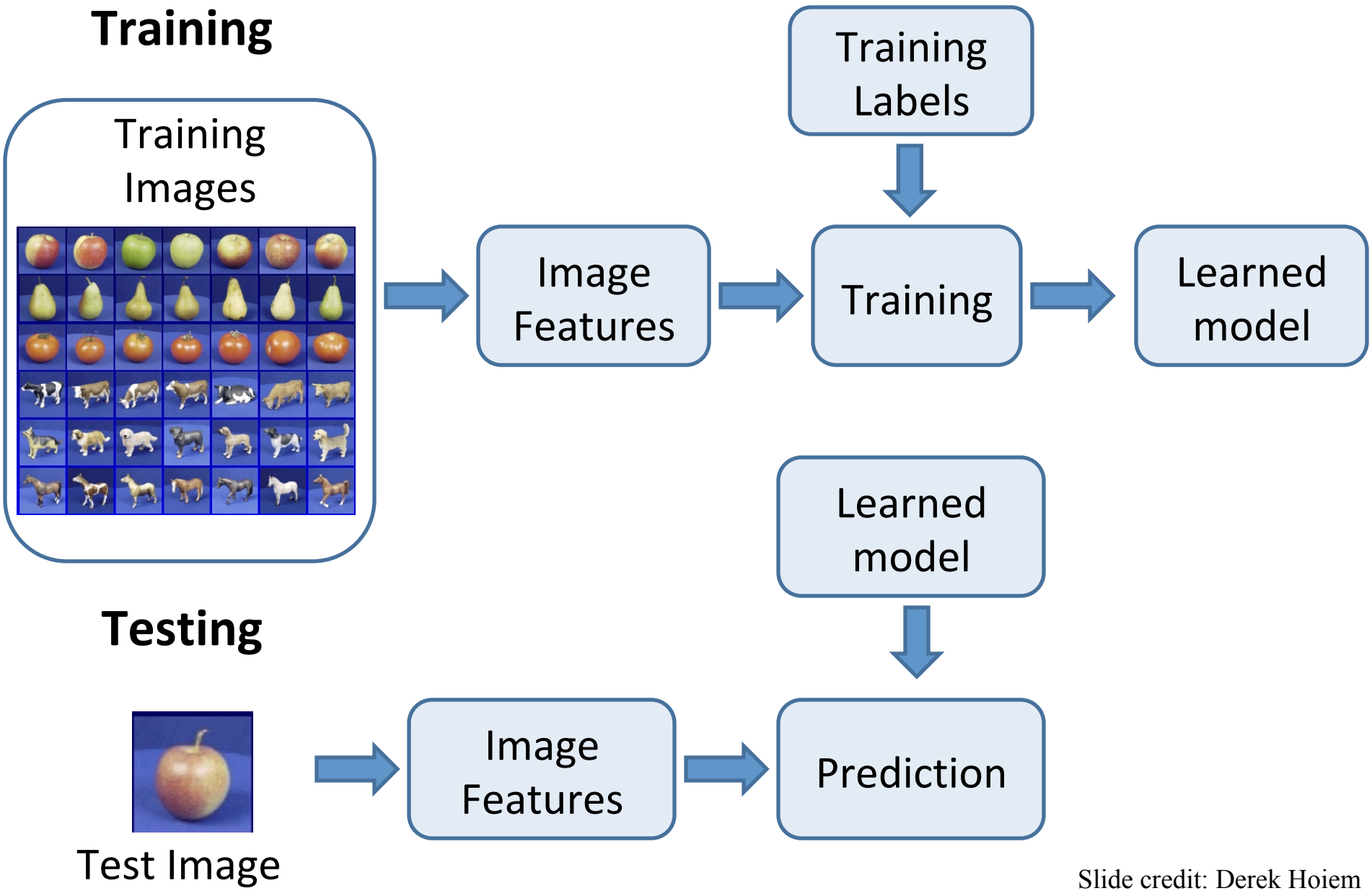
f() = “cow”

The statistical learning framework



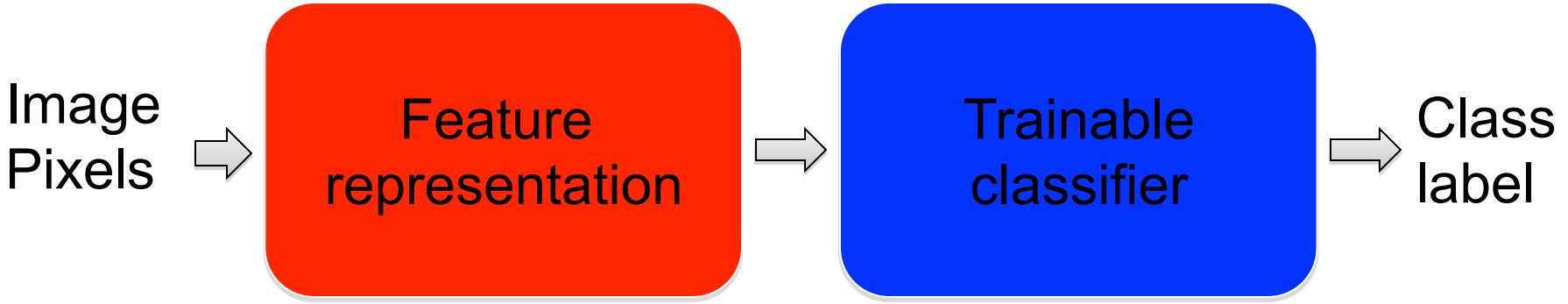
- **Training:** given a *training set* of labeled examples $\{(\mathbf{x}_1,y_1), \ldots, (\mathbf{x}_N,y_N)\}$, estimate the prediction function f by minimizing the prediction error on the training set
- **Testing:** apply f to a never before seen *test example* $\mathbf{x}$ and output the predicted value $y = f(\mathbf{x})$

Steps



Slide credit: Derek Hoiem

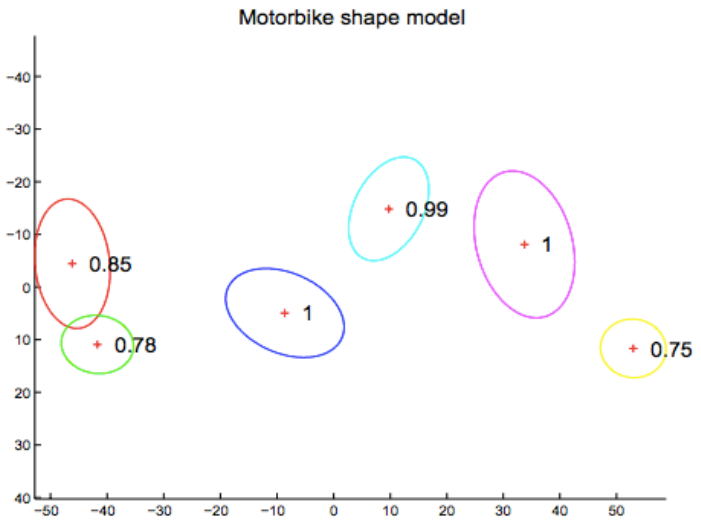
“Classic” recognition pipeline

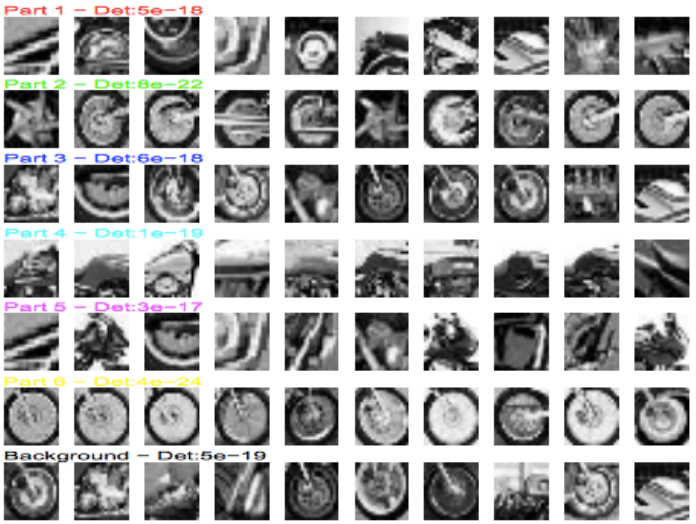


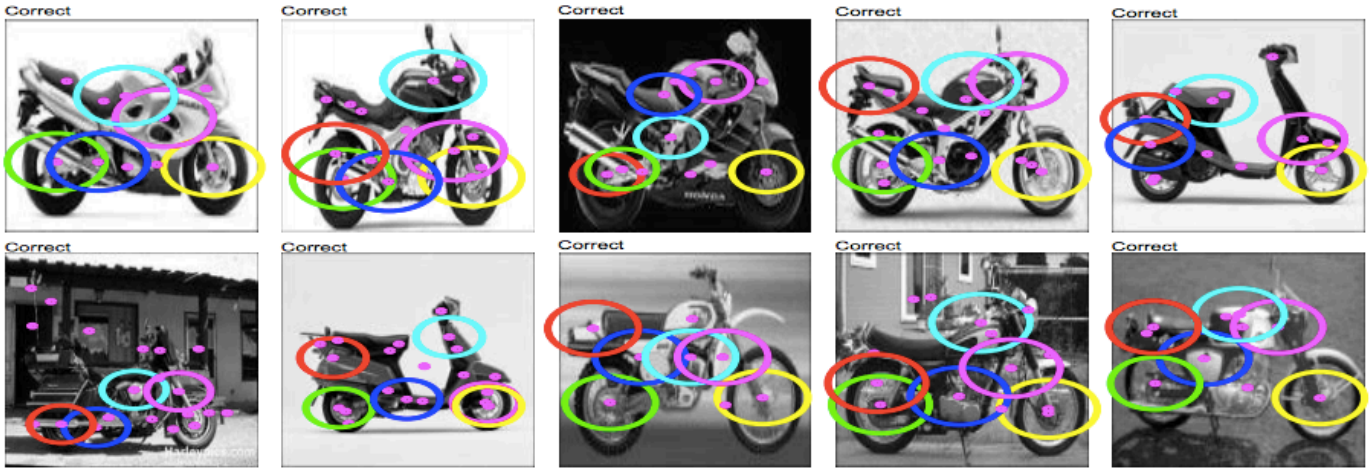
- Hand-crafted feature representation (classic computer vision)
- Off-the-shelf trainable classifier (SVM, k-NN, neural network)

“Classic” representation: Bag of (visual) features

Motivation 1: Part-based models

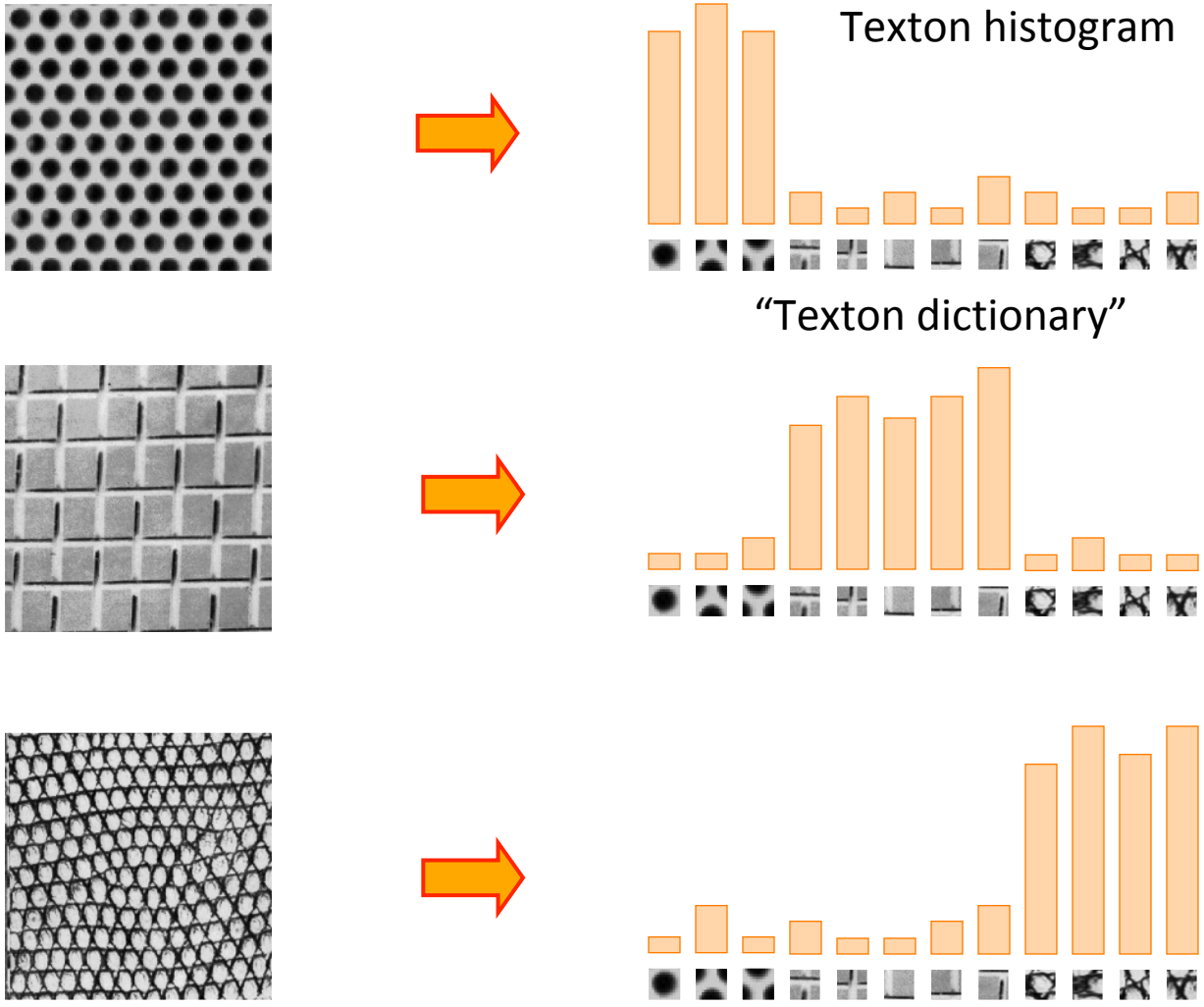






Weber, Welling & Perona (2000), Fergus, Perona & Zisserman (2003)

Motivation 2: Texture models



Julesz, 1981; Cula & Dana, 2001; Leung & Malik 2001; Mori, Belongie & Malik, 2001; Schmid 2001; Varma & Zisserman, 2002, 2003; Lazebnik, Schmid & Ponce, 2003

Motivation 3: Bags of words

- Orderless document representation: frequencies of words from a dictionary Salton & McGill (1983)

Motivation 3: Bags of words

- Orderless document representation: frequencies of words from a dictionary Salton & McGill (1983)

Motivation 3: Bags of words

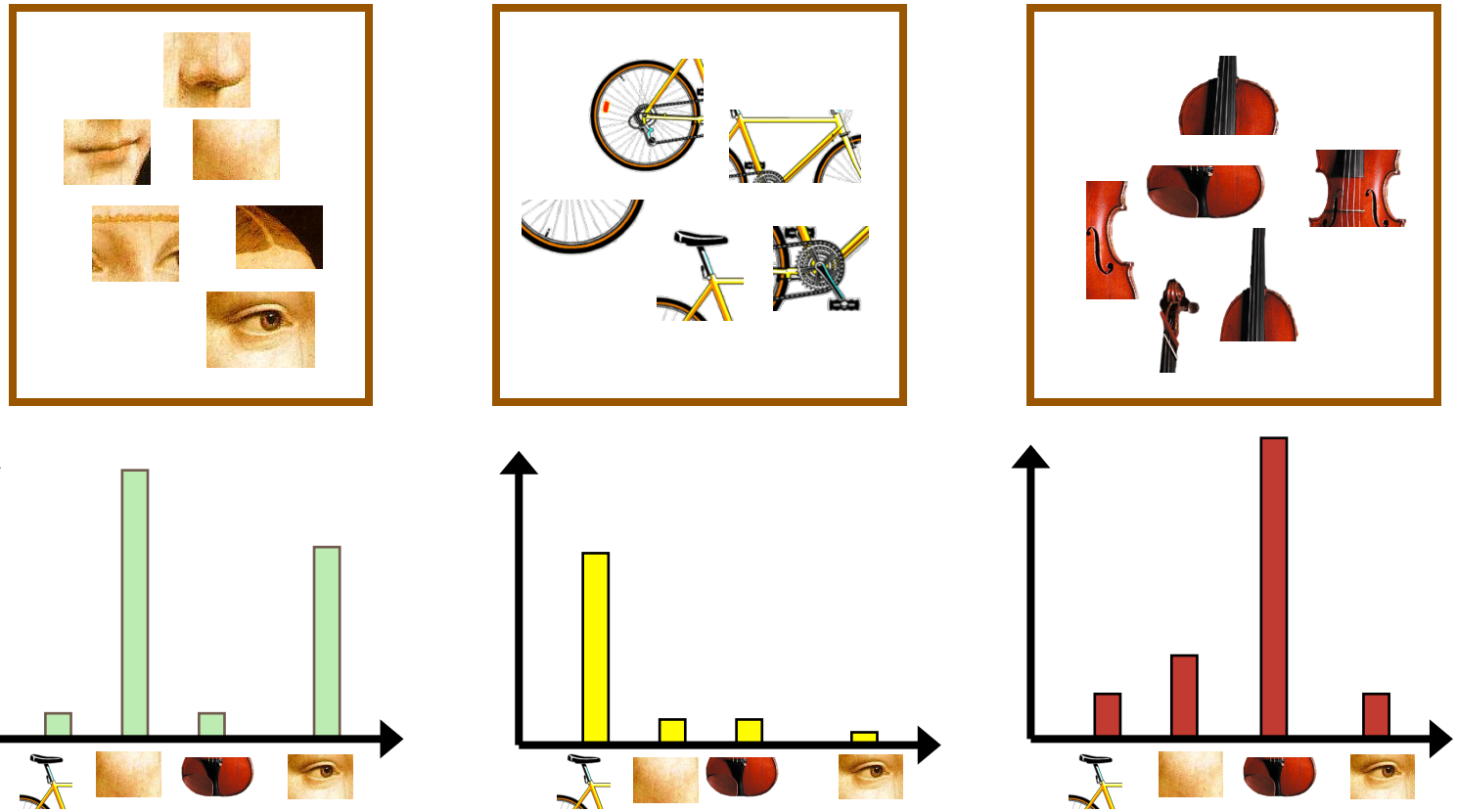
- Orderless document representation: frequencies of words from a dictionary Salton & McGill (1983)

Motivation 3: Bags of words

- Orderless document representation: frequencies of words from a dictionary Salton & McGill (1983)

Bag of features: Outline

1. Extract local features
2. Learn “visual vocabulary”
3. Quantize local features using visual vocabulary
4. Represent images by frequencies of “visual words”



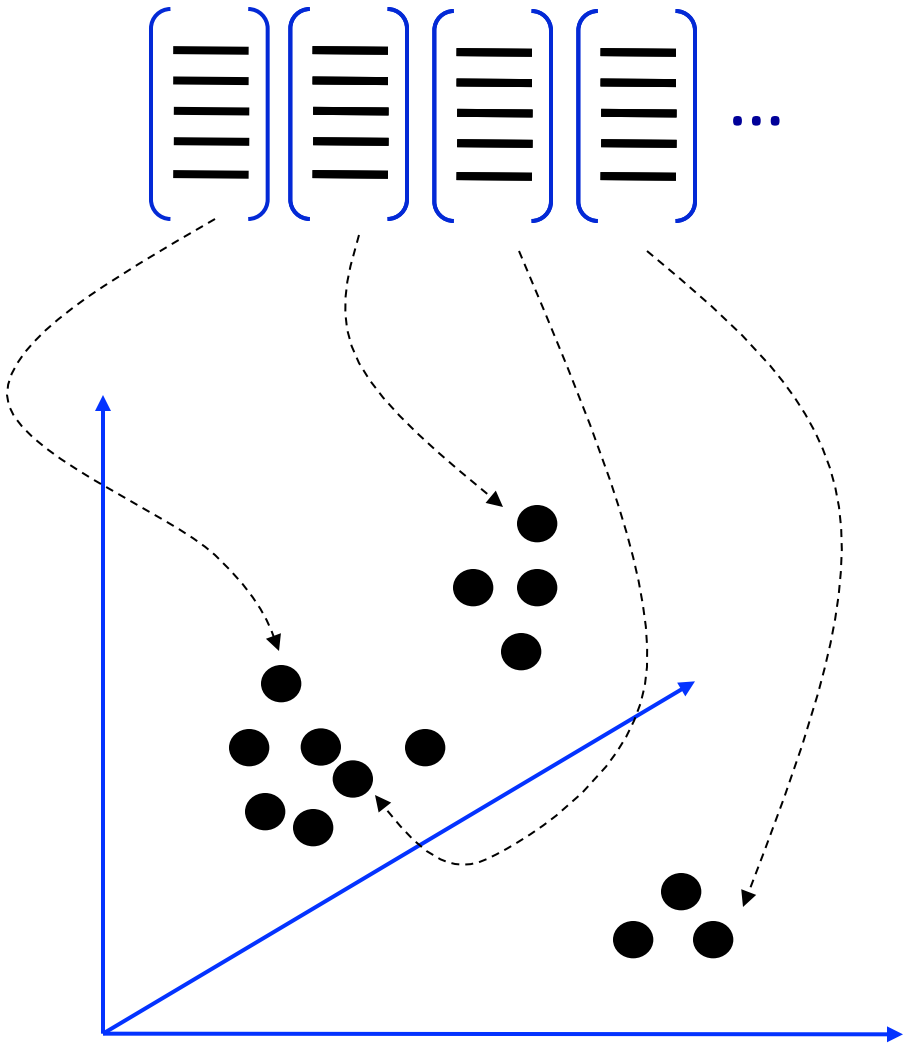
1. Local feature extraction

- Sample patches and extract descriptors

Use interest points

Use a dense grid

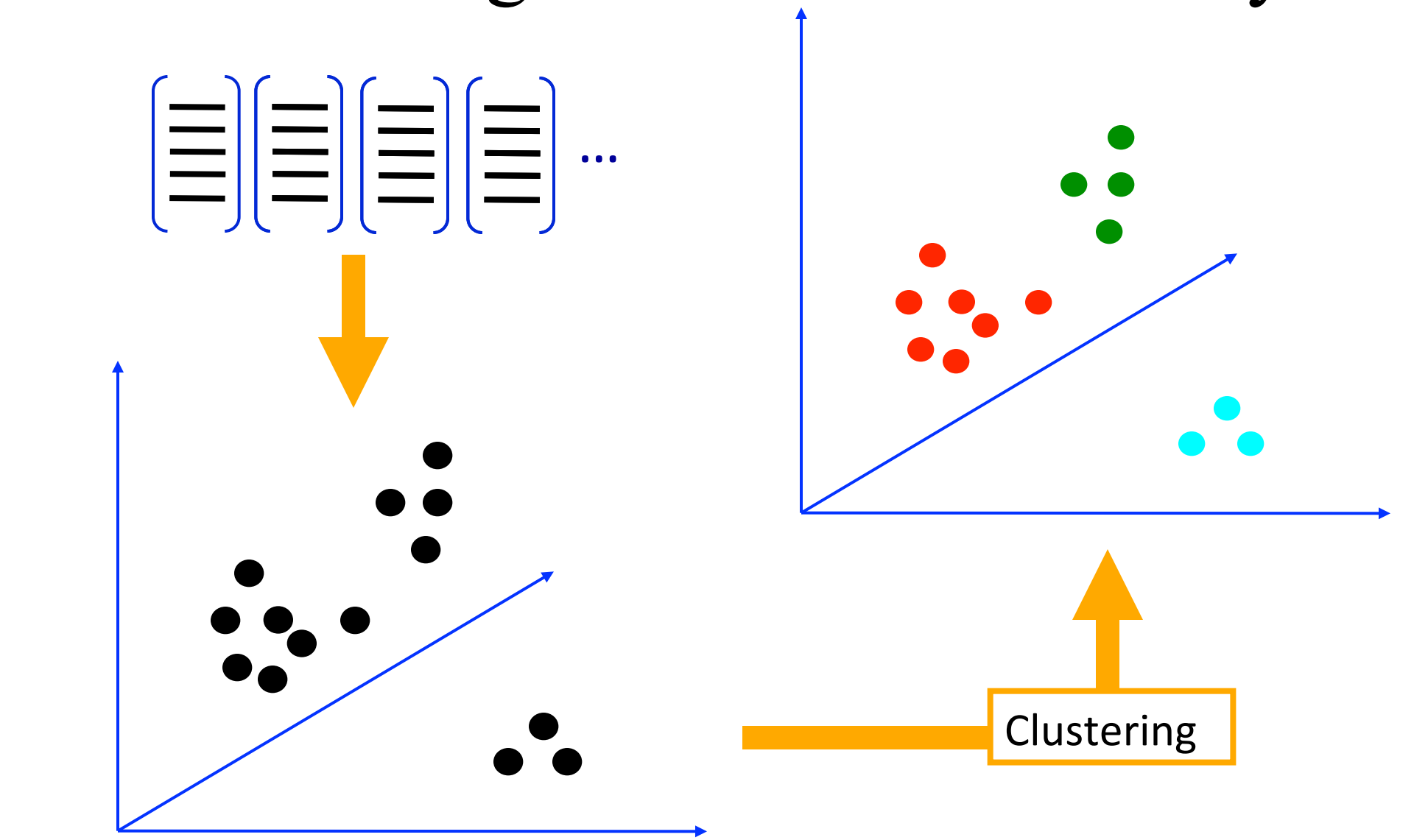
2. Learning the visual vocabulary



Extracted descriptors from the training set

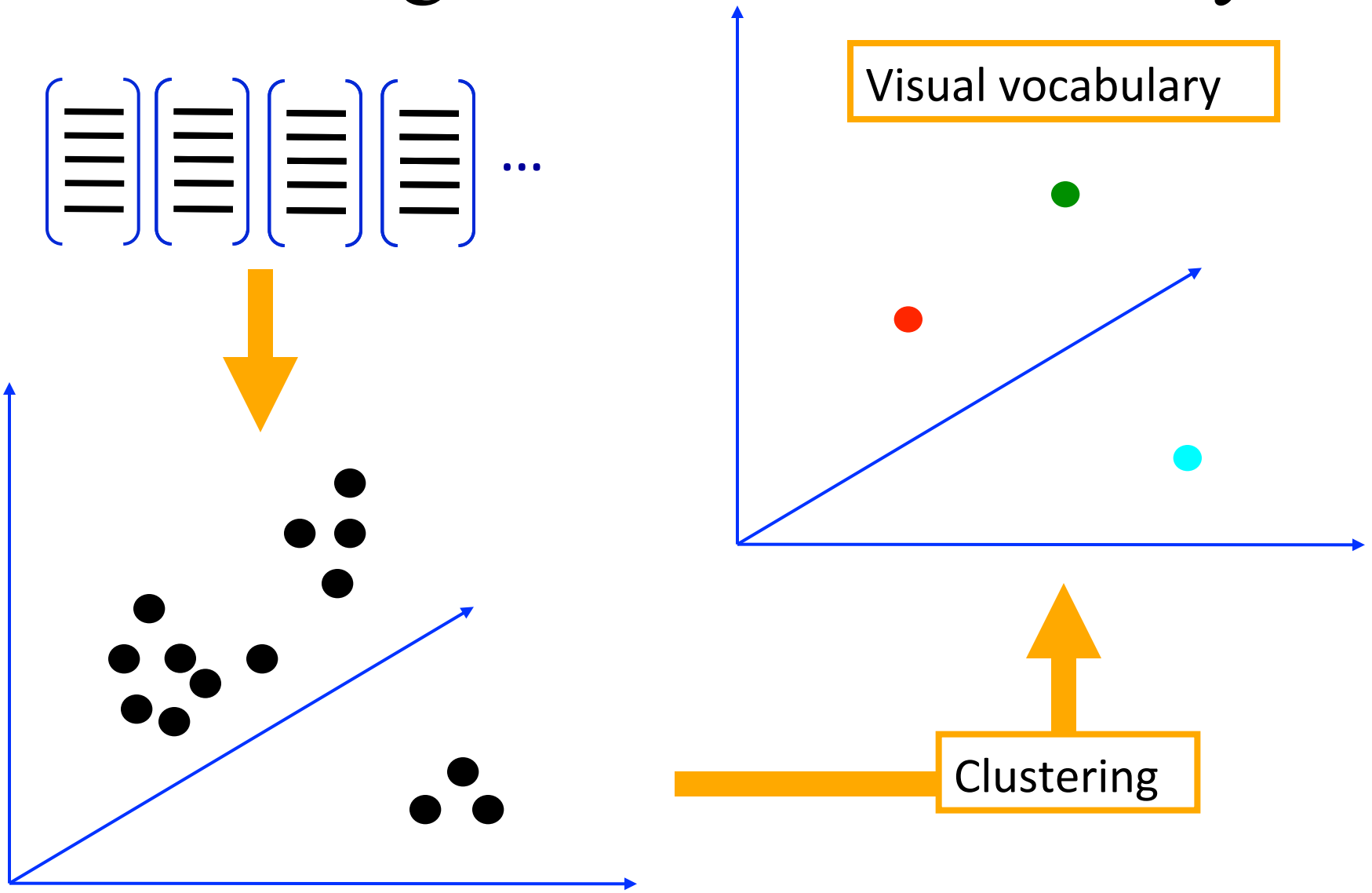
Slide credit: Josef Sivic

2. Learning the visual vocabulary



Slide credit: Josef Sivic

2. Learning the visual vocabulary



Slide credit: Josef Sivic

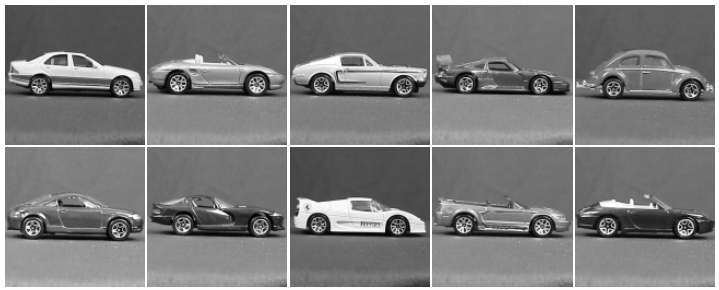
Recall: K-means clustering

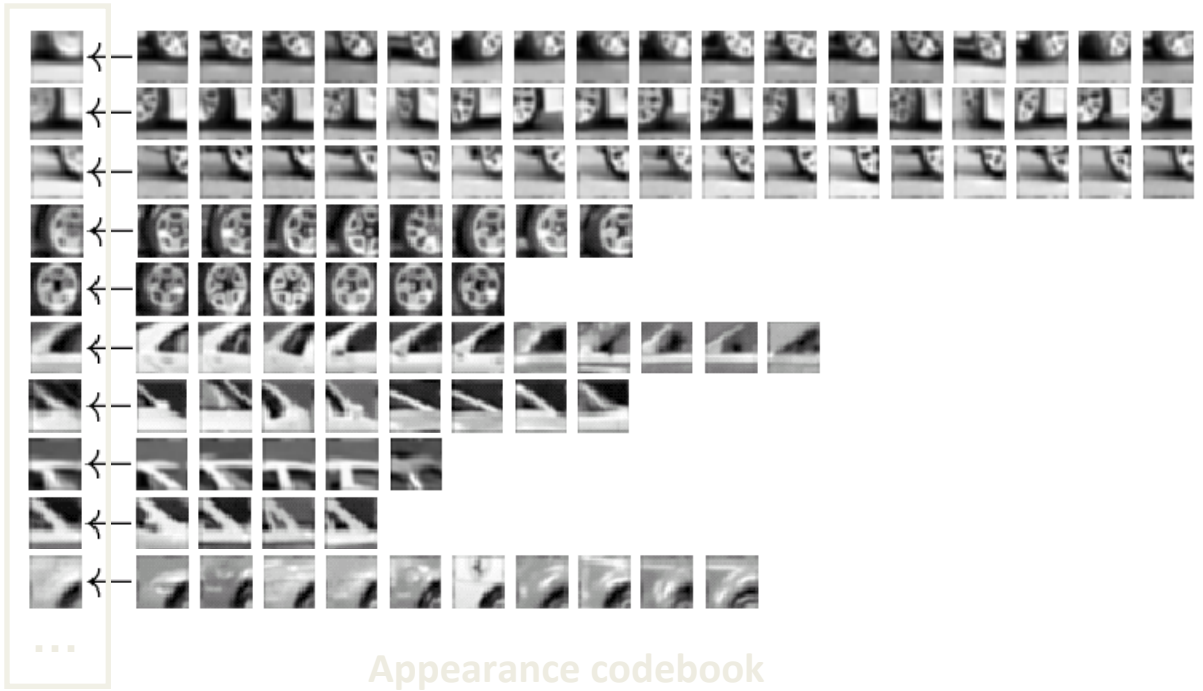
- Want to minimize sum of squared Euclidean distances between features $\mathbf{x}_i$ and their nearest cluster centers $\mathbf{m}_k$

$$D(X, M) = \sum_{\text{cluster } k} \sum_{\substack{\text{point } i \text{ in} \\ \text{cluster } k}} (\mathbf{x}_i - \mathbf{m}_k)^2$$

- Algorithm:
- Randomly initialize K cluster centers
- Iterate until convergence:
  - Assign each feature to the nearest center
  - Recompute each cluster center as the mean of all features assigned to it

Visual vocabularies

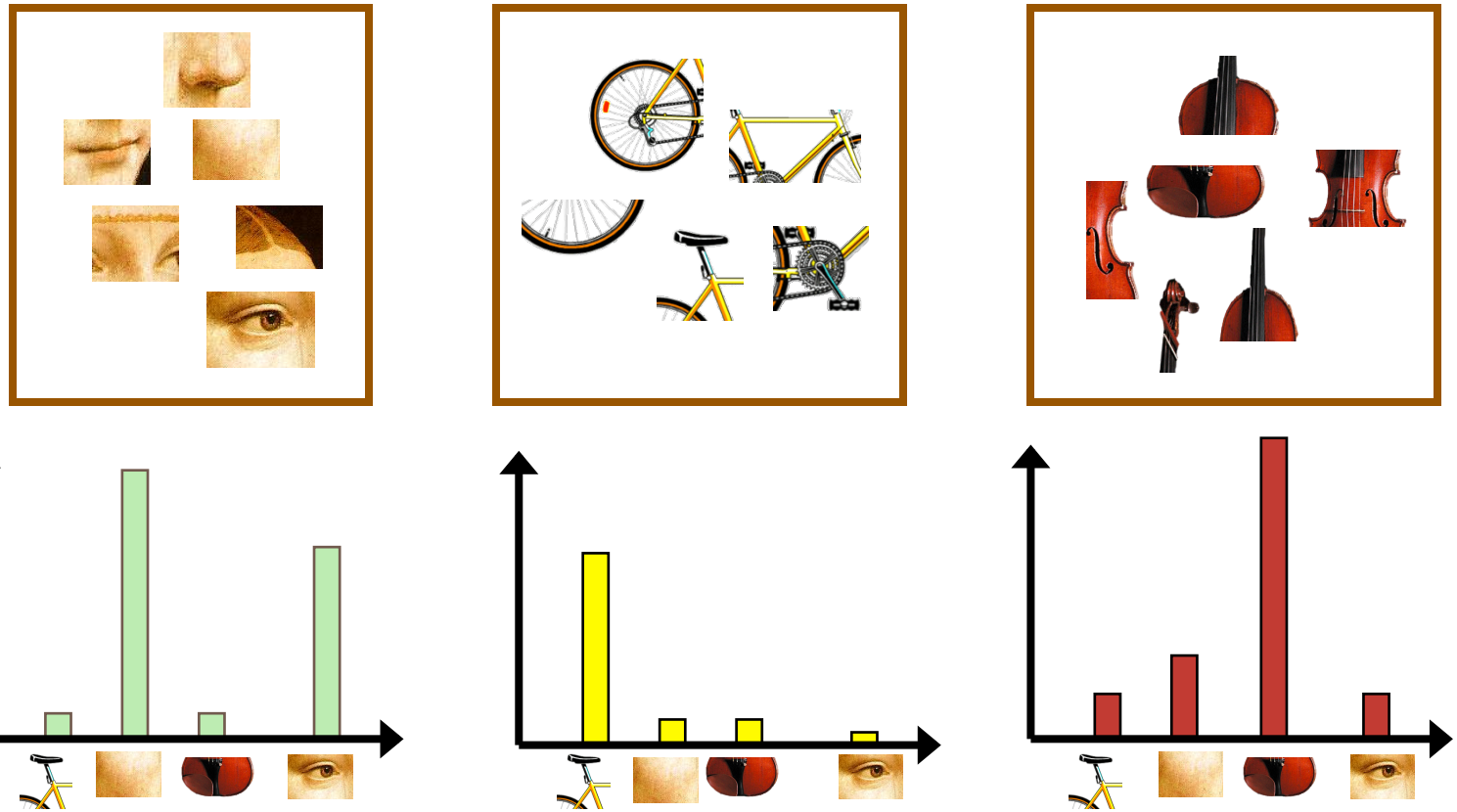




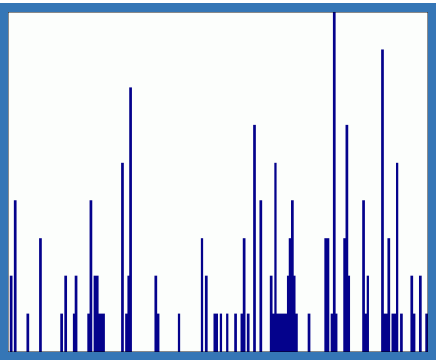
Slide credit: Bastian Leibe

Bag of features: Outline

1. Extract local features
2. Learn “visual vocabulary”
3. **Quantize local features using visual vocabulary**
4. **Represent images by frequencies of “visual words”**



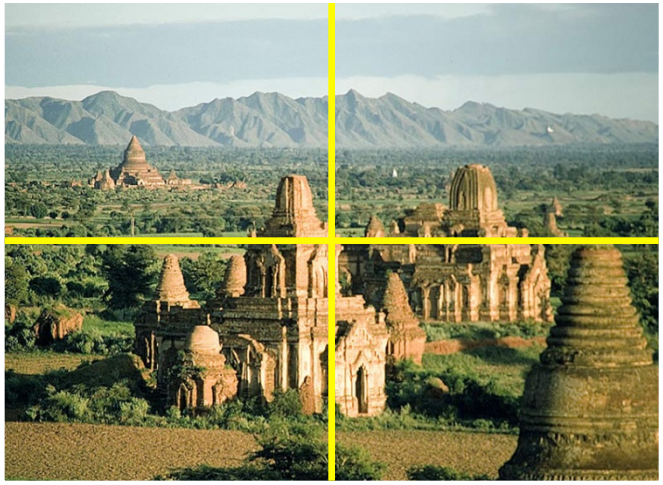
Spatial pyramids

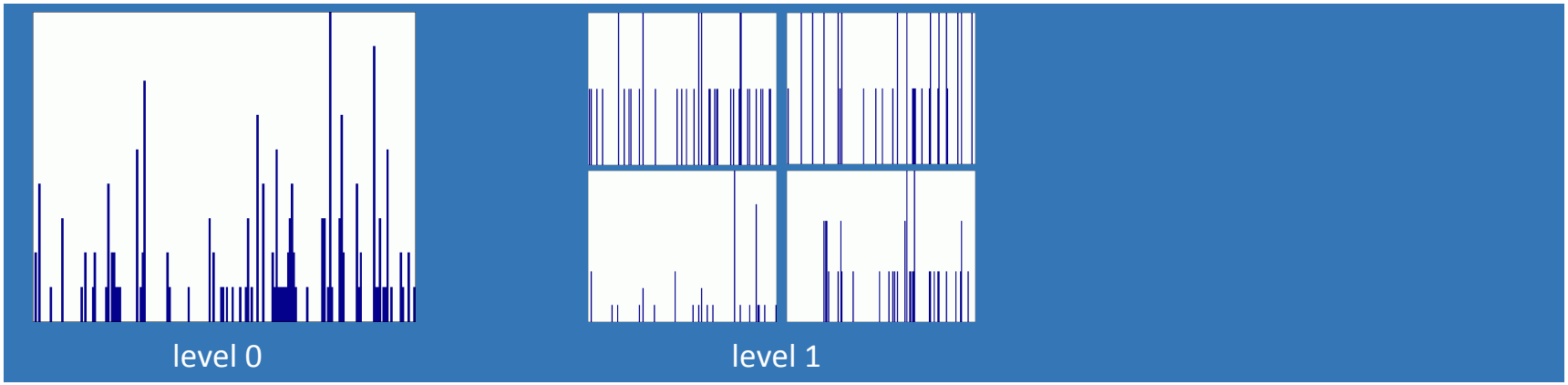


level 0

Lazebnik, Schmid & Ponce (CVPR 2006)

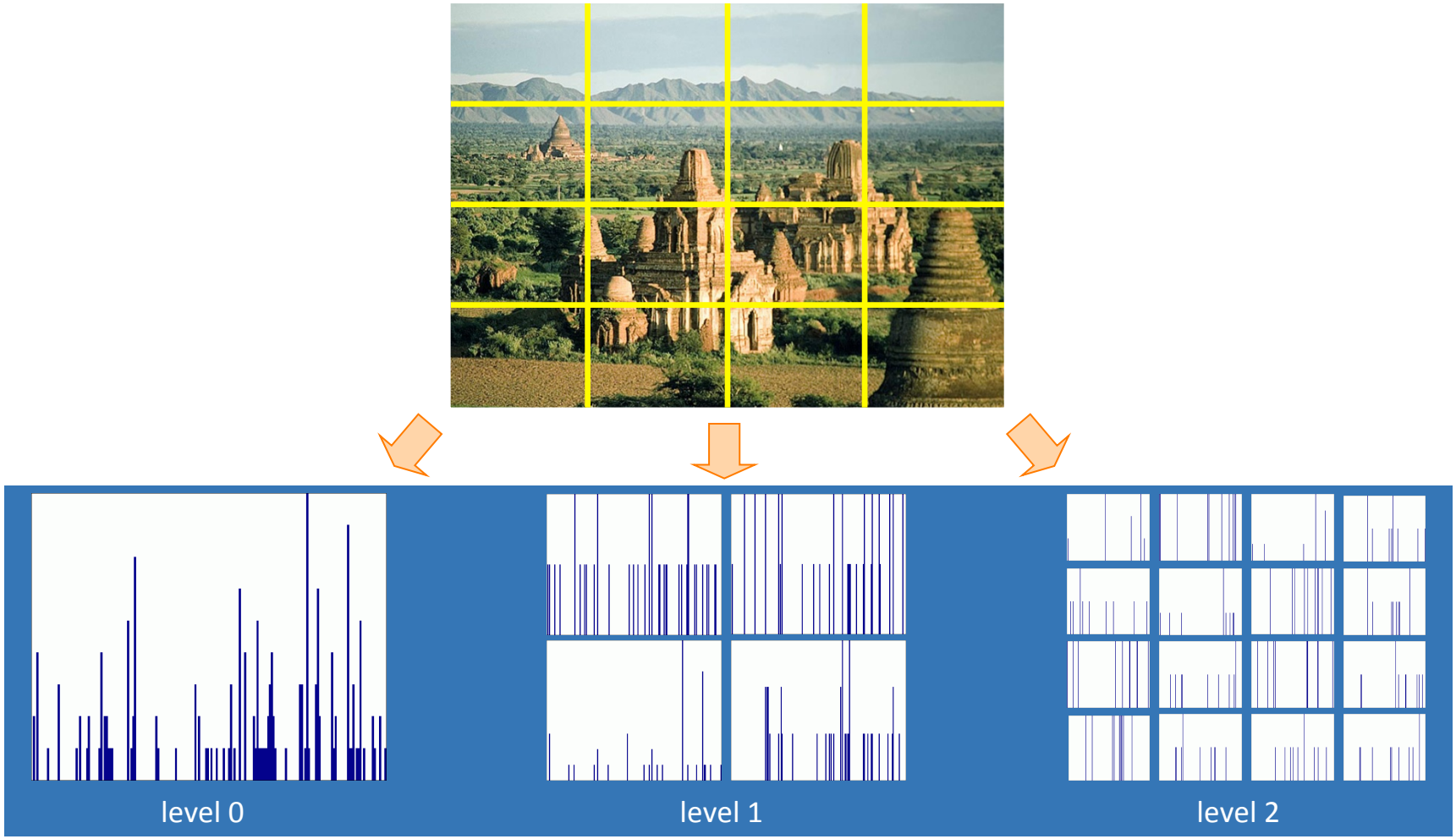
Spatial pyramids





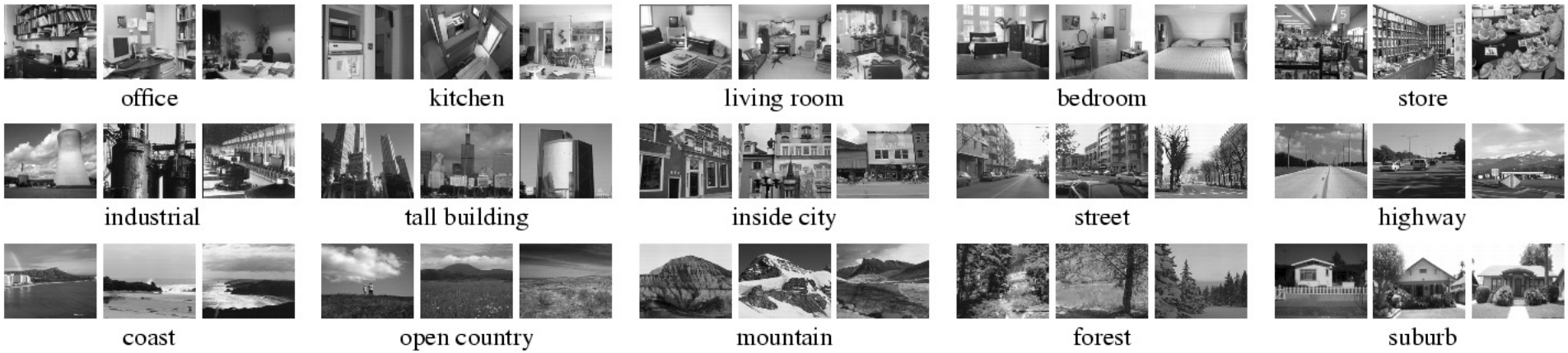
Lazebnik, Schmid & Ponce (CVPR 2006)

Spatial pyramids



Lazebnik, Schmid & Ponce (CVPR 2006)

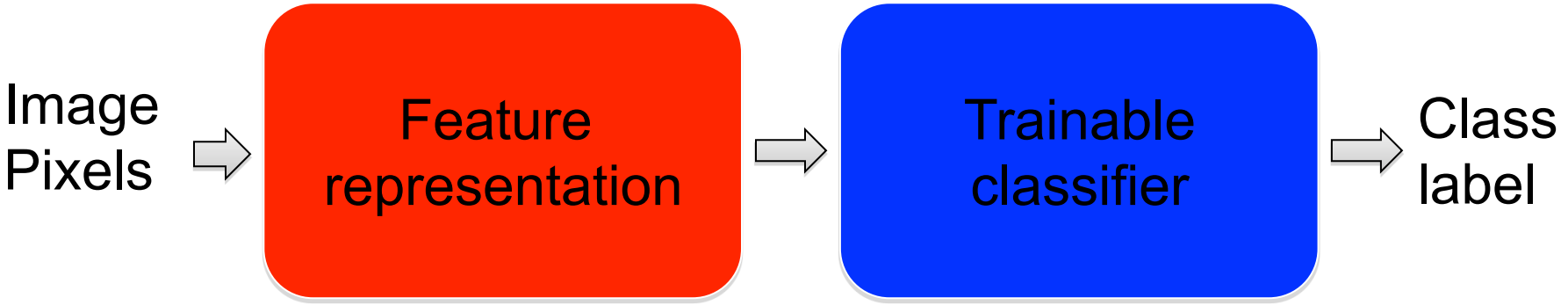
Spatial pyramids



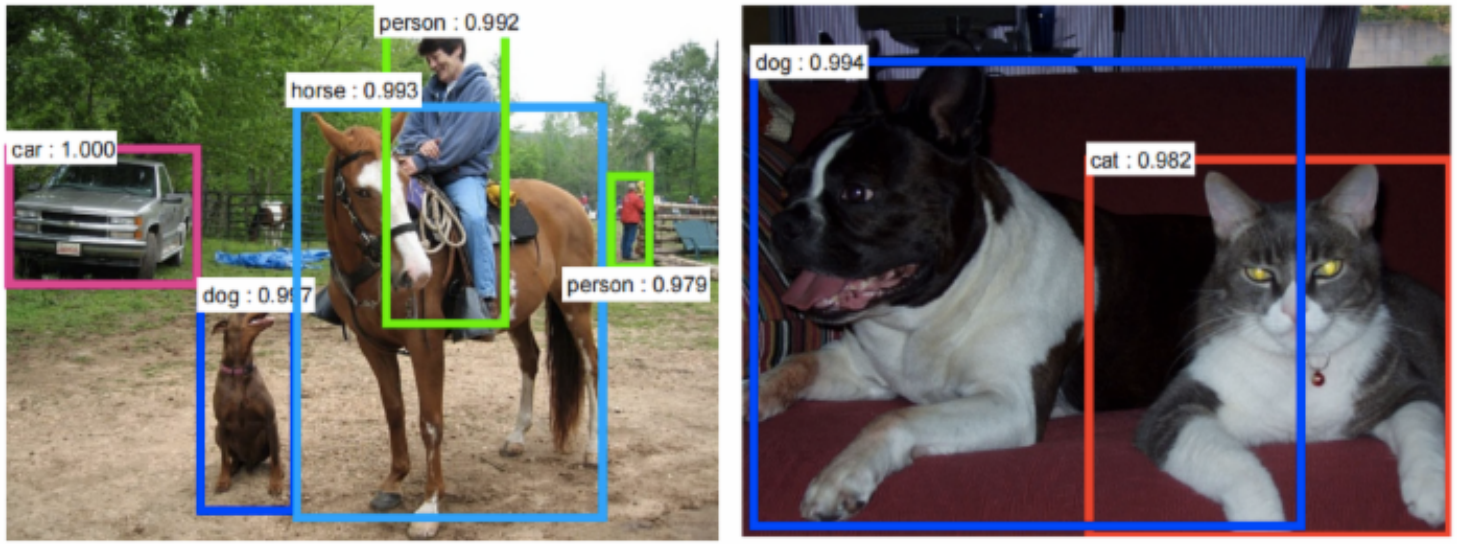
	Weak features (vocabulary size: 16)		Strong features (vocabulary size: 200)	
Level	Single-level	Pyramid	Single-level	Pyramid
0 (1×1)	45.3 ±0.5		72.2 ±0.6	
1 (2×2)	53.6 ±0.3	56.2 ±0.6	77.9 ±0.6	79.0 ±0.5
2 (4×4)	61.7 ±0.6	64.7 ±0.7	79.4 ±0.3	**81.1** ±0.3
3 (8×8)	63.3 ±0.8	**66.8** ±0.6	77.2 ±0.4	80.7 ±0.3

S. Lazebnik, C. Schmid, J. Ponce: "*Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scenes Categories*", CVPR 2006

From image classification to object detection

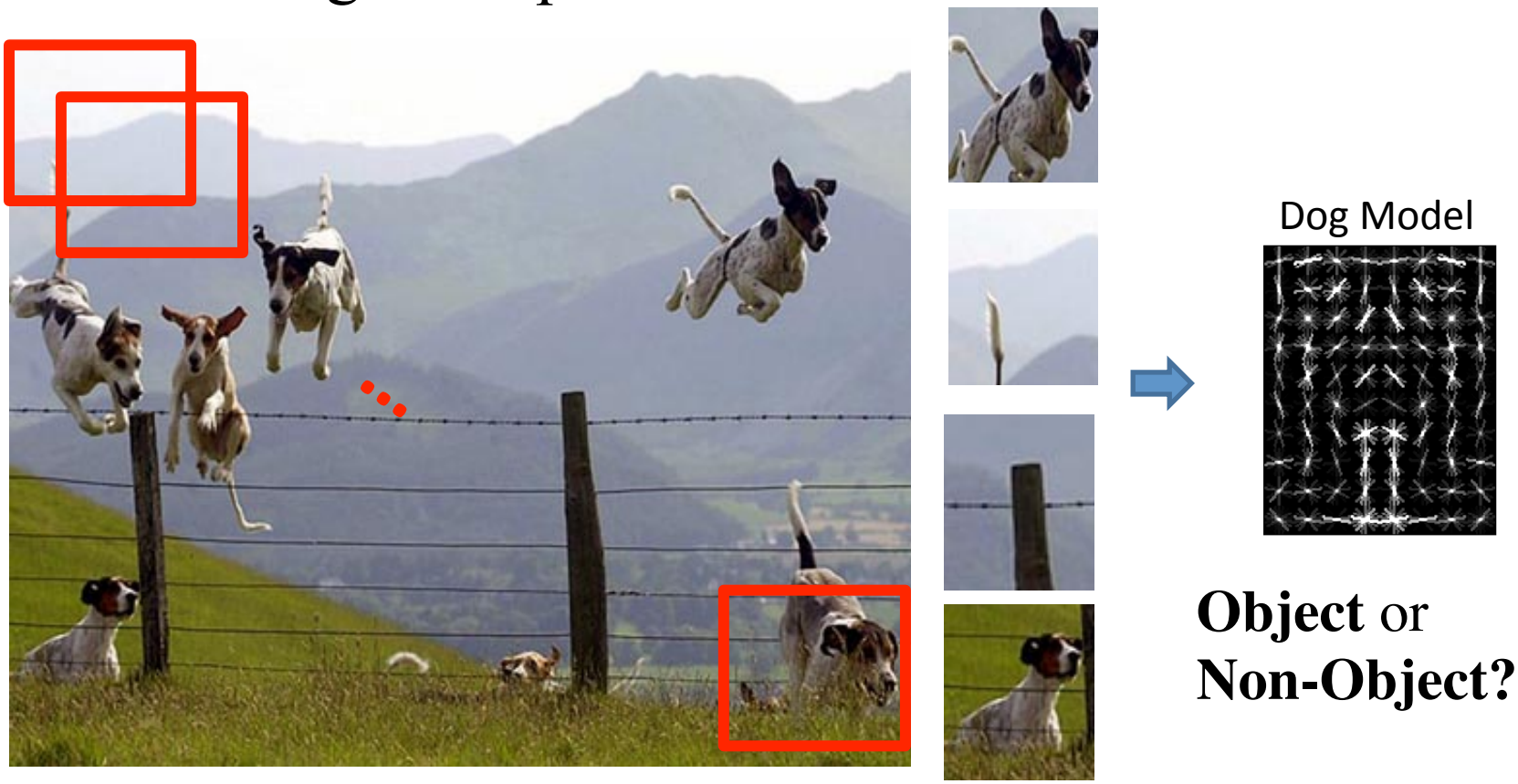


Object detection



Object Category Detection

- Focus on object search: “Where is it?”
- Build templates that quickly differentiate object patch from background patch



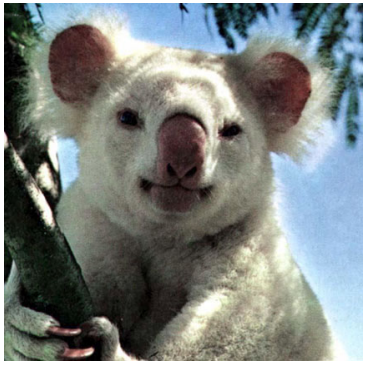
Challenges in modeling the object class

Illumination

Object pose

Clutter

Occlusions



Intra-class appearance

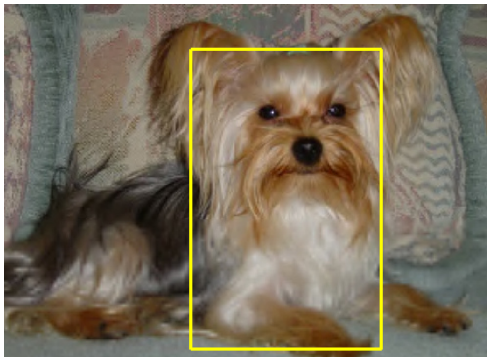
Viewpoint

Slide credit K. Grauman, B. Leibe

Challenges in modeling the object class

True Detections

Bad Localization



Confused with Similar Object

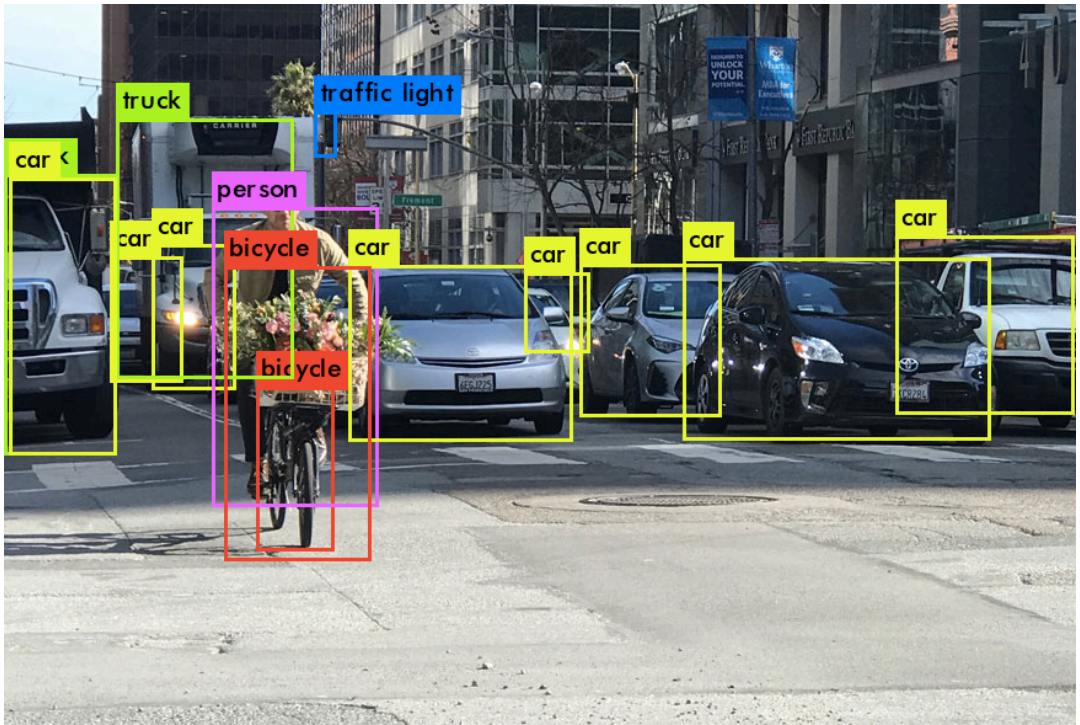
Misc. Background

Confused with Dissimilar Objects

Slide credit K. Grauman, B. Leibe

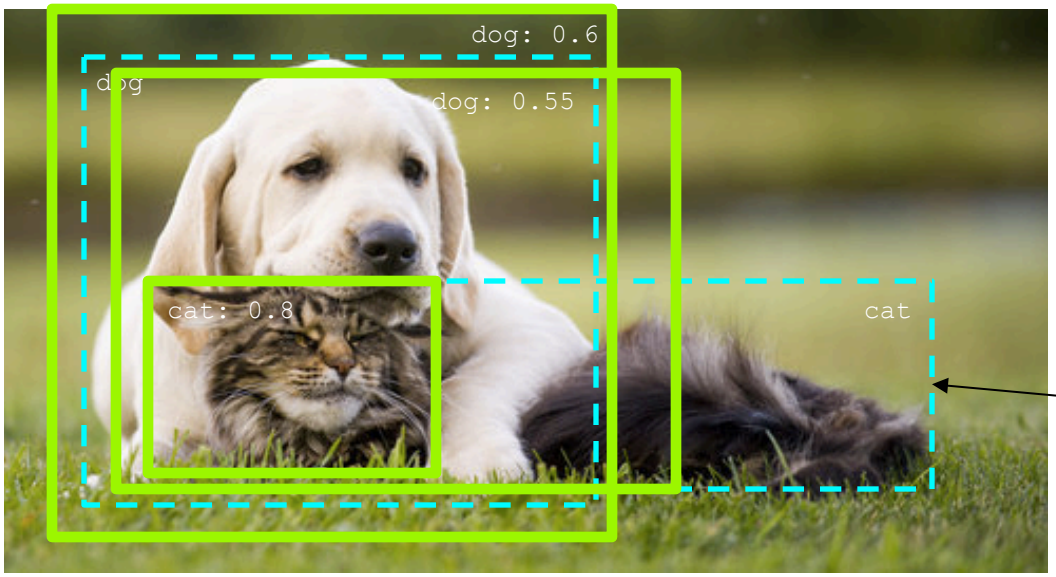
Other challenges of object detection

- Images may contain more than one class, multiple instances from the same class
- Bounding box localization
- Evaluation



Object detection evaluation

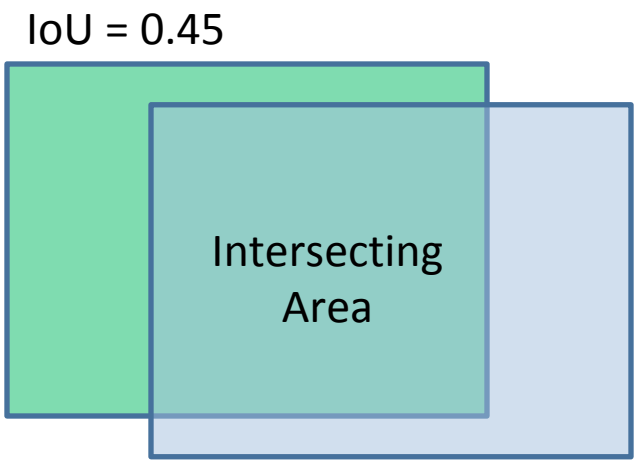
- At test time, predict bounding boxes, class labels, and confidence scores
- For each detection, determine whether it is a true or false positive
  - PASCAL criterion: Area(GT ∩ Det) / Area(GT ∪ Det) > 0.5 (threshold)
  - For multiple detections of the same ground truth box, only one considered a true positive

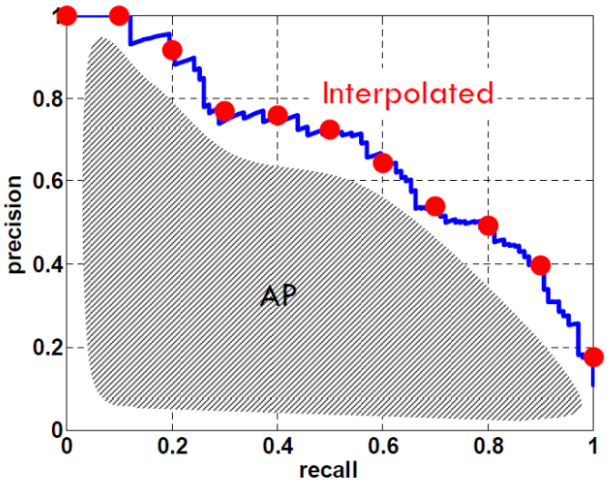


Object detection evaluation

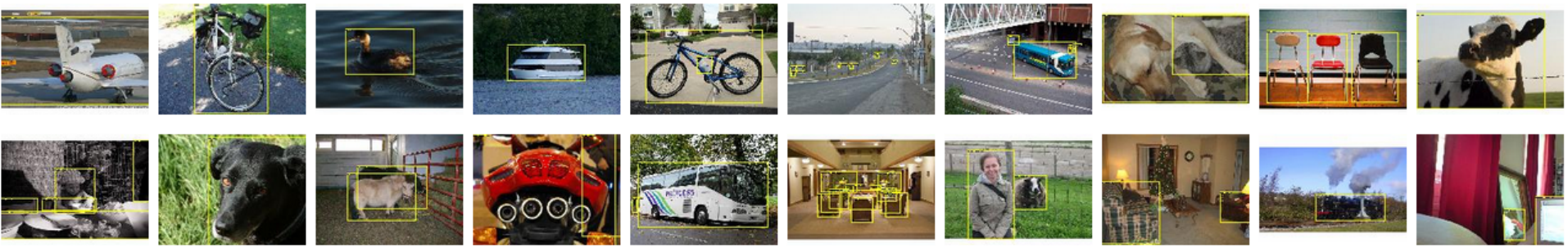
- Datasets
  - PASCAL VOC (2005-2012): 20 classes, ~20,000 images
  - MS COCO (2014-): 60 classes, ~300,000 images

- Evaluation
  - Output: for each class, predict bounding boxes (xmin, ymin, xmax, ymax) with confidences
  - Metric:
    - True detection: >= 0.5 Intersection over Union (IoU), not a duplicate
    - Precision, Recall
    - AP: area under the interpolated curve





PASCAL VOC Challenge (2005-2012)



- 20 challenge classes:
  - *Person*
  - *Animals:* bird, cat, cow, dog, horse, sheep
  - *Vehicles:* aeroplane, bicycle, boat, bus, car, motorbike, train
  - *Indoor:* bottle, chair, dining table, potted plant, sofa, tv/monitor

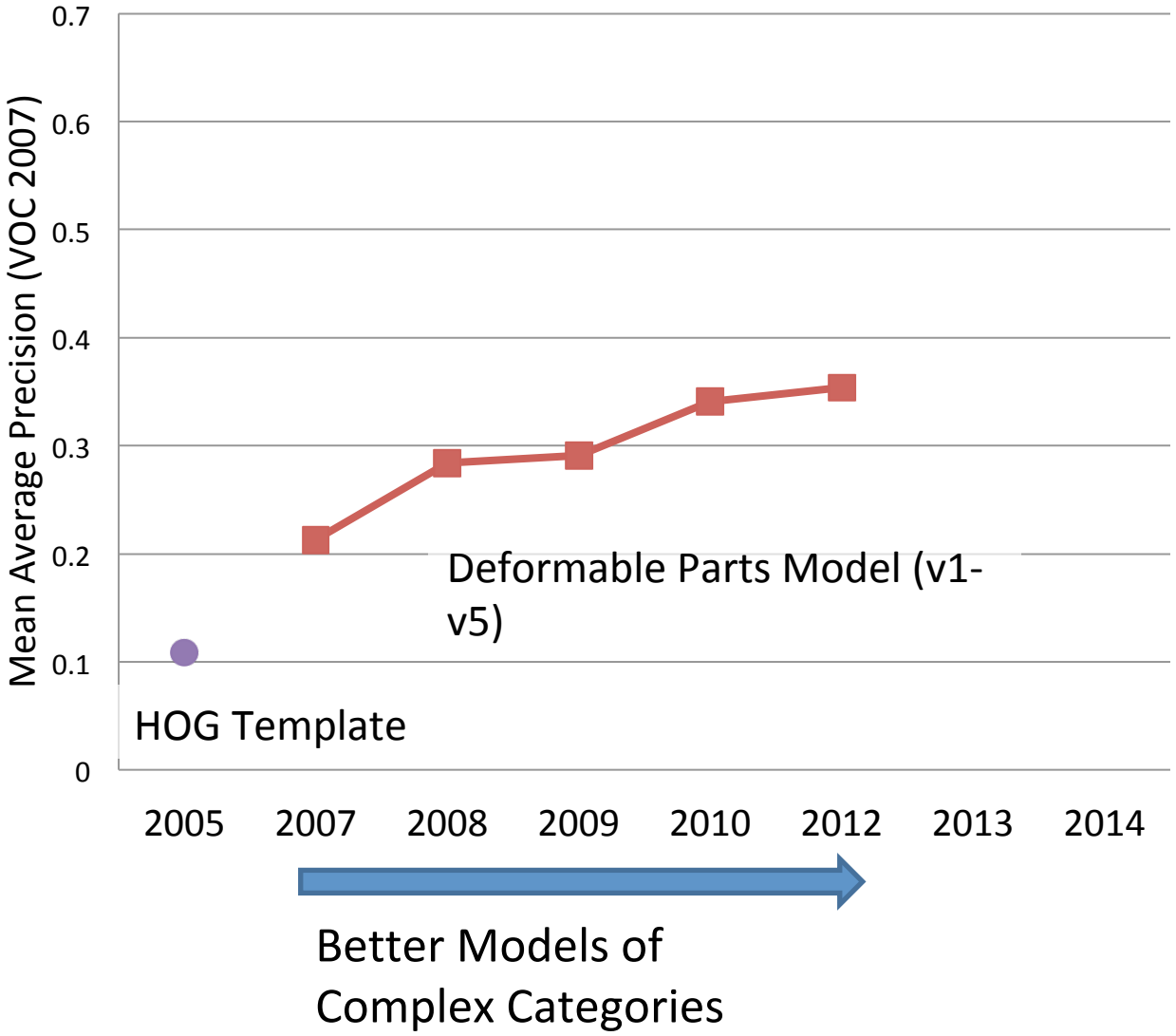
- Dataset size (by 2012): 11.5K training/validation images, 27K bounding boxes, 7K segmentations

http://host.robots.ox.ac.uk/pascal/VOC/

Improvements in object detection

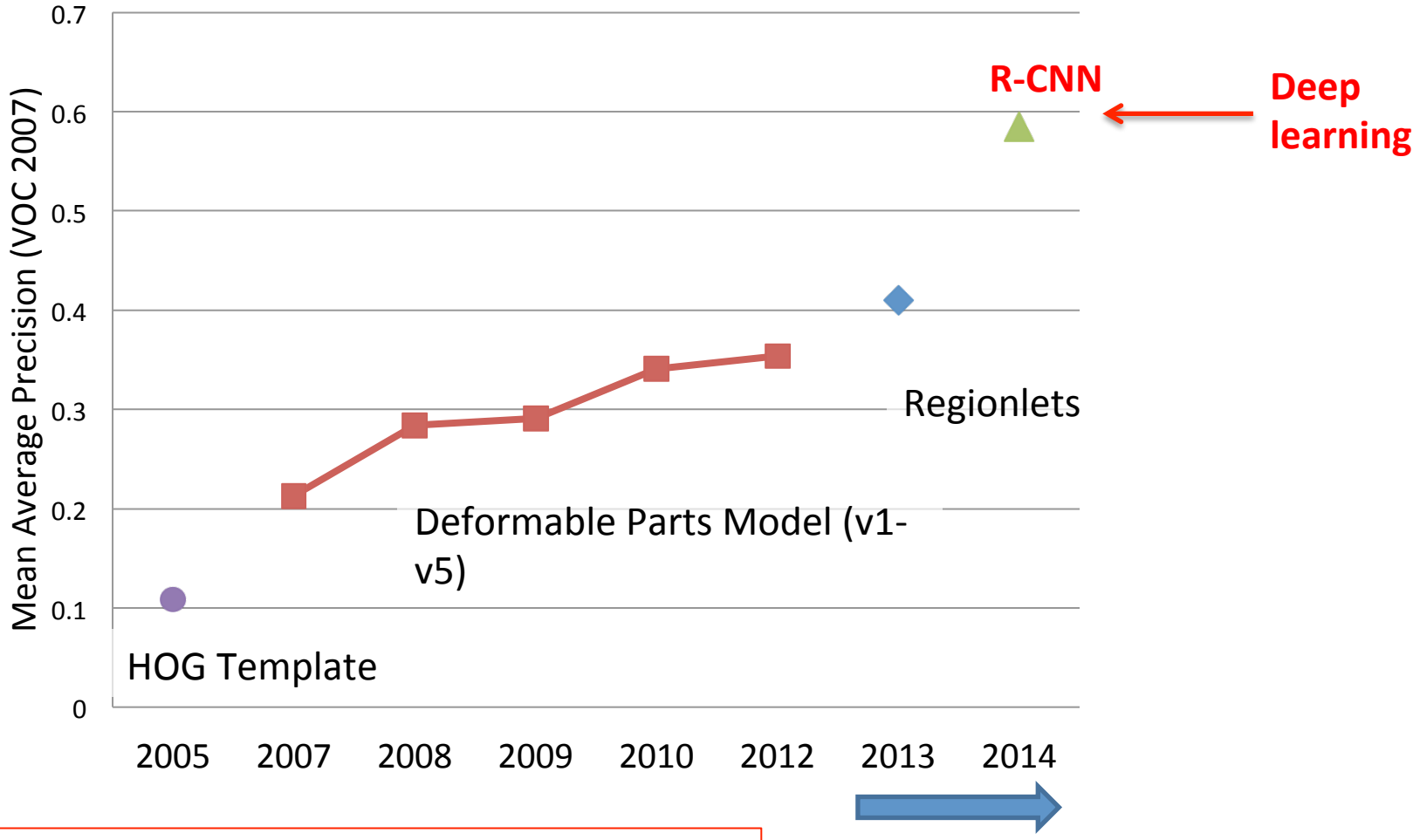
HOG: Dalal-Triggs 2005

Improvements in object detection



HOG: Dalal-Triggs 2005 DPM: Felzenszwalb et al. 2008-2012

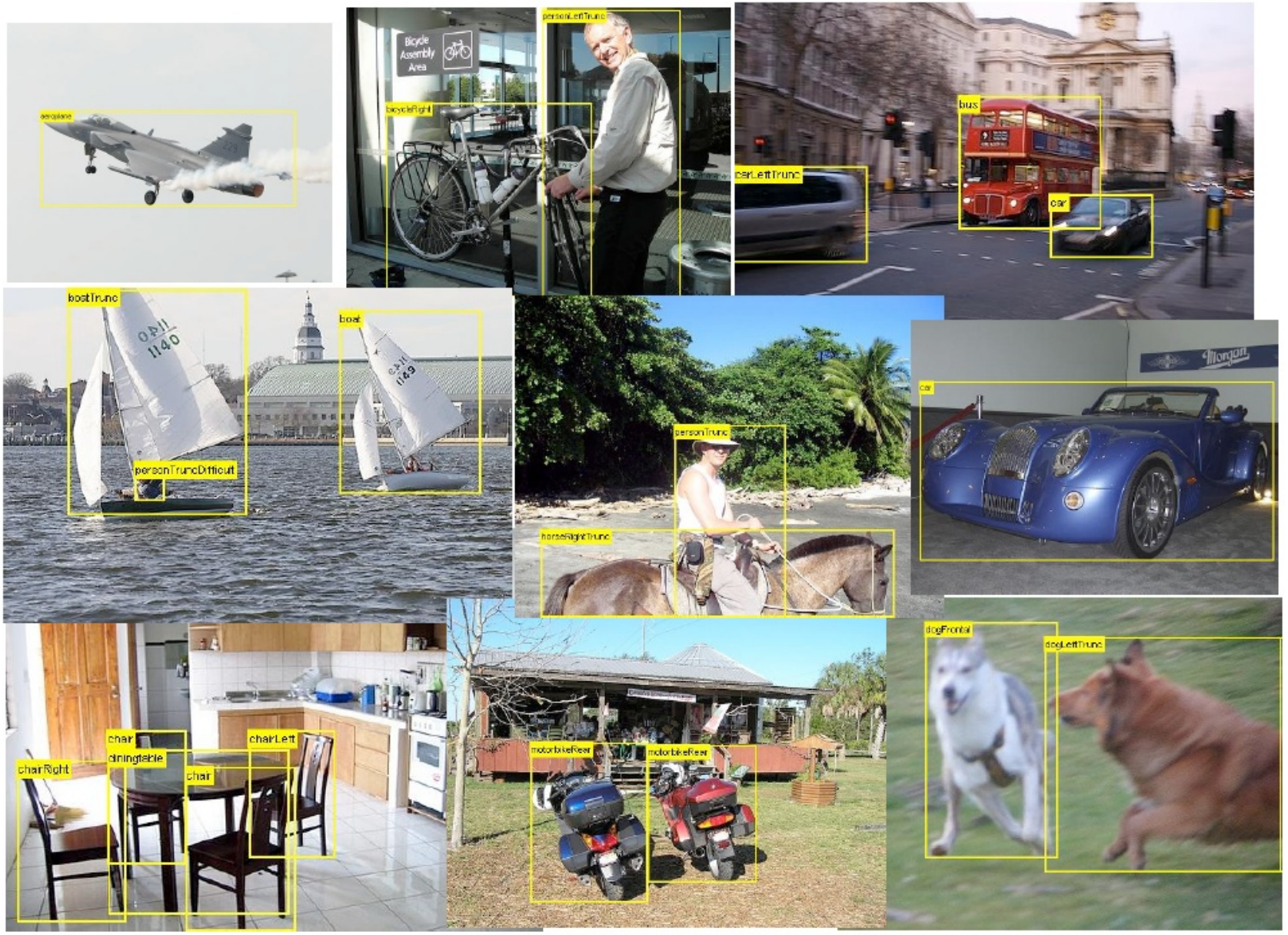
Improvements in object detection



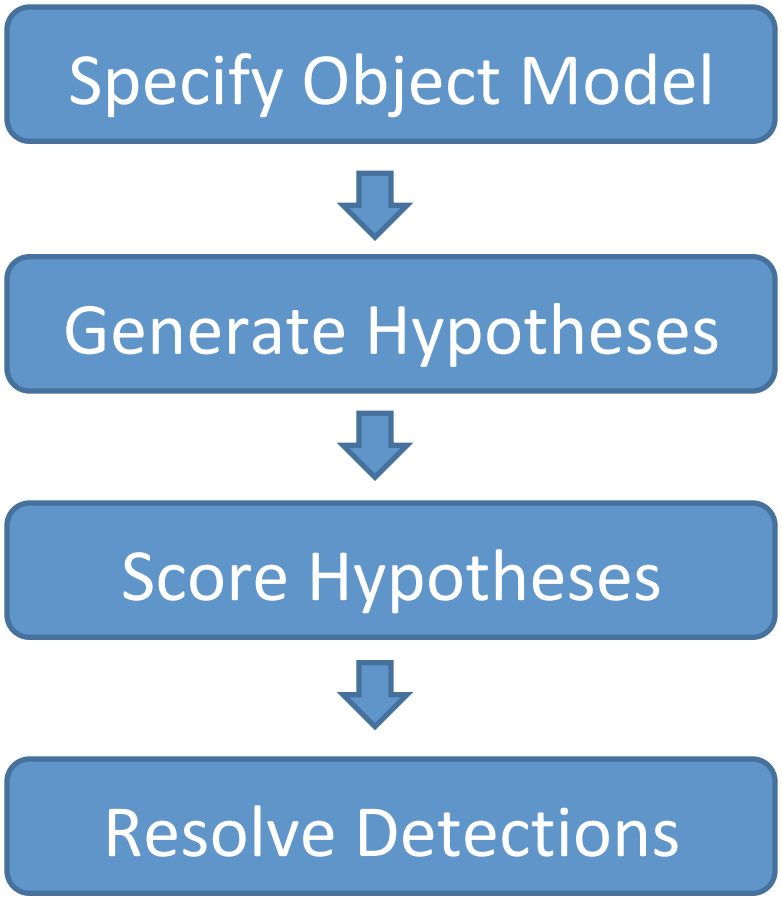
Key Advance: Learn effective features from massive amounts of labeled data *and* adapt to new tasks with less data

HOG: Dalal-Triggs 2005 DPM: Felzenszwalb et al. 2008-2012 Regionlets: Wang et al. 2013 R-CNN: Girshick et al. 2014

Detection before deep learning



General Process of Object Detection

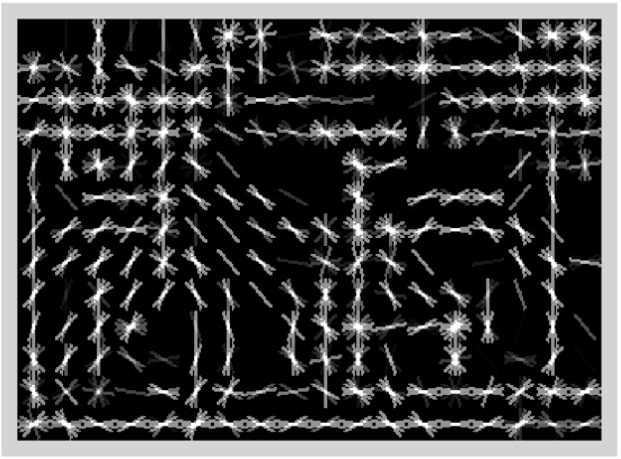


What are the object parameters?

Specifying an object model

1. Statistical Template in Bounding Box
   - Object is some (x,y,w,h) in image
   - Features defined wrt bounding box coordinates

Image

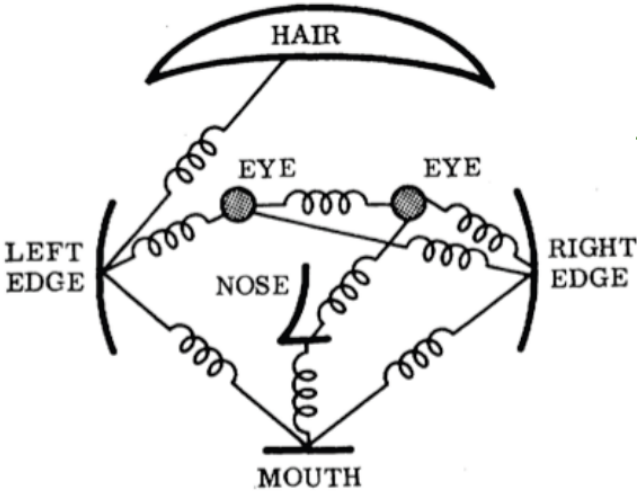


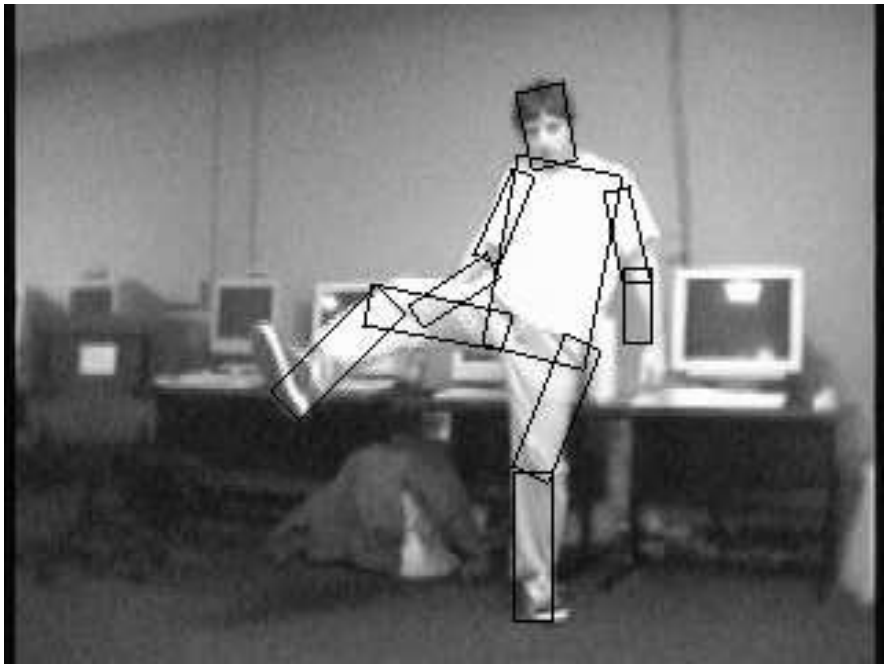
Template Visualization

Images from Felzenszwalb

Specifying an object model

2. Articulated parts model
   - Object is configuration of parts
   - Each part is detectable



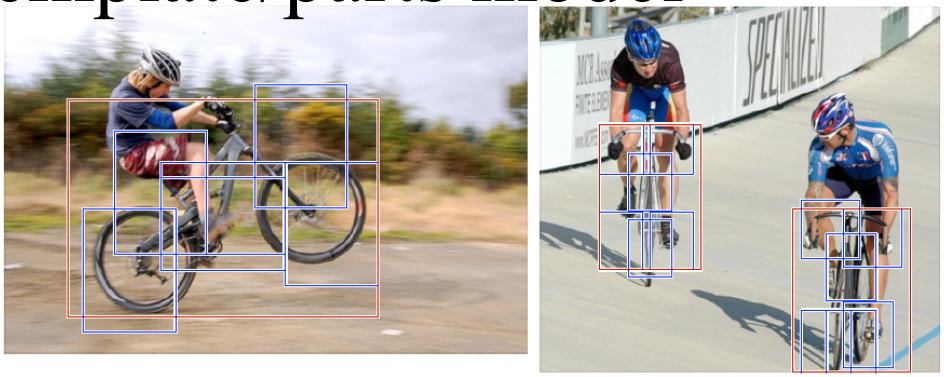


Images from Felzenszwalb

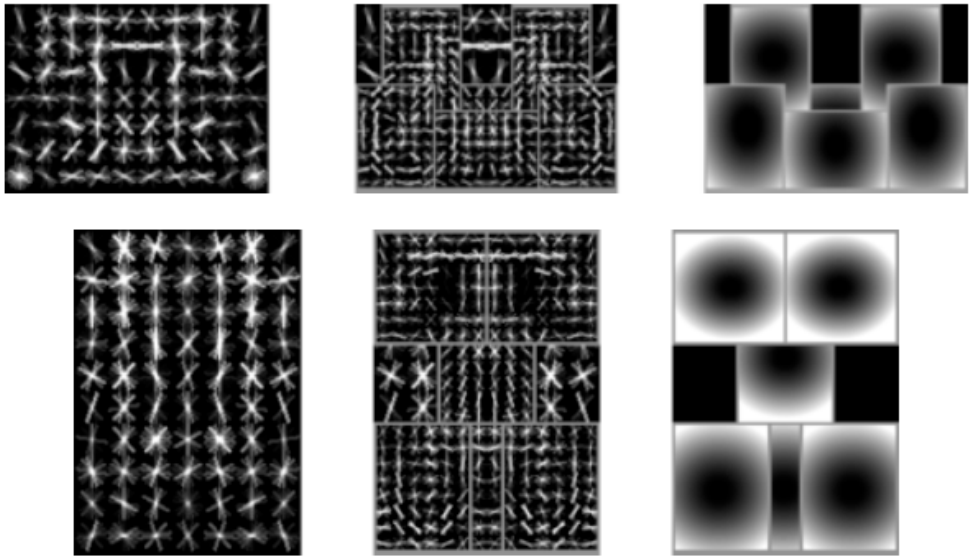
Specifying an object model

3. Hybrid template/parts model

Detections



Template Visualization



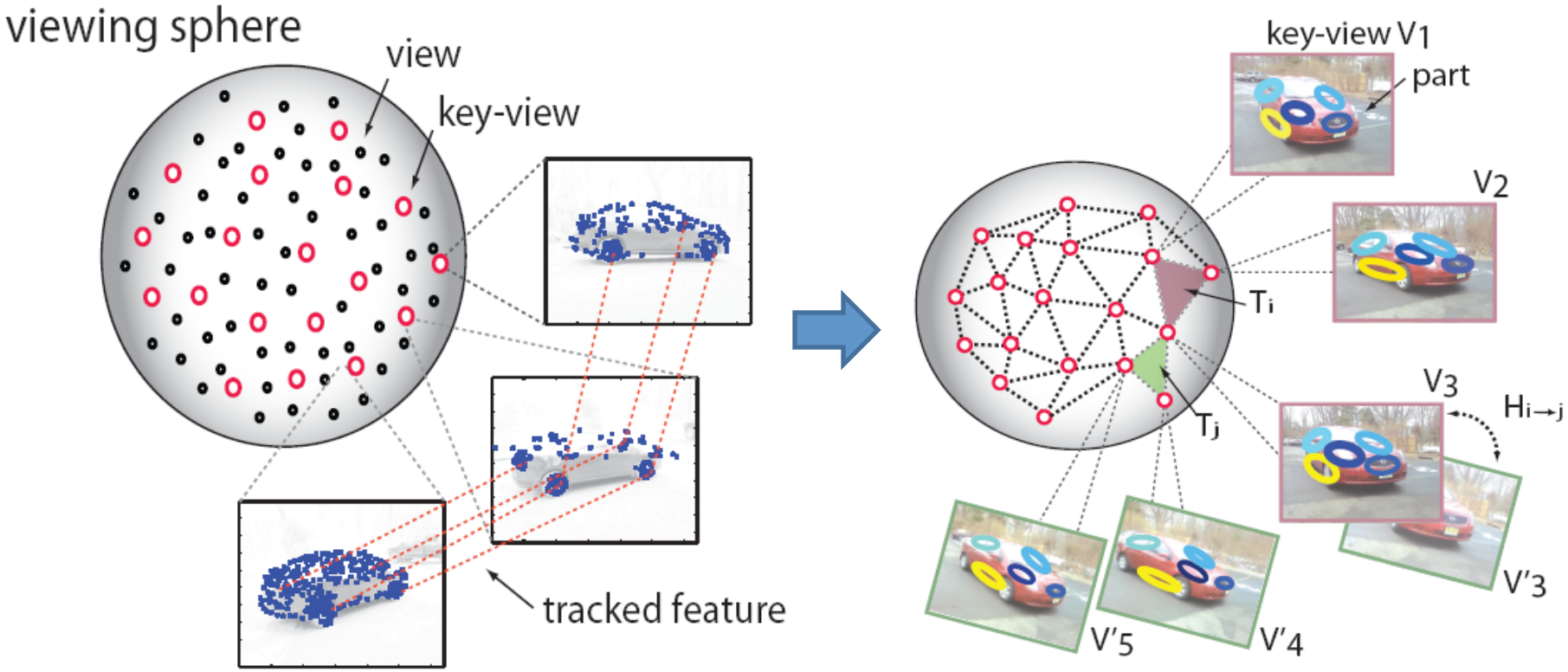
root filters coarse resolution | part filters finer resolution | deformation models

Felzenszwalb et al. 2008

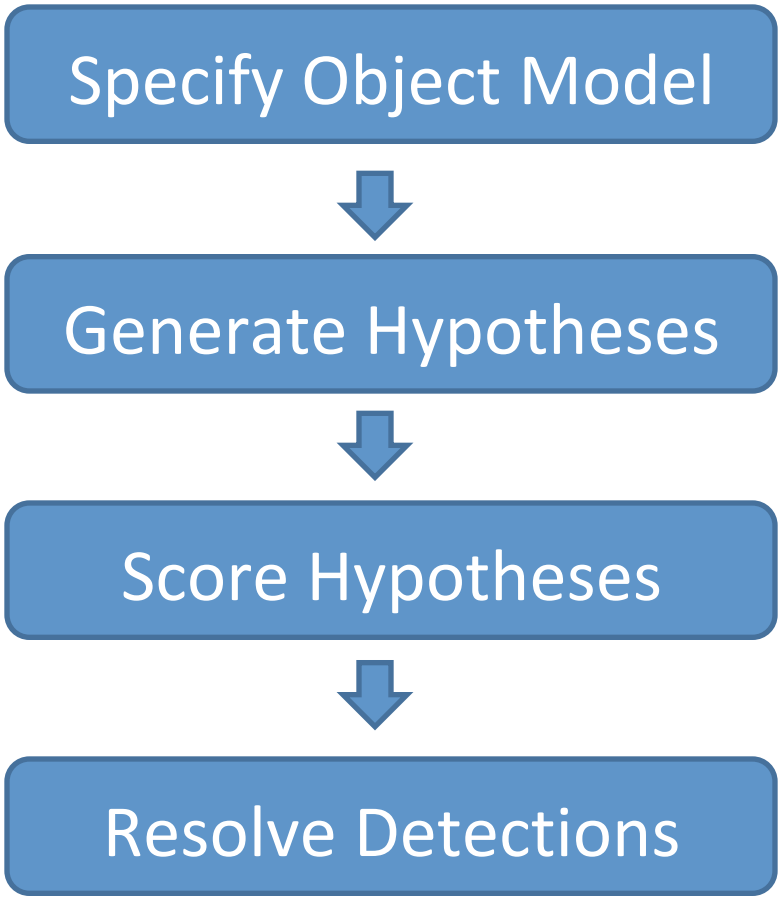
Specifying an object model

4. 3D-ish model

- Object is collection of 3D planar patches under affine transformation



General Process of Object Detection



Propose an alignment of the model to the image

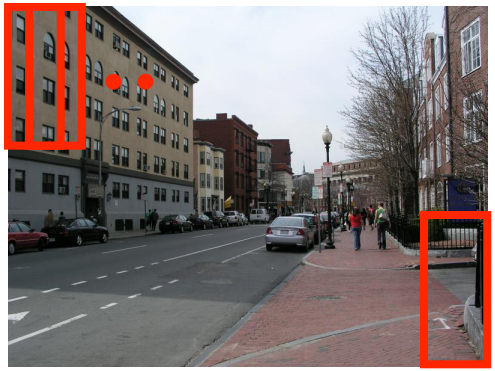
Generating hypotheses

1. Sliding window
   - Test patch at each location and scale

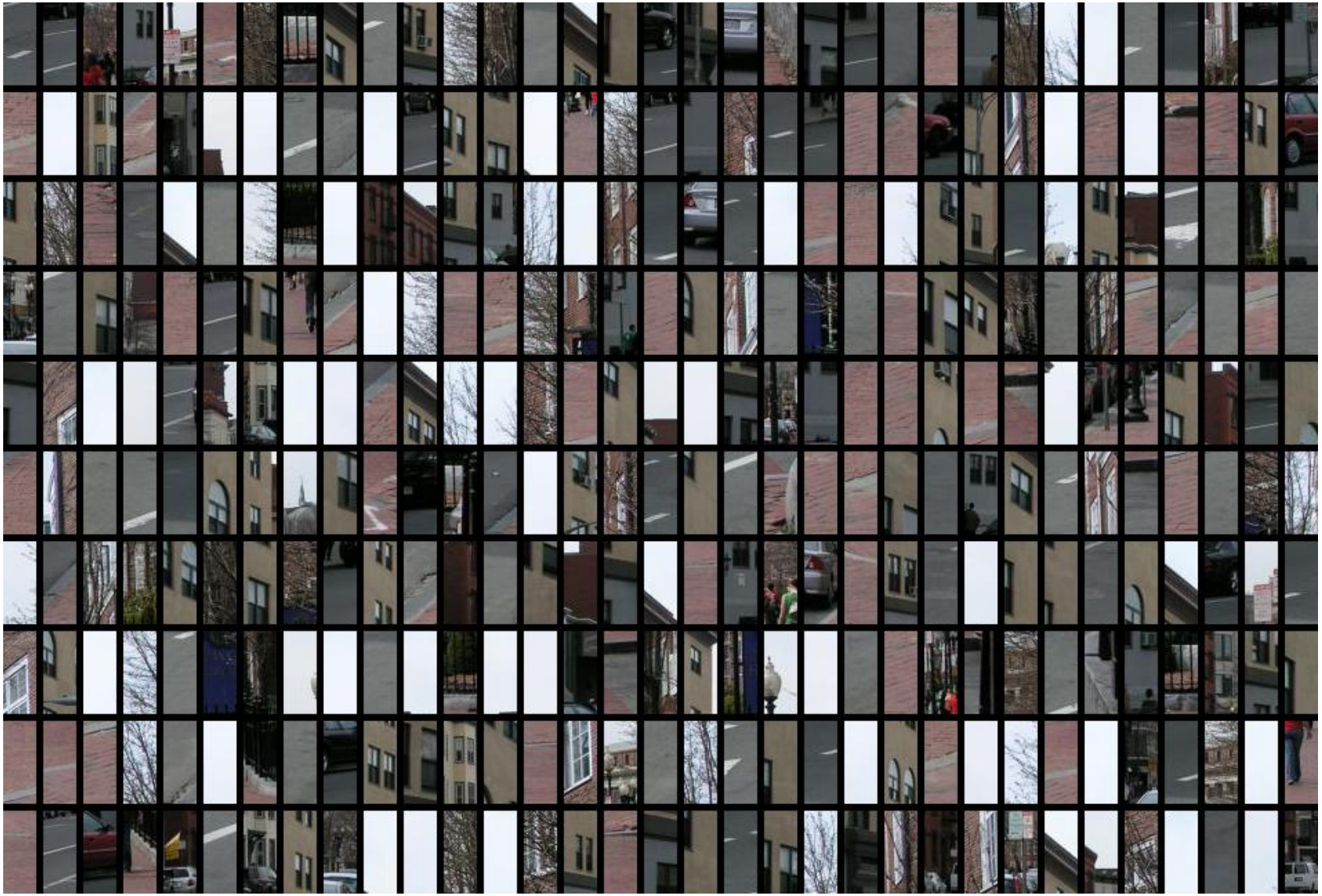
Generating hypotheses

1. Sliding window
   - Test patch at each location and scale

Sliding window: a simple alignment solution

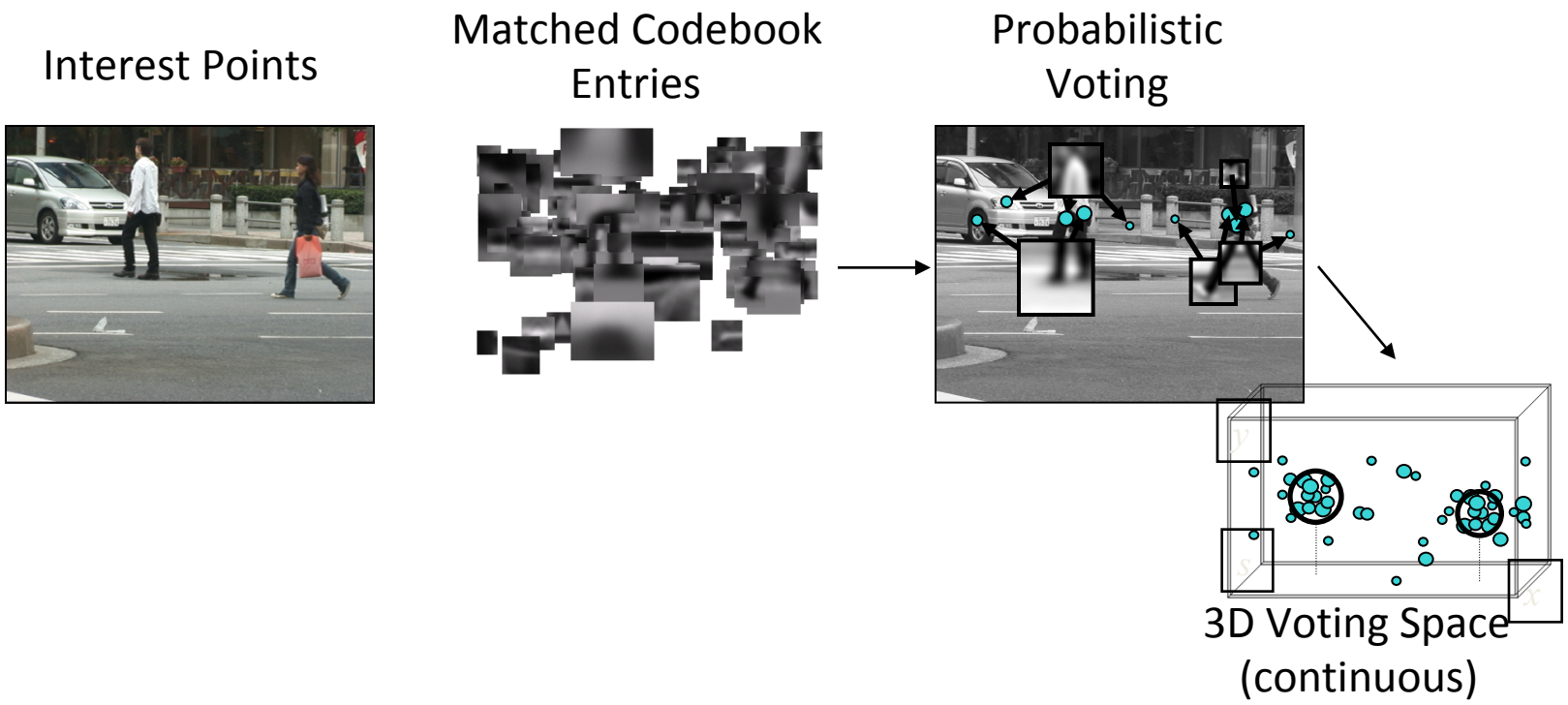


Each window is separately classified



Generating hypotheses

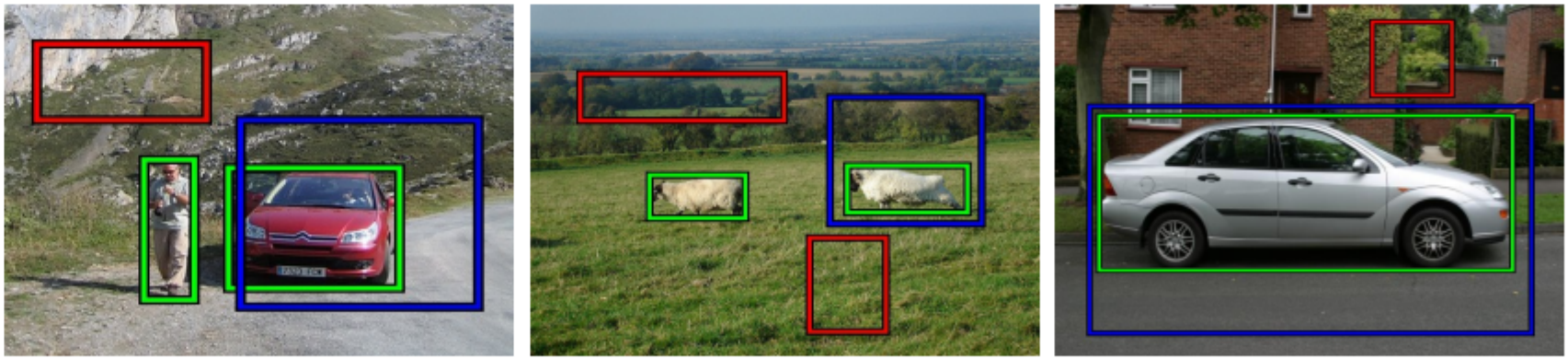
2. Voting from patches/keypoints



ISM model by Leibe et al.

Generating hypotheses

3. Region-based proposal
   - Learn to generate category-independent regions/boxes that have object-like properties.
   - Let object detector search over “proposals”, not exhaustive sliding windows



Alexe et al. Measuring the objectness of image windows, PAMI 2012

Generating hypotheses

Alexe et al. Measuring the objectness of image windows, PAMI 2012

Generating hypotheses

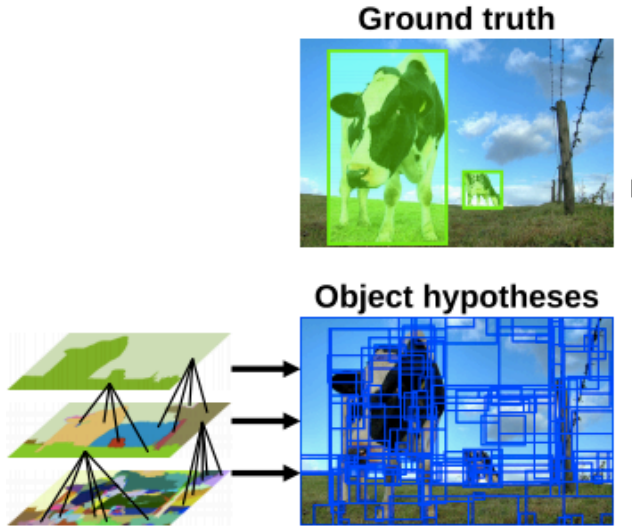
3. Region-based proposal

Endres Hoiem 2010

Generating hypotheses

3. Region-based proposal

Input Image

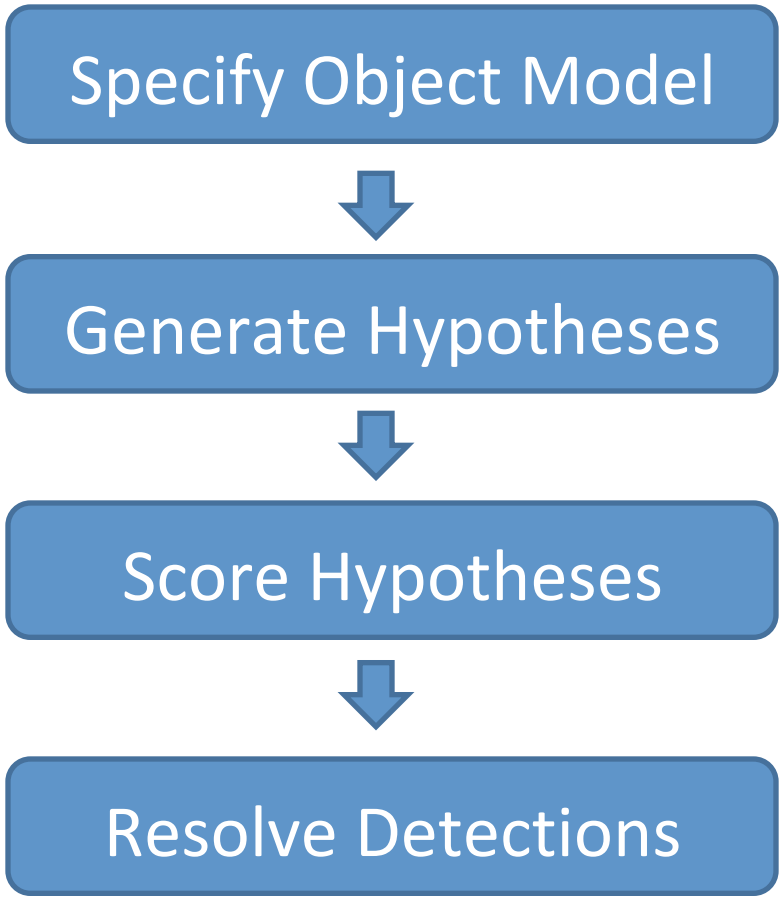


Use hierarchical segmentation: start with small *superpixels* and merge based on diverse cues

Used by R-CNN, Fast R-CNN.

J. Uijlings, K. van de Sande, T. Gevers, and A. Smeulders,
Selective Search for Object Recognition, IJCV 2013

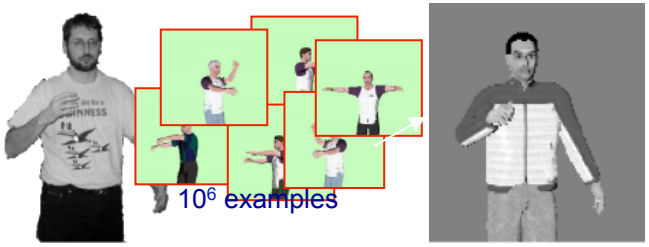
General Process of Object Detection



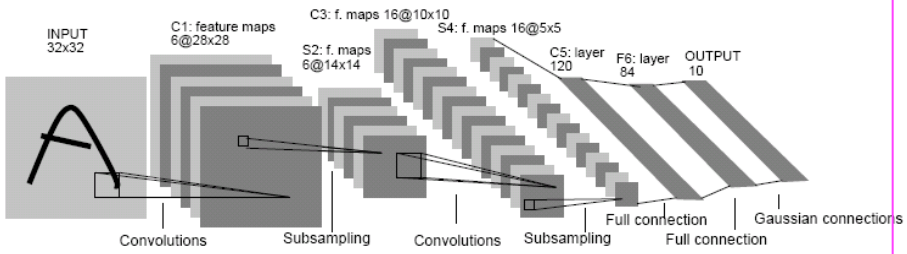
Different classifiers

Discriminative classifier construction

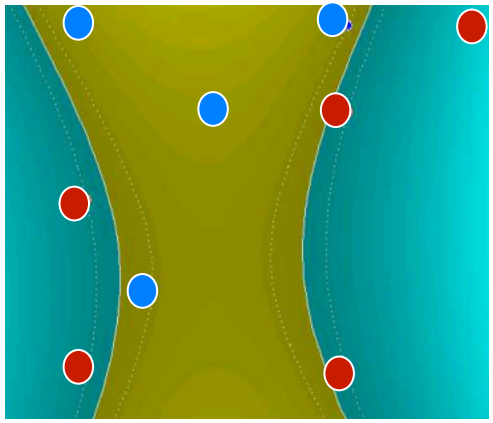
Nearest neighbor



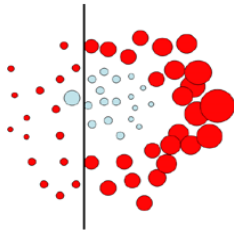
Neural networks



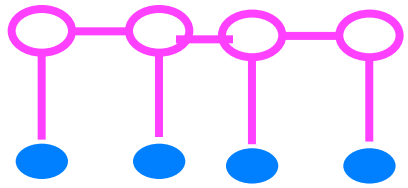
Support Vector Machines



Boosting

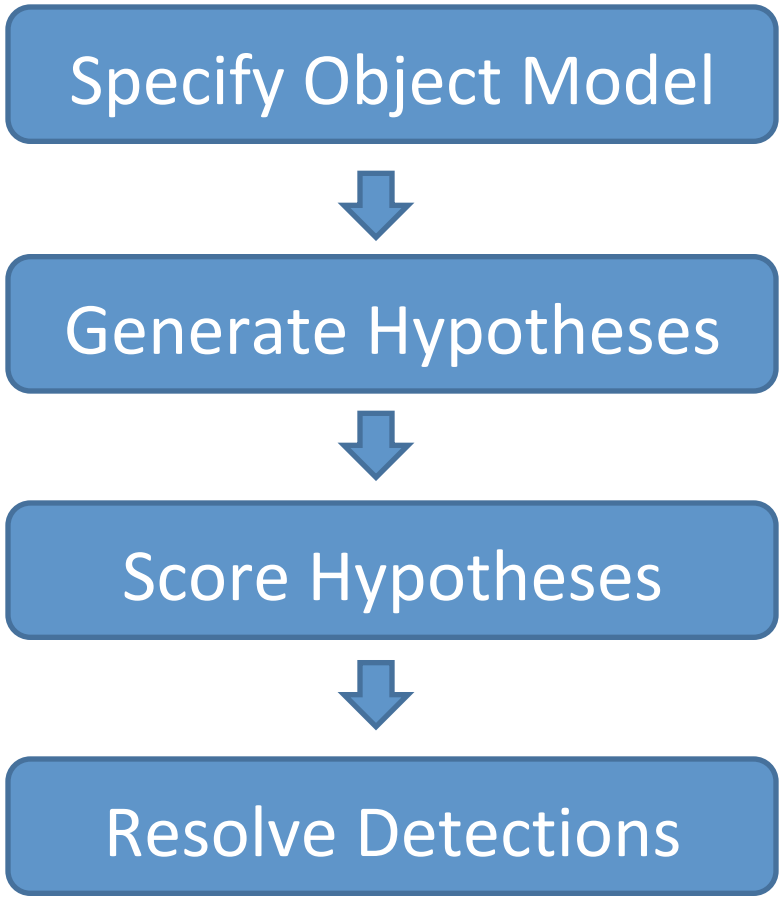


Conditional Random Fields



Slide credit Antonio Torralba

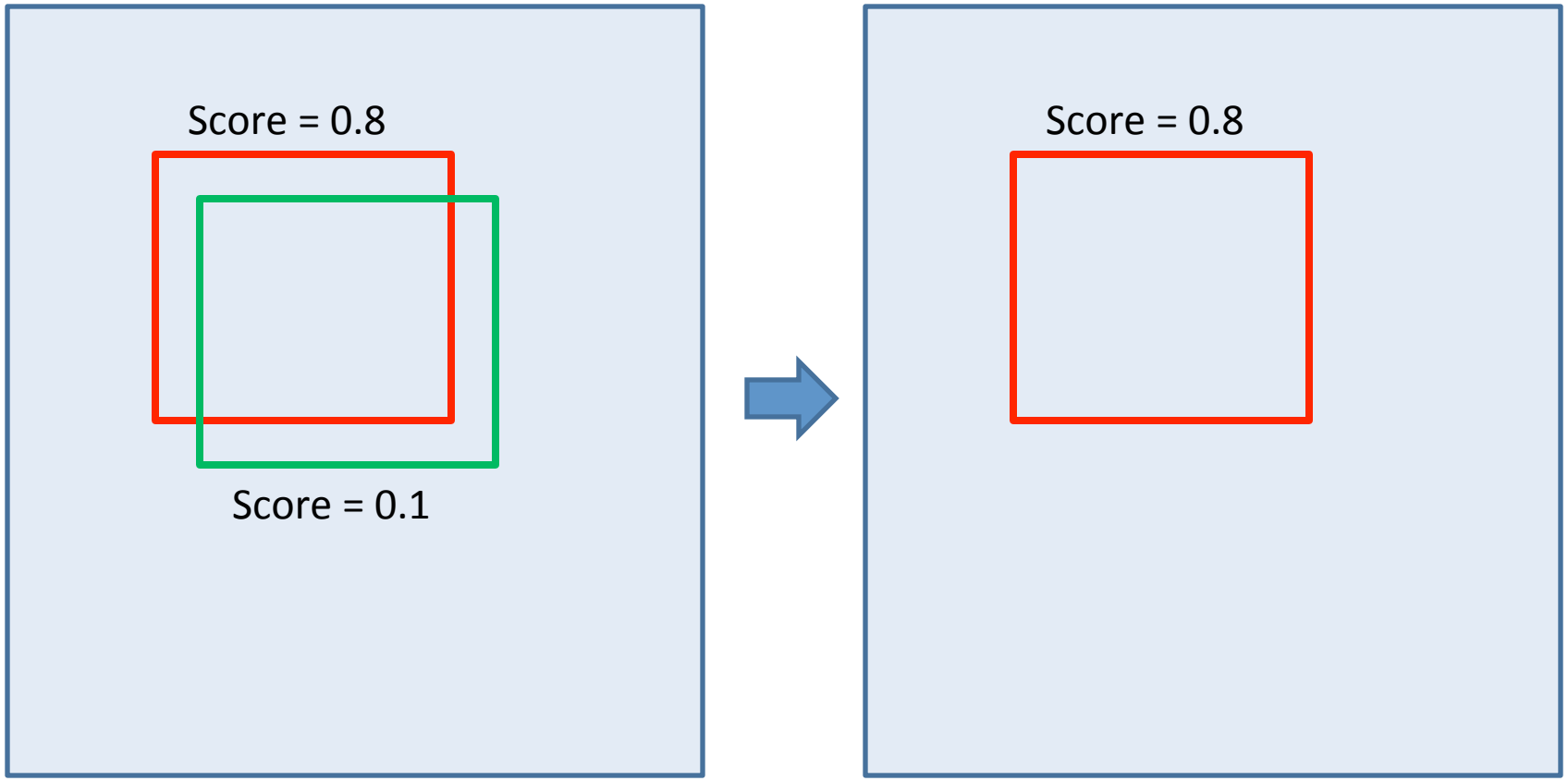
General Process of Object Detection



Optionally, rescore each proposed object based on whole set

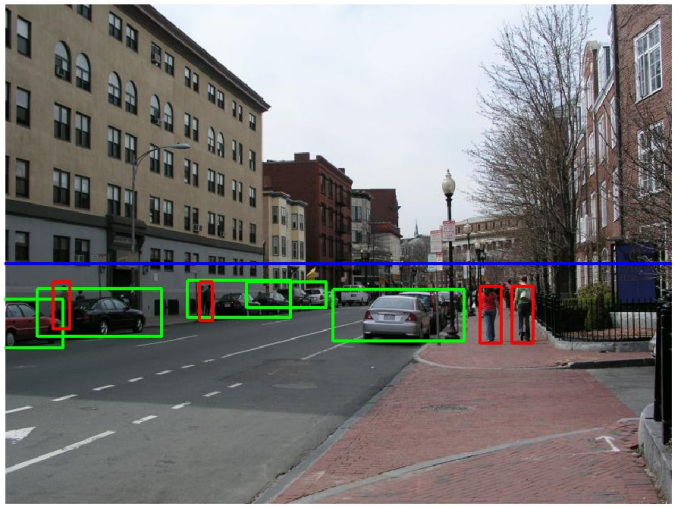
Resolving detection scores

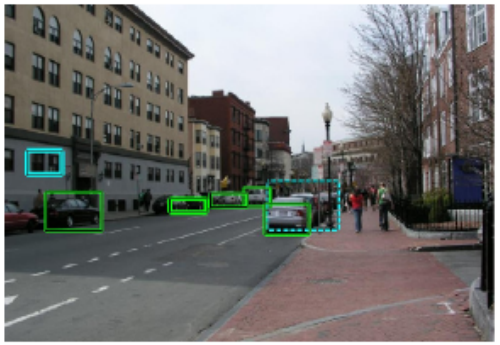
1. Non-max suppression



Resolving detection scores

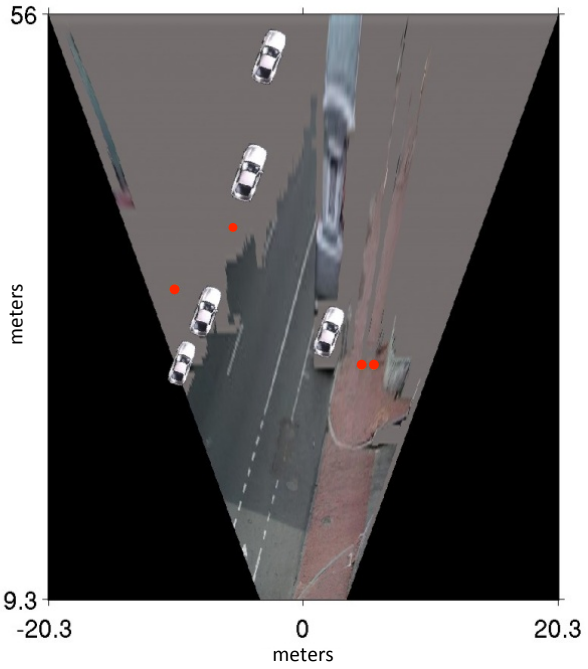
2. Context/reasoning





(g) Car Detections: Local

(h) Ped Detections: Local



Hoiem et al. 2006

Design challenges

- How to efficiently search for likely objects
  - Sliding windows require searching hundreds of thousands of positions and scales
- Feature design and scoring
  - How should appearance be modeled? What features correspond to the object?
- How to deal with different viewpoints?
  - Often train different models for a few different viewpoints
- Implementation details
  - Window size
  - Aspect ratio
  - Translation/scale step size
  - Non-maxima suppression

Histograms of oriented gradients (HOG)

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

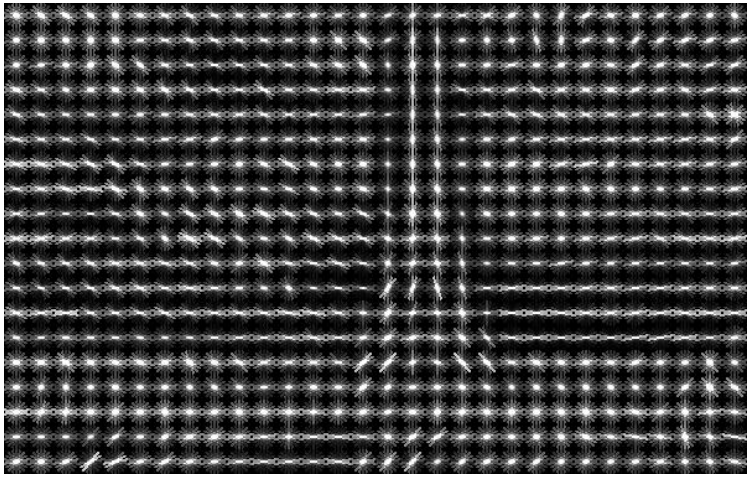


Image credit: N. Snavely

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

Pedestrian detection with HOG

- Train a pedestrian template using a linear support vector machine

positive training examples

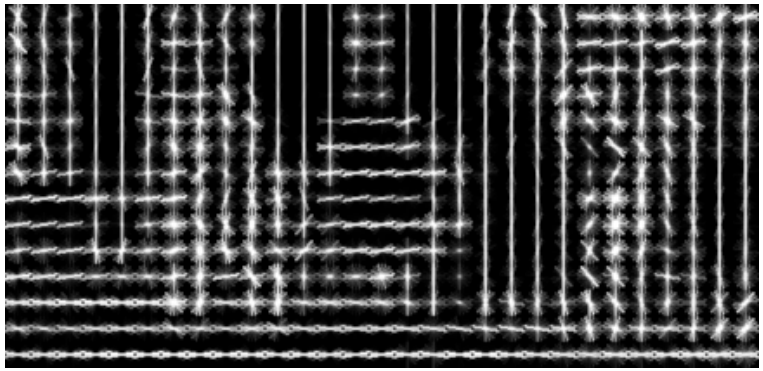
negative training examples

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

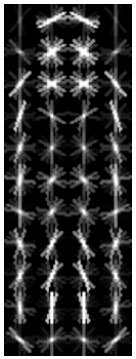
Pedestrian detection with HOG

- Train a pedestrian template using a linear support vector machine
- At test time, convolve feature map with template
- Find local maxima of response
- For multi-scale detection, repeat over multiple levels of a HOG *pyramid*

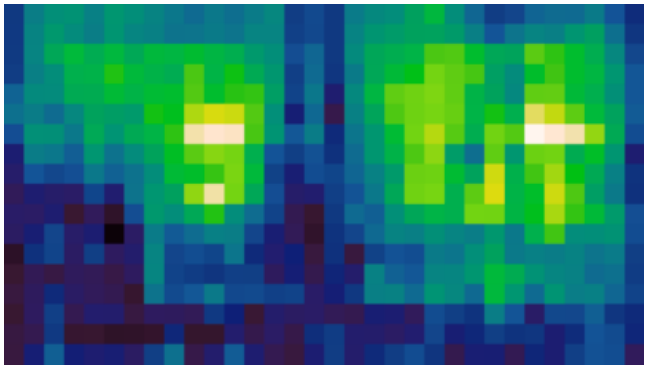
HOG feature map



Template



Detector response map

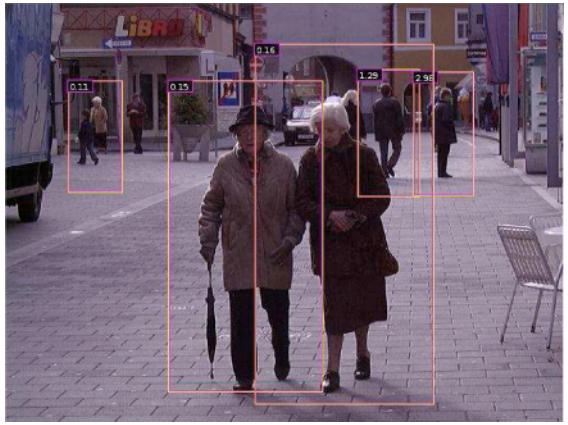


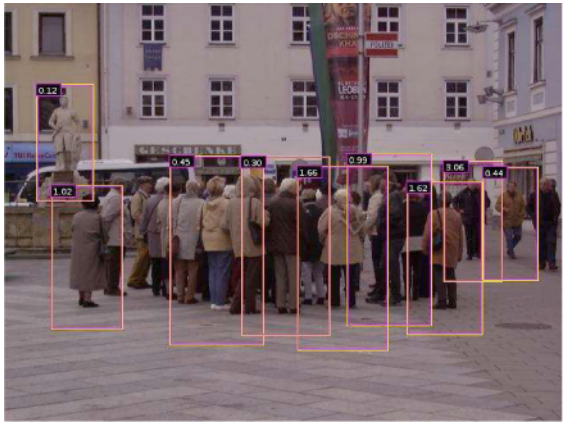
N. Dalal and B. Triggs, Histograms of Oriented Gradients for Human Detection, CVPR 2005

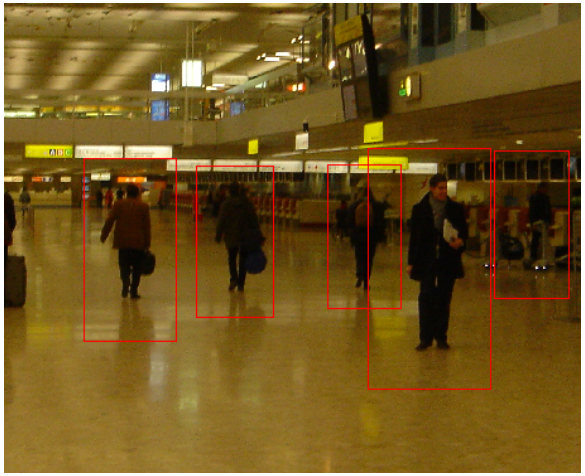
Example detections

[Dalal and Triggs, CVPR 2005]

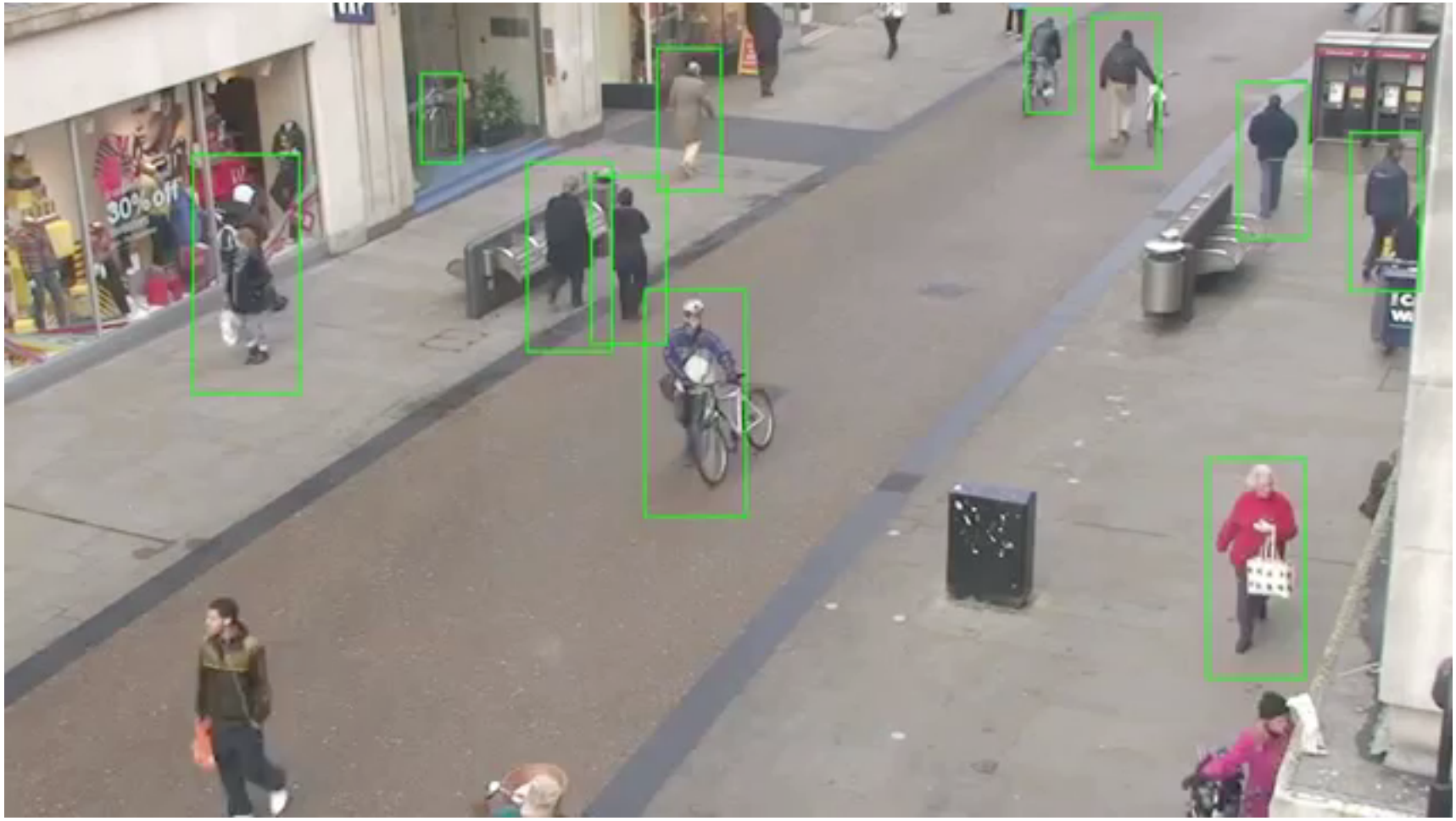
Example detections








Example detections in video



Discriminative part-based models

- Single rigid template usually not enough to represent a category
  - Many objects (e.g. humans) are articulated, or have parts that can vary in configuration

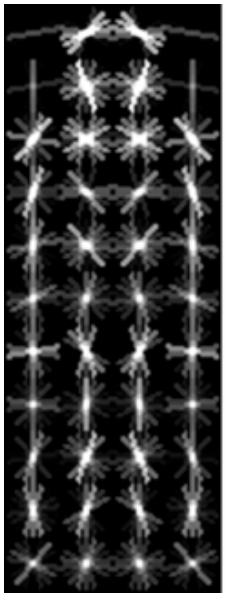
  - Many object categories look very different from different viewpoints, or from instance to instance

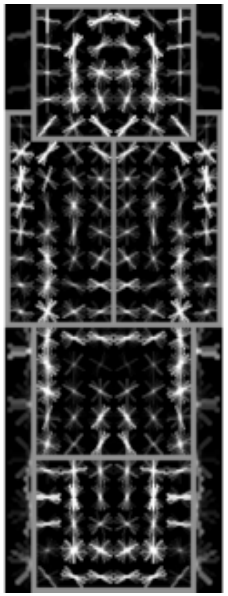
Discriminative part-based models

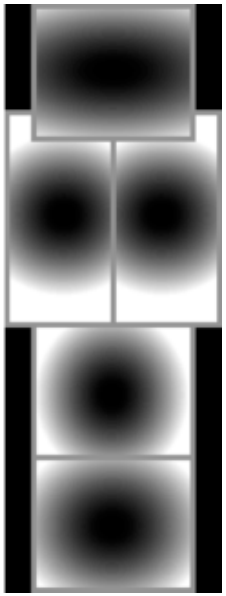
Root filter

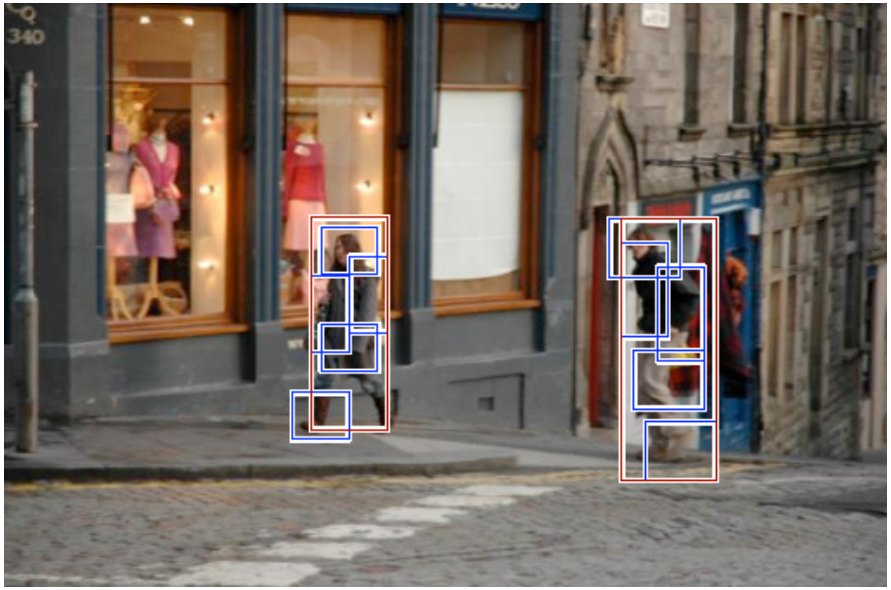
Part filters

Deformation weights





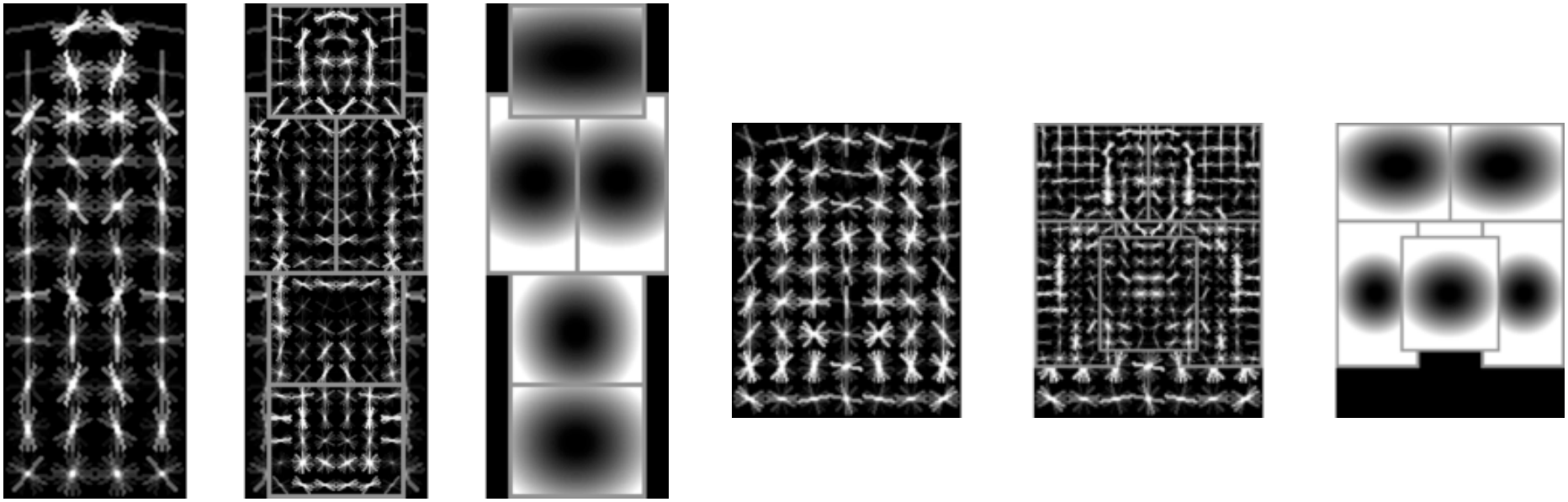




P. Felzenszwalb, R. Girshick, D. McAllester, D. Ramanan,
Object Detection with Discriminatively Trained Part Based Models, PAMI 32(9), 2010

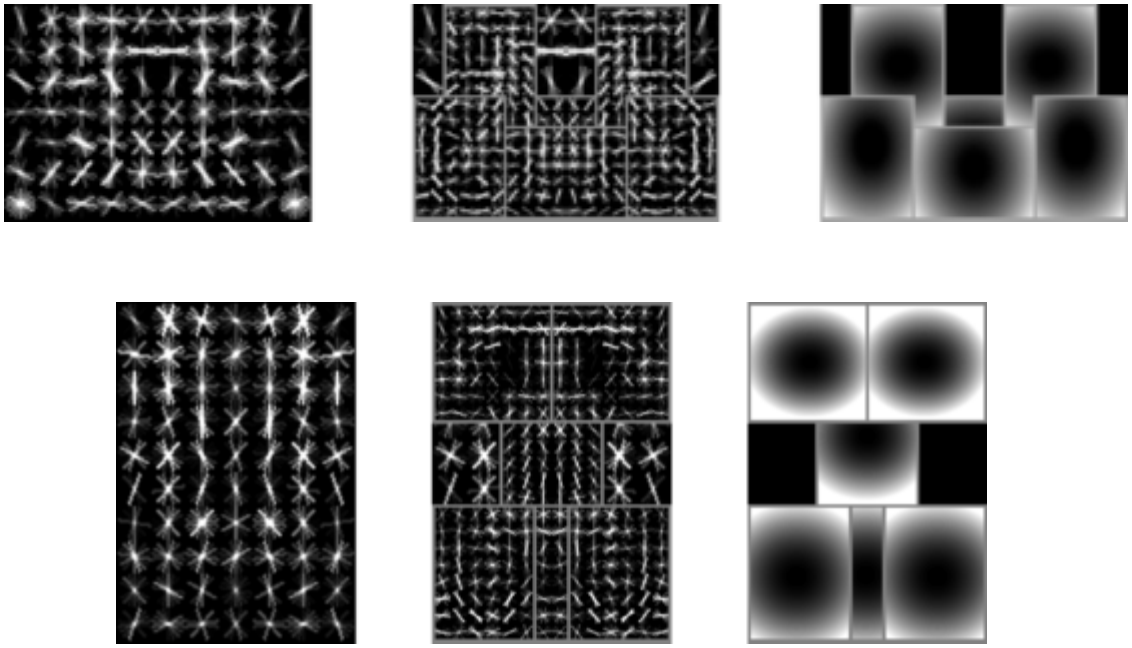
Discriminative part-based models

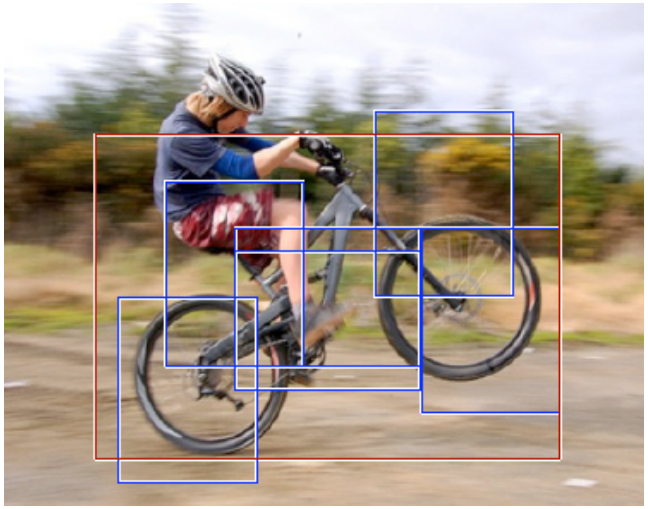
Multiple components



P. Felzenszwalb, R. Girshick, D. McAllester, D. Ramanan,
Object Detection with Discriminatively Trained Part Based Models, PAMI 32(9), 2010

Discriminative part-based models



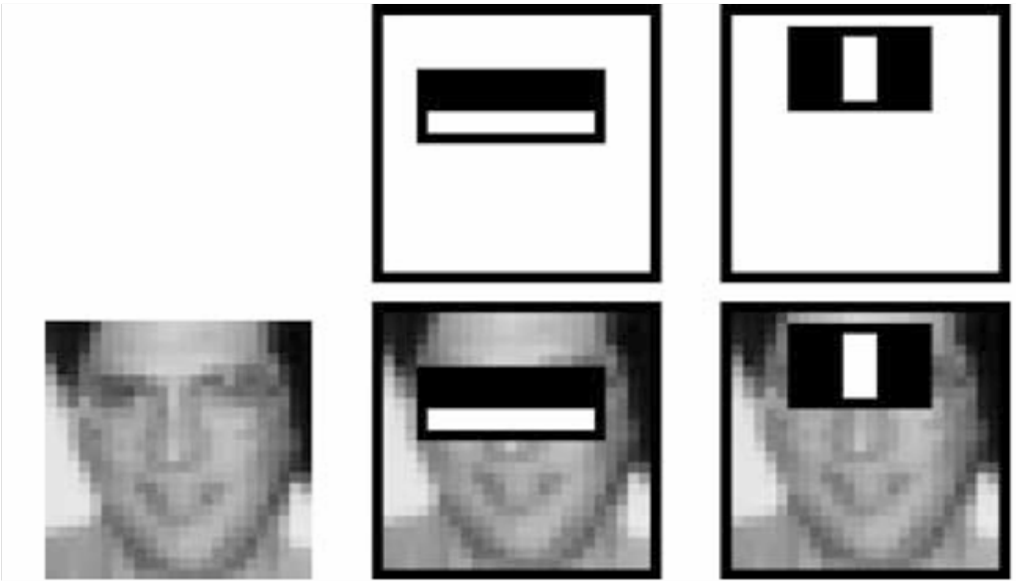


P. Felzenszwalb, R. Girshick, D. McAllester, D. Ramanan,
Object Detection with Discriminatively Trained Part Based Models, PAMI 32(9), 2010

Viola-Jones sliding window detector

Fast detection through two mechanisms

- Quickly eliminate unlikely windows
- Use features that are fast to compute



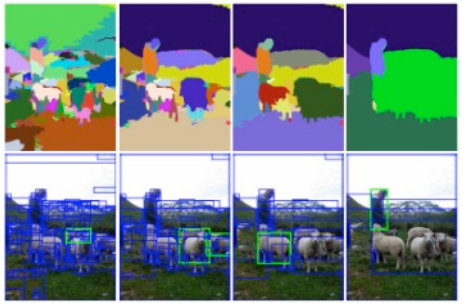
- Haar features
- Integral images
- AdaBoost

Viola and Jones. Rapid Object Detection using a Boosted Cascade of Simple Features (2001).

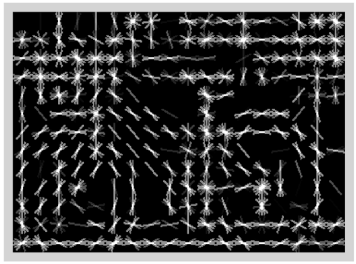
Summary: statistical templates

Propose Window → Extract Features → Classify → Post-process

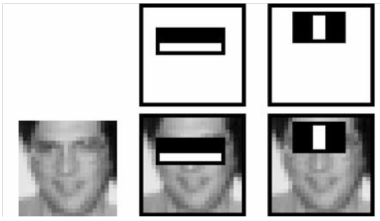
Sliding window: scan image pyramid



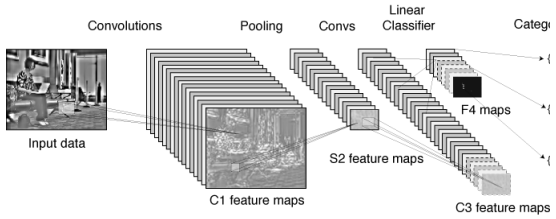
Region proposals: edge/ region-based, resize to fixed window



HOG



Fast randomized features



CNN features

SVM

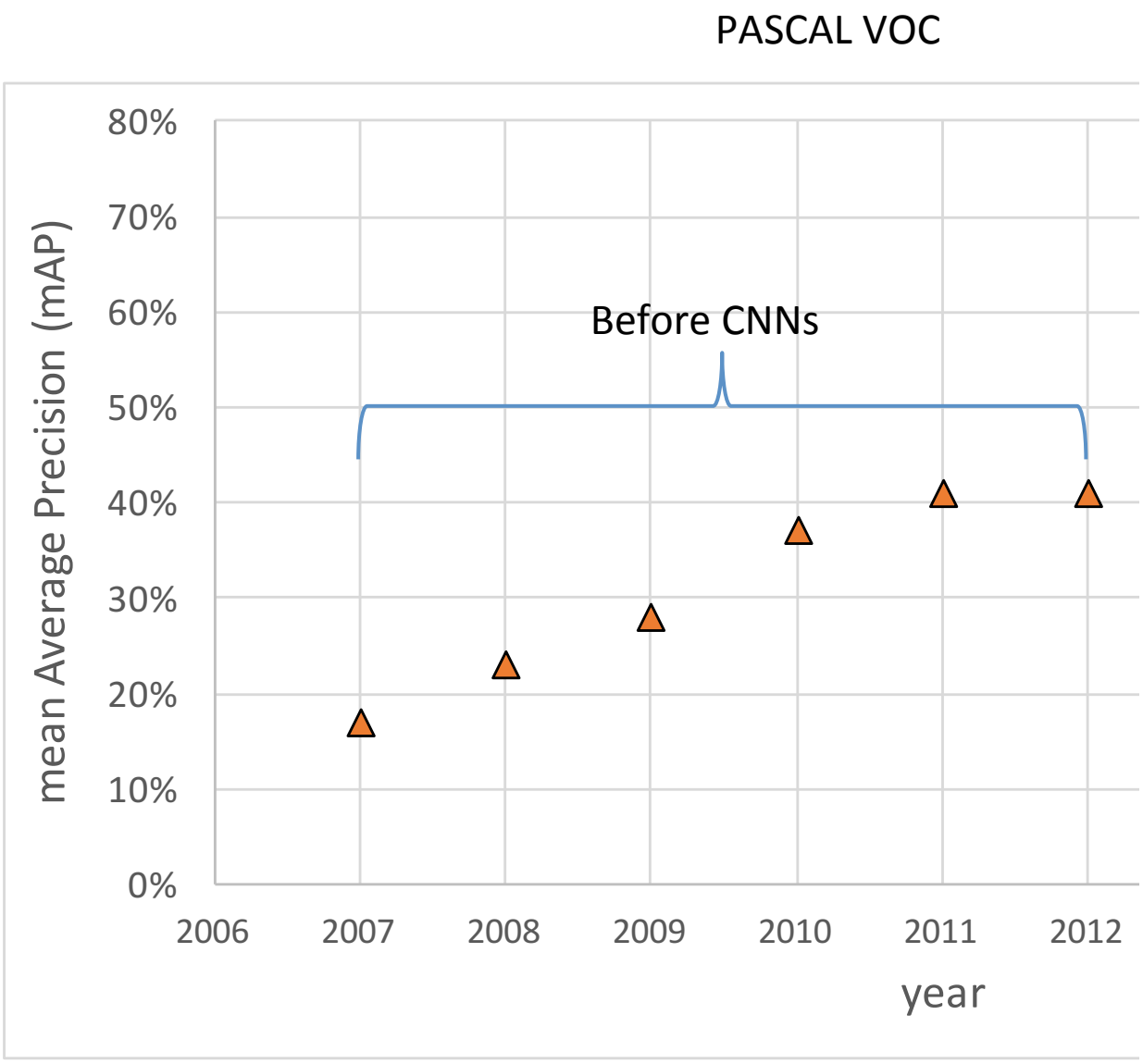
Boosted stubs

Neural network

Non-max suppression

Segment or refine localization

Progress on PASCAL detection



Progress on PASCAL detection