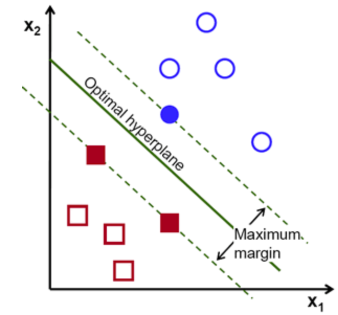
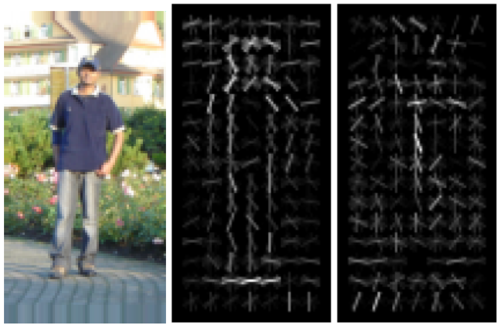
Object Detection

Ali Farhadi
Mohammad Rastegari
CSE 576

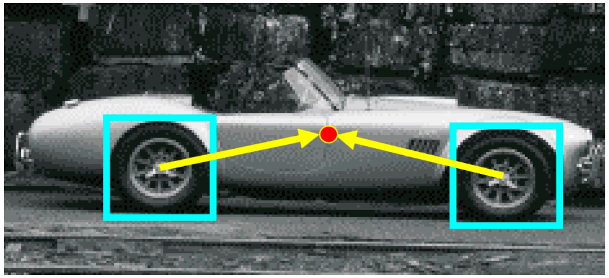
So Far



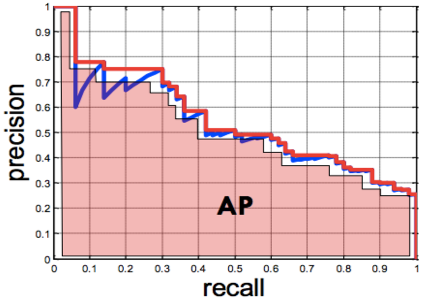
- Support Vector Machines (SVM)



- Pedestrian Detection by HOG

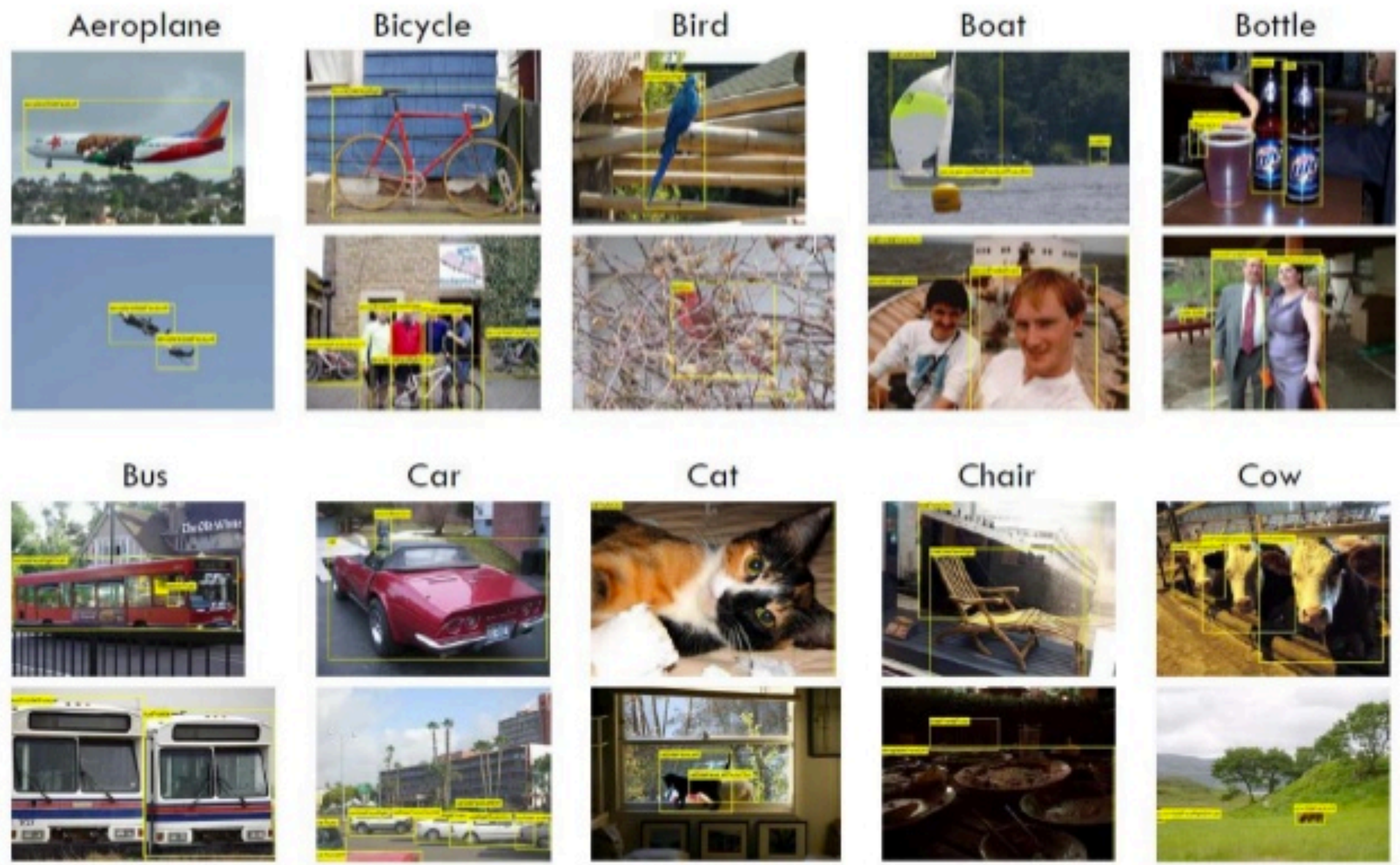


- Implicit Shape Models

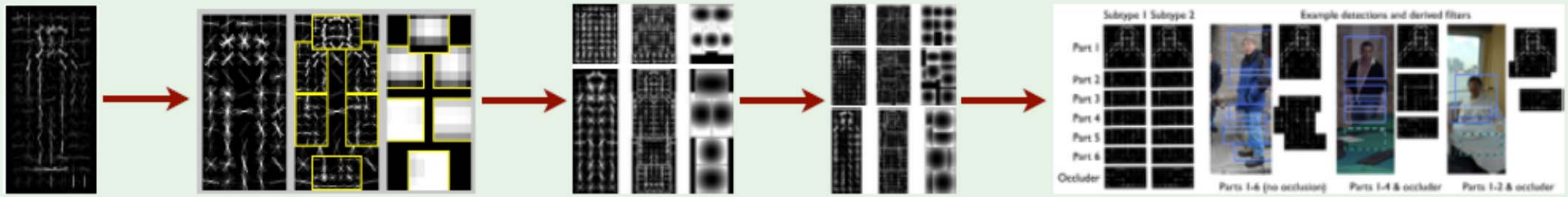


- Detector Evaluation

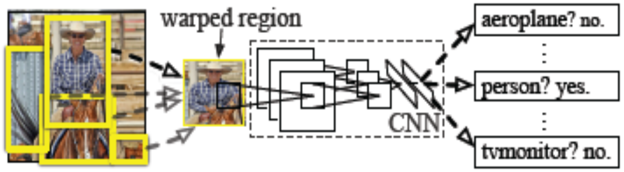
PASCAL VOC Challenge

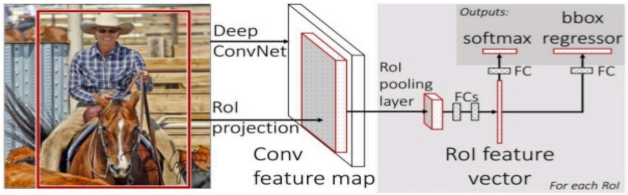


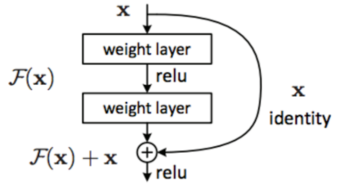
Person Detection in Pascal



AP	12%	27%	36%	45%	49%
	2005	2008	2009	2010	2011



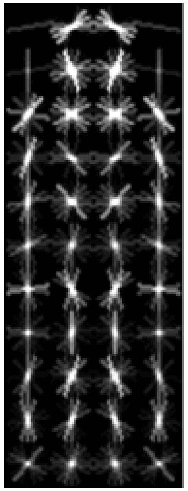


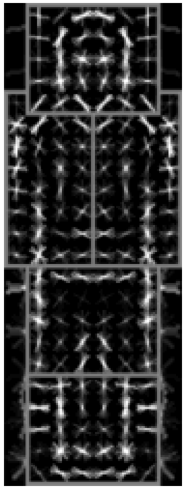


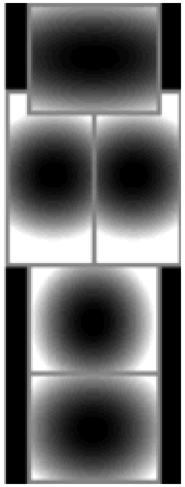
54%	58%	72%	%85	90%
2013	2014	2015	2015	2016

Deformable Part Models

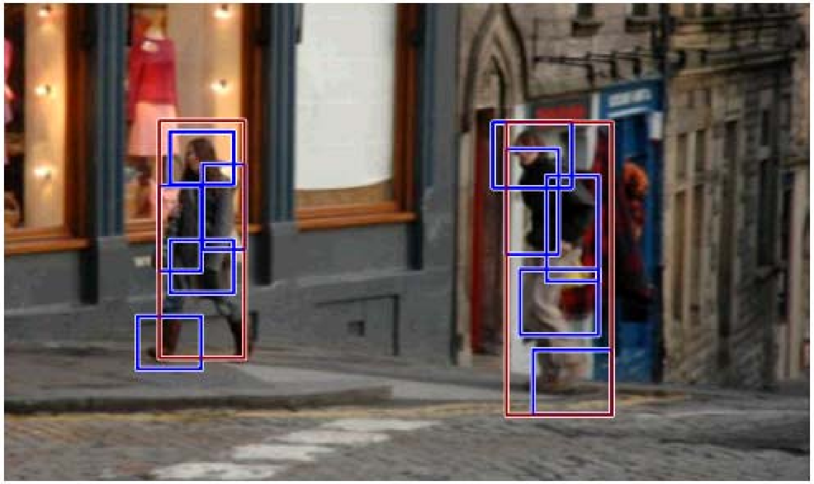
- Learn a part-based model:


Coarse
root filter


Higher resolution
part filters

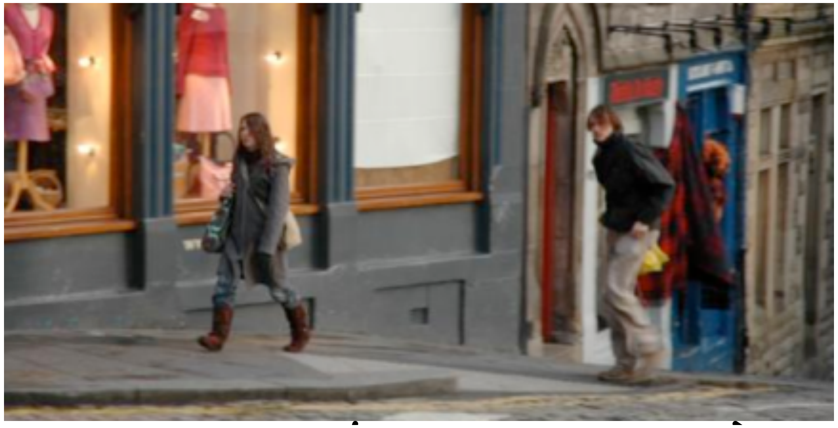

Deformation
models

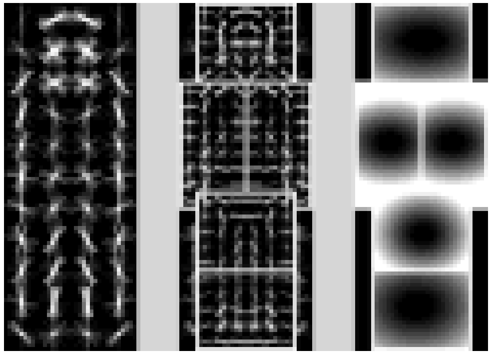
- Assumption: Number of parts → 5

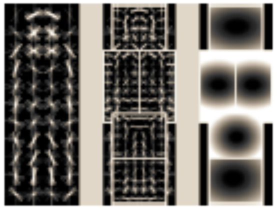

Example detection result

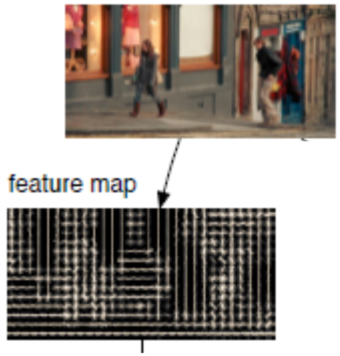
[Felzenszwalb et al, 2008]

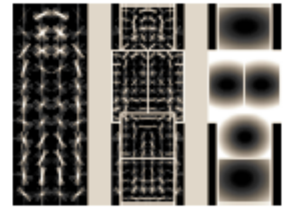
Detection

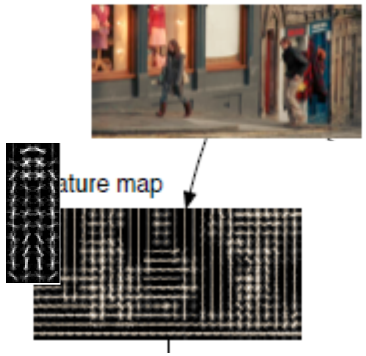


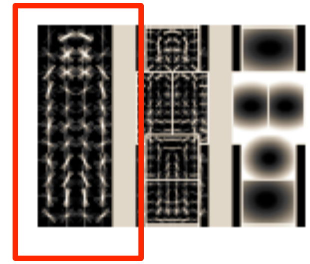


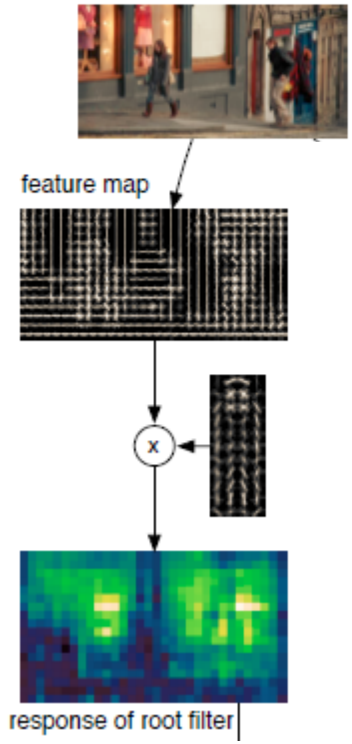


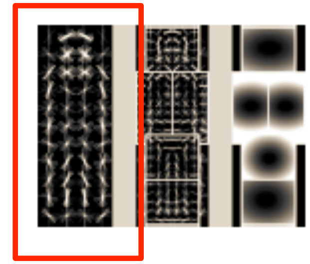

feature map

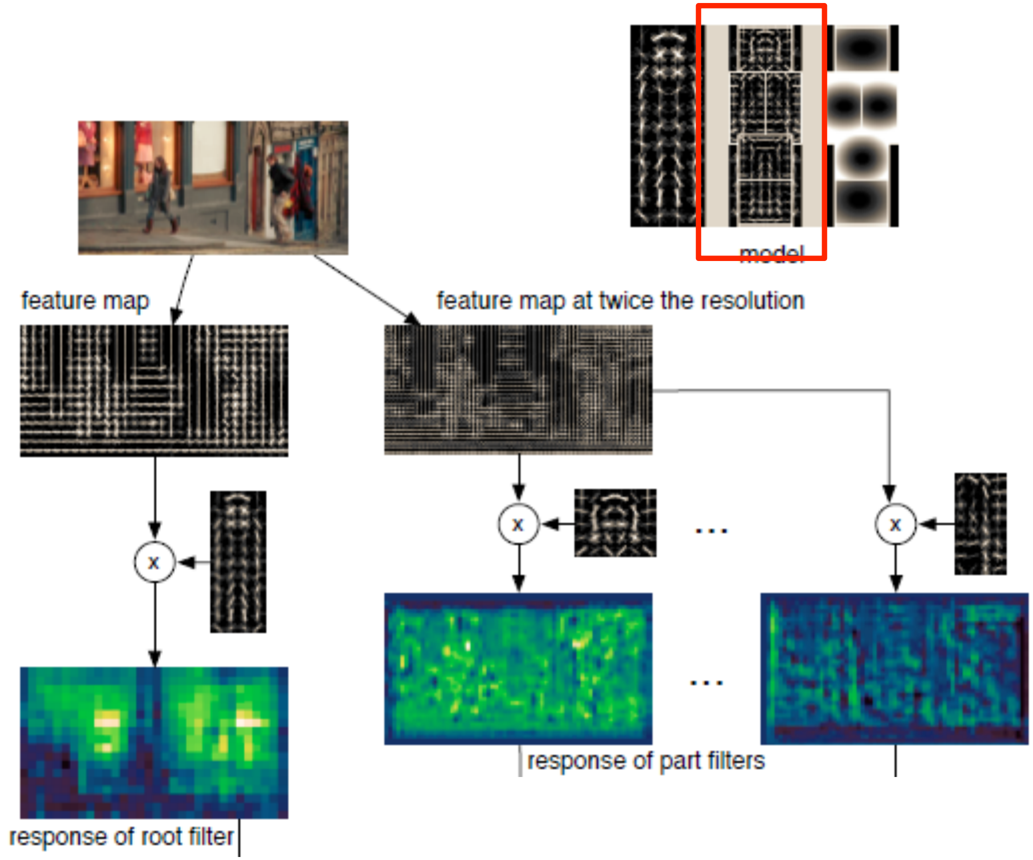



ature map

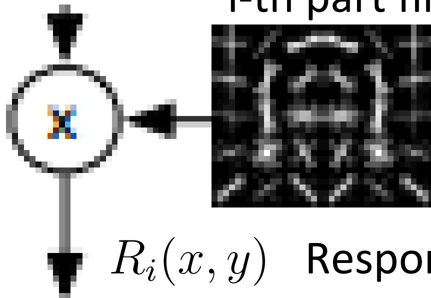



feature map
x
response of root filter

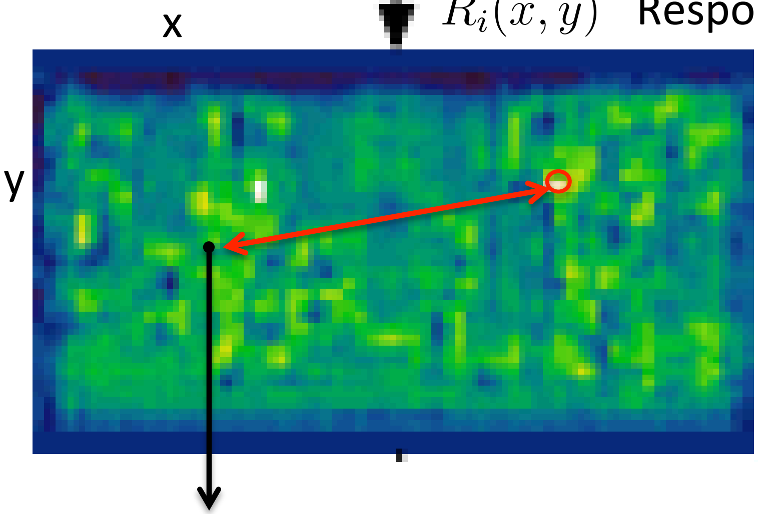


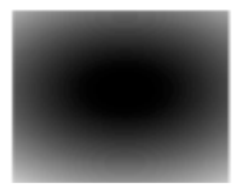

model
feature map
feature map at twice the resolution
x
x
x
...
...
response of root filter
response of part filters

i-th part filter



$R_i(x, y)$ Response to the i-th part filter at location (x,y)

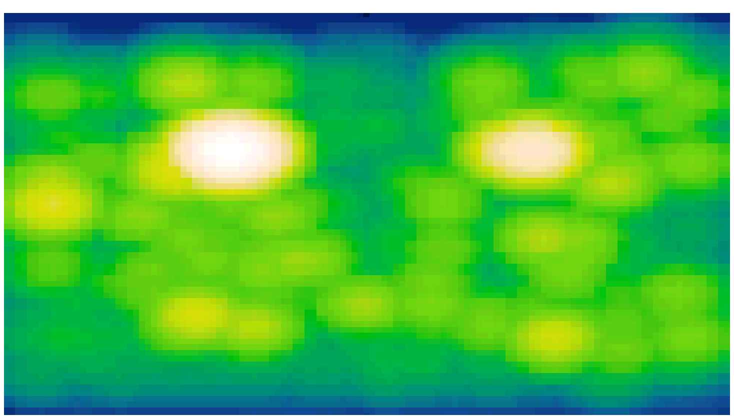




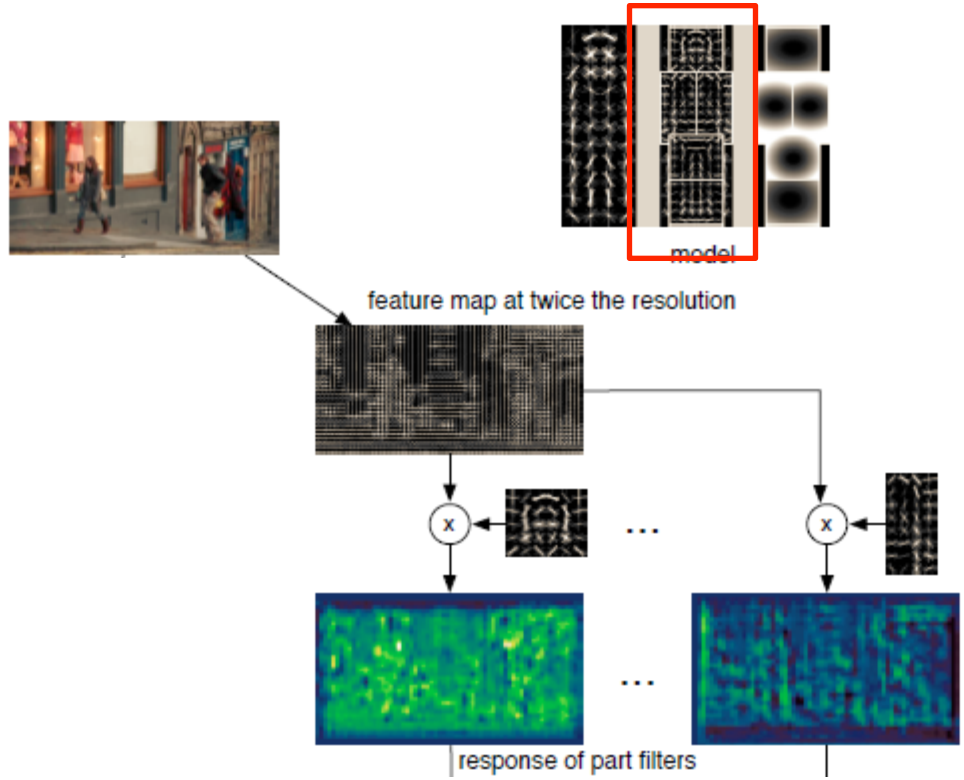
$$d_i = (0, 0, 1, 1)$$
$$\phi_d(dx, dy) = (dx, dy, dx^2, dy^2)$$

Is there a person at location (x,y)?

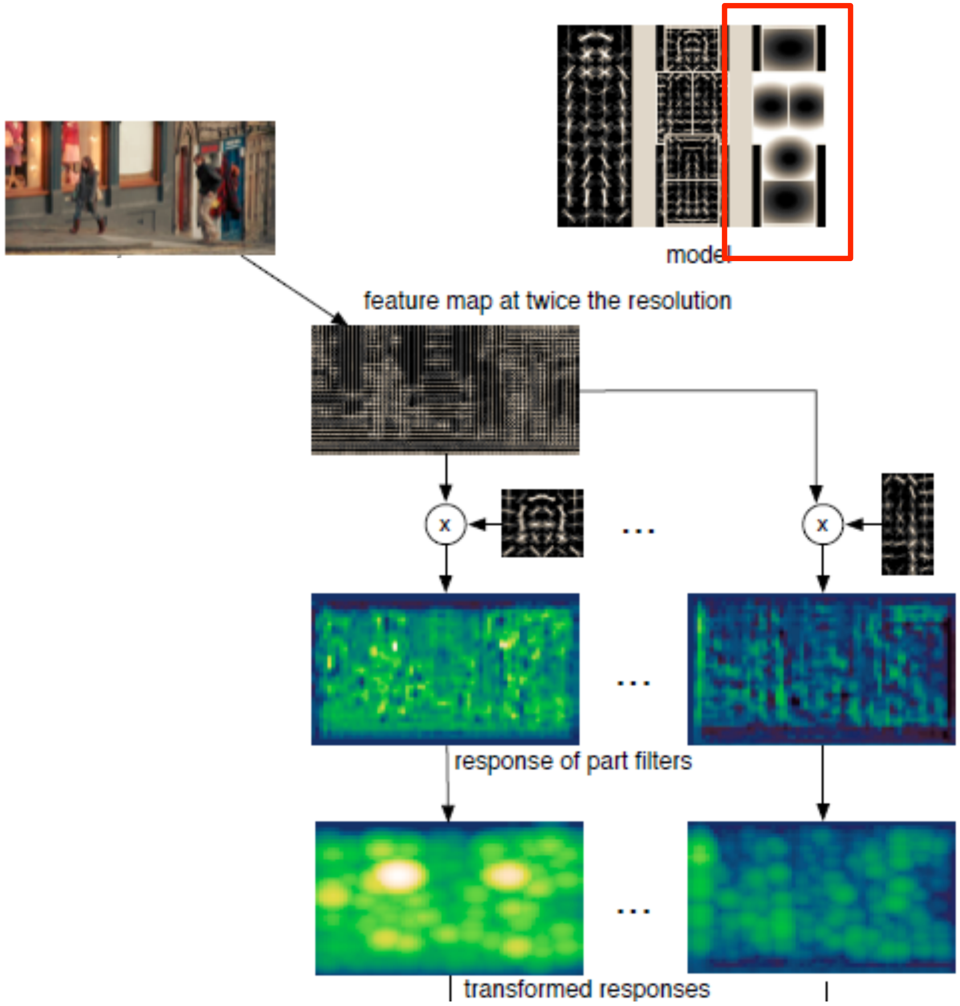
$$D_i(x, y) = \max_{dx,dy} \quad R_i(x + dx, y + dy) - d_i . \phi_d(dx, dy)$$

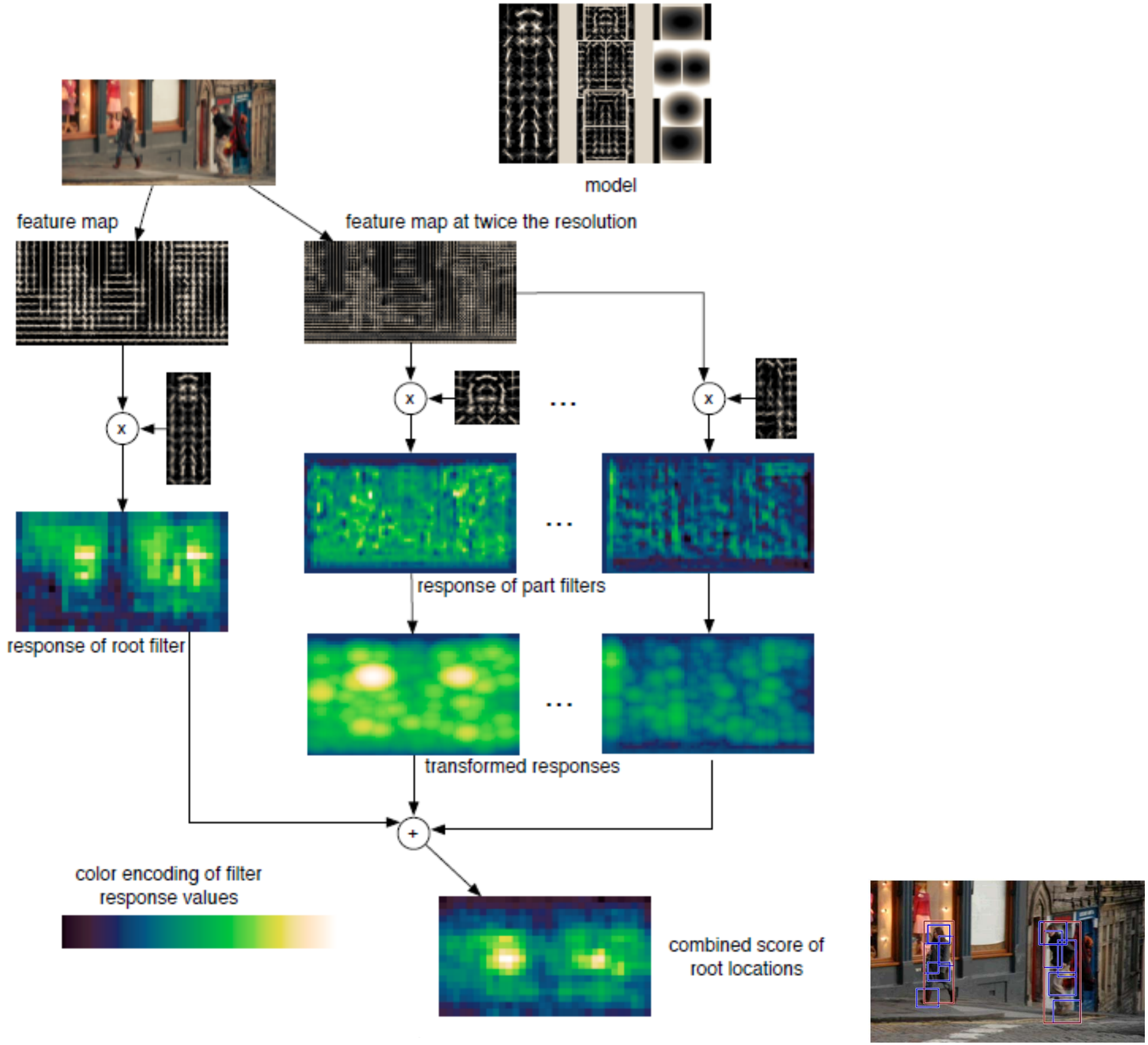


- Naïve search → O(N²)
- Generalized Distance Transform → O(N) [Felzenszwalb et al, 2004]

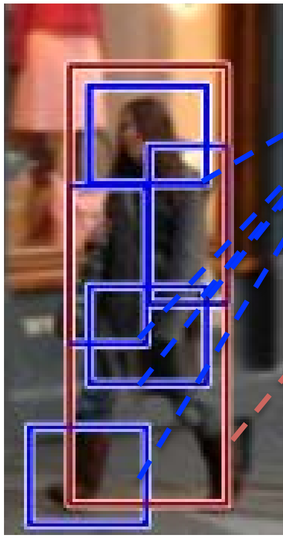


model
feature map at twice the resolution
x
x
response of part filters


model
feature map at twice the resolution
x
x
response of part filters
transformed responses


model
feature map
feature map at twice the resolution
x
x
...
x
...
response of part filters
response of root filter
...
transformed responses
+
color encoding of filter
response values
combined score of
root locations

Detection Score



$p_i = (x_i, y_i)$

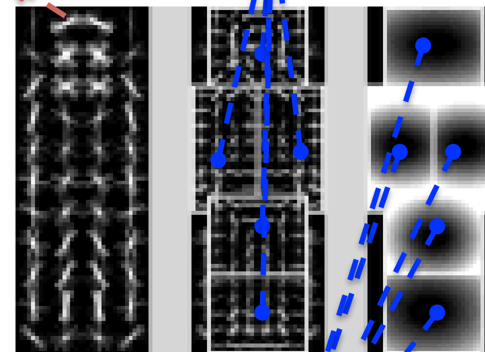
$p_1, p_2, \ldots, p_k$ = location of the parts

p_0 = location of the root

$\Delta_i = (dx_i, dy_i) = p_i - p_0$

$f_1, f_2, \ldots, f_k$ = parts filters

f_0 = root filter



$d_1, d_2, \ldots, d_k$ = deformation parameters

$\phi_h(p_i)$ = HOG feature at part p_i

$\phi_d(\Delta_i) = (dx_i, dy_i, dx_i^2, dy_i^2)$

$z = (p_0, p_1, \ldots, p_k)$

$\text{score}(z) = f_0.\phi_h(p_0) + f_1.\phi_h(p_1) + \cdots + f_k.\phi_h(p_k) - d_1\phi_d(\Delta_1) - d_2\phi_d(\Delta_2) - \cdots - d_k\phi_d(\Delta_k)$

Data Term | Spatial Term

Training

$$\text{score}(z) = f_0.\phi_h(p_0) + f_1.\phi_h(p_1) + \cdots + f_k.\phi_h(p_k) - d_1\phi_d(\Delta_1) - d_2\phi_d(\Delta_2) - \cdots - d_k\phi_d(\Delta_k)$$

Model Parameters

Need to be trained

$$w = [f_0, f_1, \ldots, f_k, d_1, d_2, \ldots, d_k]$$

$$x = [\phi_h(p_0), \phi_h(p_1), \ldots, \phi_h(p_k), -\phi_d(\Delta_1), -\phi_d(\Delta_2) - \cdots - \phi_d(\Delta_k)]$$

$$\text{score}(z) = w.x$$

W is a classifier in the space of *x*

Can we train *w* by SVM?

$$\min_w \frac{1}{2}\|w\|^2$$

$$\forall j \;\; y_j(w.x) > 1 \quad y_j \in \{-1+1\}$$

Training

$$z = (p_0, p_1, \ldots, p_k)$$

We do not have any information
about the location of the parts in train data

Z is latent

Latent-SVM: $$\min_{w,z} \frac{1}{2} \|w\|^2$$

$$\forall j \ \mathrm{score}(z) y_j > 1 \quad y_j \in \{-1 + 1\}$$

Latent SVM

$$\min_{w,z} \frac{1}{2} \|w\|^2$$

$$\forall j \ \text{score}(z) y_j > 1 \quad y_j \in \{-1 + 1\}$$

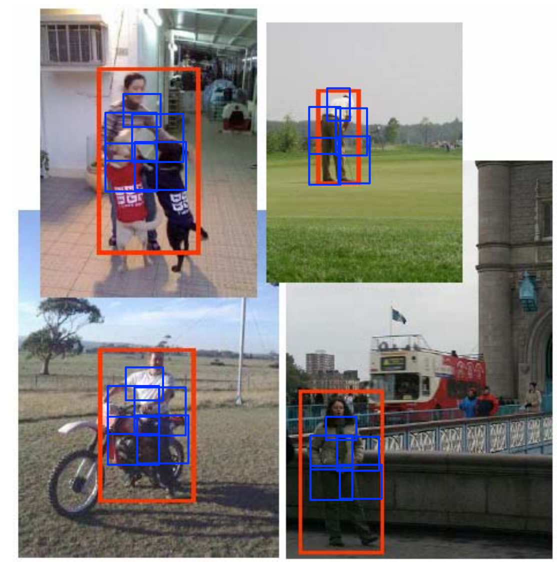
- Loop until no change in ***z,w***
  - Fix ***z*** , find ***w***
  - Fix ***w*** , find ***z***

Latent SVM

$$\min_{w,z} \frac{1}{2} \|w\|^2$$
$$\forall j \;\; \text{score}(z) y_j > 1 \quad y_j \in \{-1 + 1\}$$

- Loop until no change in ***z,w***
  - Fix ***z*** , find ***w***
  - Fix ***w*** , find ***z***

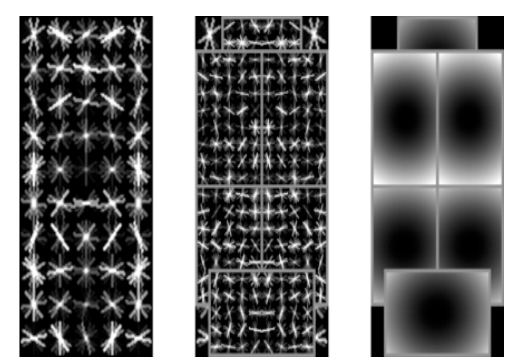
z



- Standard SVM

$$\min_{w} \frac{1}{2} \|w\|^2$$
$$\forall j \;\; y_j(w.x) > 1 \quad y_j \in \{-1 + 1\}$$

w



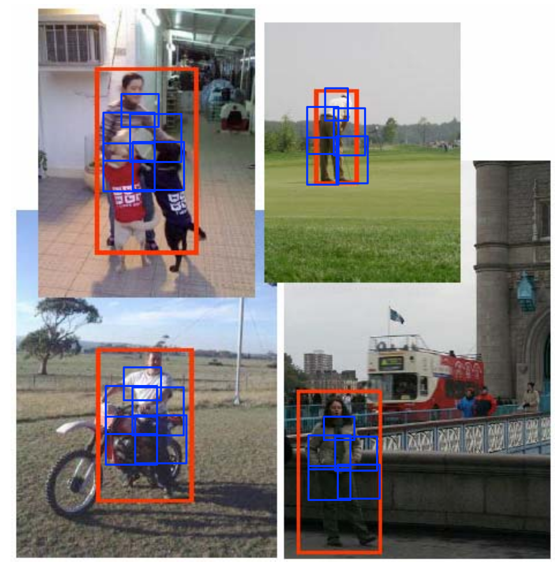
Latent SVM

$$\min_{w,z} \frac{1}{2} \|w\|^2$$

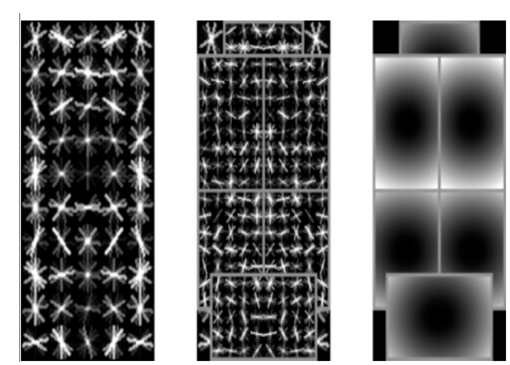
$$\forall j \ \text{score}(z) y_j > 1 \quad y_j \in \{-1 + 1\}$$

- Loop until no change in ***z,w***
  - Fix ***z*** , find ***w***
  - Fix ***w*** , find ***z***

z



w

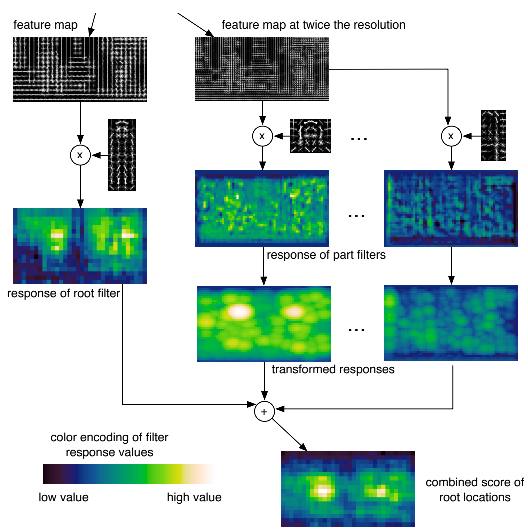


Latent SVM

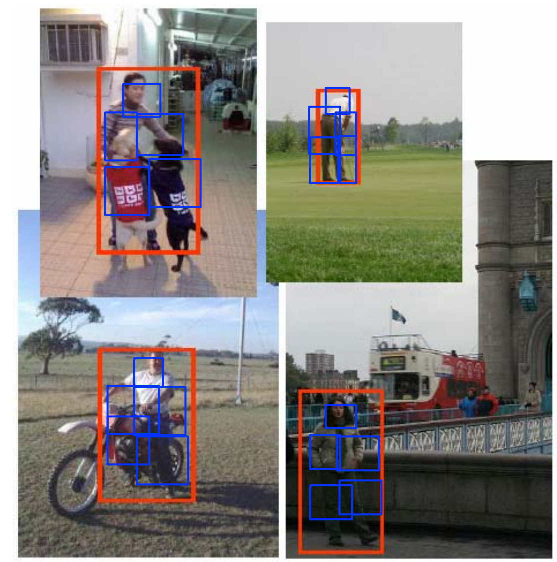
$$\min_{w,z} \frac{1}{2}\|w\|^2$$

$$\forall j \ \text{score}(z) y_j > 1 \quad y_j \in \{-1 + 1\}$$

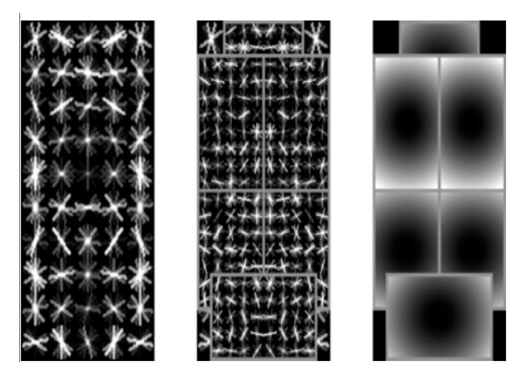
- Loop until no change in ***z,w***
  - Fix ***z*** , find ***w***
  - Fix ***w*** , find ***z***



z



w

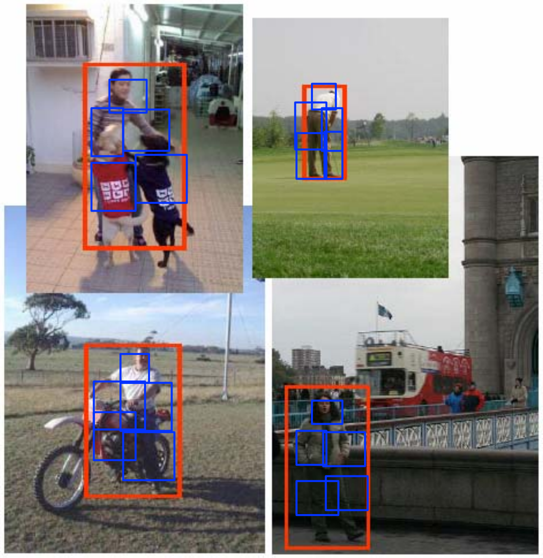


Latent SVM

$$\min_{w,z} \frac{1}{2} \|w\|^2$$
$$\forall j \ \text{score}(z) y_j > 1 \quad y_j \in \{-1 + 1\}$$

- Loop until no change in ***z,w***
  - Fix ***z*** , find ***w***
  - Fix ***w*** , find ***z***

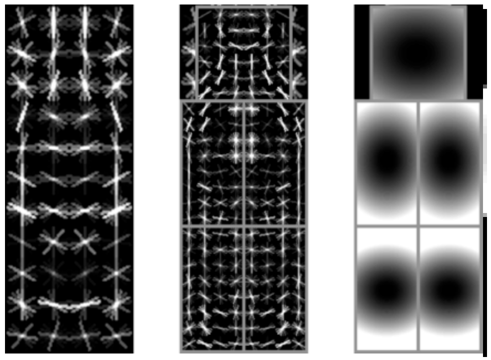
z



- Standard SVM

$$\min_{w} \frac{1}{2} \|w\|^2$$
$$\forall j \ y_j(w.x) > 1 \quad y_j \in \{-1 + 1\}$$

w



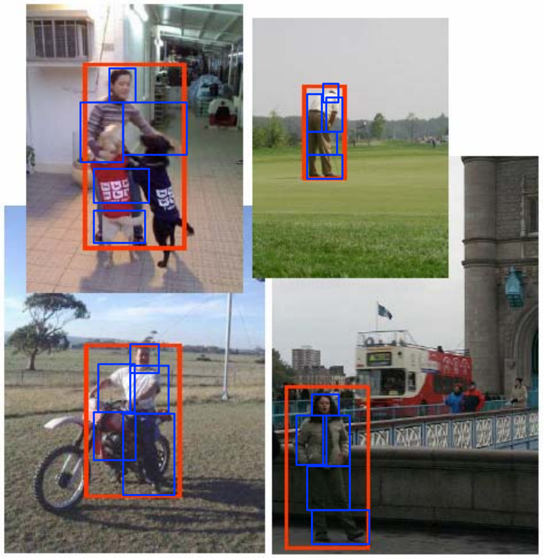
Latent SVM

$$\min_{w,z} \frac{1}{2} \|w\|^2$$

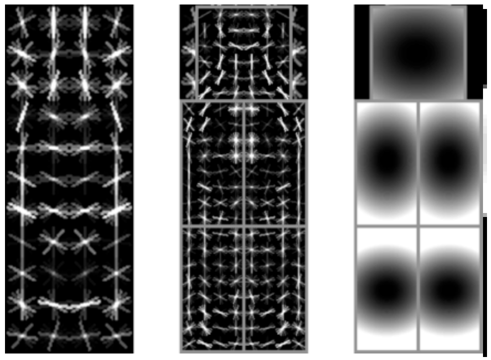
$$\forall j \ \text{score}(z) y_j > 1 \quad y_j \in \{-1 + 1\}$$

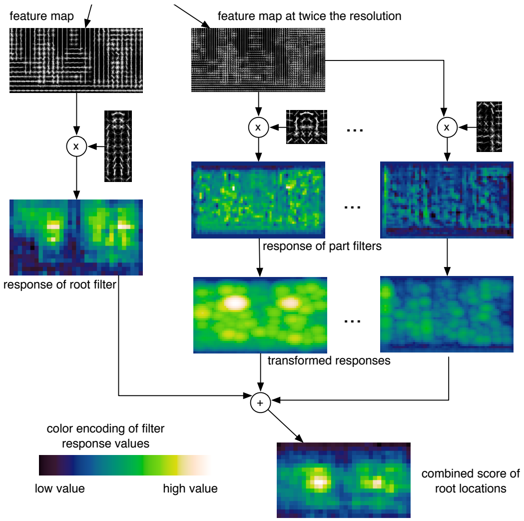
- Loop until no change in ***z,w***
  - Fix ***z*** , find ***w***
  - Fix ***w*** , find ***z***

z



w





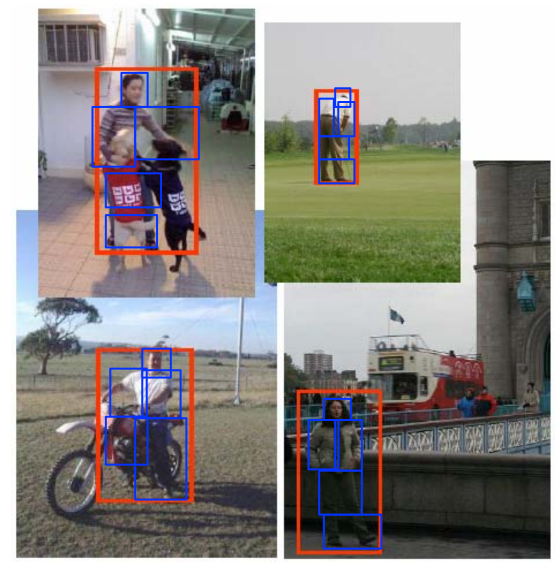
Latent SVM

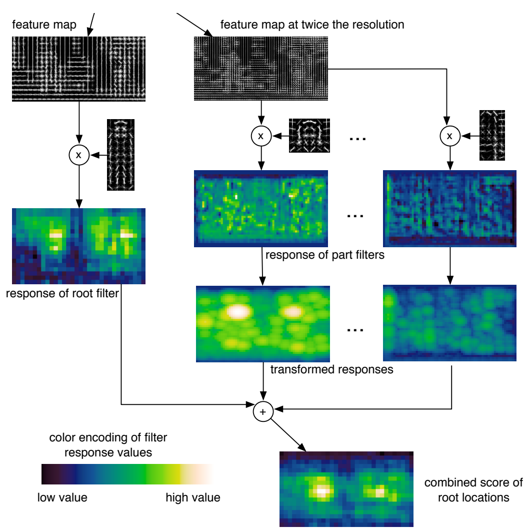
$$\min_{w,z} \frac{1}{2} \|w\|^2$$

$$\forall j \;\; \text{score}(z) y_j > 1 \quad y_j \in \{-1 + 1\}$$

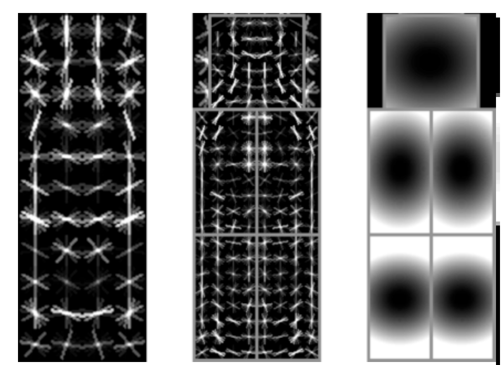
- Loop until no change in ***z,w***
  - Fix ***z*** , find ***w***
  - Fix ***w*** , find ***z***

z





w

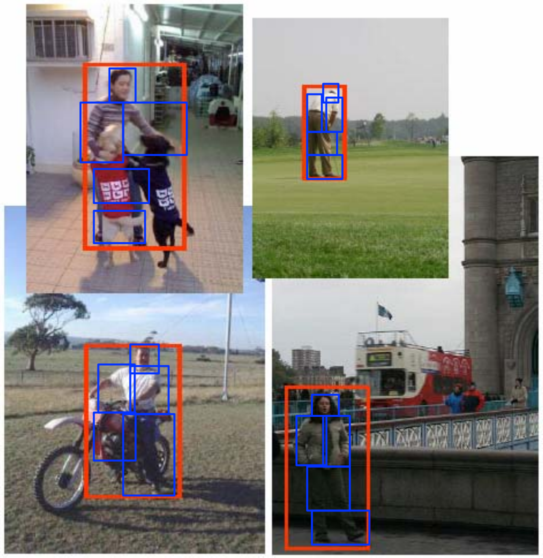


Latent SVM

$$\min_{w,z} \frac{1}{2} \|w\|^2$$
$$\forall j \;\; \text{score}(z) y_j > 1 \quad y_j \in \{-1 + 1\}$$

- Loop until no change in ***z,w***
  - Fix ***z*** , find ***w***
  - Fix ***w*** , find ***z***

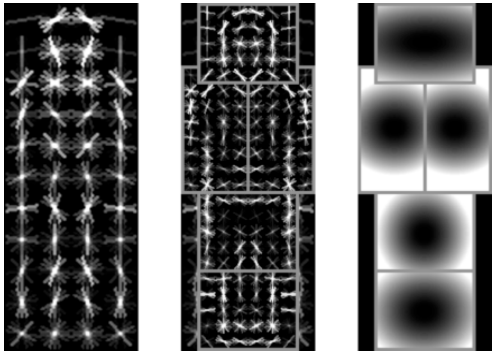
z



- Standard SVM

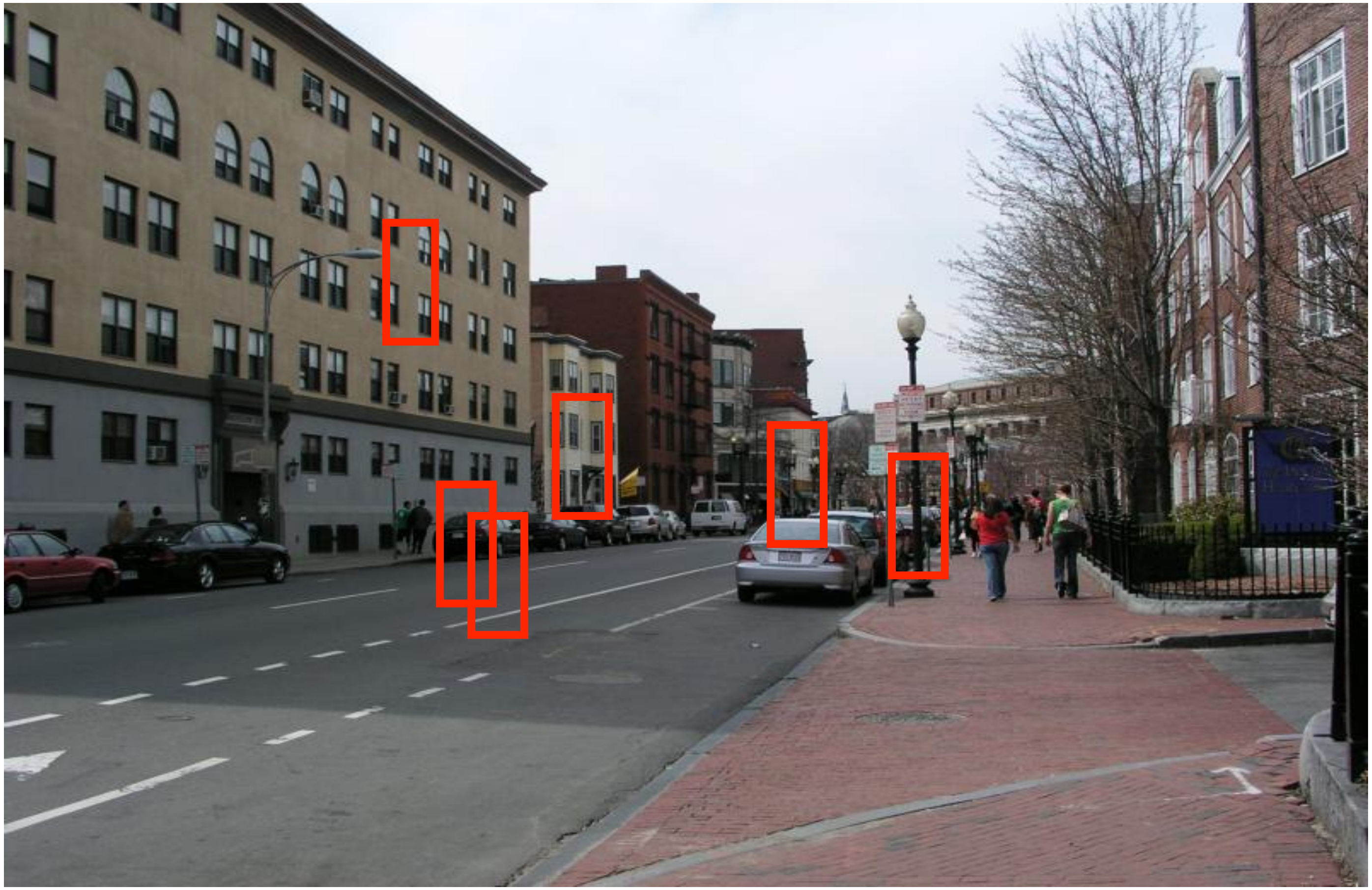
$$\min_{w} \frac{1}{2} \|w\|^2$$
$$\forall j \;\; y_j(w.x) > 1 \quad y_j \in \{-1 + 1\}$$

w



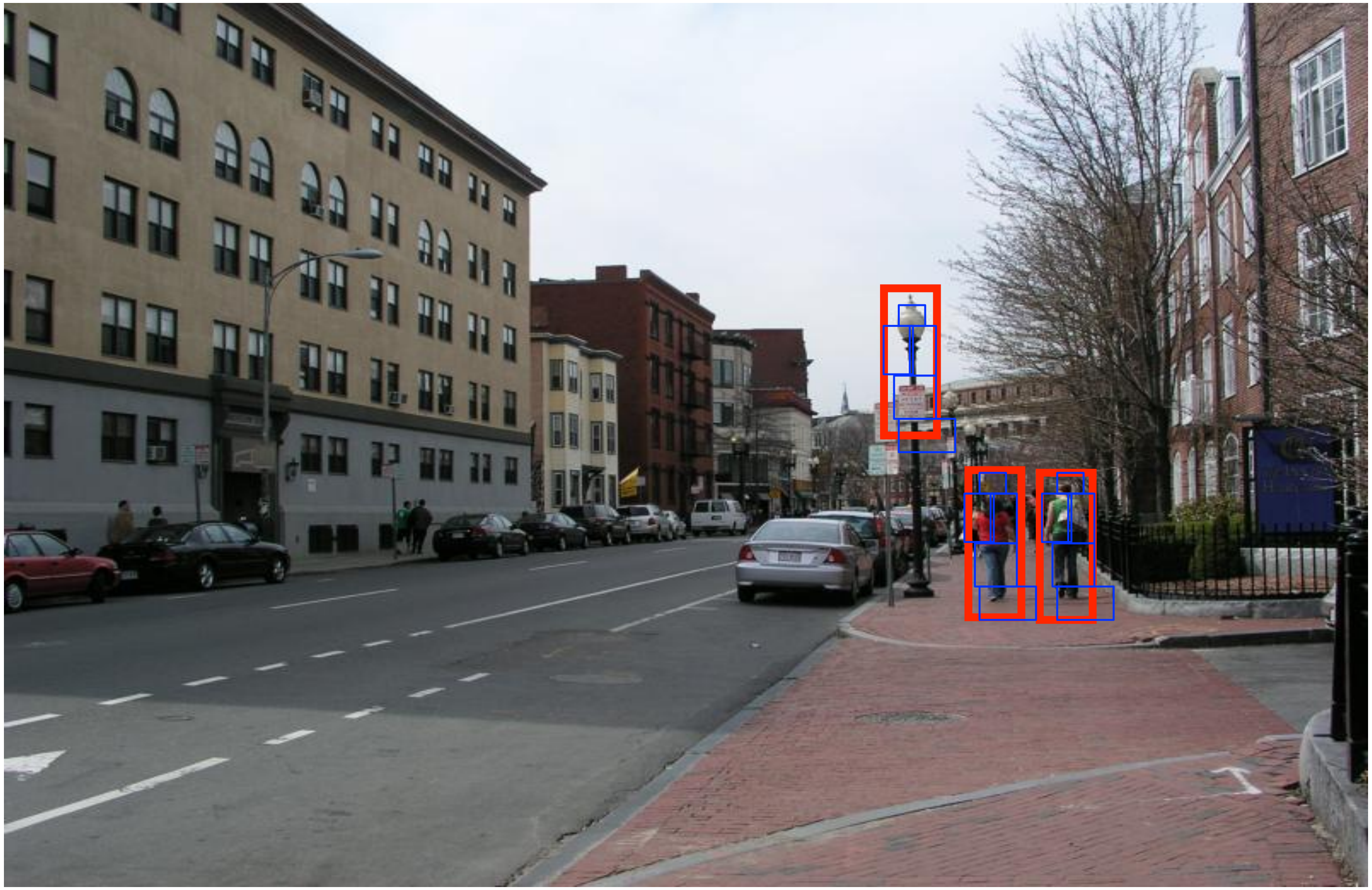
Negative Samples

- Infinite possibility for negative samples



Hard Negative Samples

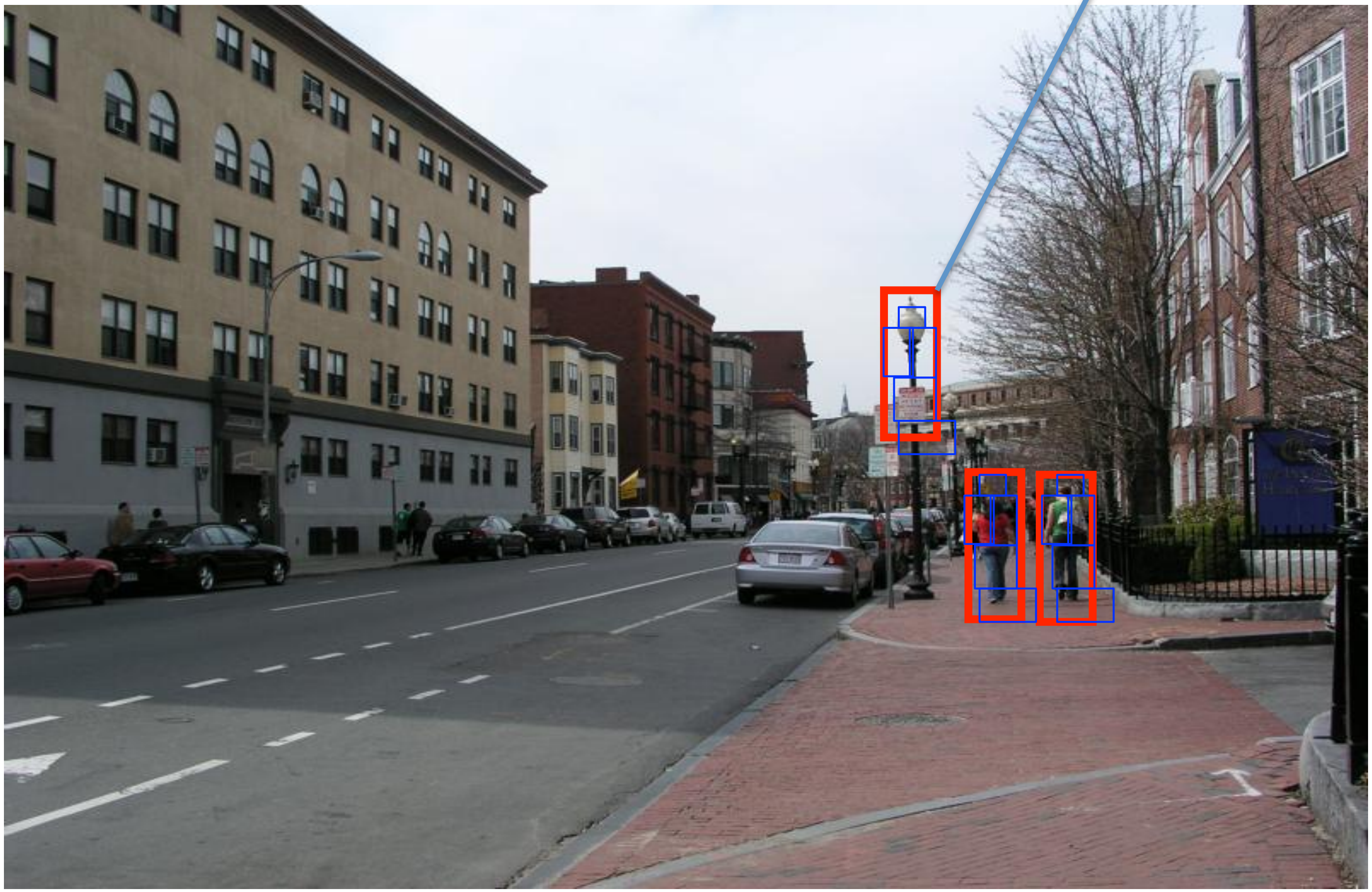
- Data Mining



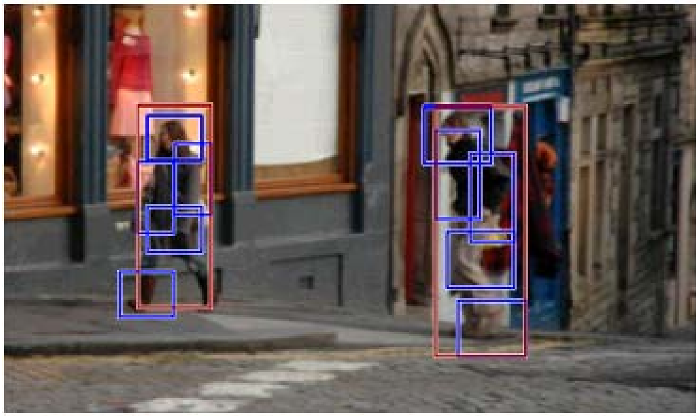
Hard Negative Samples

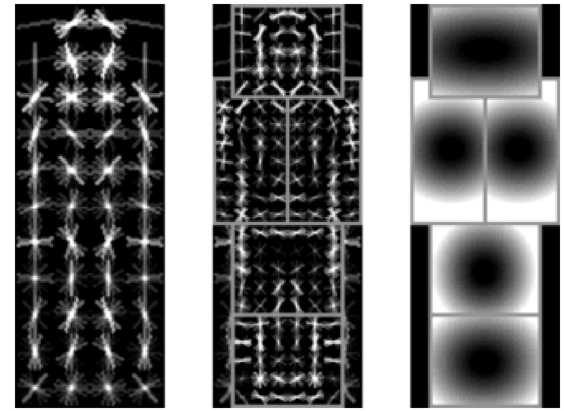
- Data Mining

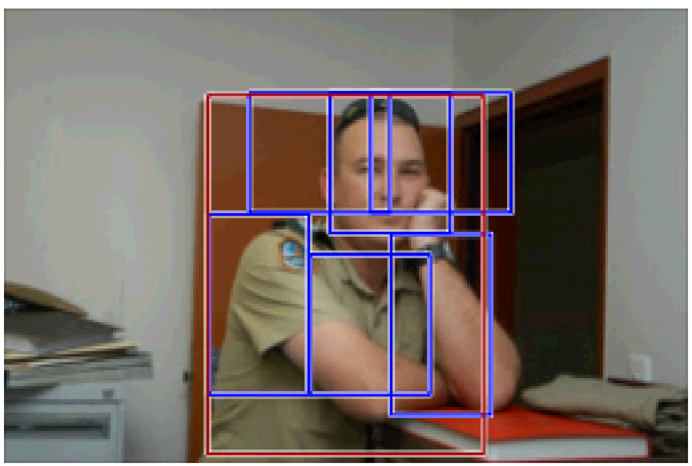
Add this as a negative sample for training

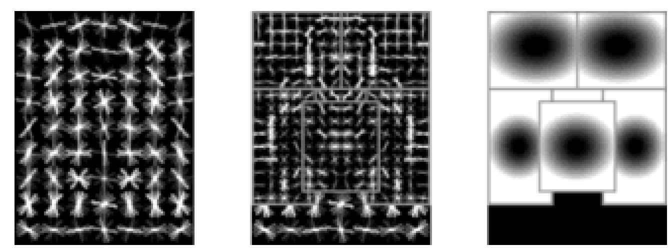


Mixture Model



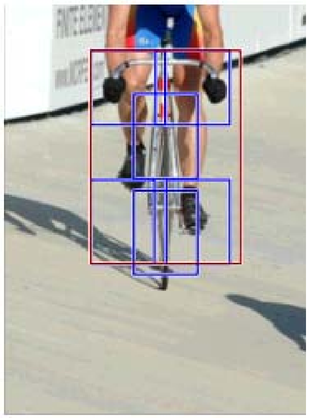


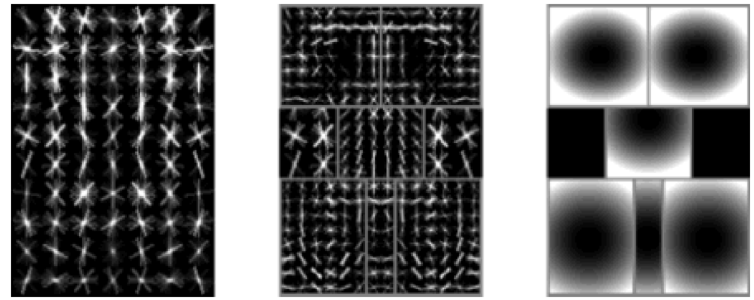


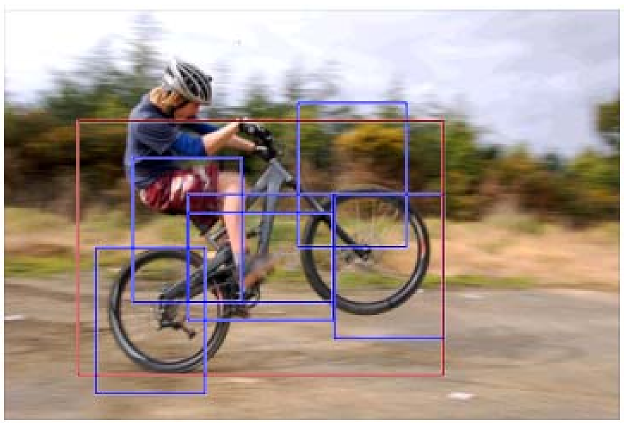


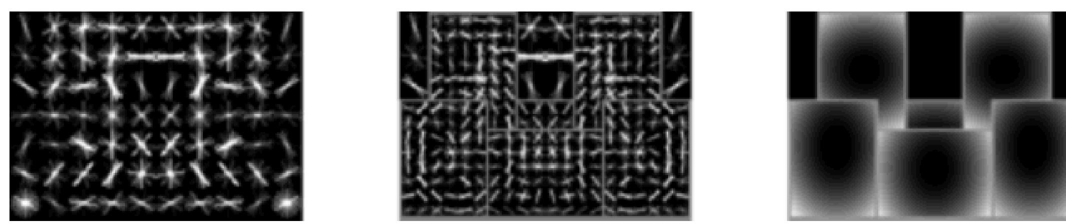
Bicycle

Bicycle

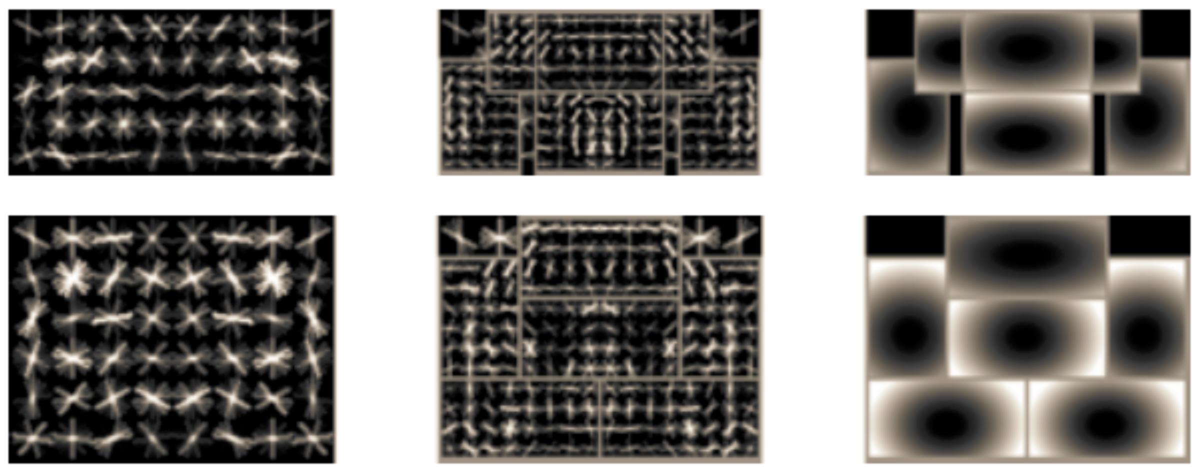








Car



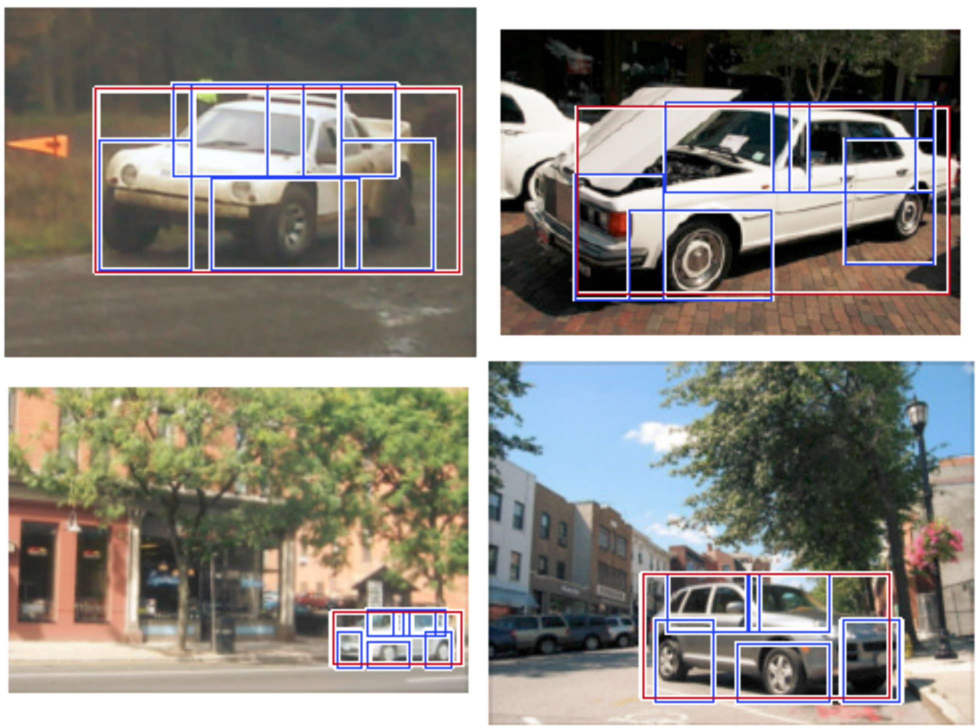
root filters
coarse resolution

part filters
finer resolution

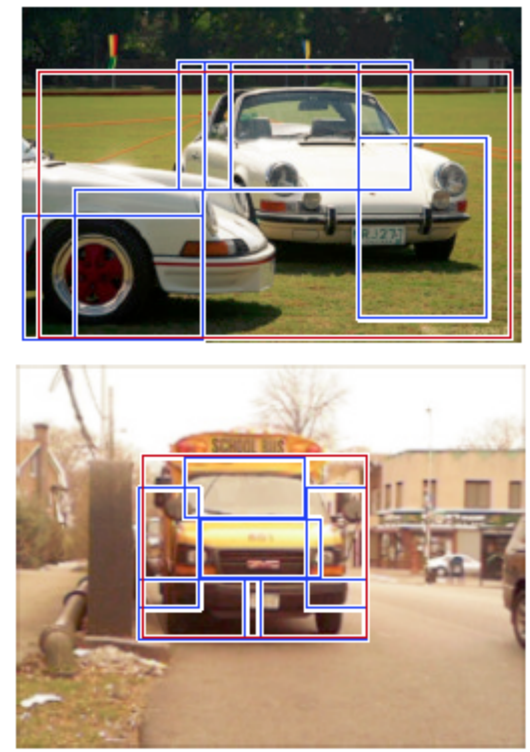
deformation
models

Car detections

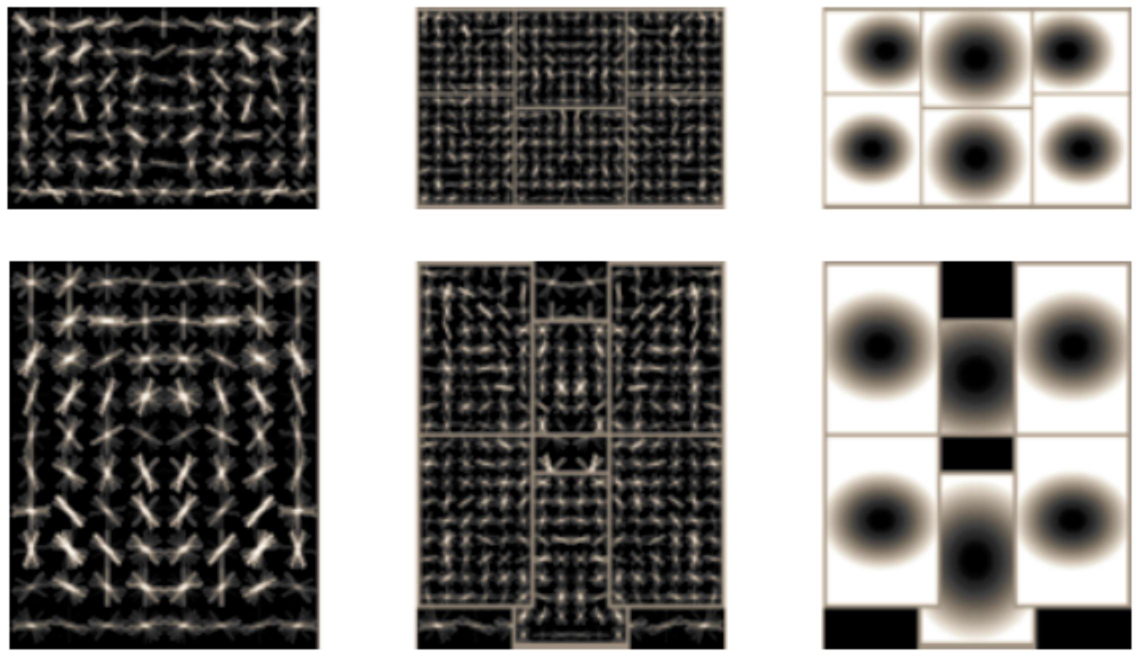
high scoring true positives



high scoring false positives



Cat



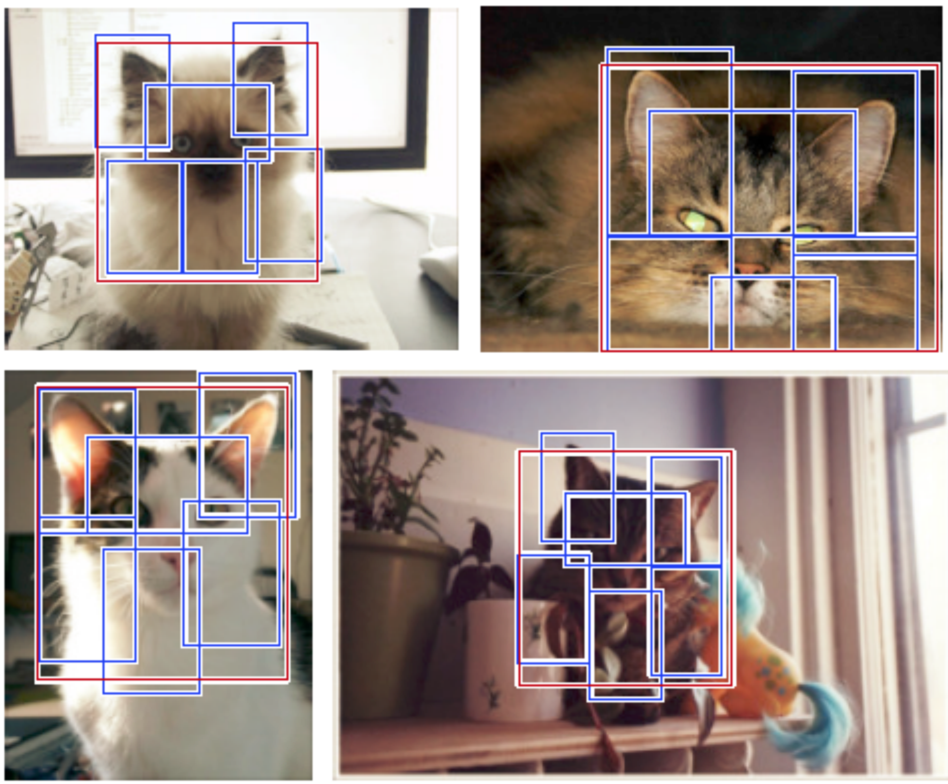
root filters
coarse resolution

part filters
finer resolution

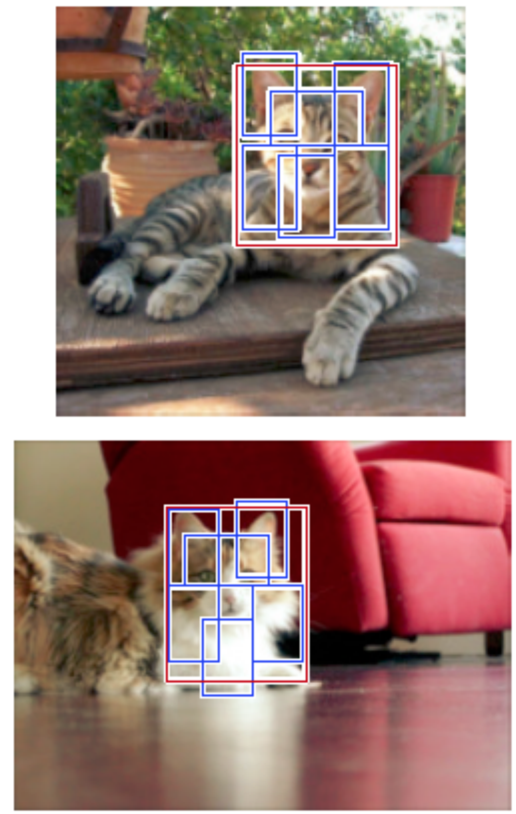
deformation
models

Cat detections

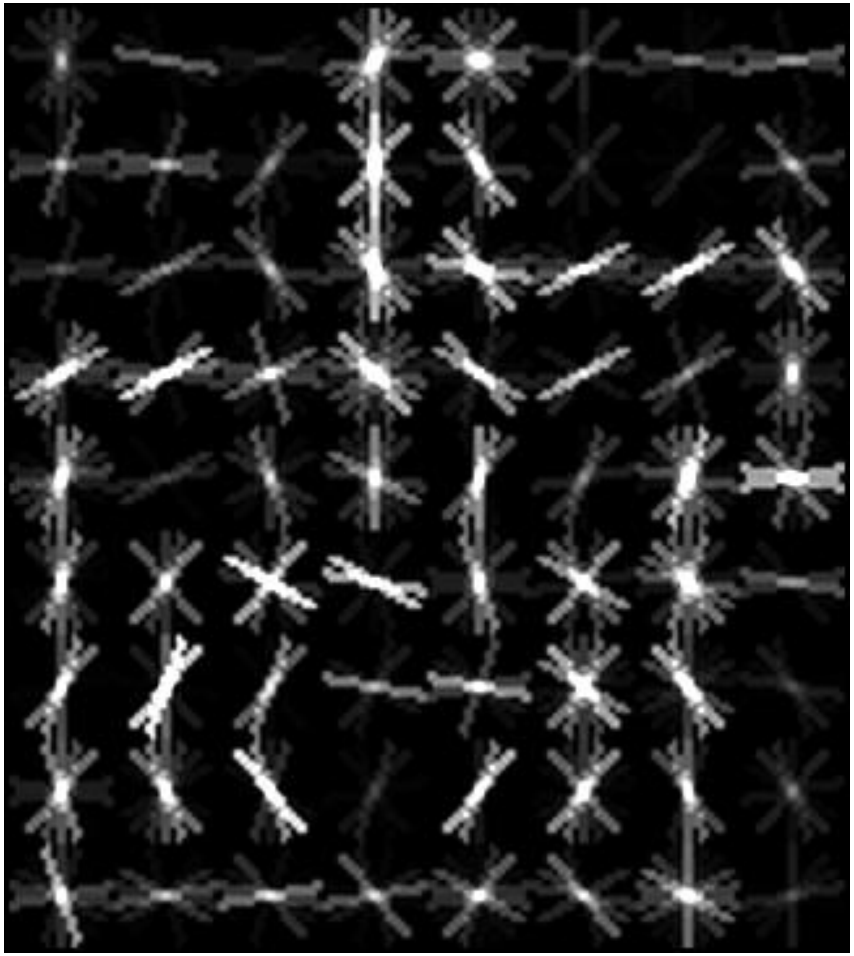
high scoring true positives



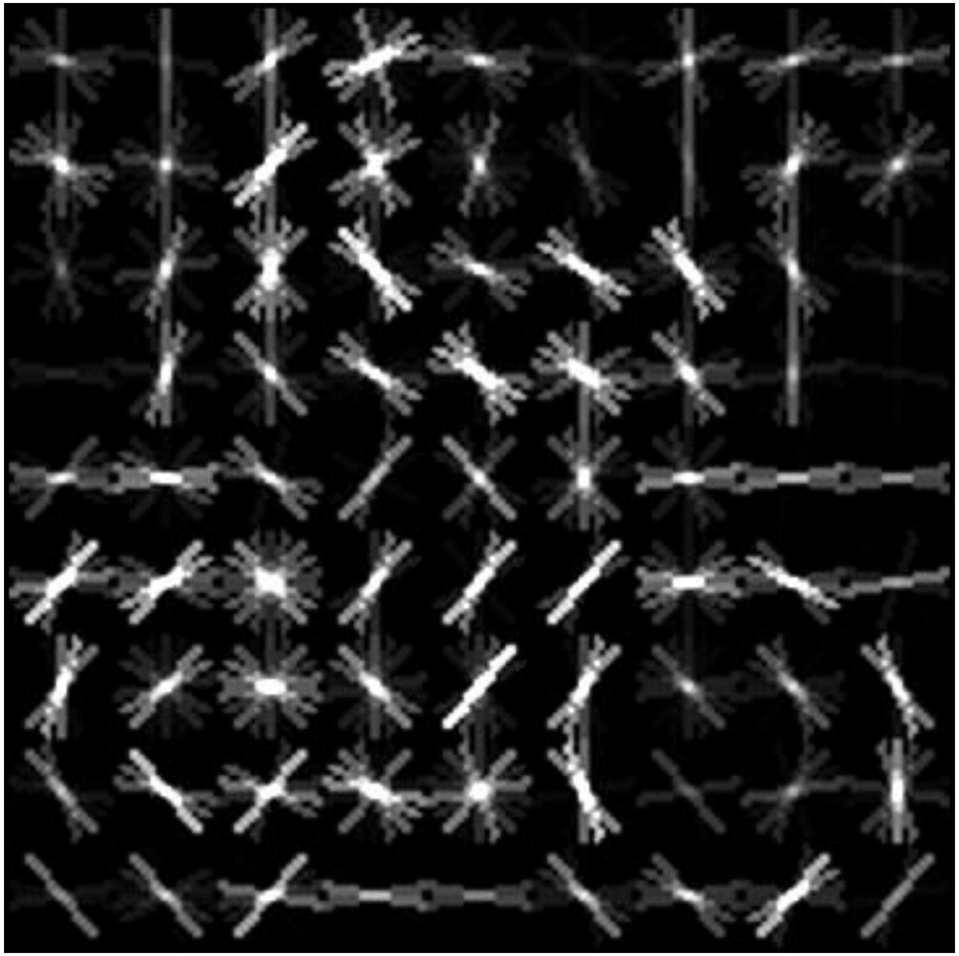
high scoring false positives
(not enough overlap)



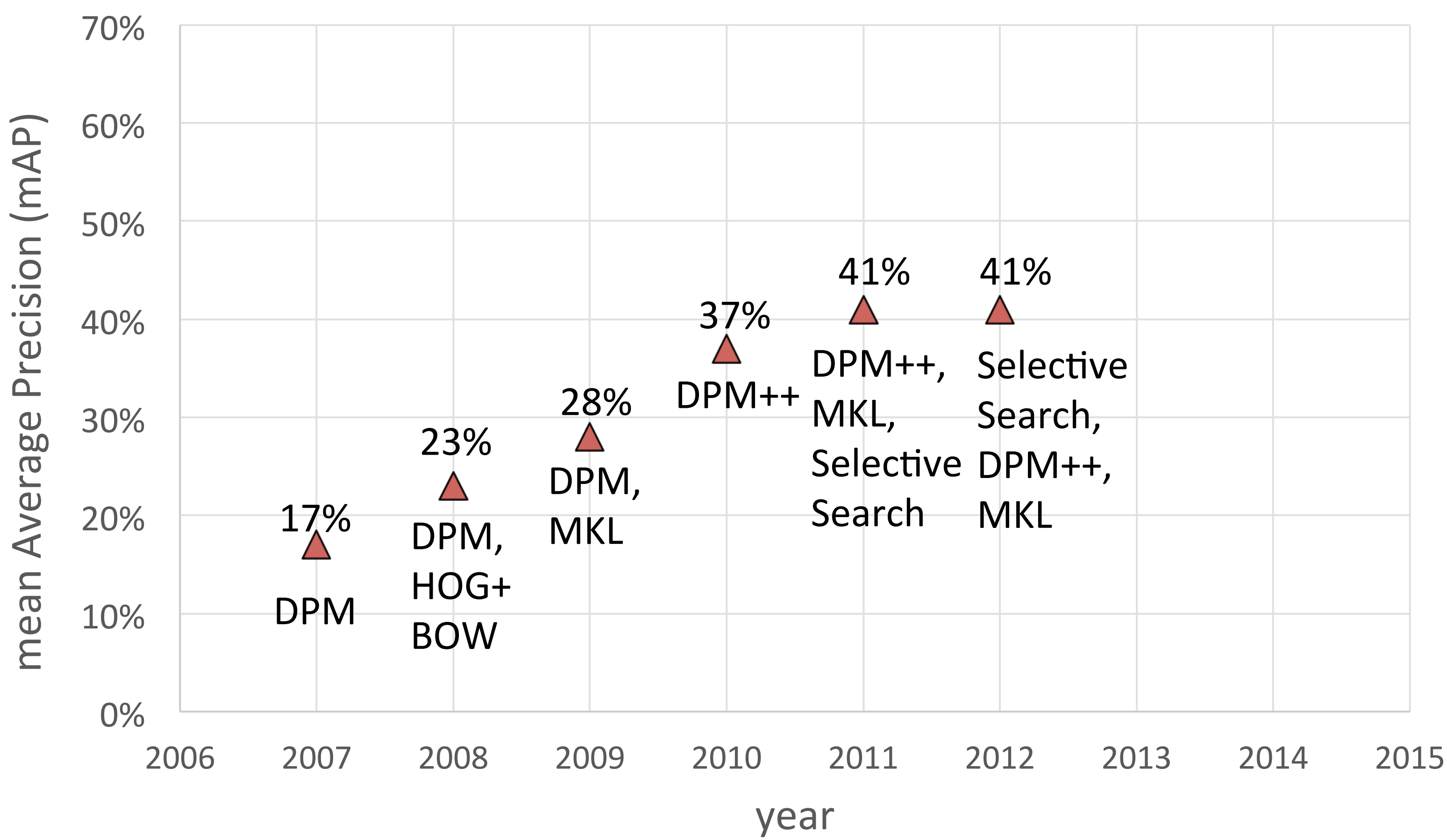
Person riding horse



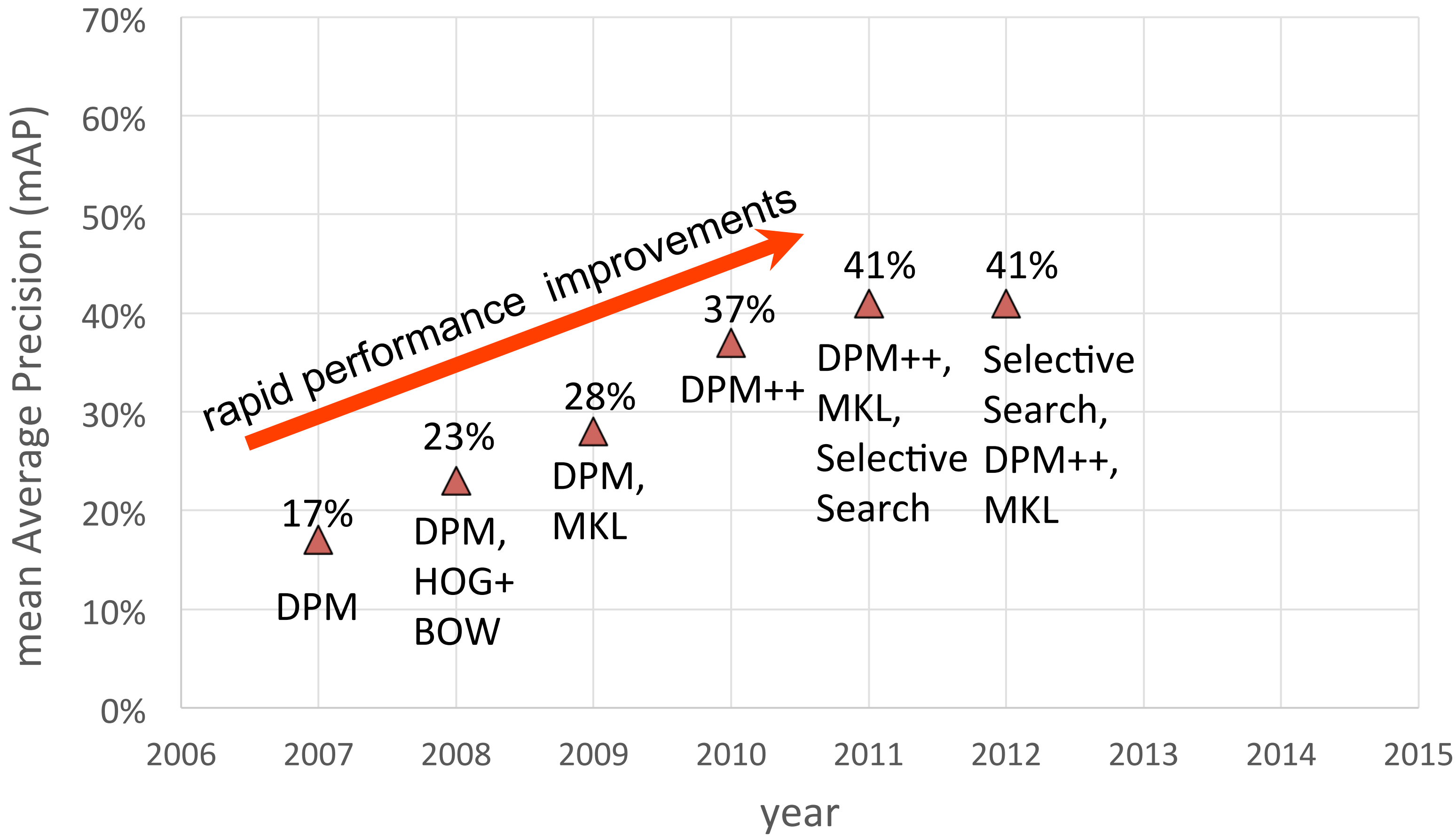
Person riding bicycle



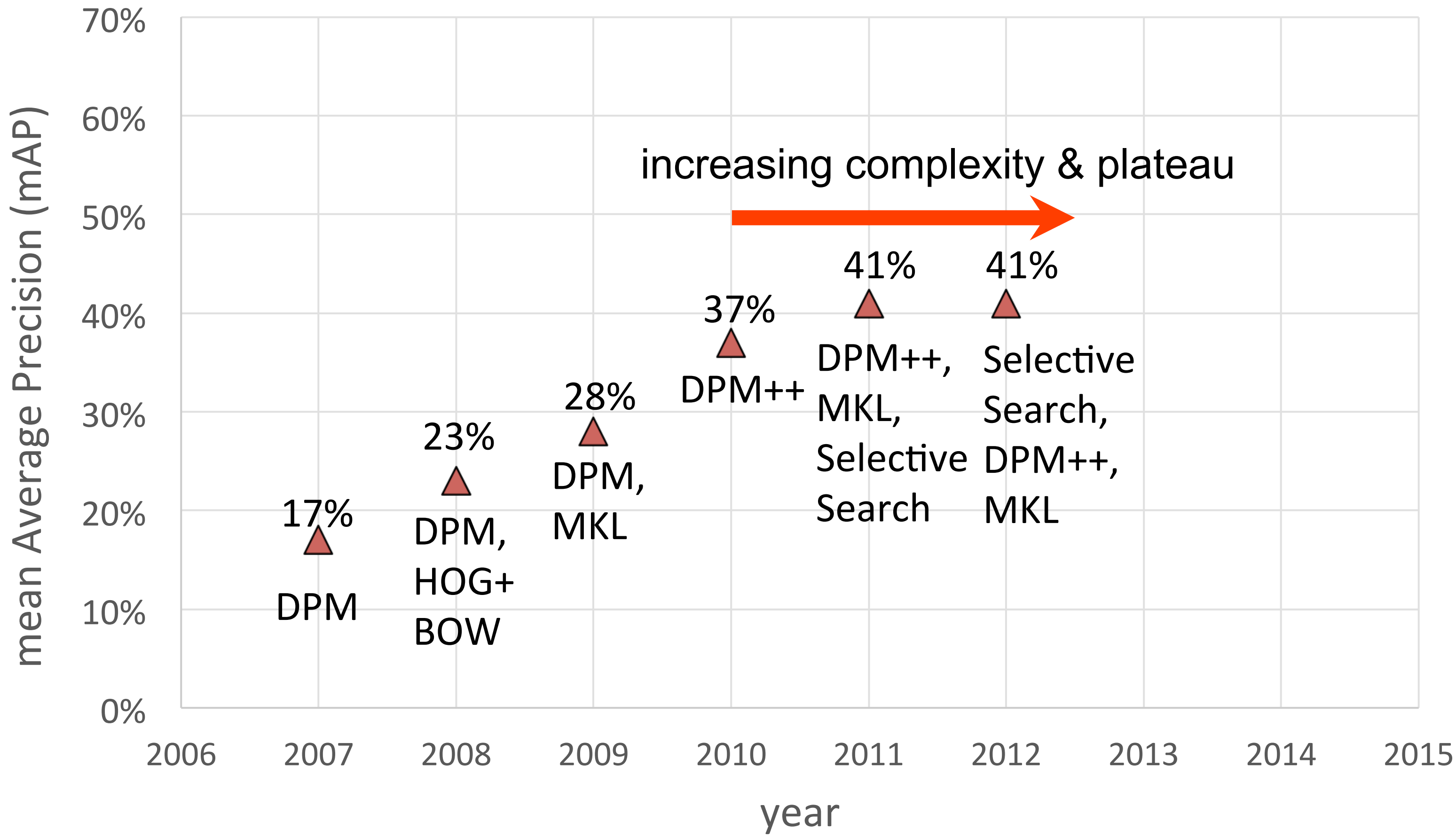
PASCAL VOC detection history



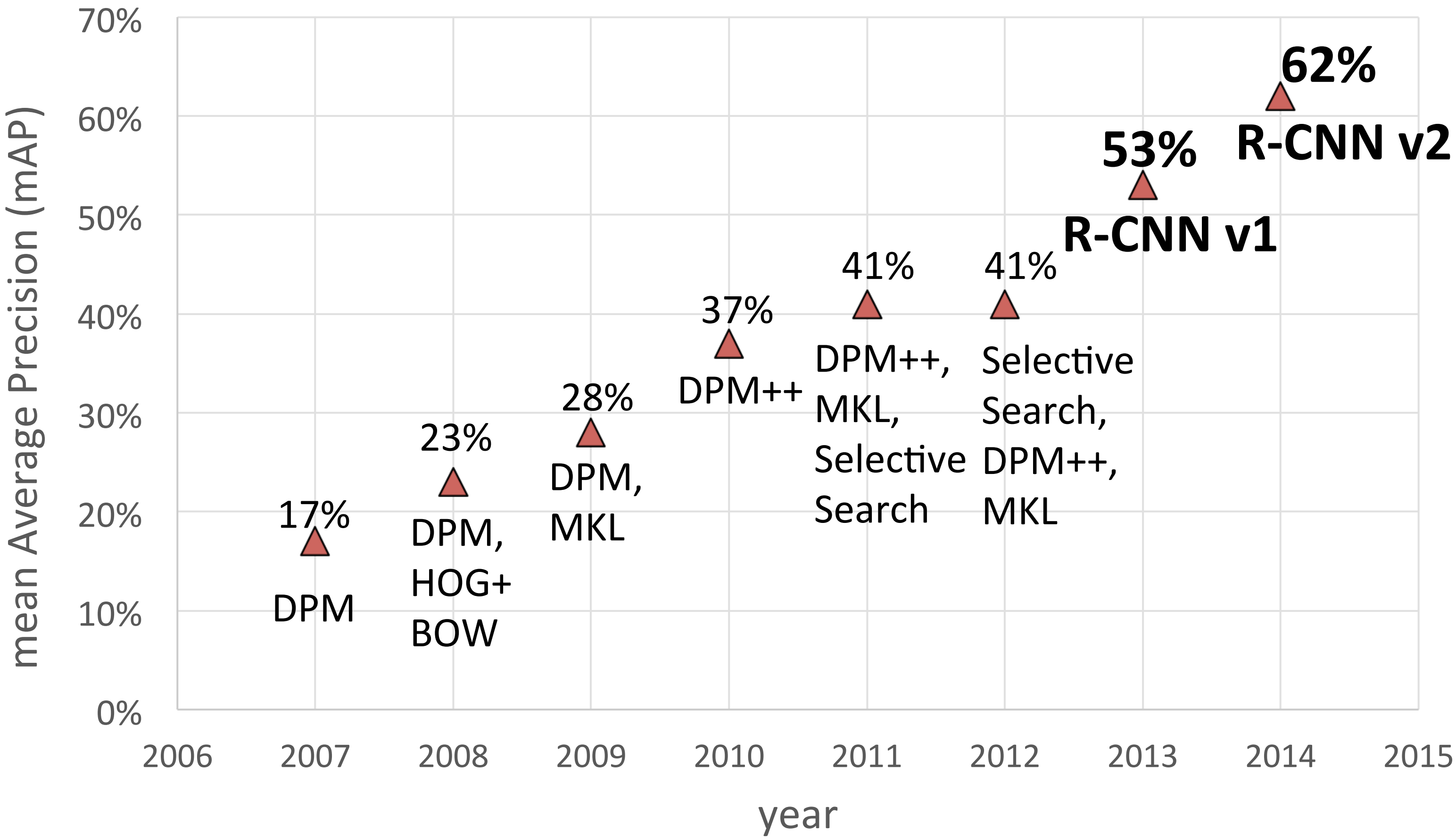
Part-based models & multiple features (MKL)



Kitchen-sink approaches

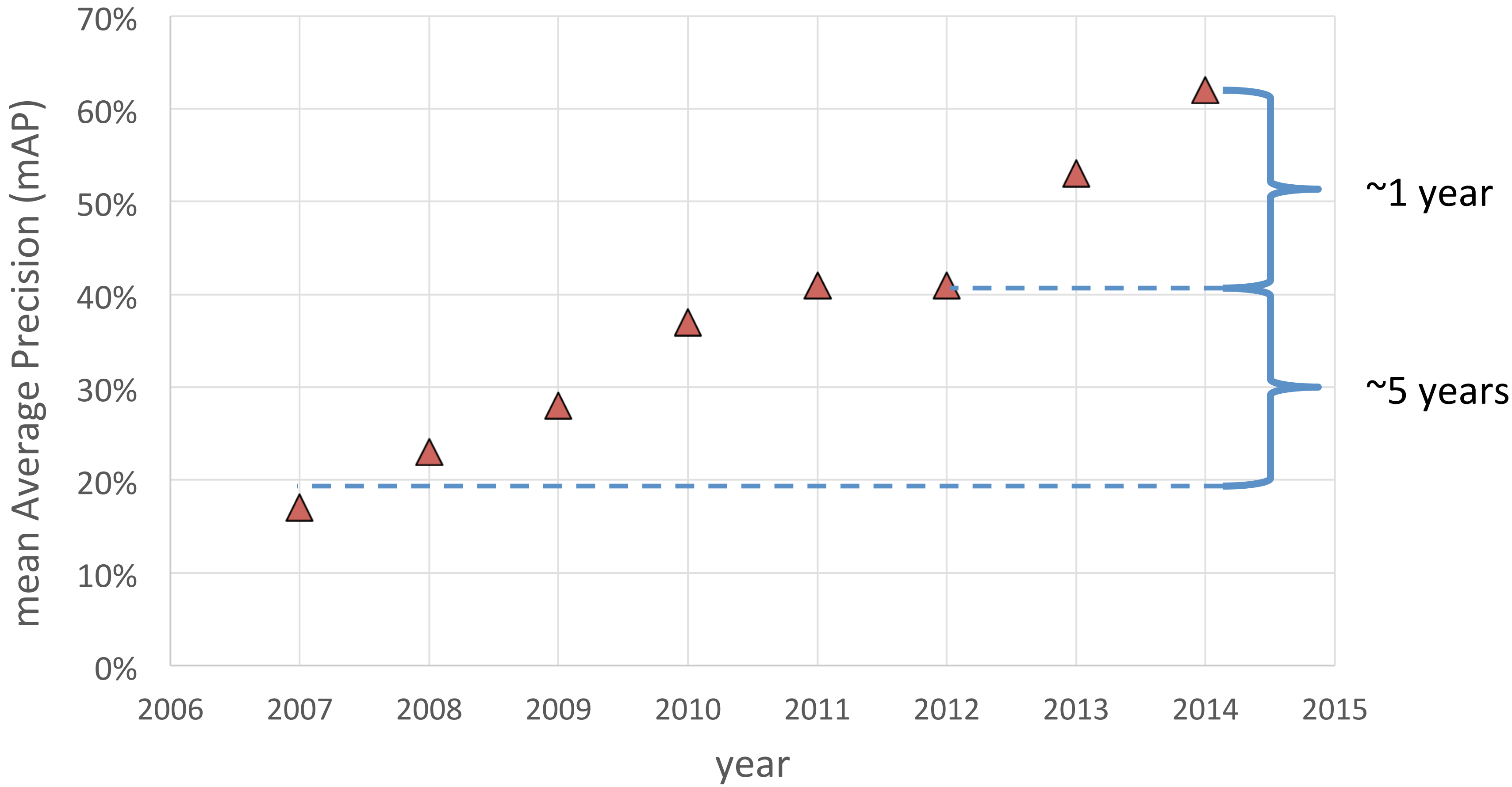


Region-based Convolutional Networks (R-CNNs)



[R-CNN. Girshick et al. CVPR 2014]

Region-based Convolutional Networks (R-CNNs)

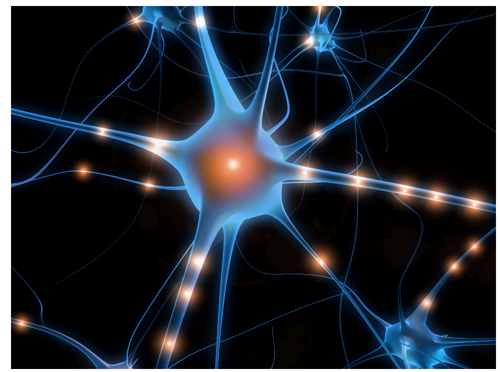


[R-CNN. Girshick et al. CVPR 2014]

Deep Neural Networks and Torch

Neural Networks A Brief History

- The 1940s: The Beginning of Neural Networks
  - Warren McCulloch and Walter Pitts (1943)
  - Threshold Logic



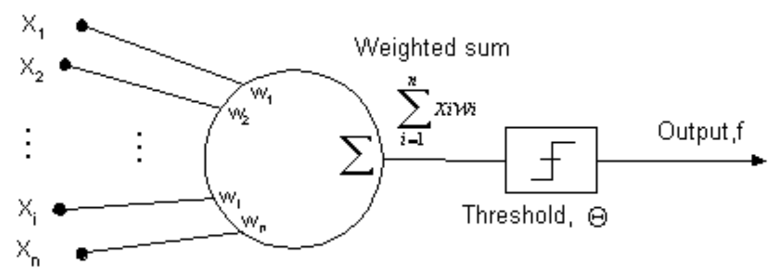
Bulletin of Mathematical Biology Vol. 52, No. 1/2, pp. 99–115, 1990.
Printed in Great Britain.

0092-8240/90$3.00+0.00
Pergamon Press plc
Society for Mathematical Biology

A LOGICAL CALCULUS OF THE IDEAS IMMANENT IN NERVOUS ACTIVITY*

■ WARREN S. MCCULLOCH AND WALTER PITTS
University of Illinois, College of Medicine,
Department of Psychiatry at the Illinois Neuropsychiatric Institute,
University of Chicago, Chicago, U.S.A.

Because of the "all-or-none" character of nervous activity, neural events and the relations among them can be treated by means of propositional logic. It is found that the behavior of every net can be described in these terms, with the addition of more complicated logical means for nets containing circles; and that for any logical expression satisfying certain conditions, one can find a net behaving in the fashion it describes. It is shown that many particular choices among possible neurophysiological assumptions are equivalent, in the sense that for every net behaving under one assumption, there exists another net which behaves under the other and gives the same results, although perhaps not in the same time. Various applications of the calculus are discussed.



$$f = 1 \text{ if } \sum_{i=1}^{n} x_i w_i \geq \Theta$$
$$= 0, \text{ otherwise}$$

Neural Networks A Brief History

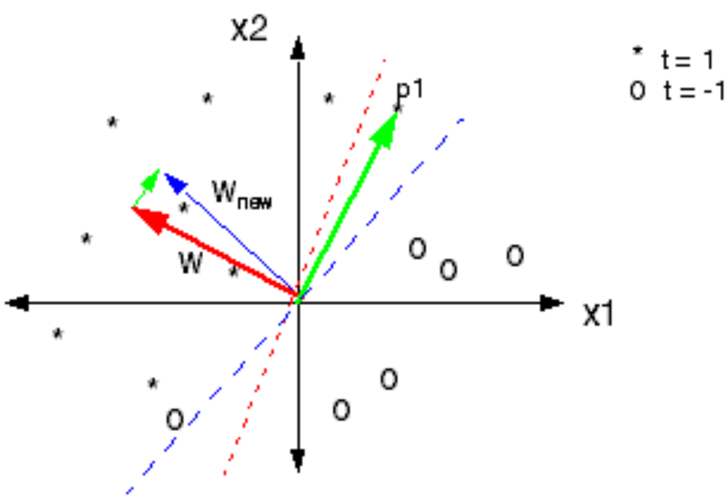
- The 1950s and 1960s: The First Golden Age of Neural Networks
  - Frank Rosenblatt (1958) created the perceptron

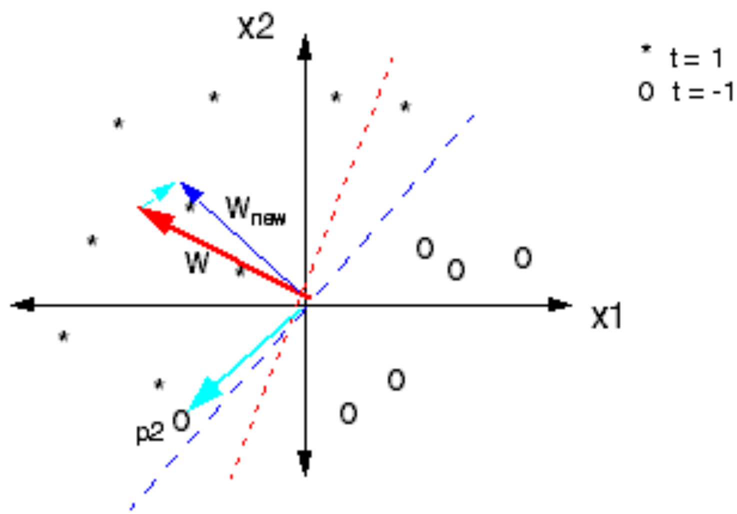
Psychological Review
Vol. 65, No. 6, 1958

THE PERCEPTRON: A PROBABILISTIC MODEL FOR INFORMATION STORAGE AND ORGANIZATION IN THE BRAIN[1]

F. ROSENBLATT

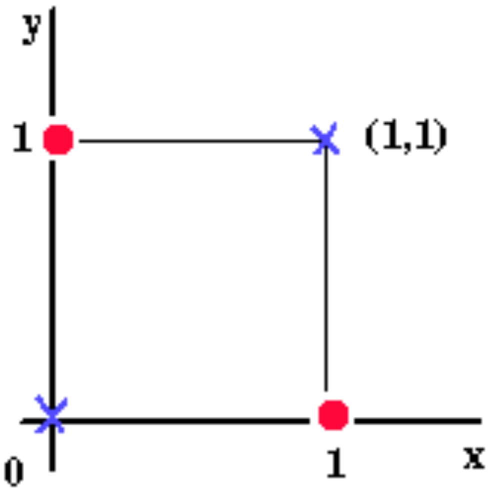
Cornell Aeronautical Laboratory



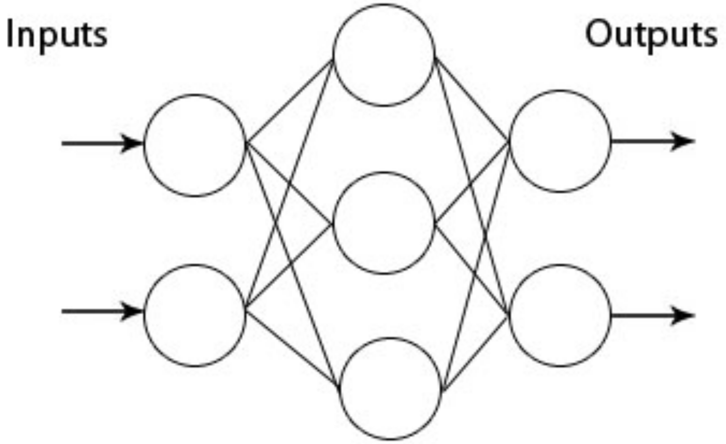


Neural Networks A Brief History

- The 1970s: The Quiet Years
  - Perceptron could not solve simple XOR problem
  - Overestimating the success of AI in research papers



Multi-Layer Perceptron : How to train?!!!

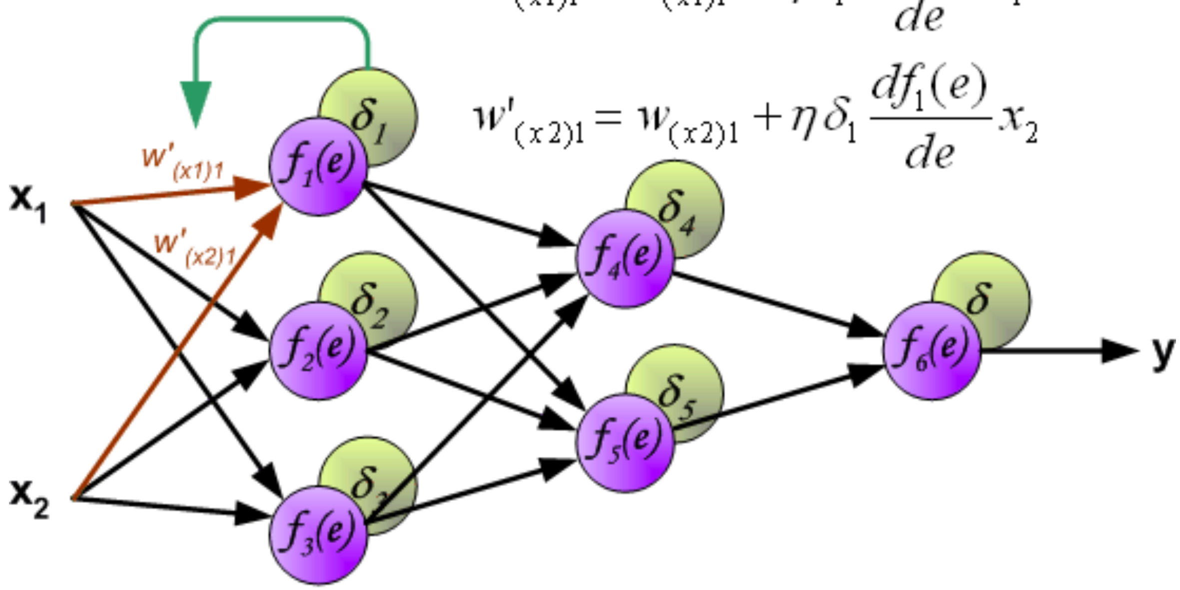


Neural Networks A Brief History

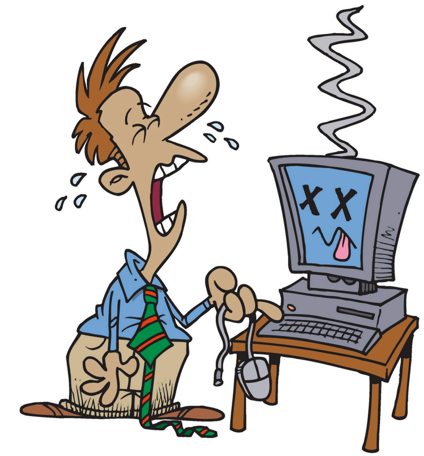
- After 1975 up to 1990: Renewed Enthusiasm
  - The Backpropagation algorithm was created by Paul Werbos (1975)

$$w'_{(x1)1} = w_{(x1)1} + \eta \delta_1 \frac{df_1(e)}{de} x_1$$

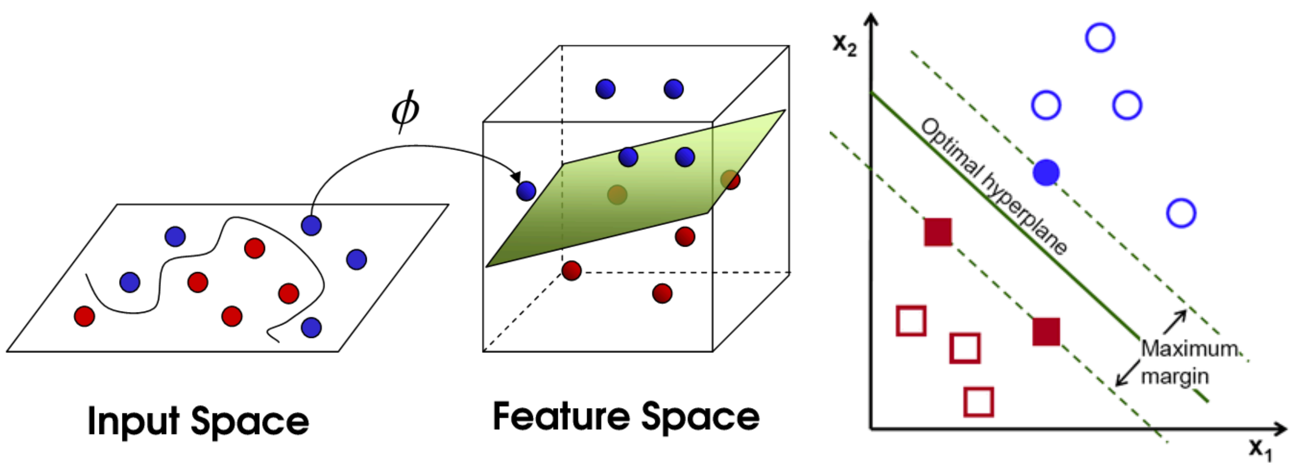
$$w'_{(x2)1} = w_{(x2)1} + \eta \delta_1 \frac{df_1(e)}{de} x_2$$



Neural Networks A Brief History

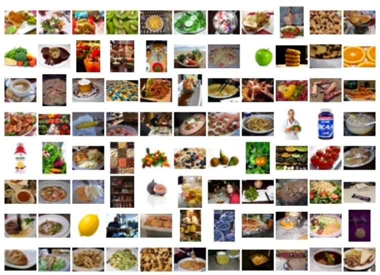


- 1990 -2012 : Long Quiet Years !!!
  - Learning large network was computationally expensive
  - Support Vector Machine took over
    - Convex Optimization
    - Nonlinear Models by Kernel Tricks



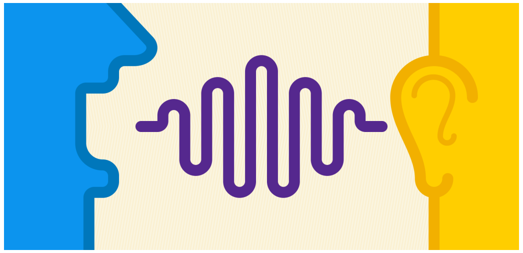
Feature Engineering

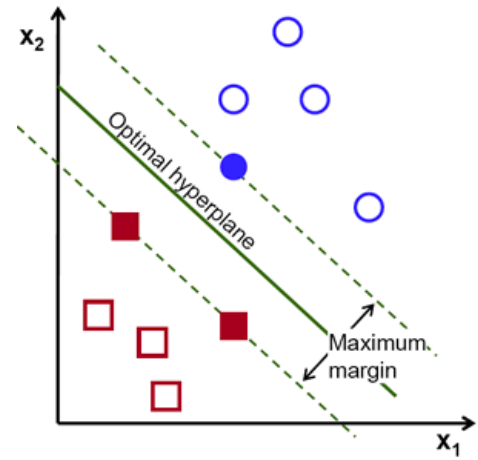
- Converting everything to a vector representation



X = [x1,x2,…,xD]

Feature Engineering





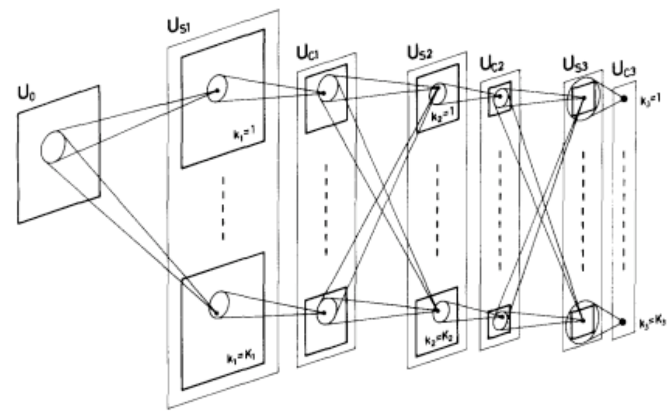
Machine Learning

Feature Learning

- Convolutional Neural Networks

Biol. Cybernetics 36, 193–202 (1980)

Biological Cybernetics



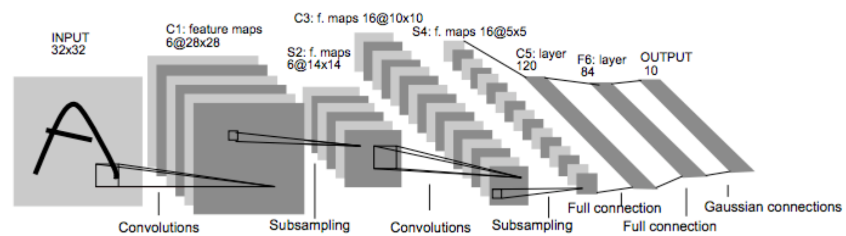
Neocognitron: A Self-organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position

Kunihiko Fukushima
NHK Broadcasting Science Research Laboratories, Kinuta, Setagaya, Tokyo, Japan

PROC. OF THE IEEE, NOVEMBER 1998 1

Gradient-Based Learning Applied to Document Recognition

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner



https://www.youtube.com/watch?v=Qil4kmvm2Sw

Feature Learning

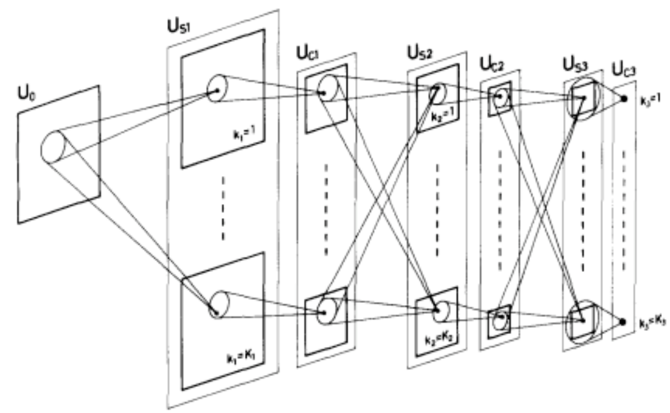
- Convolutional Neural Networks

Biol. Cybernetics 36, 193–202 (1980)

Biological Cybernetics

Neocognitron: A Self-organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position

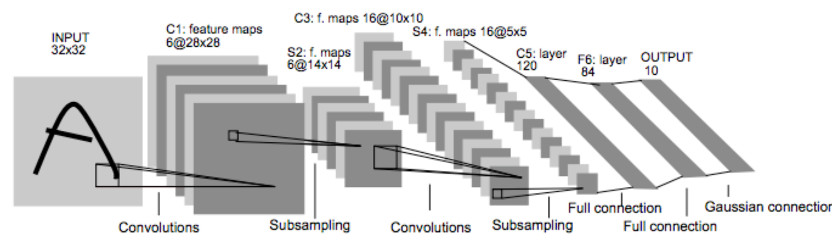
Kunihiko Fukushima
NHK Broadcasting Science Research Laboratories, Kinuta, Setagaya, Tokyo, Japan



PROC. OF THE IEEE, NOVEMBER 1998 1

Gradient-Based Learning Applied to Document Recognition

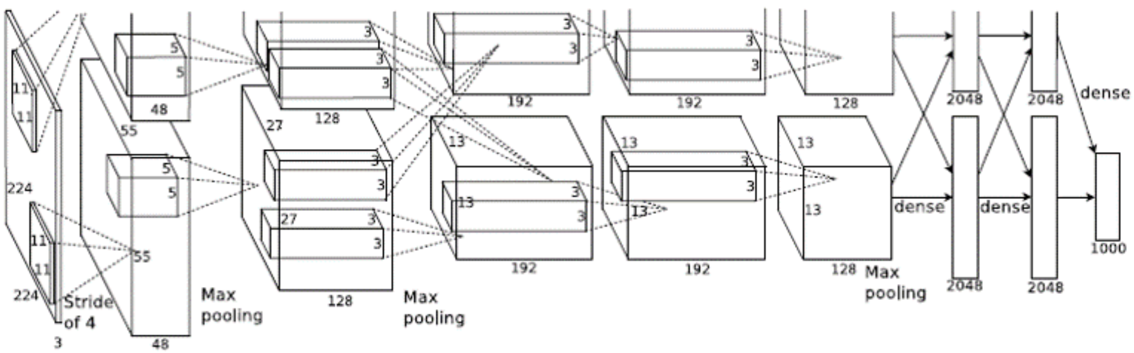
Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner

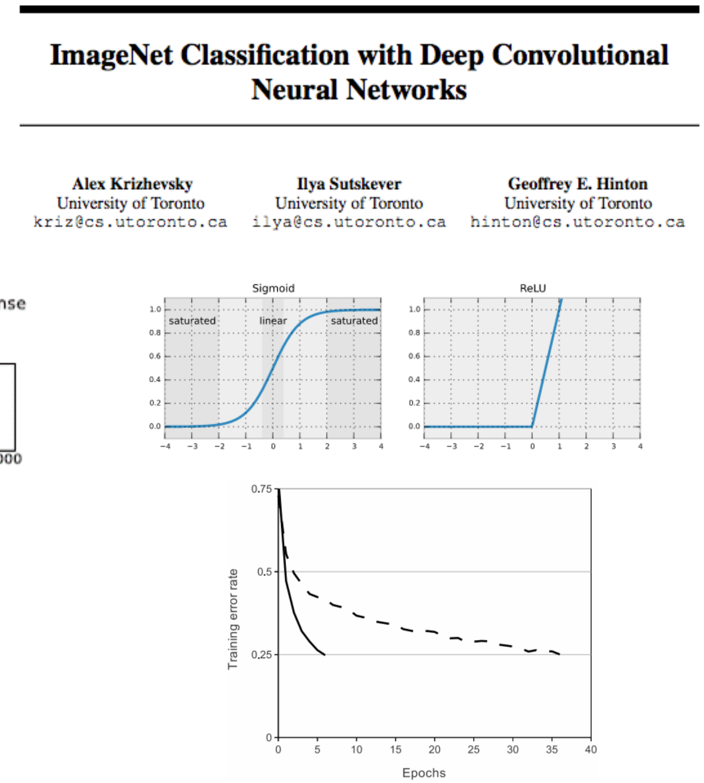


https://www.youtube.com/watch?v=Qil4kmvm2Sw

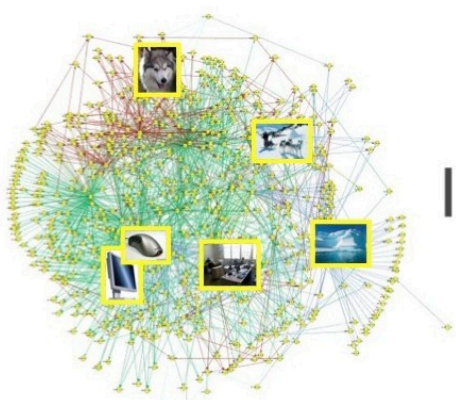
GPU and BigData

- AlexNet (2012)





- ImageNet



IMAGENET