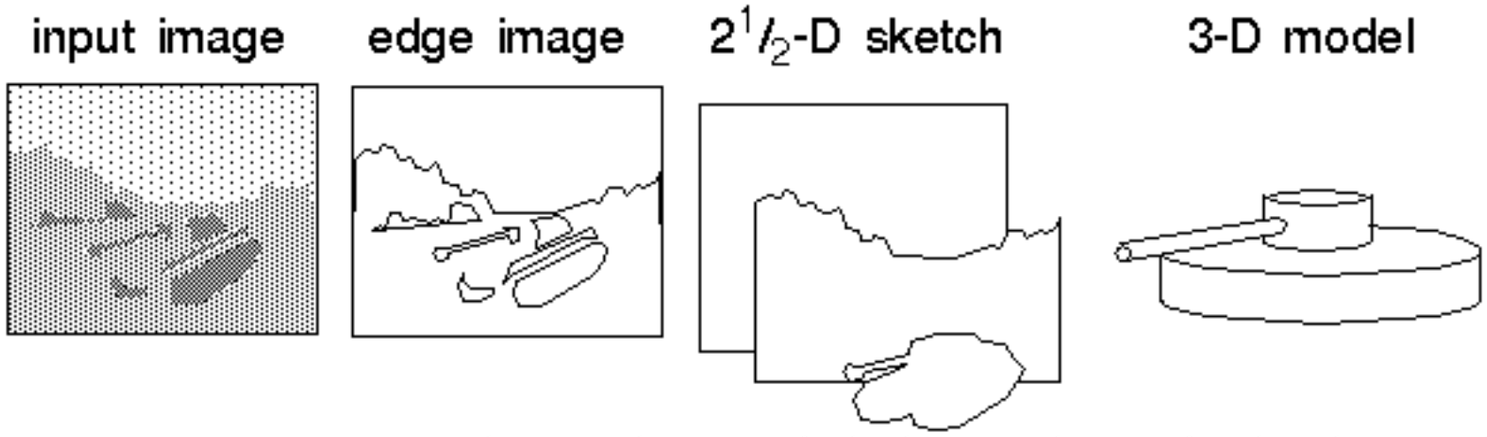
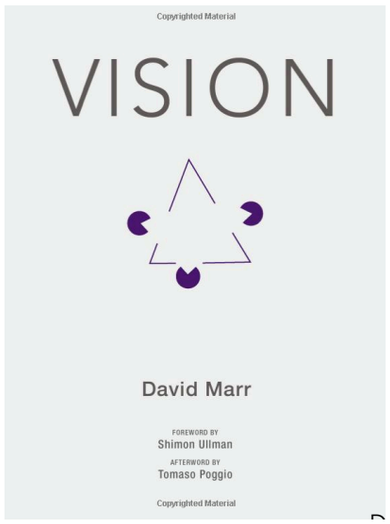
Outline

- Motivation
- How to define problem?
- Case example: finding people
- Statistical classification

Zoom-out to the big picture



Low-level Mid-level High-level



(We'll talk more about mid-level grouping toward end of semester)

Formalizing recognition

Output

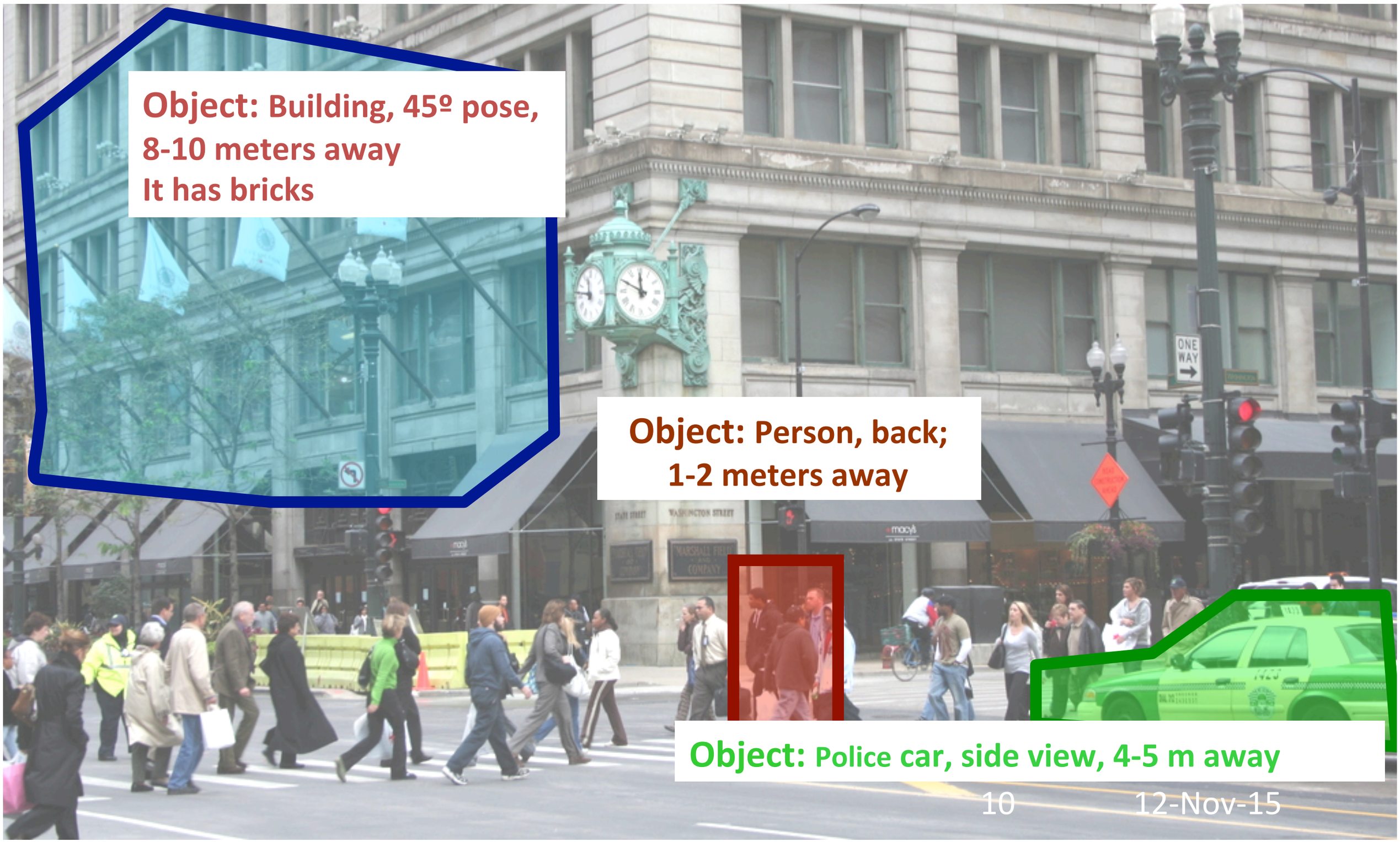
What are outputs of interest?
How should space of outputs be represented?

yes-no answers: does this contain a car?

Image classification is this a beach?

Detection Does this contain a car? Where?

Attributes Object poses + materials



Activities What are these people doing?

Instance recognition

Recognition meet geometry

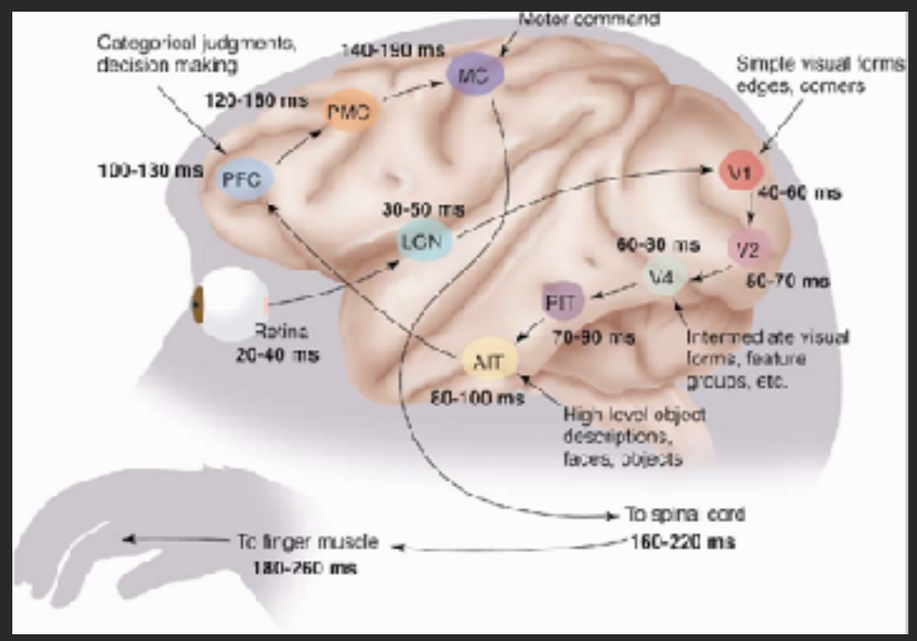
Formalizing recognition

Human vision experiment: what can people describe when looking at images?

Tricky question to answer….

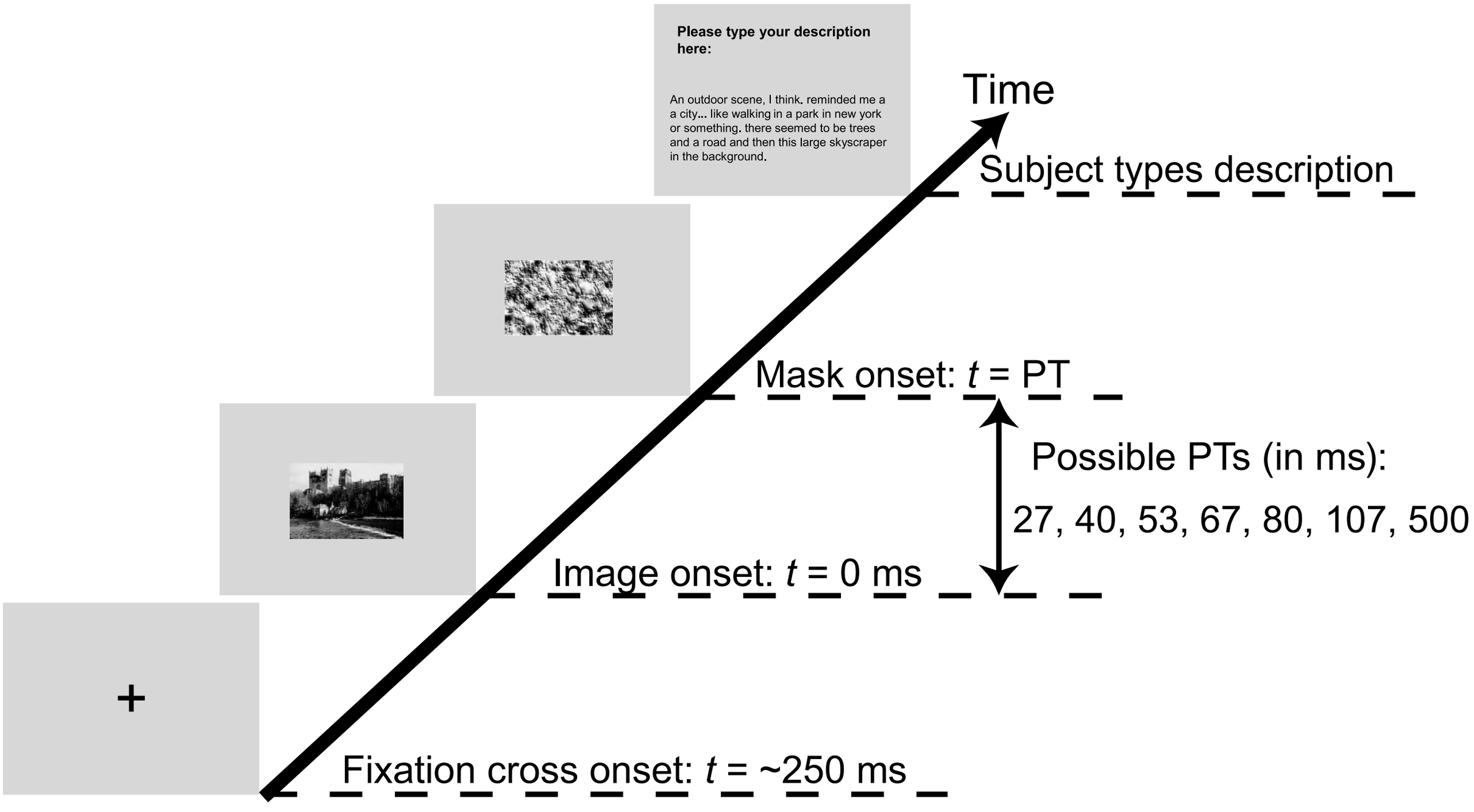
Rapid scene catgorization

People can distinguish high-level concepts (animal/transport) in under 150ms (Thorpe)



Appears to suggest feed-forward computations suffice (or at least dominate)

What do we perceive in a glance of a real-world scene?



PT = 107 ms

This is outdoors. A black, furry dog is running/walking towards the right of the picture. His tail is in the air and his mouth is open. Either he had a ball in his mouth or he was chasing after a ball. (Subject EC)

PT = 500 ms

I saw a black dog carrying a gray frisbee in the center of the photograph. The dog was walking near the ocean, with waves lapping up on the shore. It seemed to be a gray day out. (Subject JB)

Inside a house, like a living room, with chairs and sofas and tables, no ppl. (Subject HS)

A room full of musical instruments. A piano in the foreground, a harp behind that, a guitar hanging on the wall (to the right). It looked like there was also a window behind the harp, and perhaps a bookcase on the left. (Subject RW)

PT = 107 ms

This is outdoors. A black, furry dog is running/walking towards the right of the picture. His tail is in the air and his mouth is open. Either he had a ball in his mouth or he was chasing after a ball. (Subject EC)

PT = 500 ms

I saw a black dog carrying a gray frisbee in the center of the photograph. The dog was walking near the ocean, with waves lapping up on the shore. It seemed to be a gray day out. (Subject JB)

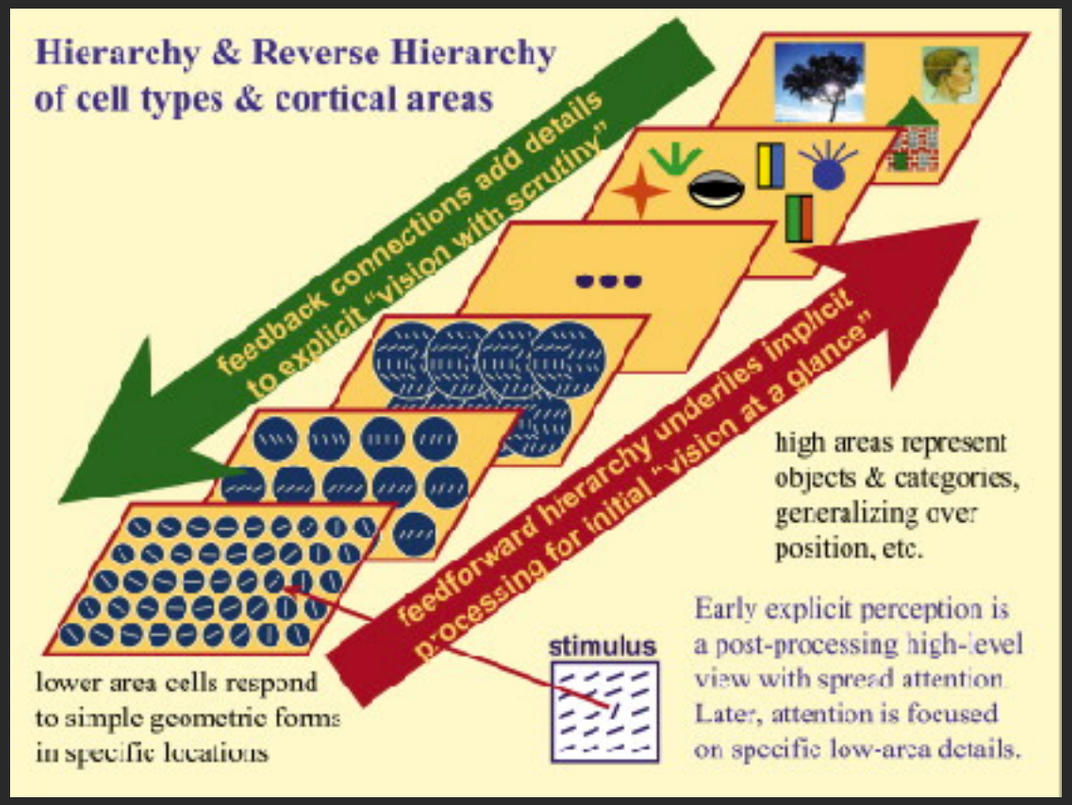
Inside a house, like a living room, with chairs and sofas and tables, no ppl. (Subject HS)

A room full of musical instruments. A piano in the foreground, a harp behind that, a guitar hanging on the wall (to the right). It looked like there was also a window behind the harp, and perhaps a bookcase on the left. (Subject RW)

Should language be the right output?

“The more you look, the more you see”

Hochstein & Ahissar 02



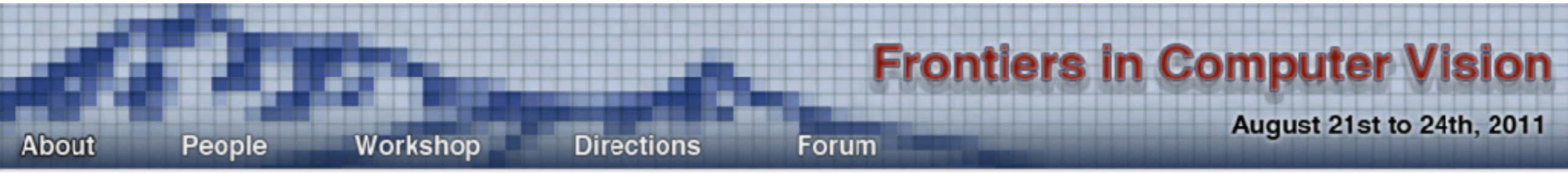
Vision at a glance (feedforward)

Rapid scene categorization

Vision with scrutiny (+feedback)

Fine-grained recognition Spatial localization for manipulation

Hilbert problems of vision



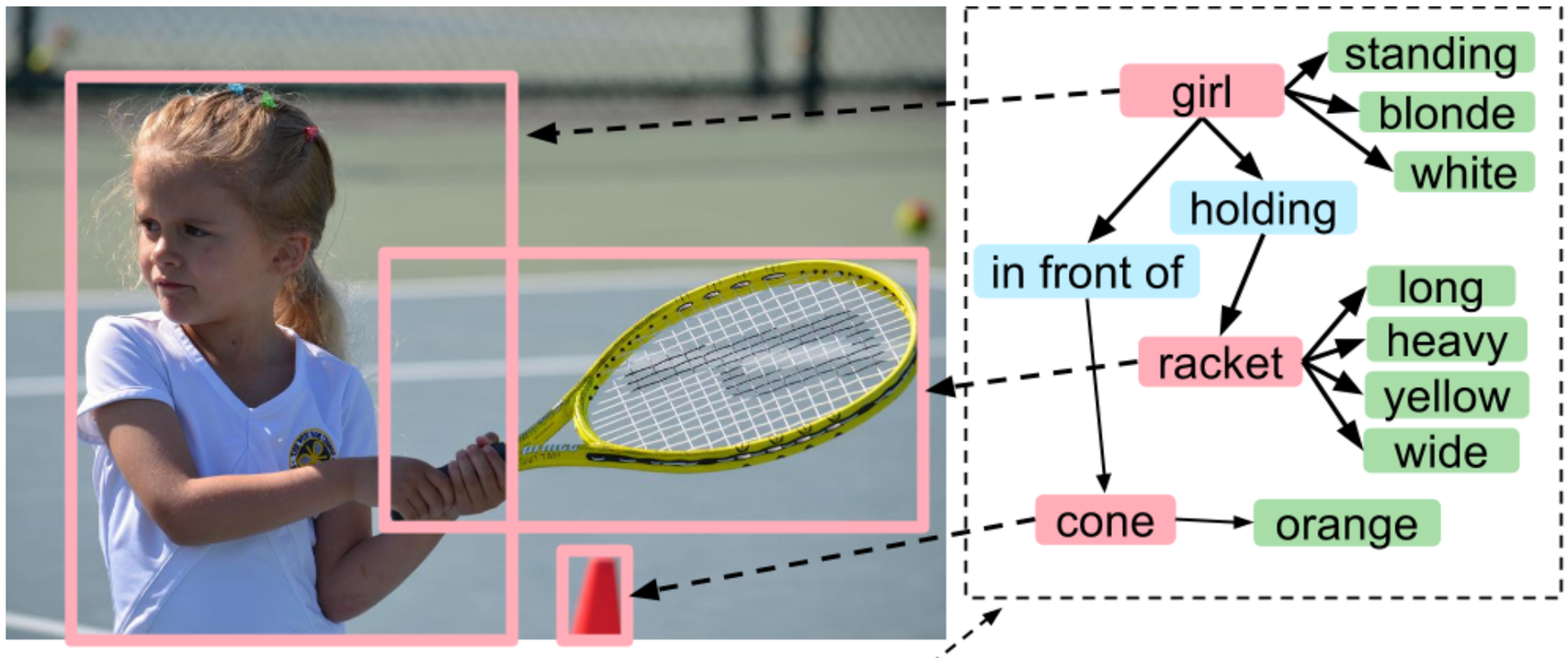
Vigorous discussion

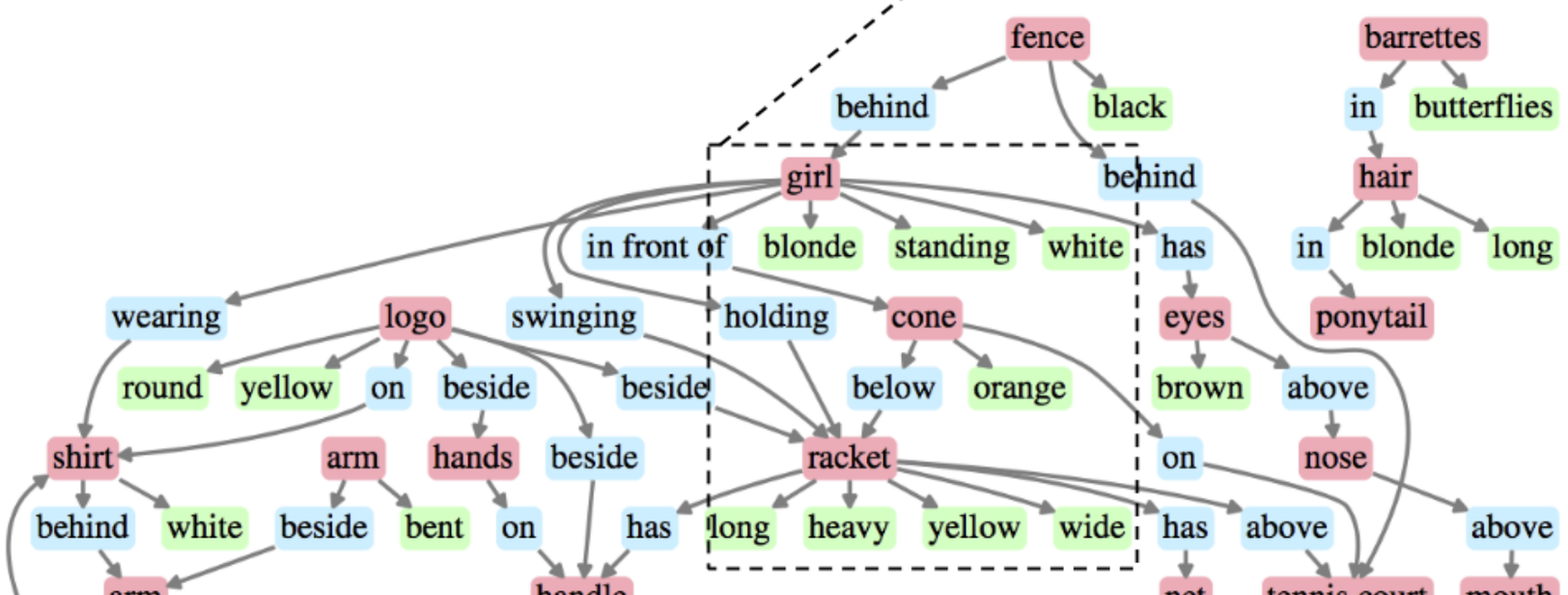
Hierarchical annotation

Image Retrieval using Scene Graphs

Justin Johnson[1], Ranjay Krishna[1], Michael Stark[2], Li-Jia Li[3,4], David A. Shamma[3],
Michael S. Bernstein[1], Li Fei-Fei[1]

[1]Stanford University, [2]Max Planck Institute for Informatics, [3]Yahoo Labs, [4]Snapchat

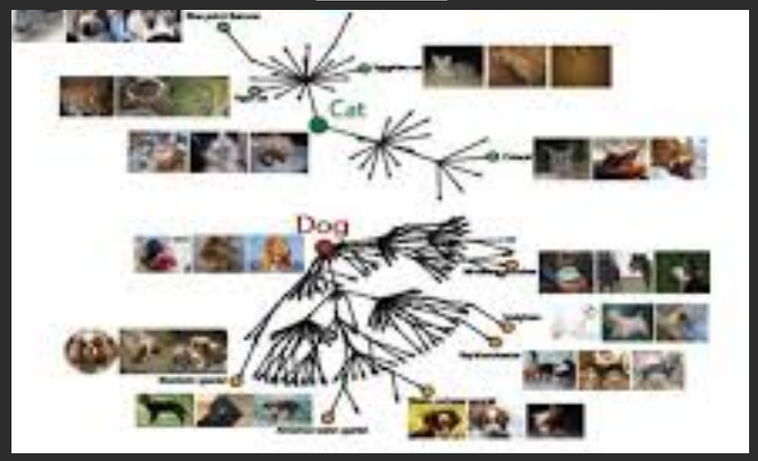




Other hiearchies for organizing recognition outputs

Partonomies
(people are made out of arms and legs)

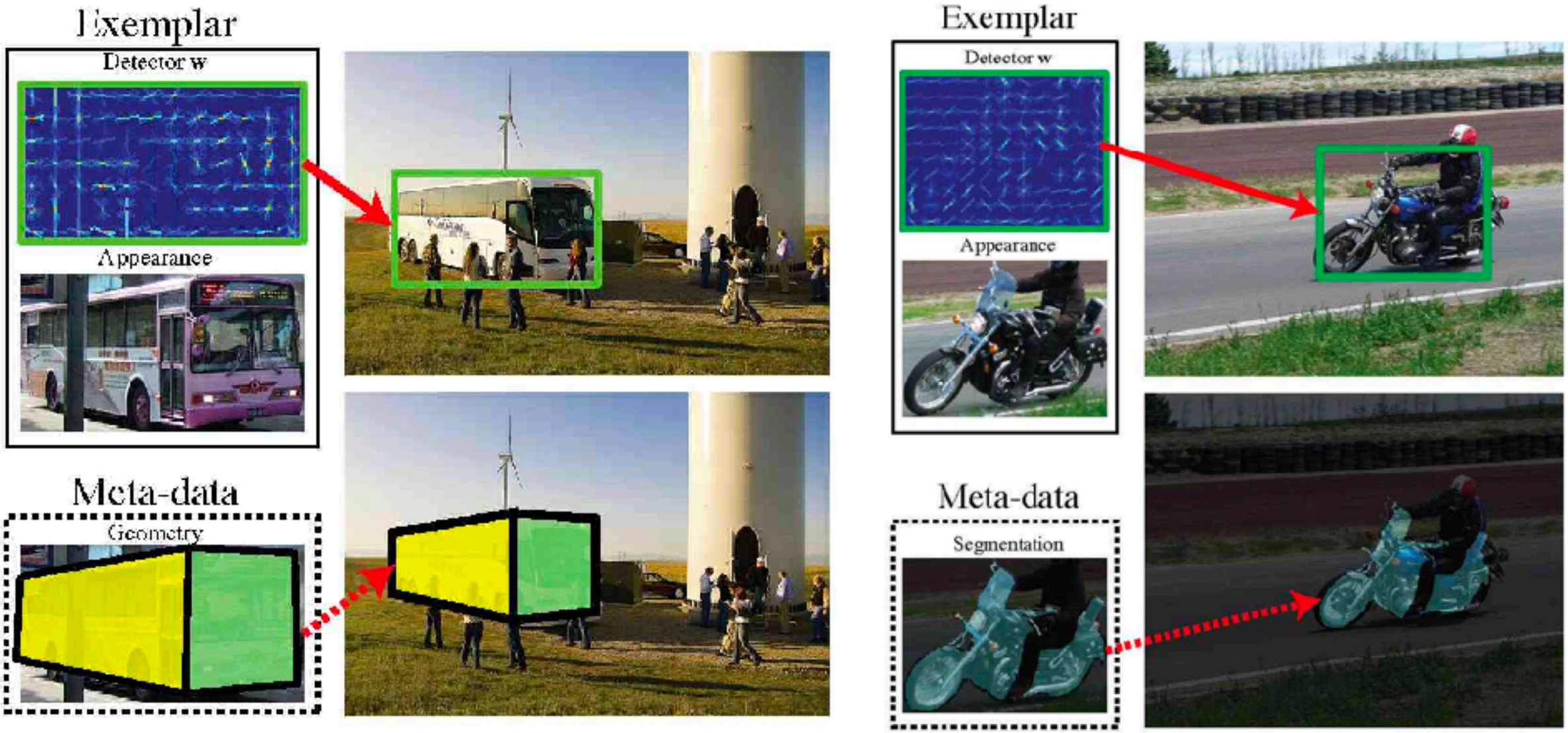
Taxonomies
(dogs and cats are types of mammals)



Spatiotemporal Relations
(people stand on floors)

Associative memory

Big-data philoshopy: ask not “what is this”, but “what is this like”?

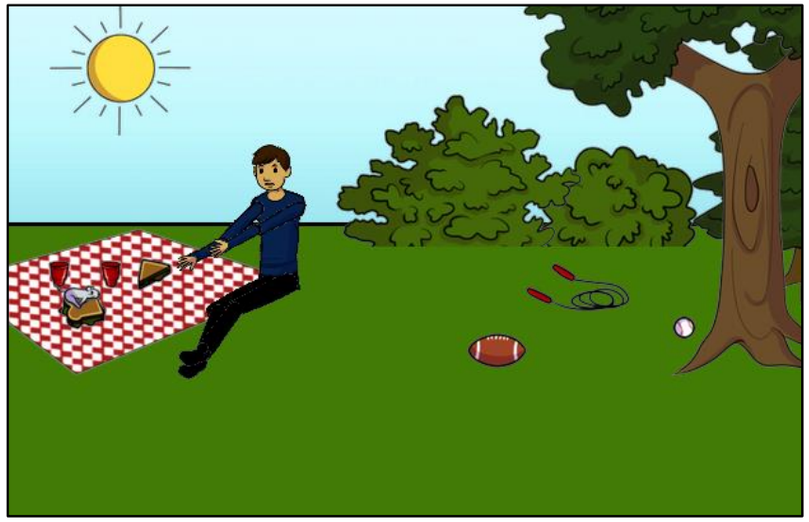


Malisiewicz et al, “Exemplar SVMs”

Visual Question Answering (VQA)

What color are her eyes?
What is the mustache made of?

How many slices of pizza are there?
Is this a vegetarian pizza?



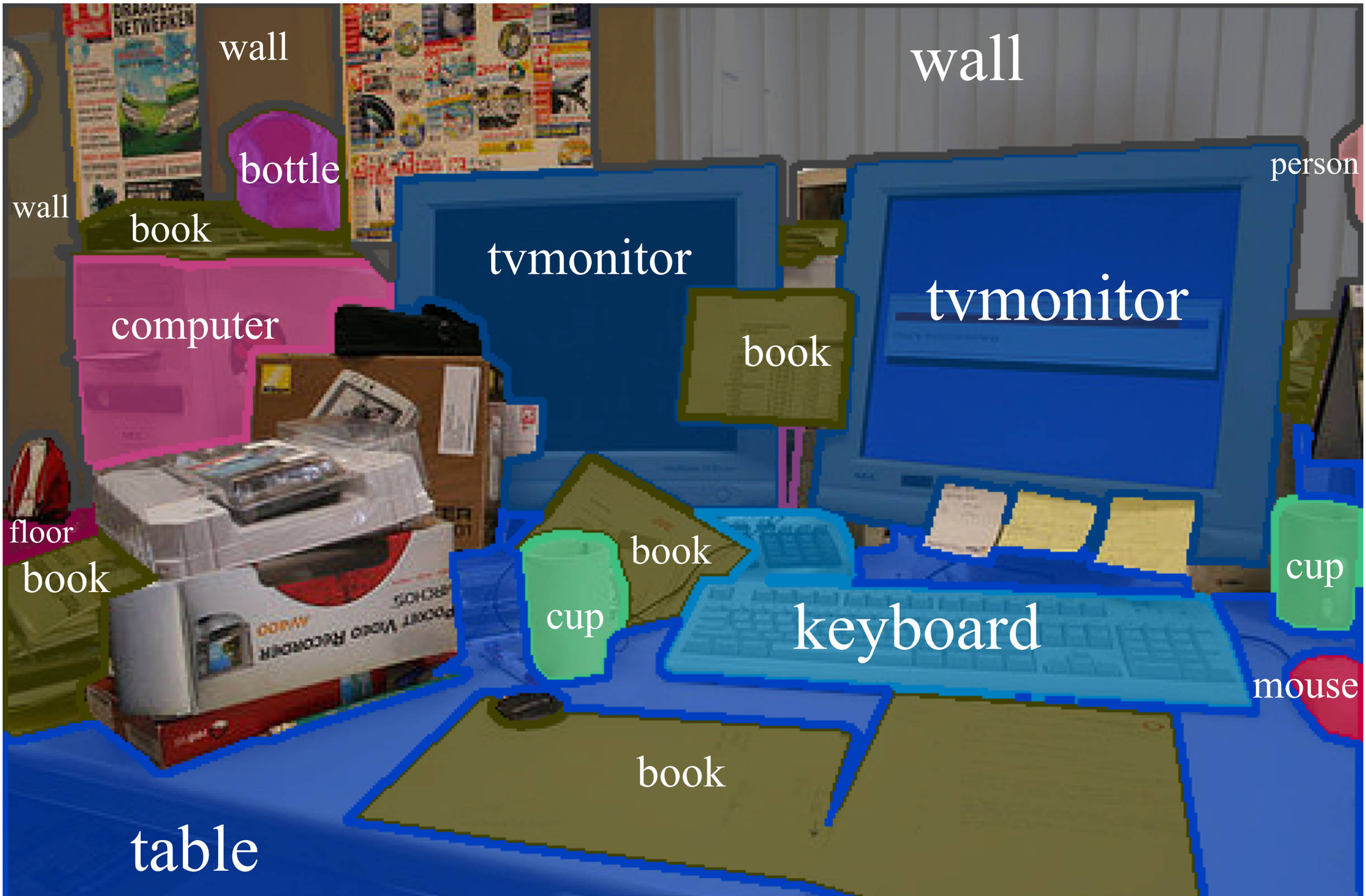
Is this person expecting company?
What is just under the tree?

Does it appear to be rainy?
Does this person have 20/20 vision?

Generate questions via Amazon MTurk:

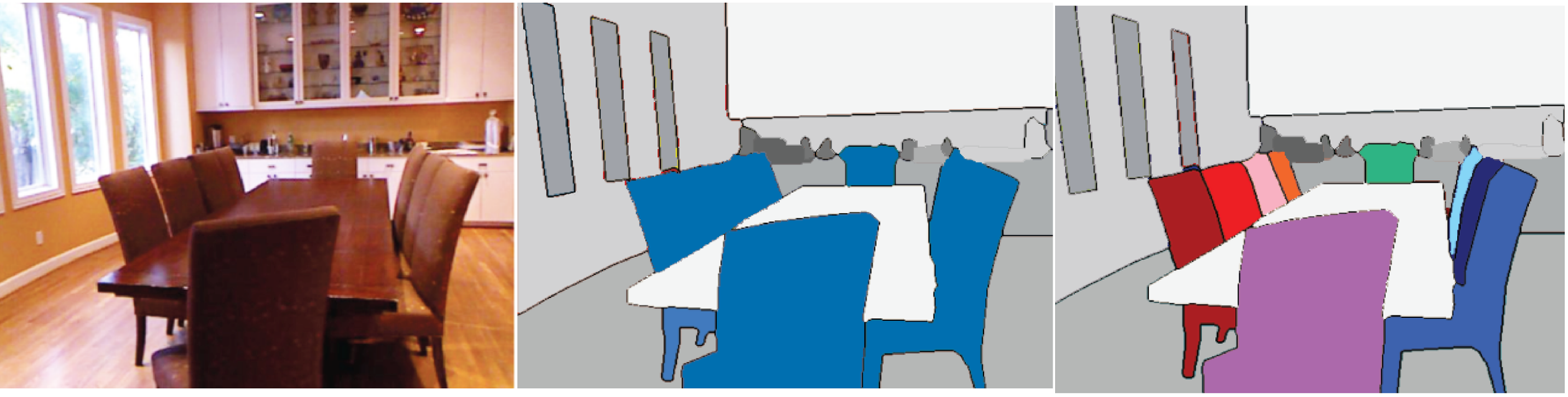
"*We have built a smart robot. It understands a lot about images. It can recognize and name all the objects, it knows where the objects are, it can recognize the scene* (e.g., *kitchen, beach), people's expressions and poses, and properties of objects* (e.g., *color of objects, their texture). Your task is to stump this smart robot!*".

My (current) favorite answer: semantic image segmentation



A “detail”: instance segmentation

Augment semantic labels with an instance ID



Let’s give it a try

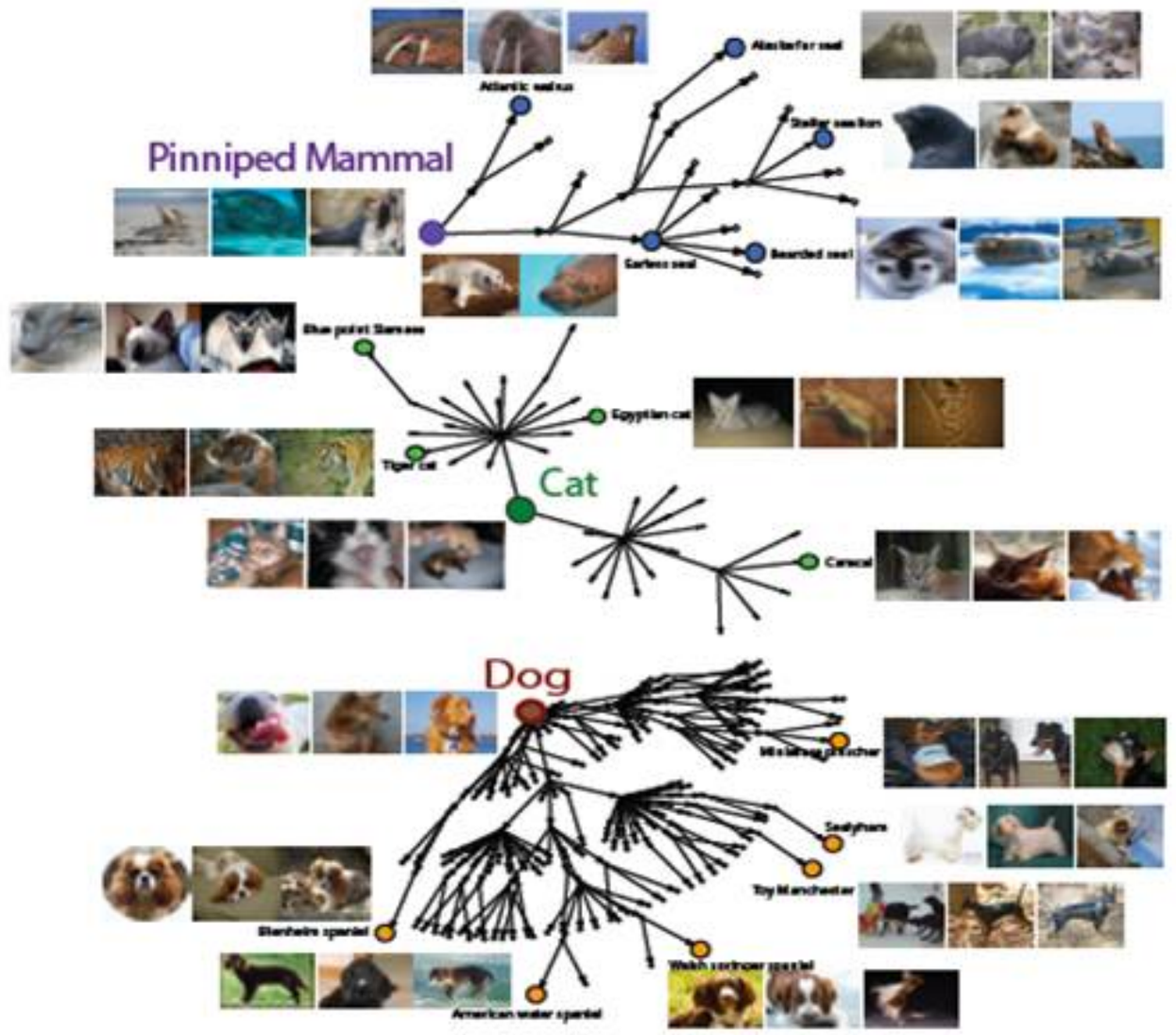
Notes on image annotation

Adela Barriuso, Antonio Torralba
Computer Science and Artificial Intelligence Laboratory (CSAIL), Massachusetts Institute of Technology

“ I can see the ceiling, a wall and a ladder, but I do not know how to annotate what is on the right side of the picture. Maybe I just need to admit that I can not solve this picture in an easy and fast way. But if I was forced ”

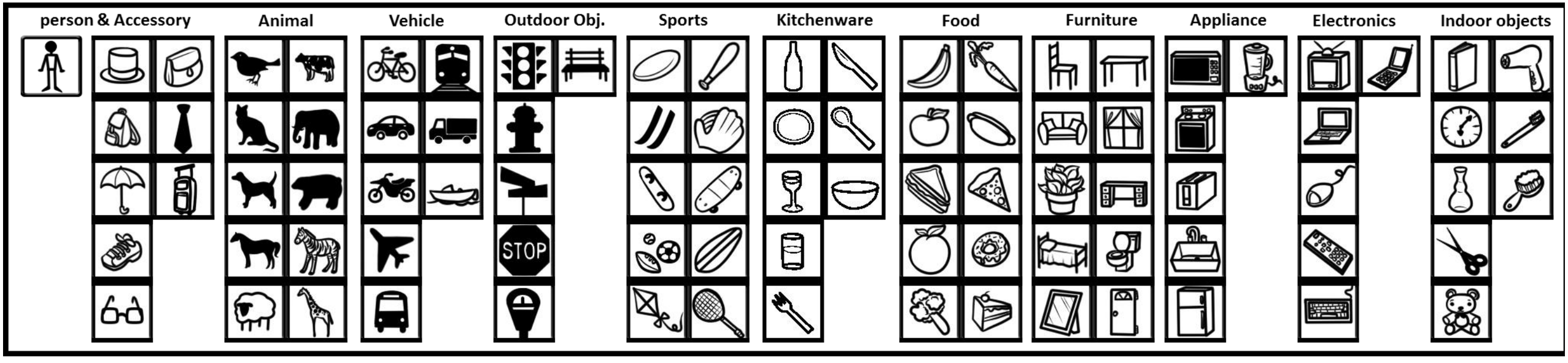
Semantic blindspots

Can we define a canonical list of objects, attributes, actions, materials….?



ImageNet (cf. WordNet, VerbNet, FrameNet,..)

Crowdsourced dataset construction



Task: select images that have WRONG object contour for toothbrush.
Examples:

Right Object Contour

Wrong Object Contour (not toothbrush, only contains parts of visible object contour, or multiple objects)

Tips: use n and b keys to move between rows of image.

1

2

3

4

"If you know what can be done with a ... object, what it can be used for, you can call it whatever you please"
J. J Gibson [14]

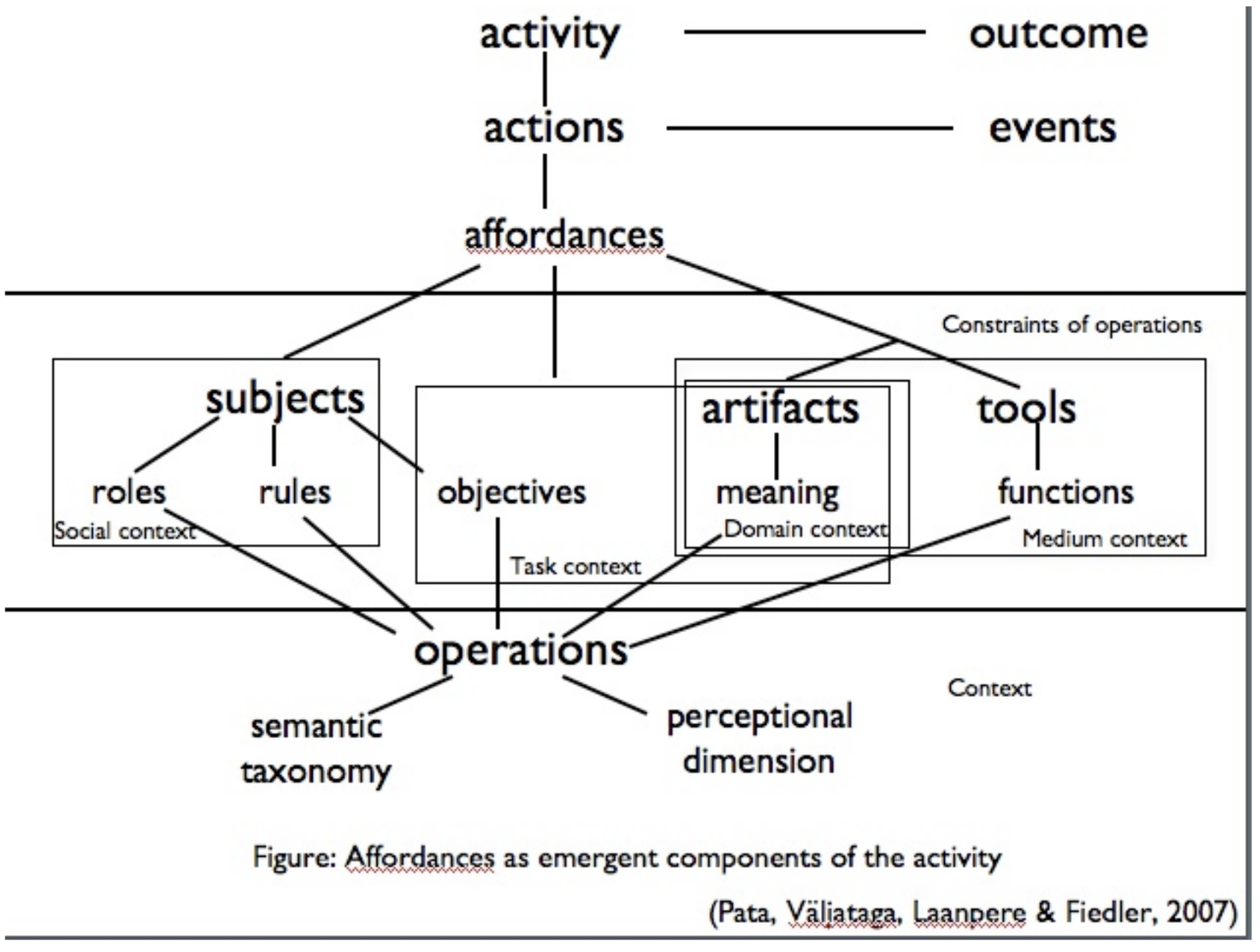


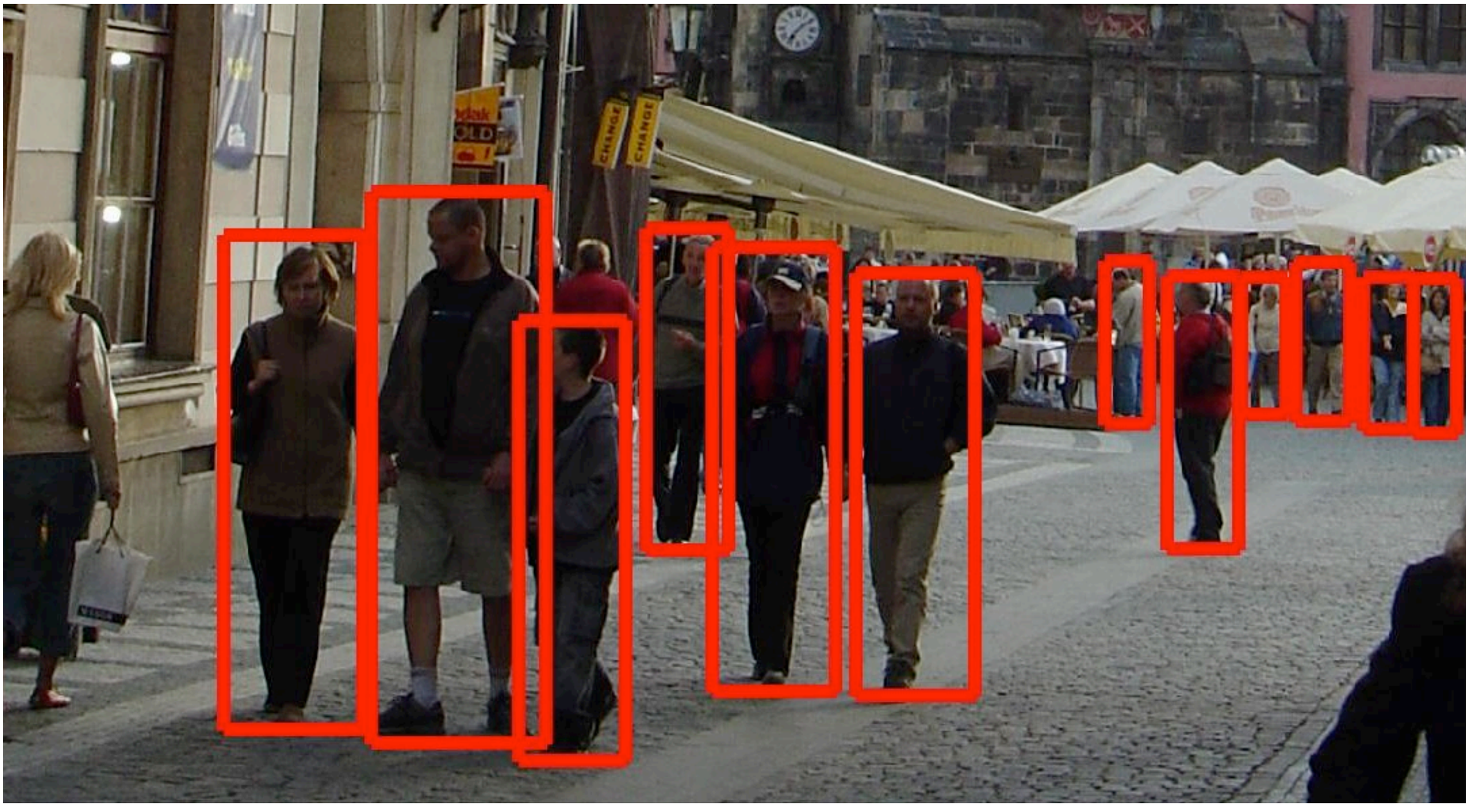
Figure: Affordances as emergent components of the activity

(Pata, Väljataga, Laanpere & Fiedler, 2007)

Outline

- Motivation
- How to define problem?
- Case example: finding people
- Statistical classification

Goal: detecting pedestrians



Thought experiment: let's build a person detector (HW4).
Why is this difficult?

variation in illumination

variation in appearance

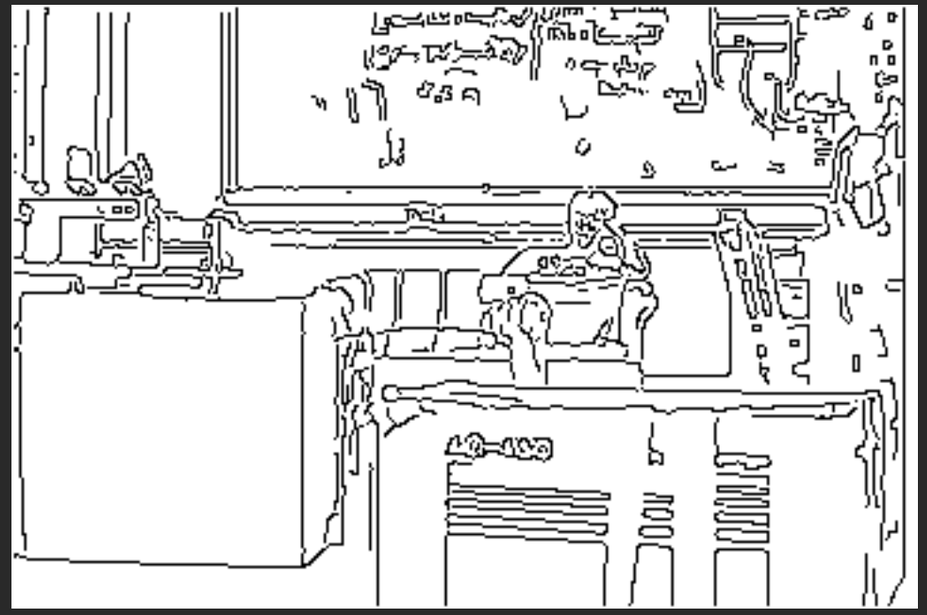
variation in pose, viewpoint

occlusion & clutter

Classic "nuisance factors" for general object recognition

Main idea: use “invariant features”

edges!



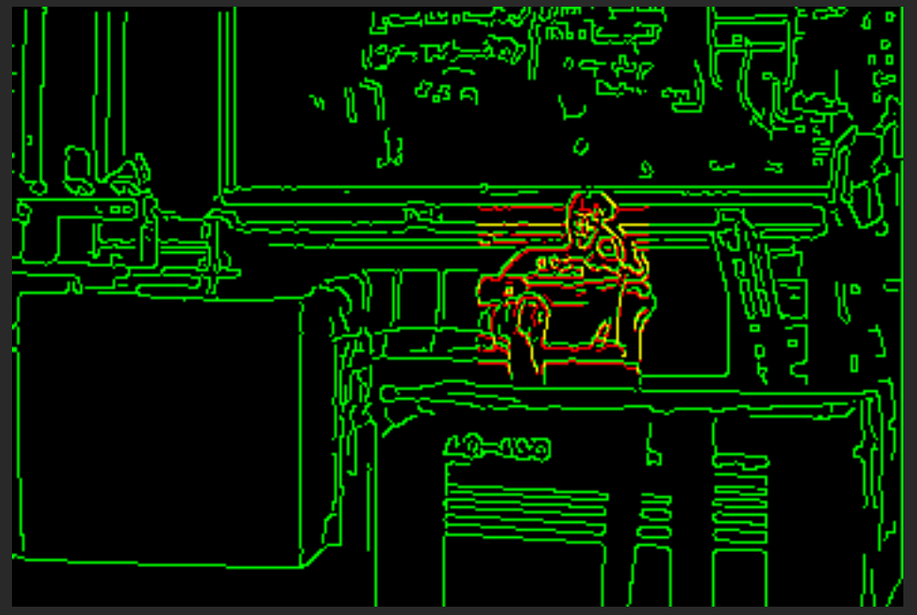
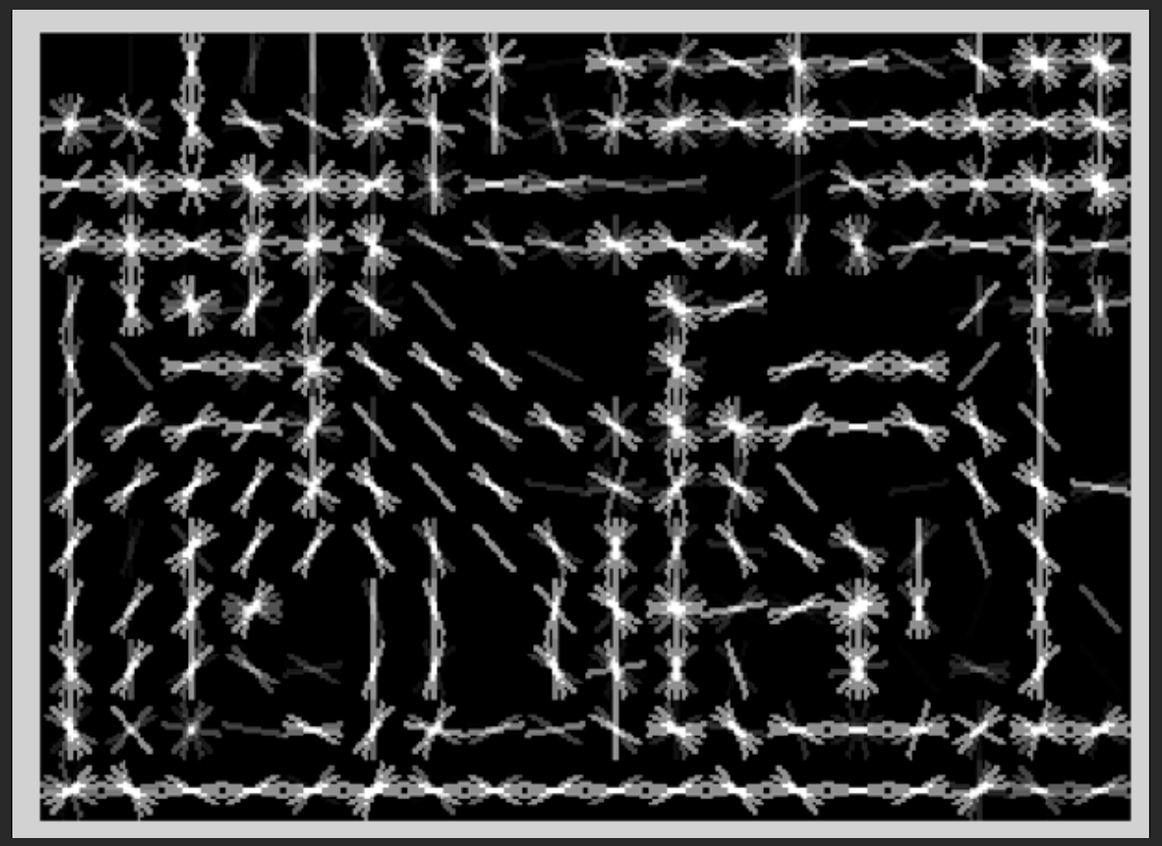


Image features:

Histograms of oriented gradients (HOG)

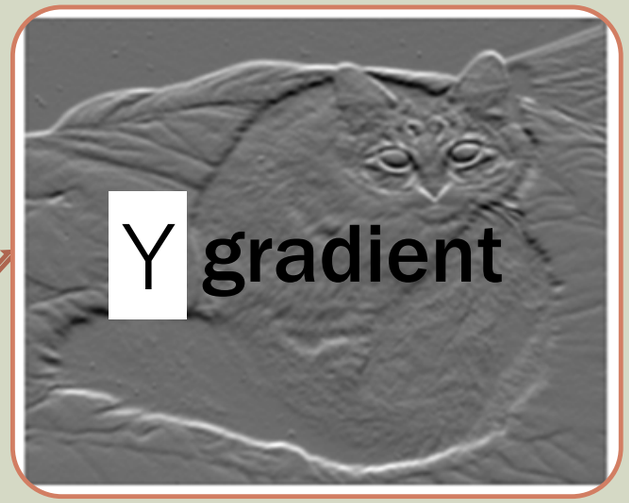


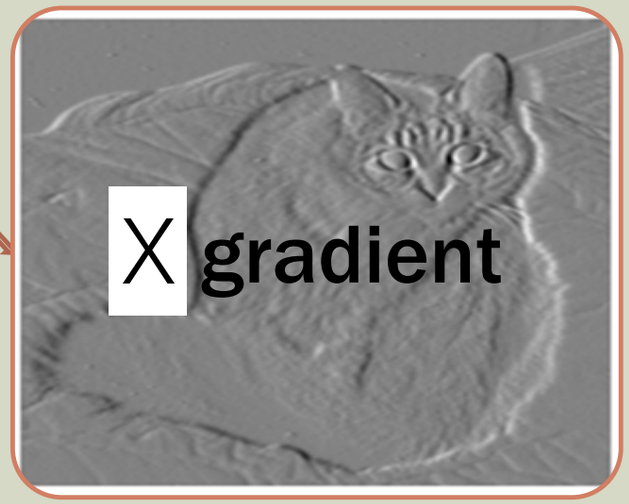
Bin gradients from 8x8 pixel neighborhoods into 9 orientations

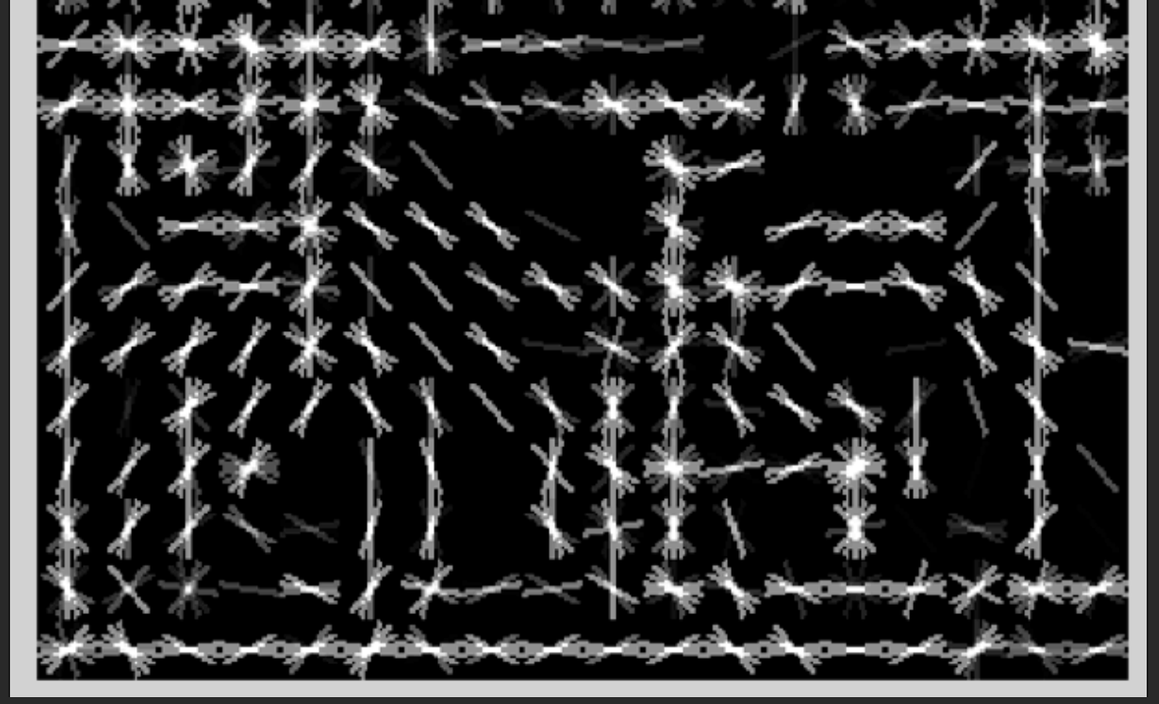
(Dalal & Triggs 05)

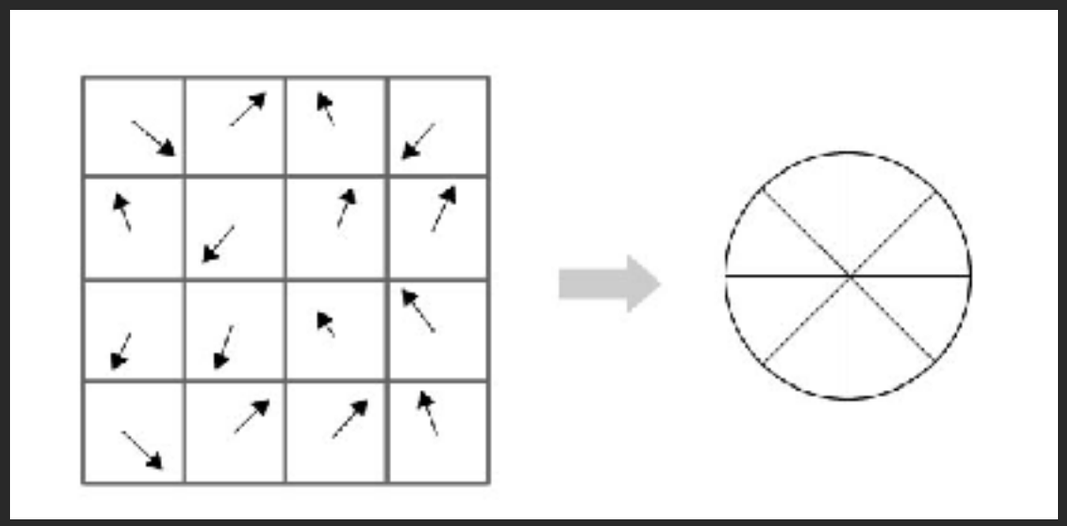
(Simplified) HOG construction

- Convolve the image with discrete derivative mask
  - [-1,1]
  - [-1,1]’







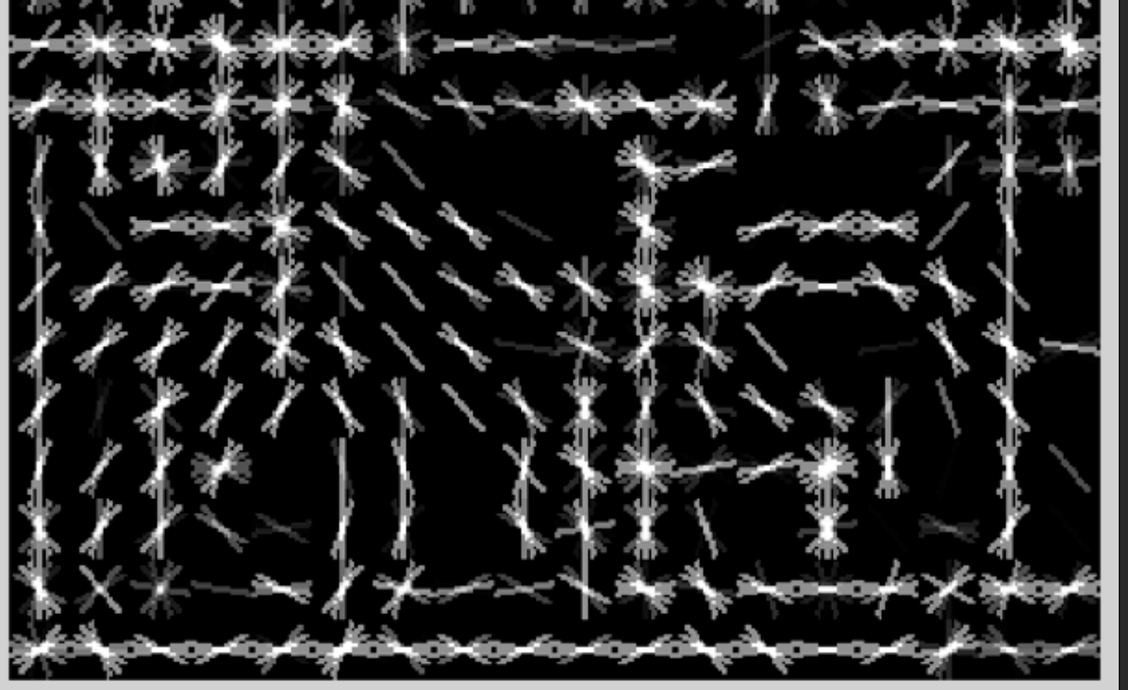


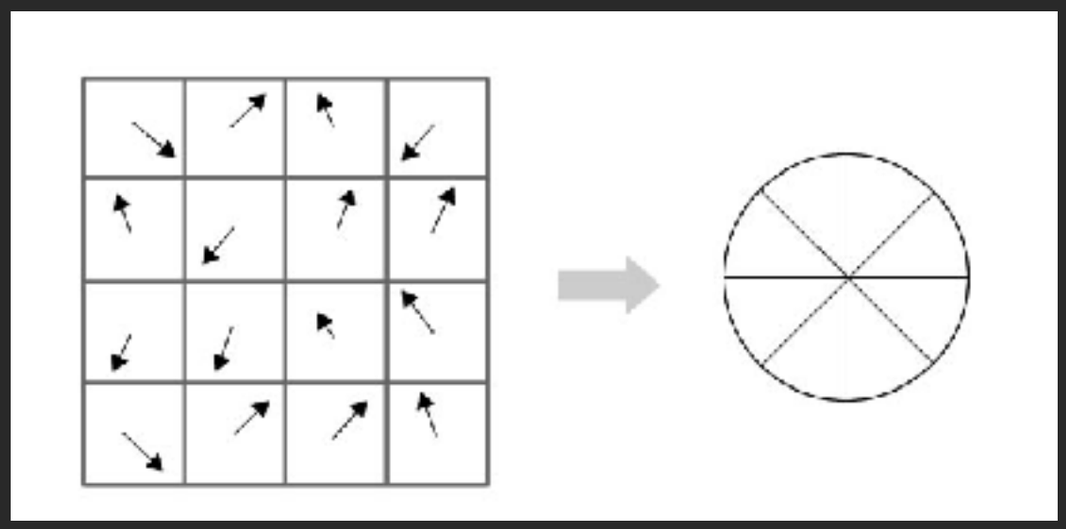
Quantize each gradient into one of $n_o = 9$ orientations

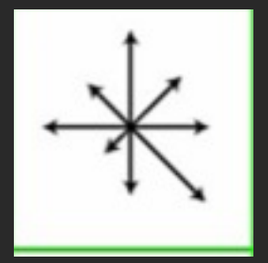
Do we want to put gradients 180 apart in same or different bins?

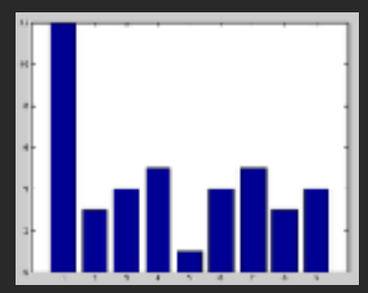
What should be the angle range of each bin?

[H x W] ->[H x W x 9]
“orientation channel array”





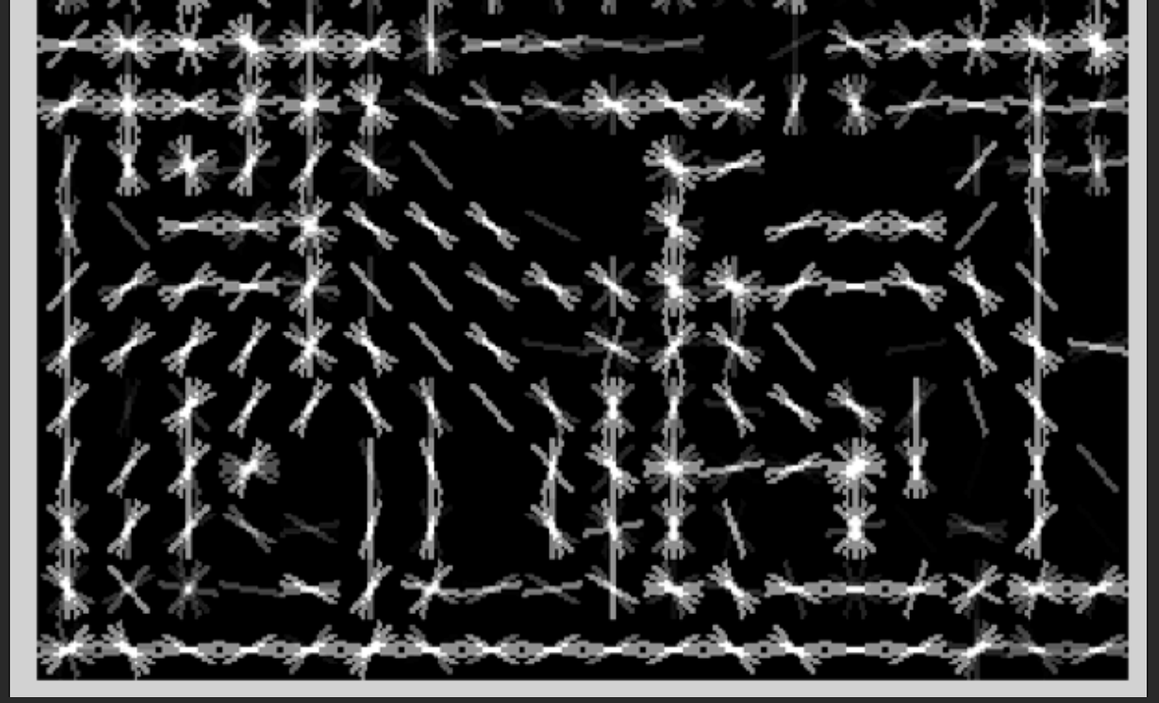


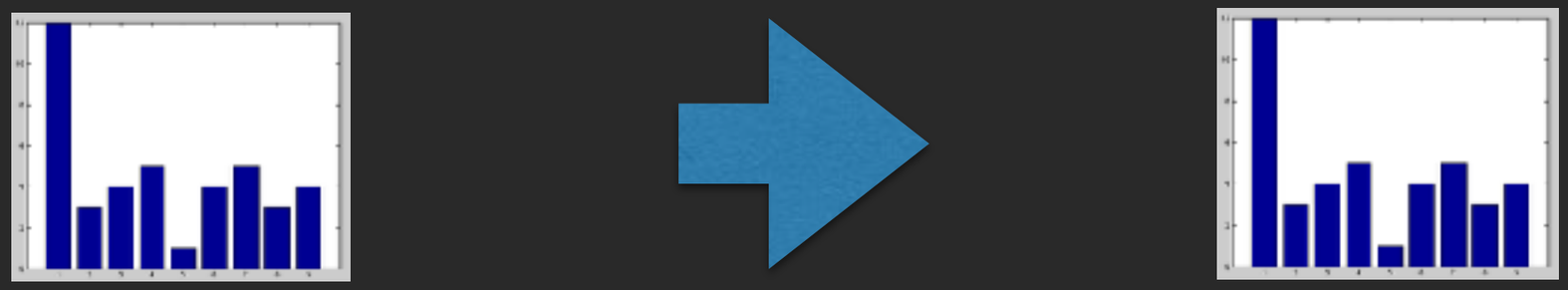


[H x W x 9] – > [H/8 x W/8 x 9]

Count up orientation bins over 8x8 pixel neighborhoods. (im2col)

Get some spatial invariance (sort of)…





[H/8 x W/8 x 9] – > [H/8 x W/8 x 9]

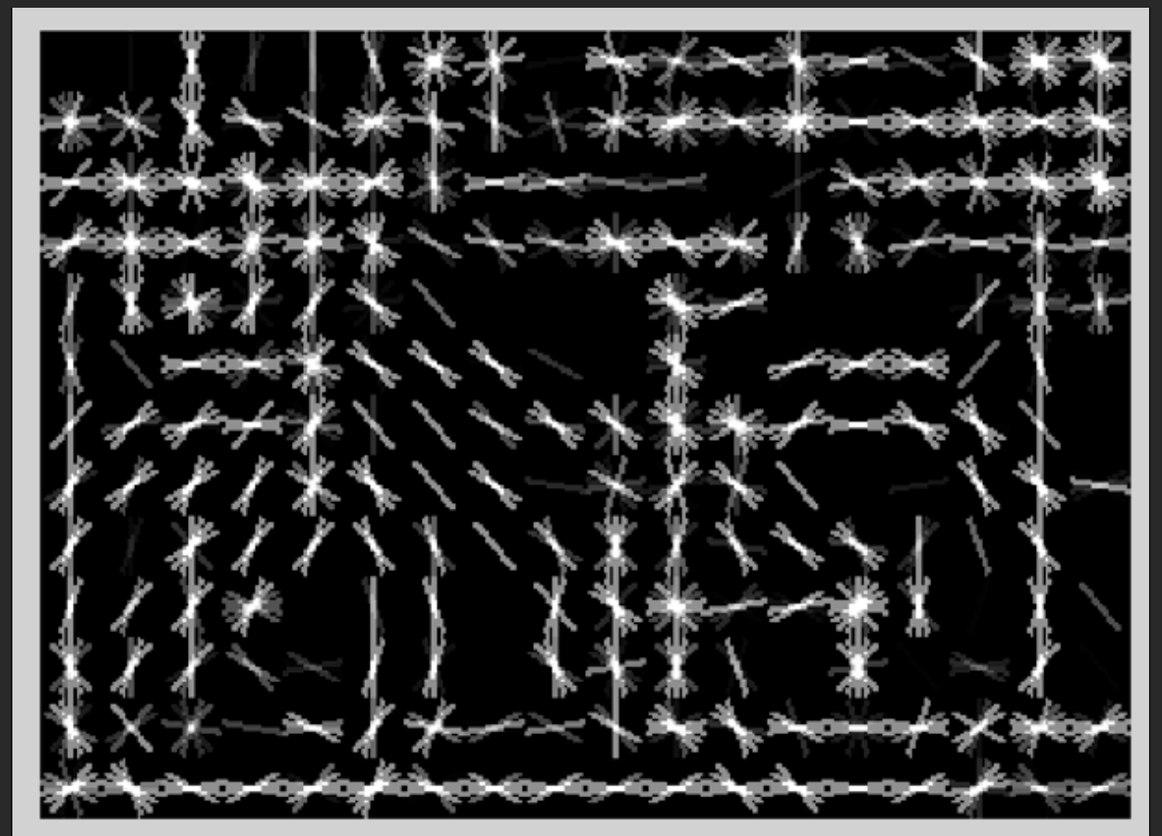
Re-normalize 9 numbers so that their sum is 1

Get some lighting invariance (sort of)…

Histograms of oriented gradients (HOG)

(note that actual HOG construction is a bit more intricate)

1. Work with raw gradients instead of thresholded gradients
2. Normalize with respect to histograms of 2x2 neighborhoods

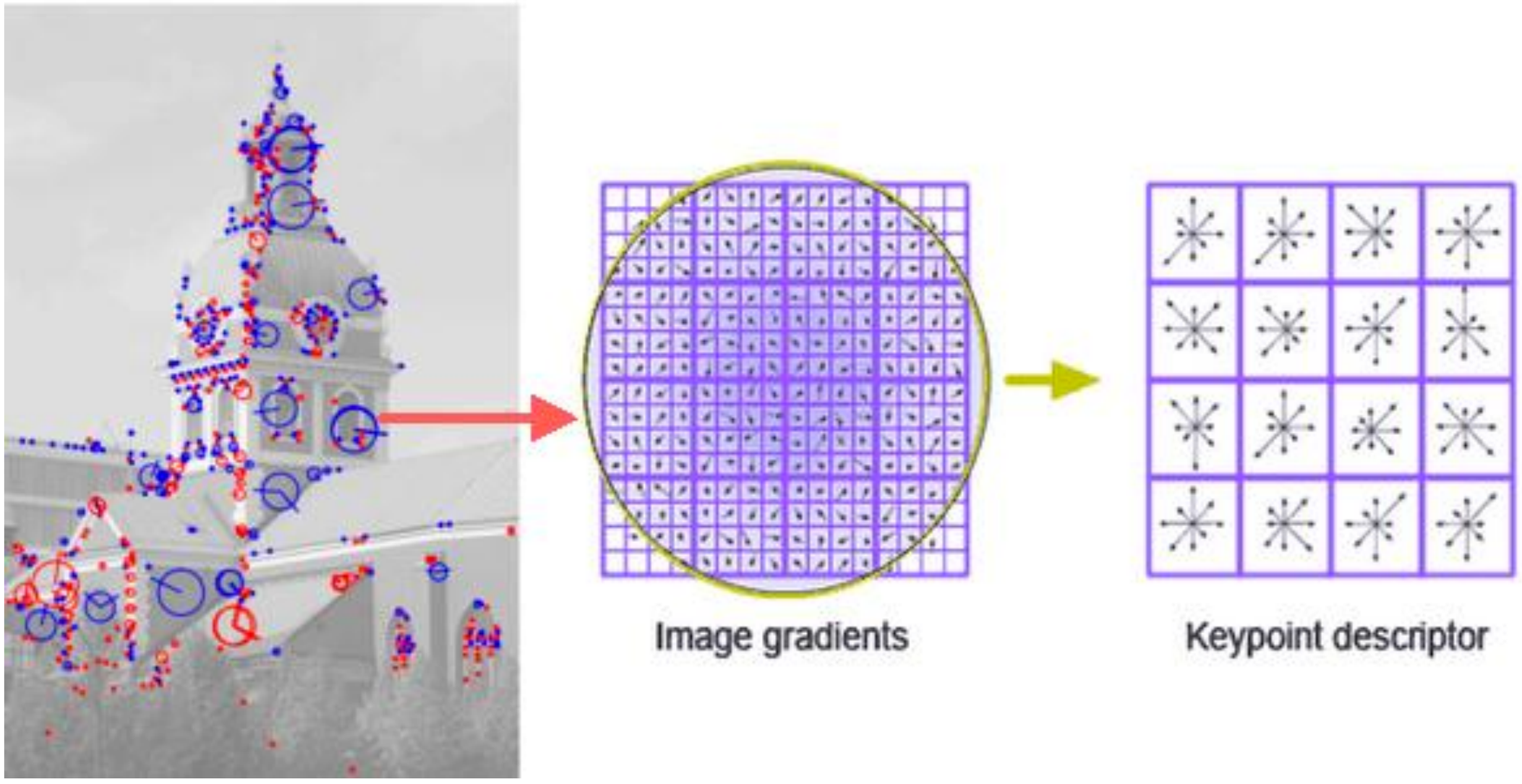


Bin gradients from 8x8 pixel neighborhoods into 9 orientations

Image [H x W] -> Image Descriptor = [H/8 x W/8 x 9]

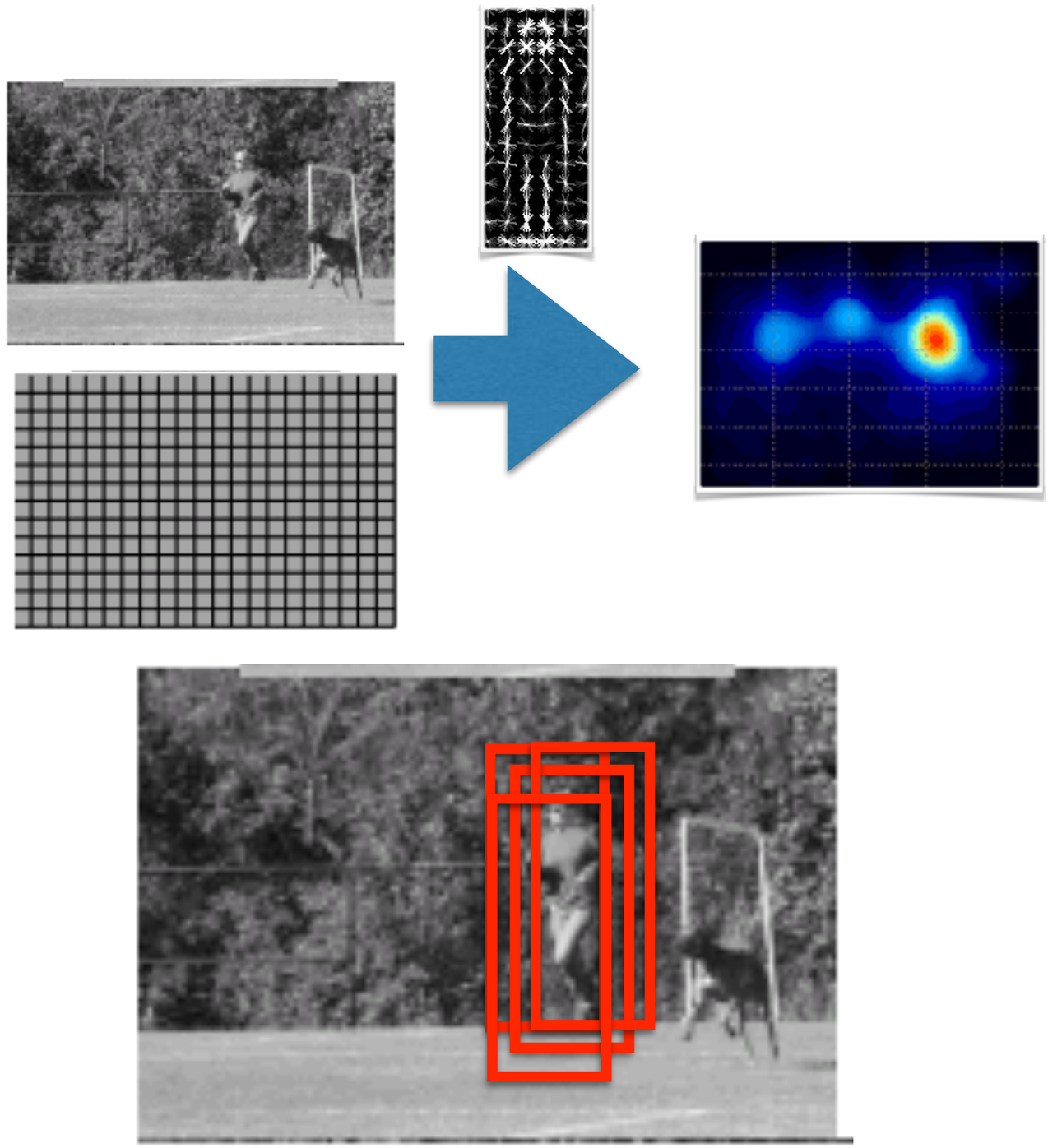
Recall SIFT

http://en.wikipedia.org/wiki/Scale-invariant_feature_transform



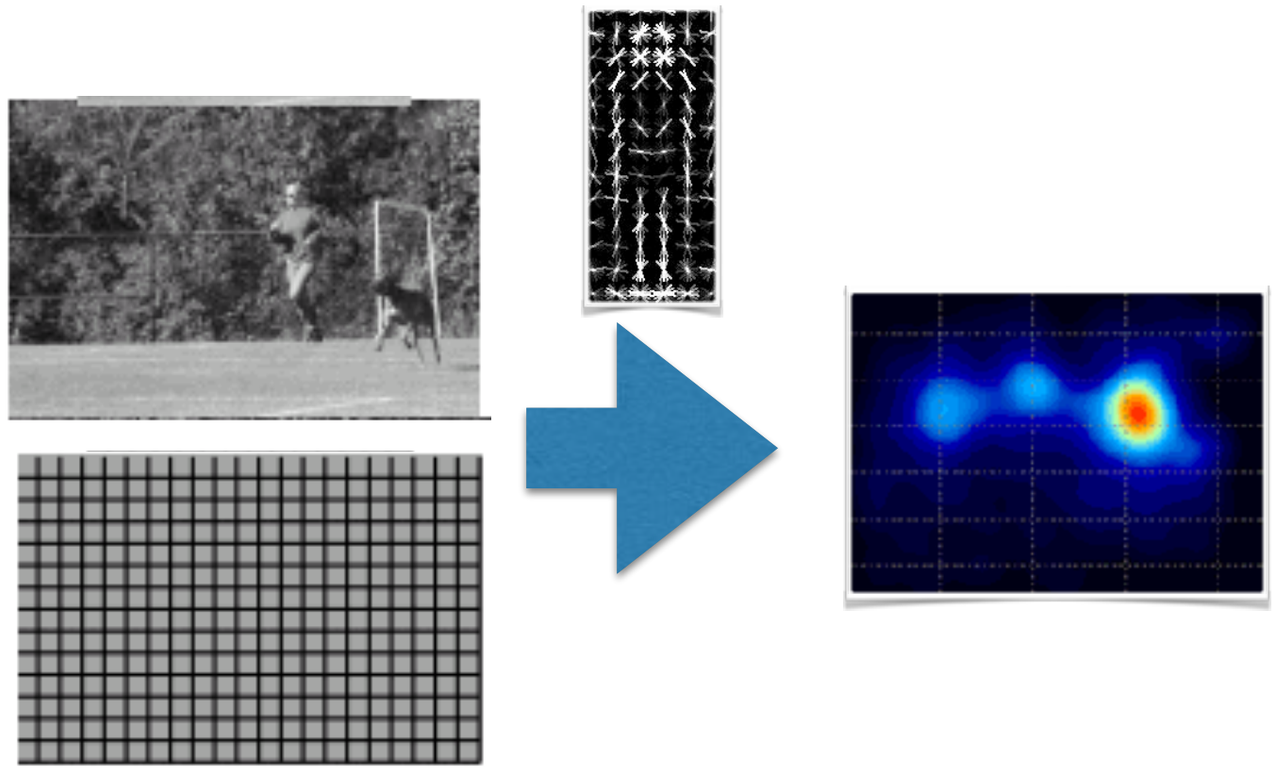
One can interpret HOG precisely as dense grid of SIFT descriptors (of size 2x2x9), computed at grid points of 8x8 pixel shifts

Template scoring



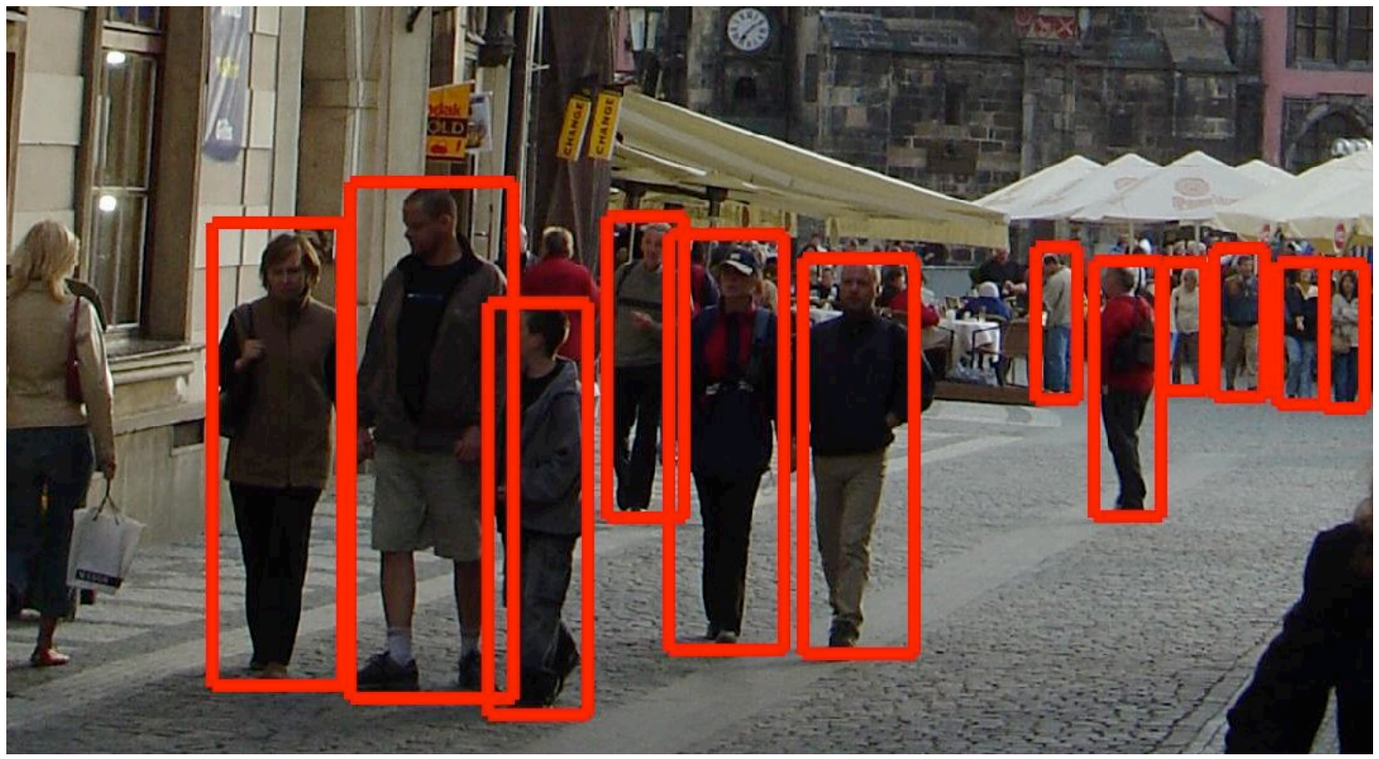
How can we fix output from naive thresholding?

Nonmaximal suppression



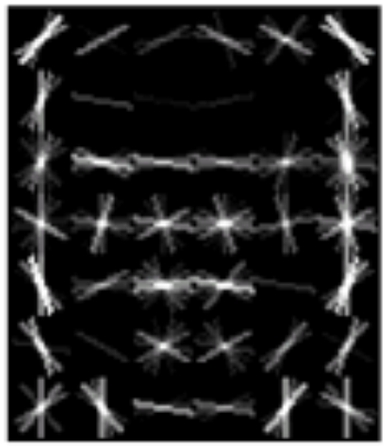
1. Find highest scoring location
2. Zero-out responses that overlap
3. Repeat until highest remaining score is below a threshold

Pedestrian detection

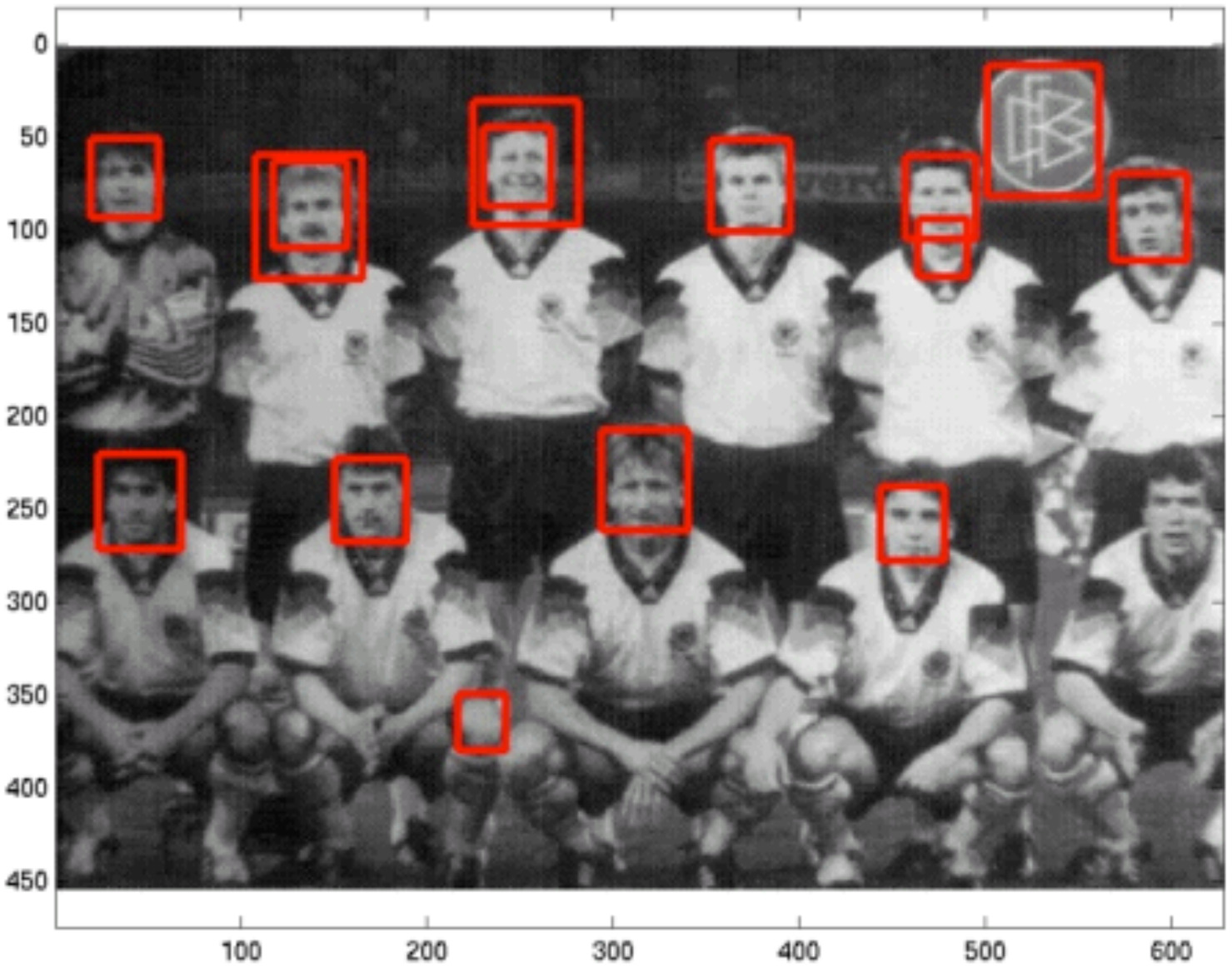


Dalal and Triggs “Histograms of Gradients”

Face detection



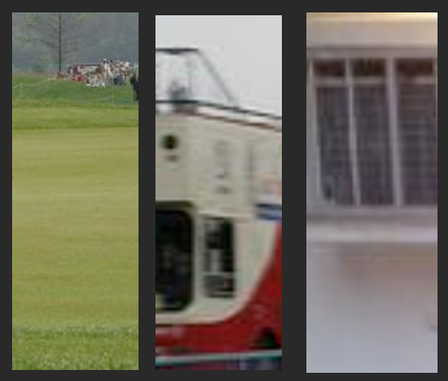
template

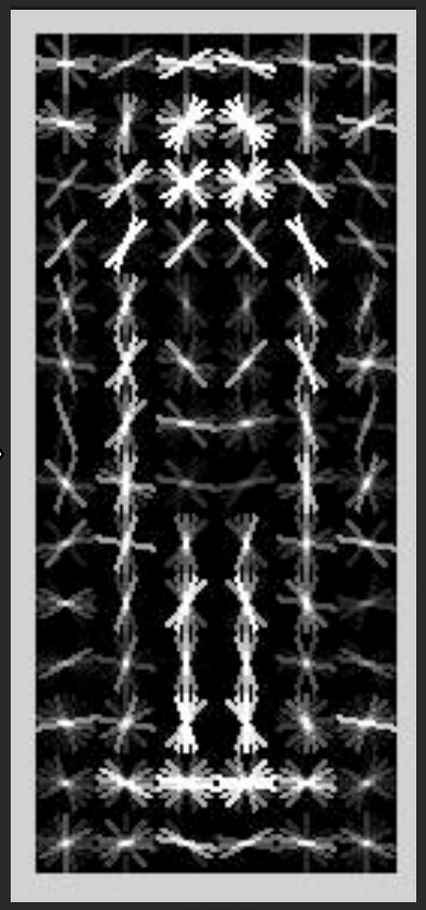


Template classifiers

pos

neg





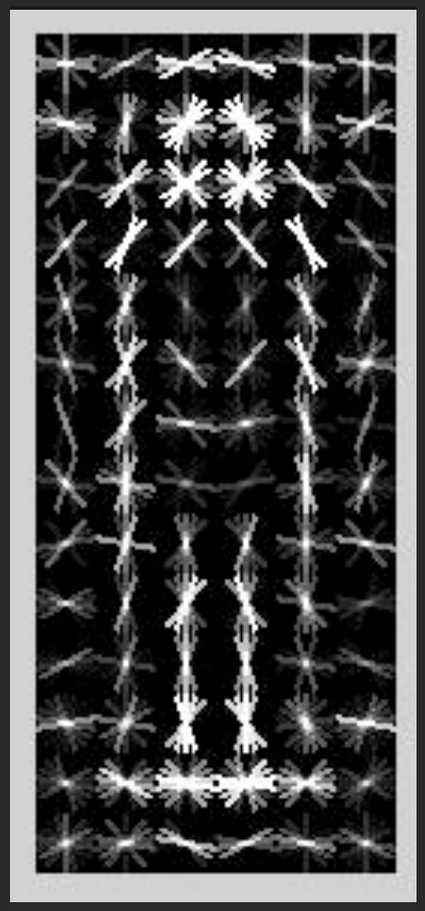
w

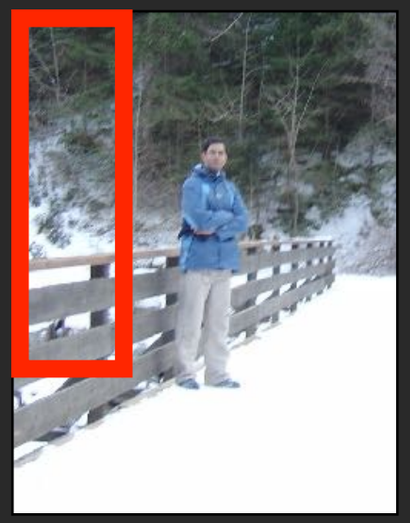
w = weights for orientation and spatial bins 

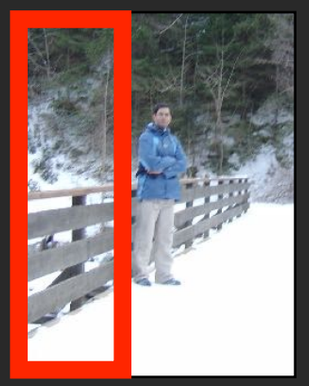
$w \cdot x > 0$

Train with a linear classifier (perceptron, logistic regression, SVMs...)

Search over scales







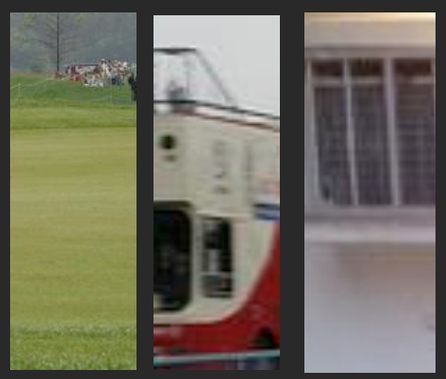
Outline

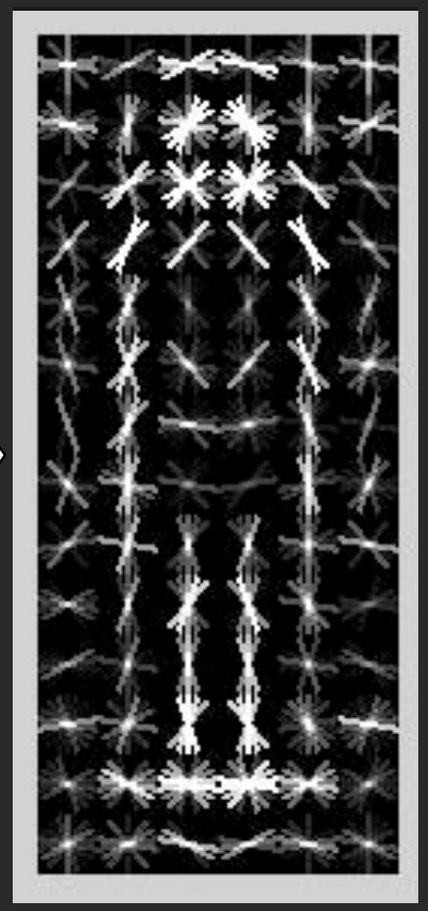
- Motivation
- How to define problem?
- Case example: finding people
- Statistical classification

Template classifiers

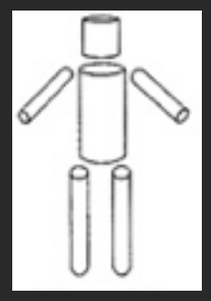
pos

neg





contrast with “old school”:

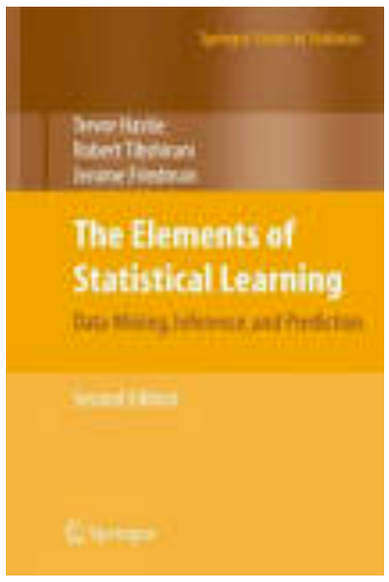


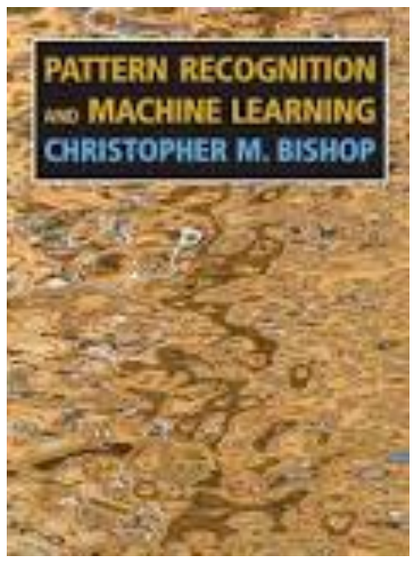
Statistical classification (20 min review!)

Why?

- This is the world in which we live - statistical models from data overpower classic “hand-designed” models
- Basic linear classification forms basis for nonlinear models (deep learning)

Good texts:



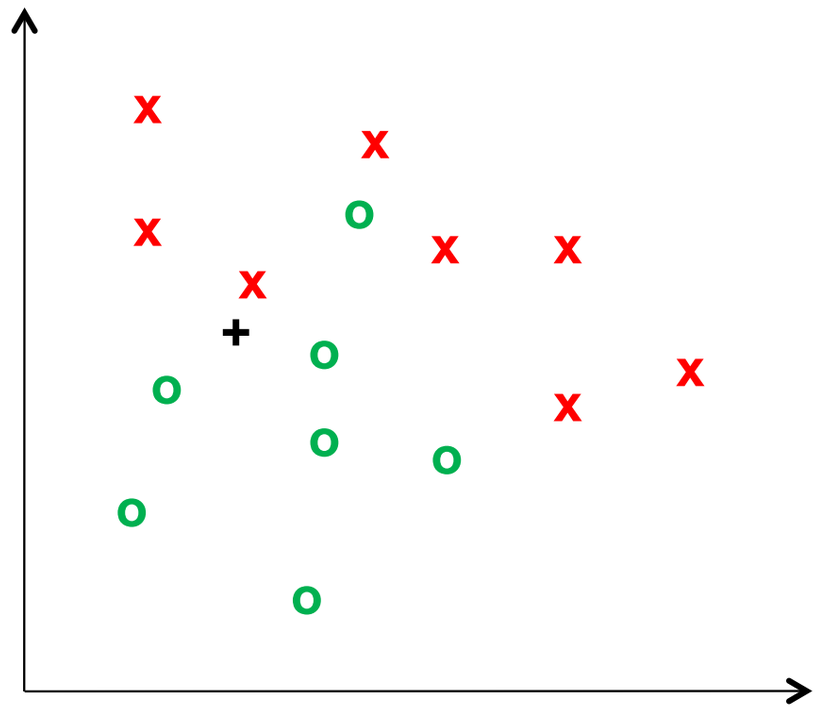


Statistical classification

$x \in R^N$

$y \in \{-1, 1\}$

$\{(x_i, y_i)\}$



Recall N ~ 10x5x9 for typical HOG templates

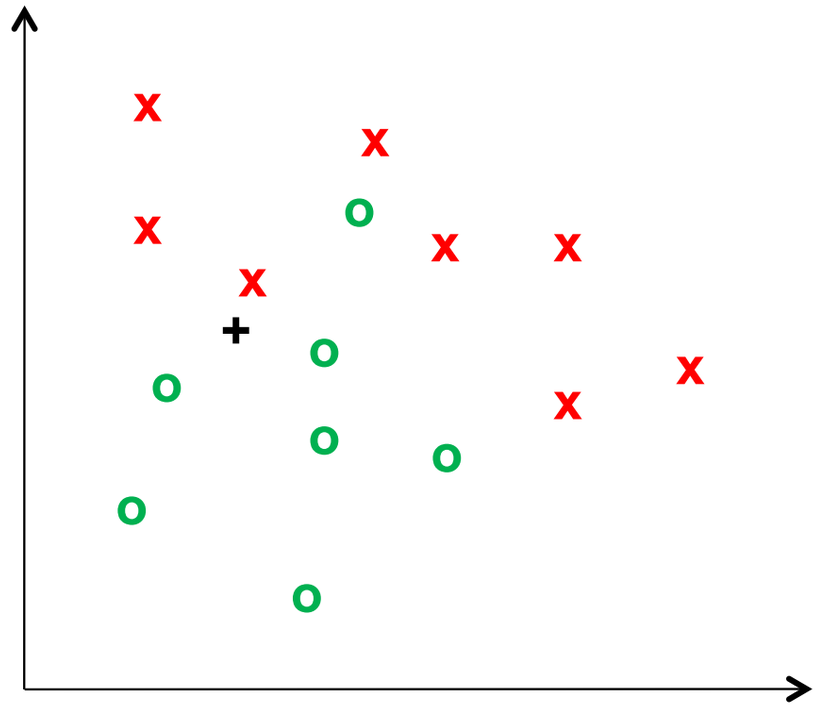
Given training points (xi,yi), learn function f(x) that predicts a label {-1,1}

Statistical classification

$x \in R^N$

$y \in \{-1, 1\}$

$\{(x_i, y_i)\}$



Version 0: nearest neighbor classification

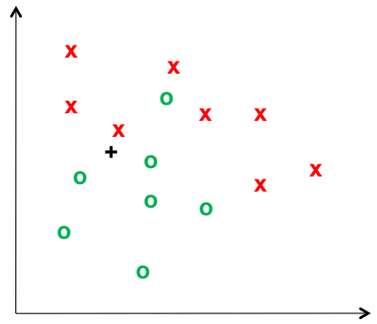
Train time:0
Test time: expensive

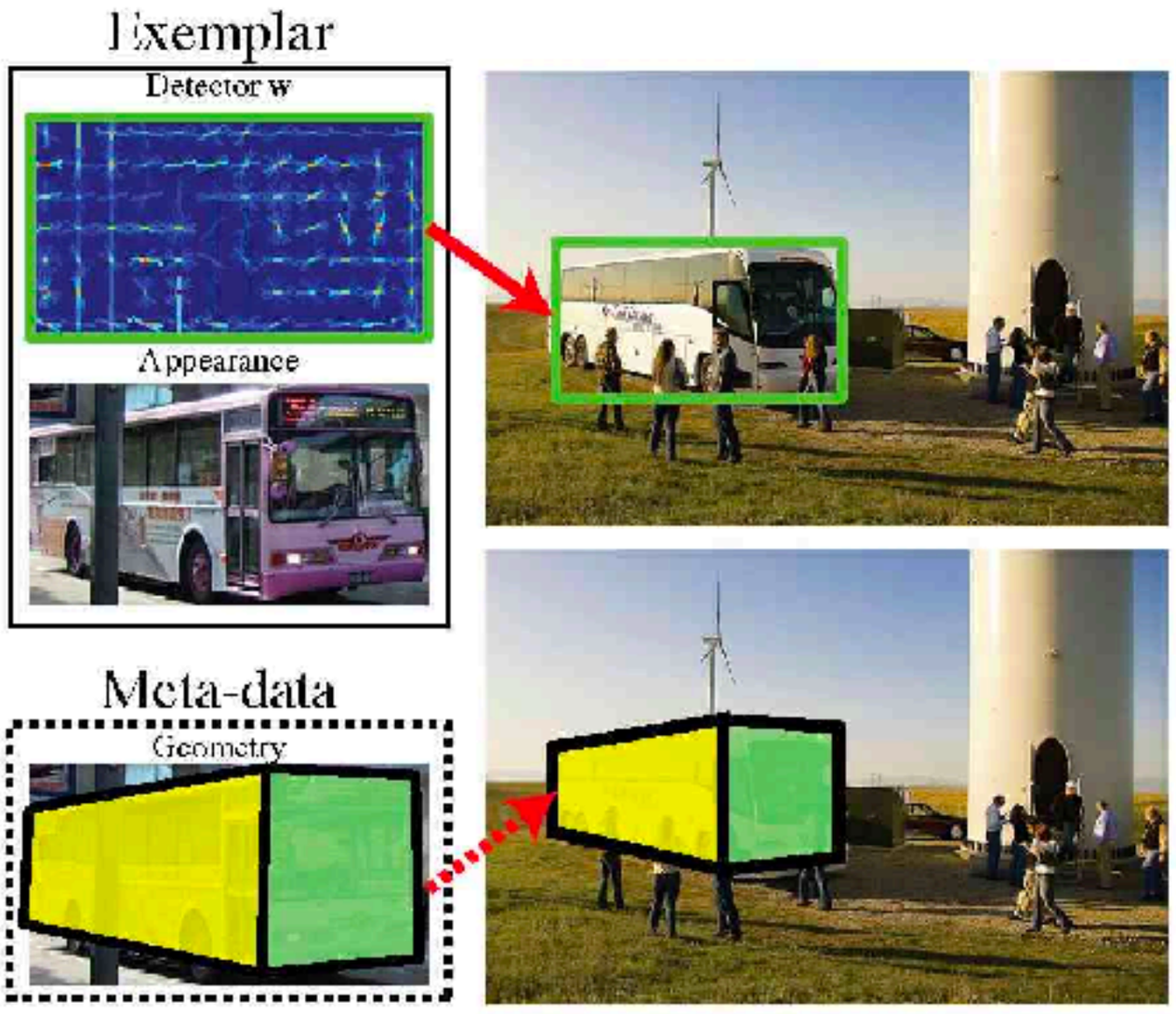
Trivially handles multiple classes

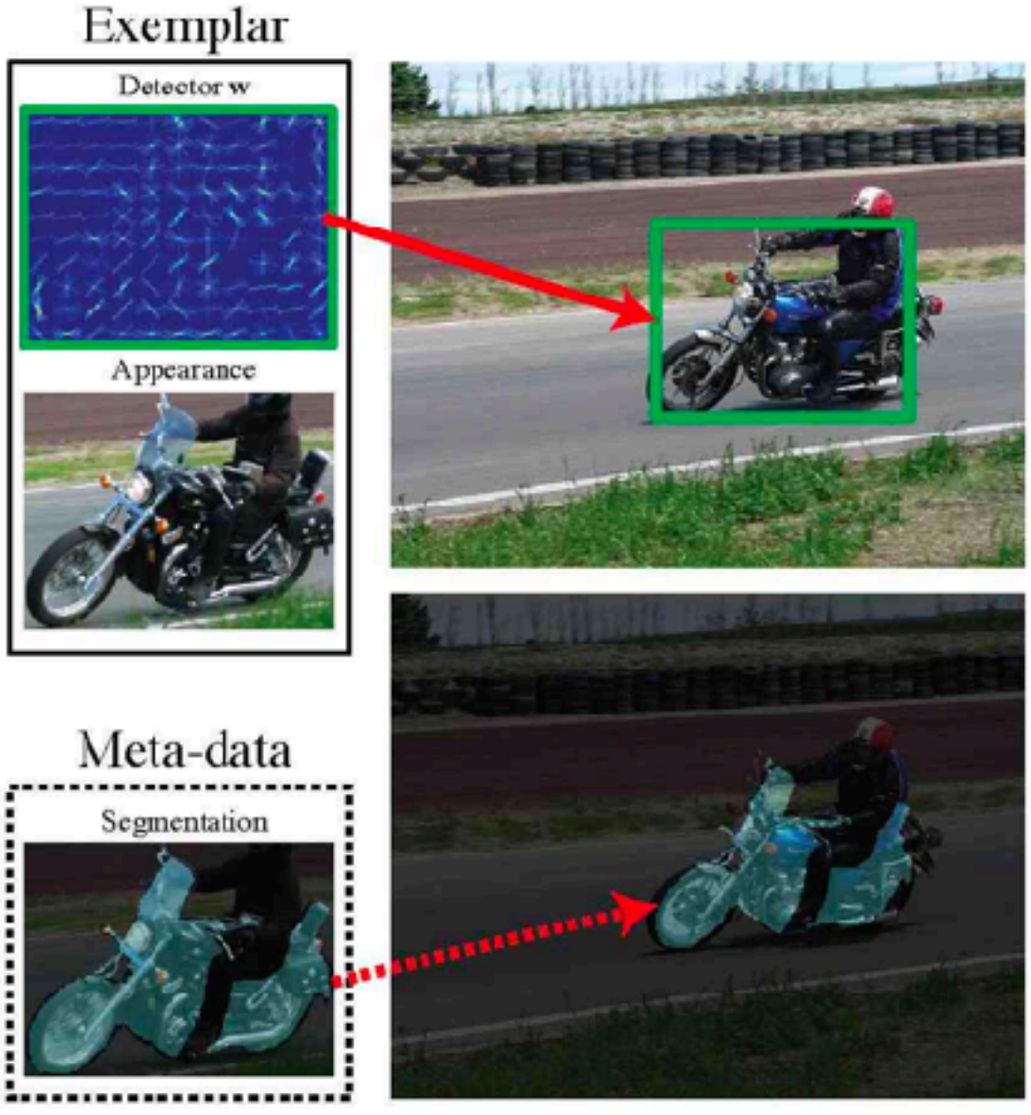
Is surprisingly powerful!

Statistical classification

Ask not what is this, but “what is this like”?







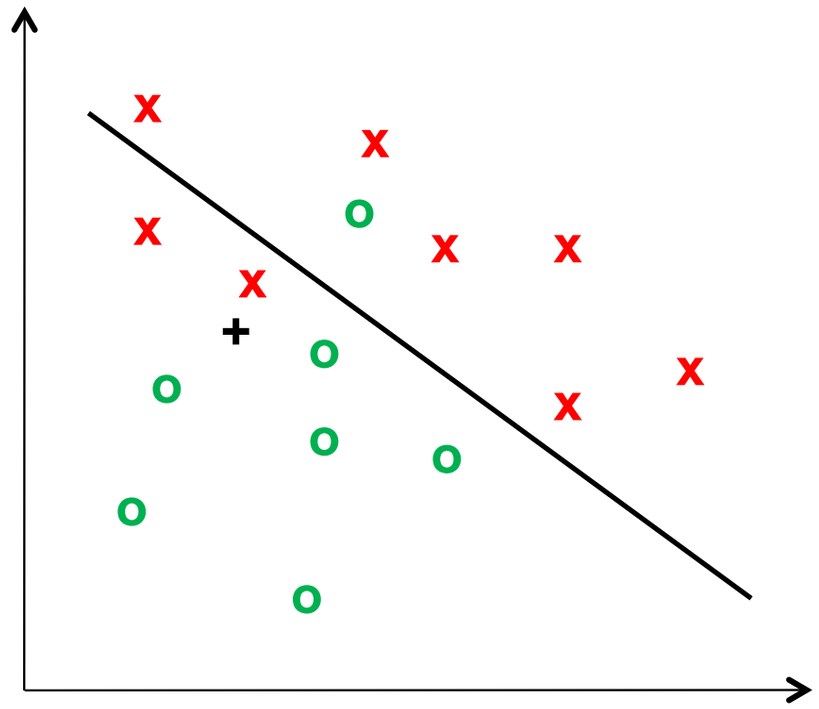
(in my view) this is the heart of all data-driven learning

Statistical classification

$x \in R^N$

$y \in \{-1, 1\}$

$\{(x_i, y_i)\}$

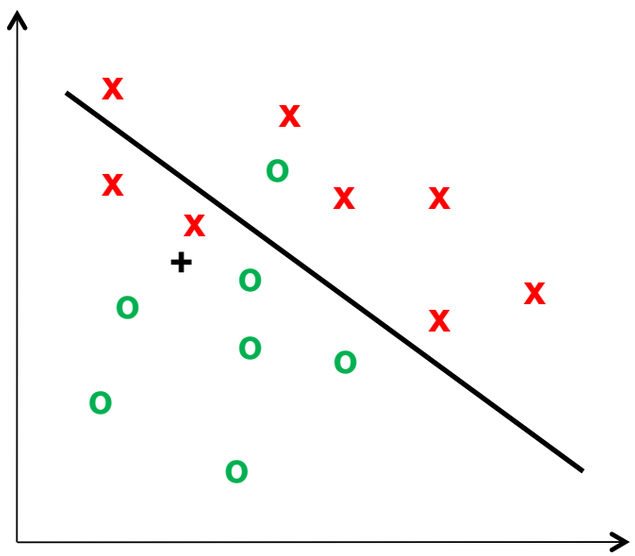


Given training points (xi,yi), learn function f(x) that predicts a label {-1,1}

Version 0: nearest neighbor classification

Version 1: linear classification

What's the best line?



$$\min_w \sum_i [y_i \neq thresh(w \cdot x_i)]$$

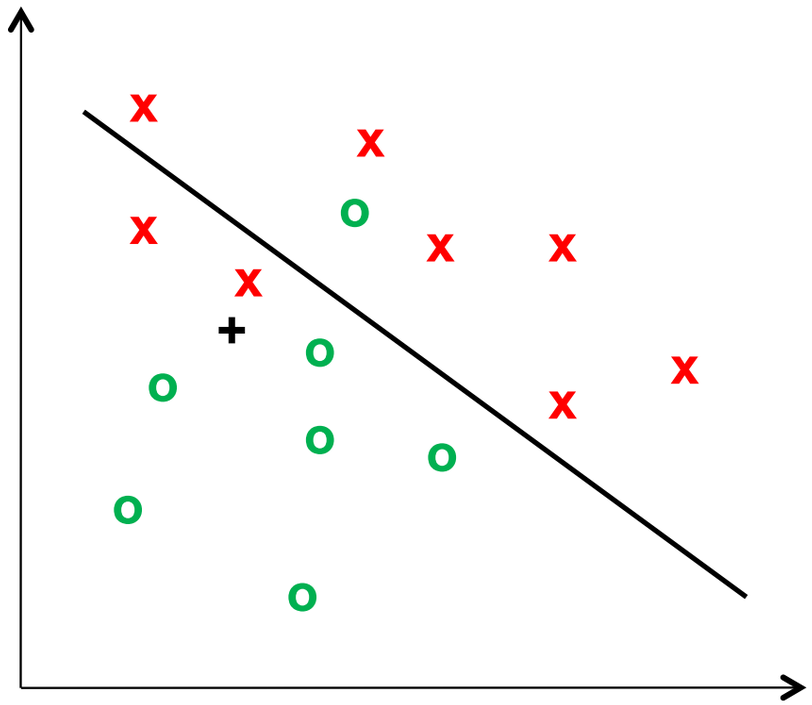
$$\text{thresh}(x) = \begin{cases} 1, & \text{if} \quad x > 0 \\ -1, & \text{otherwise} \end{cases}$$

Find w that minimizes mistakes on training data

[Hard to optimize]

Aside: doesn't this restrict line to go through origin?

What's the best line?

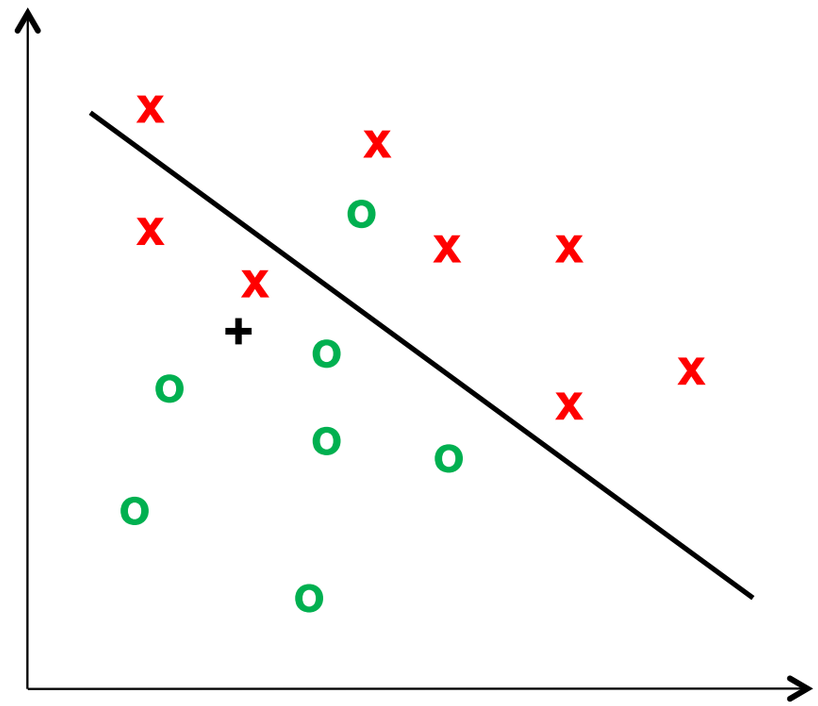


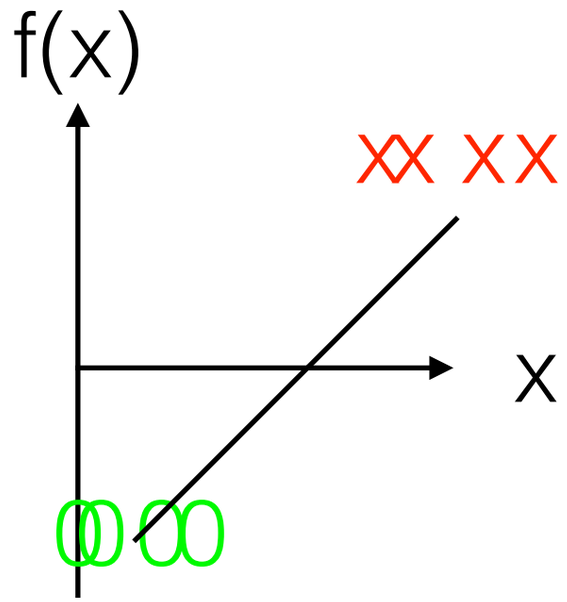
$$\min_w \frac{\lambda}{2}||w||^2 + \sum_i (y_i - w \cdot x_i)^2$$

Easy to optimize - least squares!

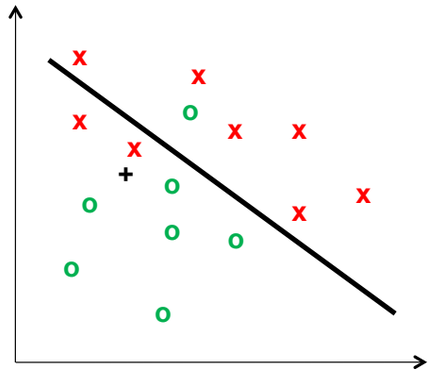
The first term is a *regularizer* (prevents overfitting and makes optimization easier)

Alternate visualization: heightfeild





Unified notation: regularized loss minimization



want this to always be large

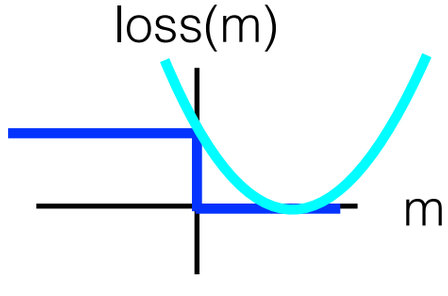
$$\min_w \lambda R(w) + \sum_i loss(y_i f_w(x_i))$$

$$R(w) = \frac{1}{2}||w||^2$$

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

$$loss_{squared}(m) = (1-m)^2$$

$$loss_{01}(m) = I(m < 0)$$



Birds-eye view of ML

$$\min_w \lambda R(w) + \sum_i loss(y_i f_w(x_i))$$

$$R_{L2}(w) = \frac{1}{2}||w||^2 \quad \text{(L2 regularization)}$$

$$R_{L1}(w) = \sum_j |w_j| \quad \text{(L1/sparse regularization)}$$

$$loss_{squared}(m) = (1-m)^2 \quad \text{(Linear regression)}$$

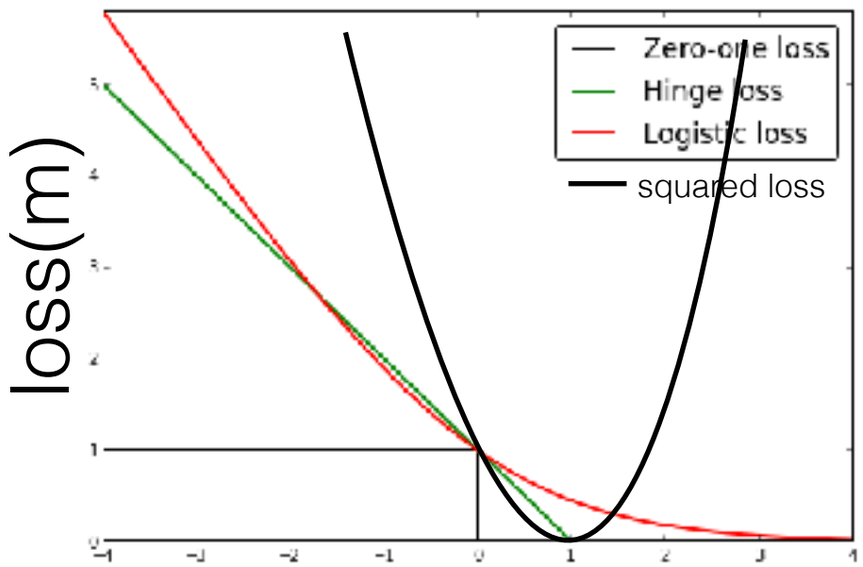
$$loss_{log}(m) = \log(1+e^{-m}) \quad \text{(Logistic regression)}$$

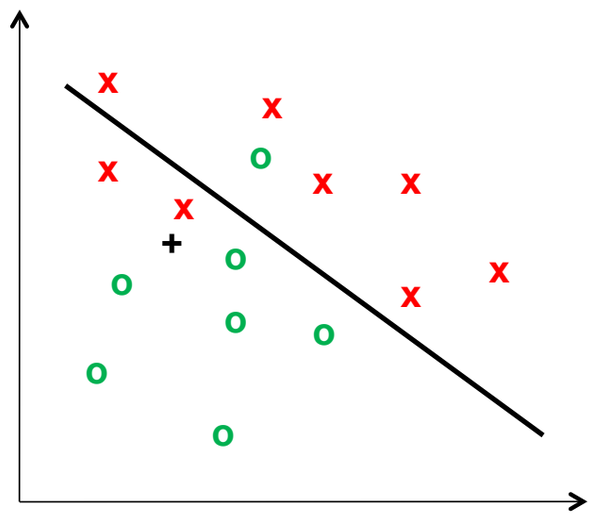
$$loss_{hinge}(m) = \max(0, 1+m) \quad \text{(Support vector machine)}$$

$$f_w(x_i) = w^T x_i \quad \text{Linear classifier}$$

$$f_w(x_i) = CNN_w(x_i) \quad \text{Nonlinear classifier}$$

Learning with losses





$$loss_{01}(m) = I(m < 0)$$

$$loss_{squared}(m) = (1 - m)^2$$

$$loss_{log}(m) = \log(1 + e^{-m})$$

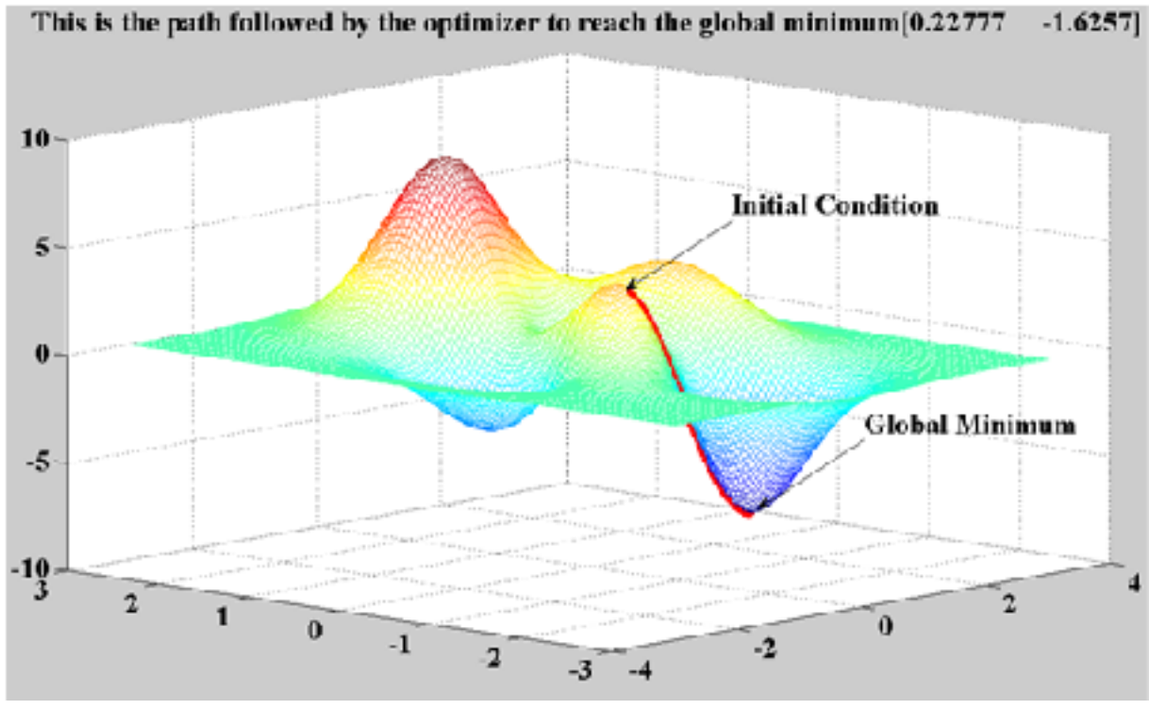
$$loss_{hinge}(m) = \max(0, 1 + m)$$

Bottom line: other than 01 and squared loss, most loss functions look similar (to me)

Learning with gradient descent

$$\min_w L(w) \quad \text{where} \quad L(w) = \frac{1}{2}||w||^2 + \sum_i (f_w(x_i) - y_i)^2$$

$$w := w - step * (w + \sum_i (f_w(x_i) - y_i) \frac{\partial f_w(x_i)}{\partial w})$$

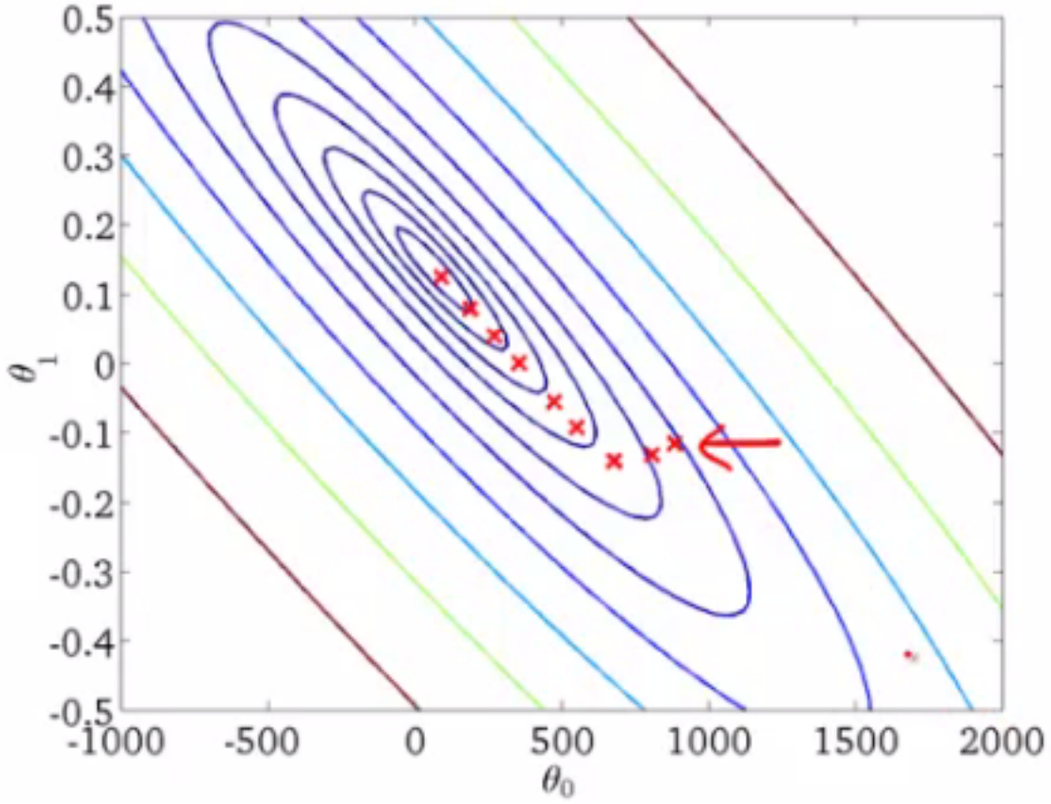


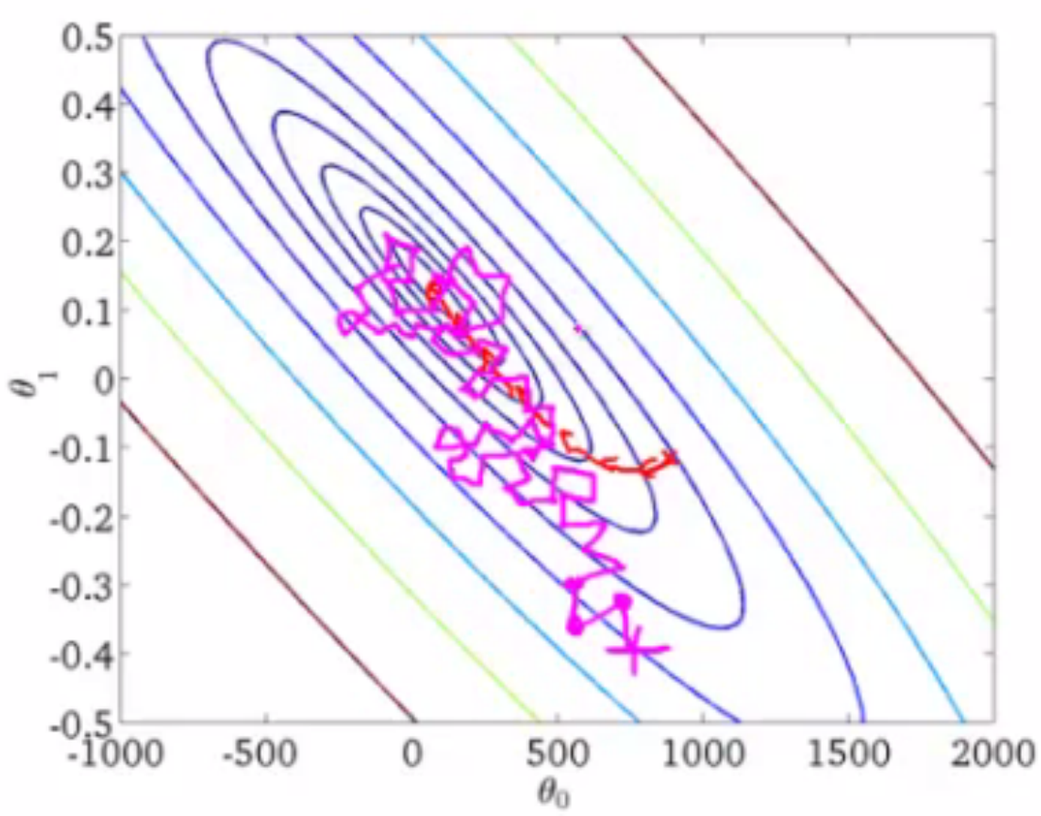
Early 2000's: obsession with convex L(w)

Large-scale learning: stochastic gradient descent

$$\min_w \frac{1}{2}||w||^2 + \sum_i (f_w(x_i) - y_i)^2$$

$$w := w - step * \left(w + (f_w(x_i) - y_i)\frac{\partial f_w(x_i)}{\partial w}\right)$$

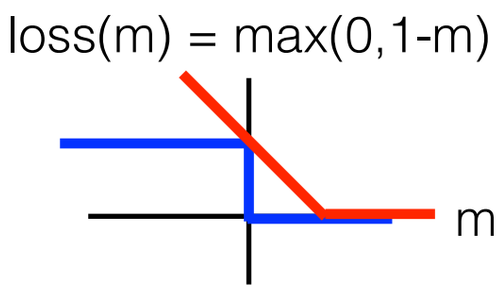


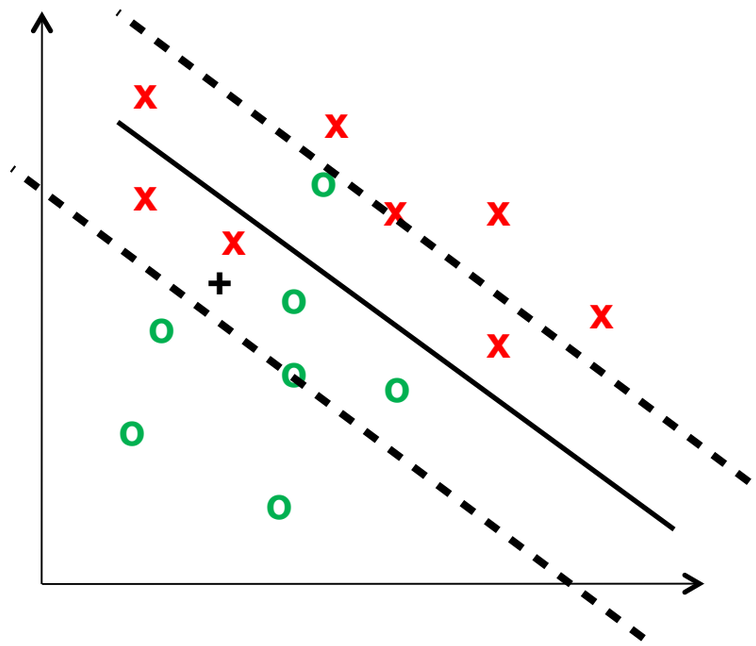


Trivially “out-of-core”: most contemporary models are trained in this manner

Special case: hinge loss (SVMs)

$$\min_w L(w) \quad \text{where} \quad L(w) = \frac{\lambda}{2}||w||^2 + \sum_i \max(0, 1 - y_i w \cdot x_i)$$



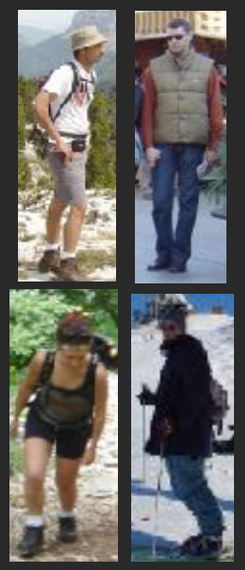


Positive examples that score better than 1 and negative examples that score less than 1 do not affect the objective function

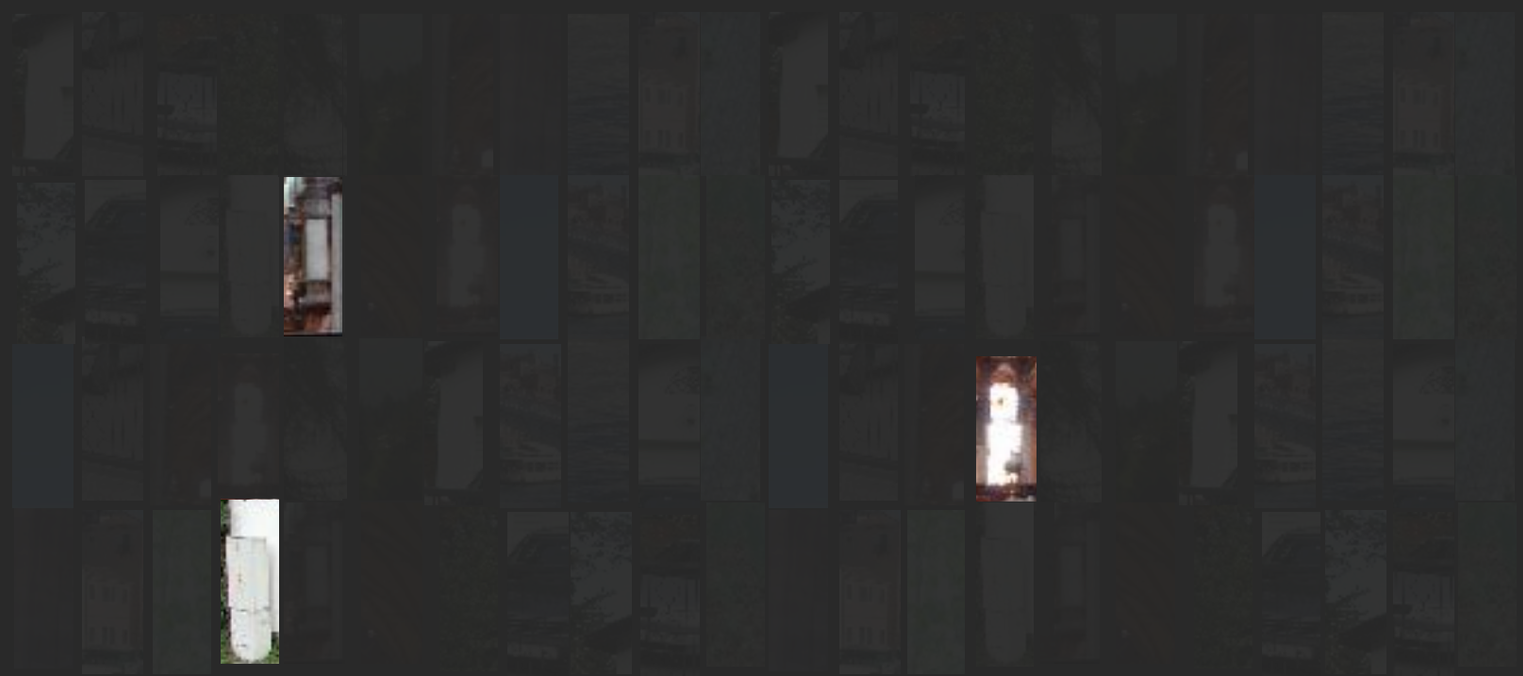
Implication: we can through them away without changing the solution

Large-scale learning

pos



neg



Our test set distribution is highly imbalanced; so should be the training set

(hundreds of positives, hundreds of millions of negatives)

Support vector machines (SVMs) are attractive because they generate sparse learning problems

(turn continuous search over parameters into combinatorial search over data)

Large-scale learning for SVMs

Lots of large-scale solvers for quadratic programs (SVMS)

Two flavors

Batch: Require access to all training data
(guarantees on convergence)

Online: Require access to on-the-fly training data
(usually stochastic in practice)

In-between: Support-vectors fit in memory, but data doesn’t
(Relatively unexplored!)

Large-scale learning for SVMs

In practice, can get near-optimal models with a single pass of "dual coordinate descent" through large datasets

A. Bordes, L. Bottou, P. Gallinari, and J. Weston. Solving multiclass support vector machines with LaRank. In *ICML*, pages 89–96. ACM, 2007. 7

A. Bordes, S. Ertekin, J. Weston, and L. Bottou. Fast kernel classifiers with online and active learning. *The Journal of Machine Learning Research*, 6:1579–1619, 2005. 8

L. Bottou and O. Bousquet. The tradeoffs of large scale learning. *Advances in neural information processing systems*, 20:161–168, 2008. 1

http://www.csie.ntu.edu.tw/~cjlin/liblinear/

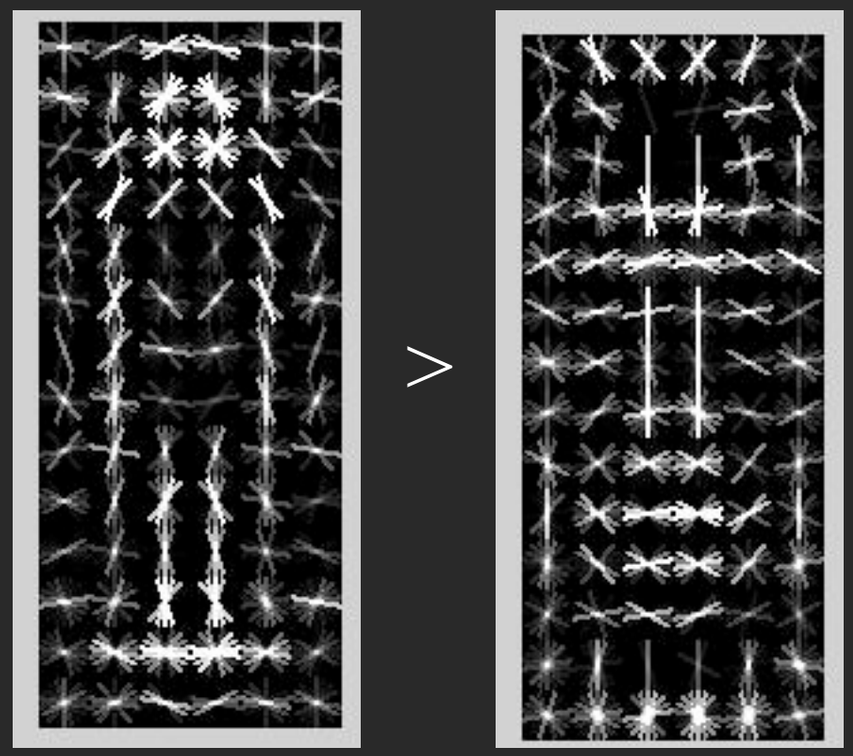
How to interpret positive and negative weights?

$$w \cdot x > 0$$

$$(w_{pos} - w_{neg}) \cdot x > 0$$

$$w_{pos} \cdot x > w_{neg} \cdot x$$

Pedestrian template



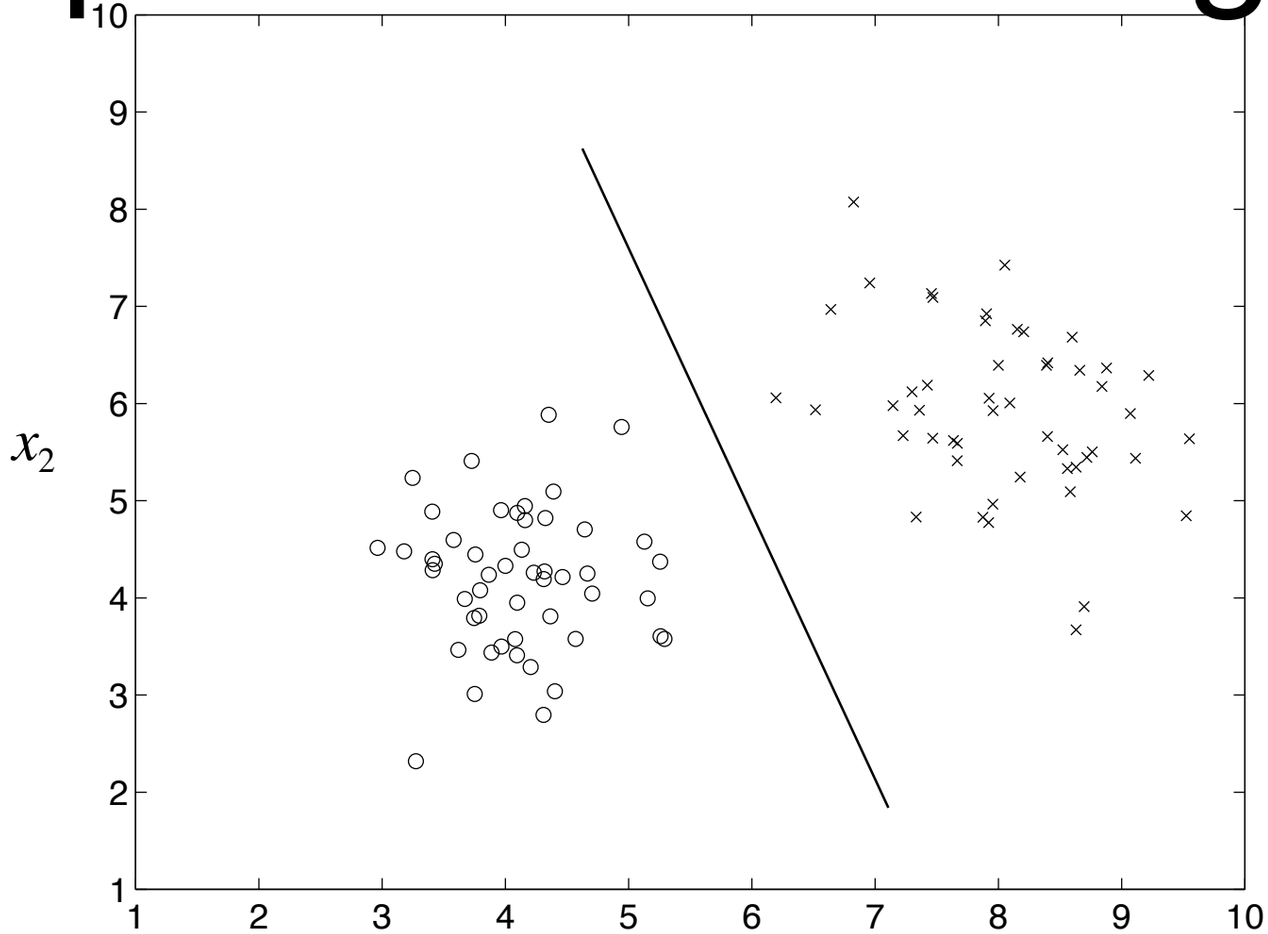
Pedestrian background template

Right approach is to compete pedestrian, pillar, doorway... models

Background class is hard to model - easier to penalize particular vertical edges

Historically, model-based approaches tend not to model negative set

But what if we actually try to model positives and negatives?



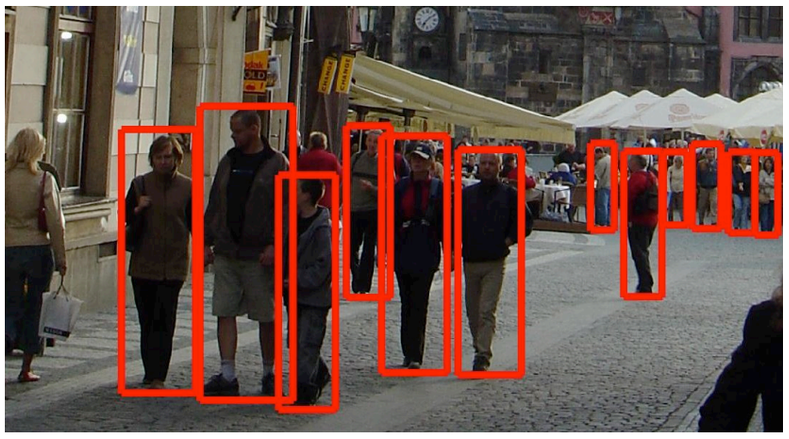
Simple generative model P(x,y) = P(y) P(x|y):

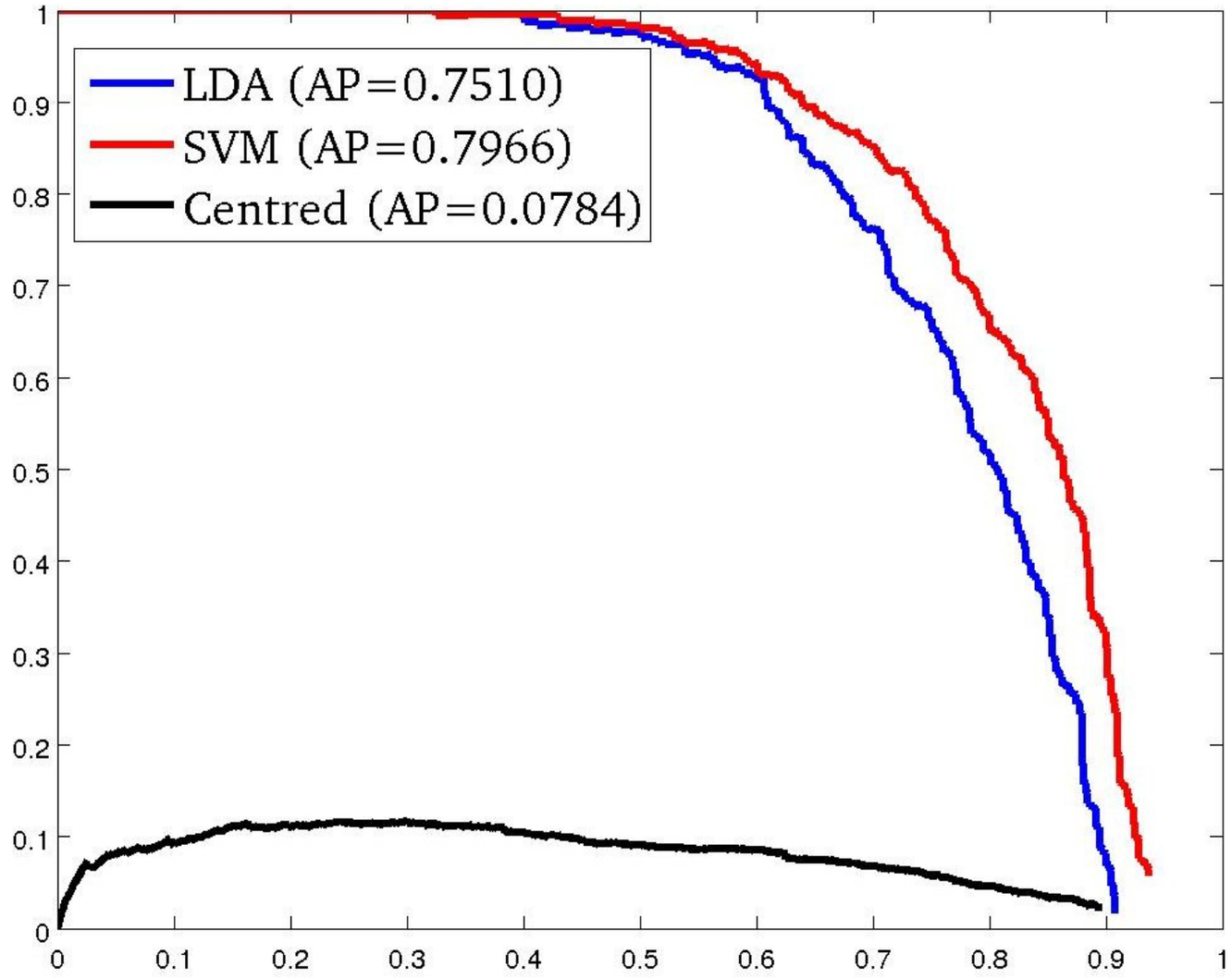
$$y \in \{-1, 1\}$$

$$x \in R^N$$

1. P(y): Flip (biased) coin to select class ‘k’
2. P(x|y): Sample from Gaussian (mu_k, Sigma_k)

A look ahead





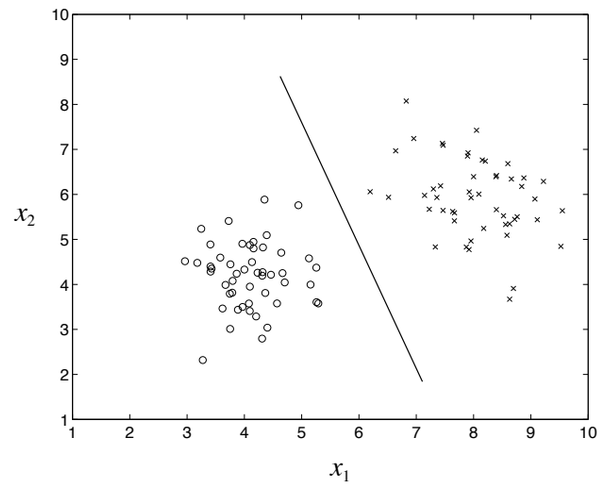
(a) AP

Simple linear discriminant analysis gets us 90% of the way there…

Class-conditional Gaussians

If we assume all classes have same Sigma….

(For notational ease, let's assume priors for class y in {0,1} are equal)



$$p(y=1|x) = \frac{p(x|y=1)p(y=1)}{p(x|y=1)p(y=1) + p(x|y=0)p(y=0)}$$

Plug in the following and simplify:

$$p(y=1) = .5$$

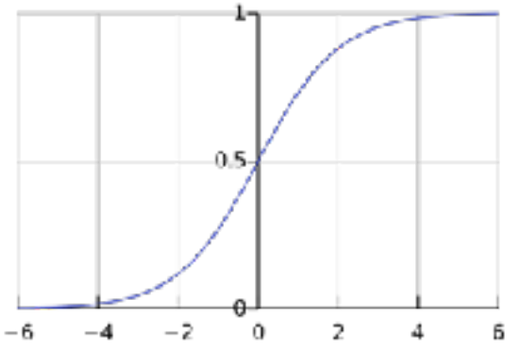
$$p(x|y) = \frac{1}{\sqrt{(2\pi)^d|\Sigma|}} e^{-\frac{1}{2}(x-\mu_y)^T\Sigma^{-1}(x-\mu_y)}$$

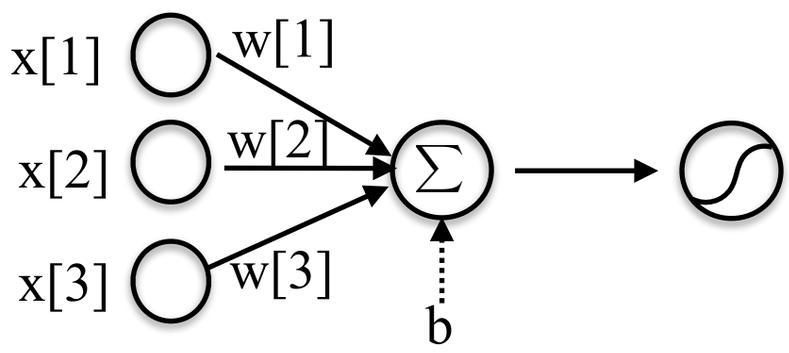
Class-conditional Gaussians

If we assume all classes have same Sigma….

$$p(y=1|x) = sigmoid(w \cdot x + b)$$

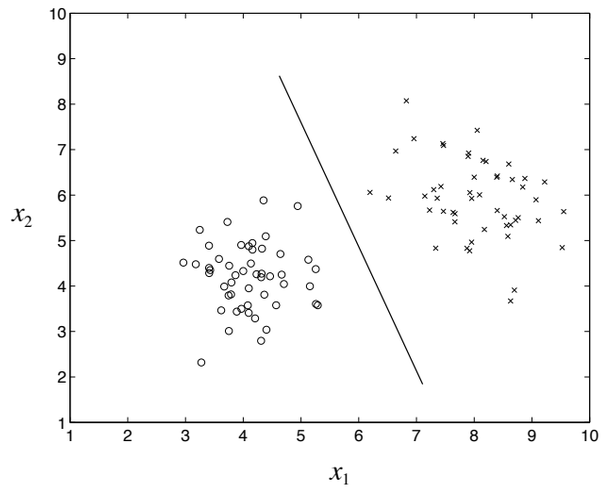
$$sigmoid(x) = \frac{1}{1+e^{-x}}$$

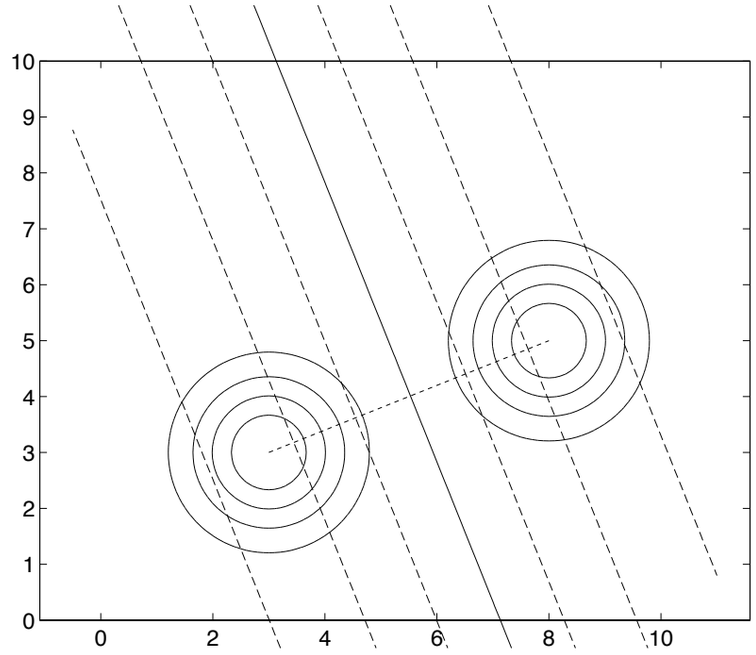




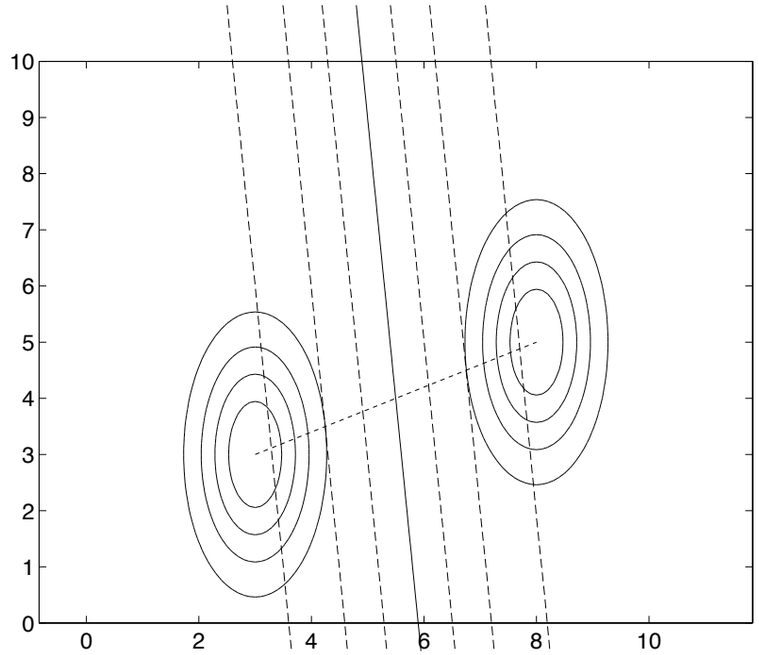
$$p(y=1|x) > .5 \quad \text{when} \quad f_w(x) = w \cdot x + b > 0, \quad w = \Sigma^{-1}(\mu_1 - \mu_0)$$

Class-conditional Gaussians





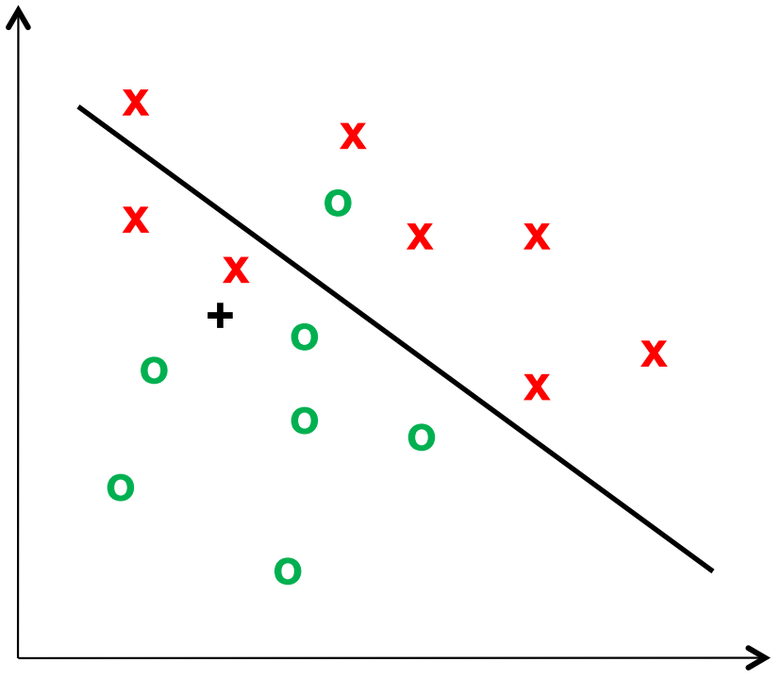
(a)

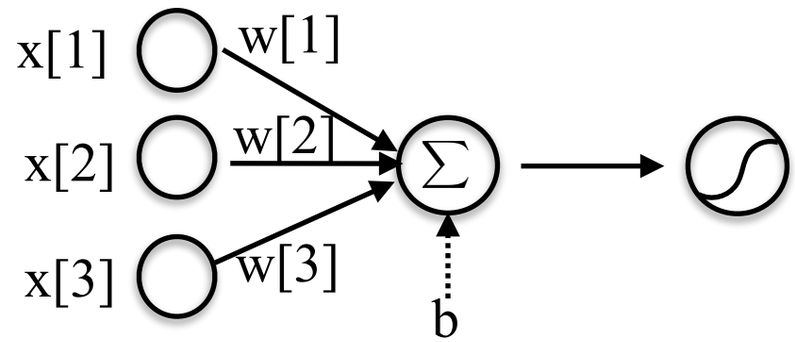


(b)

What happens if Sigmas for the two classes are different?

Alternative: discriminative fitting (logistic regression)





$$p(y = 1|x) = sigmoid(w \cdot x + b)$$

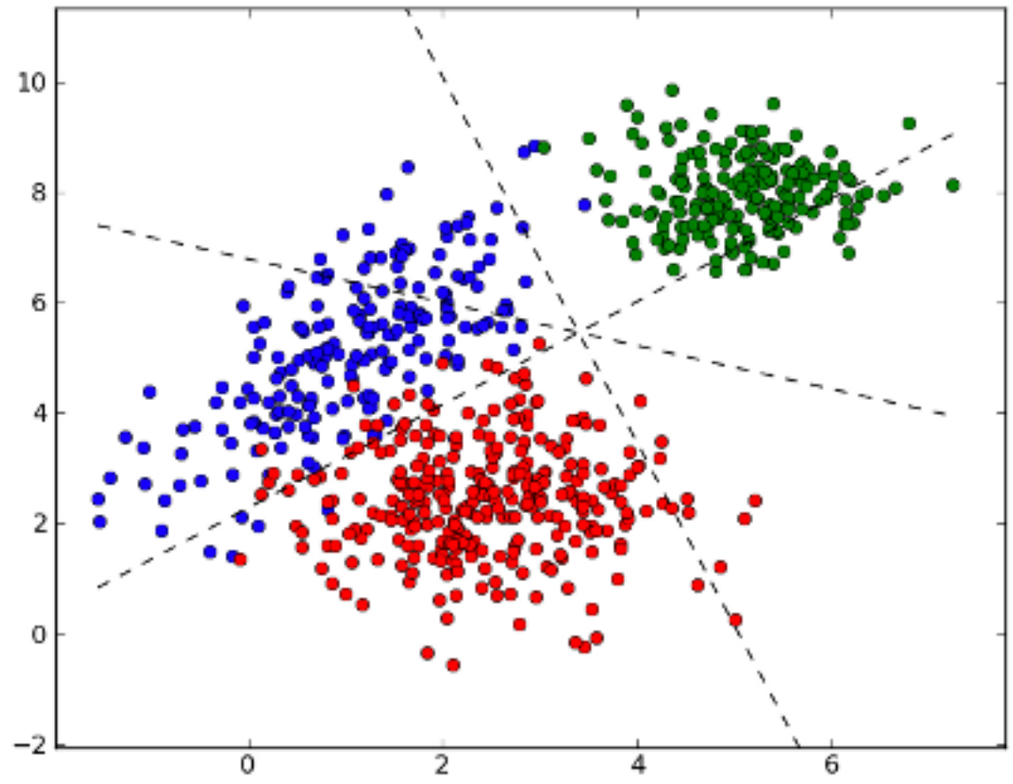
Given data $\{(x_i,y_i)\}$, directly fit parameters ‘w,b’ to maximize

$$\sum_i \log p(y_i|x_i)$$

This is sometimes called cross-entropy minimization, and is equivalent to miniming with log-loss!

Extension to multiple classes

Same derivation of class-conditional Gaussians for K classes (assuming all have same covariance)

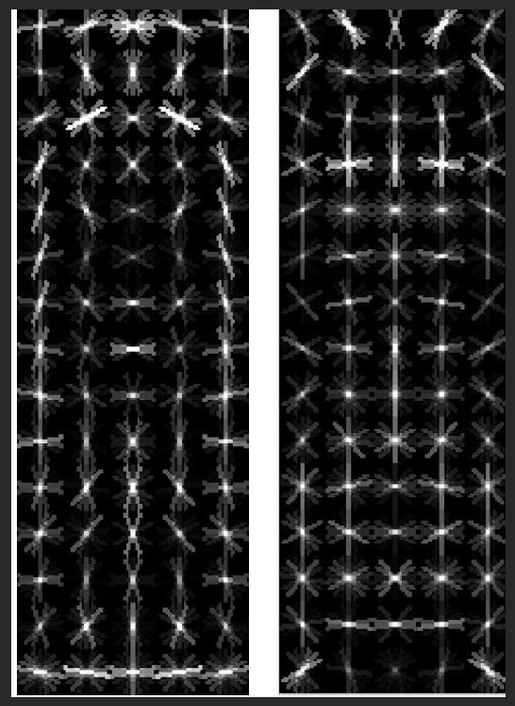


$$p(y = k|x) = \frac{e^{w_k \cdot x}}{\sum_i e^{w_i \cdot x}}$$

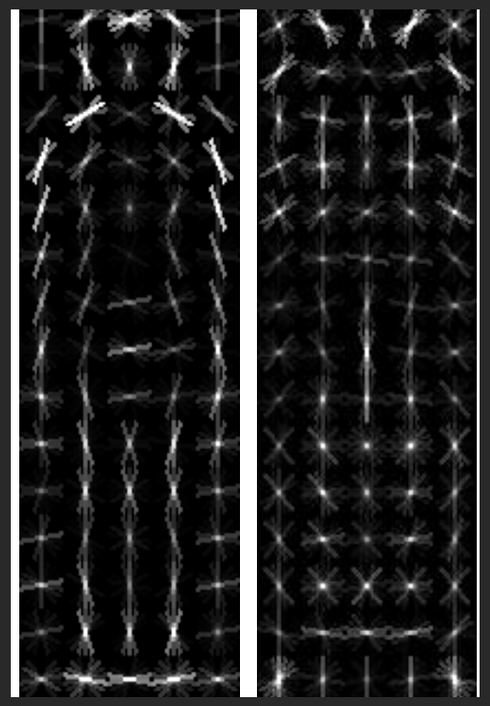
Known as the *softmax* function

When parameters are fit to maximize log p(y|x), known as softmax cross-entropy minimmization

Let’s build up intuition with generative model

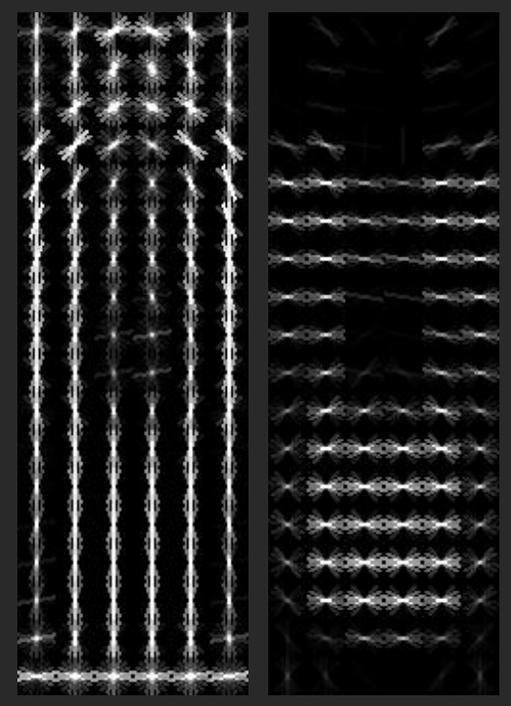


SVM



Gaussian model

$$w = \Sigma^{-1}(\mu_1 - \mu_0)$$



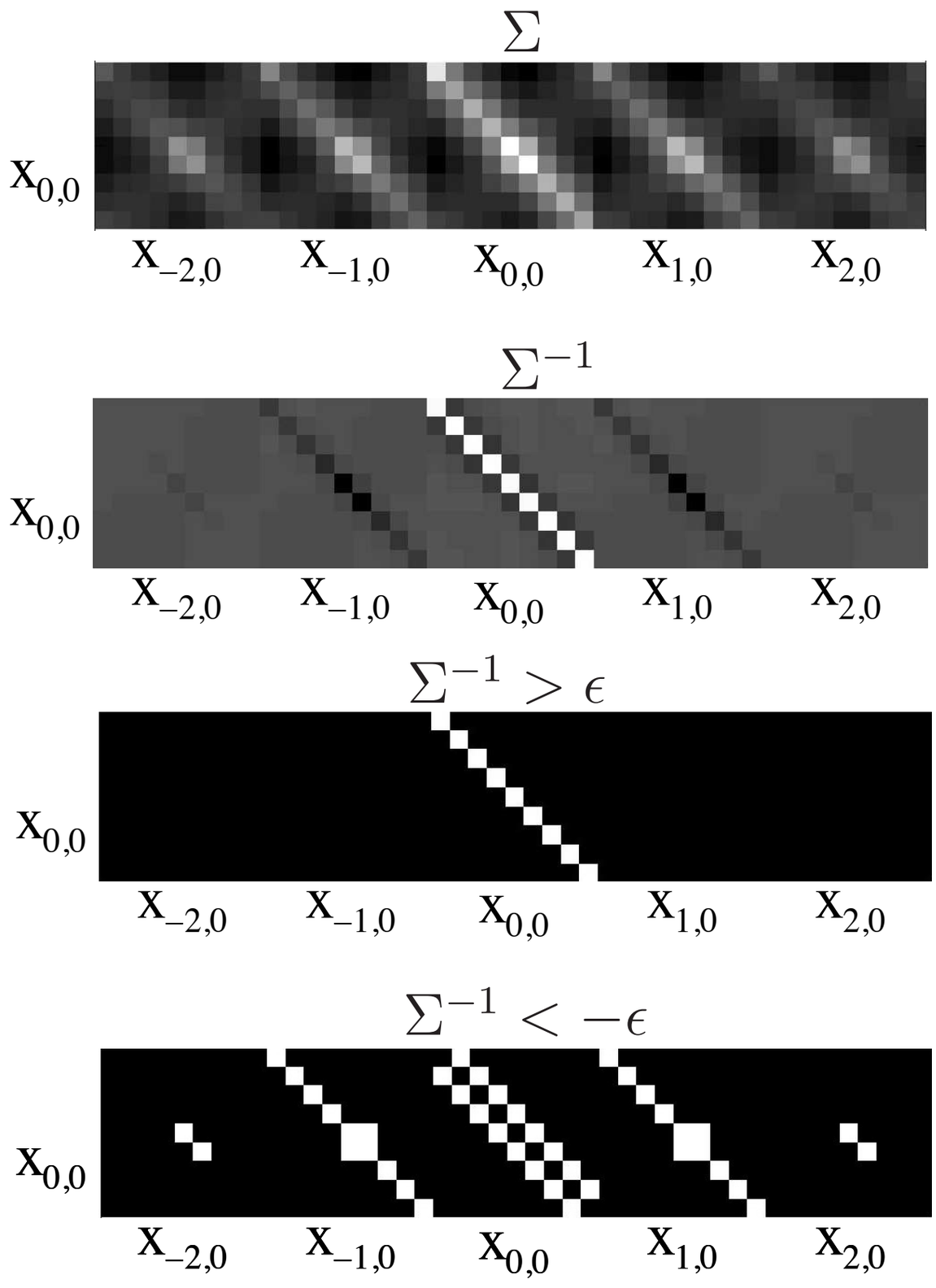
Centered model

$$w = \mu_1 - \mu_0$$

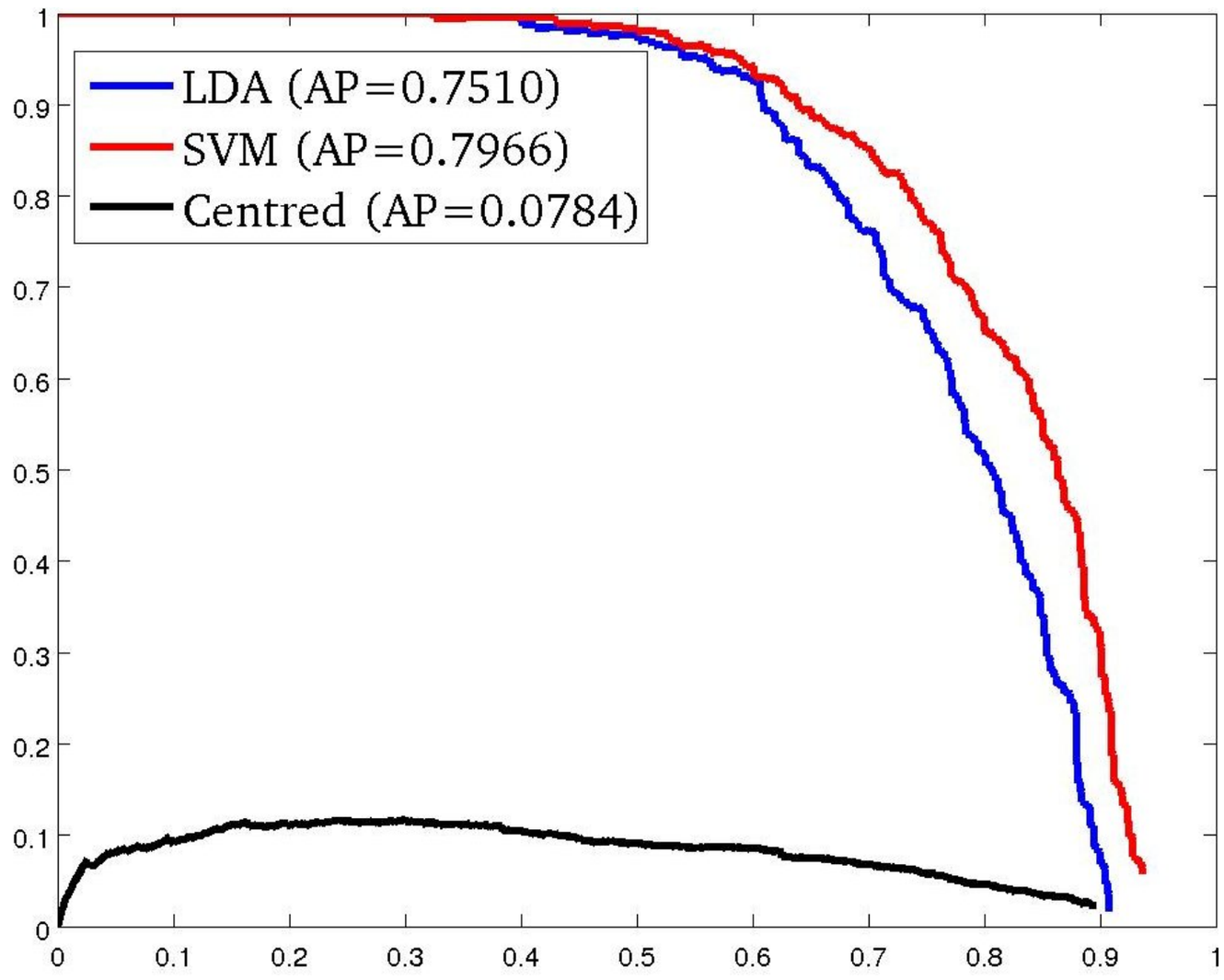
Learn templates with generic (de)correlation model

Hariharan, Malik, Ramanan ECCV 12

What's the covariance matrix capturing?



The numbers



(a) AP

Simple gaussian model gets us 90% of the way there…

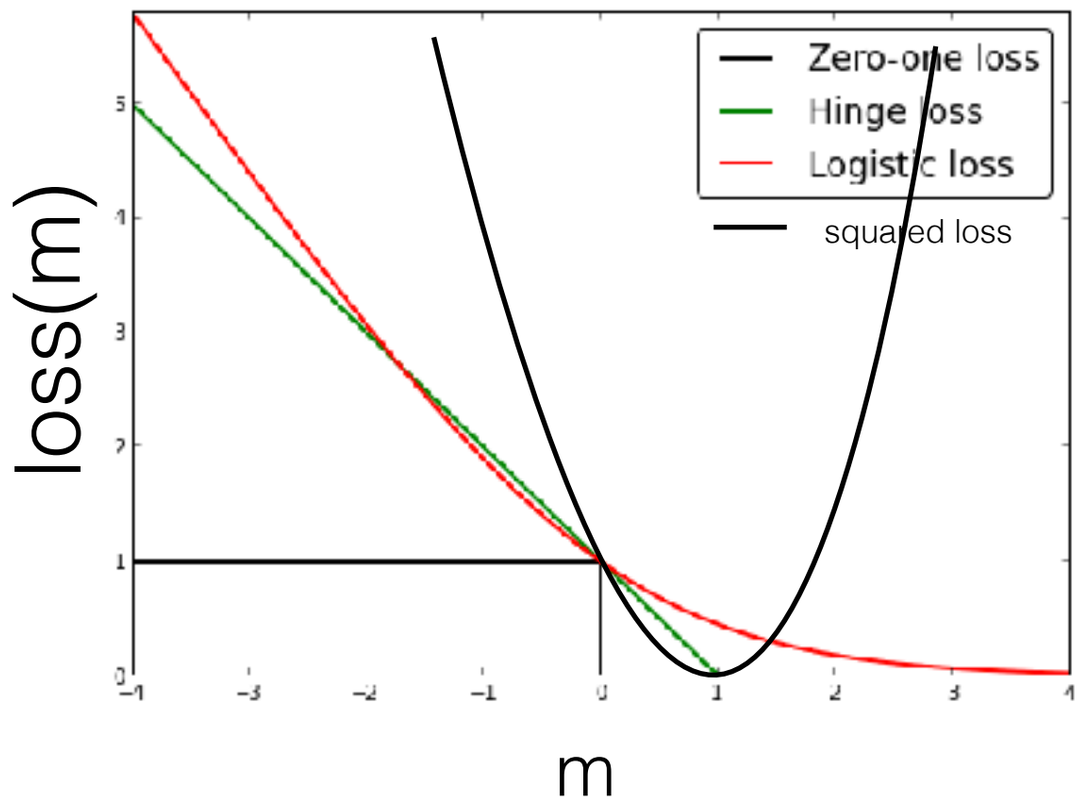
Parting thoughts on class-conditional Gaussians

1. Also known as Linear Discriminant Analysis (LDA) or Fischer Discriminant Analysis (FDA) when derived using other criteria (maximizing ratio of between to within-class variances)

2. One can also obtain the LDA / FDA solution by discriminative learning with a squared error loss

(Hastie et al, Elements of Statistical Learning)

Implies the distinction between generative and discriminative models can be blurred…



Parting thoughts on statistical classification

- Loss functions: hinge, log-loss, squared loss
- SVMs generate sparse optimization problems
- Generative models are promising, but current state-of-the-art relies on discrimative loss minimization

Things that will appear later:

Cross-entropy loss: minimizing cross-entropy of binary prediction is equivalent to log-loss

Soft-max loss: minimizing cross-entropy of K-way prediction is equivalent to soft-max loss

Outline

- Motivation
- How to define problem?
- Case example: finding people
- Statistical classification