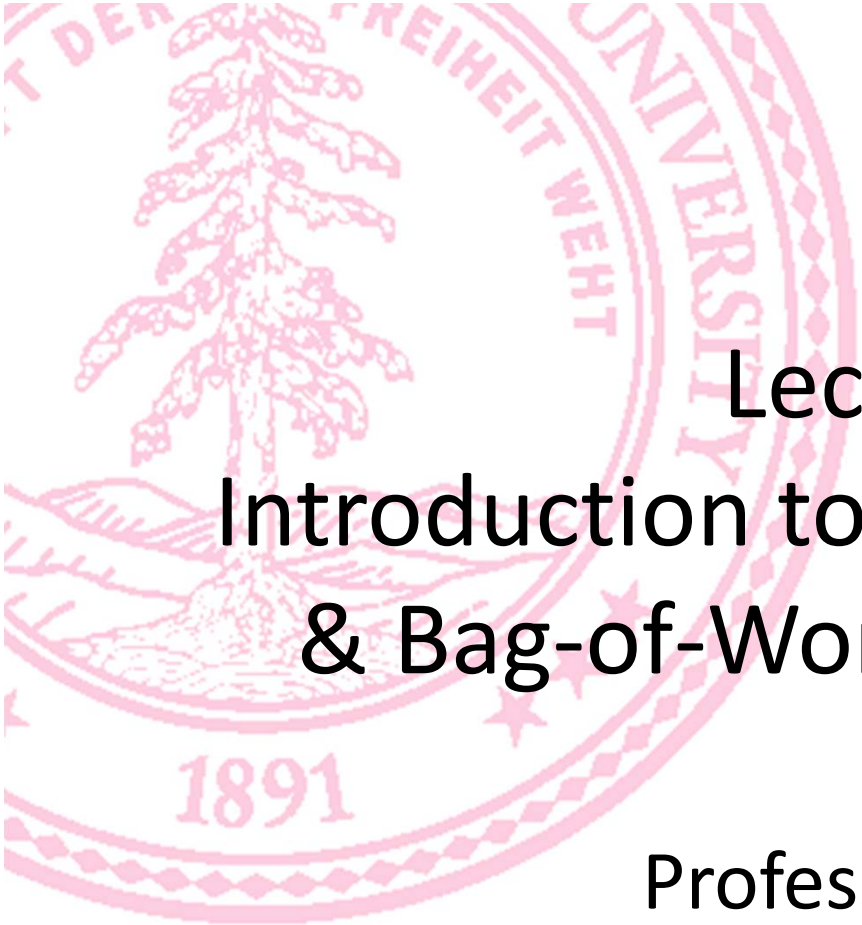


# Lecture 14:
# Introduction to Object Recognition & Bag-of-Words (BoW) Models

Professor Fei-Fei Li

Stanford Vision Lab

# What we will learn today?

- Introduction to object recognition
  - Representation
  - Learning
  - Recognition
- Bag of Words models (Problem Set 4 (Q2))
  - Basic representation
  - Different learning and recognition algorithms

# What are the different visual recognition tasks?

# Classification:

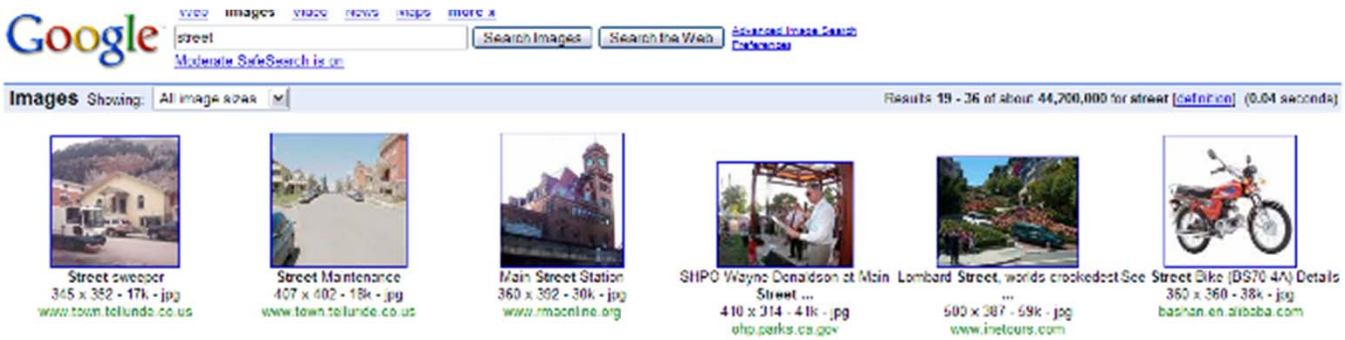
Does this image contain a building? [yes/no]

# Classification:

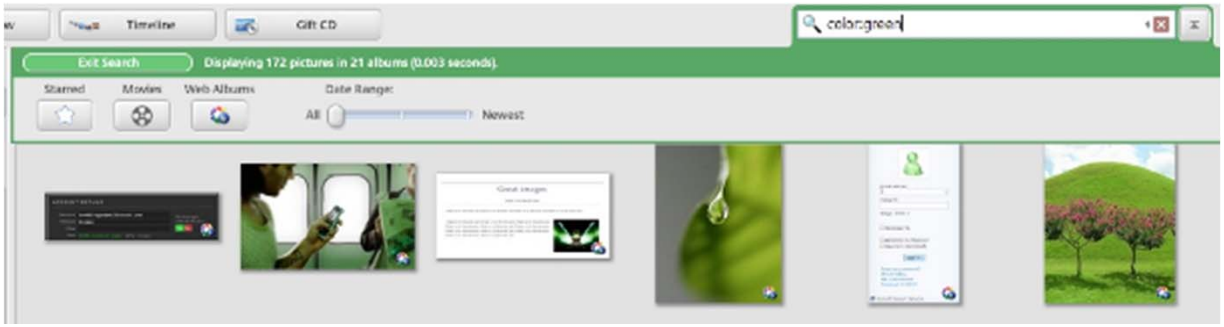
Is this an beach?

# Image Search

flickr webshots Ask Images Cydral picsearch Google Image Search altavista




# Organizing photo collections

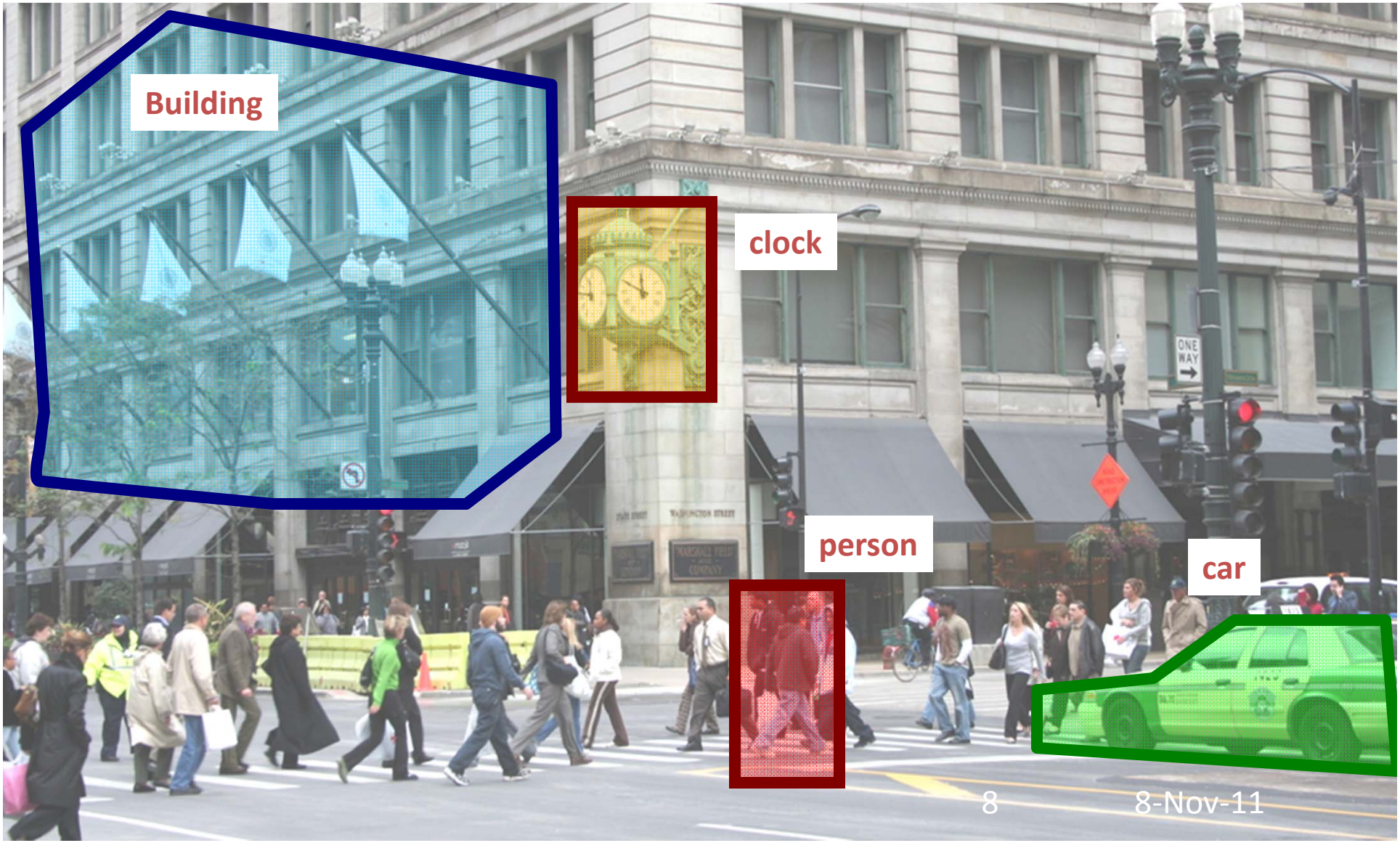


# Detection:

Does this image contain a car? [where?]

# Detection:

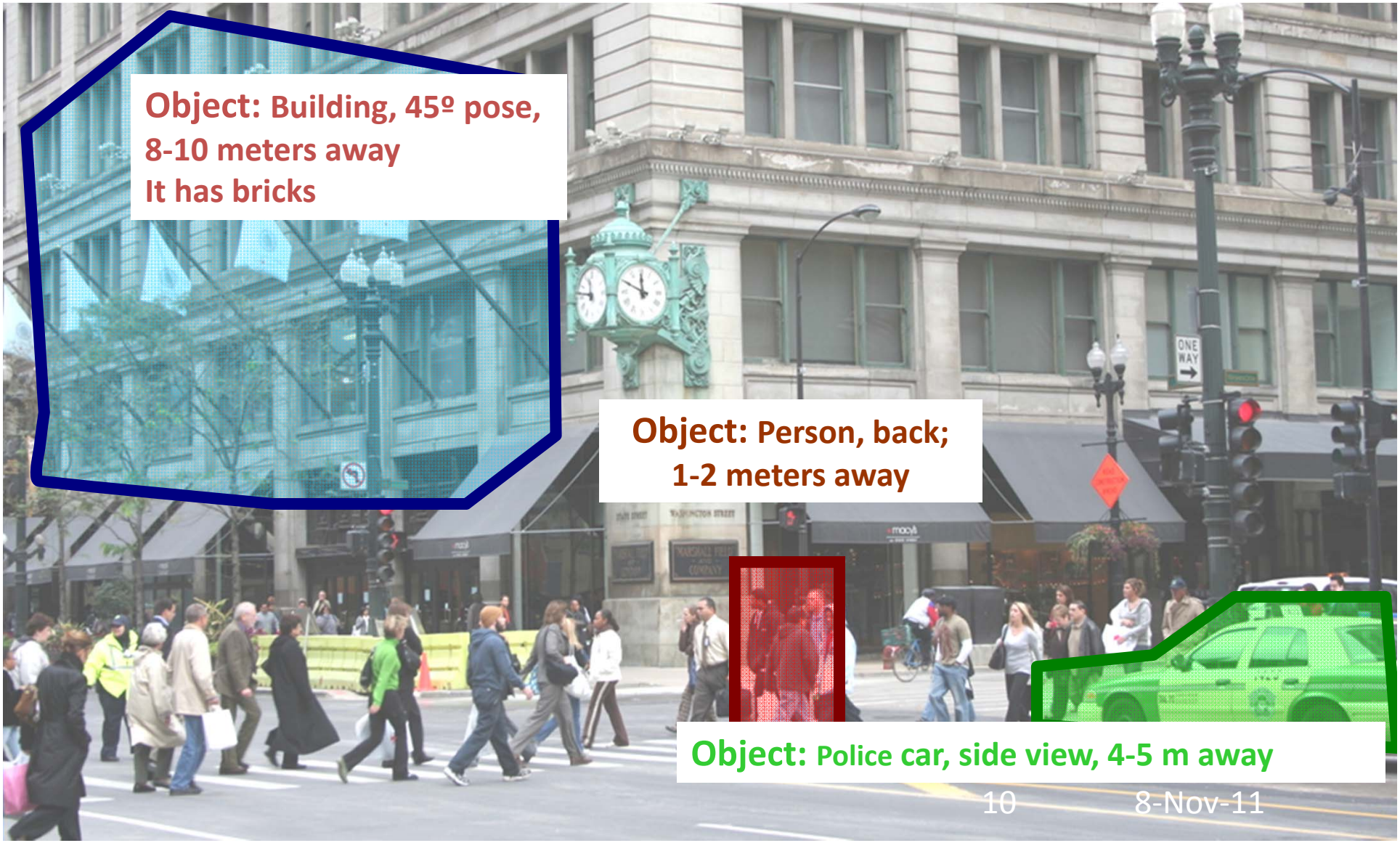
Which object does this image contain? [where?]



# Detection:

Accurate localization (segmentation)

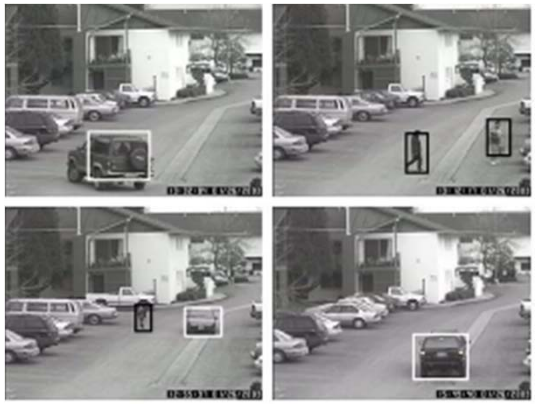
# **Detection:** Estimating object semantic & geometric attributes



# Applications of computer vision


Computational photography

Assistive technologies



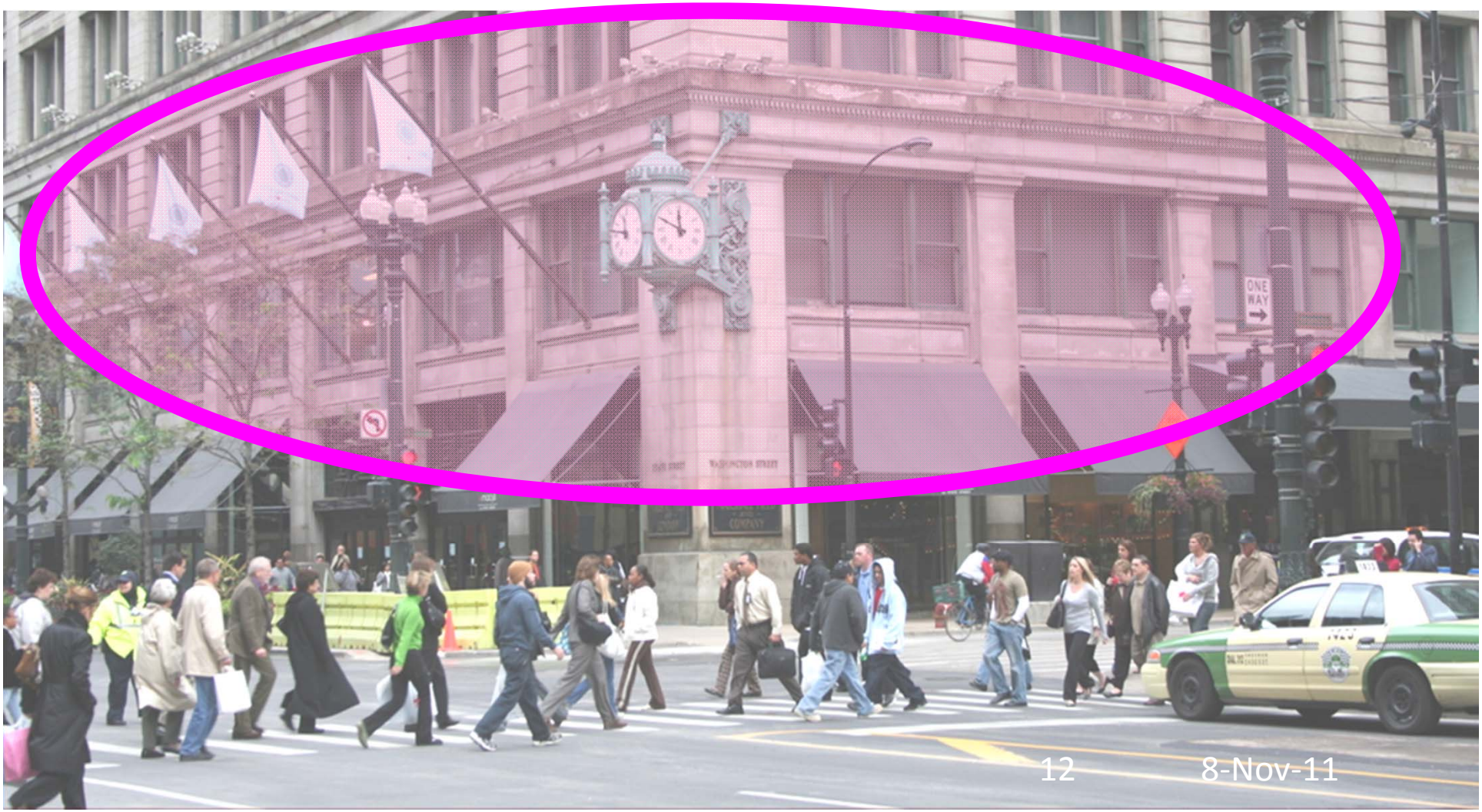
Surveillance

Security

Assistive driving

# Categorization vs Single instance recognition

Does this image contain the Chicago Macy building's?

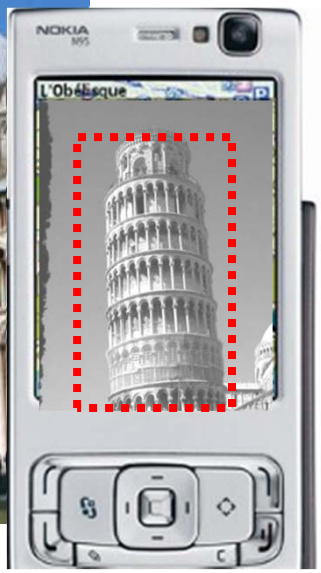


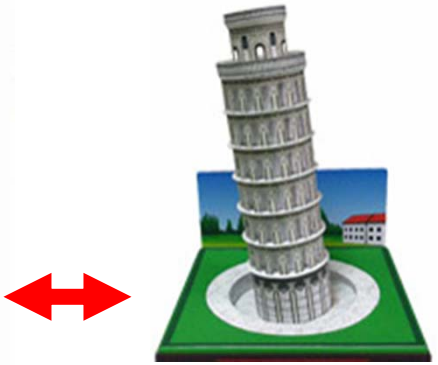
# Categorization vs Single instance recognition

Where is the crunchy nut?


# Applications of computer vision

- Recognizing landmarks in mobile platforms



+ GPS

# Activity or Event recognition

What are these people doing?

# Visual Recognition

- Design algorithms that are capable to
  - Classify images or videos
  - Detect and localize objects
  - Estimate semantic and geometrical attributes
  - Classify human activities and events

## Why is this challenging?

# How many object categories are there?

# Challenges: viewpoint variation

Michelangelo 1475-1564

# Challenges: illumination

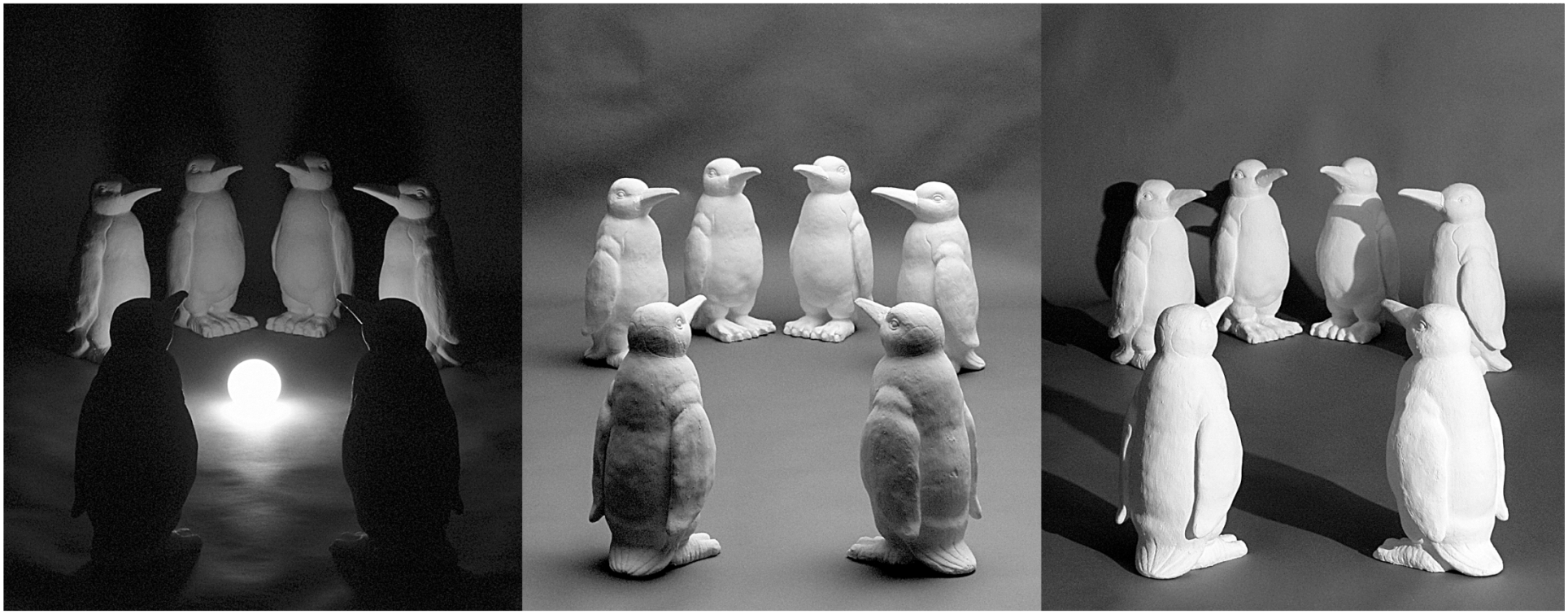


image credit: J. Koenderink

# Challenges: scale

# Challenges: deformation

# Challenges: occlusion

Magritte, 1957

# Challenges: background clutter

Kilmeny Niland. 1995

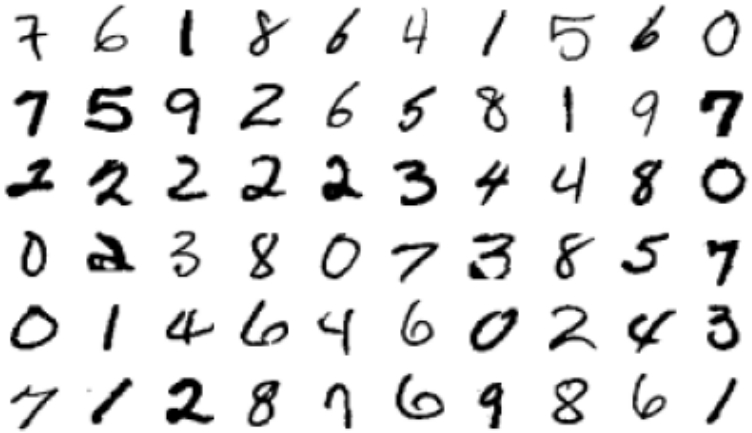
# Challenges: intra-class variation






# Some early works on object categorization

- Turk and Pentland, 1991
- Belhumeur, Hespanha, & Kriegman, 1997
- Schneiderman & Kanade 2004
- Viola and Jones, 2000



- Amit and Geman, 1999
- LeCun et al. 1998
- Belongie and Malik, 2002

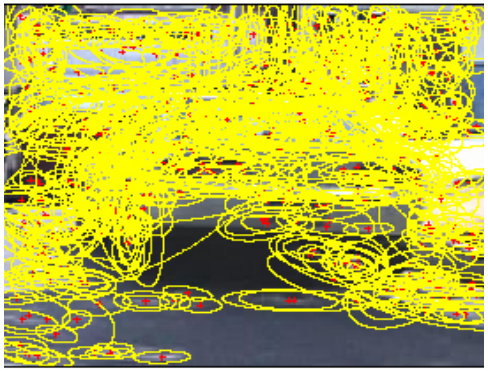
- Schneiderman & Kanade, 2004
- Argawal and Roth, 2002
- Poggio et al. 1993

# Basic issues

- Representation
  - How to represent an object category; which classification scheme?
- Learning
  - How to learn the classifier, given training data
- Recognition
  - How the classifier is to be used on novel data

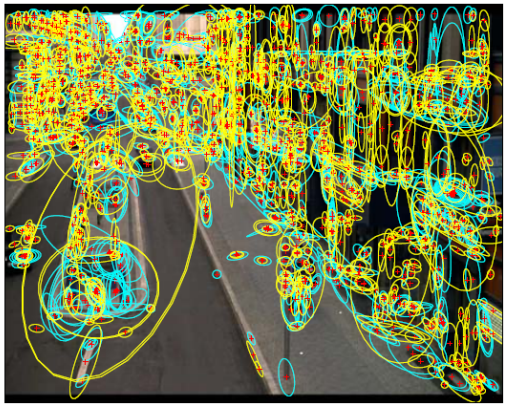
# Representation

- Building blocks: Sampling strategies

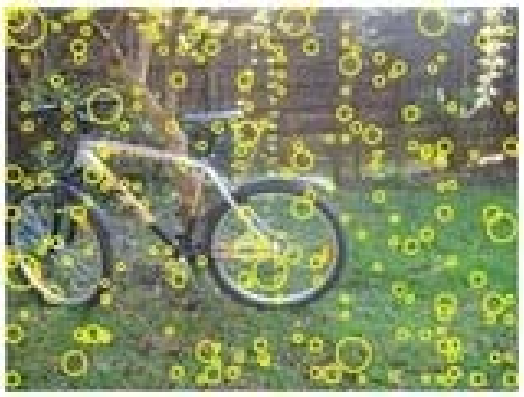


Interest operators

Dense, uniformly



Multiple interest operators

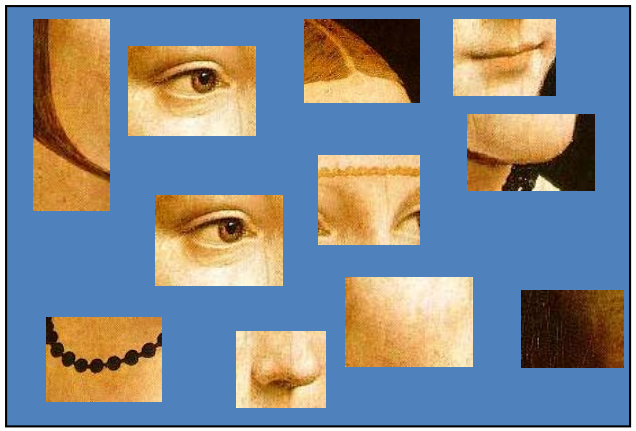


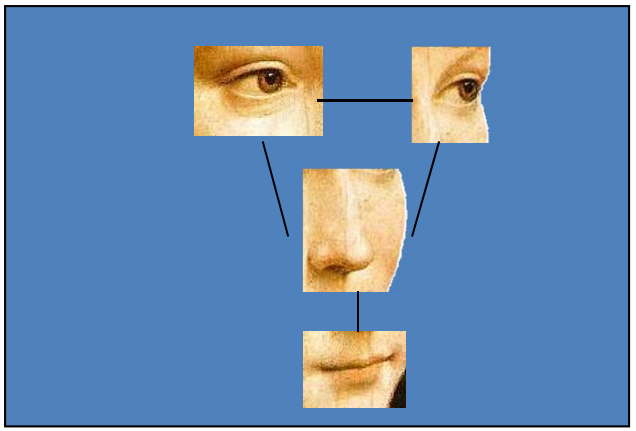
Randomly

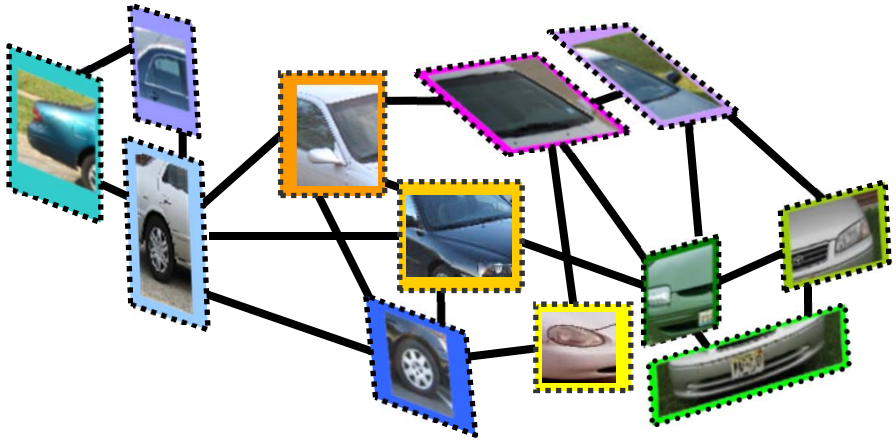
Image credits: L. Fei-Fei, E. Nowak, J. Sivic

# Representation

– Appearance only or location and appearance

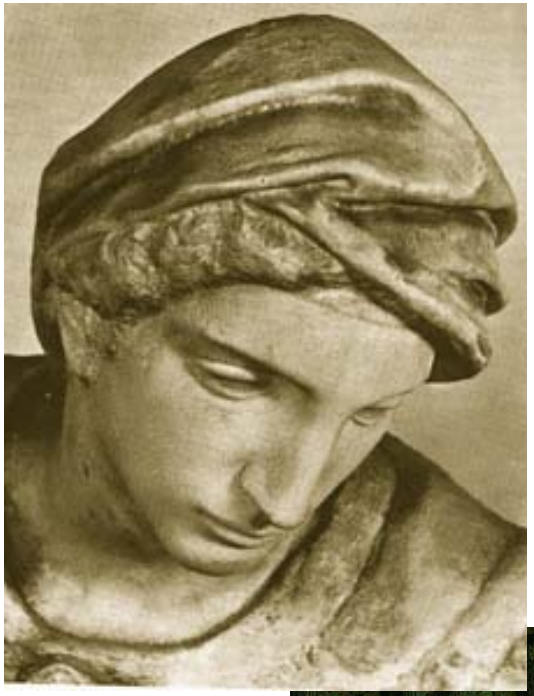


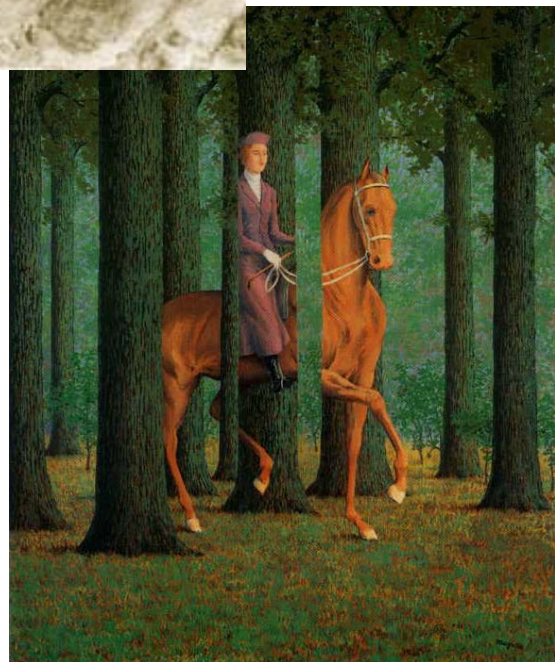




# Representation

- Invariances
  - View point
  - Illumination
  - Occlusion
  - Scale
  - Deformation
  - Clutter
  - etc.





# Representation

- To handle intra-class variability, it is convenient to describe an object categories using probabilistic models
- Object models: Generative vs Discriminative vs hybrid

# Object categorization: the statistical viewpoint

$$p(zebra \mid image)$$

vs.

$$p(no\ zebra \mid image)$$

- Bayes rule: $P(A|B) = \frac{P(B|A)\,P(A)}{P(B)}$

$$\frac{p(zebra \mid image)}{p(no\ zebra \mid image)}$$

# Object categorization: the statistical viewpoint

$$p(zebra \mid image)$$

vs.

$$p(no\ zebra \mid image)$$

- Bayes rule: $P(A|B) = \frac{P(B|A)\,P(A)}{P(B)}$

$$\underbrace{\frac{p(zebra \mid image)}{p(no\ zebra \mid image)}}_{\text{posterior ratio}} = \underbrace{\frac{p(image \mid zebra)}{p(image \mid no\ zebra)}}_{\text{likelihood ratio}} \cdot \underbrace{\frac{p(zebra)}{p(no\ zebra)}}_{\text{prior ratio}}$$

# Object categorization: the statistical viewpoint

- **Discriminative methods model posterior**
- **Generative methods model likelihood and prior**

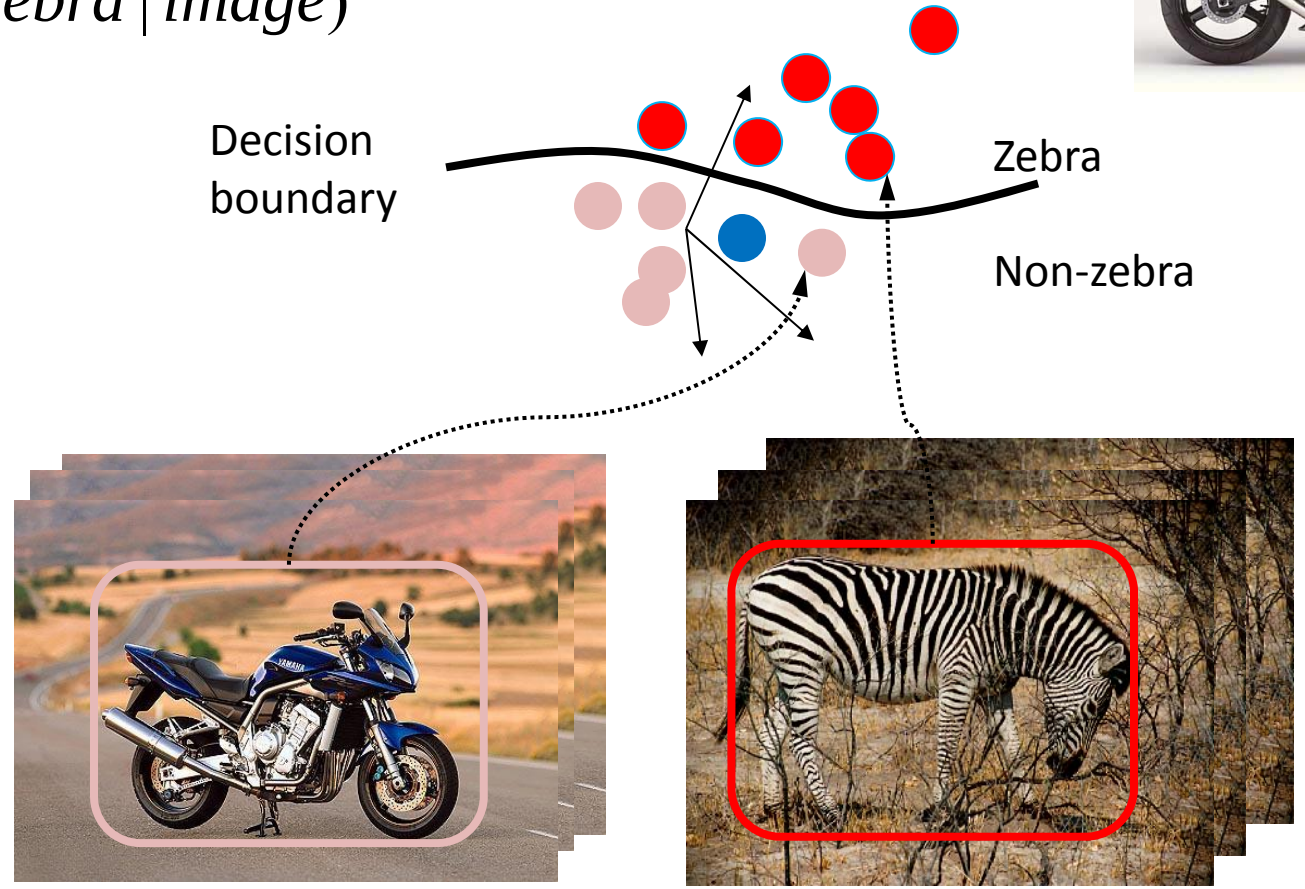
- Bayes rule:

$$\underbrace{\frac{p(zebra \mid image)}{p(no\ zebra \mid image)}}_{\text{posterior ratio}} = \underbrace{\frac{p(image \mid zebra)}{p(image \mid no\ zebra)}}_{\text{likelihood ratio}} \cdot \underbrace{\frac{p(zebra)}{p(no\ zebra)}}_{\text{prior ratio}}$$

# Discriminative models

- Modeling the posterior ratio:

$$\frac{p(zebra \mid image)}{p(no\ zebra \mid image)}$$



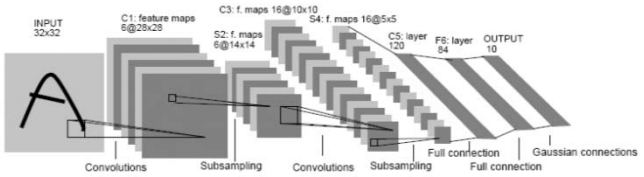
# Discriminative models

**Nearest neighbor**

$10^6$ examples

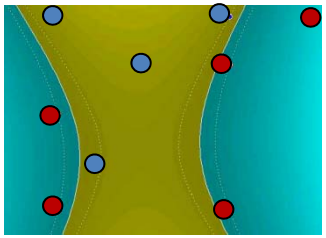
Shakhnarovich, Viola, Darrell 2003
Berg, Berg, Malik 2005...

**Neural networks**




LeCun, Bottou, Bengio, Haffner 1998
Rowley, Baluja, Kanade 1998
…

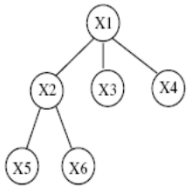
**Support Vector Machines**



Guyon, Vapnik, Heisele, Serre, Poggio…

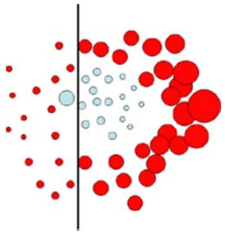
**Latent SVM**

**Structural SVM**




Felzenszwalb 00
Ramanan 03…

**Boosting**



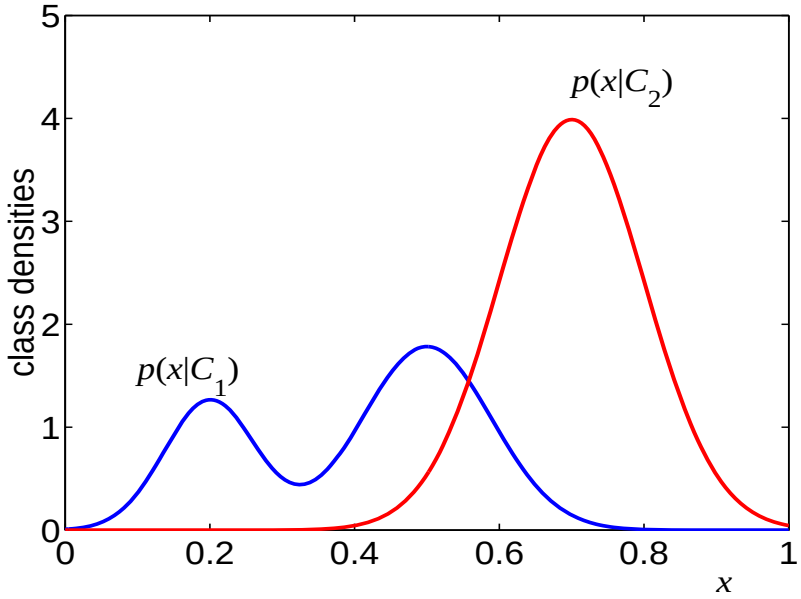
Viola, Jones 2001,
Torralba et al. 2004,
Opelt et al. 2006,…

Source: Vittorio Ferrari, Kristen Grauman, Antonio Torralba

# Generative models

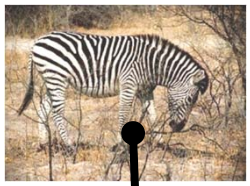
- Modeling the likelihood ratio:

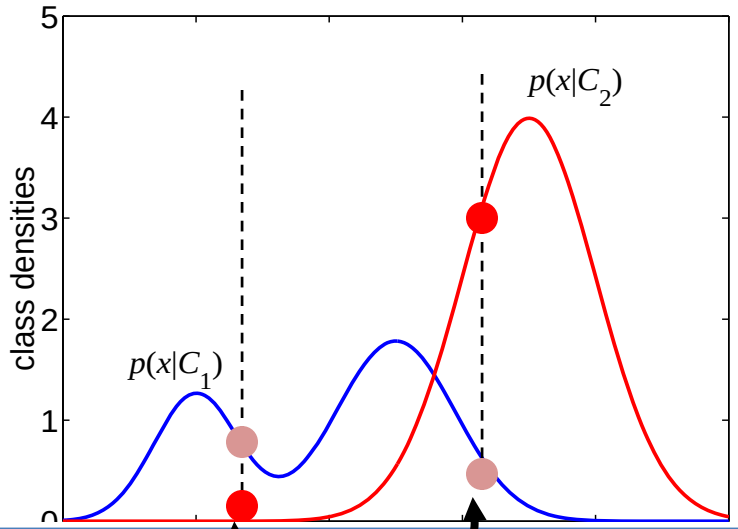
$$\frac{p(image \mid zebra)}{p(image \mid no\ zebra)}$$



# Generative models

| $p(image \mid zebra)$ | $p(image \mid no\ zebra)$ |
|---|---|
| High | Low |
| Low | High |





# Generative models

- Naïve Bayes classifier
  - Csurka Bray, Dance & Fan, 2004
- Hierarchical Bayesian topic models (e.g. pLSA and LDA)
  - Object categorization: Sivic et al. 2005, Sudderth et al. 2005
  - Natural scene categorization: Fei-Fei et al. 2005
- 2D Part based models
  - Constellation models: Weber et al 2000; Fergus et al 200
  - Star models: ISM (Leibe et al 05)
- 3D part based models:
  - multi-aspects: Sun, et al, 2009

# Basic issues

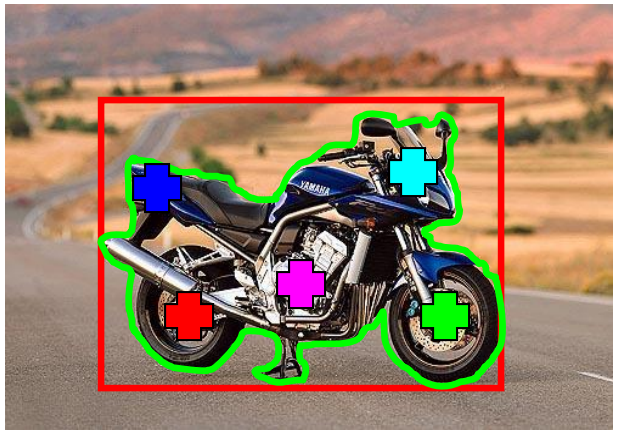
- Representation
  - How to represent an object category; which classification scheme?
- Learning
  - How to learn the classifier, given training data
- Recognition
  - How the classifier is to be used on novel data

# Learning

- Learning parameters: What are you maximizing? Likelihood (Gen.) or performances on train/validation set (Disc.)

# Learning

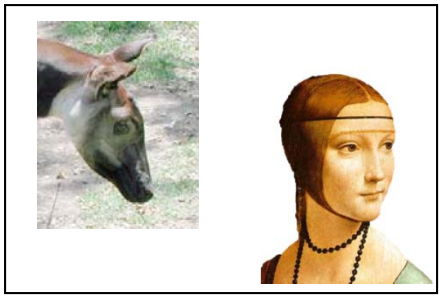
- Learning parameters: What are you maximizing? Likelihood (Gen.) or performances on train/validation set (Disc.)
- Level of supervision
  - Manual segmentation; bounding box; image labels; noisy labels
- Batch/incremental
- Priors



# Learning

- Learning parameters: What are you maximizing? Likelihood (Gen.) or performances on train/validation set (Disc.)
- Level of supervision
  - Manual segmentation; bounding box; image labels; noisy labels

- Batch/incremental
- Priors
- Training images:
  - Issue of overfitting
  - Negative images for discriminative methods



# Basic issues

- Representation
  - How to represent an object category; which classification scheme?
- Learning
  - How to learn the classifier, given training data
- Recognition
  - How the classifier is to be used on novel data

# Recognition

– Recognition task: classification, detection, etc..

# Recognition

– Recognition task

– Search strategy: Sliding Windows

Viola, Jones 2001,

- Simple
- Computational complexity (x,y, S, $\theta$, N of classes)

- BSW by Lampert et al 08

- Also, Alexe, et al 10

# Recognition

– Recognition task

– Search strategy: Sliding Windows

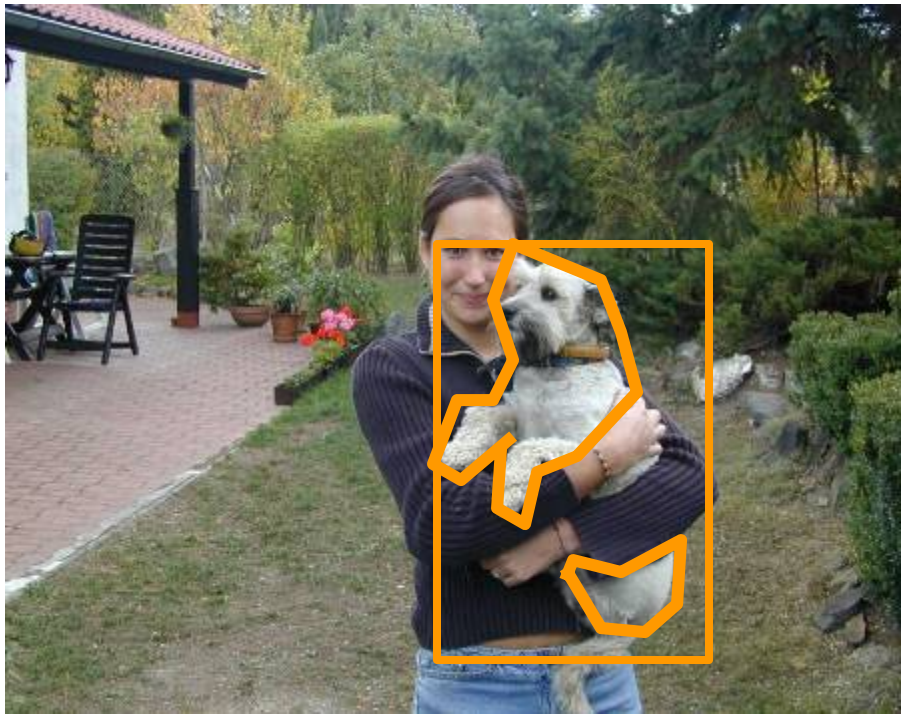
Viola, Jones 2001,

- Simple
- Computational complexity (x,y, S, $\theta$, N of classes)

  - BSW by Lampert et al 08

  - Also, Alexe, et al 10

- Localization
  - Objects are not boxes



# Recognition

– Recognition task

– Search strategy: Sliding Windows

Viola, Jones 2001,

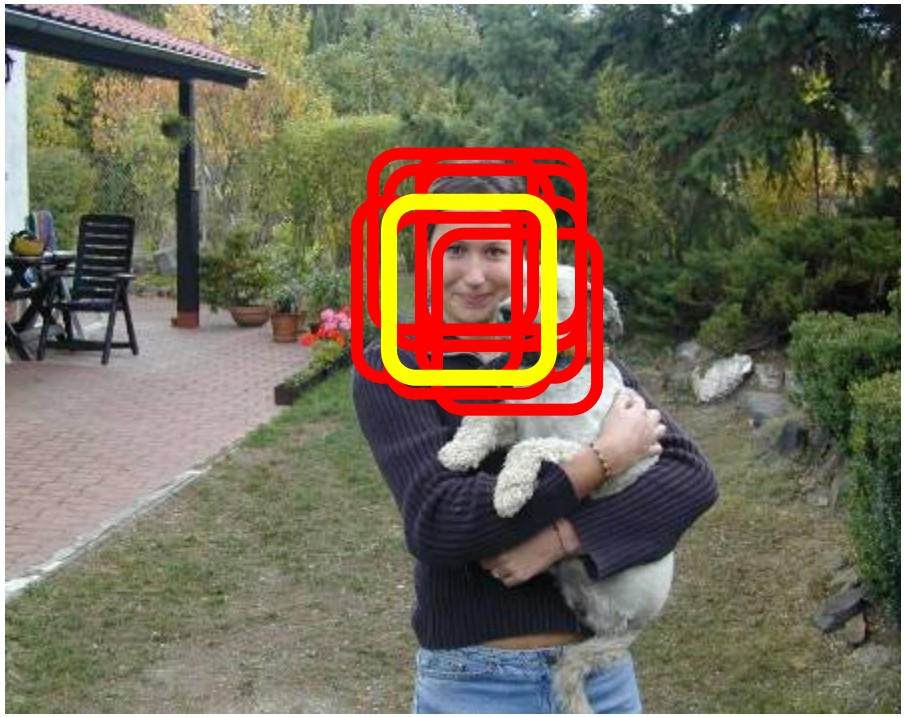
- Simple
- Computational complexity (x,y, S, $\theta$, N of classes)

  - BSW by Lampert et al 08

  - Also, Alexe, et al 10

- Localization
  - Objects are not boxes
  - Prone to false positive

**Non max suppression:**
Canny '86
....
Desai et al , 2009



# Recognition

- Recognition task
- Search strategy
- Attributes

•Savarese, 2007
•Sun et al 2009
• Liebelt et al., '08, 10
•Farhadi et al 09

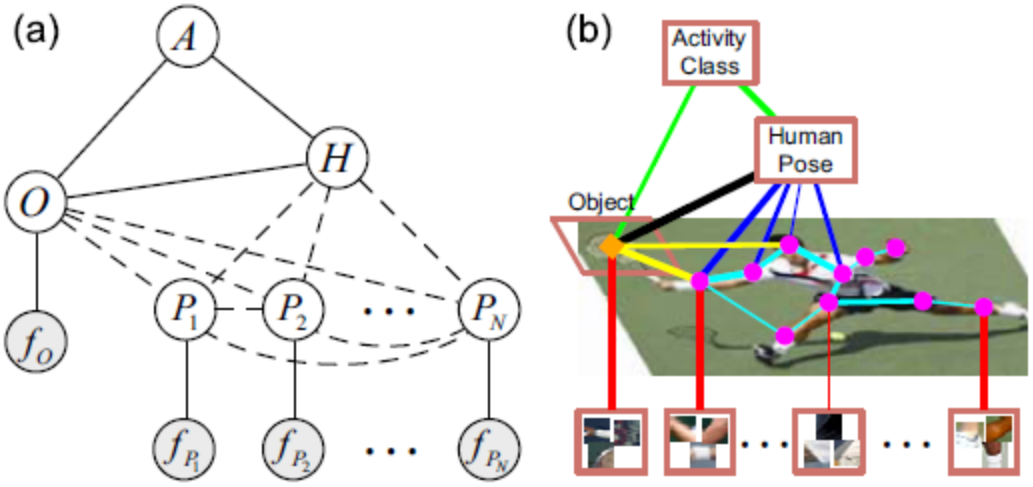
**Category: car**
**Azimuth = 225º**
**Zenith = 30º**

**- It has metal**
**- it is glossy**
**- has wheels**

•Farhadi et al 09
• Lampert et al 09
• Wang & Forsyth 09

# Recognition

- Recognition task
- Search strategy
- Attributes
- Context

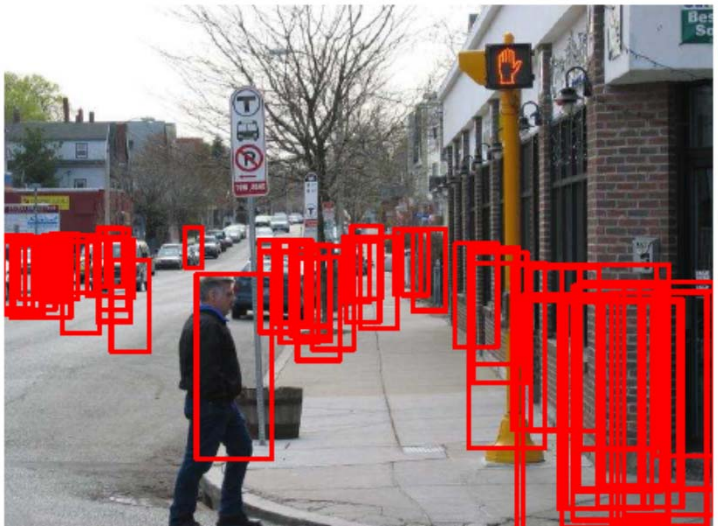



**Semantic:**

- Torralba et al 03
- Rabinovich et al 07
- Gupta & Davis 08
- Heitz & Koller 08
- L-J Li et al 08
- Yao & Fei-Fei 10

**Geometric**

- Hoiem, et al 06
- Gould et al 09
- Bao, Sun, Savarese 10



# Basic issues

- Representation
  - How to represent an object category; which classification scheme?
- Learning
  - How to learn the classifier, given training data
- Recognition
  - How the classifier is to be used on novel data

# Part 1: Bag-of-words models

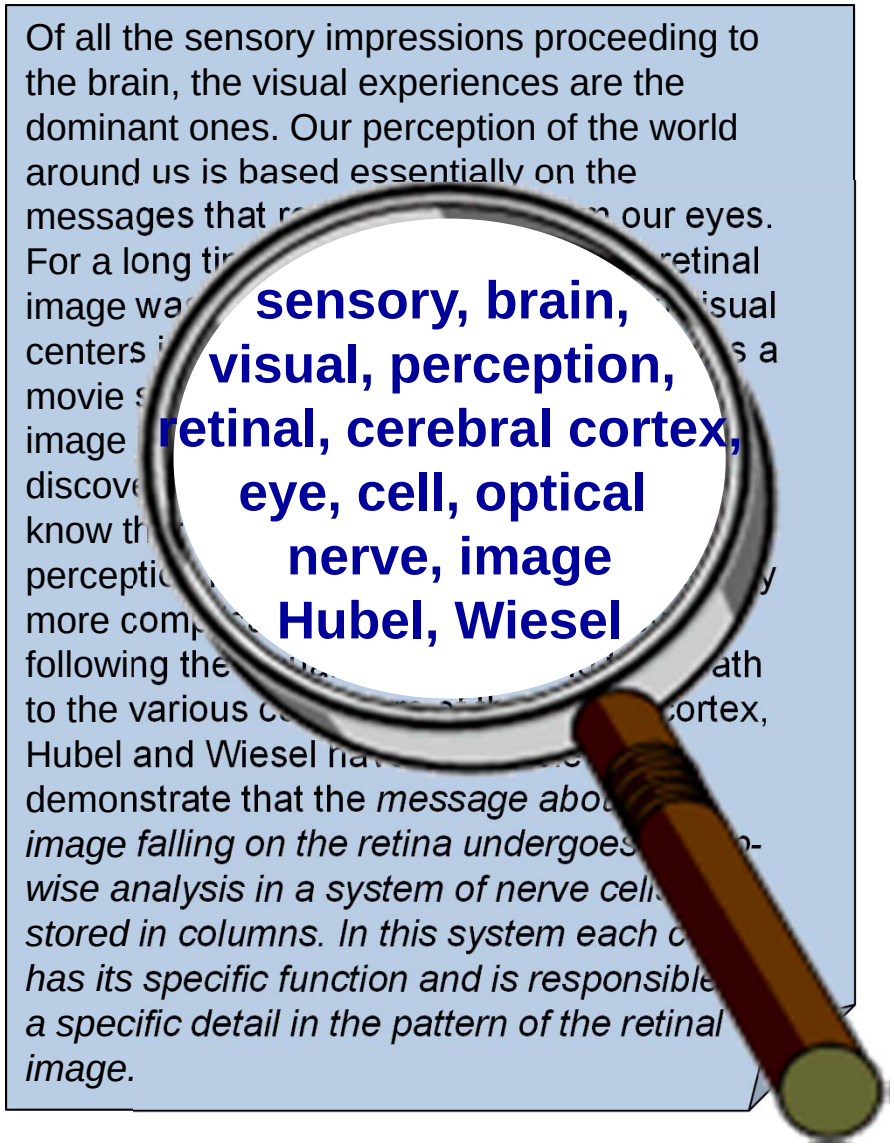
This segment is based on the tutorial *"Recognizing and Learning Object Categories: Year 2007"*, by Prof L. Fei-Fei, A. Torralba, and R. Fergus

# Related works

- Early "bag of words" models: mostly texture recognition
  - Cula & Dana, 2001; Leung & Malik 2001; Mori, Belongie & Malik, 2001; Schmid 2001; Varma & Zisserman, 2002, 2003; Lazebnik, Schmid & Ponce, 2003;
- Hierarchical Bayesian models for documents (pLSA, LDA, etc.)
  - Hoffman 1999; Blei, Ng & Jordan, 2004; Teh, Jordan, Beal & Blei, 2004
- Object categorization
  - Csurka, Bray, Dance & Fan, 2004; Sivic, Russell, Efros, Freeman & Zisserman, 2005; Sudderth, Torralba, Freeman & Willsky, 2005;
- Natural scene categorization
  - Vogel & Schiele, 2004; Fei-Fei & Perona, 2005; Bosch, Zisserman & Munoz, 2006

# Object → Bag of 'words'

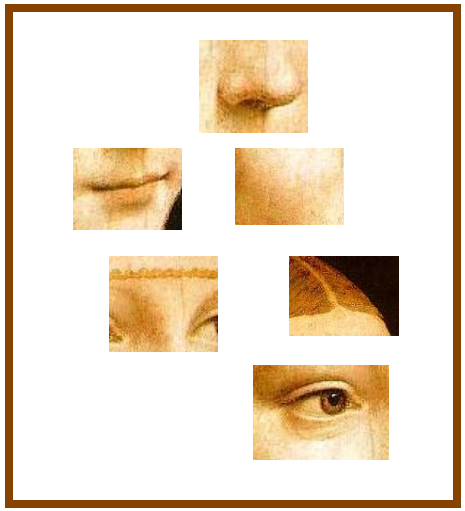
# Analogy to documents




# definition of “BoW”

– Independent features

face

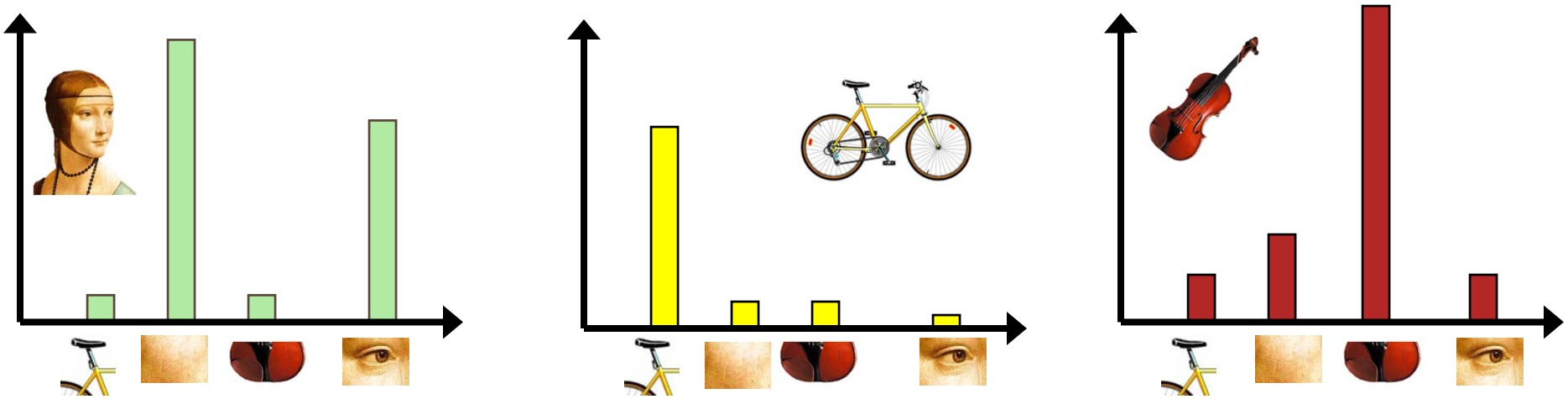


bike

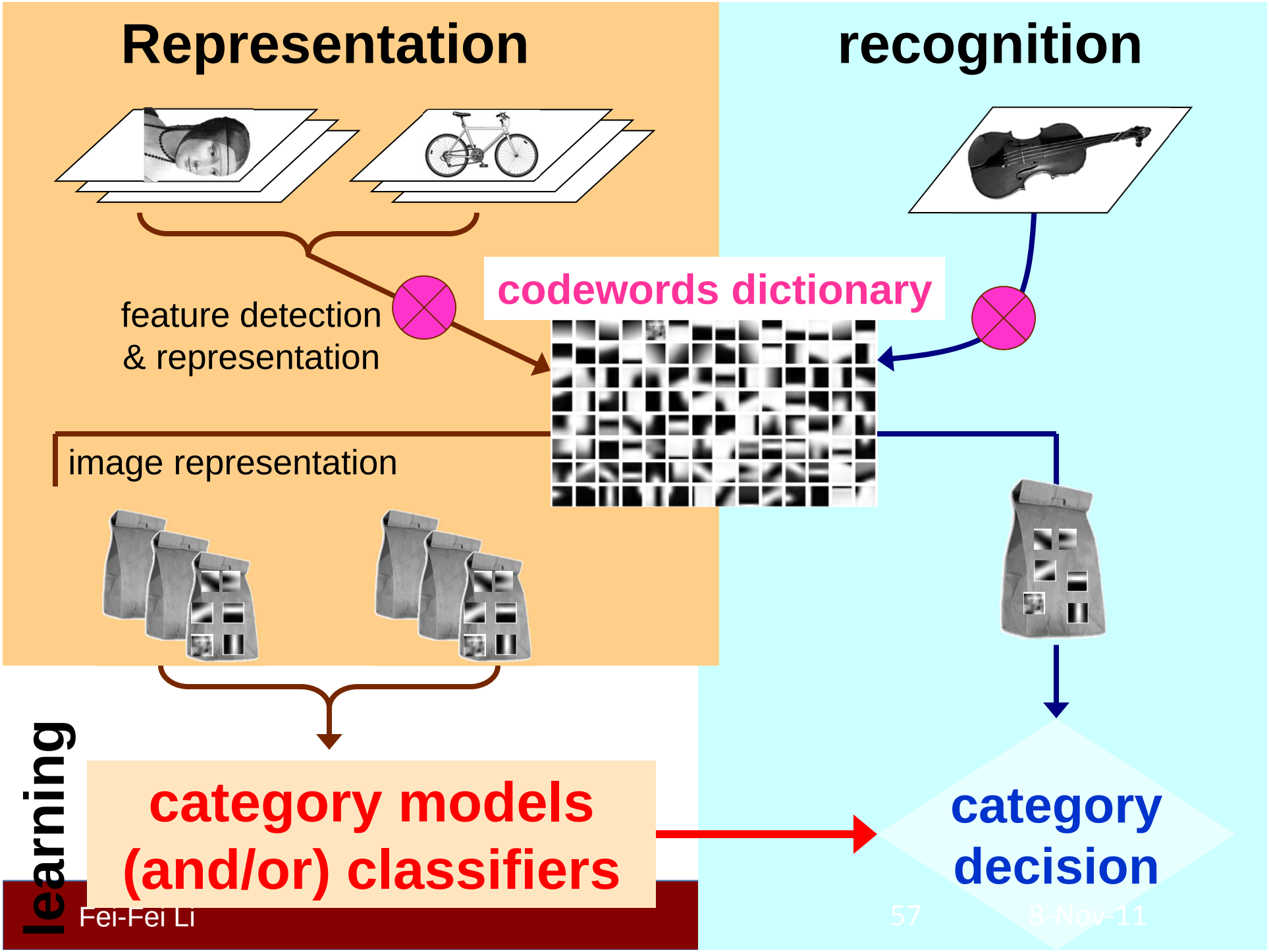
violin

# definition of “BoW”

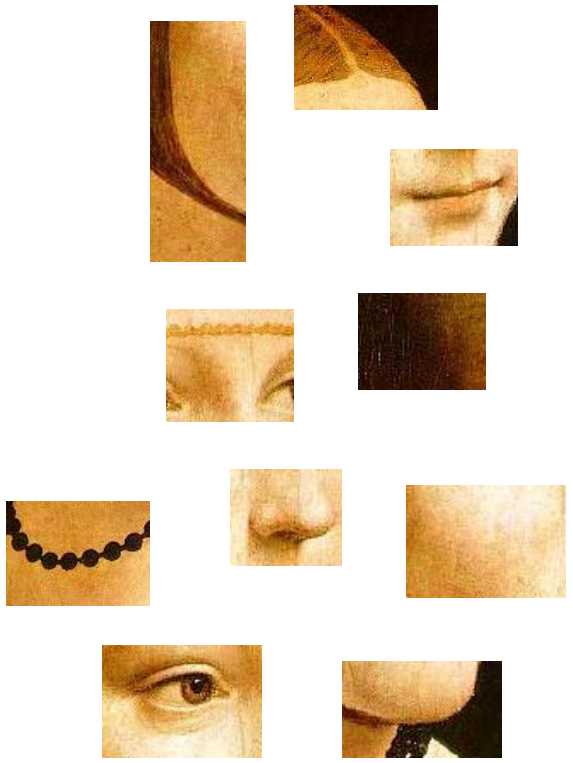
- Independent features
- histogram representation



codewords dictionary


Representation
recognition
feature detection
& representation
codewords dictionary
image representation
learning
category models
(and/or) classifiers
category
decision

# 1.Feature detection and representation

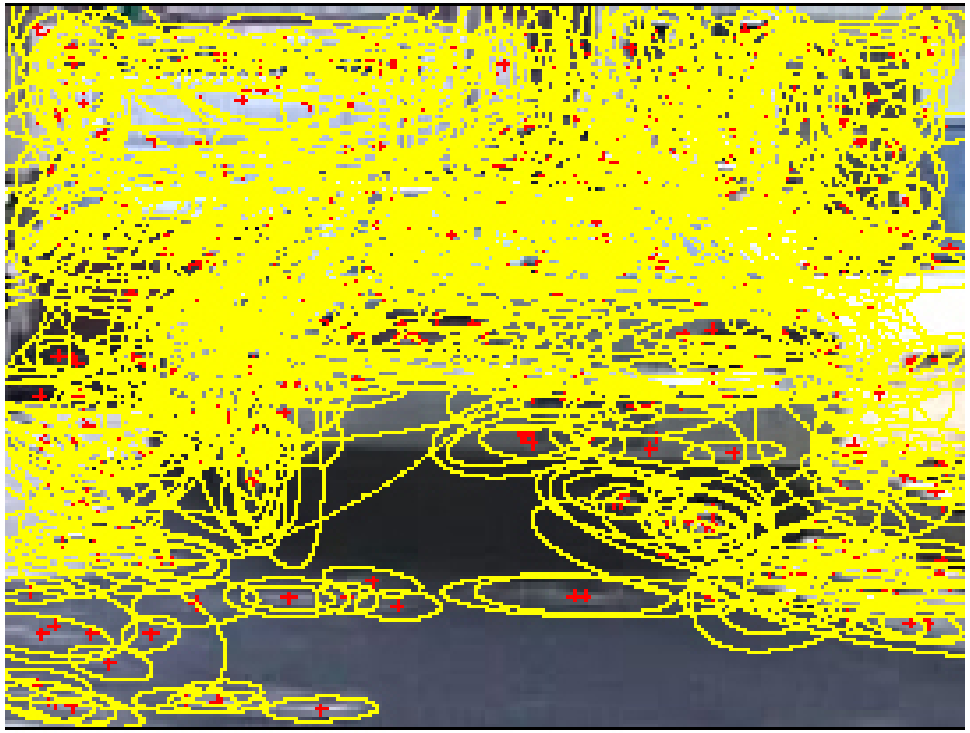


# 1.Feature detection and representation

- Regular grid
  - Vogel & Schiele, 2003
  - Fei-Fei & Perona, 2005

# 1.Feature detection and representation

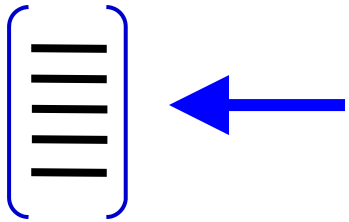
- Regular grid
  - Vogel & Schiele, 2003
  - Fei-Fei & Perona, 2005
- Interest point detector
  - Csurka, et al. 2004
  - Fei-Fei & Perona, 2005
  - Sivic, et al. 2005



# 1.Feature detection and representation

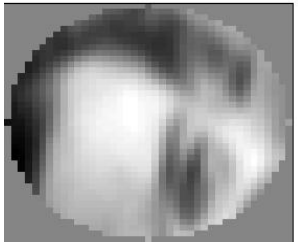
- Regular grid
  - Vogel & Schiele, 2003
  - Fei-Fei & Perona, 2005
- Interest point detector
  - Csurka, Bray, Dance & Fan, 2004
  - Fei-Fei & Perona, 2005
  - Sivic, Russell, Efros, Freeman & Zisserman, 2005
- Other methods
  - Random sampling (Vidal-Naquet & Ullman, 2002)
  - Segmentation based patches (Barnard, Duygulu, Forsyth, de Freitas, Blei, Jordan, 2003)

# 1.Feature detection and representation

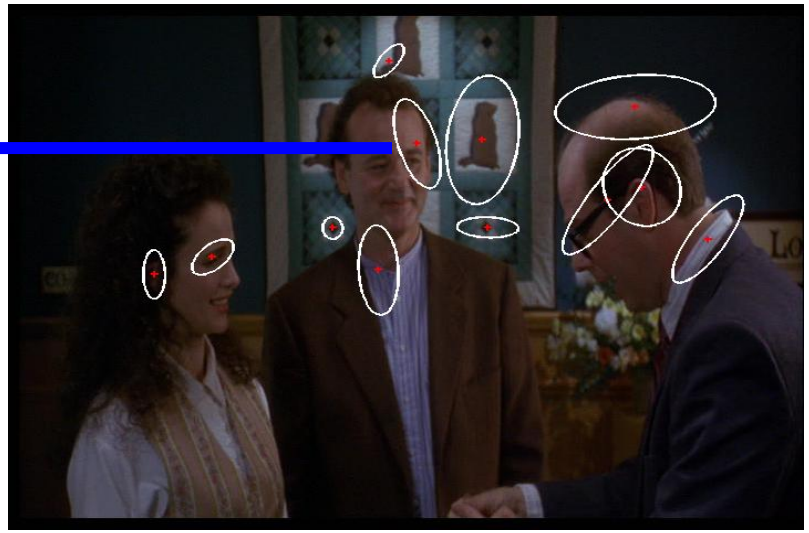


**Compute SIFT descriptor**

[Lowe'99]



**Normalize patch**



Detect patches

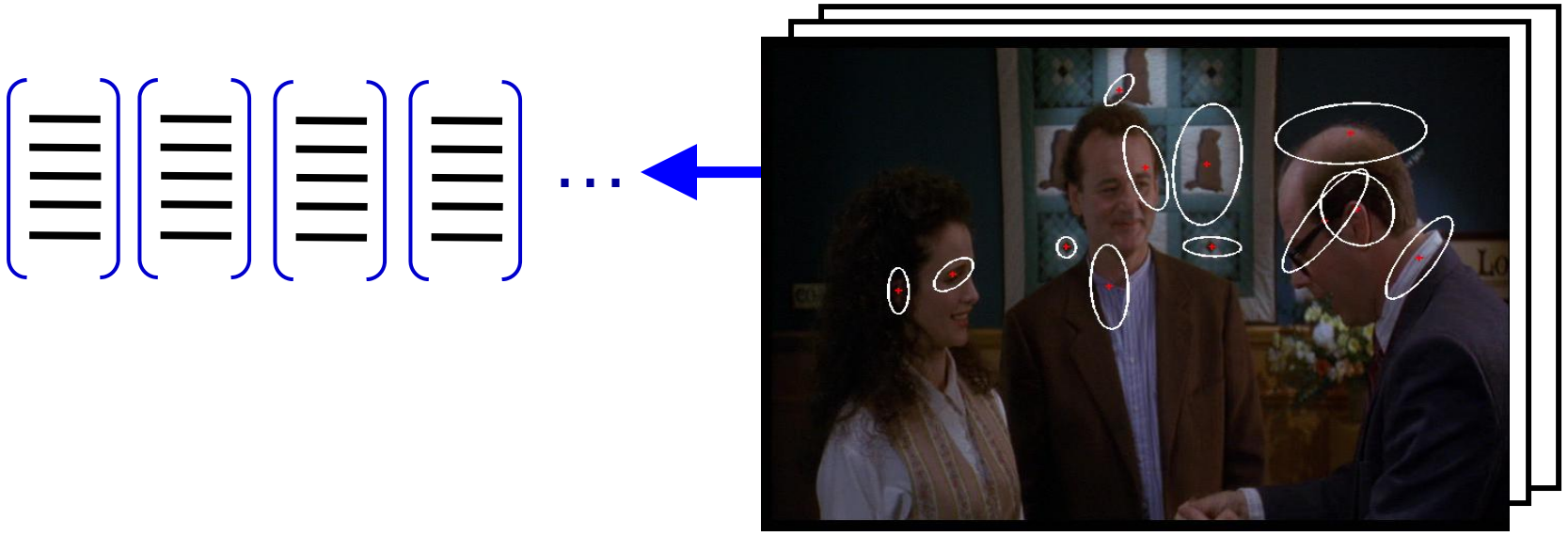
[Mikojaczyk and Schmid '02]

[Mata, Chum, Urban & Pajdla, '02]

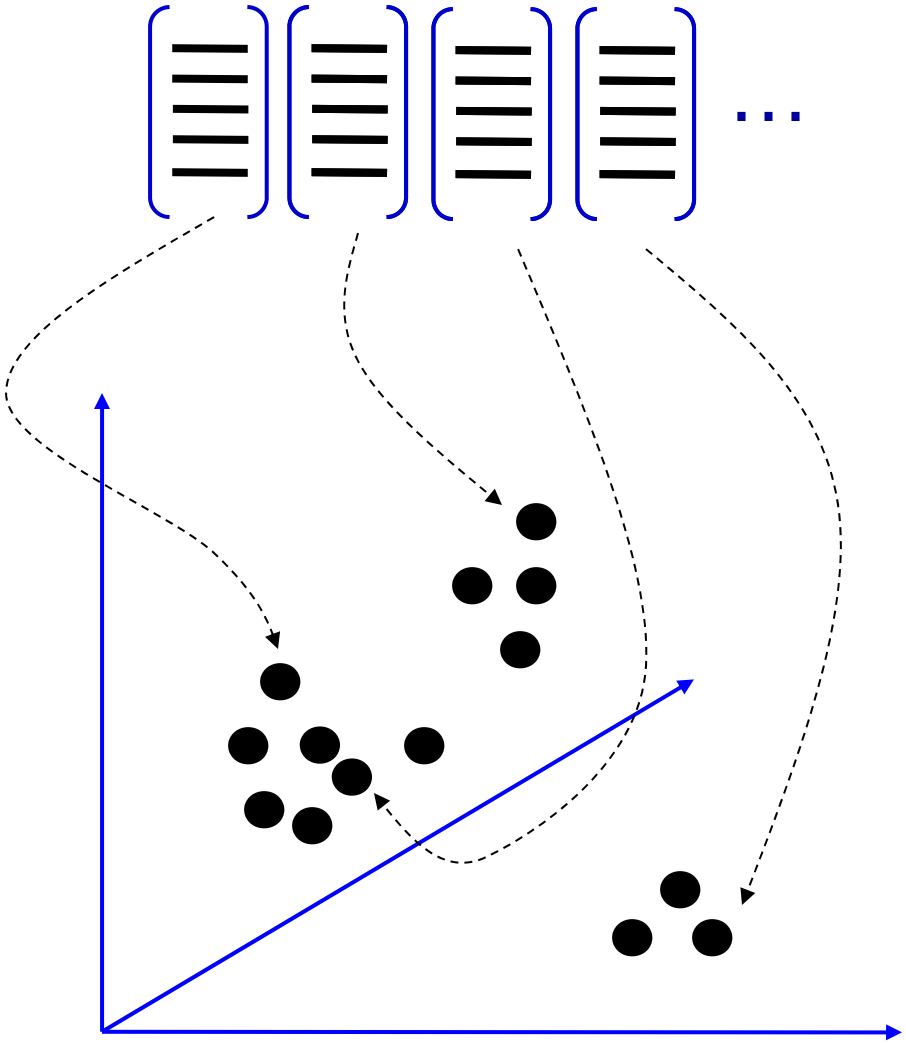
[Sivic & Zisserman, '03]

Slide credit: Josef Sivic

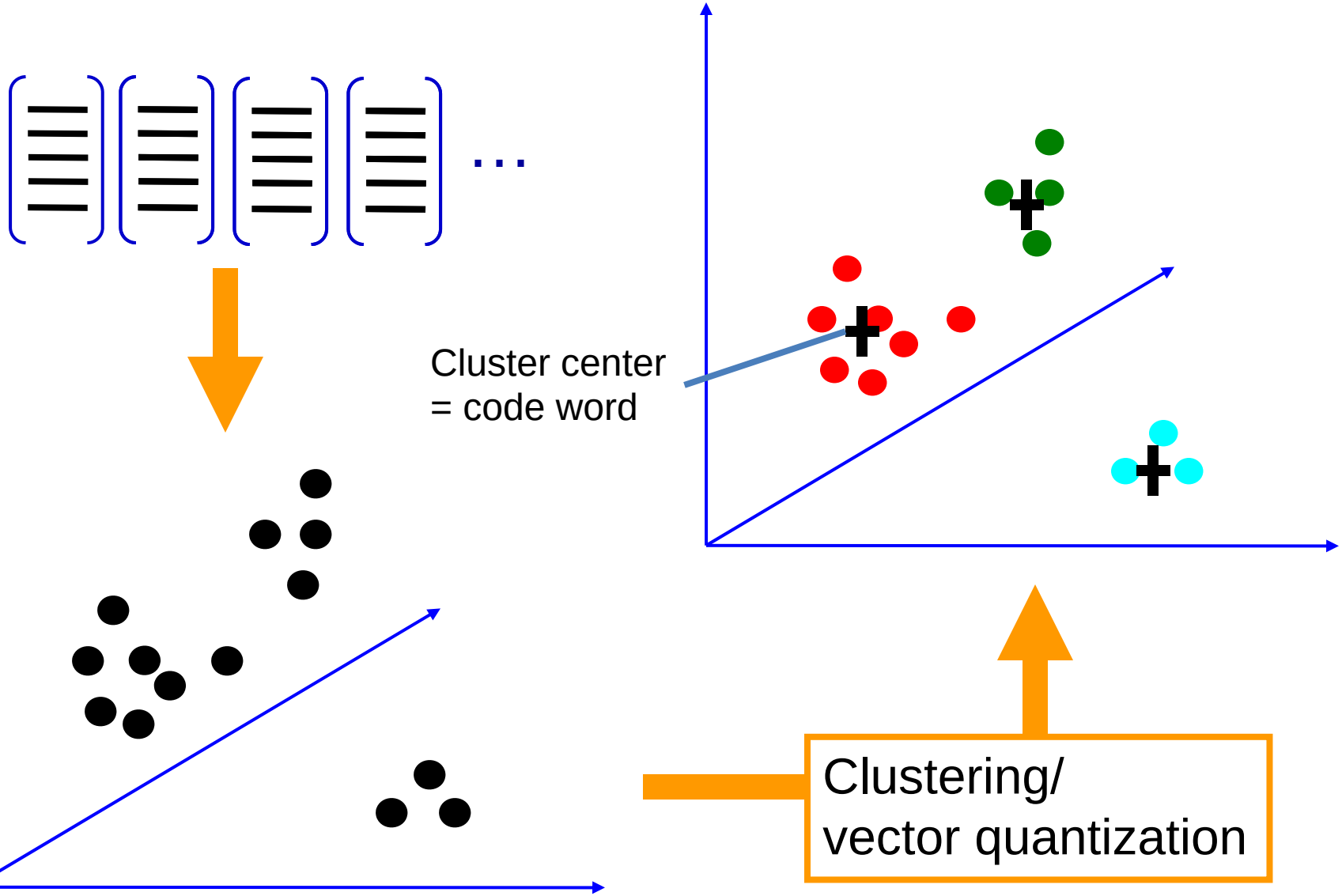
# 1.Feature detection and representation



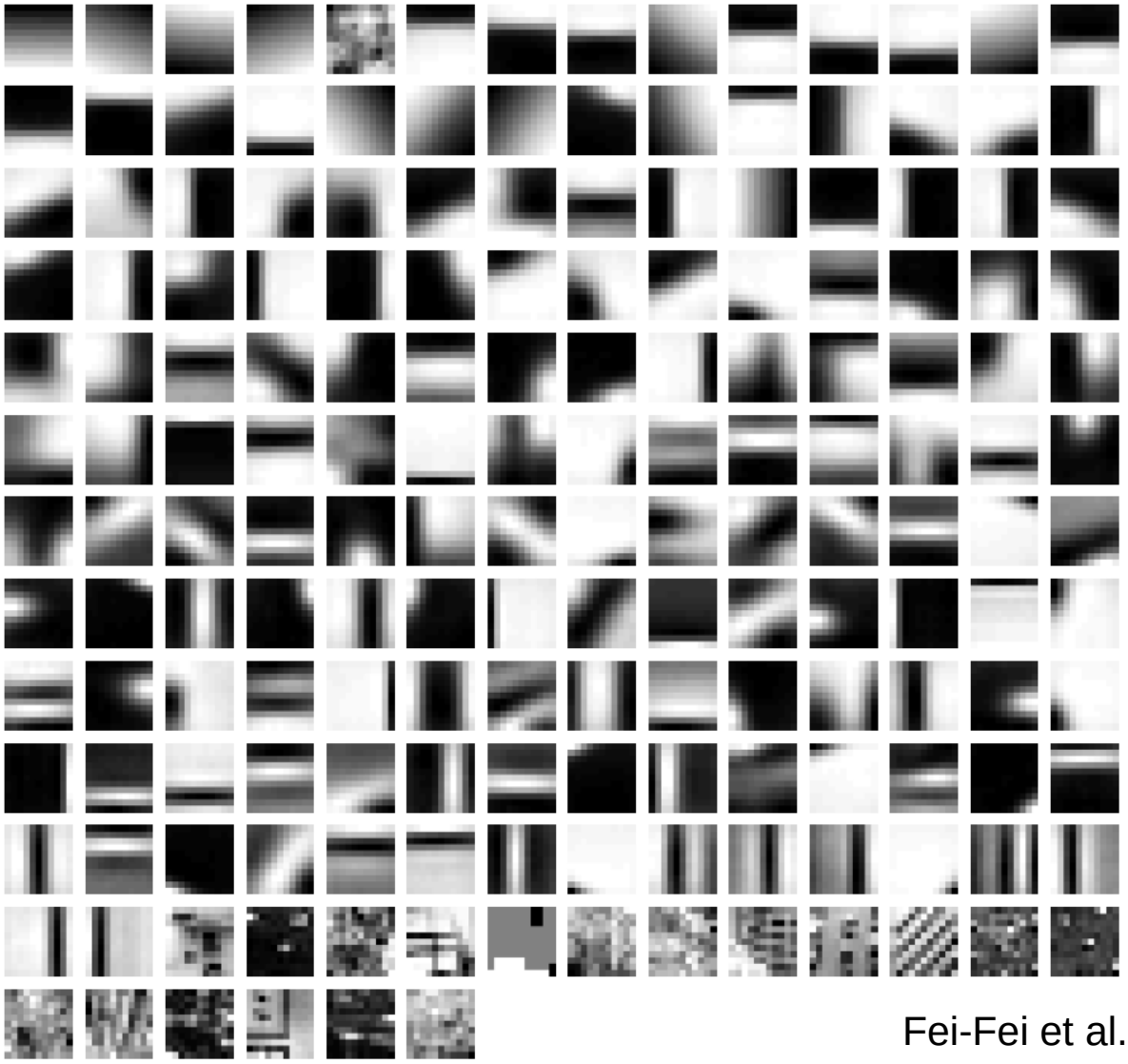
# 2. Codewords dictionary formation



# 2. Codewords dictionary formation

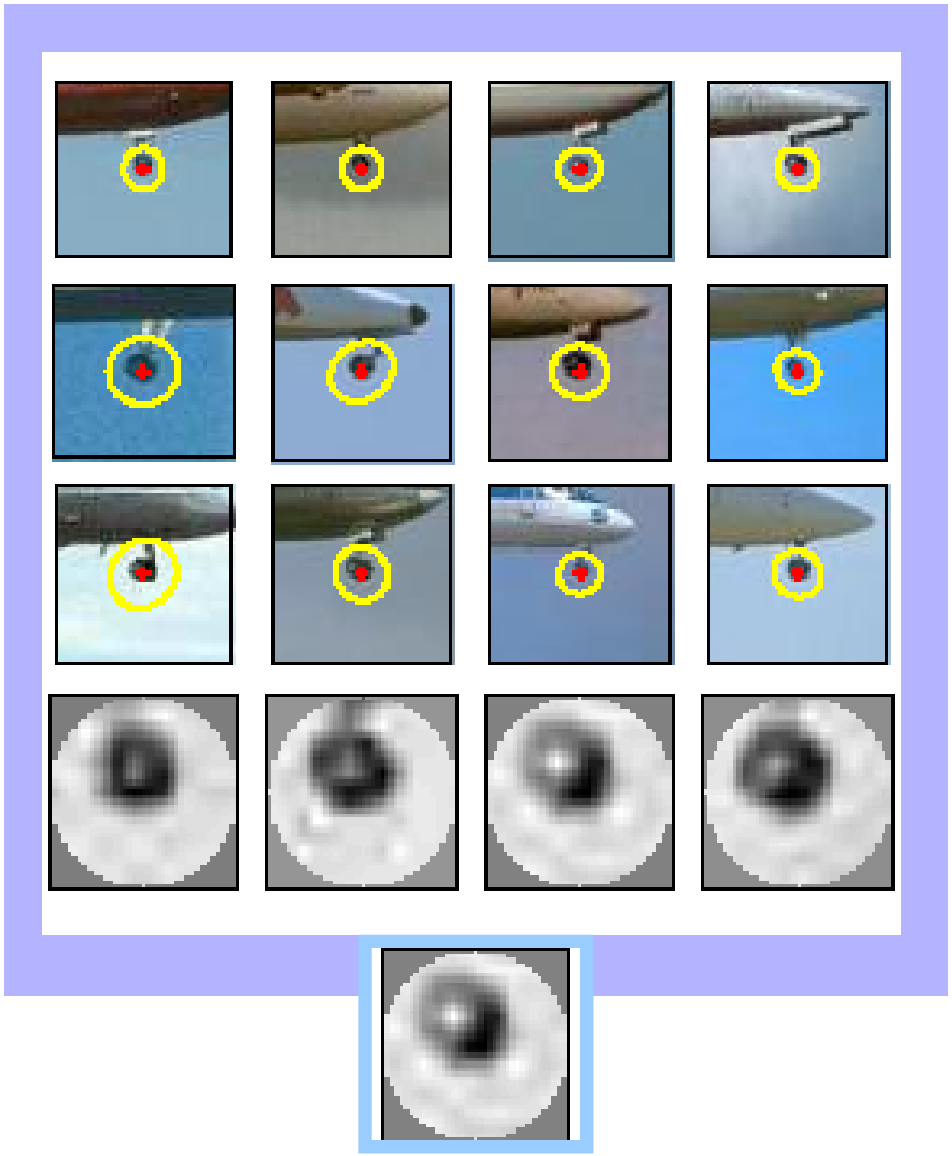


# 2. Codewords dictionary formation



Fei-Fei et al. 2005

# Image patch examples of codewords

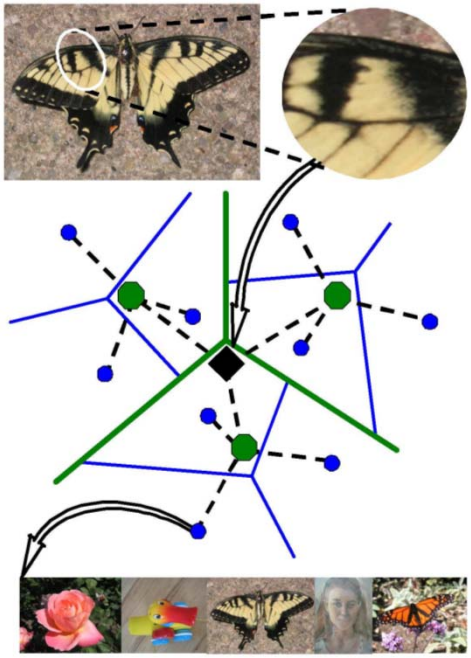


Sivic et al. 2005

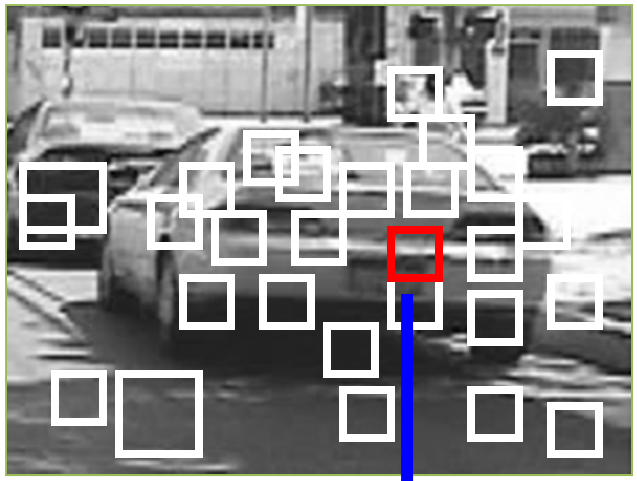
# Visual vocabularies: Issues

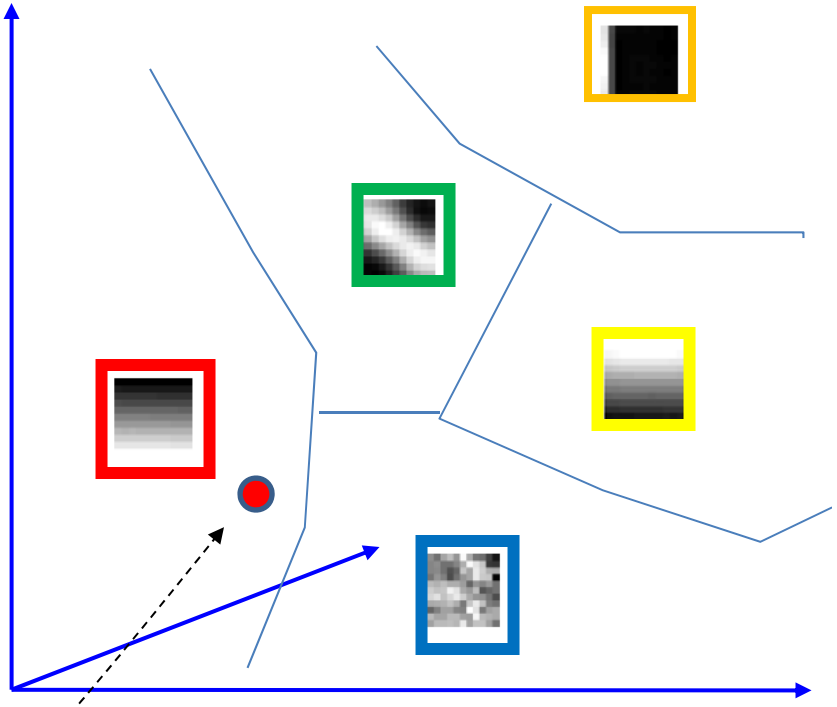
- How to choose vocabulary size?
  - Too small: visual words not representative of all patches
  - Too large: quantization artifacts, overfitting

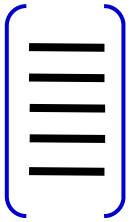
- Computational efficiency
  - Vocabulary trees
    (Nister & Stewenius, 2006)

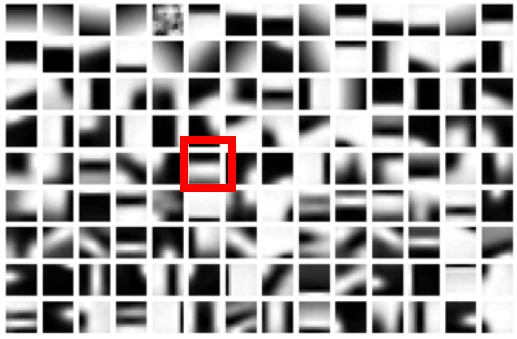


# 3. Bag of word representation





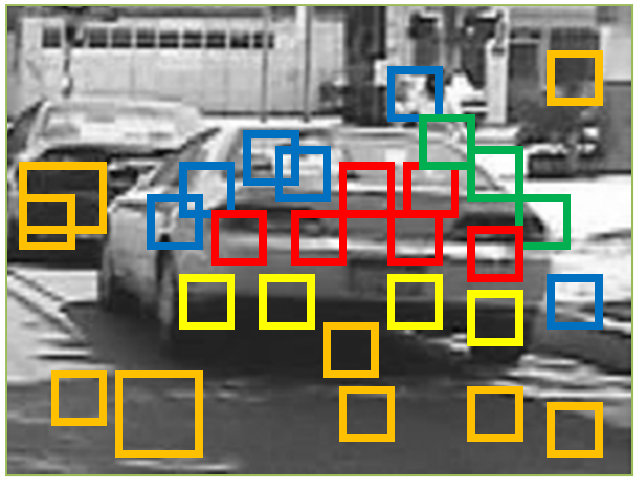


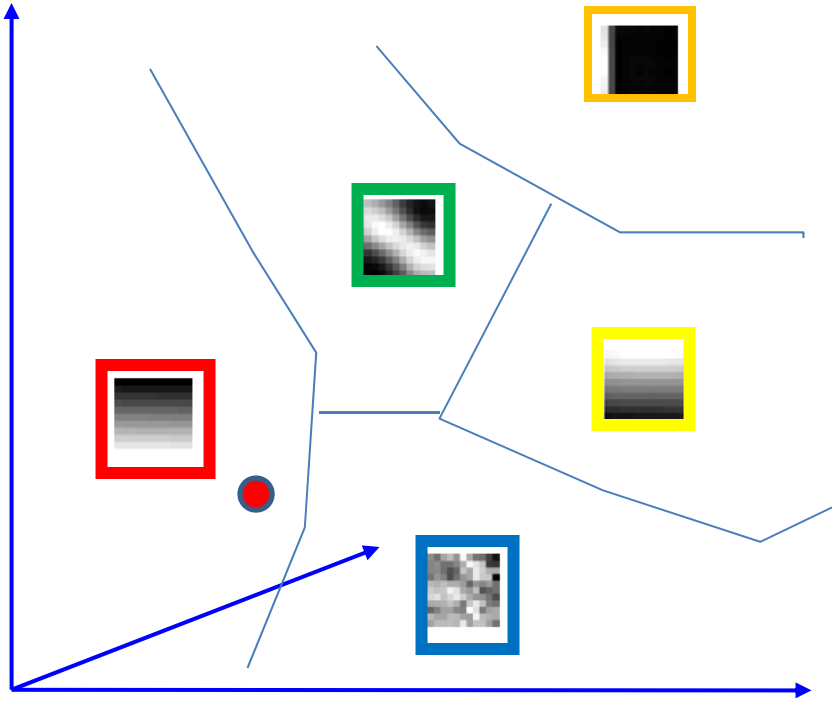


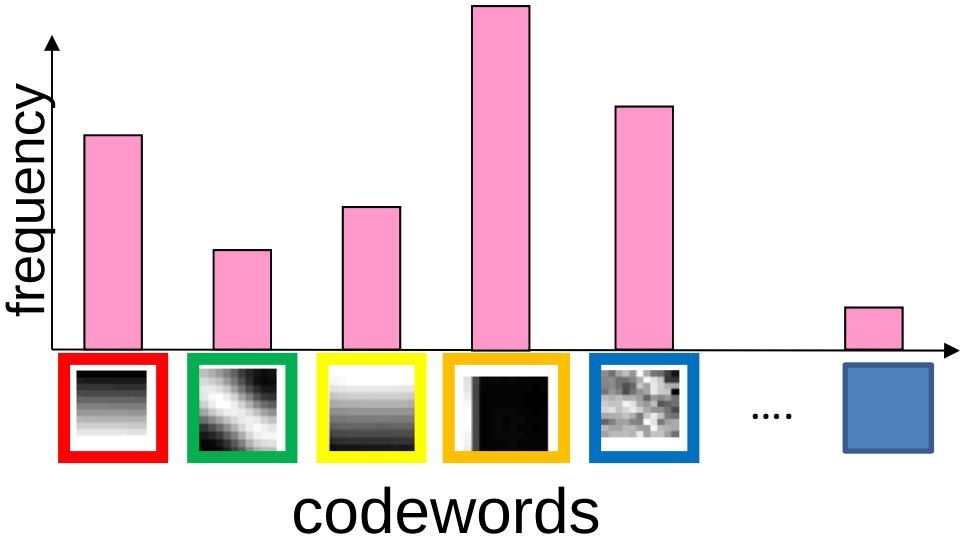
Codewords dictionary

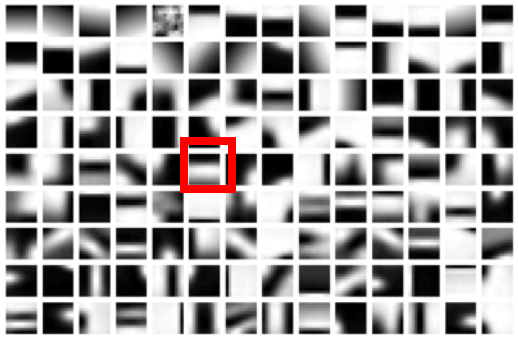
- Nearest neighbors assignment
- K-D tree search strategy

# 3. Bag of word representation



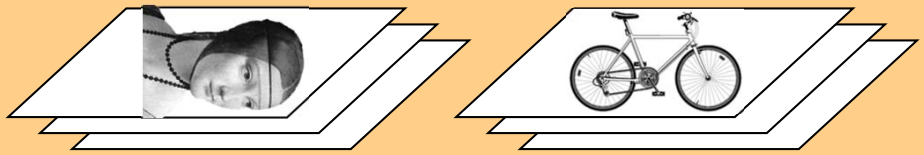




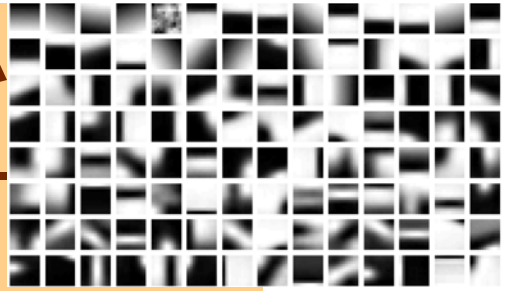
Codewords dictionary

# Representation



2.

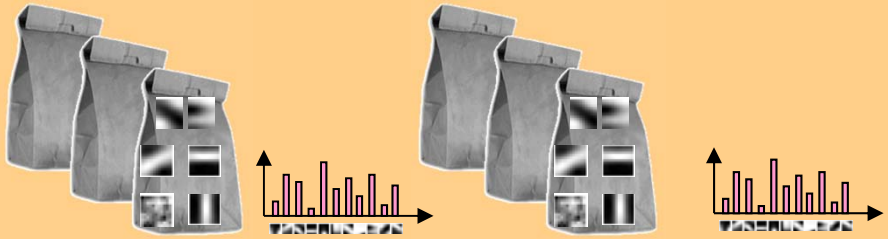
codewords dictionary



1. feature detection & representation

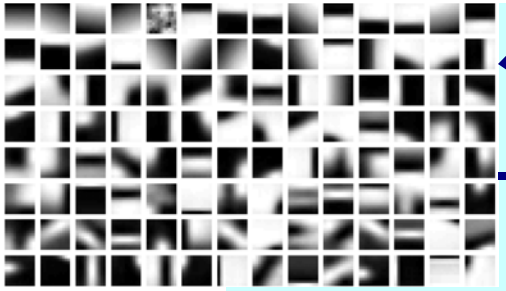
image representation

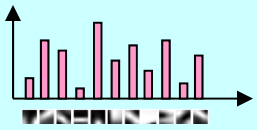
3.

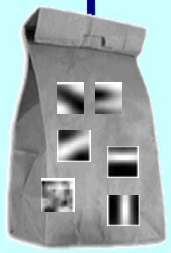


# Learning and Recognition

codewords dictionary







**category models (and/or) classifiers**

**category decision**

# Learning and Recognition

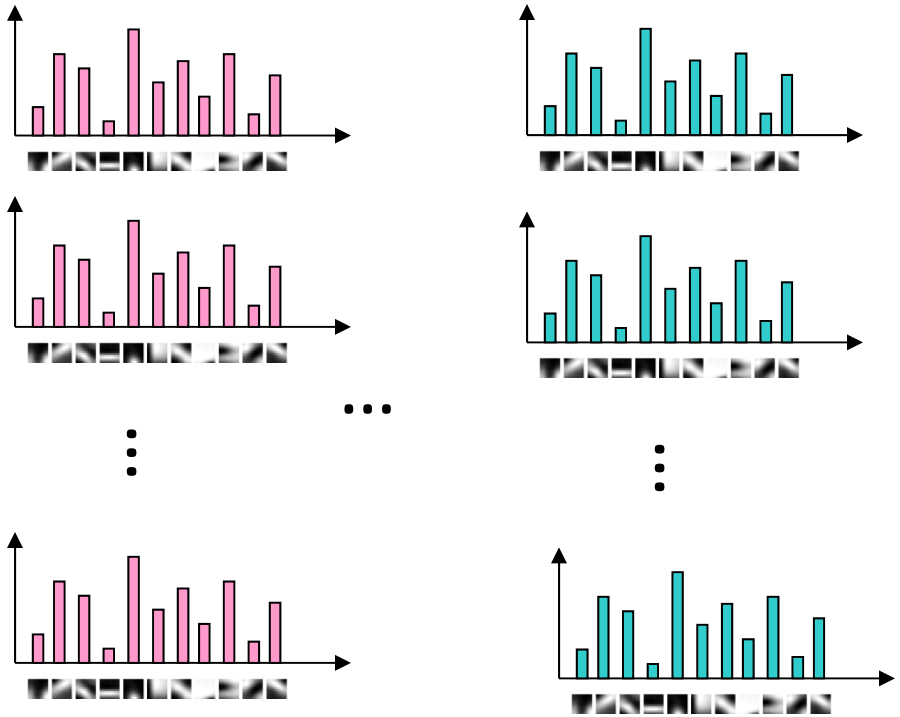
1. Discriminative method:
   - NN
   - SVM

2. Generative method:
   - graphical models

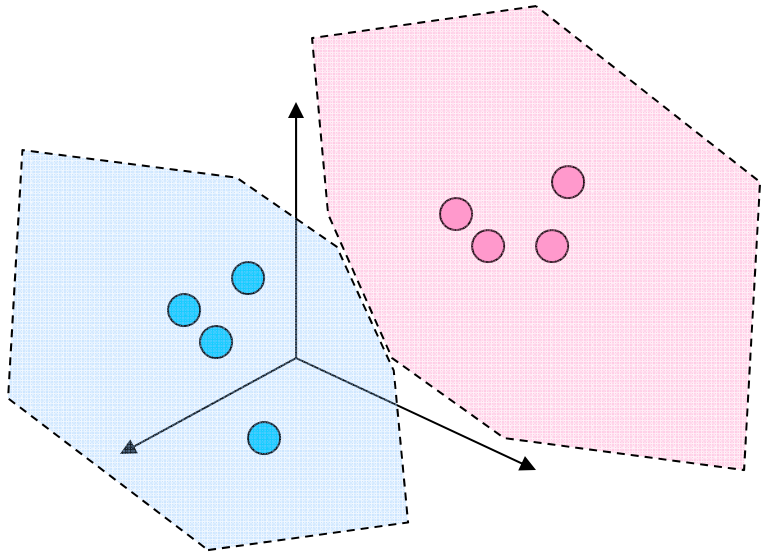
**category models (and/or) classifiers**

# Discriminative classifiers

**category models**

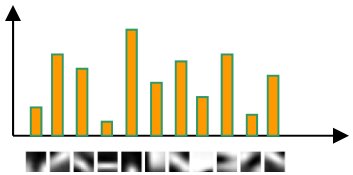



Model space



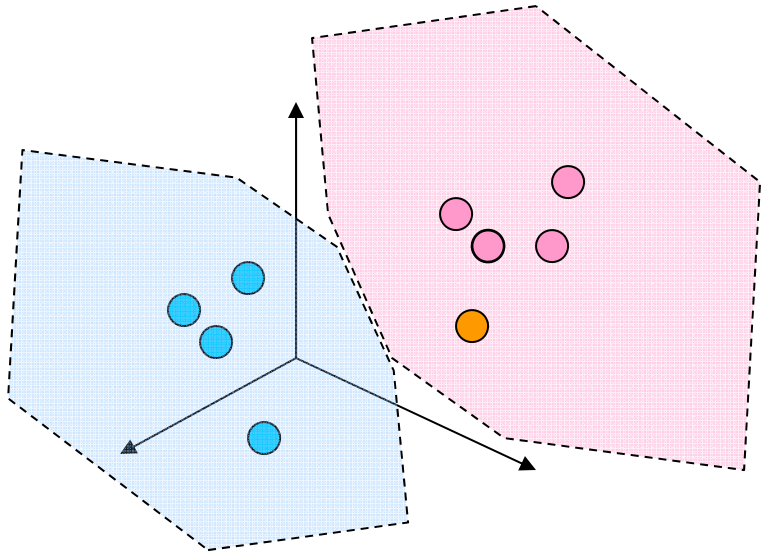
# Discriminative classifiers

**Query image**



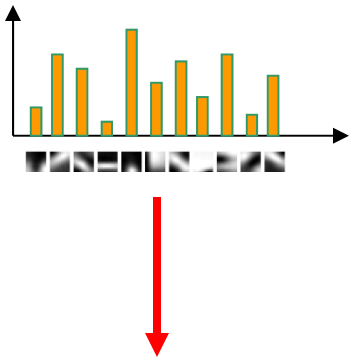
Winning class: pink

Model space



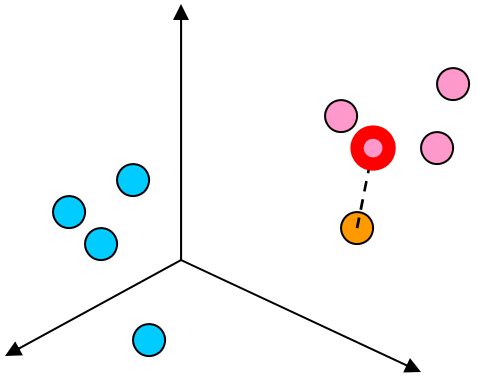
# Nearest Neighbors classifier

**Query image**



Winning class: pink

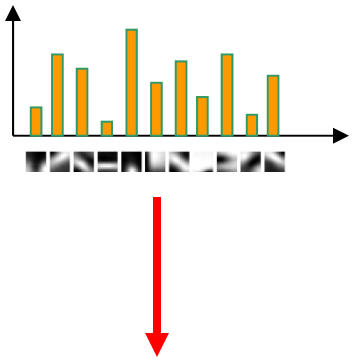
Model space



- Assign label of nearest training data point to each test data point

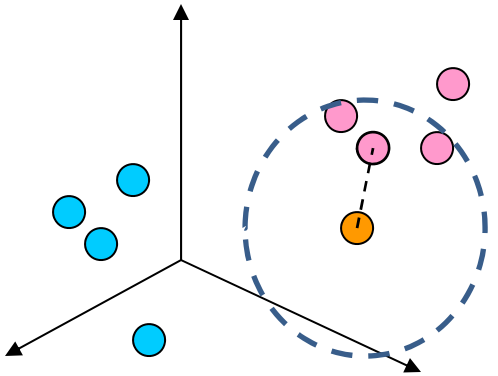
# K- Nearest Neighbors classifier

**Query image**



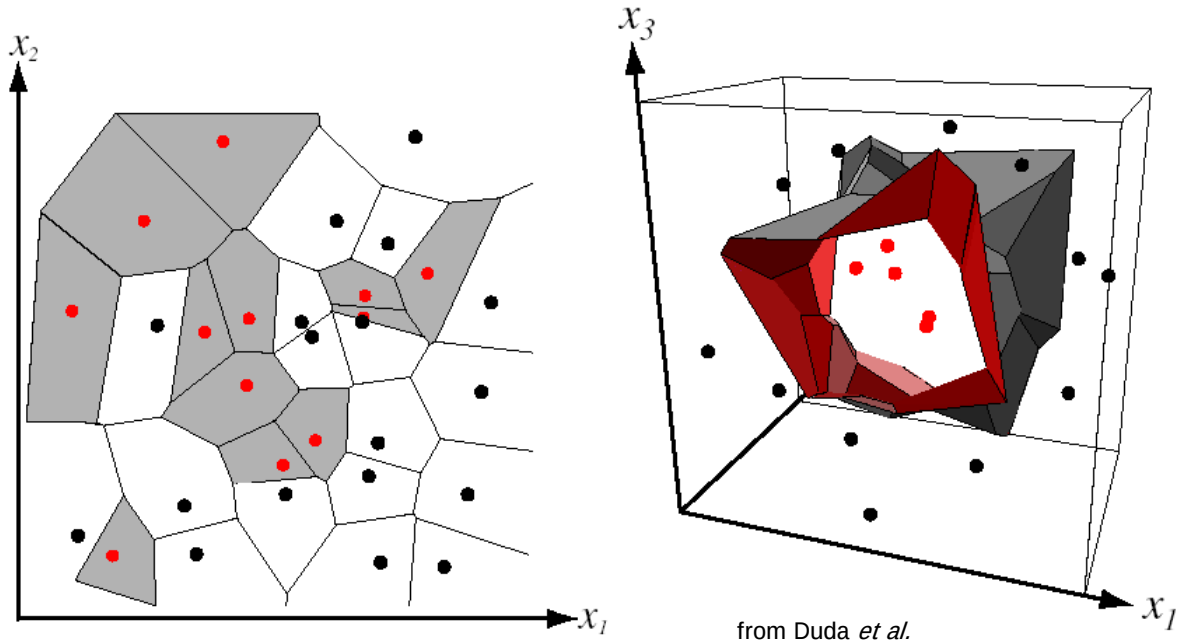
Winning class: pink

Model space



- For a new point, find the k closest points from training data
- Labels of the k points “vote” to classify
- Works well provided there is lots of data and the distance function is good

# K- Nearest Neighbors classifier




from Duda *et al.*

- Voronoi partitioning of feature space for 2-category 2-D and 3-D data
- For k dimensions: k-D tree = space-partitioning data structure for organizing points in a *k*-dimensional space
- Enable efficient search
- Nice tutorial: http://www.cs.umd.edu/class/spring2002/cmsc420-0401/pbasic.pdf

# Functions for comparing histograms

- L1 distance

$$D(h_1, h_2) = \sum_{i=1}^{N} | h_1(i) - h_2(i) |$$

- $\chi^2$ distance

$$D(h_1, h_2) = \sum_{i=1}^{N} \frac{(h_1(i) - h_2(i))^2}{h_1(i) + h_2(i)}$$

- Quadratic distance (*cross-bin*)

$$D(h_1, h_2) = \sum_{i,j} A_{ij} (h_1(i) - h_2(j))^2$$

Jan Puzicha, Yossi Rubner, Carlo Tomasi, Joachim M. Buhmann: Empirical Evaluation of Dissimilarity Measures for Color and Texture. ICCV 1999

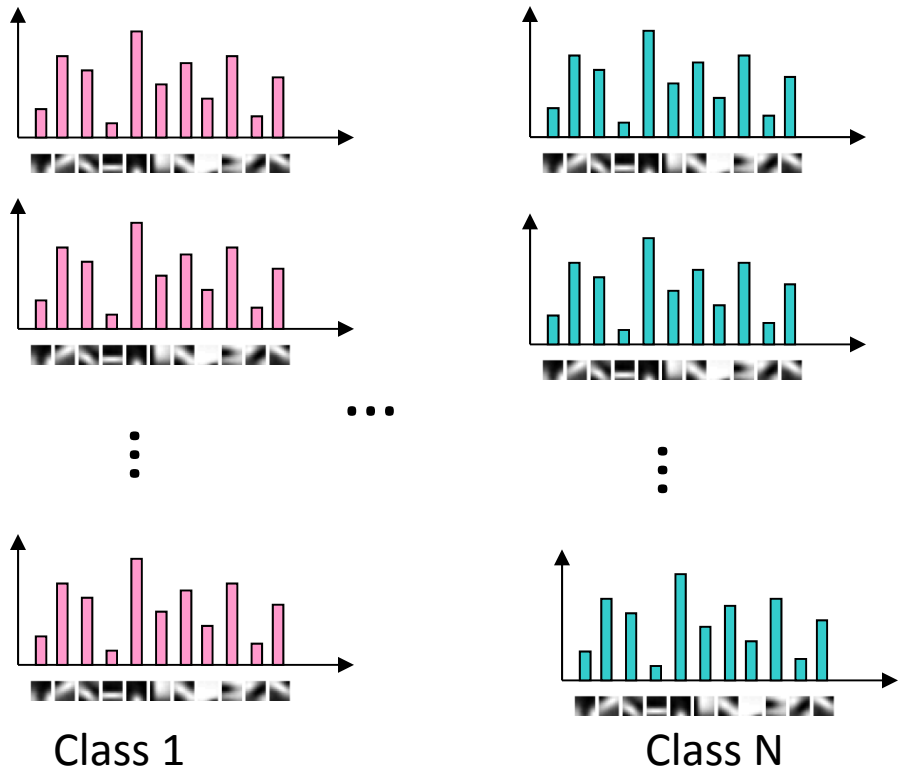
# Learning and Recognition

1. Discriminative method:
   - NN
   - SVM

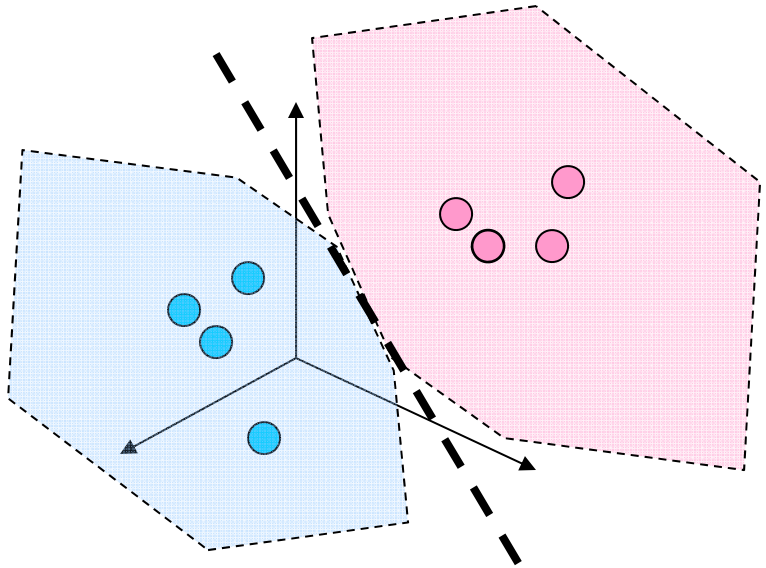
2. Generative method:
   - graphical models

# Discriminative classifiers (linear classifier)

**category models**

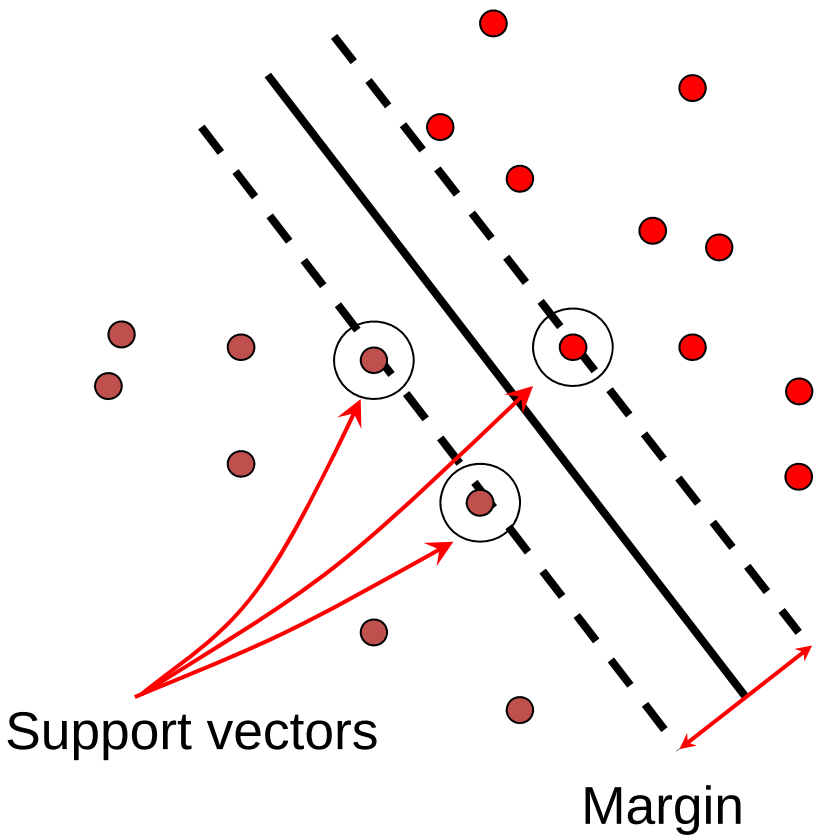



Model space



# Support vector machines

- Find hyperplane that maximizes the *margin* between the positive and negative examples




Support vectors: $\mathbf{x}_i \cdot \mathbf{w} + b = \pm 1$

Distance between point and hyperplane: $\frac{|\mathbf{x}_i \cdot \mathbf{w} + b|}{\|\mathbf{w}\|}$

Margin = $2 / \|\mathbf{w}\|$

Solution: $\mathbf{w} = \sum_i \alpha_i y_i \mathbf{x}_i$

Classification function (decision boundary):

$$\mathbf{w} \cdot \mathbf{x} + b = \sum_i \alpha_i y_i \mathbf{x}_i \cdot \mathbf{x} + b$$

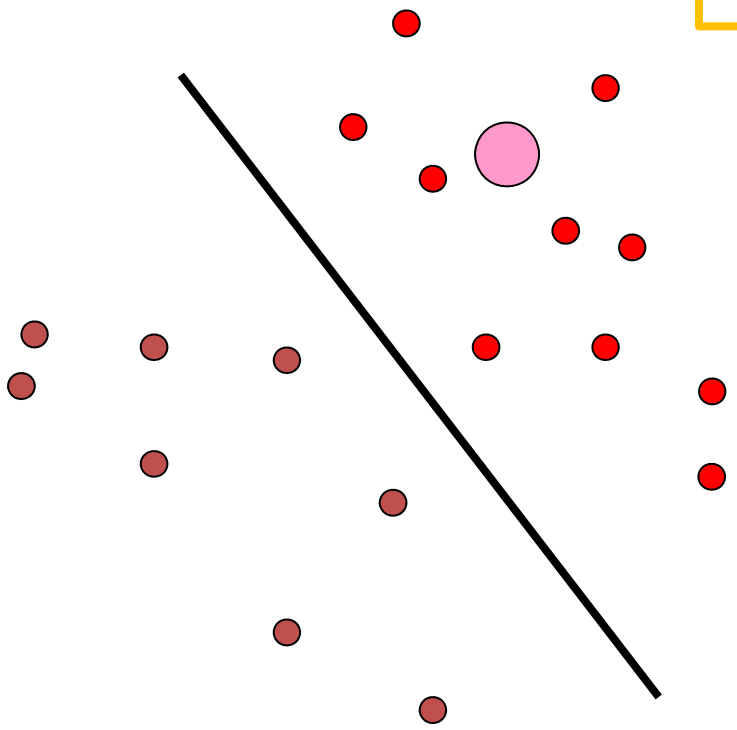
Credit slide: S. Lazebnik

# Support vector machines

- Classification

$$\mathbf{w} \cdot \mathbf{x} + b = \sum_i \alpha_i y_i \mathbf{x}_i \cdot \mathbf{x} + b$$

Test point



Margin

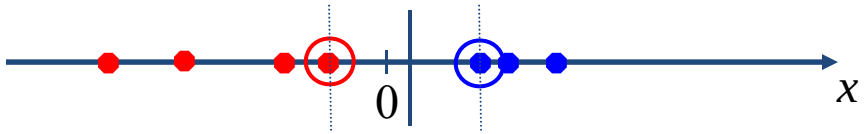
$$if \quad \mathbf{x} \cdot \mathbf{w} + b \geq 0 \rightarrow \quad class\ 1$$

$$if \quad \mathbf{x} \cdot \mathbf{w} + b < 0 \rightarrow \quad class\ 2$$

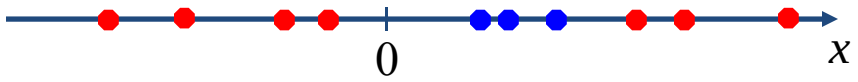
C. Burges, A Tutorial on Support Vector Machines for Pattern Recognition, Data Mining and Knowledge Discovery, 1998

# Nonlinear SVMs

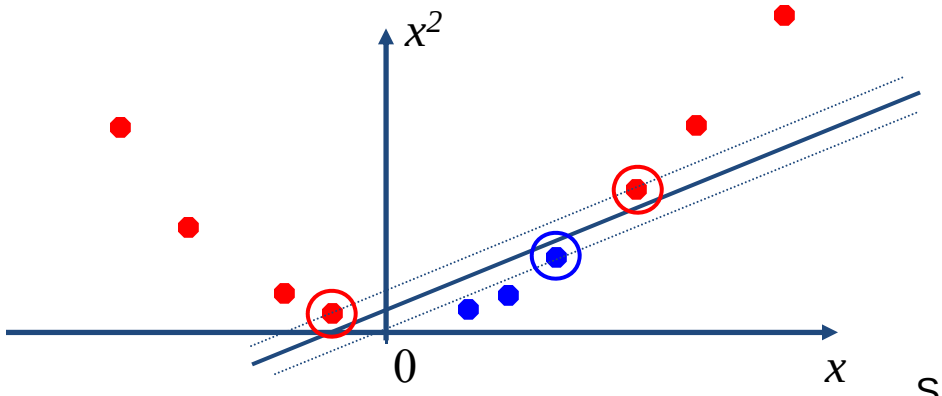
- Datasets that are linearly separable work out great:




- But what if the dataset is just too hard?



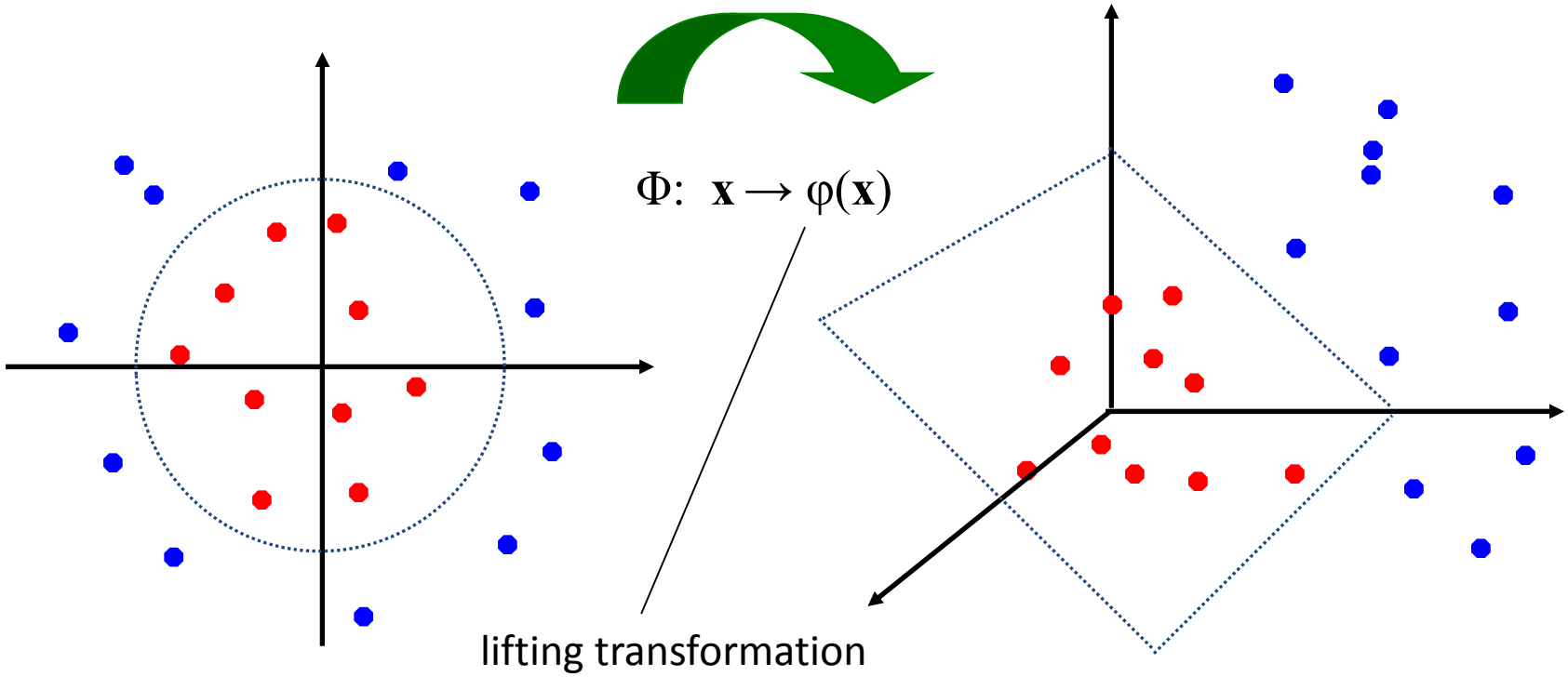

- We can map it to a higher-dimensional space:




Slide credit: Andrew Moore

# Nonlinear SVMs

- General idea: the original input space can always be mapped to some higher-dimensional feature space where the training set is separable:




Slide credit: Andrew Moore

# Nonlinear SVMs

- Nonlinear decision boundary in the original feature space:

$$\sum_i \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) + b$$

- *The kernel K* = product of the lifting transformation $\boldsymbol{\varphi}(\mathbf{x})$:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \boldsymbol{\varphi}(\mathbf{x}_i) \cdot \boldsymbol{\varphi}(\mathbf{x}_j)$$

NOTE:

- It is not required to compute $\boldsymbol{\varphi}(\mathbf{x})$ explicitly:
- The kernel must satisfy the "Mercer inequality"

C. Burges, A Tutorial on Support Vector Machines for Pattern Recognition, Data Mining and Knowledge Discovery, 1998

# Kernels for bags of features

- Histogram intersection kernel:

$$I(h_1, h_2) = \sum_{i=1}^{N} \min(h_1(i), h_2(i))$$

- Generalized Gaussian kernel:

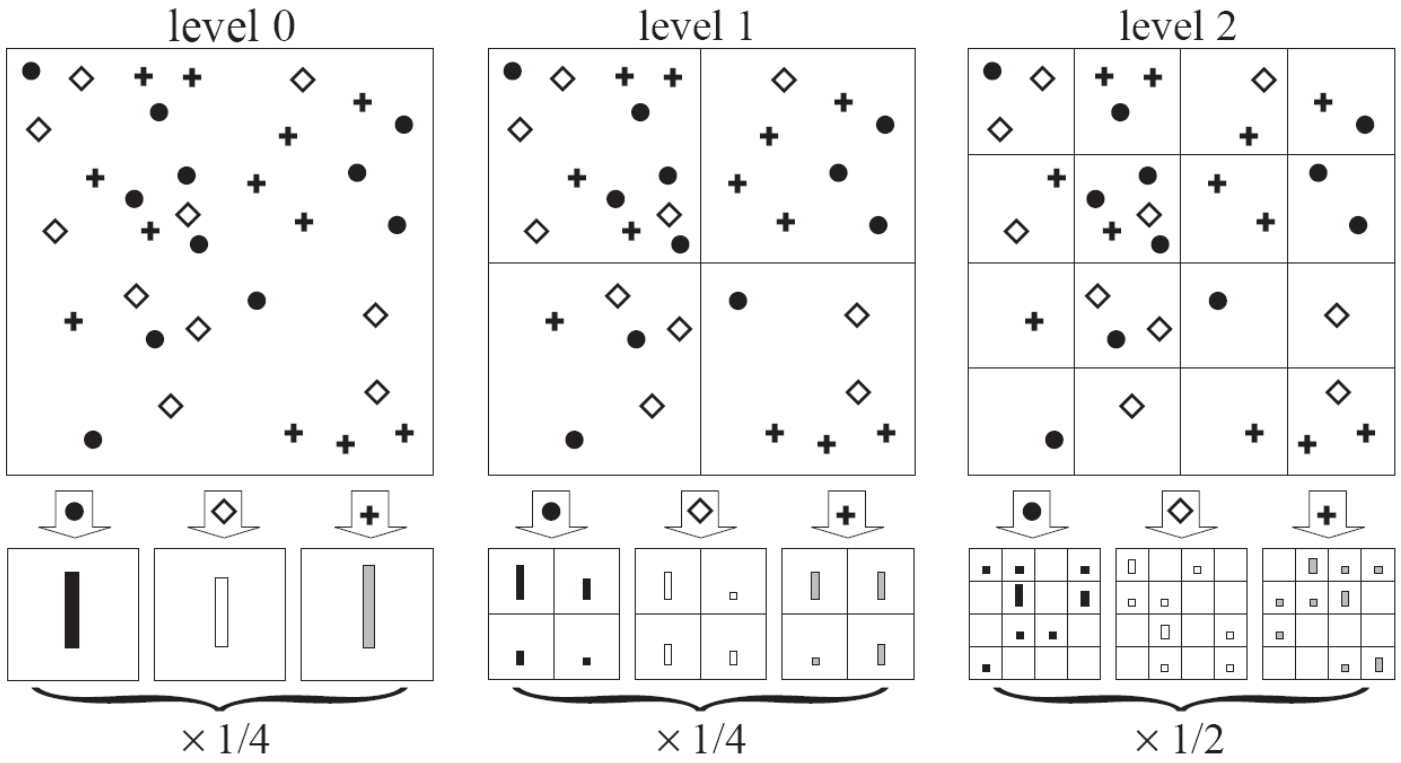
$$K(h_1, h_2) = \exp\left(-\frac{1}{A} D(h_1, h_2)^2\right)$$

- *D* can be Euclidean distance, $\chi^2$ distance etc…

J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid, Local Features and Kernels for Classifcation of Texture and Object Categories: A Comprehensive Study, IJCV 2007

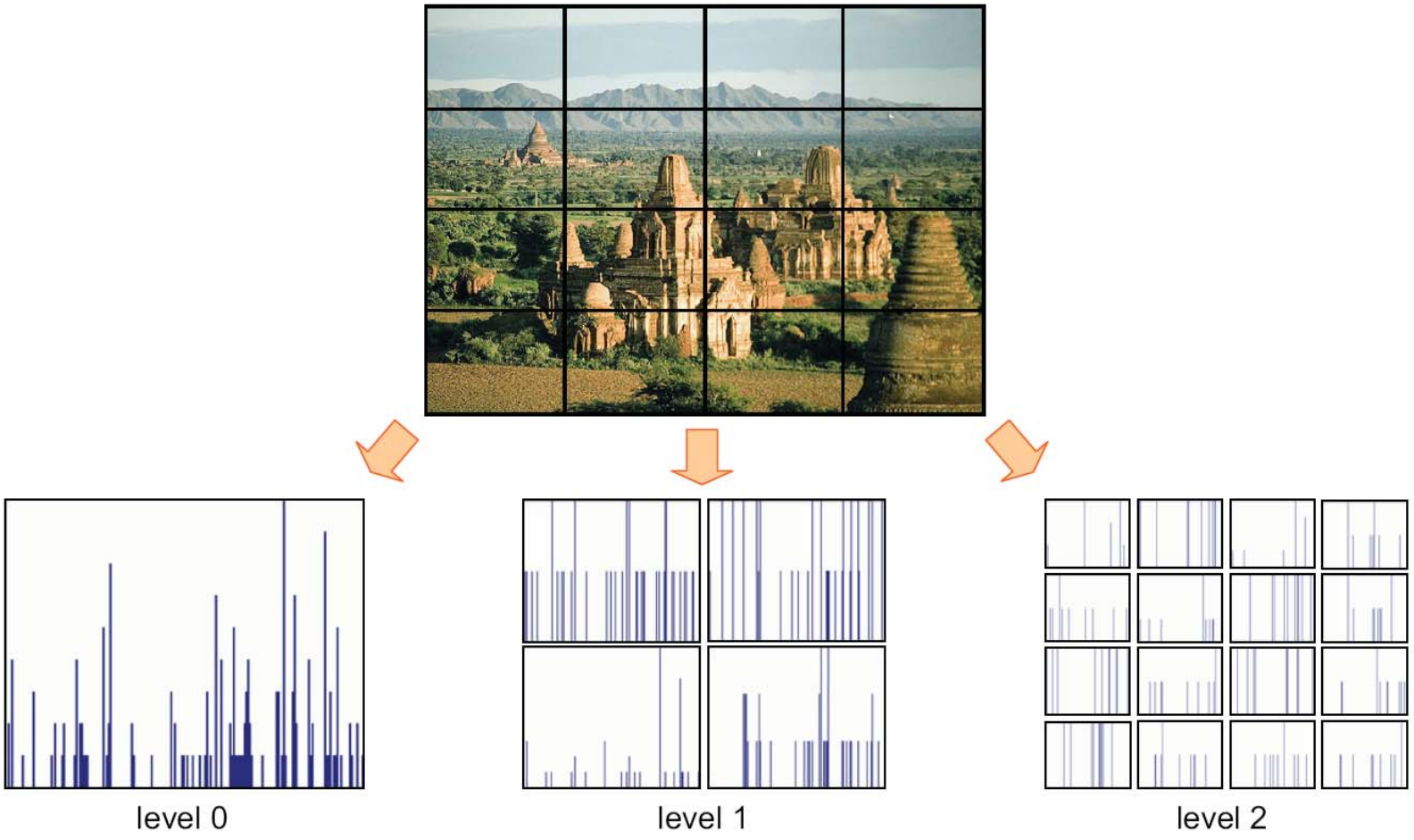
# Pyramid match kernel

- Fast approximation of Earth Mover's Distance
- Weighted sum of histogram intersections at mutliple resolutions (linear in the number of features instead of cubic)




K. Grauman and T. Darrell. The Pyramid Match Kernel: Discriminative Classification with Sets of Image Features, ICCV 2005.

# Spatial Pyramid Matching




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

# What about multi-class SVMs?

- No “definitive” multi-class SVM formulation
- In practice, we have to obtain a multi-class SVM by combining multiple two-class SVMs
- One vs. others
  - Traning: learn an SVM for each class vs. the others
  - Testing: apply each SVM to test example and assign to it the class of the SVM that returns the highest decision value
- One vs. one
  - Training: learn an SVM for each pair of classes
  - Testing: each learned SVM “votes” for a class to assign to the test example

Credit slide: S. Lazebnik

# SVMs: Pros and cons

- Pros
  - Many publicly available SVM packages: http://www.kernel-machines.org/software
  - Kernel-based framework is very powerful, flexible
  - SVMs work very well in practice, even with very small training sample sizes

- Cons
  - No "direct" multi-class SVM, must combine two-class SVMs
  - Computation, memory
    - During training time, must compute matrix of kernel values for every pair of examples
    - Learning can take a very long time for large-scale problems

# Object recognition results

- ETH-80 database of 8 object classes
  (*Eichhorn and Chapelle 2004)*
- Features:
  - Harris detector
  - PCA-SIFT descriptor, *d*=10

| Kernel | Complexity | Recognition rate |
|---|---|---|
| Match *[Wallraven et al.]* | $O(dm^2)$ | 84% |
| Bhattacharyya affinity *[Kondor & Jebara]* | $O(dm^3)$ | 85% |
| Pyramid match | $O(dmL)$ | 84% |

Slide credit: Kristen Grauman

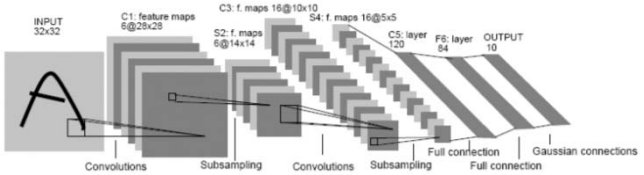
# Discriminative models

**Nearest neighbor**

$10^6$ examples

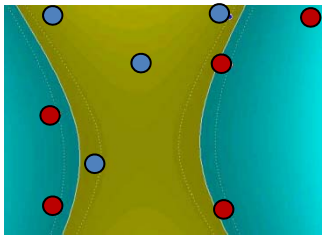
Shakhnarovich, Viola, Darrell 2003
Berg, Berg, Malik 2005...

**Neural networks**




LeCun, Bottou, Bengio, Haffner 1998
Rowley, Baluja, Kanade 1998
…

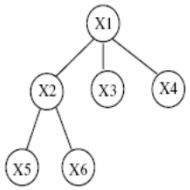
**Support Vector Machines**



Guyon, Vapnik, Heisele, Serre, Poggio…

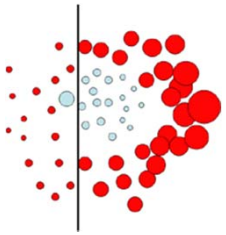
**Latent SVM**

**Structural SVM**




Felzenszwalb 00
Ramanan 03…

**Boosting**



Viola, Jones 2001,
Torralba et al. 2004,
Opelt et al. 2006,…

Source: Vittorio Ferrari, Kristen Grauman, Antonio Torralba

# Learning and Recognition

1. Discriminative method:
   - NN
   - SVM

2. Generative method:
   - graphical models

→ Model the probability distribution that produces a given bag of features

# Generative models

1. Naïve Bayes classifier
   - Csurka Bray, Dance & Fan, 2004

2. Hierarchical Bayesian text models (pLSA and LDA)
   - Background: Hoffman 2001, Blei, Ng & Jordan, 2004
   - Object categorization: Sivic et al. 2005, Sudderth et al. 2005
   - Natural scene categorization: Fei-Fei et al. 2005

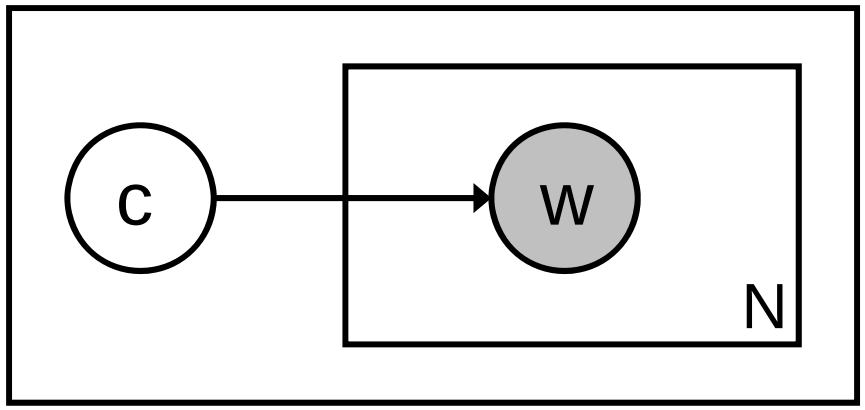
# Some notations

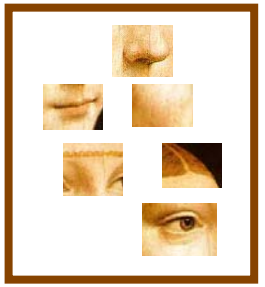
- **w:** a collection of all N codewords in the image

  $\mathbf{w} = [w_1, w_2, \ldots, w_N]$

- c: category of the image

# the Naïve Bayes model



Graphical model

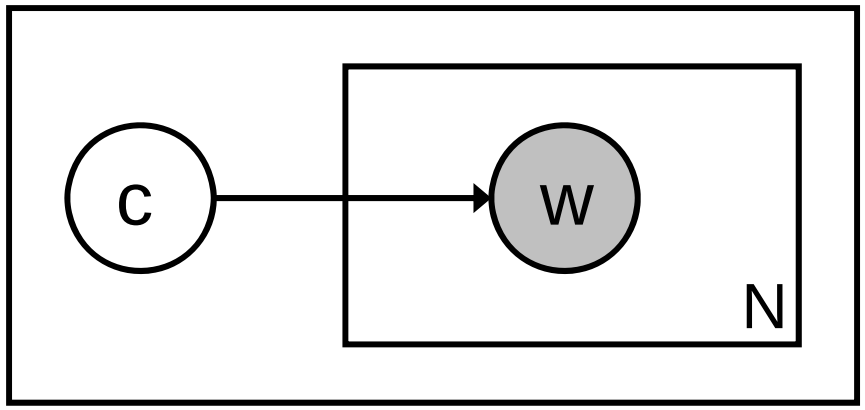
**Posterior** = probability that image **I** is of category c

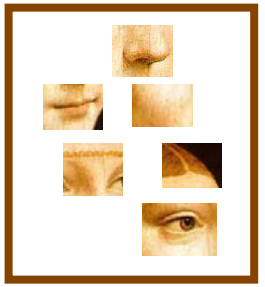
$$p(c \mid w) \propto p(c)\, p(w \mid c)$$

Prior prob. of the object classes

Image likelihood given the class

# the Naïve Bayes model



Graphical model

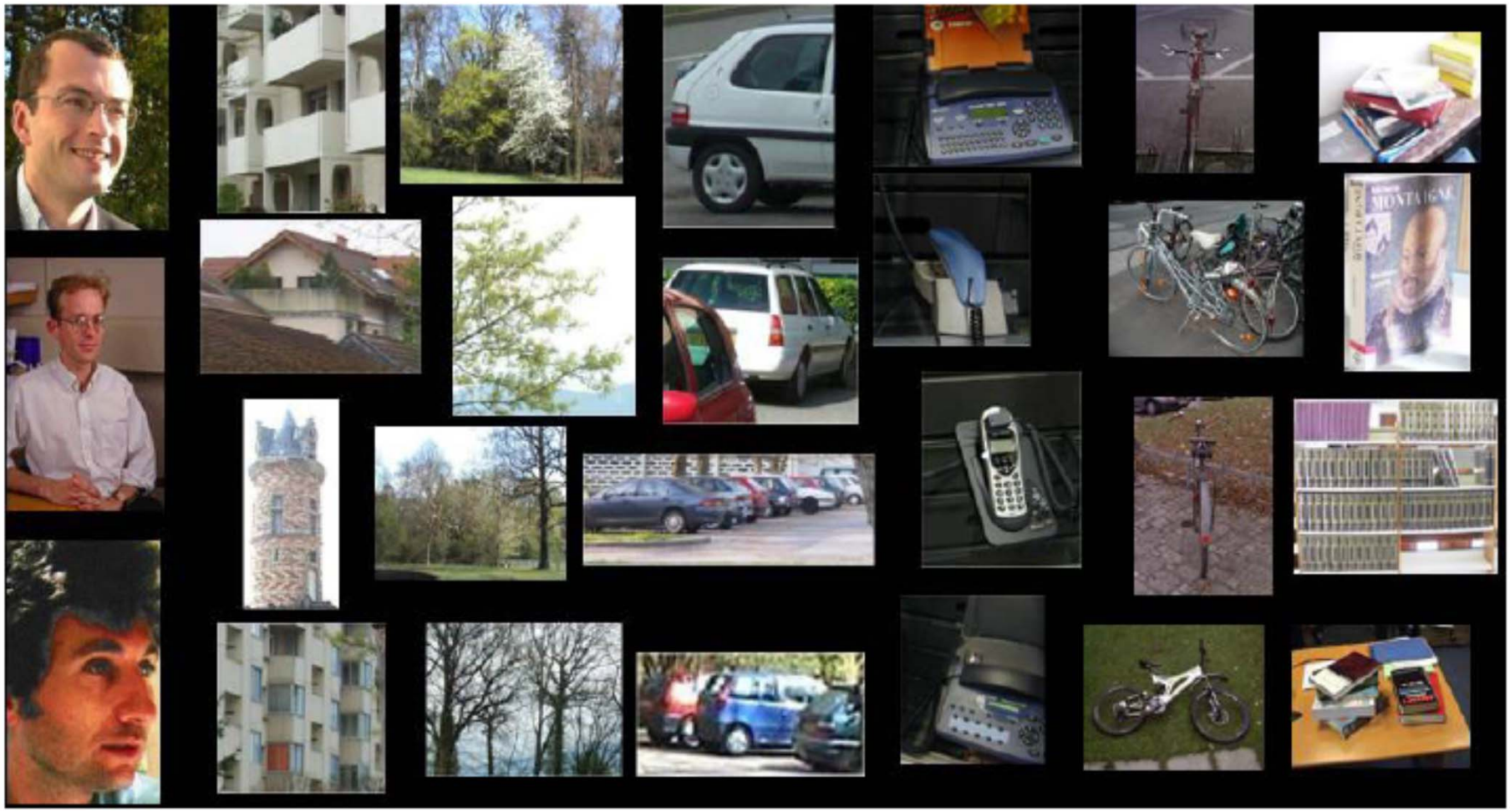
$$c^* = \arg\max_c \; p(c \mid w) \propto p(c)\, p(w \mid c) = p(c) \prod_{n=1}^{N} p(w_n \mid c)$$

Object class decision

Likelihood of ith visual word given the class

Estimated by empirical frequencies of code words in images from a given class

Our in-house database contains 1776 images in seven classes[1]: faces, buildings, trees, cars, phones, bikes and books. Fig. 2 shows some examples from this dataset.



Csurka et al. 2004

**Table 1.** Confusion matrix and the mean rank for the best vocabulary (*k=1000*).

| True classes → | *faces* | *buildings* | *trees* | *cars* | *phones* | *bikes* | *books* |
|---|---|---|---|---|---|---|---|
| ***faces*** | **76** | 4 | 2 | 3 | 4 | 4 | 13 |
| ***buildings*** | 2 | **44** | 5 | 0 | 5 | 1 | 3 |
| ***trees*** | 3 | 2 | **80** | 0 | 0 | 5 | 0 |
| ***cars*** | 4 | 1 | 0 | **75** | 3 | 1 | 4 |
| ***phones*** | 9 | 15 | 1 | 16 | **70** | 14 | 11 |
| ***bikes*** | 2 | 15 | 12 | 0 | 8 | **73** | 0 |
| ***books*** | 4 | 19 | 0 | 6 | 7 | 2 | **69** |
| ***Mean ranks*** | 1.49 | 1.88 | 1.33 | 1.33 | 1.63 | 1.57 | 1.57 |

Csurka et al. 2004

# Other generative BoW models

- Hierarchical Bayesian topic models (e.g. pLSA and LDA)
  - Object categorization: Sivic et al. 2005, Sudderth et al. 2005
  - Natural scene categorization: Fei-Fei et al. 2005

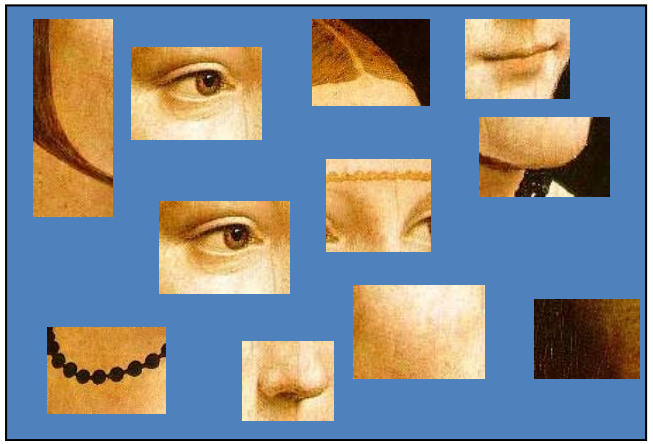
# Generative vs discriminative

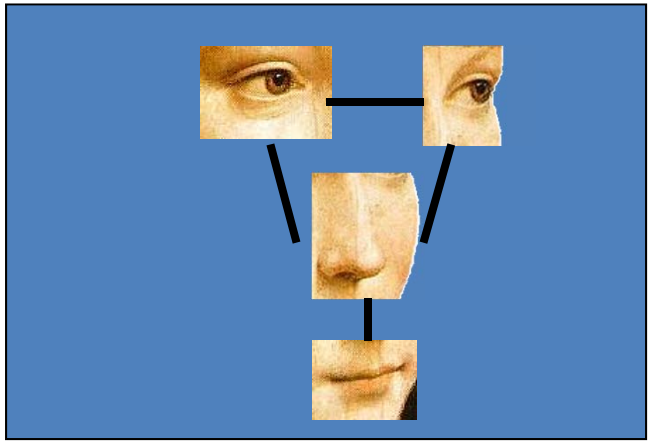
- Discriminative methods
  - Computationally efficient & fast

- Generative models
  - Convenient for weakly- or un-supervised, incremental training
  - Prior information
  - Flexibility in modeling parameters

# Weakness of BoW the models

- No rigorous geometric information of the object components
- It's intuitive to most of us that objects are made of parts – no such information
- Not extensively tested yet for
  - View point invariance
  - Scale invariance
- Segmentation and localization unclear





# What have learned today?

- Introduction to object recognition
  - Representation
  - Learning
  - Recognition
- Bag of Words models (Problem Set 4 (Q2))
  - Basic representation
  - Different learning and recognition algorithms