

Lecture 15: Object recognition: Bag of Words models & Part-based generative models

Professor Fei-Fei Li
Stanford Vision Lab

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

What we will learn today?

- Bag of Words model (**Problem Set 4 (Q2)**)
 - Basic representation
 - Different learning and recognition algorithms
- Constellation model
 - Weakly supervised training
 - One-shot learning (supplementary materials)
- (**Problem Set 4 (Q1)**)



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

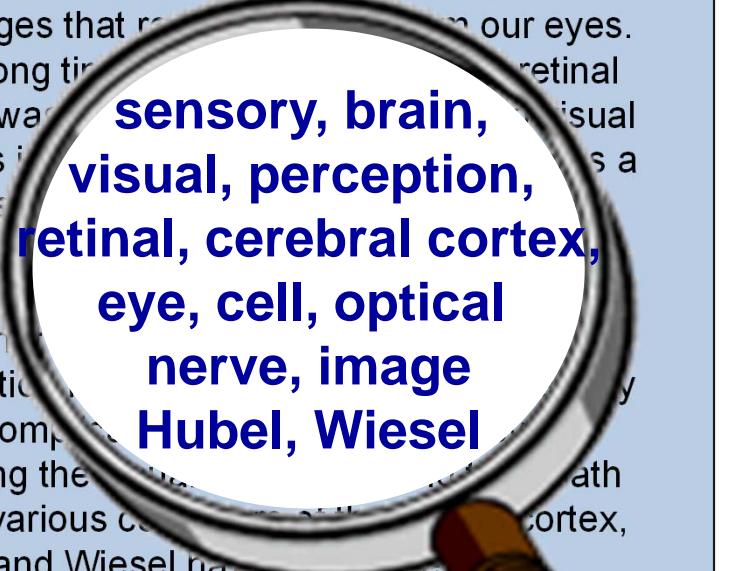
Object

Bag of ‘words’



Analogy to documents

Of all the sensory impressions proceeding to the brain, the visual experiences are the dominant ones. Our perception of the world around us is based essentially on the messages that reach us through our eyes. For a long time it was believed that the retinal image was processed by the visual centers in the brain. This was a movie screen analogy. In 1960, a retinal image was discovered to pass through the eye, cell, optical nerve, image Hubel, Wiesel following the optic nerve to the cerebral cortex, Hubel and Wiesel have demonstrated that the message about the image falling on the retina undergoes top-down analysis in a system of nerve cells stored in columns. In this system each column has its specific function and is responsible for a specific detail in the pattern of the retinal image.



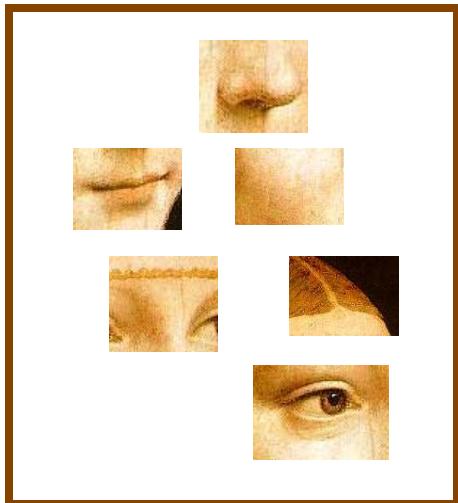
China is forecasting a trade surplus of \$90bn (£51bn) to \$100bn this year, a threefold increase on 2004's \$32bn. The Commerce Ministry said the surplus would be created by a predicted 30% increase in exports to \$750bn, compared with \$660bn. The Chinese government annoy the US by deliberately keeping the yuan at a low level. China's central bank, the People's Bank, has agreed to let the value of the yuan rise. The Chinese government also needs to encourage domestic demand so that the economy can grow in the country. China has been allowed to let the yuan against the dollar rise in value and permitted it to trade within a narrow band, but the US wants the yuan to be allowed to trade freely. However, Beijing has made it clear that it will take its time and tread carefully before allowing the yuan to rise further in value.



definition of “BoW”

– Independent features

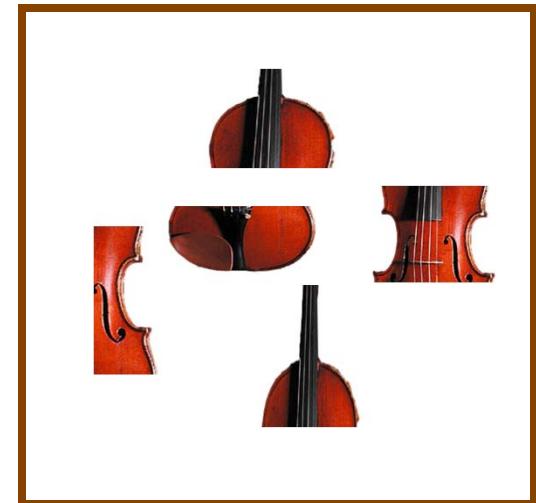
face



bike

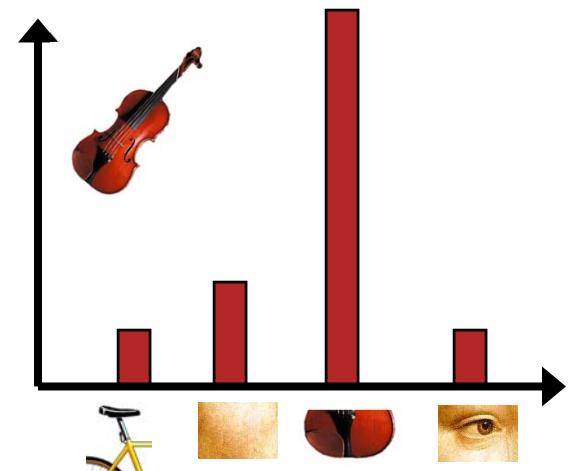
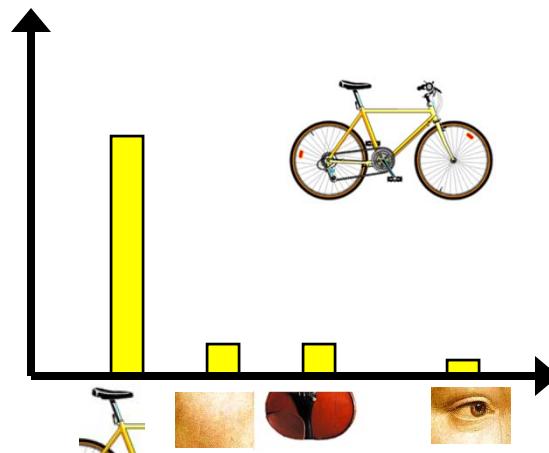
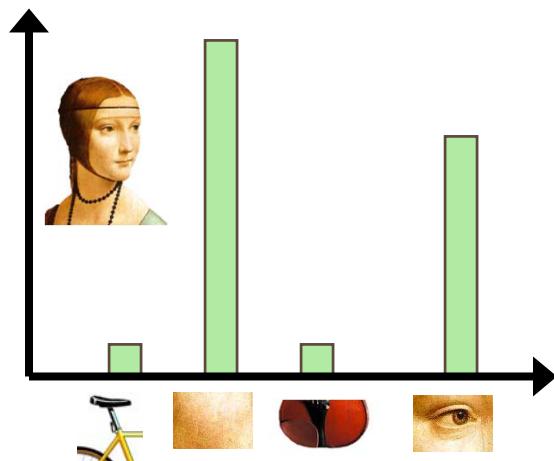


violin



definition of “BoW”

- Independent features
- histogram representation



codewords dictionary

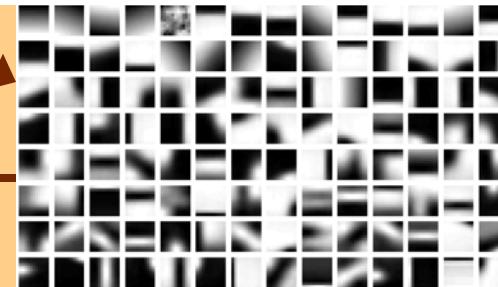
Representation

recognition

feature detection
& representation

codewords dictionary

image representation



learning

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

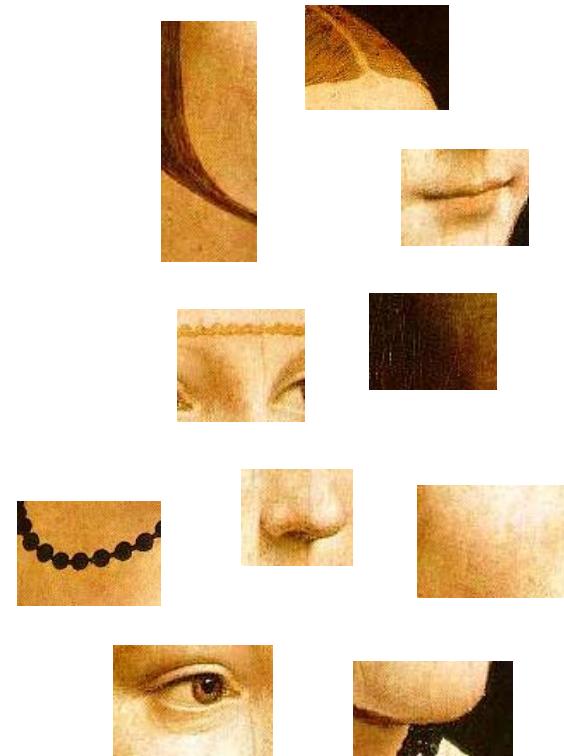
Fei-Fei Li

10

**category
decision**

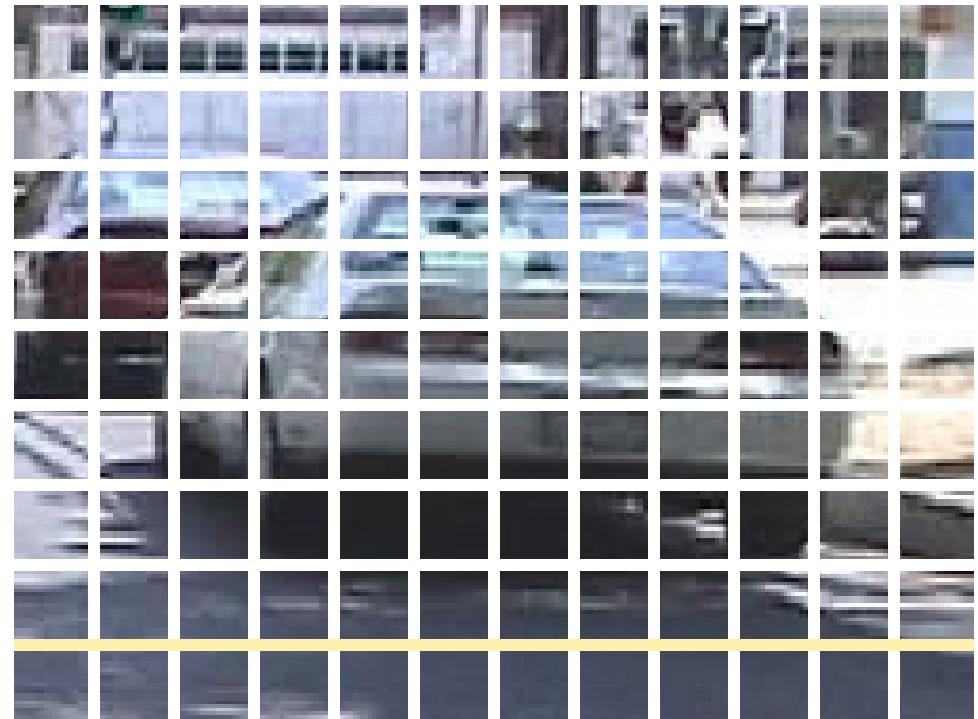
14-Nov-11

1. Feature detection and representation



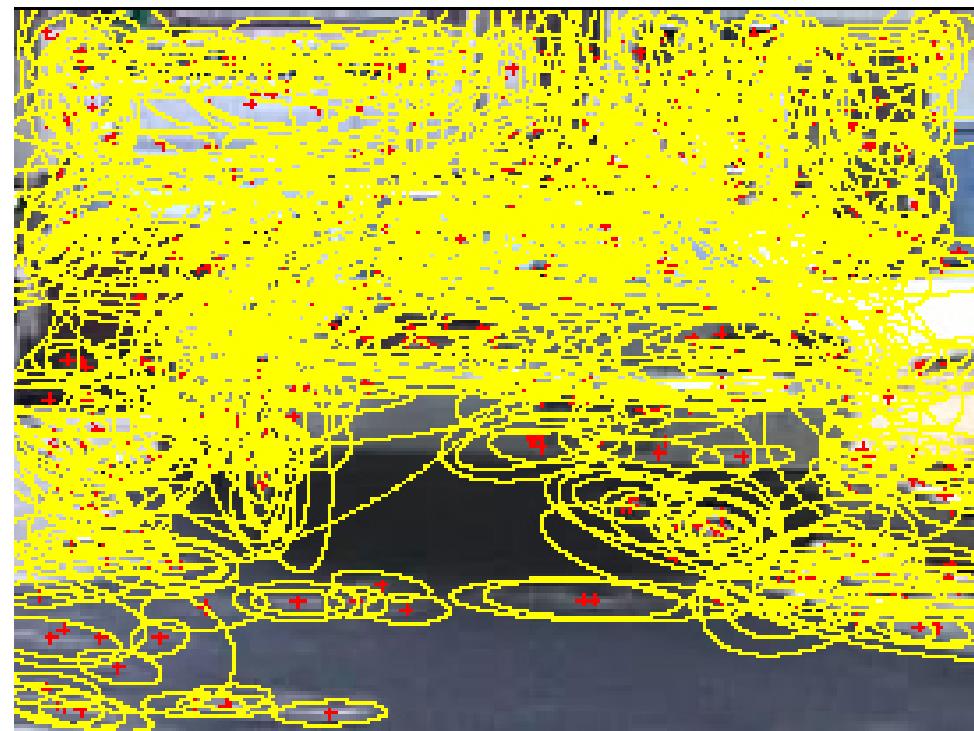
1. Feature detection and representation

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



1. Feature detection and representation

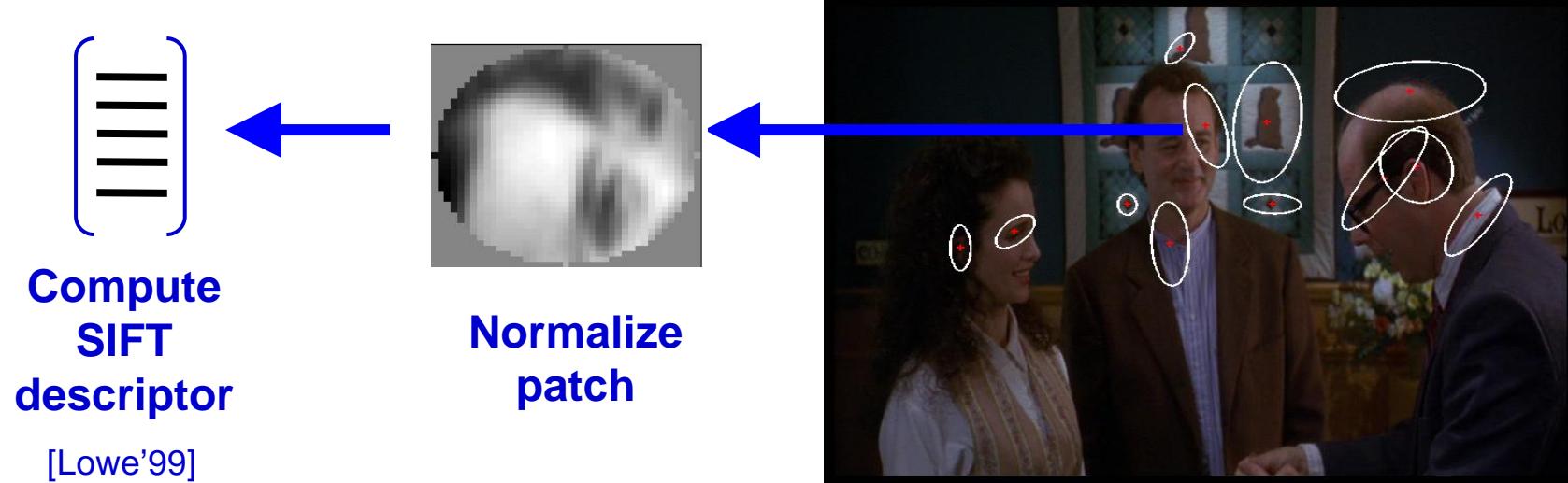
- Regular grid
 - Vogel & Schiele, 2003
 - Fei-Fei & Perona, 2005
- Interest point detector
 - Csurka, et al. 2004
 - Fei-Fei & Perona, 2005
 - Sivic, et al. 2005



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



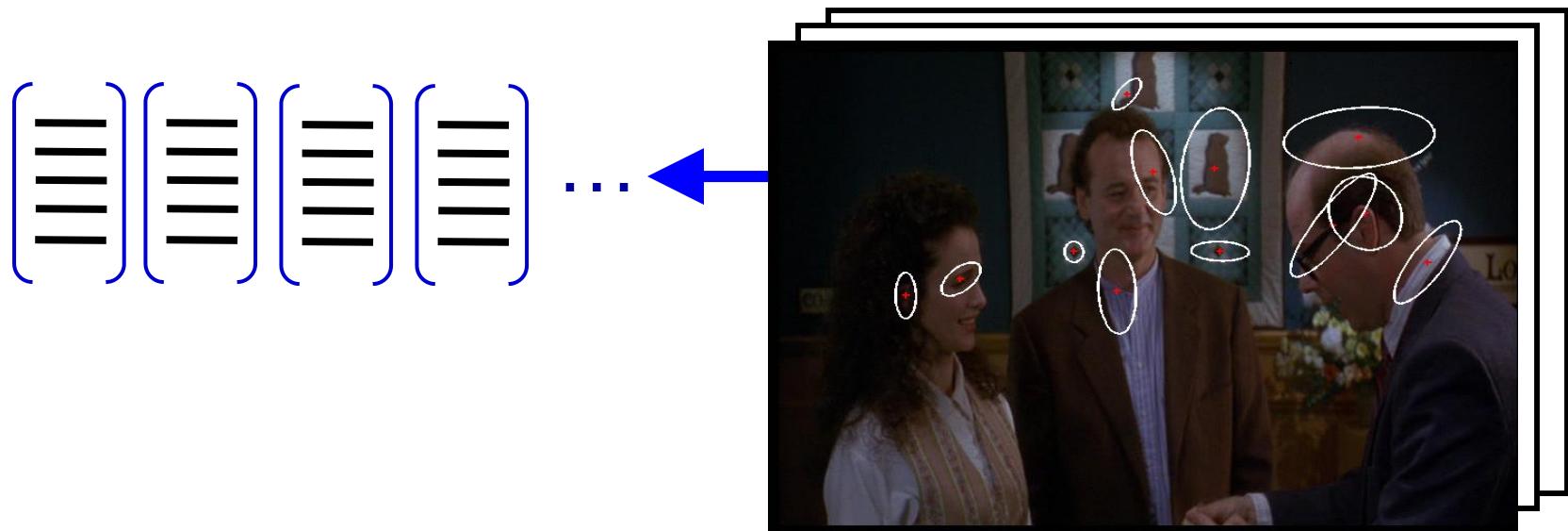
[Mikojaczyk and Schmid '02]

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

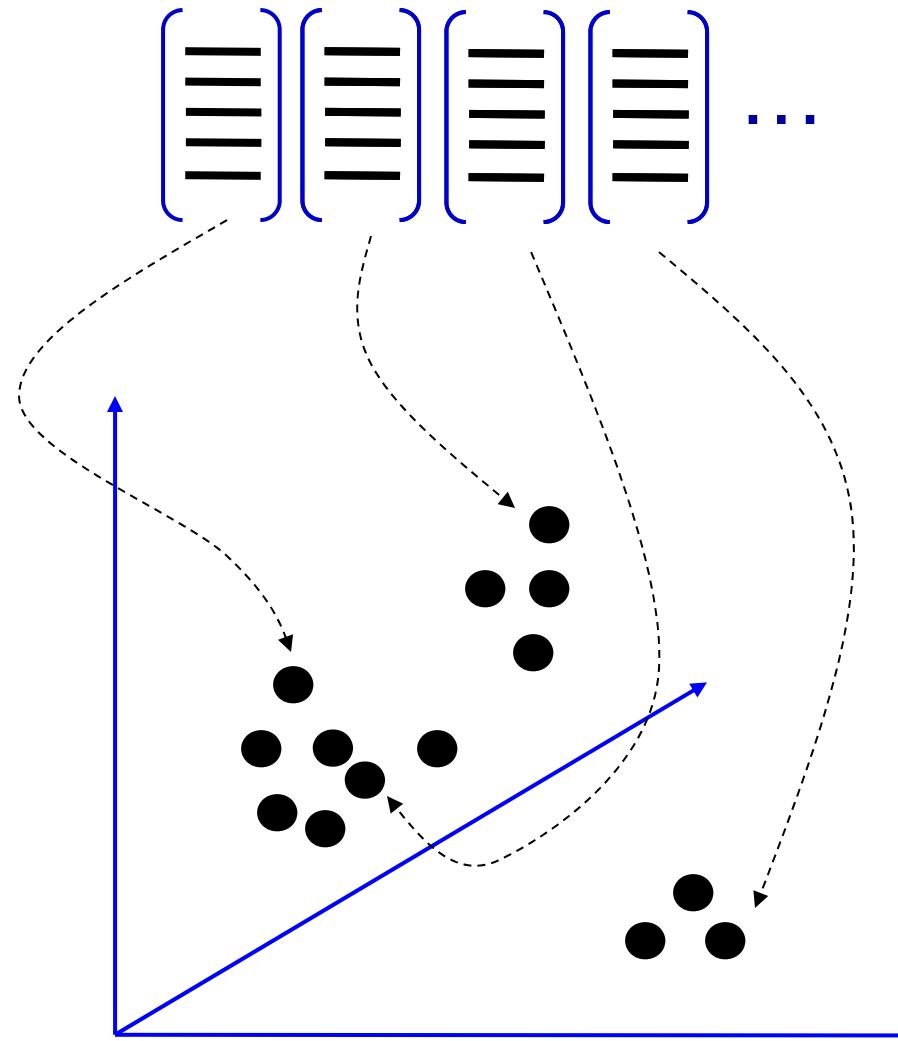
[Sivic & Zisserman, '03]

Slide credit: Josef Sivic

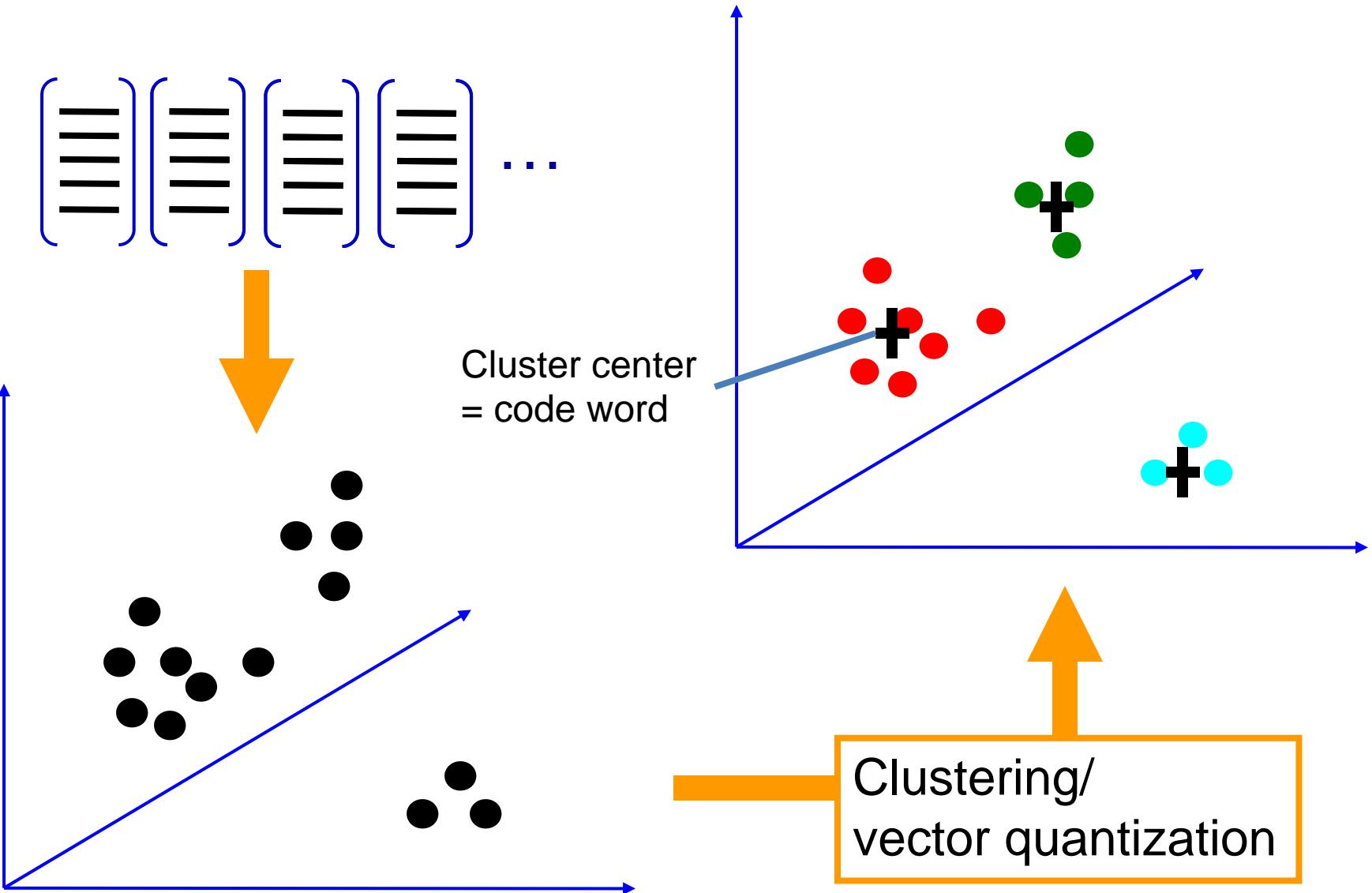
1. Feature detection and representation



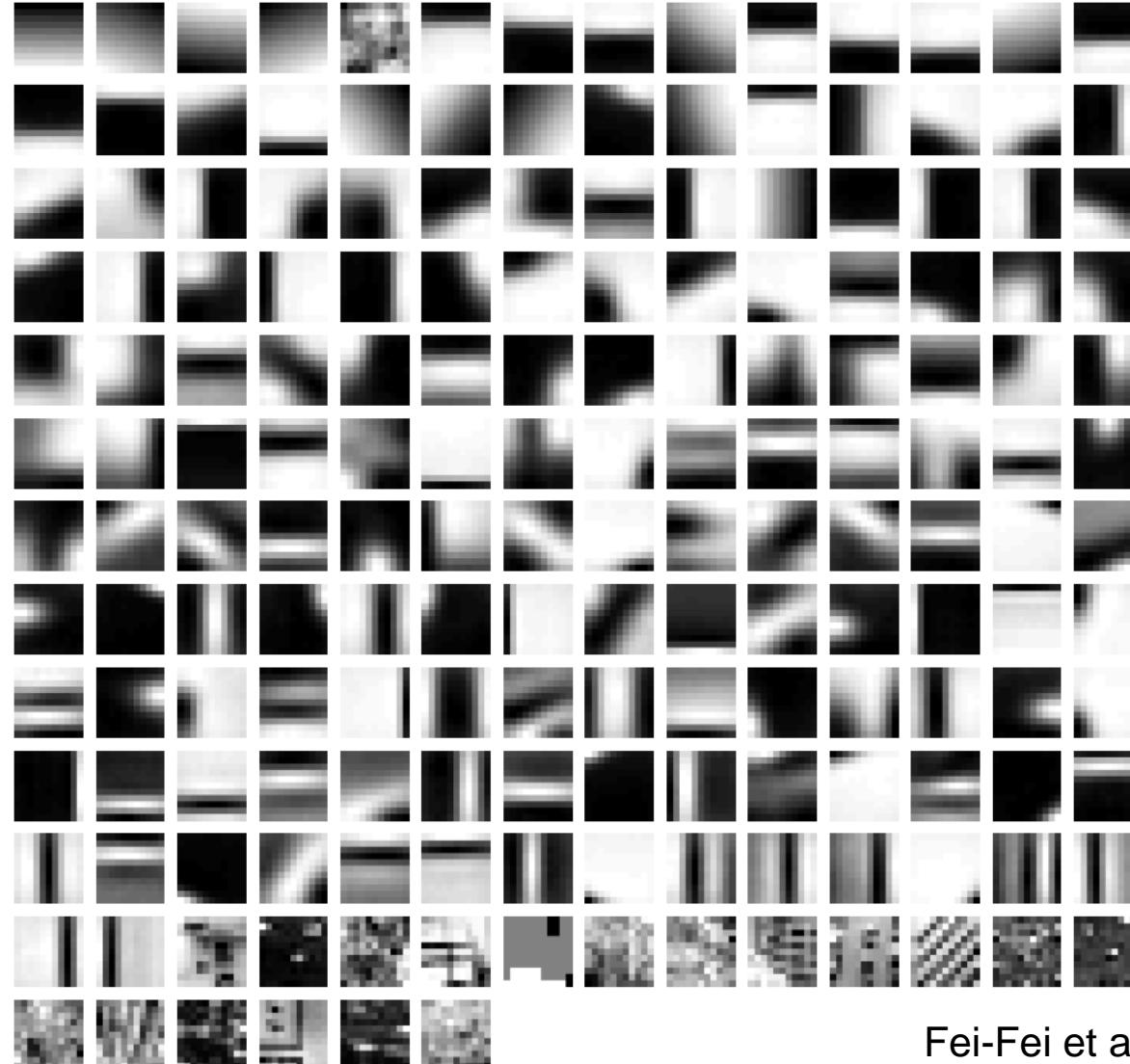
2. Codewords dictionary formation



2. Codewords dictionary formation

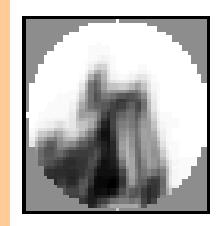
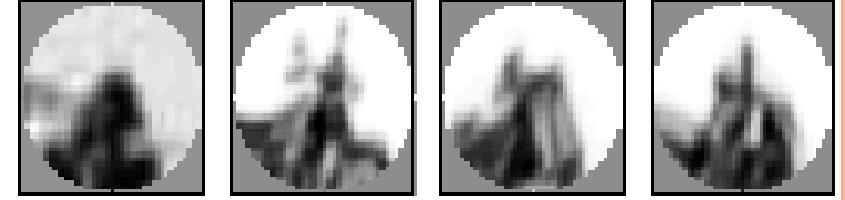
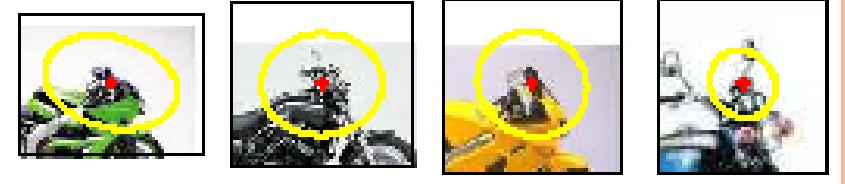
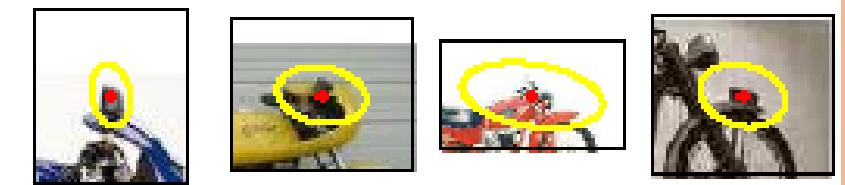
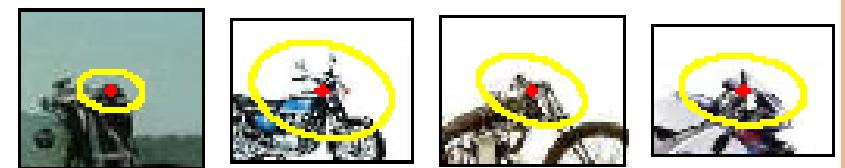
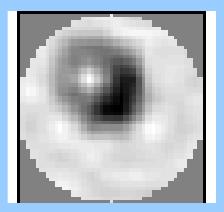
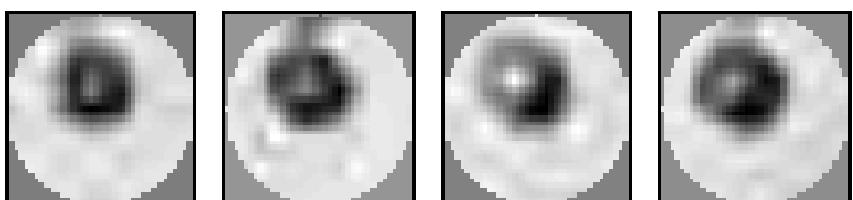
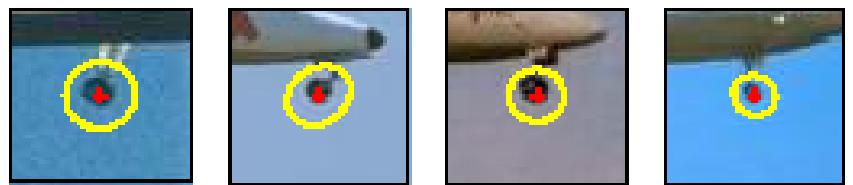


2. Codewords dictionary formation



Fei-Fei et al. 2005

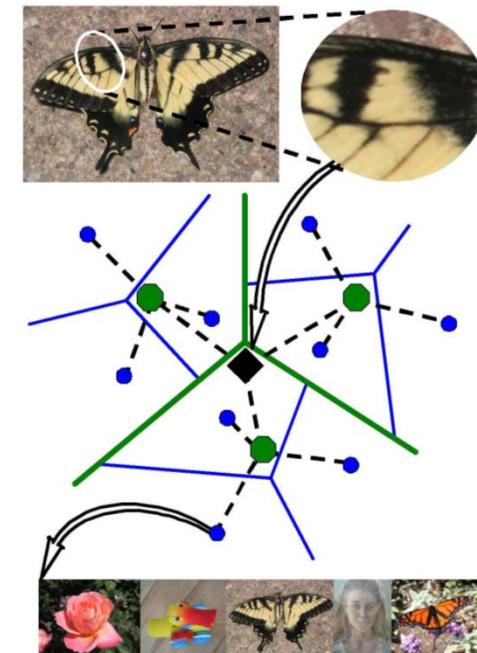
Image patch examples of codewords



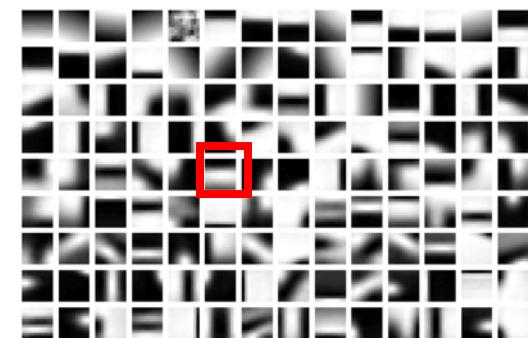
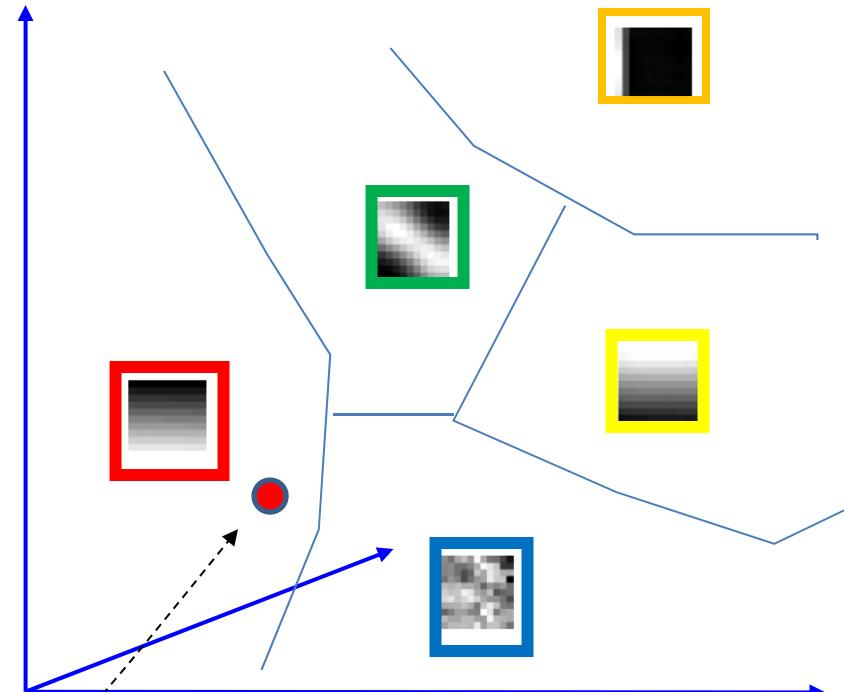
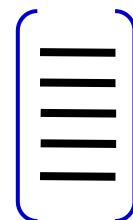
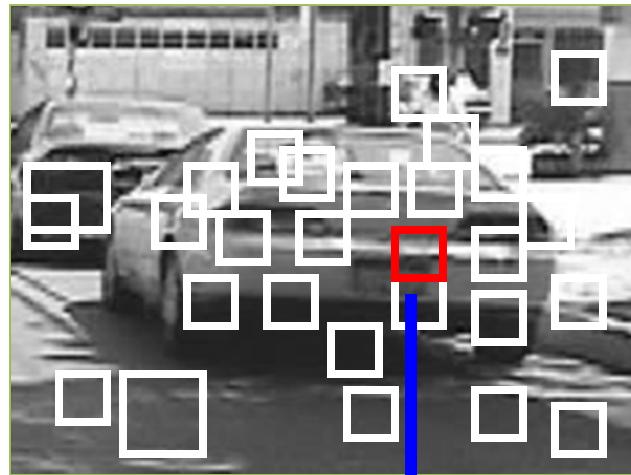
Sivic et al. 2005

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)



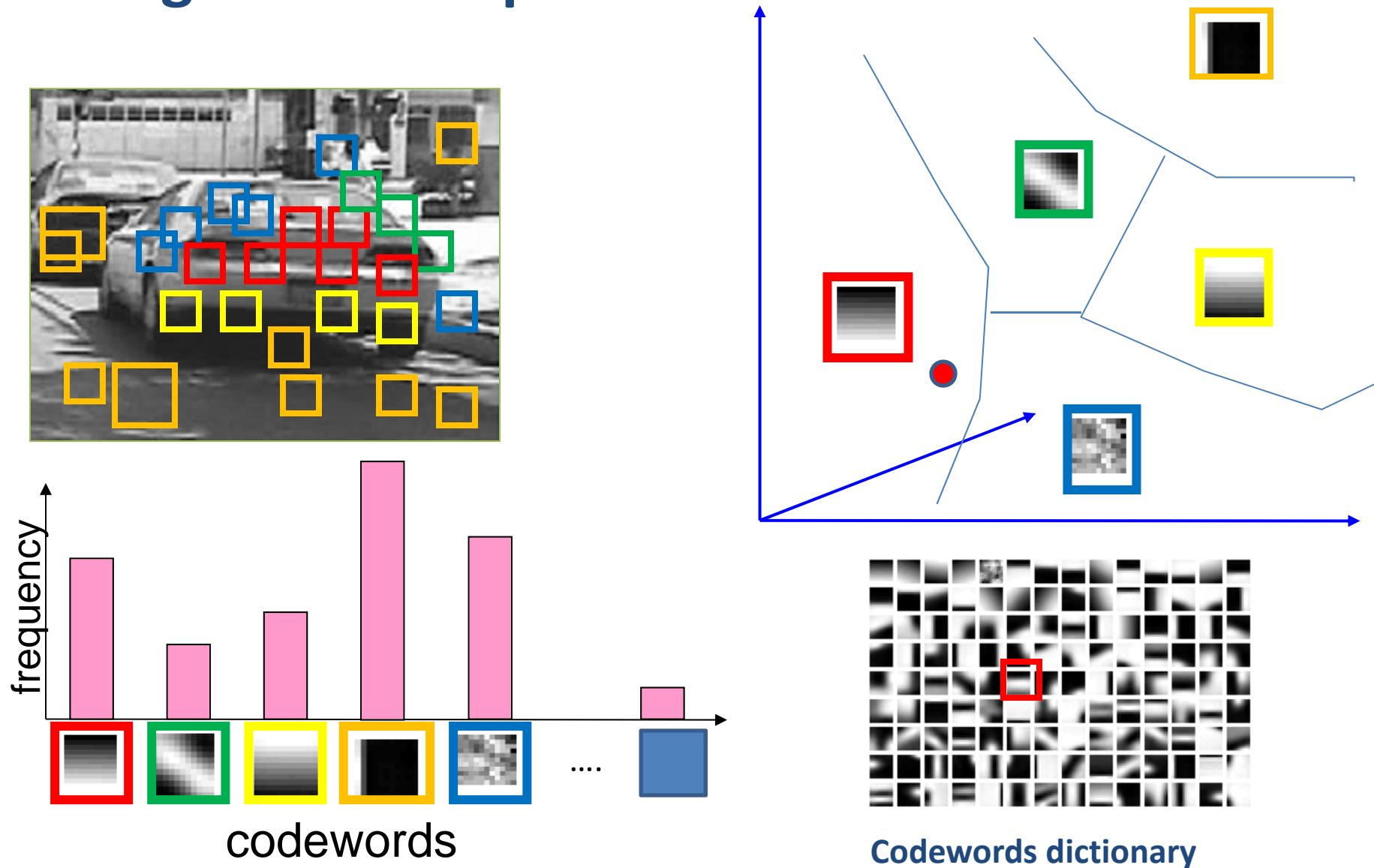
3. Bag of word representation



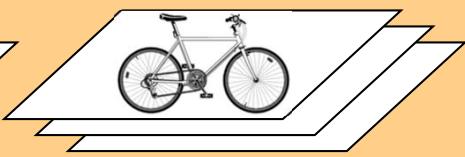
Codewords dictionary

- Nearest neighbors assignment
- K-D tree search strategy

3. Bag of word representation



Representation



1. feature detection & representation

2. codewords dictionary

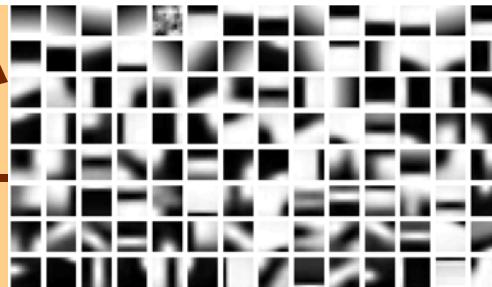
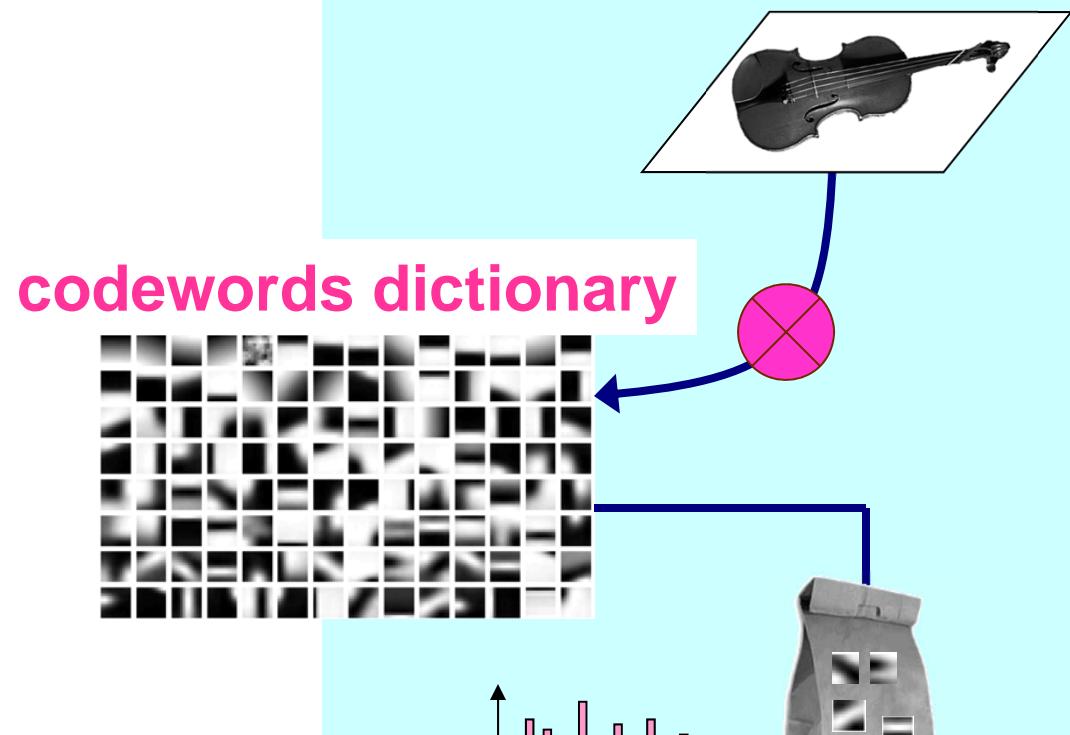


image representation

- 3.



Learning and Recognition



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

Fei-Fei Li

**category
decision**

25

14-Nov-11

Learning and Recognition

1. Discriminative method:

- NN
- SVM

2. Generative method:

- graphical models

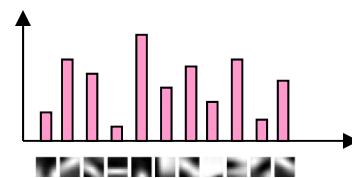
**category models
(and/or) classifiers**

Discriminative classifiers

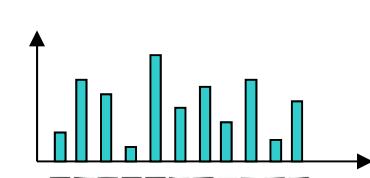
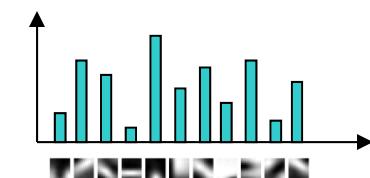
category models



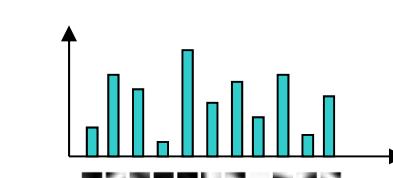
⋮



Class 1

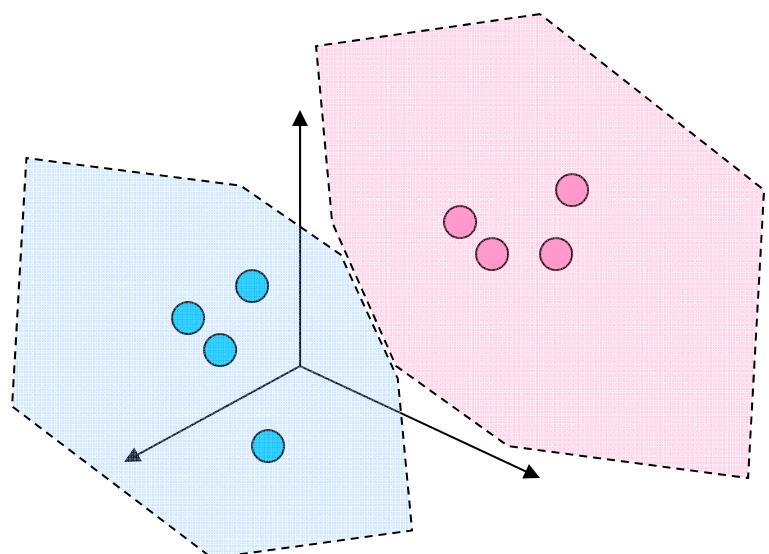


⋮



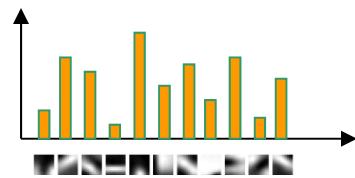
Class N

Model space



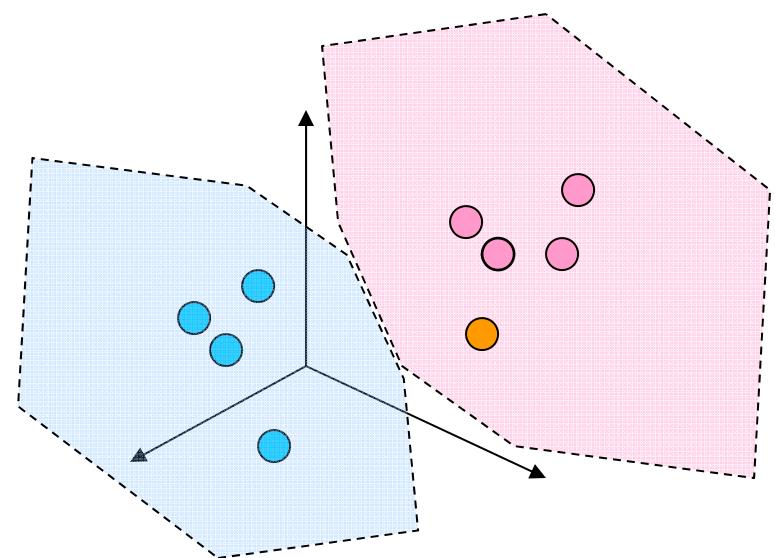
Discriminative classifiers

Query image



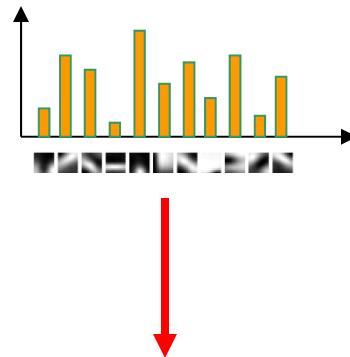
Winning class: pink

Model space



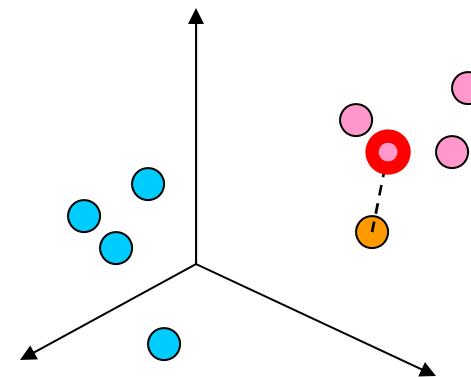
Nearest Neighbors classifier

Query image



Winning class: pink

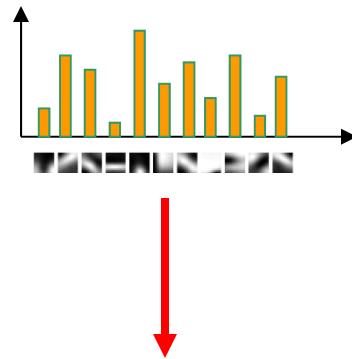
Model space



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

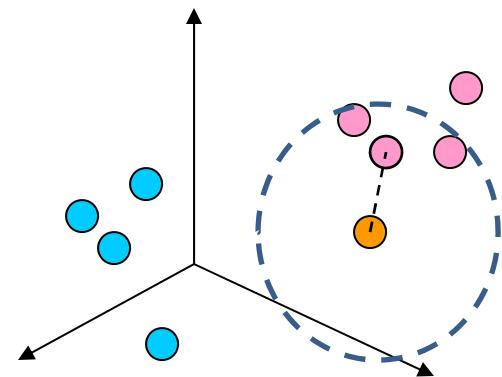
K- Nearest Neighbors classifier

Query image



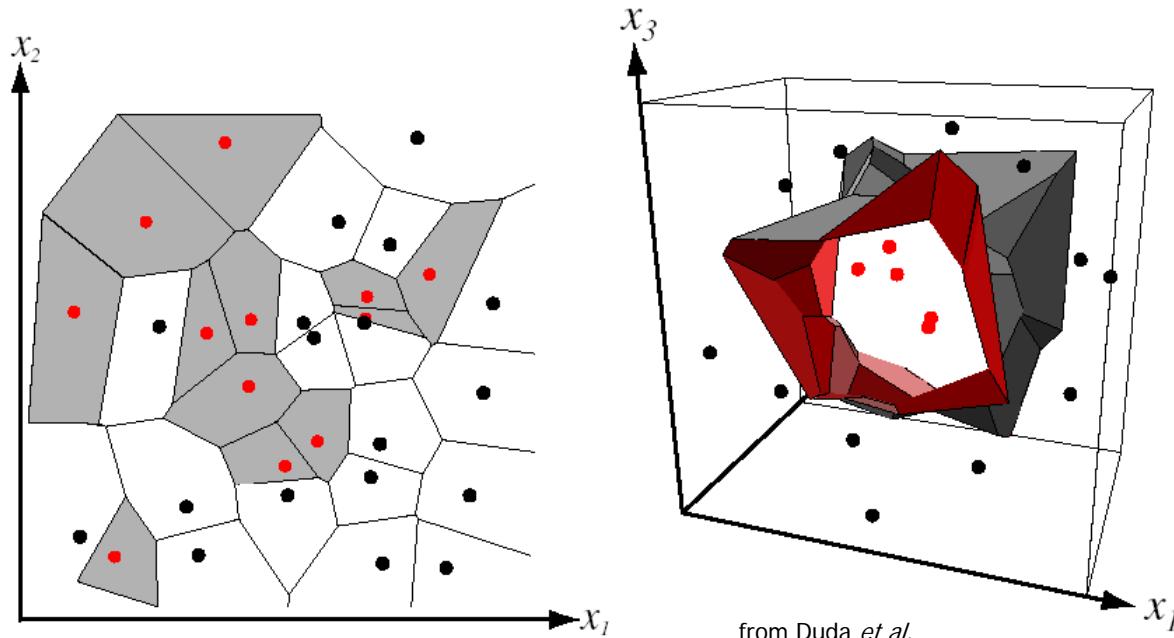
Winning class: pink

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



- 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

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

- χ^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:

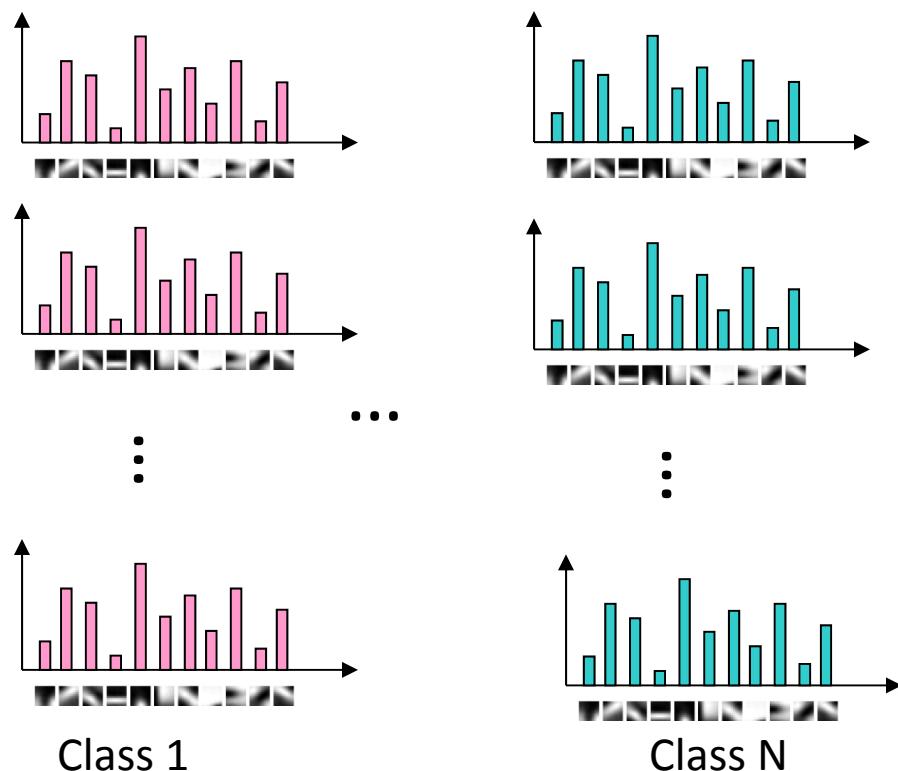
- NN
- SVM

2. Generative method:

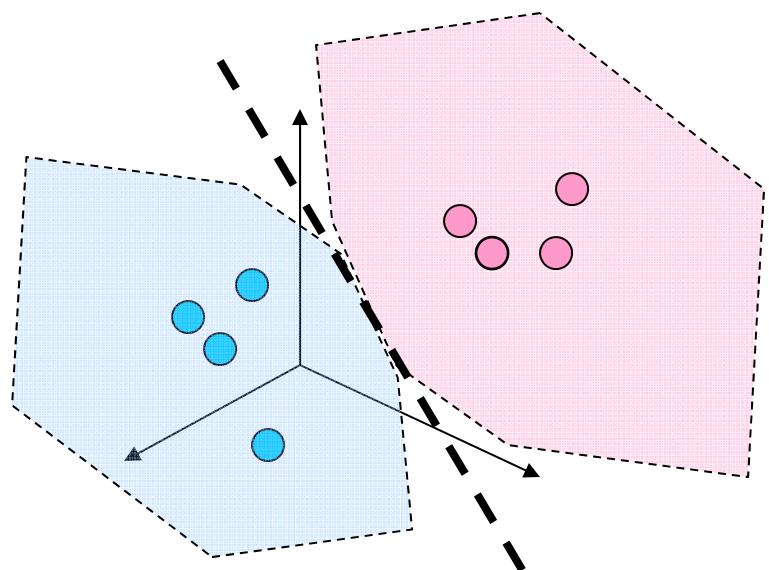
- graphical models

Discriminative classifiers (linear classifier)

category models

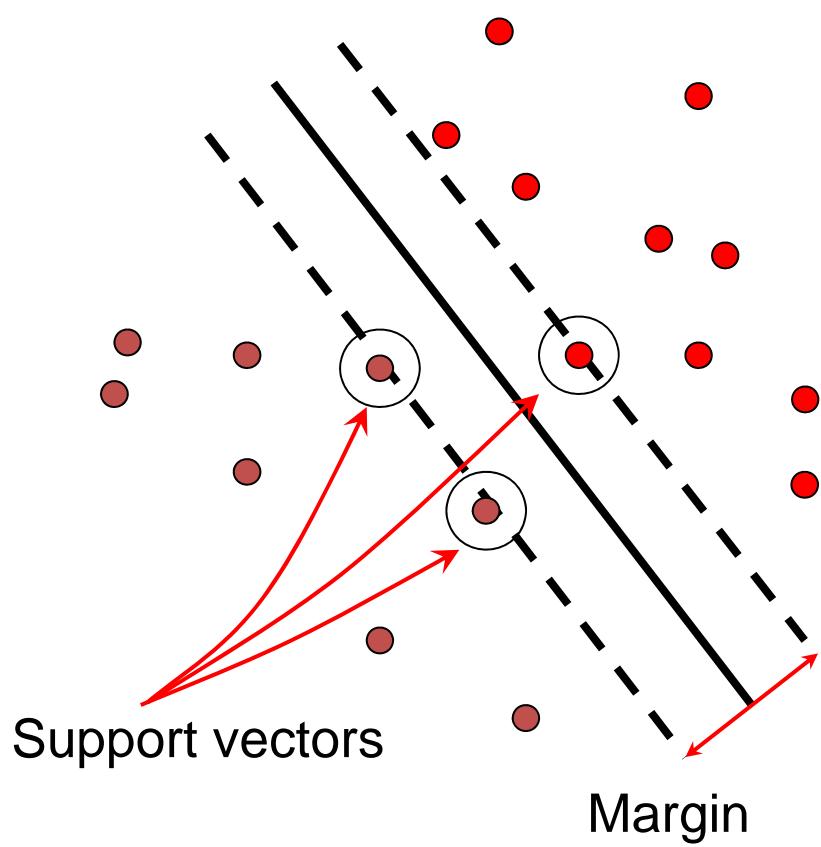


Model space



Support vector machines

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



Credit slide: S. Lazebnik

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

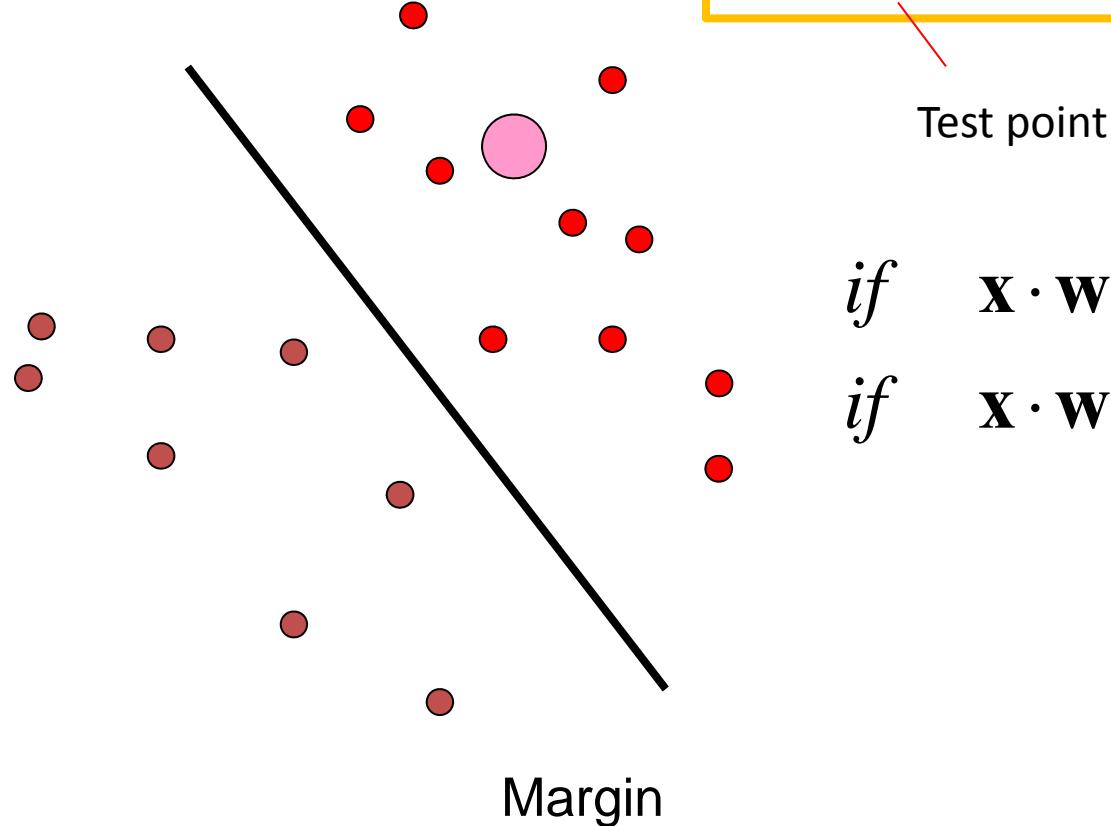
Solution:
$$\mathbf{w} = \sum_i \alpha_i y_i \mathbf{x}_i$$

Classification function (decision boundary):

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

Support vector machines

- Classification



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

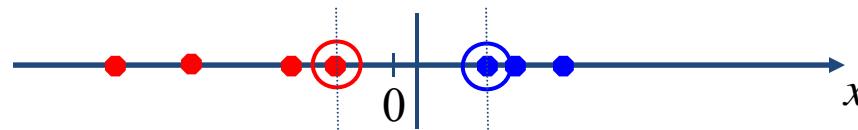
Test point

if $\mathbf{x} \cdot \mathbf{w} + b \geq 0 \rightarrow \text{class 1}$
if $\mathbf{x} \cdot \mathbf{w} + b < 0 \rightarrow \text{class 2}$

C. Burges, [A Tutorial on Support Vector Machines for Pattern Recognition](#), Data Mining and Knowledge Discovery, 1998

Nonlinear SVMs

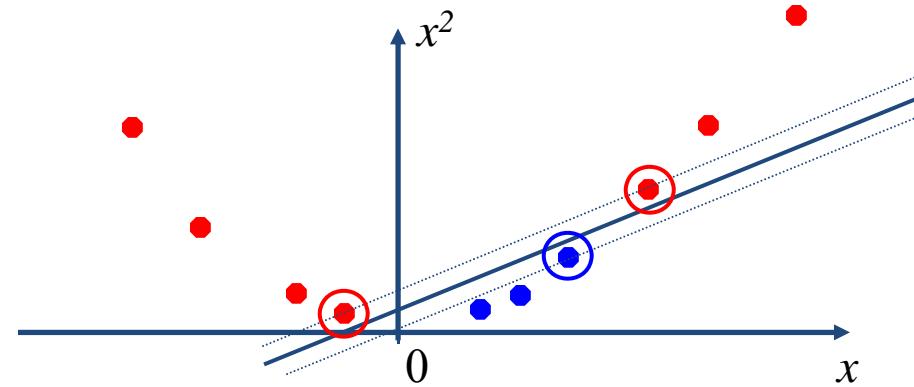
- Datasets that are linearly separable work out great:



- But what if the dataset is just too hard?



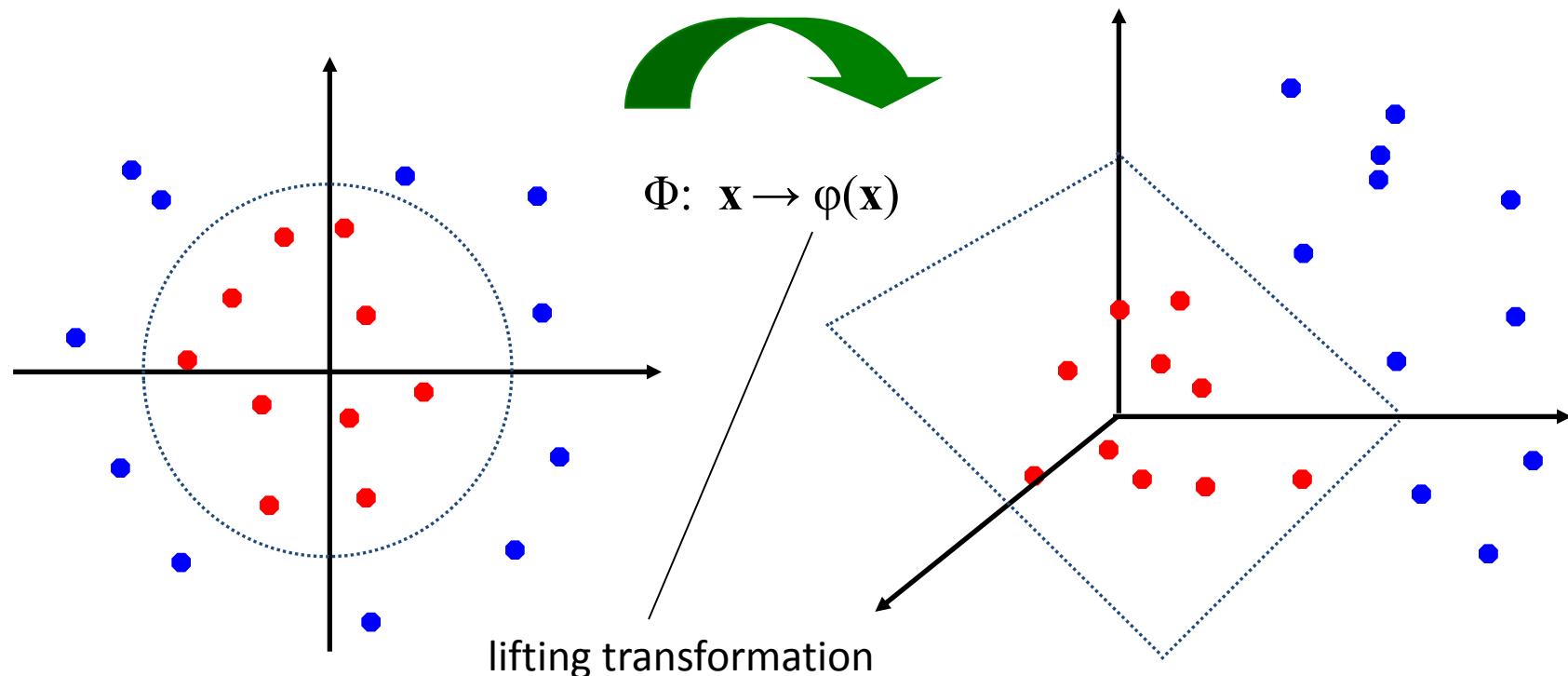
- We can map it to a higher-dimensional space:



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 $\varphi(\mathbf{x})$:*

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

NOTE:

- It is not required to compute $\varphi(\mathbf{x})$ 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:

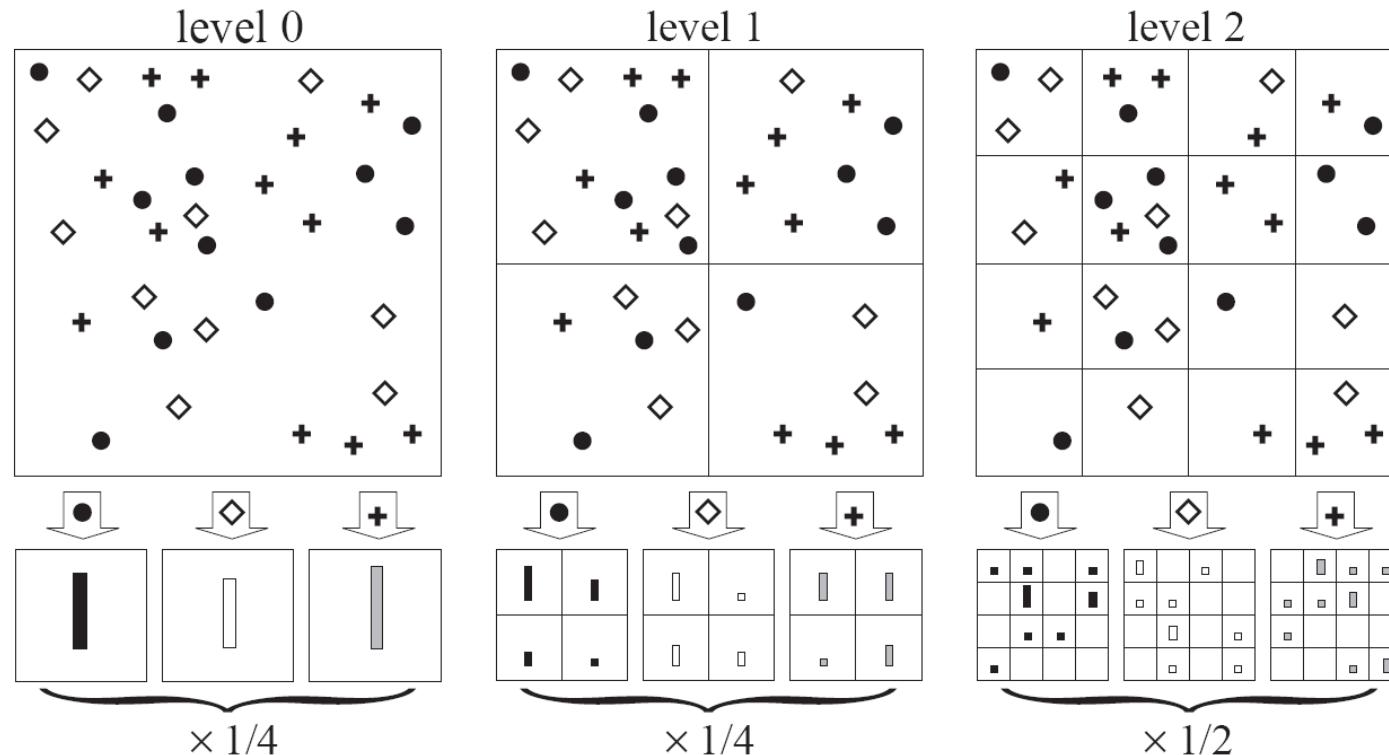
$$K(h_1, h_2) = \exp\left(-\frac{1}{A} D(h_1, h_2)^2\right)$$

- D can be Euclidean distance, χ^2 distance etc...

J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid, [Local Features and Kernels for Classification of Texture and Object Categories: A Comprehensive Study](#), IJCV 2007

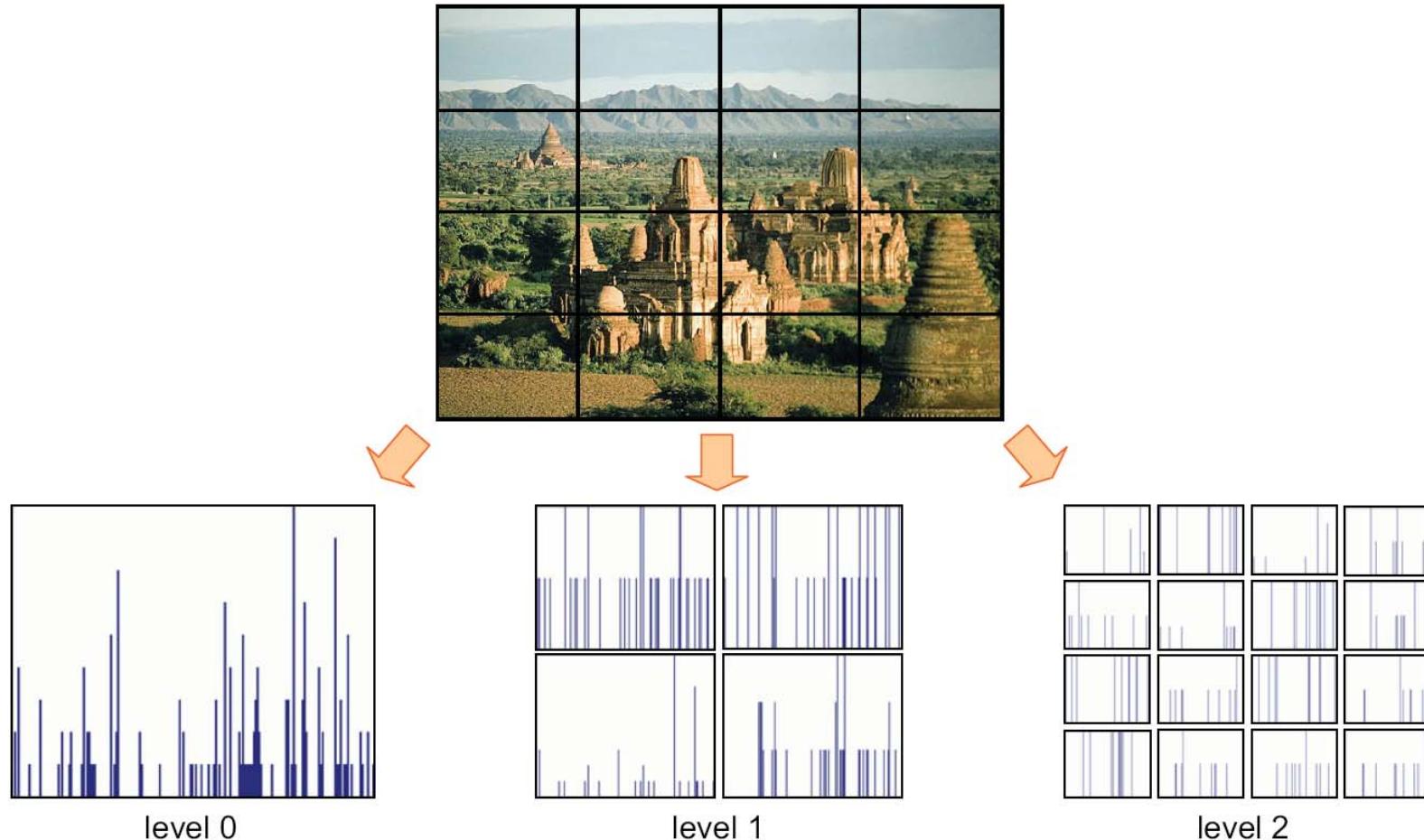
Pyramid match kernel

- Fast approximation of Earth Mover's Distance
- Weighted sum of histogram intersections at multiple 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
 - Training: 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, $d=10$

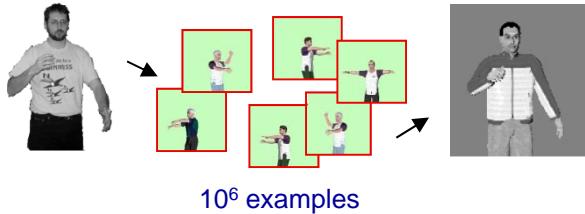


Kernel	Complexity	Recognition rate
Match [Wallraven et al.]	$O(dm^2)$	84%
Bhattacharyya affinity [Kondor & Jebara]	$O(dm^3)$	85%
Pyramid match	$O(dmL)$	84%

Slide credit: Kristen Grauman

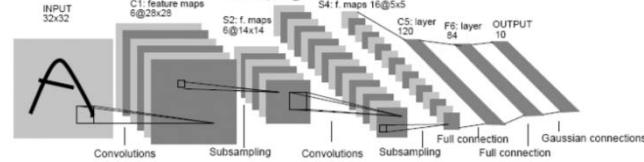
Discriminative models

Nearest neighbor



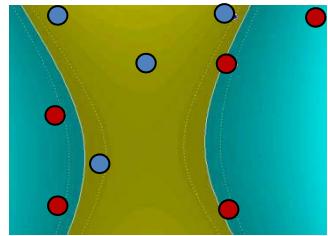
Shakhnarovich, Viola, Darrell 2003
Berg, Berg, Malik 2005...

Neural networks



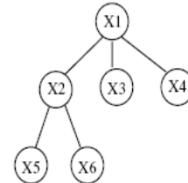
LeCun, Bottou, Bengio, Haffner 1998
Rowley, Baluja, Kanade 1998
...

Support Vector Machines



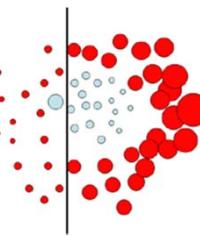
Guyon, Vapnik, Heisele,
Serre, Poggio...

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

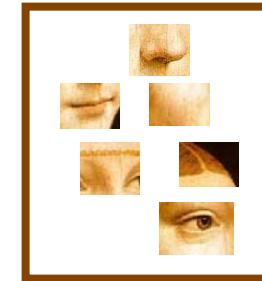
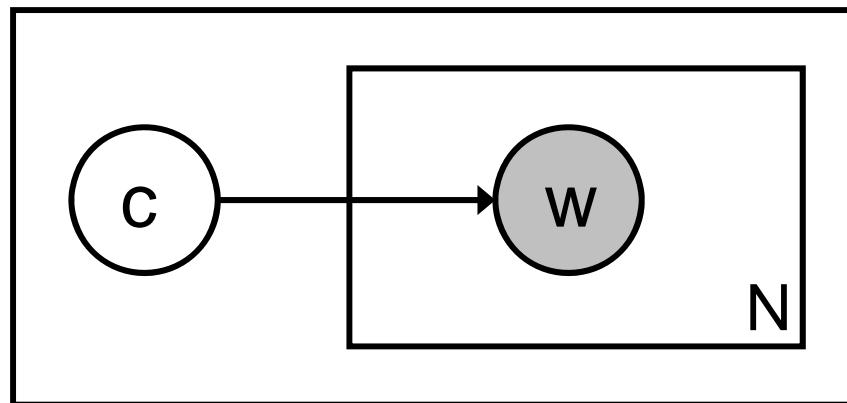
2. Hierarchical Bayesian text models (pLSA and LDA)

- 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

- \mathbf{w} : a collection of all N codewords in the image
$$\mathbf{w} = [w_1, w_2, \dots, w_N]$$
- c : category of the image

the Naïve Bayes model



Graphical model

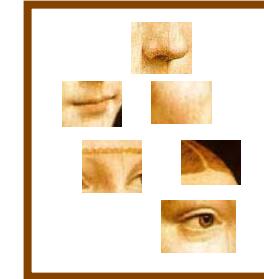
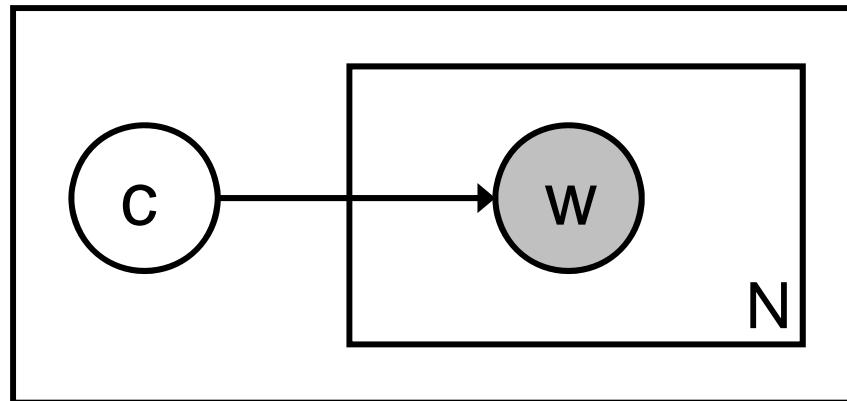
$$\text{Posterior} = p(c \mid w) \propto p(c)p(w \mid c)$$

probability
that image I is
of category c

Prior prob. of the object classes

Image likelihood given the class

the Naïve Bayes model



Graphical model

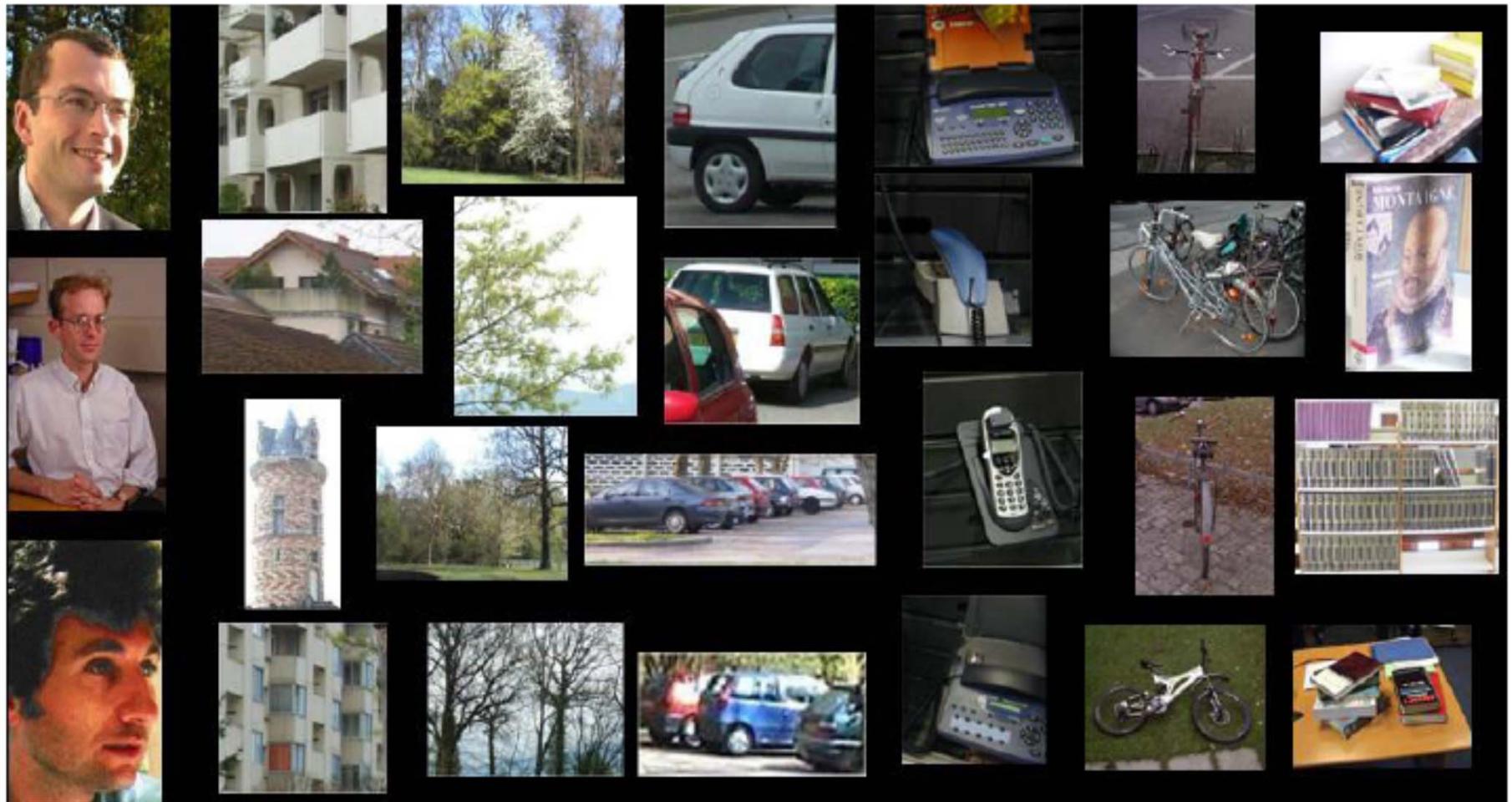
$$c^* = \arg \max_c p(c | w) \propto p(c)p(w | c) = p(c) \prod_{n=1}^N p(w_n | c)$$

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 classes¹: 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 →	<i>faces</i>	<i>buildings</i>	<i>trees</i>	<i>cars</i>	<i>phones</i>	<i>bikes</i>	<i>books</i>
<i>faces</i>	76	4	2	3	4	4	13
<i>buildings</i>	2	44	5	0	5	1	3
<i>trees</i>	3	2	80	0	0	5	0
<i>cars</i>	4	1	0	75	3	1	4
<i>phones</i>	9	15	1	16	70	14	11
<i>bikes</i>	2	15	12	0	8	73	0
<i>books</i>	4	19	0	6	7	2	69
<i>Mean ranks</i>	1.49	1.88	1.33	1.33	1.63	1.57	1.57

Csurka et al. 2004

Other generative BoW models

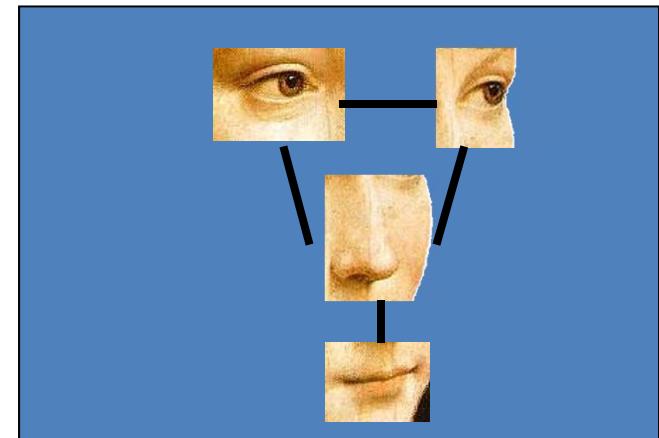
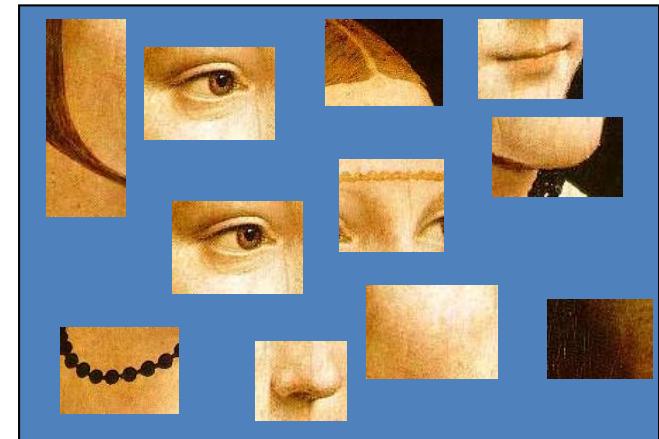
- Hierarchical Bayesian topic models (e.g. pLSA and LDA)
 - Object categorization: Sivic et al. 2005, Sudderth et al. 2005
 - Natural scene categorization: Fei-Fei et al. 2005

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

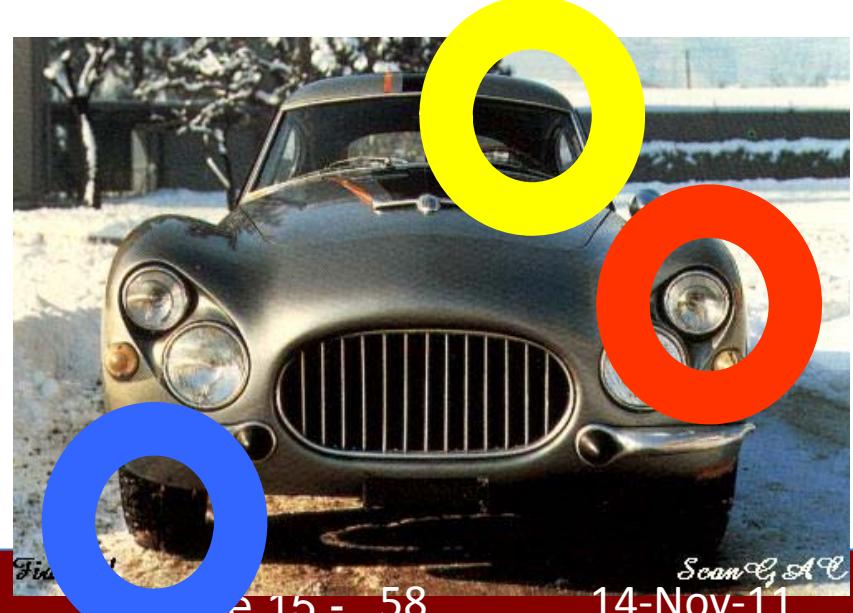
- No rigorous geometric information of the object components
- It's intuitive to most of us that objects are made of parts – no such information
- Not extensively tested yet for
 - View point invariance
 - Scale invariance
- Segmentation and localization unclear



What we will learn today?

- Bag of Words model (Problem Set 4 (Q2))
 - Basic representation
 - Different learning and recognition algorithms
- Constellation model
 - Weakly supervised training
 - One-shot learning (supplementary materials)
- (Problem Set 4 (Q1))

Model: Parts and Structure



10.1.3.2

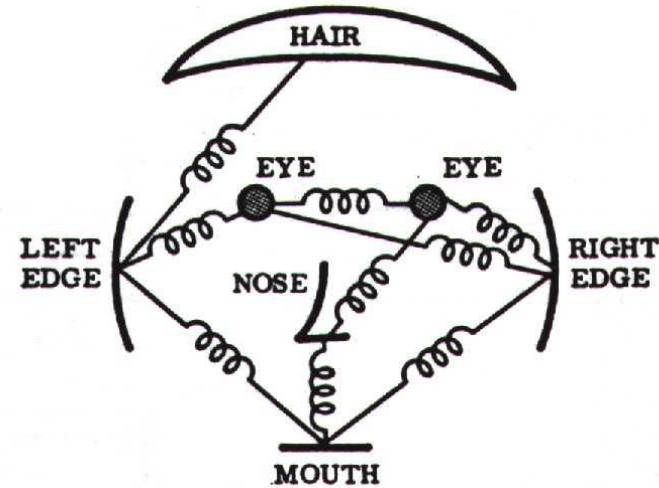
Final

e 15 - 58

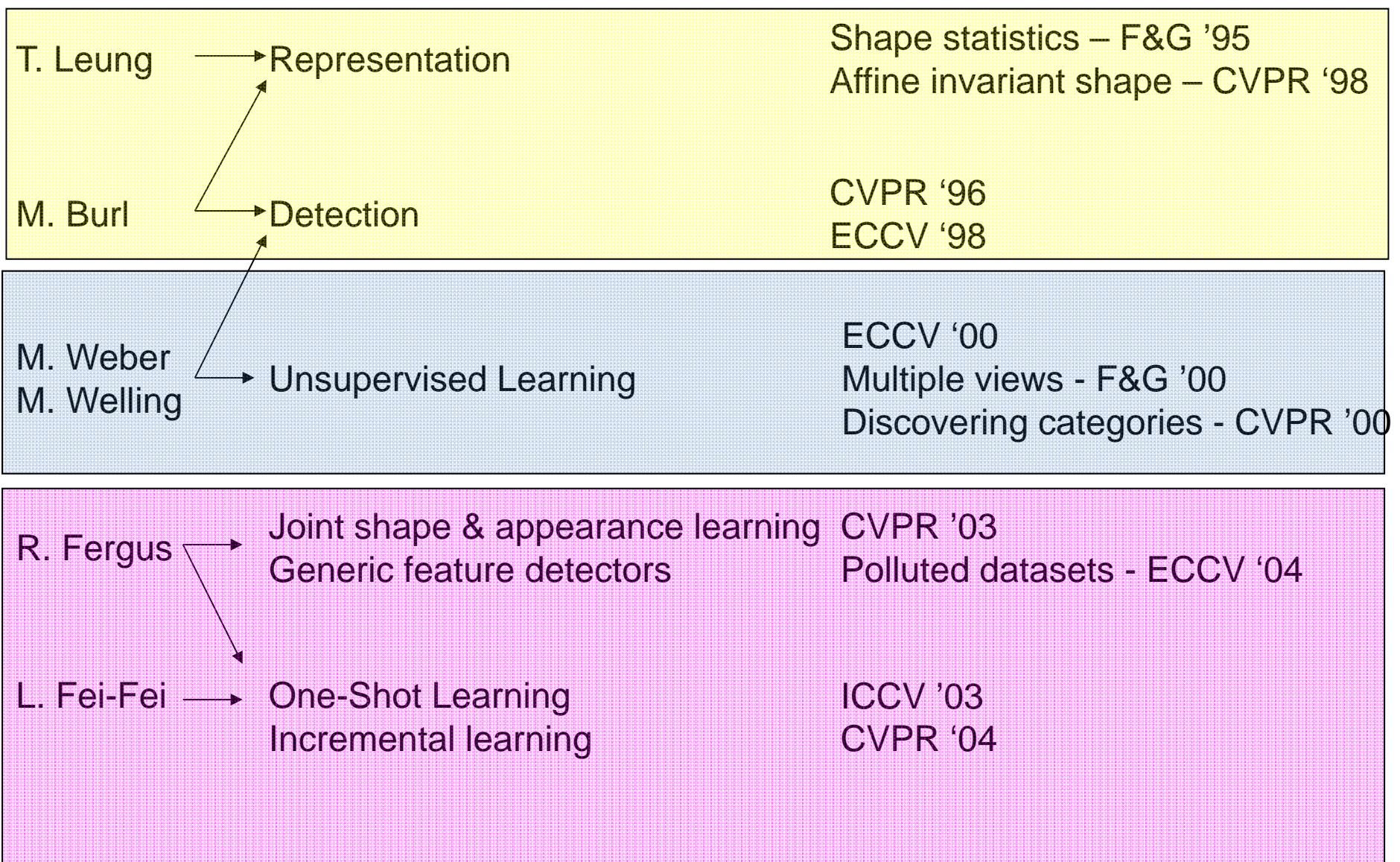
Scan & AC
14-Nov-11

Parts and Structure Literature

- Fischler & Elschlager 1973
- Yuille '91
- Brunelli & Poggio '93
- Lades, v.d. Malsburg et al. '93
- Cootes, Lanitis, Taylor et al. '95
- Amit & Geman '95, '99
- et al. Perona '95, '96, '98, '00, '03
- Huttenlocher et al. '00
- Agarwal & Roth '02
- etc...



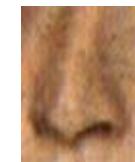
The Constellation Model



Deformations



A



B



C

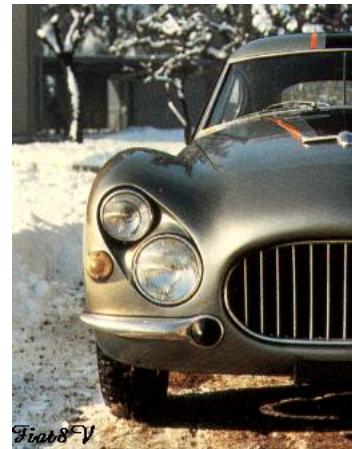


D

Presence / Absence of Features



www.corbis.com

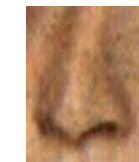


Fiat 8V

occlusion



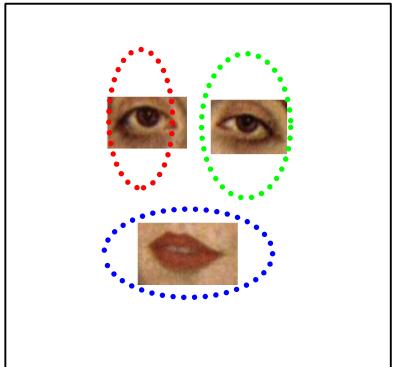
Background clutter



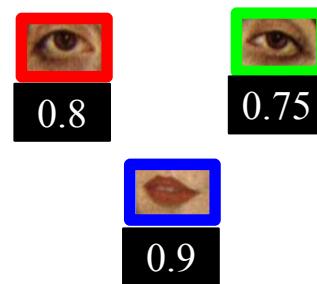
Generative probabilistic model

Foreground model

Gaussian shape pdf

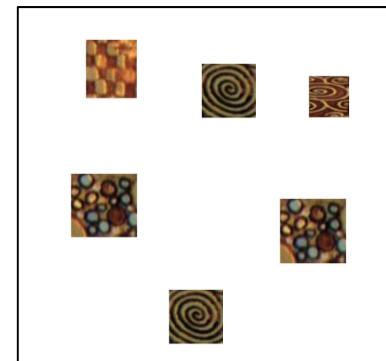


Prob. of detection



Clutter model

Uniform shape pdf



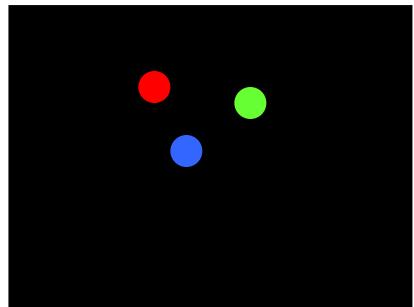
detections

$$\begin{aligned} p_{\text{Poisson}}(N_1/\lambda_1) \\ p_{\text{Poisson}}(N_2/\lambda_2) \\ p_{\text{Poisson}}(N_3/\lambda_3) \end{aligned}$$

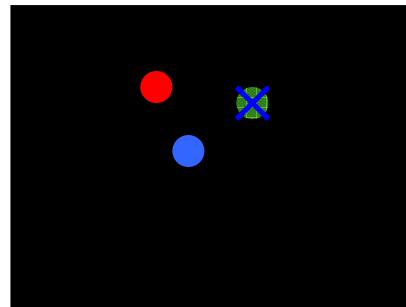
Assumptions: (a) Clutter independent of foreground detections
(b) Clutter detections independent of each other

Example

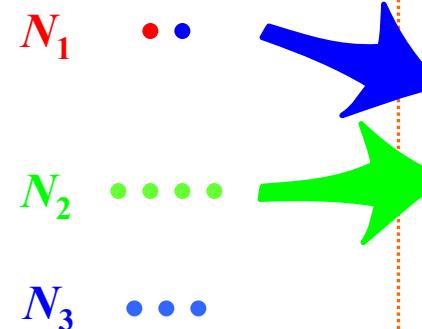
1. Object Part Positions



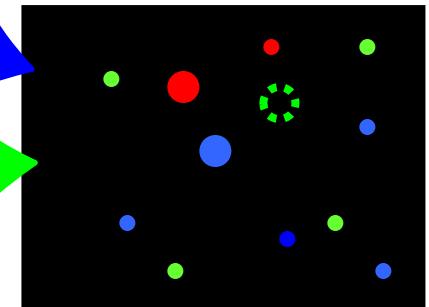
2. Part Absence



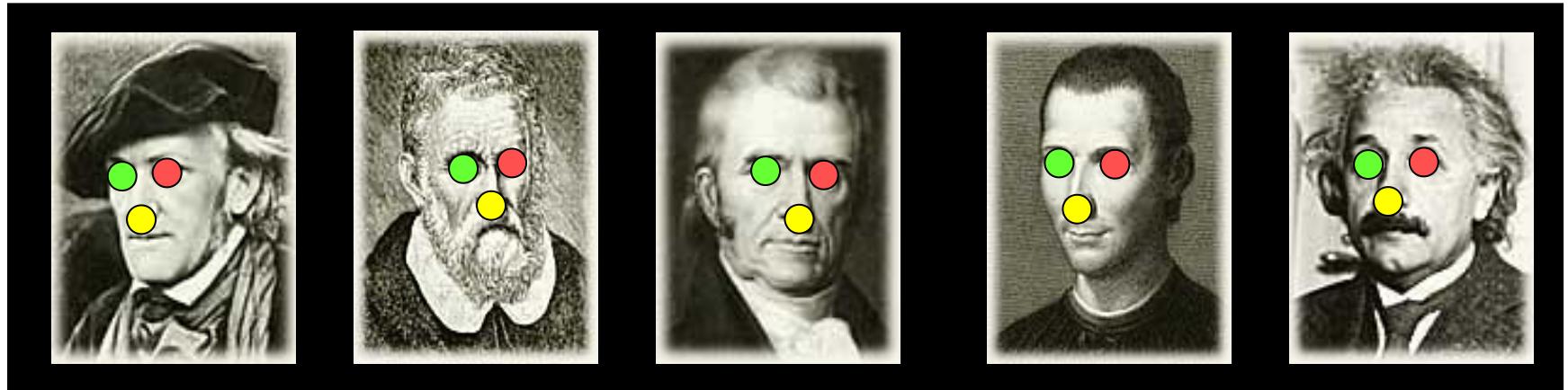
3a. N false detect



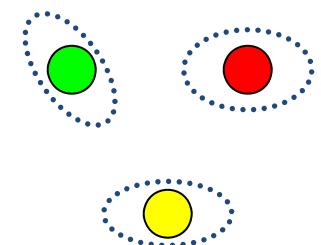
3b. Position f. detect



Learning Models `Manually'

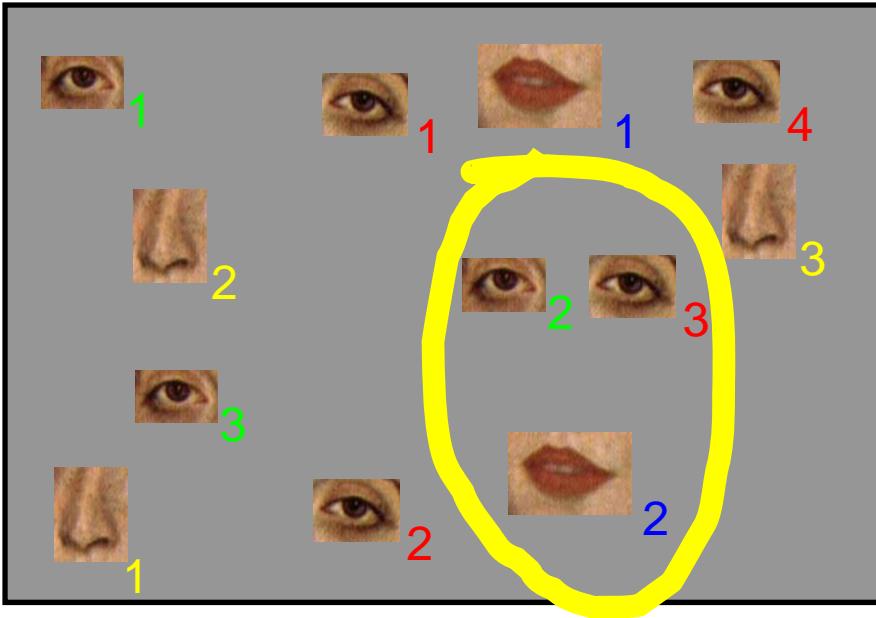


- Obtain set of training images
- Choose parts
- Label parts by hand, train detectors
- Learn model from labeled parts



Recognition

1. Run part detectors exhaustively over image



$$h = \begin{pmatrix} 0 \dots N_1 \\ 0 \dots N_2 \\ 0 \dots N_3 \\ 0 \dots N_4 \end{pmatrix}$$

e.g. $h = \begin{pmatrix} 2 \\ 3 \\ 0 \\ 2 \end{pmatrix}$

2. Try different combinations of detections in model
 - Allow detections to be missing (occlusion)
3. Pick hypothesis which maximizes:
$$\frac{p(Data | Object, Hyp)}{p(Data | Clutter, Hyp)}$$
4. If ratio is above threshold then, instance detected

So far.....

- Representation
 - Joint model of part locations
 - Ability to deal with background clutter and occlusions
- Learning
 - Manual construction of part detectors
 - Estimate parameters of shape density
- Recognition
 - Run part detectors over image
 - Try combinations of features in model
 - Use efficient search techniques to make fast

Unsupervised Learning

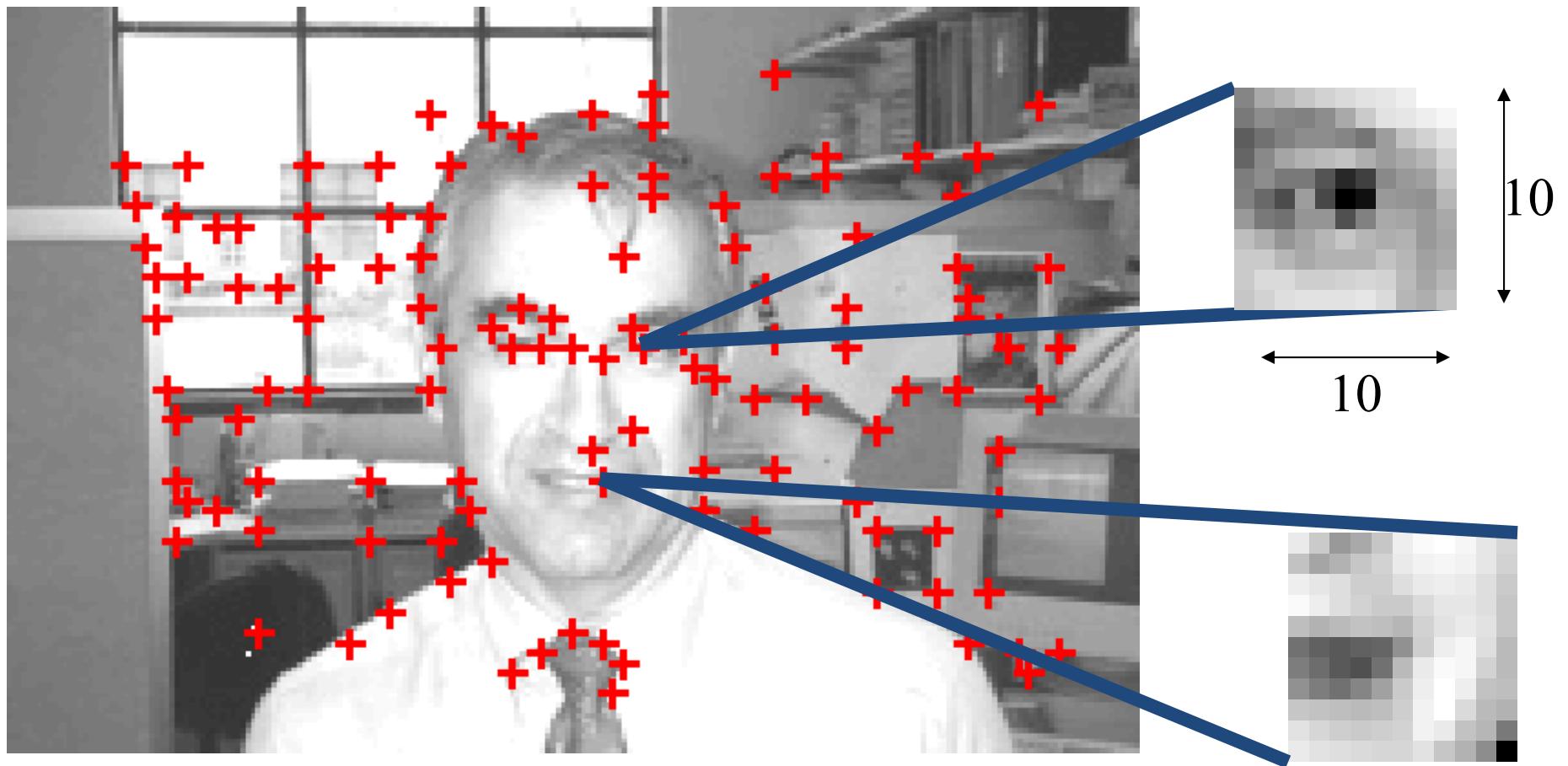
Weber & Welling et. al.

(Semi) Unsupervised learning



- Know if image contains object or not
- But no segmentation of object or manual selection of features

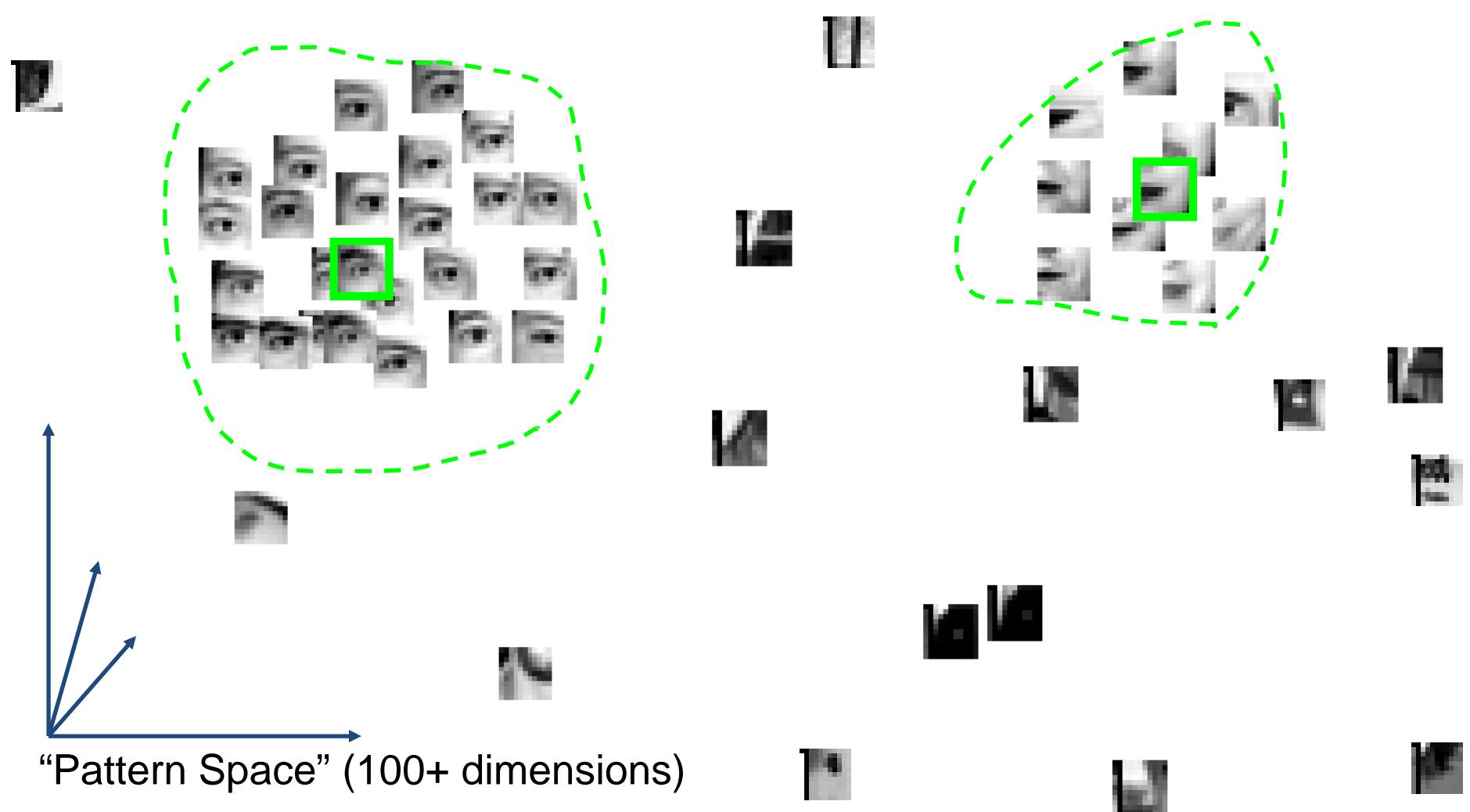
Unsupervised detector training - 1



- Highly textured neighborhoods are selected automatically
- produces 100-1000 patterns per image



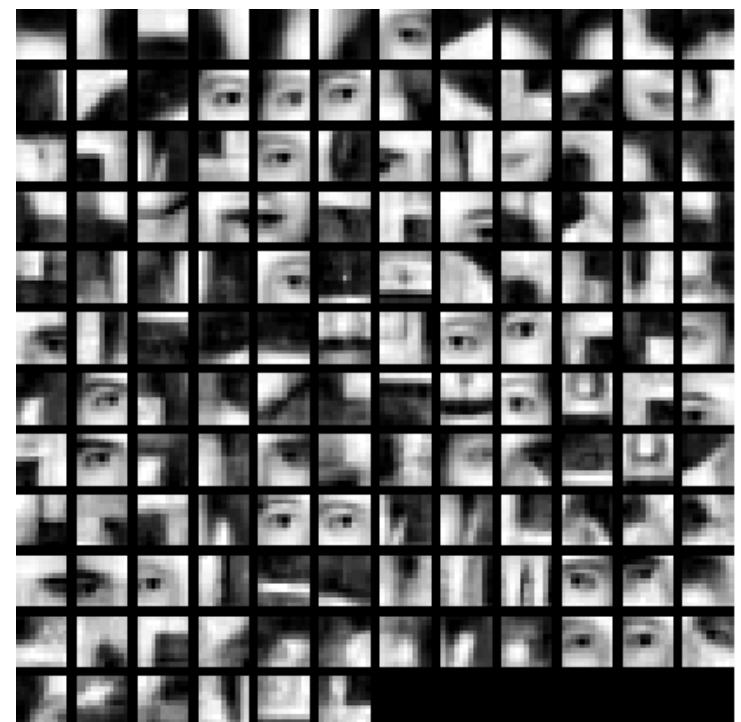
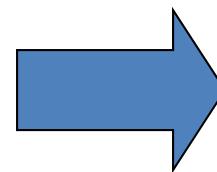
Unsupervised detector training - 2



Unsupervised detector training - 3



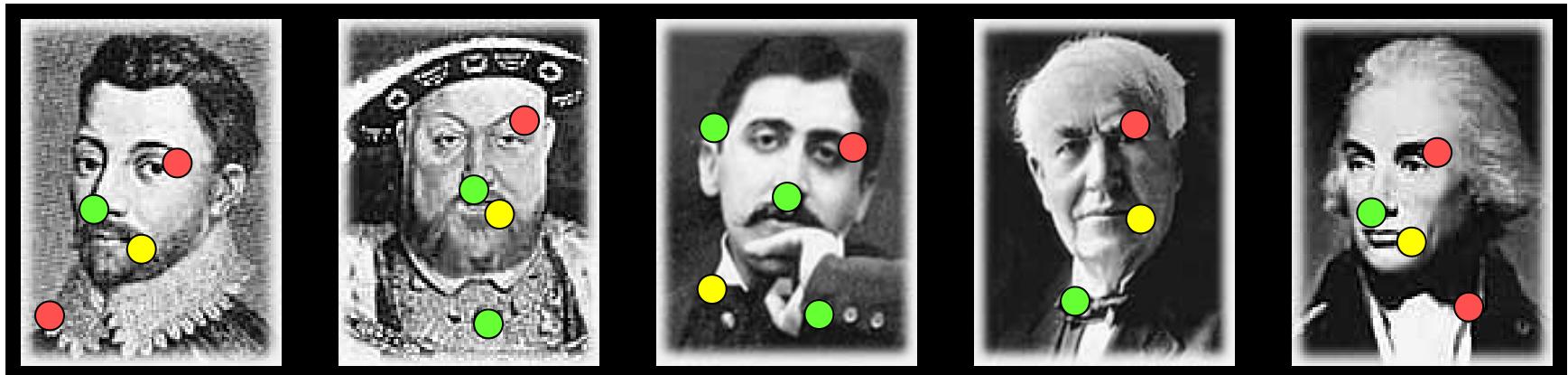
100-1000 images



~100 detectors

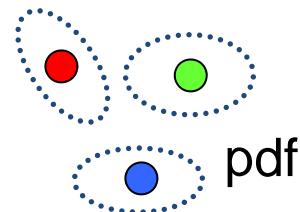
Learning

- Take training images. Pick set of detectors. Apply detectors.
- Task: Estimation of model parameters
- Chicken and Egg type problem, since we initially know neither:
 - Model parameters
 - Assignment of regions to foreground / background
- Let the assignments be a hidden variable and use EM algorithm to learn them and the model parameters

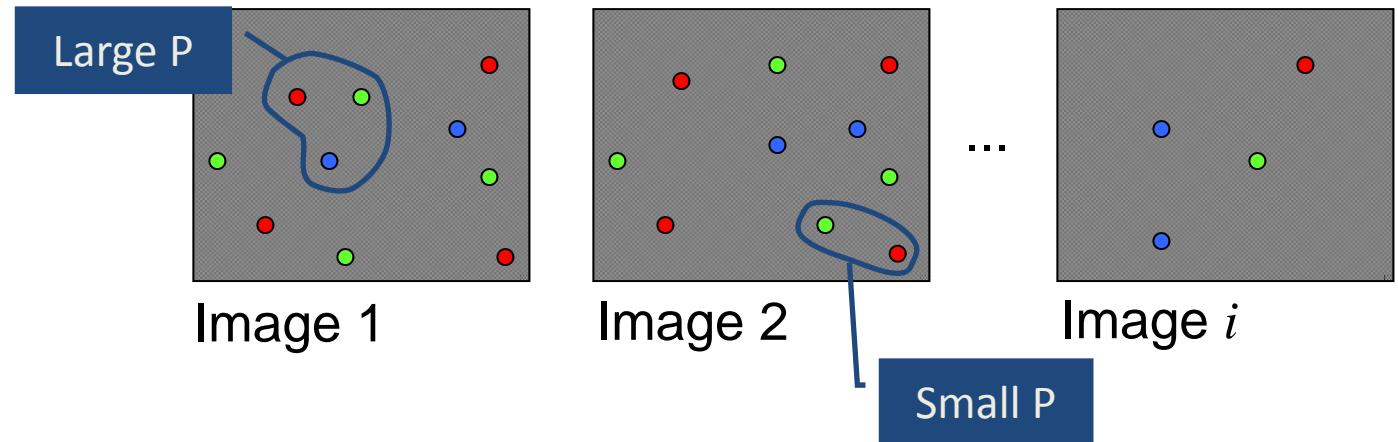


ML using EM

1. Current estimate



2. Assign probabilities to constellations



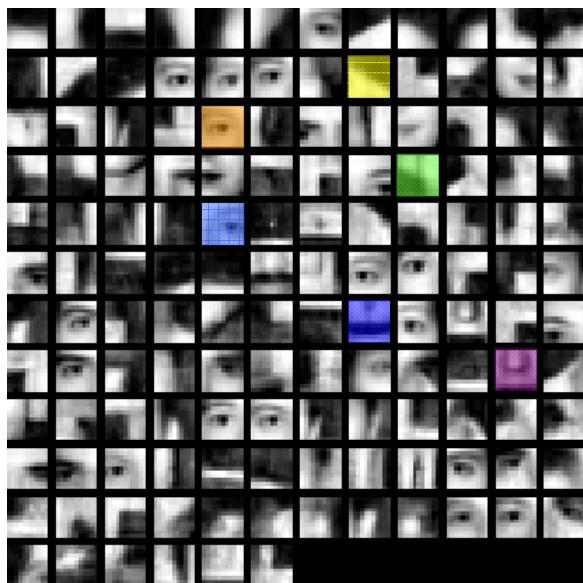
3. Use probabilities as weights to re-estimate parameters. Example: μ

$$\text{Large P} \times \begin{matrix} \bullet \\ \bullet \\ \bullet \end{matrix} + \text{Small P} \times \begin{matrix} \bullet \\ \bullet \\ \bullet \end{matrix} + \dots = \begin{matrix} \bullet \\ \bullet \\ \bullet \end{matrix}$$

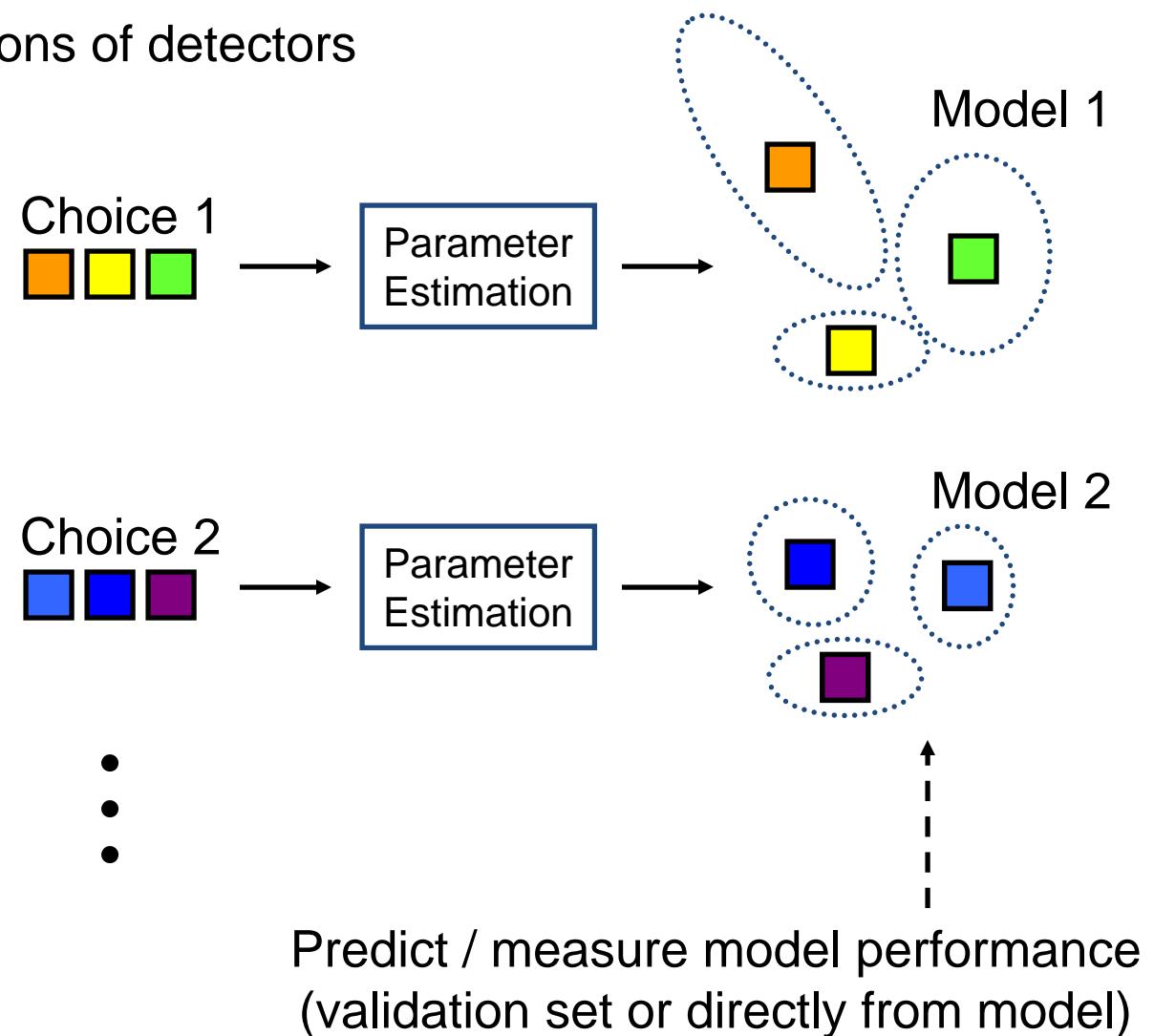
new estimate of μ

Detector Selection

- Try out different combinations of detectors
(Greedy search)



Detectors (≈ 100)



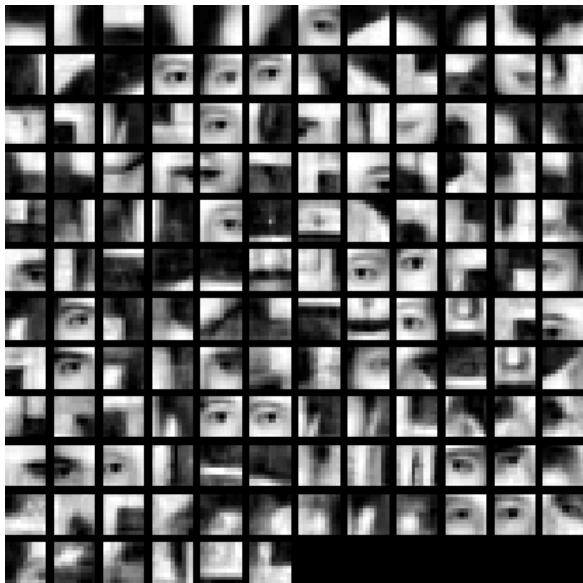
Frontal Views of Faces



- 200 Images (100 training, 100 testing)
- 30 people, different for training and testing

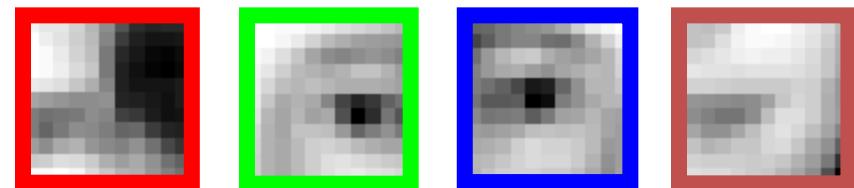
Learned face model

Pre-selected Parts

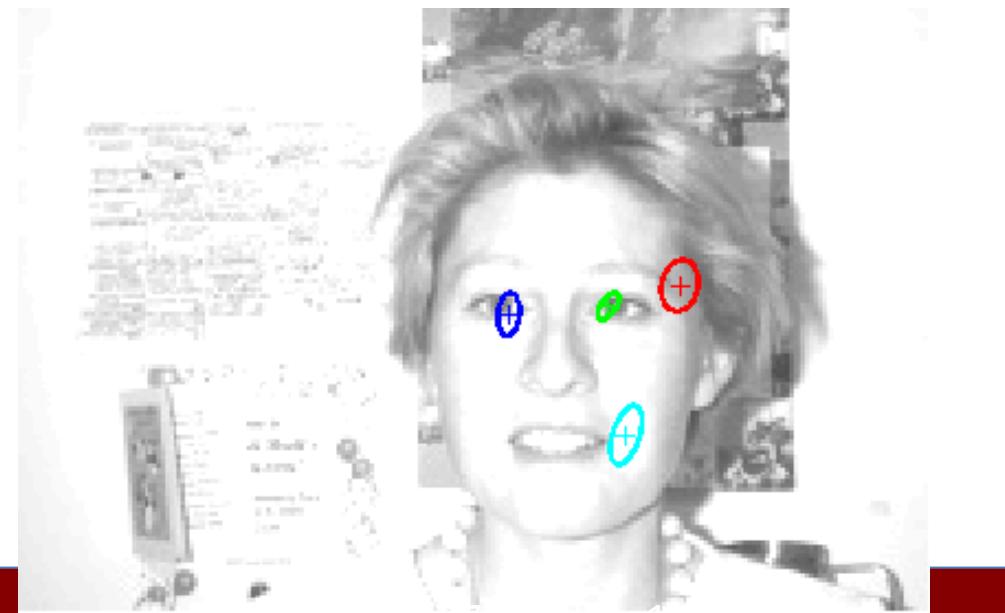


Test Error: 6% (4 Parts)

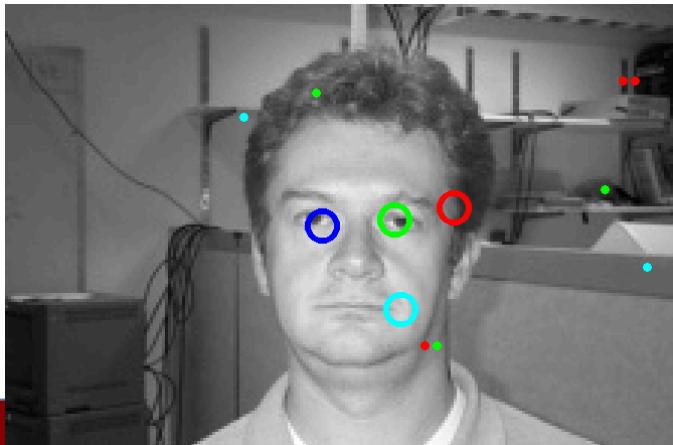
Parts in Model



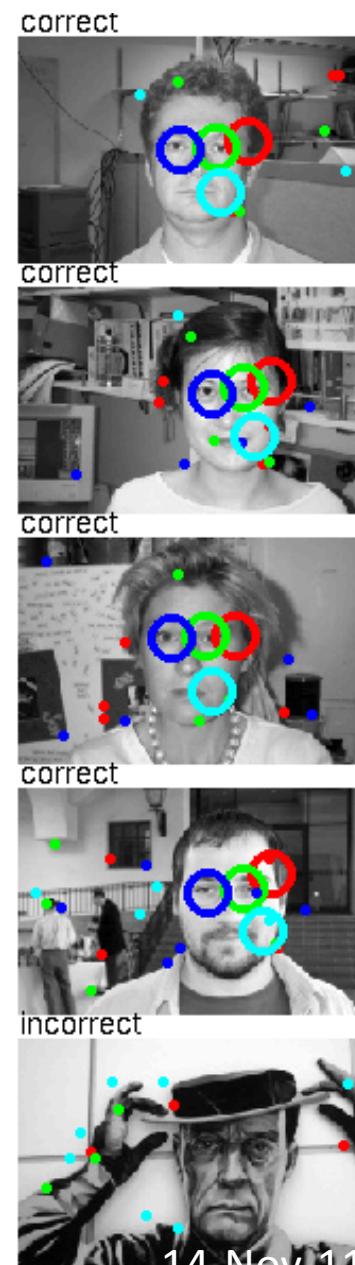
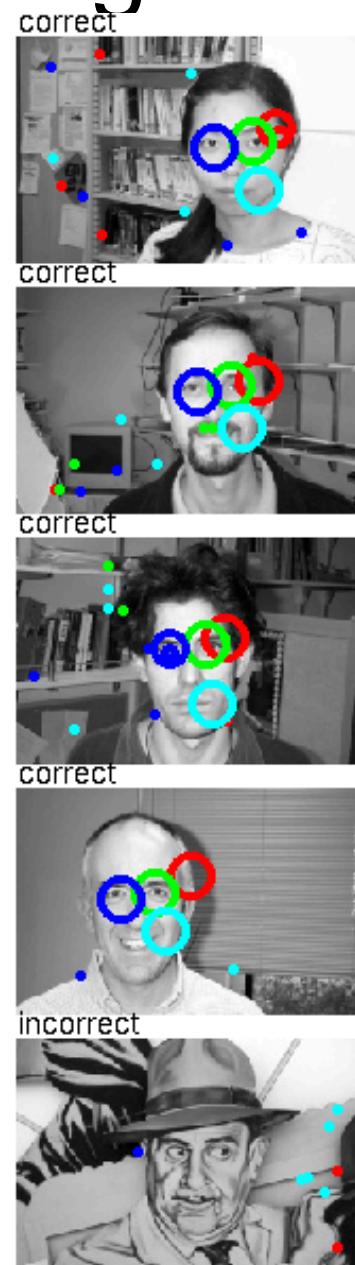
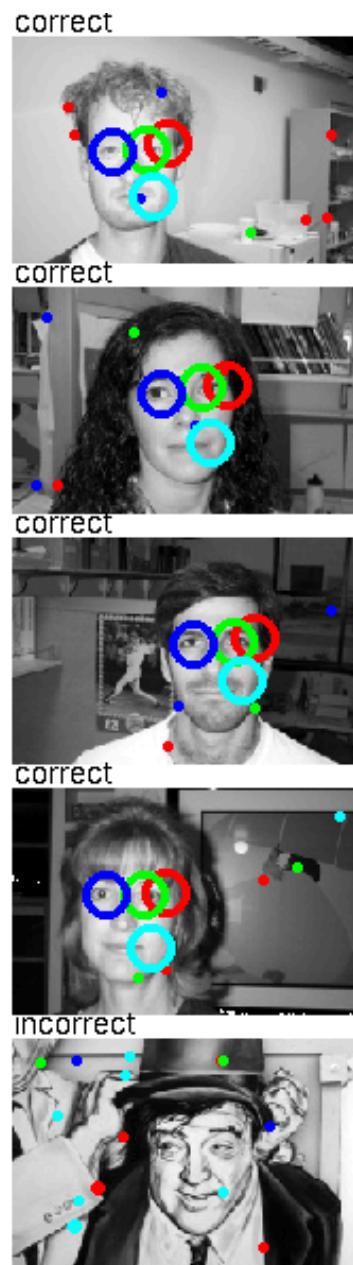
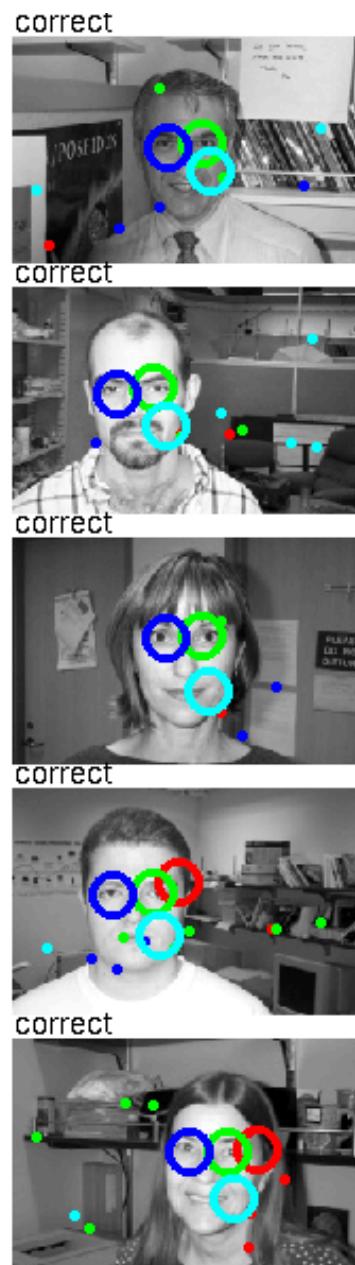
Model Foreground pdf



Sample Detection

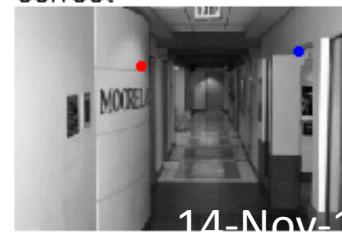
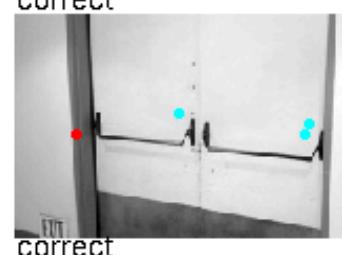
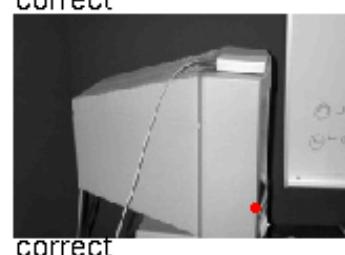
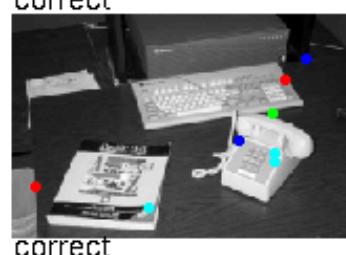
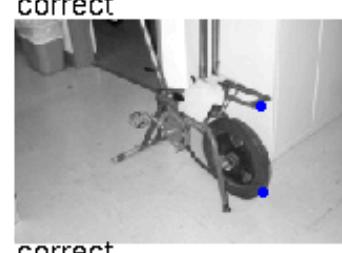
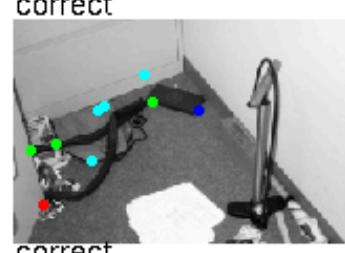
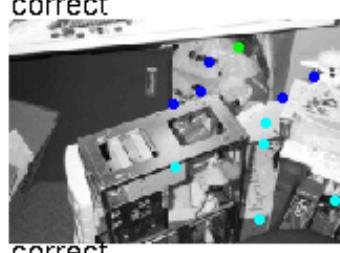
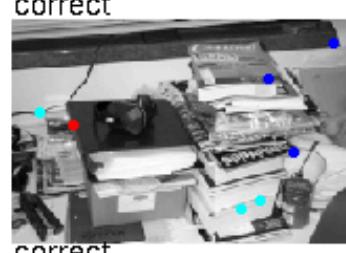
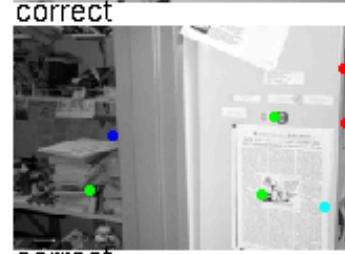
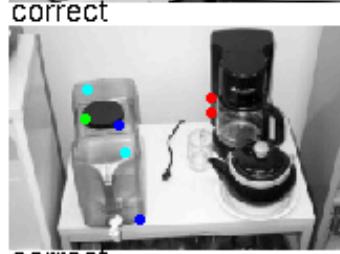
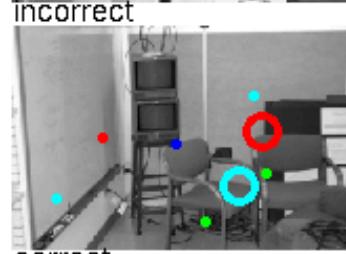
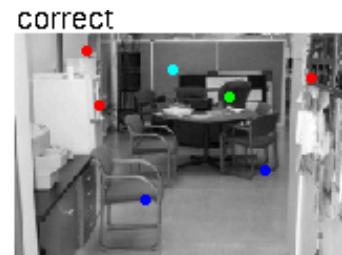
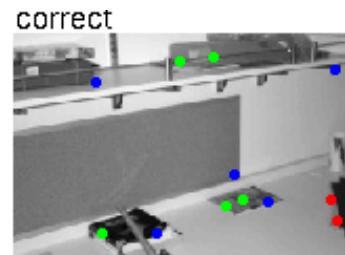
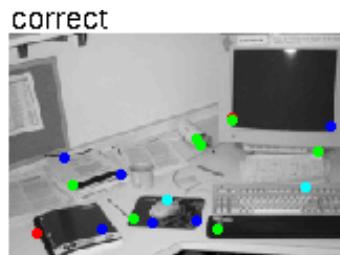


Face images



14 Nov 11

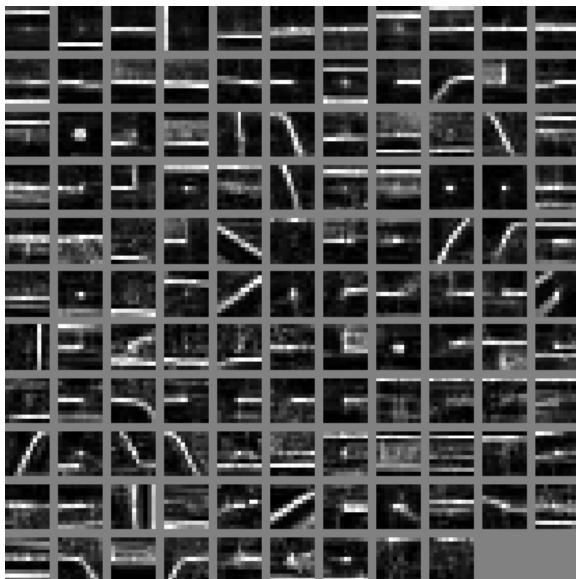
Background images



14-Nov-1

Car from Rear

Preselected Parts

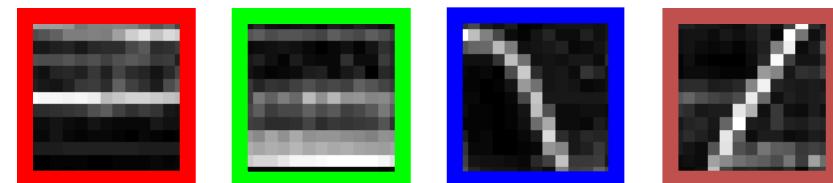


Sample Detection

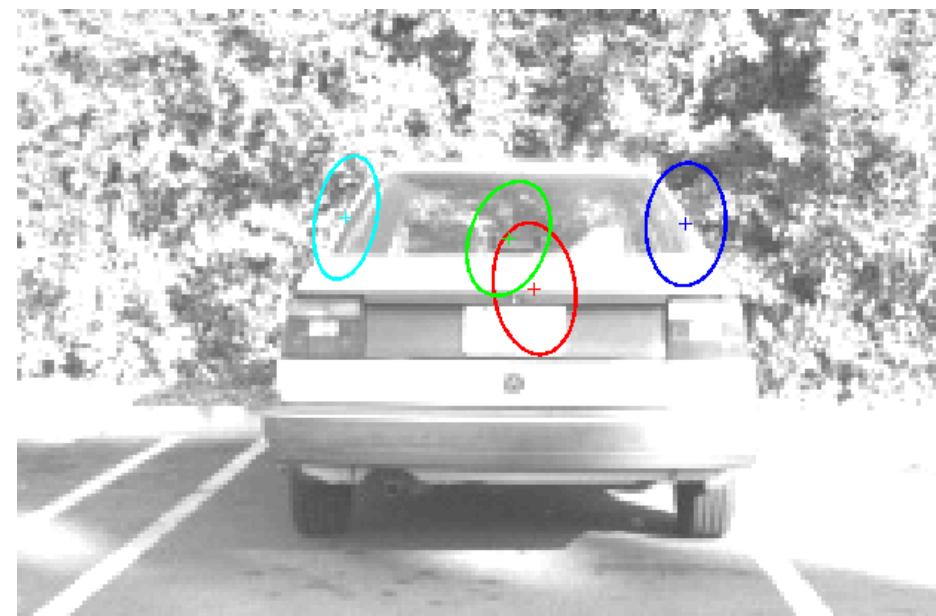


Test Error: 13% (5 Parts)

Parts in Model



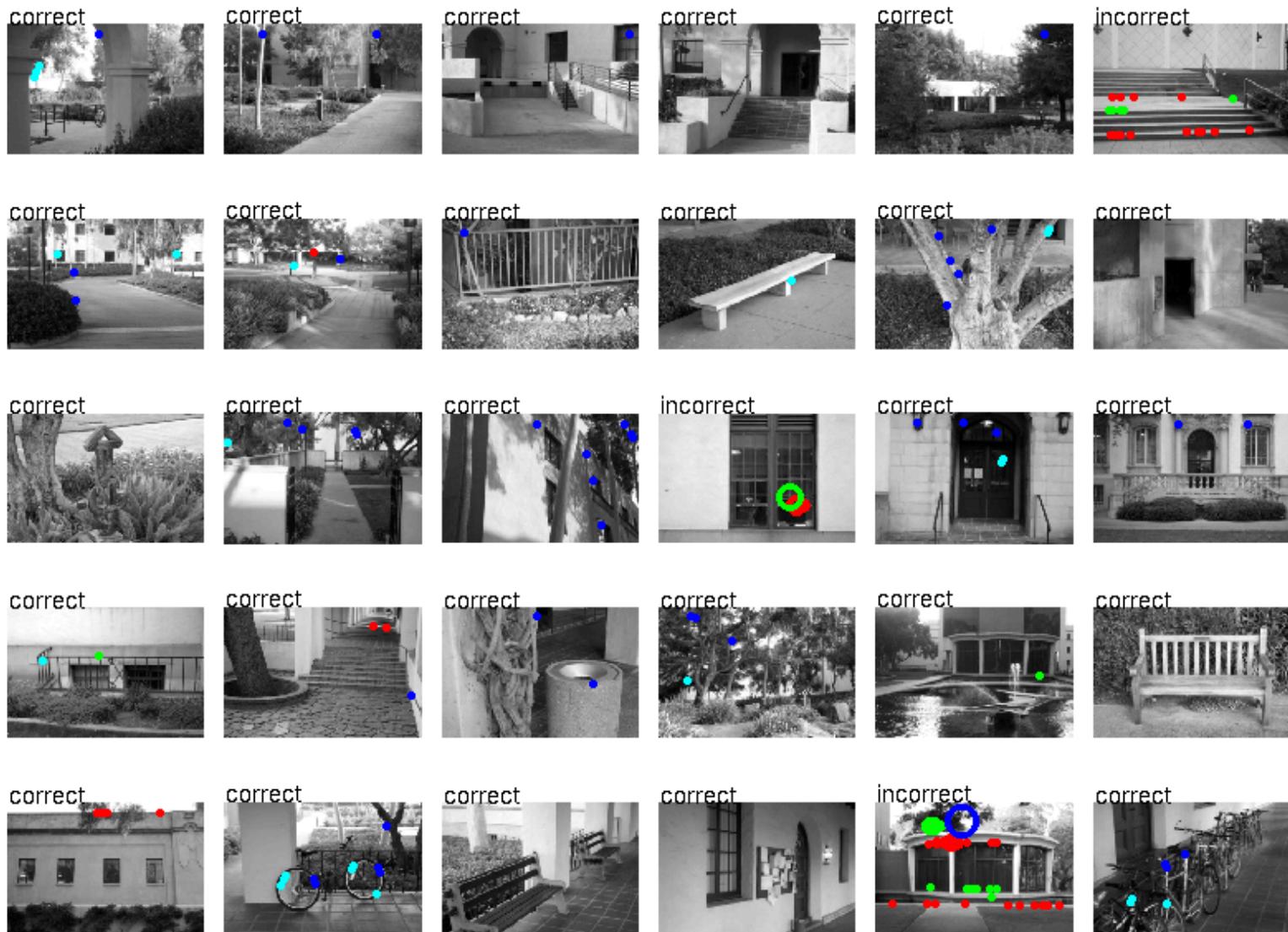
Model Foreground pdf



Detections of Cars



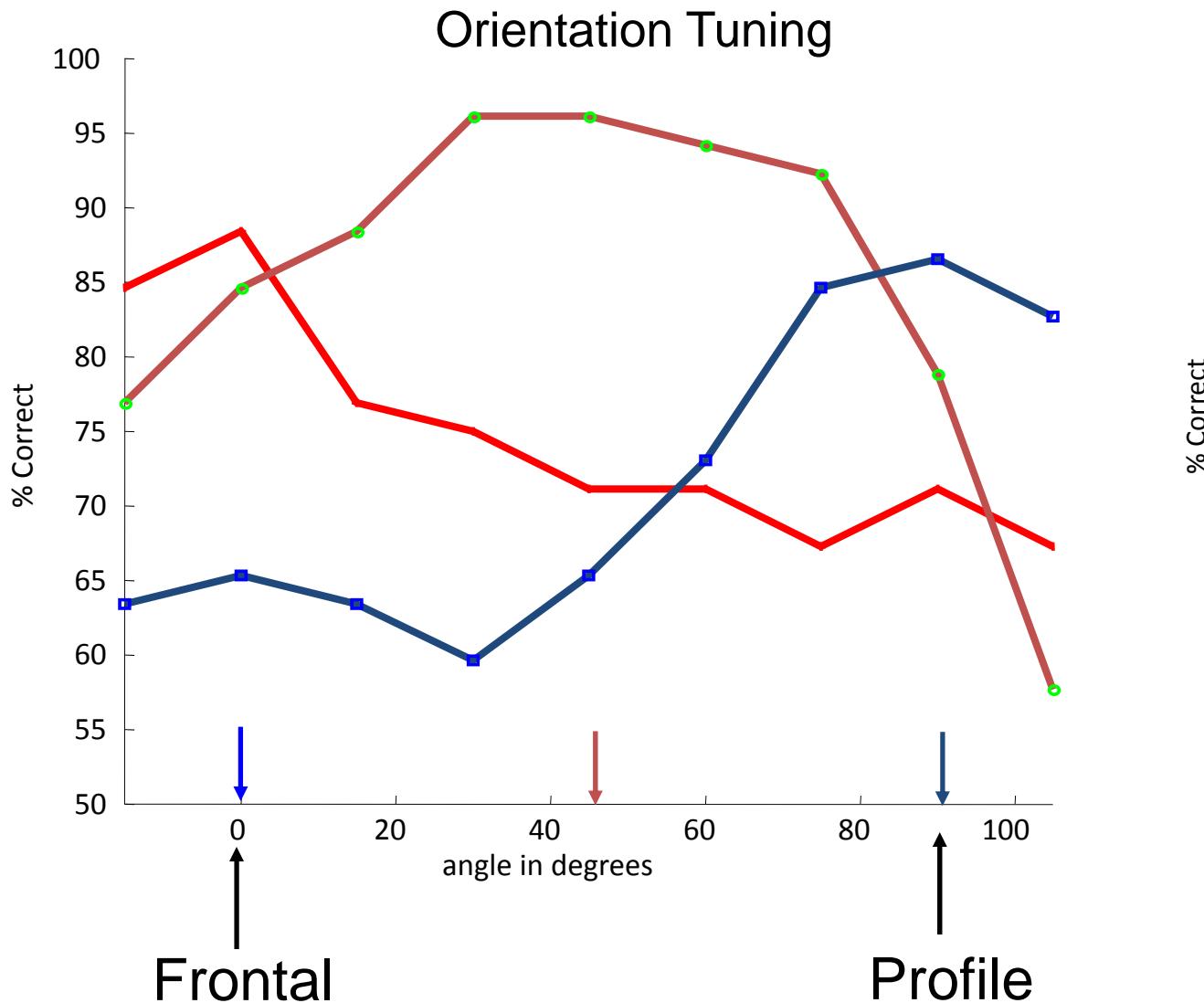
Background Images



3D Object recognition – Multiple mixture components



3D Orientation Tuning



So far (2).....

- Representation
 - Multiple mixture components for different viewpoints
- Learning
 - Now semi-unsupervised
 - Automatic construction and selection of part detectors
 - Estimation of parameters using EM
- Recognition
 - As before
- Issues:
 - Learning is slow (many combinations of detectors)
 - Appearance learnt first, then shape

Issues

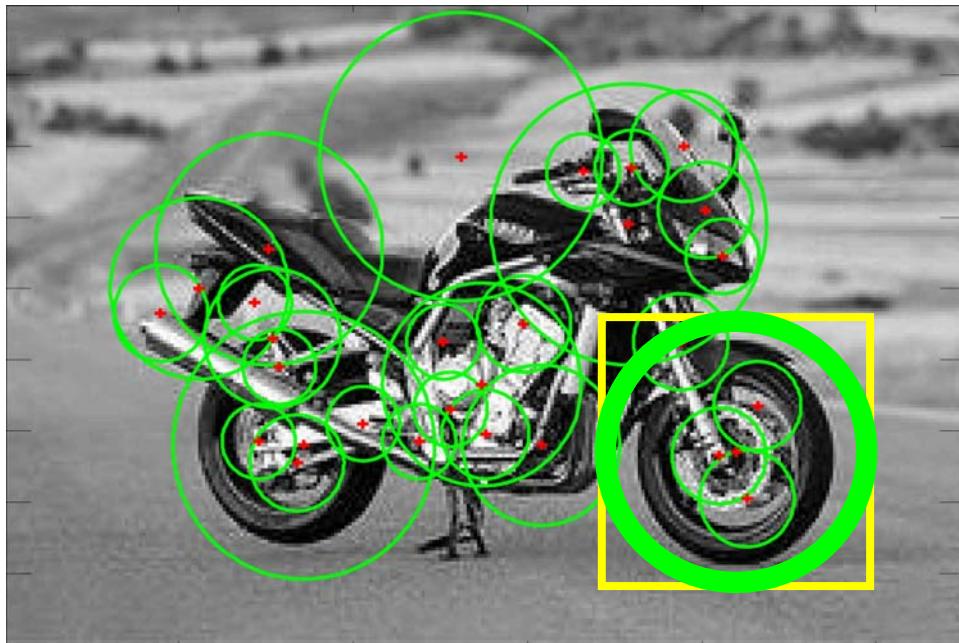
- Speed of learning
 - Slow (many combinations of detectors)
- Appearance learnt first, then shape
 - Difficult to learn part that has stable location but variable appearance
 - Each detector is used as a cross-correlation filter, giving a hard definition of the part's appearance
- Would like a fully probabilistic representation of the object

Object categorization

Fergus et. al.

CVPR '03, IJCV '06

Detection & Representation of regions



Appearance

- Find regions within image
- Use salient region operator
(Kadir & Brady 01)

Location

(x,y) coords. of region centre

Scale

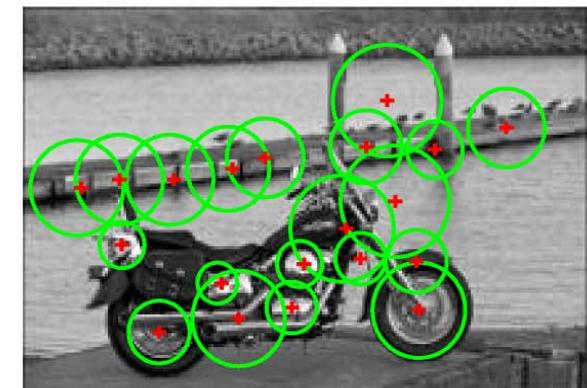
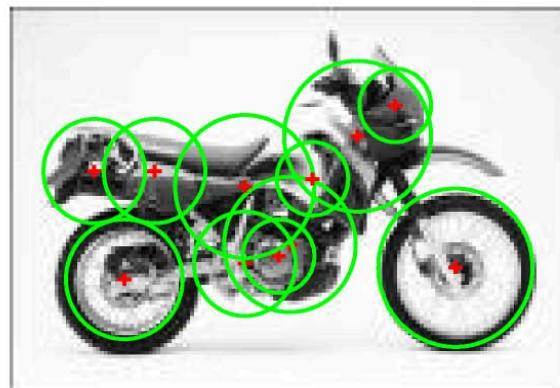
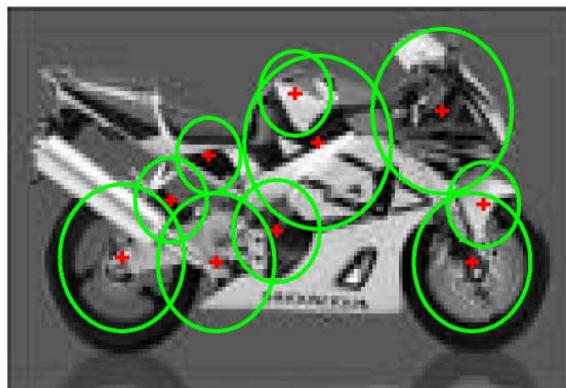
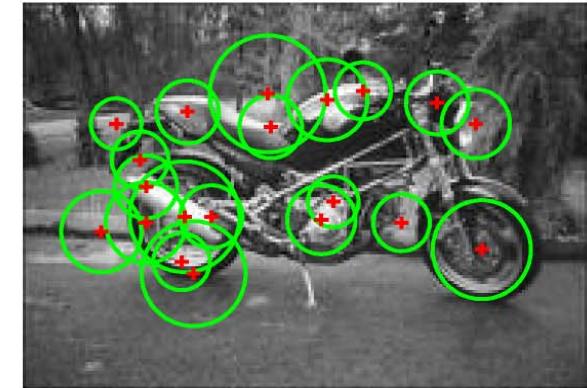
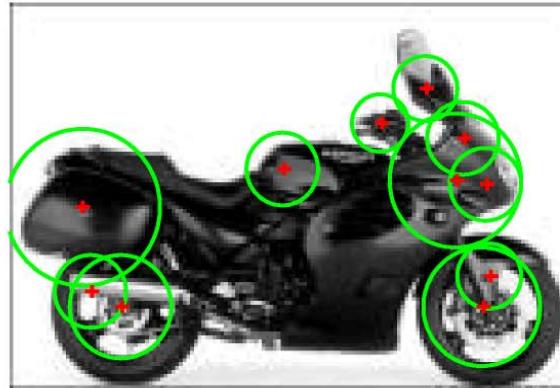
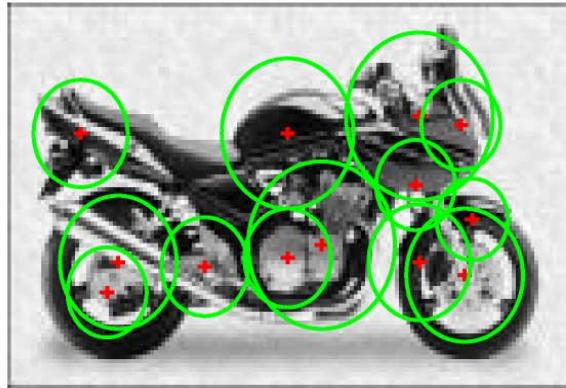
Radius of region (pixels)



Gives representation of appearance in low-dimensional vector space

Motorbikes example

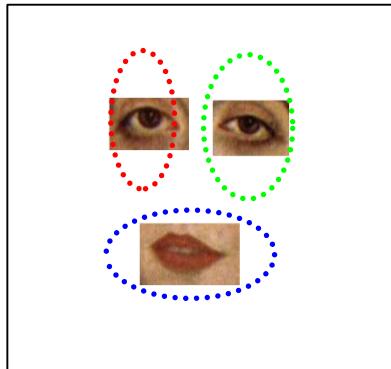
- Kadir & Brady saliency region detector



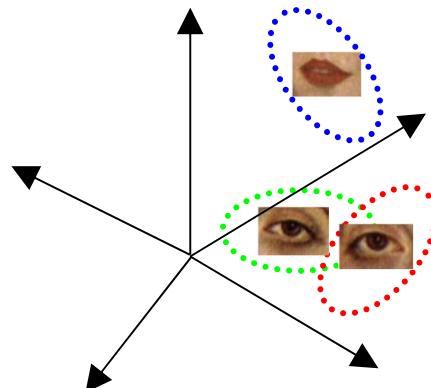
Generative probabilistic model (2)

Foreground model

Gaussian shape pdf

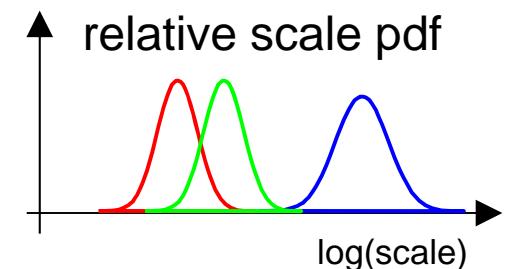


Gaussian part appearance pdf



based on Burl, Weber et al. [ECCV '98, '00]

Gaussian relative scale pdf

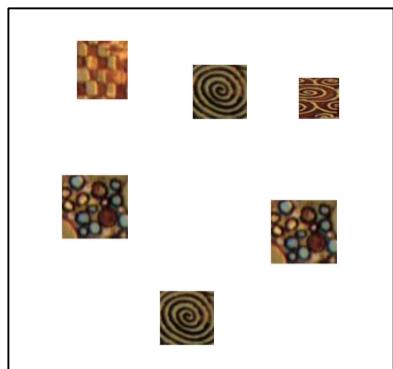


Prob. of detection

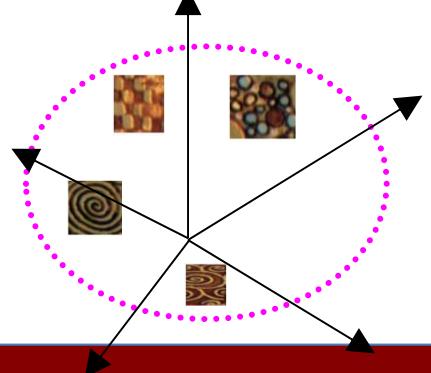


Clutter model

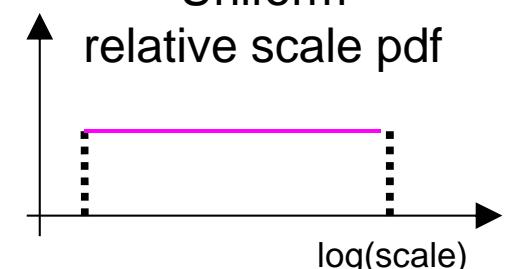
Uniform shape pdf



Gaussian background appearance pdf



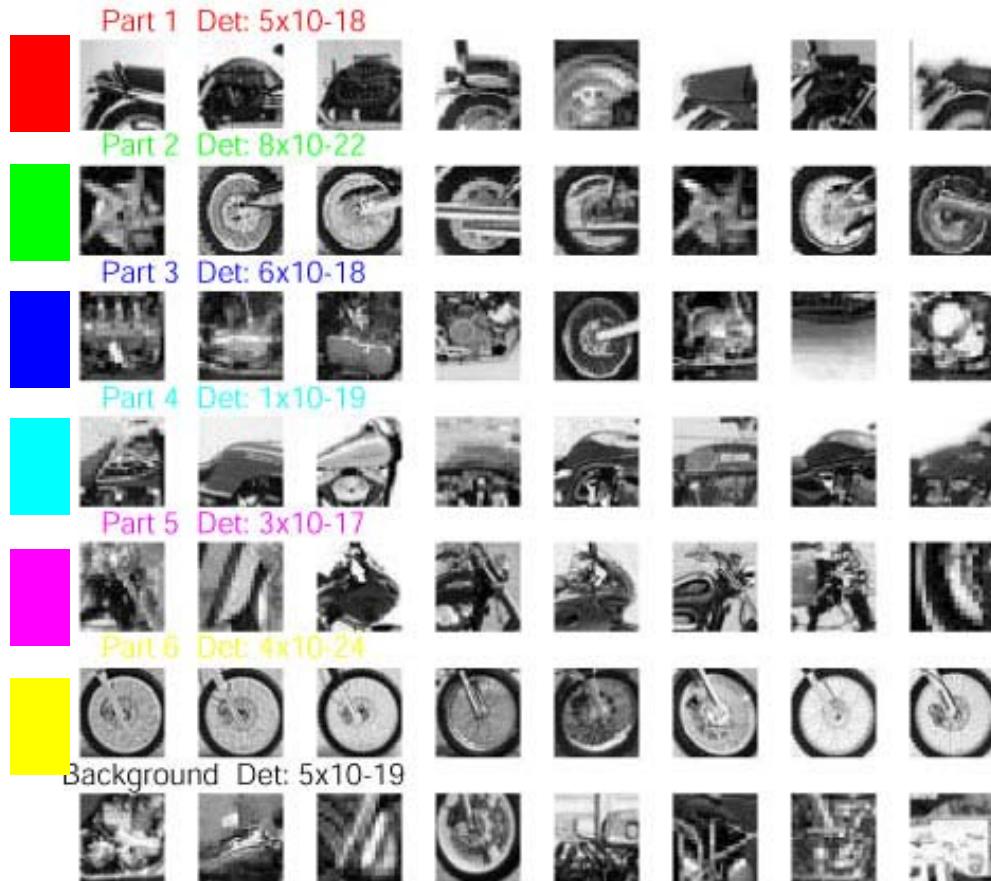
Uniform relative scale pdf



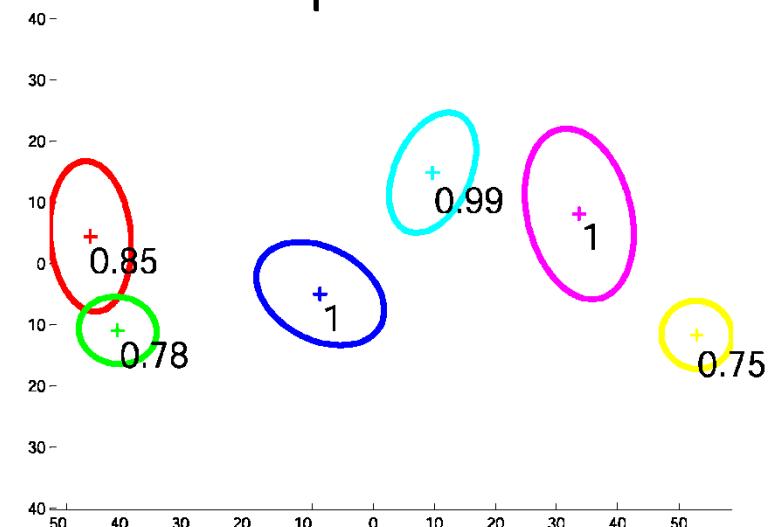
Poisson pdf on # detections

Motorbikes

Samples from appearance model

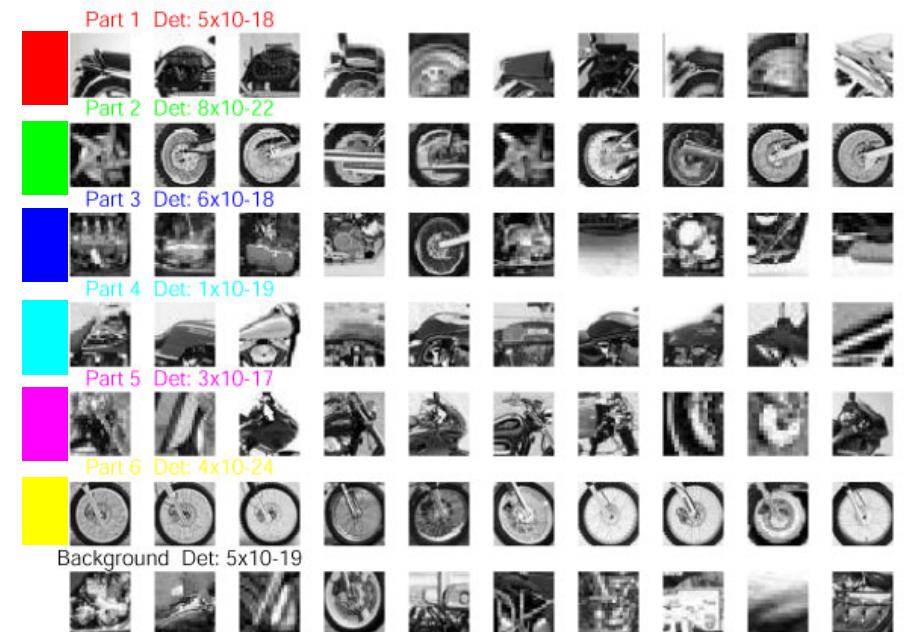
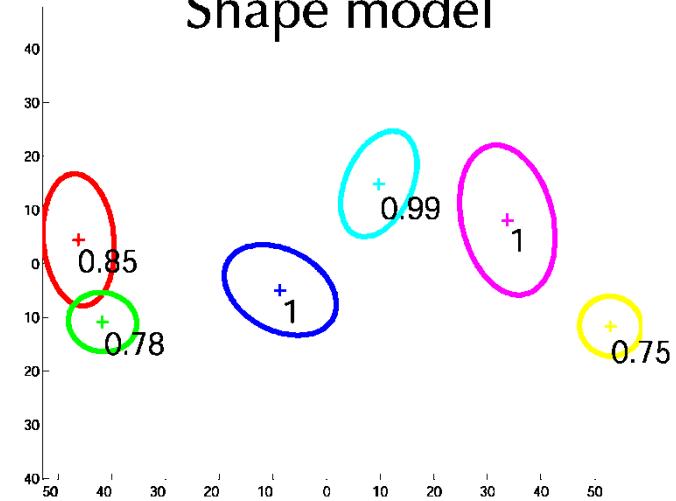
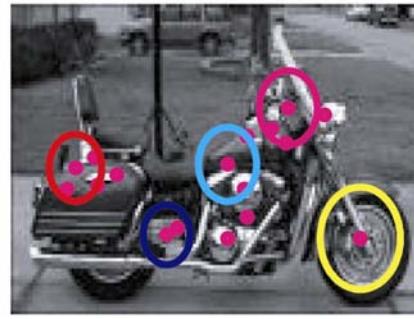
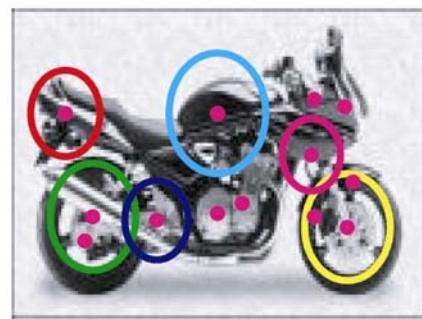
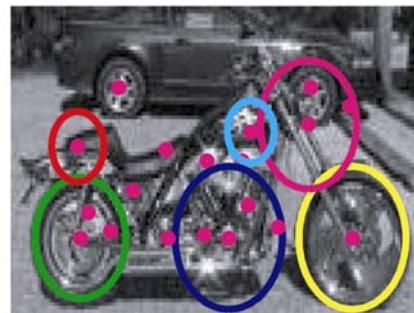
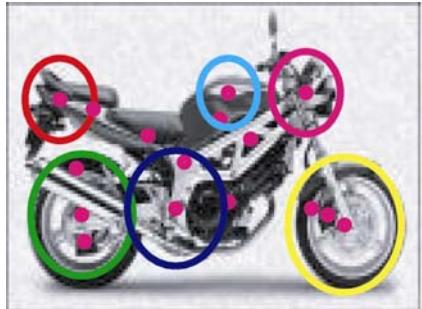


Shape model



Recognized Motorbikes

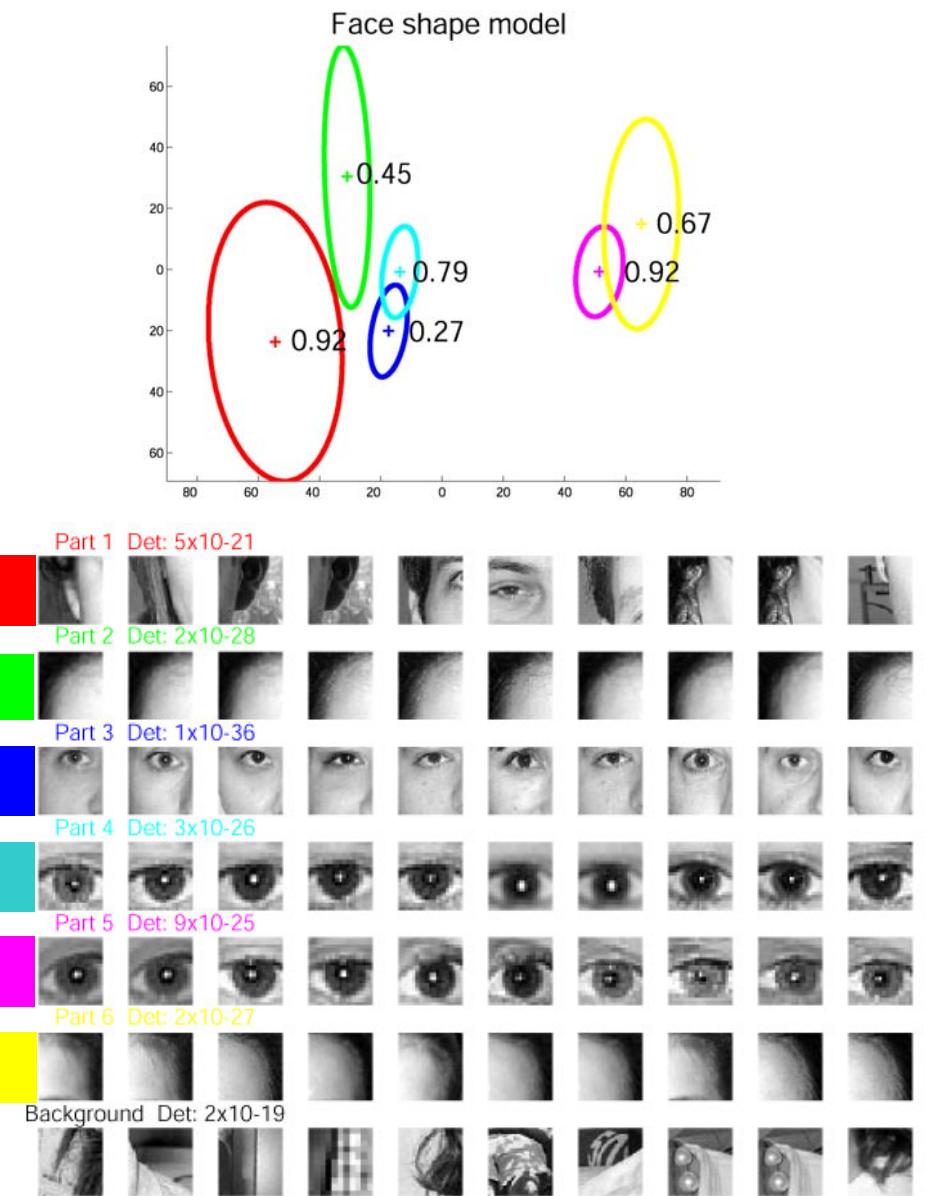
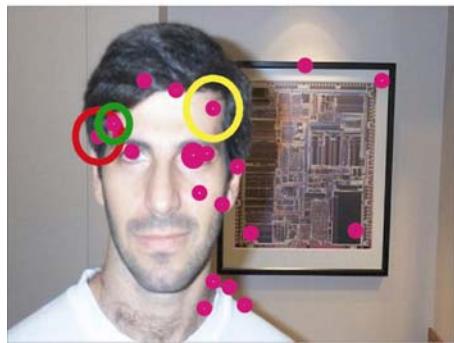
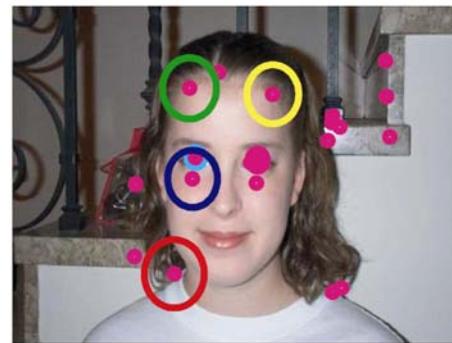
Shape model



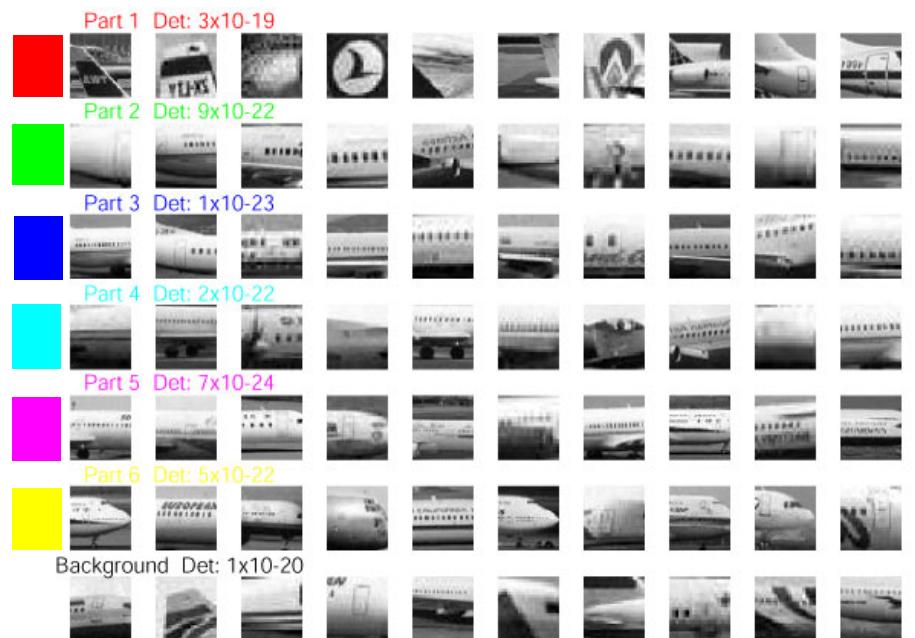
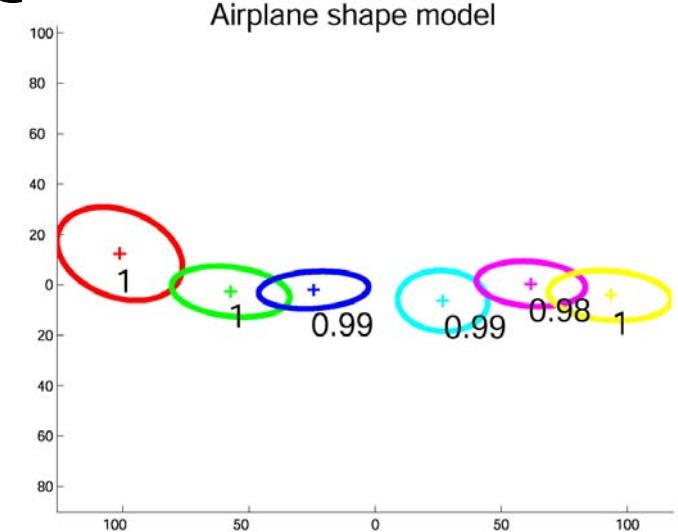
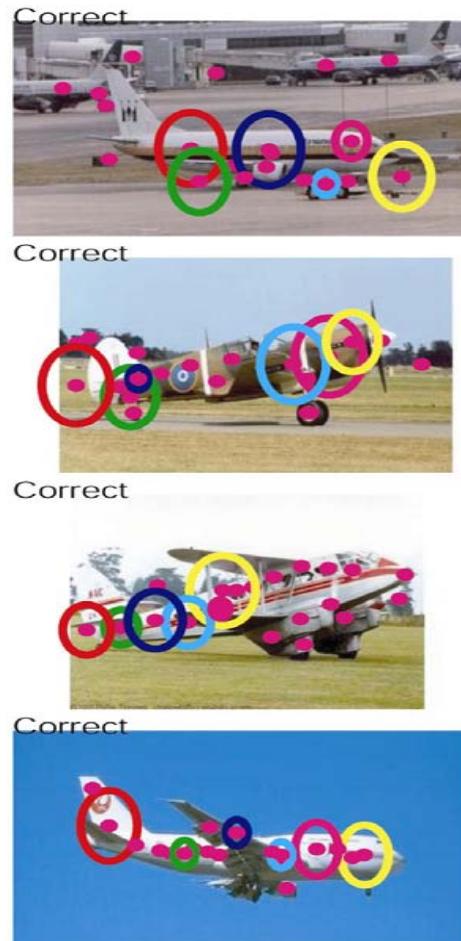
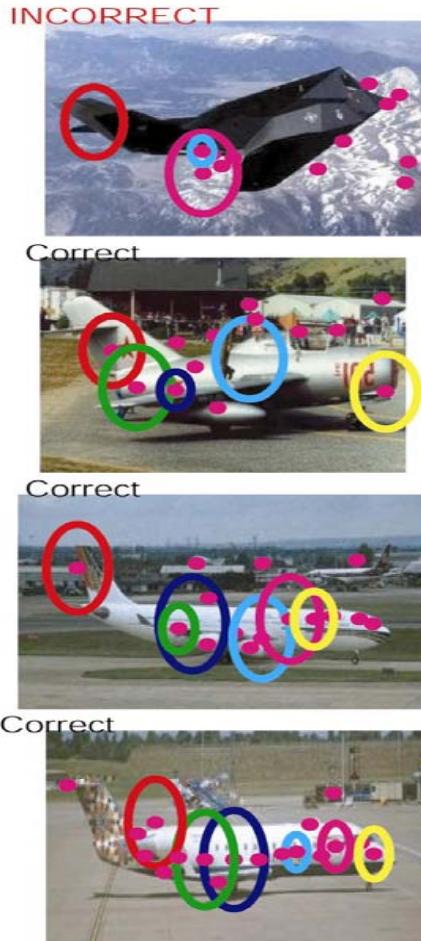
Background images evaluated with motorbike model



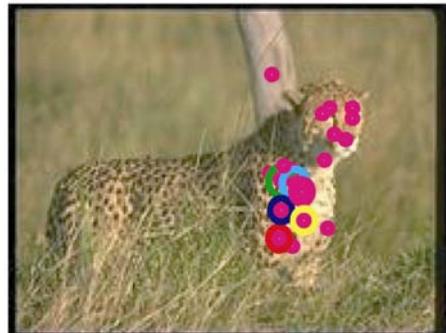
Frontal faces



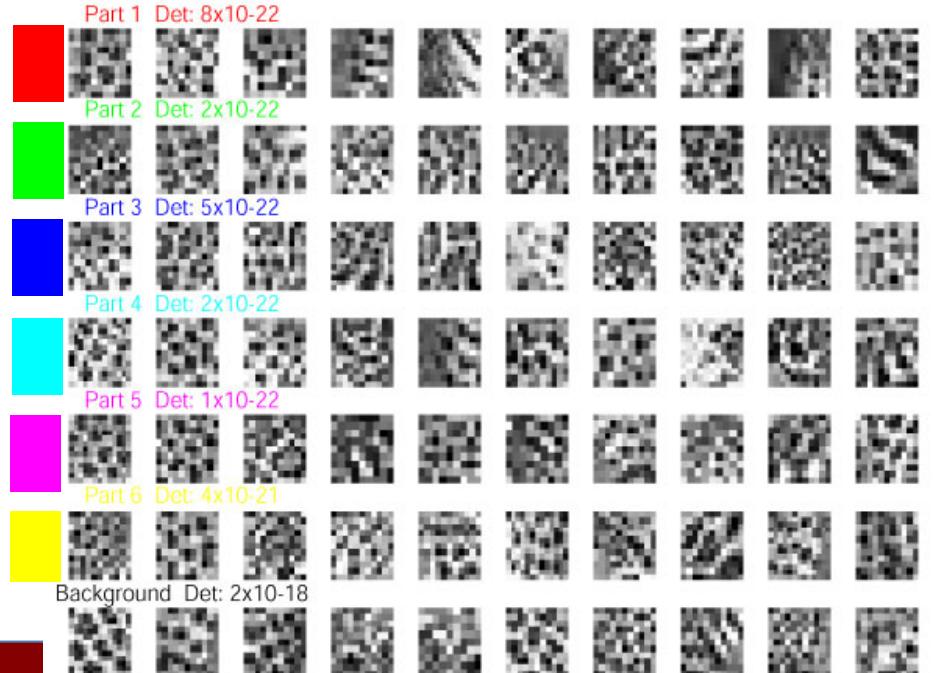
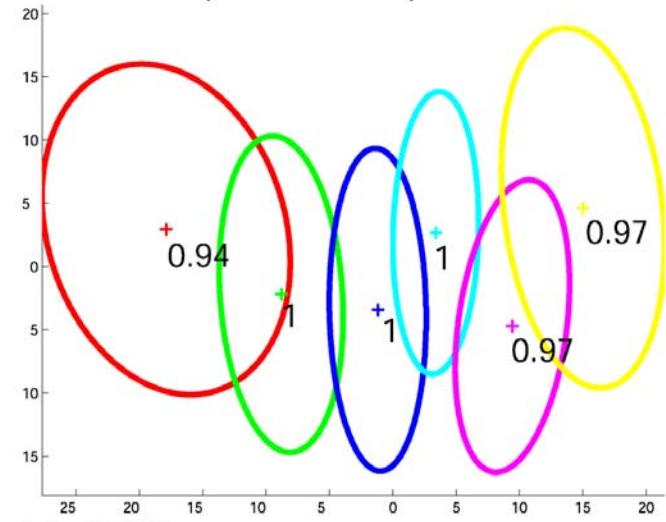
Airplanes



Spotted cats



Spotted cat shape model



Summary of results

Dataset	Fixed scale experiment	Scale invariant experiment
Motorbikes	7.5	6.7
Faces	4.6	4.6
Airplanes	9.8	7.0
Cars (Rear)	15.2	9.7
Spotted cats	10.0	10.0

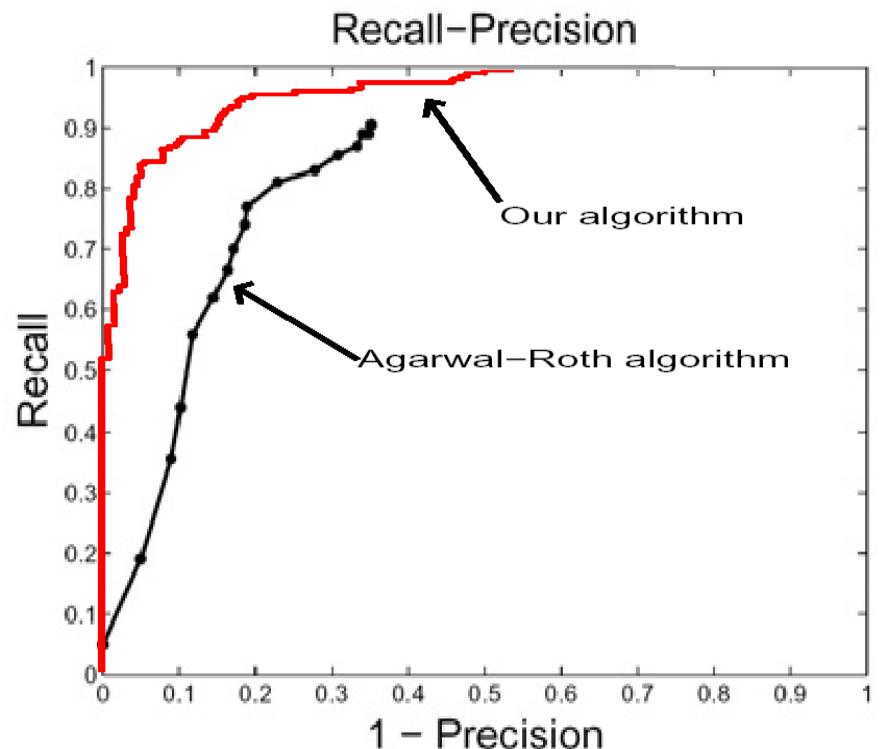
% equal error rate

Note: Within each series, same settings used for all datasets

Comparison to other methods

Dataset	Ours	Others	
Motorbikes	7.5	16.0	Weber et al. [ECCV '00]
Faces	4.6	6.0	Weber
Airplanes	9.8	32.0	Weber
Cars (Side)	11.5	21.0	Agarwal Roth [ECCV '02]

% equal error rate



Why this design?

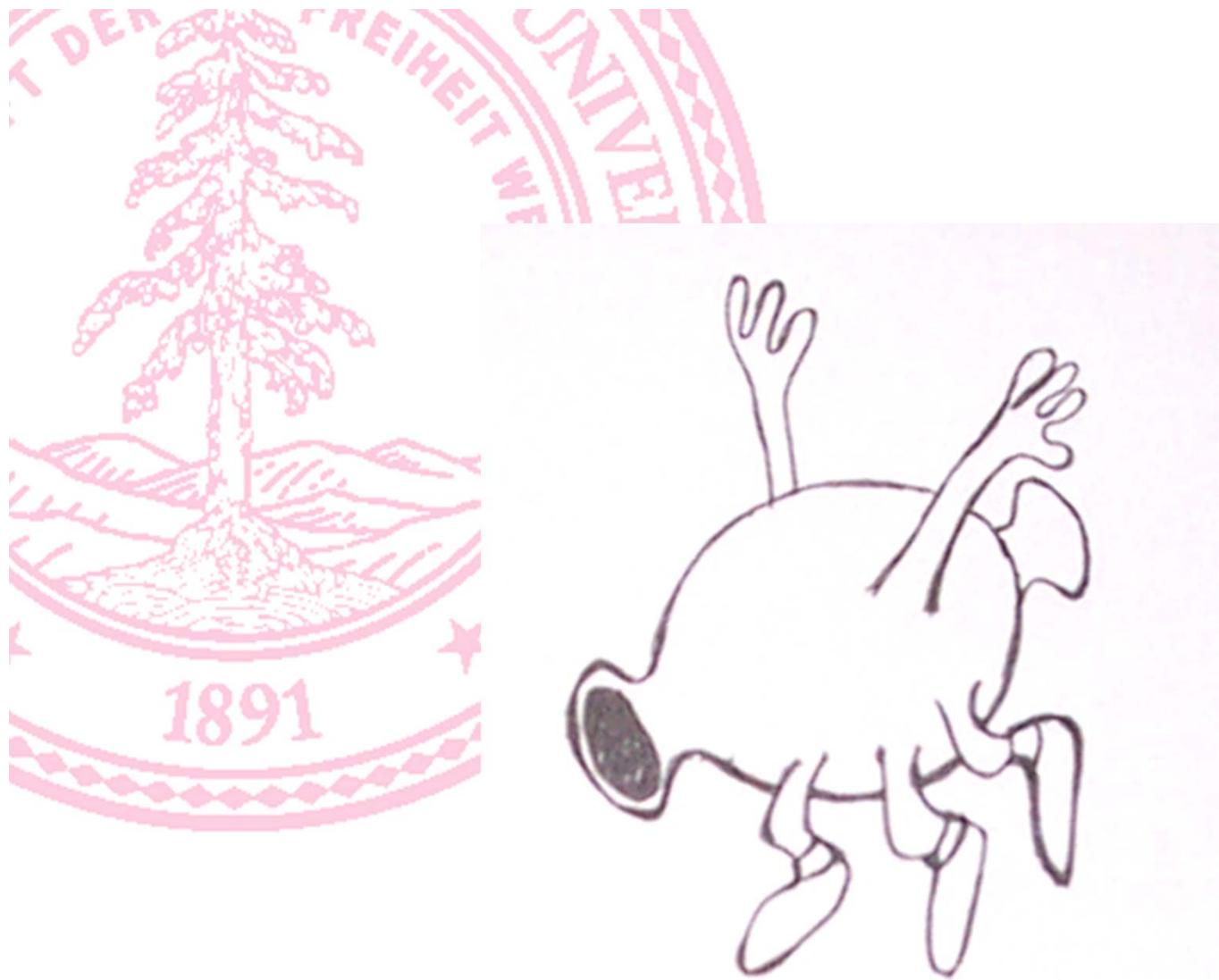
- Generic features seem to well in finding consistent parts of the object
- Some categories perform badly – different feature types needed
- Why PCA representation?
 - Tried ICA, FLD, Oriented filter responses etc.
 - But PCA worked best
- Fully probabilistic representation lets us use tools from machine learning community

What we have learned today?

- Bag of Words model (**Problem Set 4 (Q2)**)
 - Basic representation
 - Different learning and recognition algorithms
- Constellation model
 - Weakly supervised training
 - One-shot learning (supplementary materials)
- (**Problem Set 4 (Q1)**)

Supplementary materials

- One-Shot learning using Constellation Model



S. Savarese, 2003



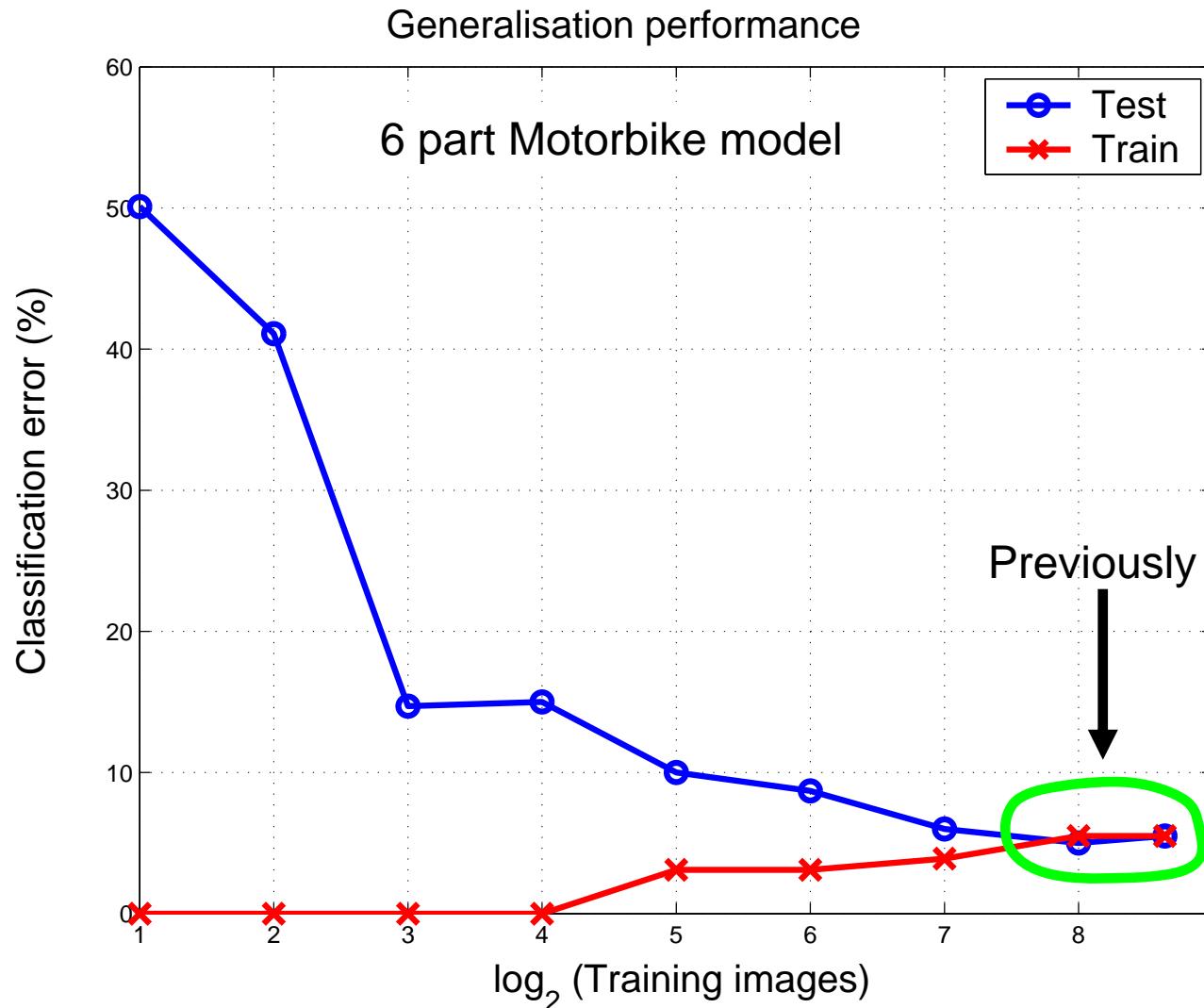
One-Shot learning

Fei-Fei et. al.

ICCV '03, PAMI '06

Algorithm	Training Examples	Categories
Burl, et al. Weber, et al. Fergus, et al.	200 ~ 400	Faces, Motorbikes, Spotted cats, Airplanes, Cars
Viola et al.	~10,000	Faces
Schneiderman, et al.	~2,000	Faces, Cars
Rowley et al.	~500	Faces

Number of training examples

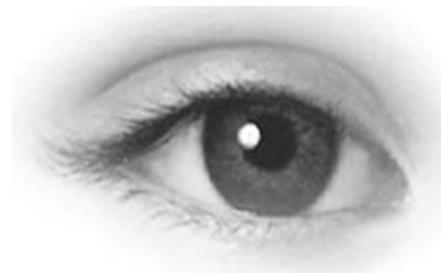


How do we do better than what statisticians have told us?

- Intuition 1: use **Prior** information
- Intuition 2: make best use of training information

Prior knowledge: means

Appearance

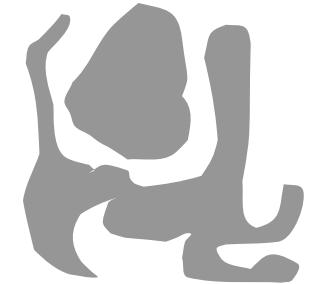
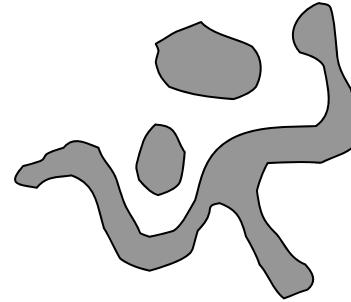
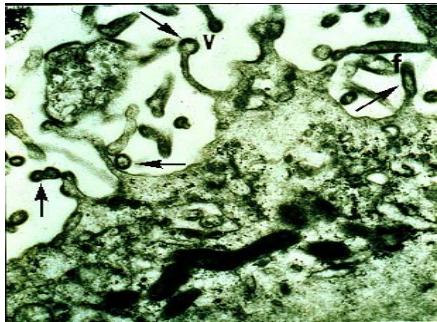


Shape



likely

unlikely



Bayesian framework

$P(\text{object} \mid \text{test, train})$ vs. $P(\text{clutter} \mid \text{test, train})$

Bayes Rule

$$p(\text{test} \mid \text{object, train}) p(\text{object})$$

Expansion by parametrization

$$\int p(\text{test} \mid \theta, \text{object}) p(\theta \mid \text{object, train}) d\theta$$

Bayesian framework

$P(\text{object} | \text{test, train})$ vs. $P(\text{clutter} | \text{test, train})$

Bayes Rule

$$p(\text{test} | \text{object, train}) p(\text{object})$$

Expansion by parametrization

$$\int p(\text{test} | \theta, \text{object}) p(\theta | \text{object, train}) d\theta$$

Previous Work: $\delta(\theta^{\text{ML}})$

Bayesian framework

$P(\text{object} \mid \text{test, train})$ vs. $P(\text{clutter} \mid \text{test, train})$

Bayes Rule

$$p(\text{test} \mid \text{object, train}) p(\text{object})$$

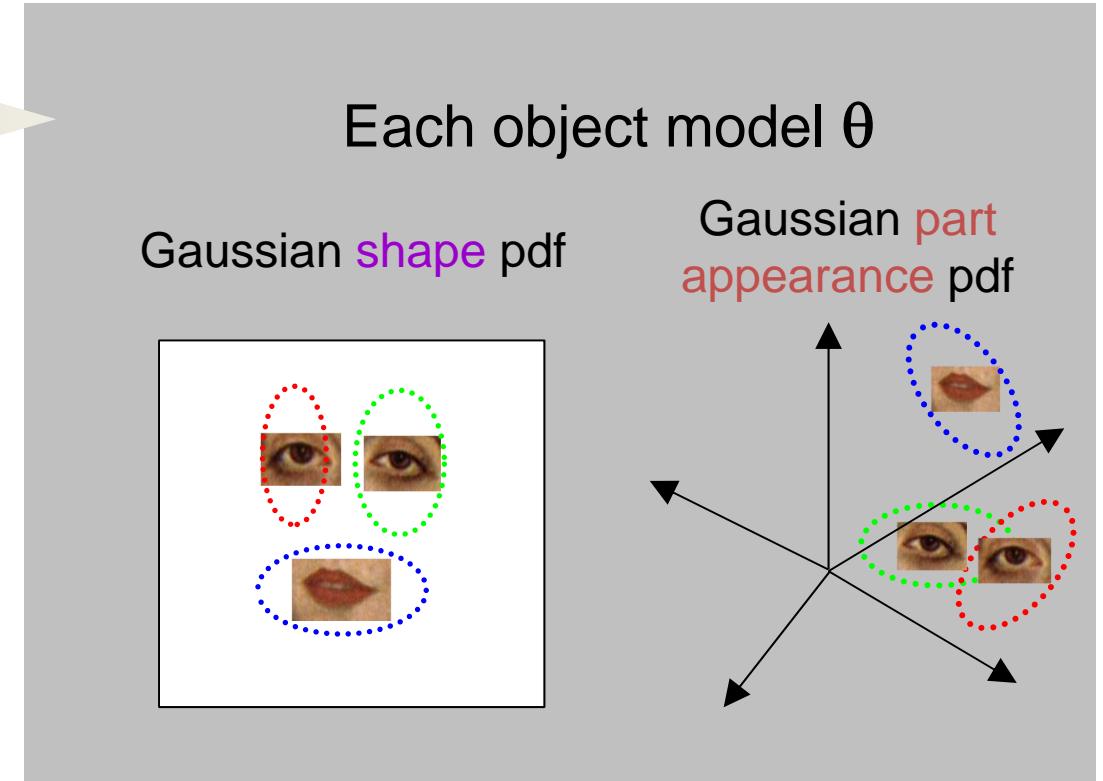
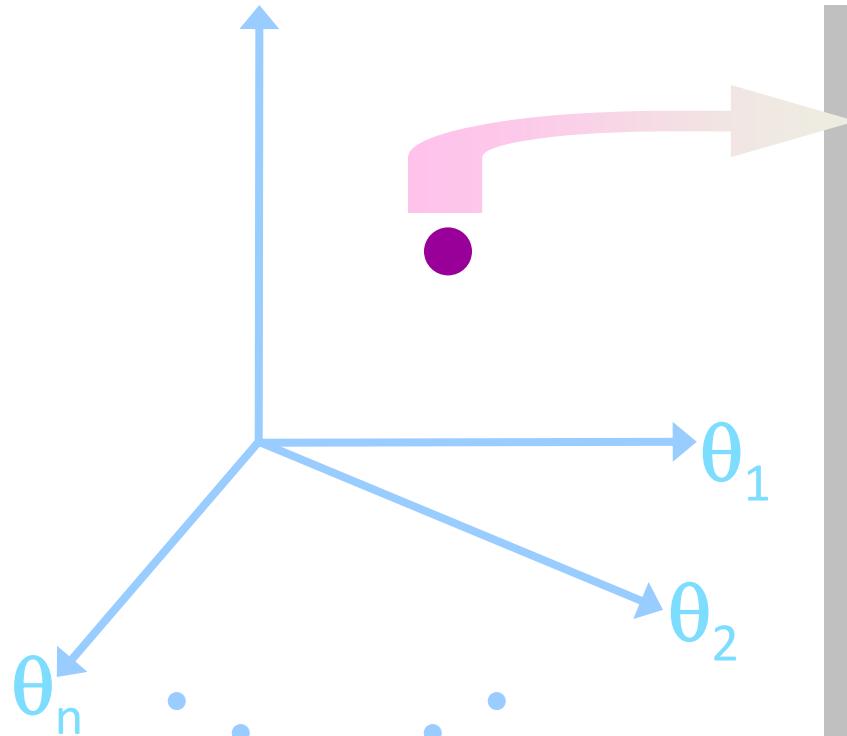
Expansion by parametrization

$$\int p(\text{test} \mid \theta, \text{object}) p(\theta \mid \text{object, train}) d\theta$$

One-Shot learning: $p(\text{train} \mid \theta, \text{object}) p(\theta)$

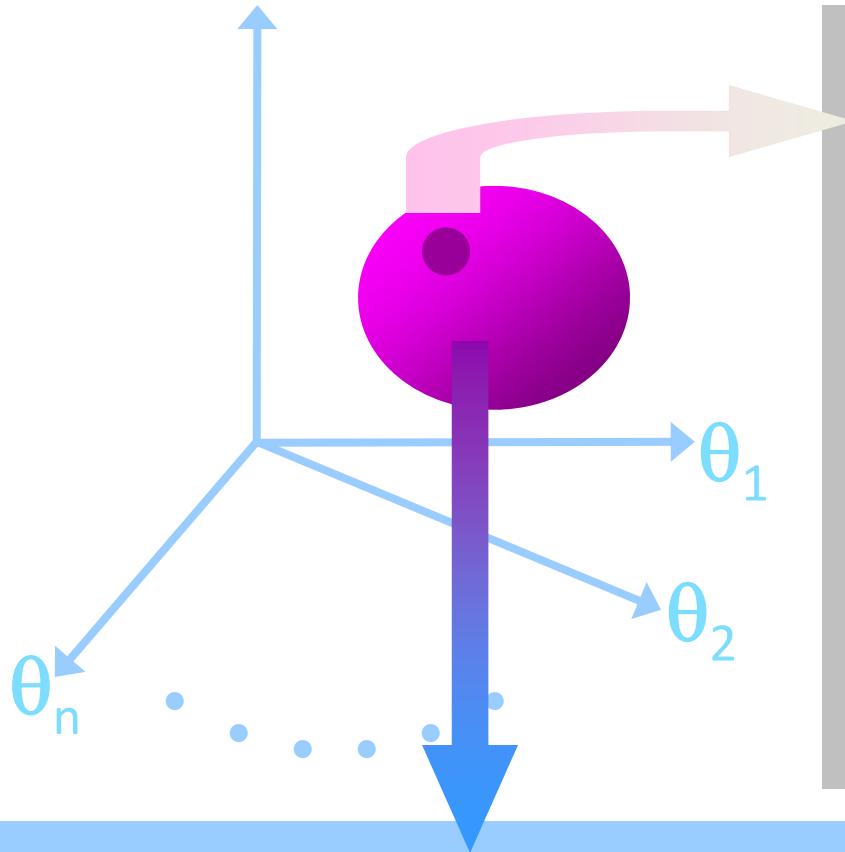
Model Structure

model (θ) space



Model Structure

model (θ) space

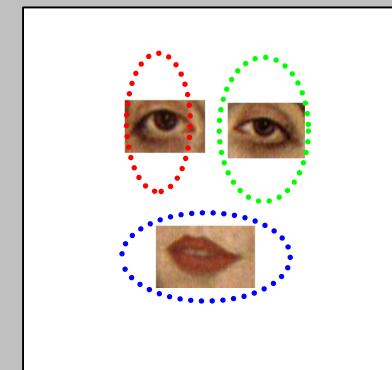


model distribution: $p(\theta)$

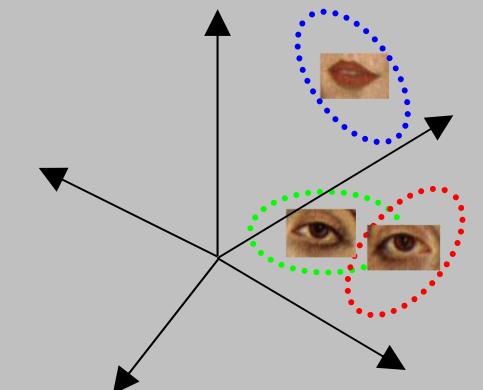
- conjugate distribution of $p(\text{train}|\theta, \text{object})$

Each object model θ

Gaussian shape pdf



Gaussian part appearance pdf

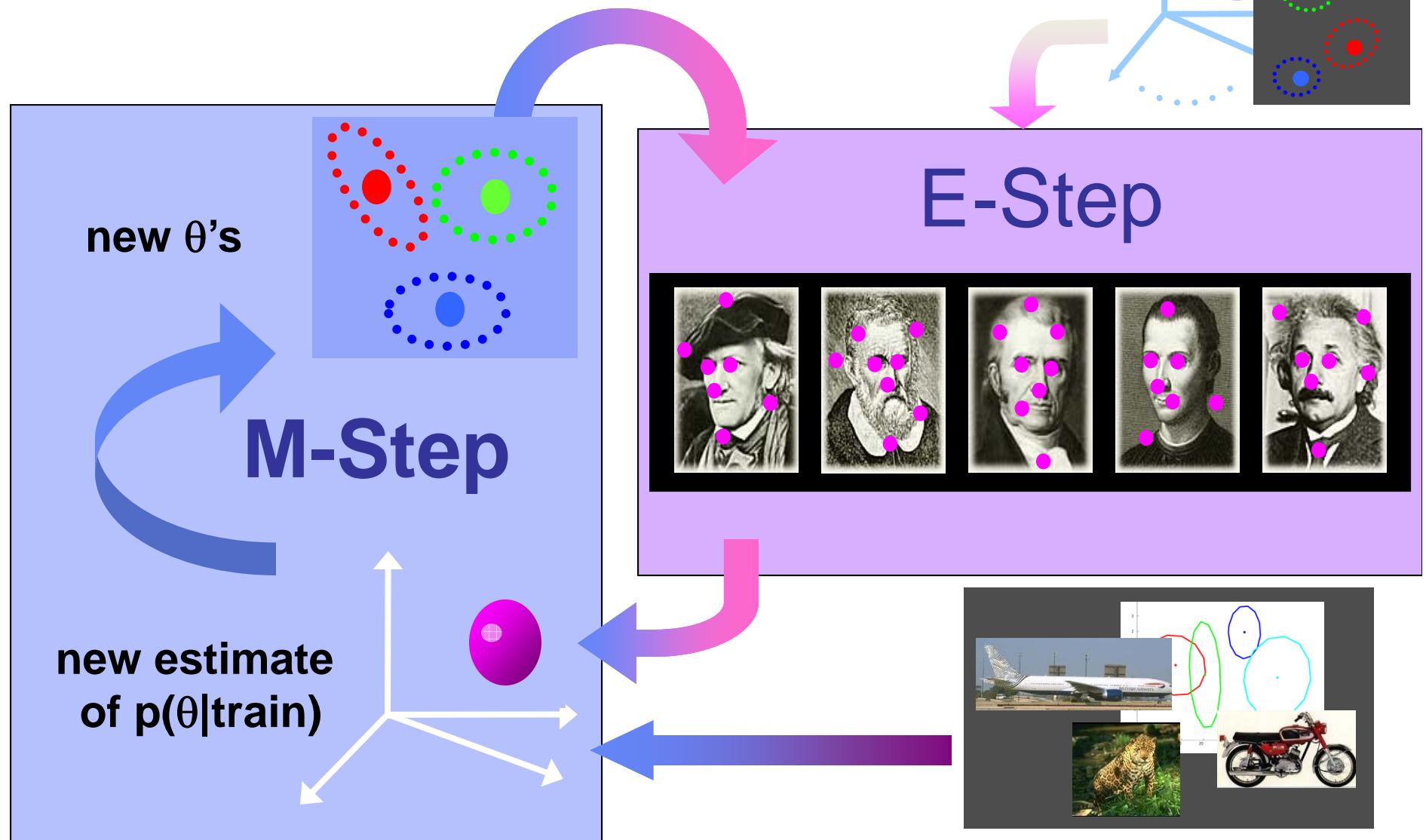


Learning Model Distribution

$$p(\theta | \text{object, train}) \propto p(\text{train} | \theta, \text{object}) p(\theta)$$

- use **Prior** information
- Bayesian learning
 - marginalize over theta
 - ❖ **Variational EM** (Attias, Hinton, Minka, etc.)

Variational EM



Experiments

Training:

1- 6 randomly
drawn images

Testing:

50 fg/ 50 bg images
object present/absent

Datasets



faces



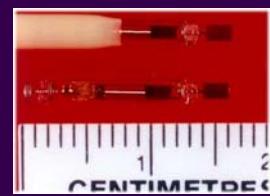
airplanes



spotted cats



motorbikes



Faces



Motorbikes



Airplanes



Spotted cats



117

14-Nov-11

Experiments: obtaining priors



airplanes

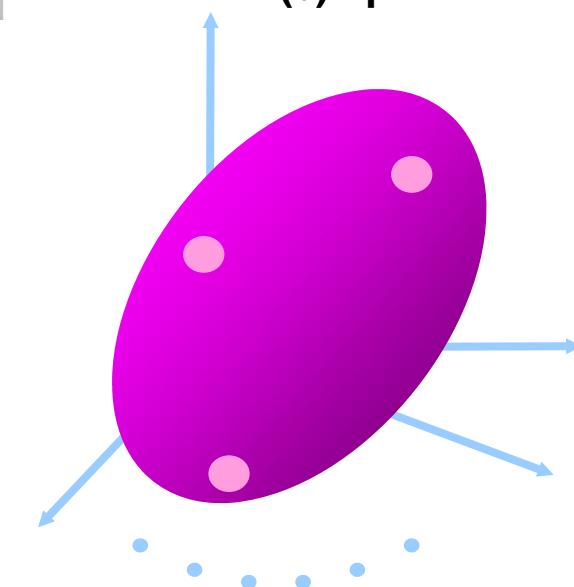


spotted cats



motorbikes

model (θ) space



faces

Experiments: obtaining priors



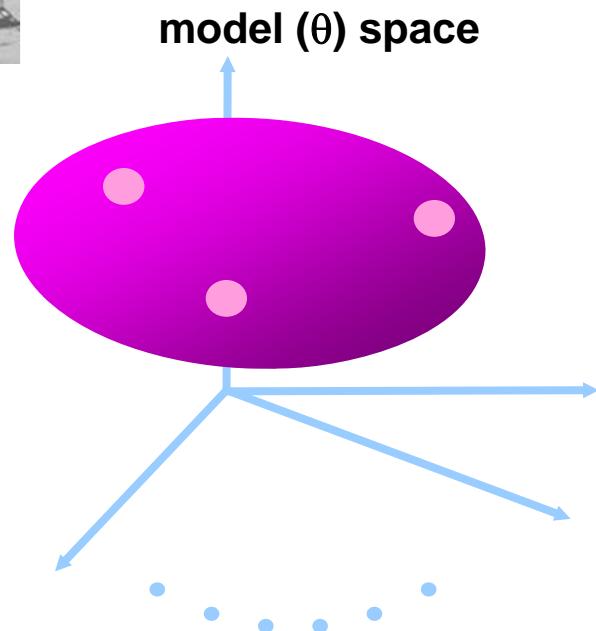
airplanes



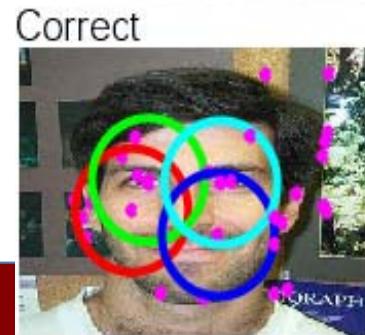
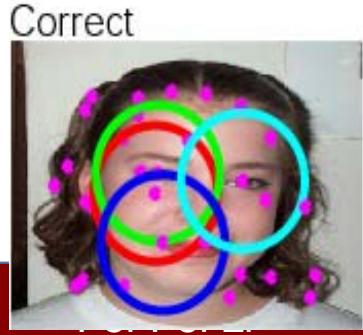
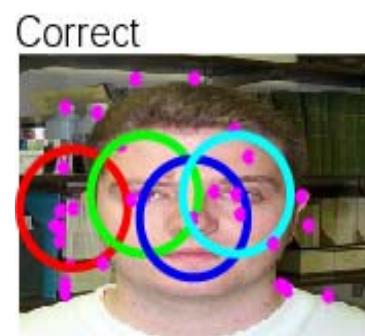
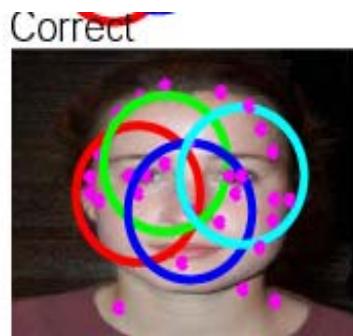
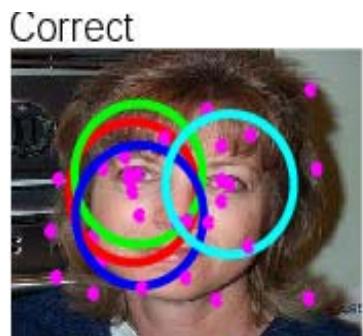
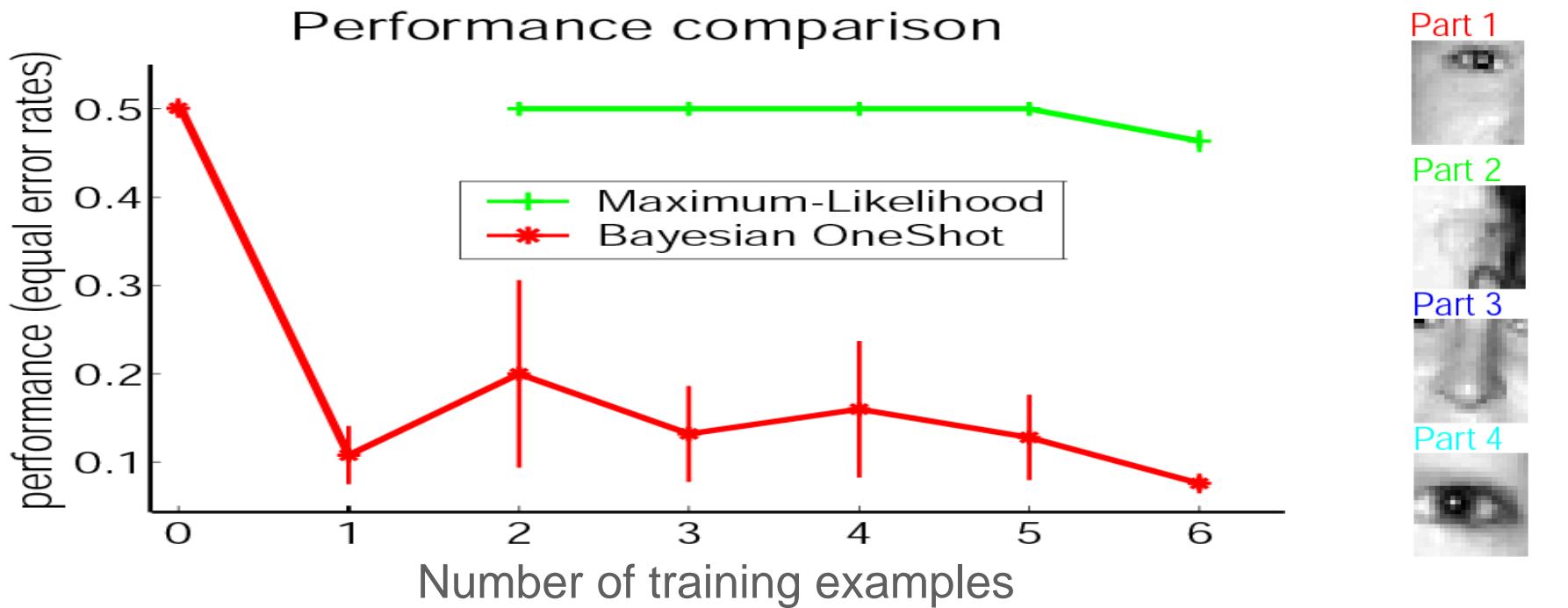
faces



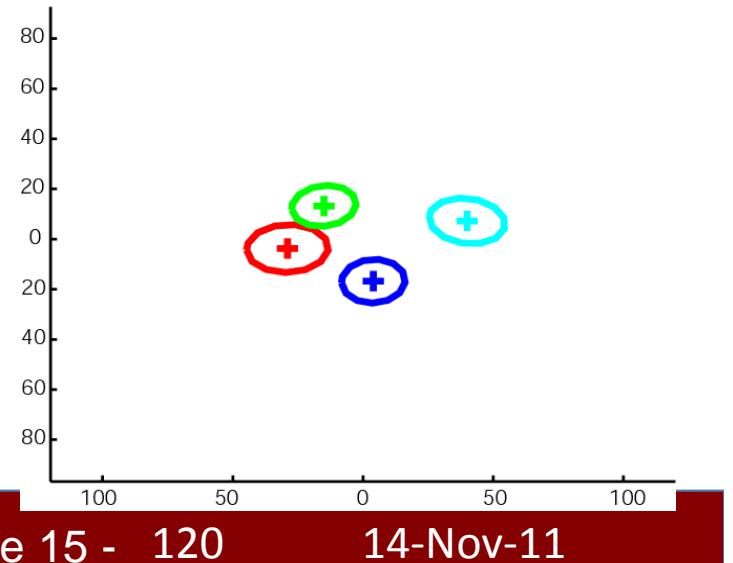
motorbikes

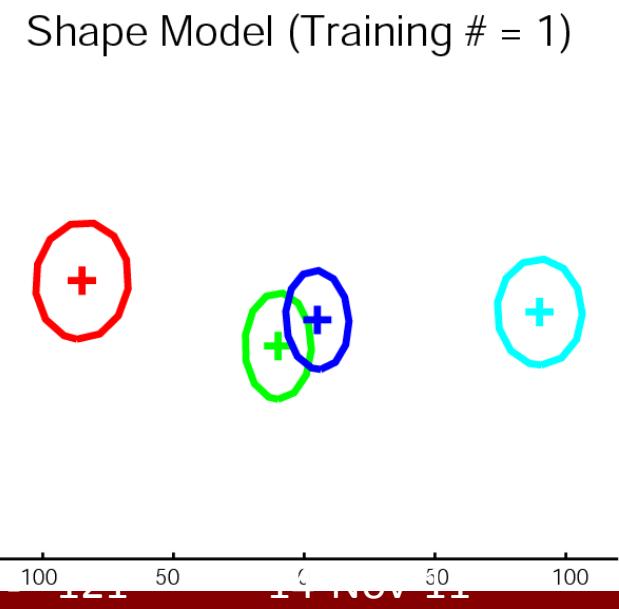
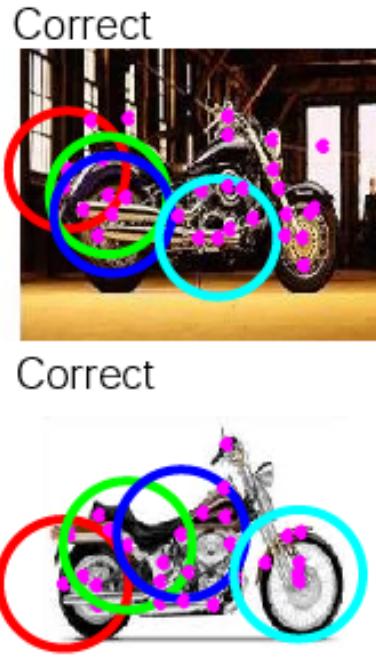
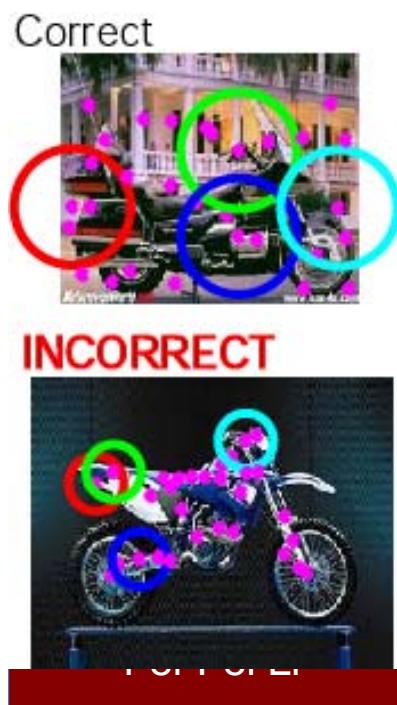
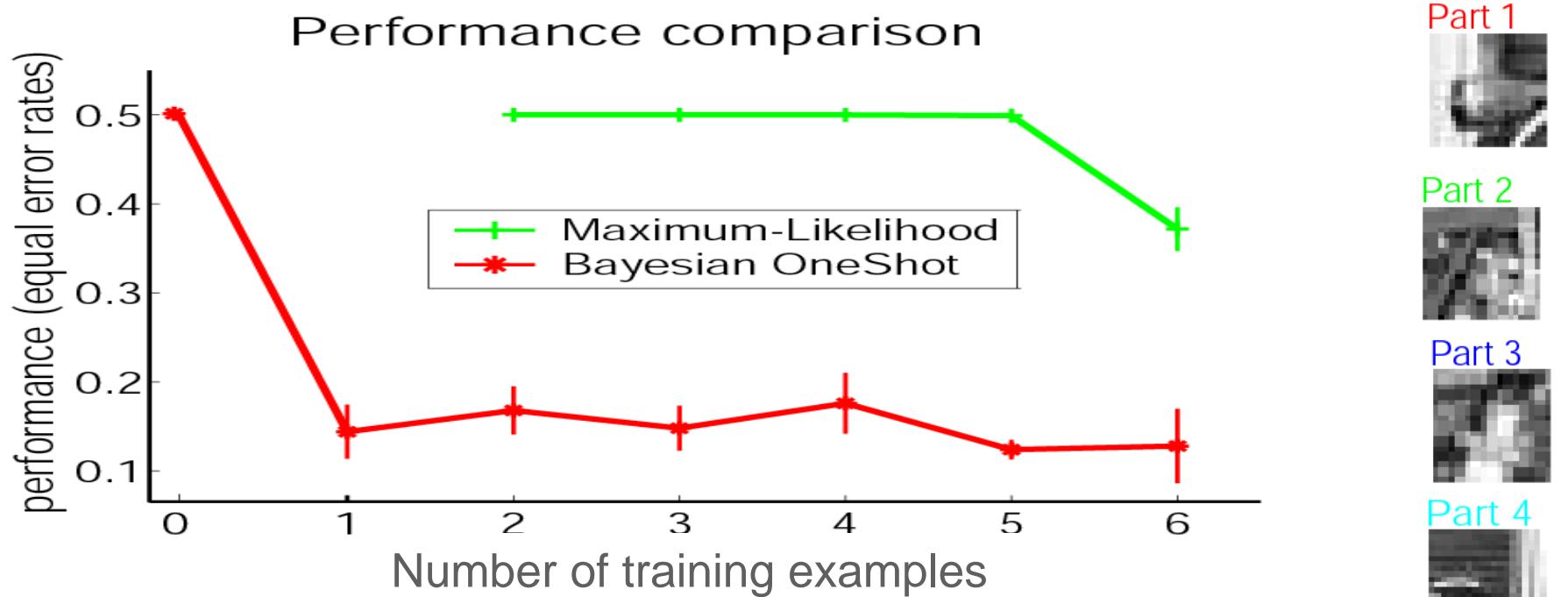


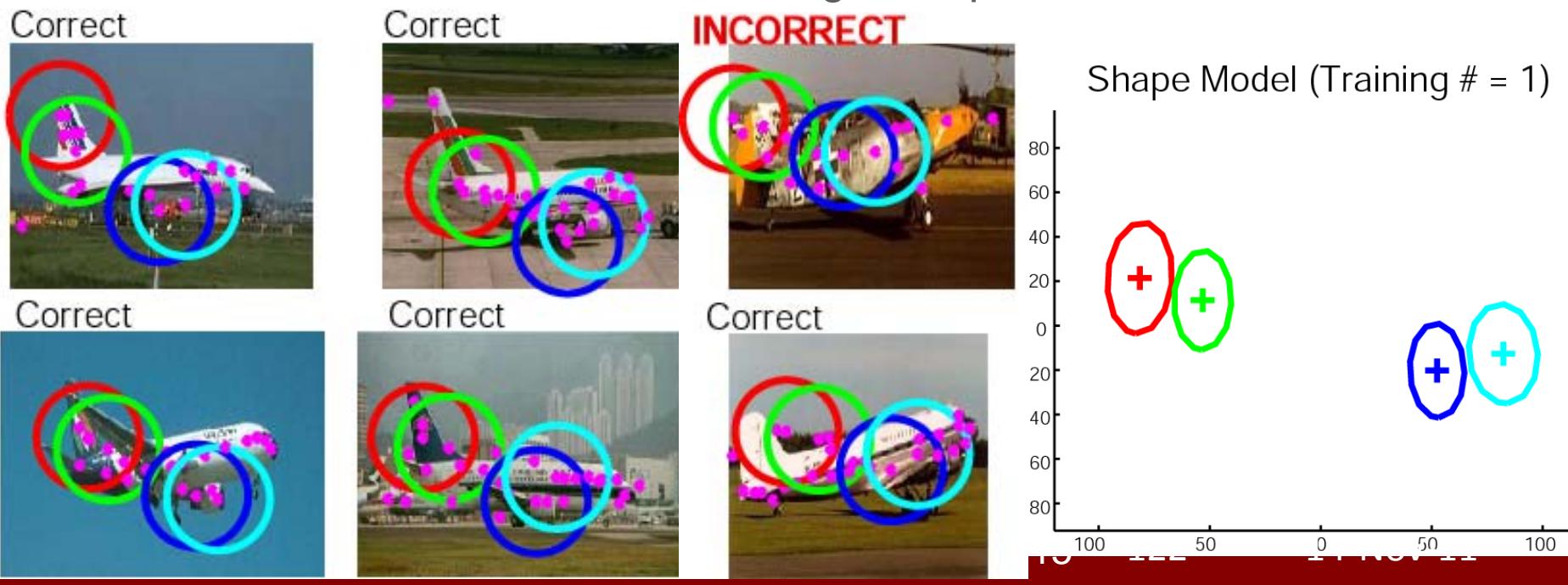
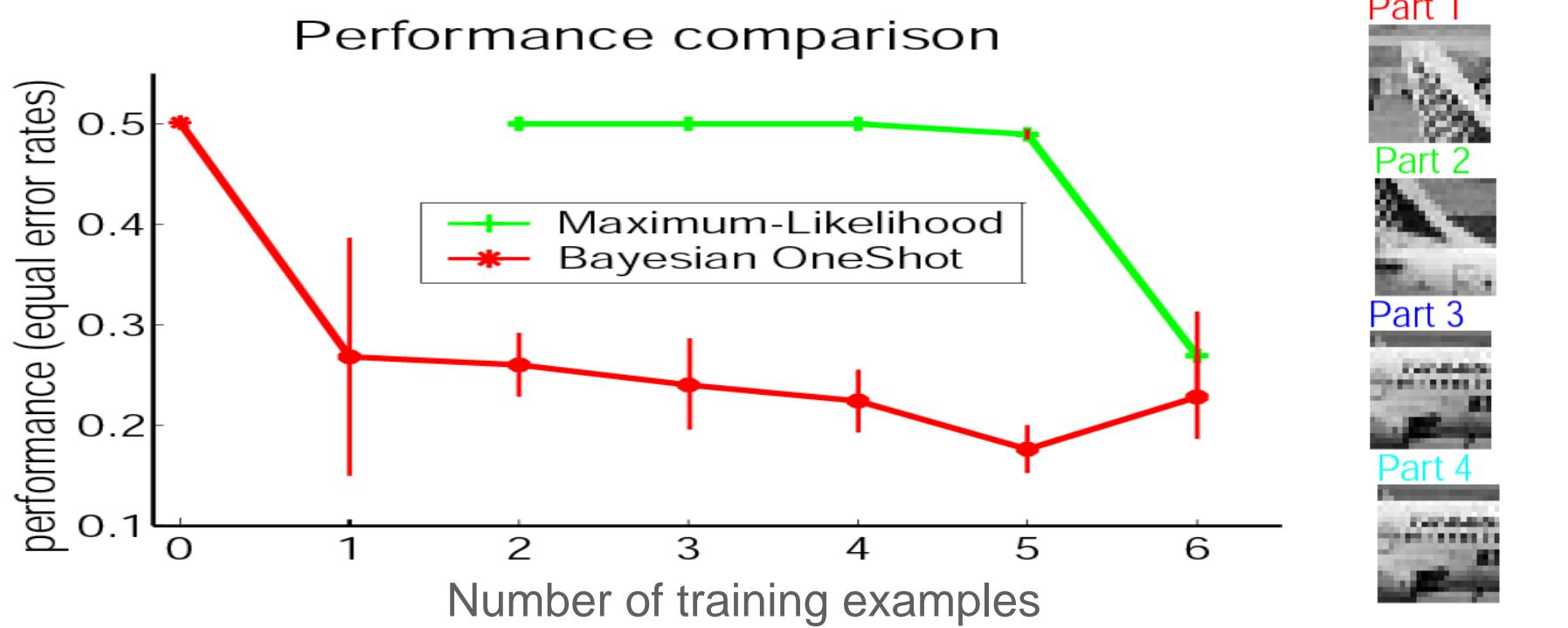
spotted cats

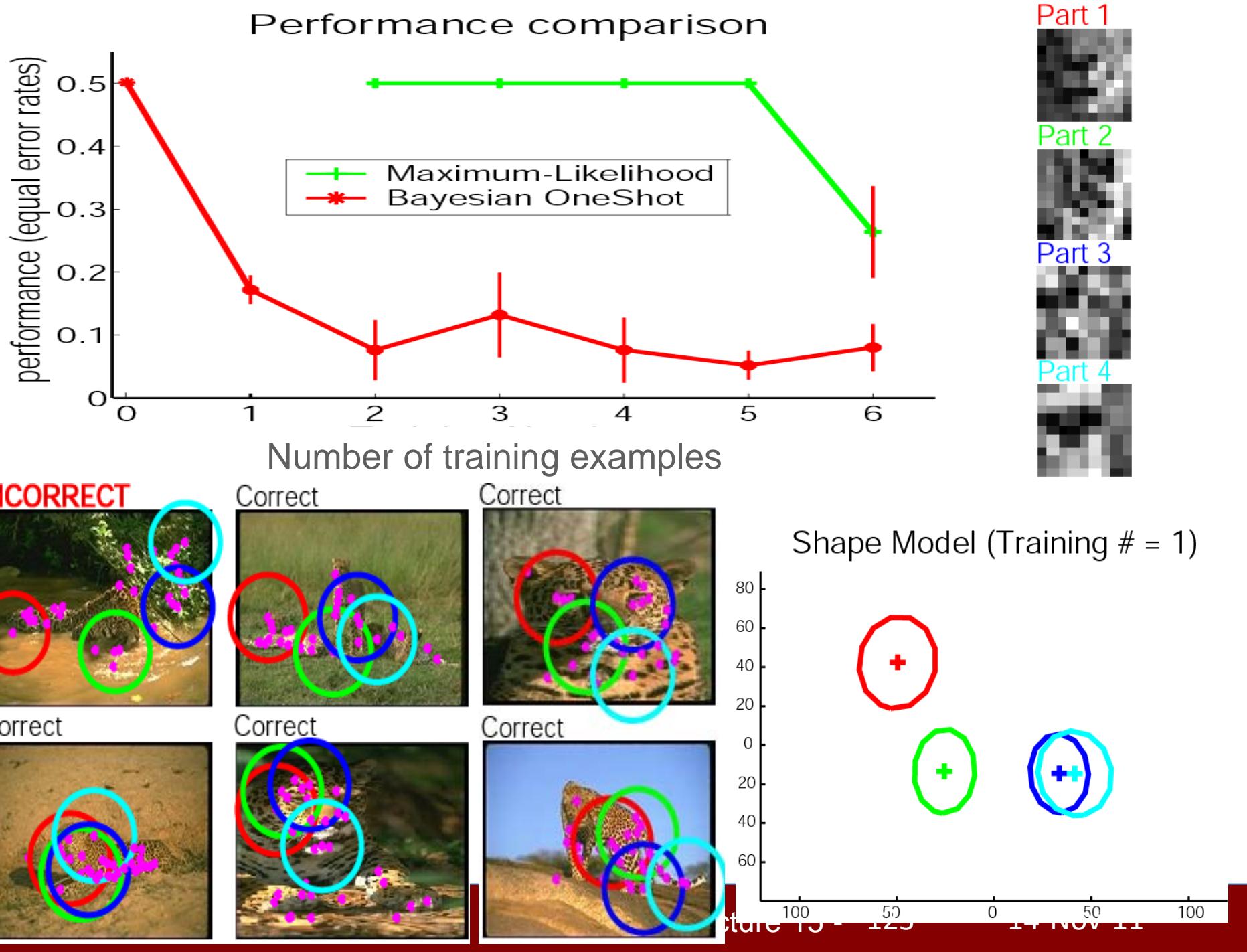


Shape Model (Training # = 1)









Algorithm	Training Examples	Categories	Results(error)
Burl, et al. Weber, et al. Fergus, et al.	200 ~ 400	Faces, Motorbikes, Spotted cats, Airplanes, Cars	5.6 - 10 %
Viola et al.	~10,000	Faces	7-21%
Schneiderman, et al.	~2,000	Faces, Cars	5.6 – 17%
Rowley et al.	~500	Faces	7.5 – 24.1%
Bayesian One-Shot	1 ~ 5	Faces, Motorbikes, Spotted cats, Airplanes	8 – 15 %