Lecture 13
Visual recognition

- Announcements

Lecture 13
Visual recognition

- Object classification bag of words models
  - Discriminative methods
  - Generative methods

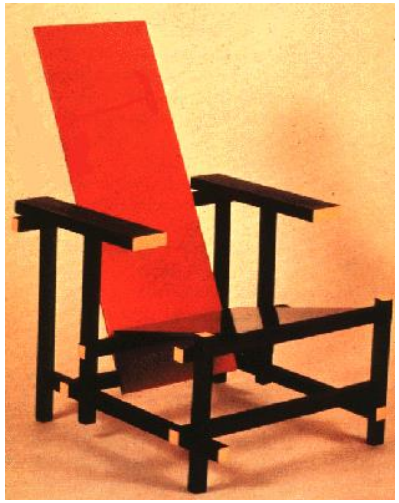
- Object classification by PCA and FLD

Challenges

Variability due to:

- View point
- Illumination
- Occlusions
- Intra-class variability

Challenges: intra-class variation

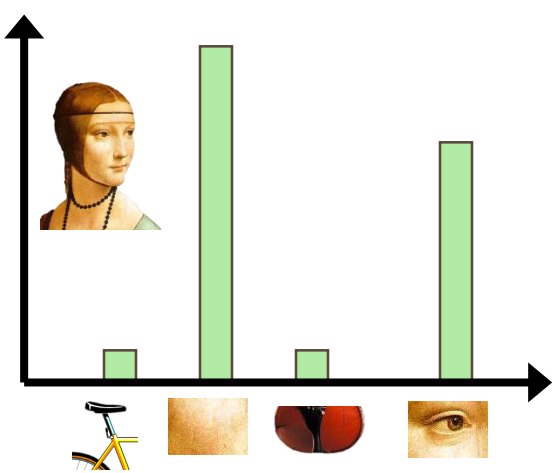


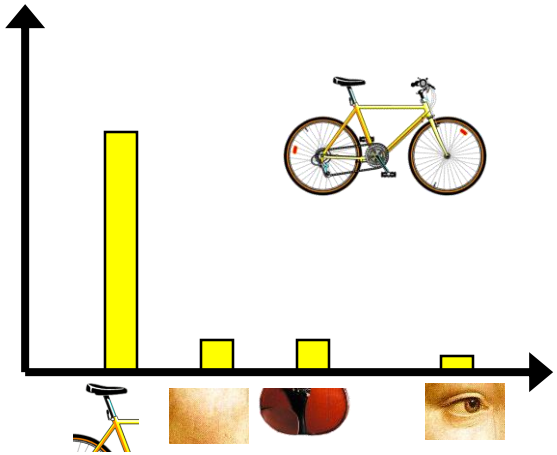
Basic properties

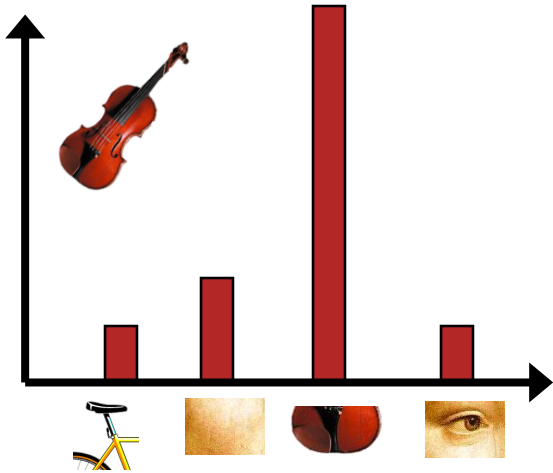
- Representation
  - How to represent an object category; which classification scheme?
- Learning
  - How to learn the classifier, given training data
- Recognition
  - How the classifier is to be used on novel data

definition of “BoW”

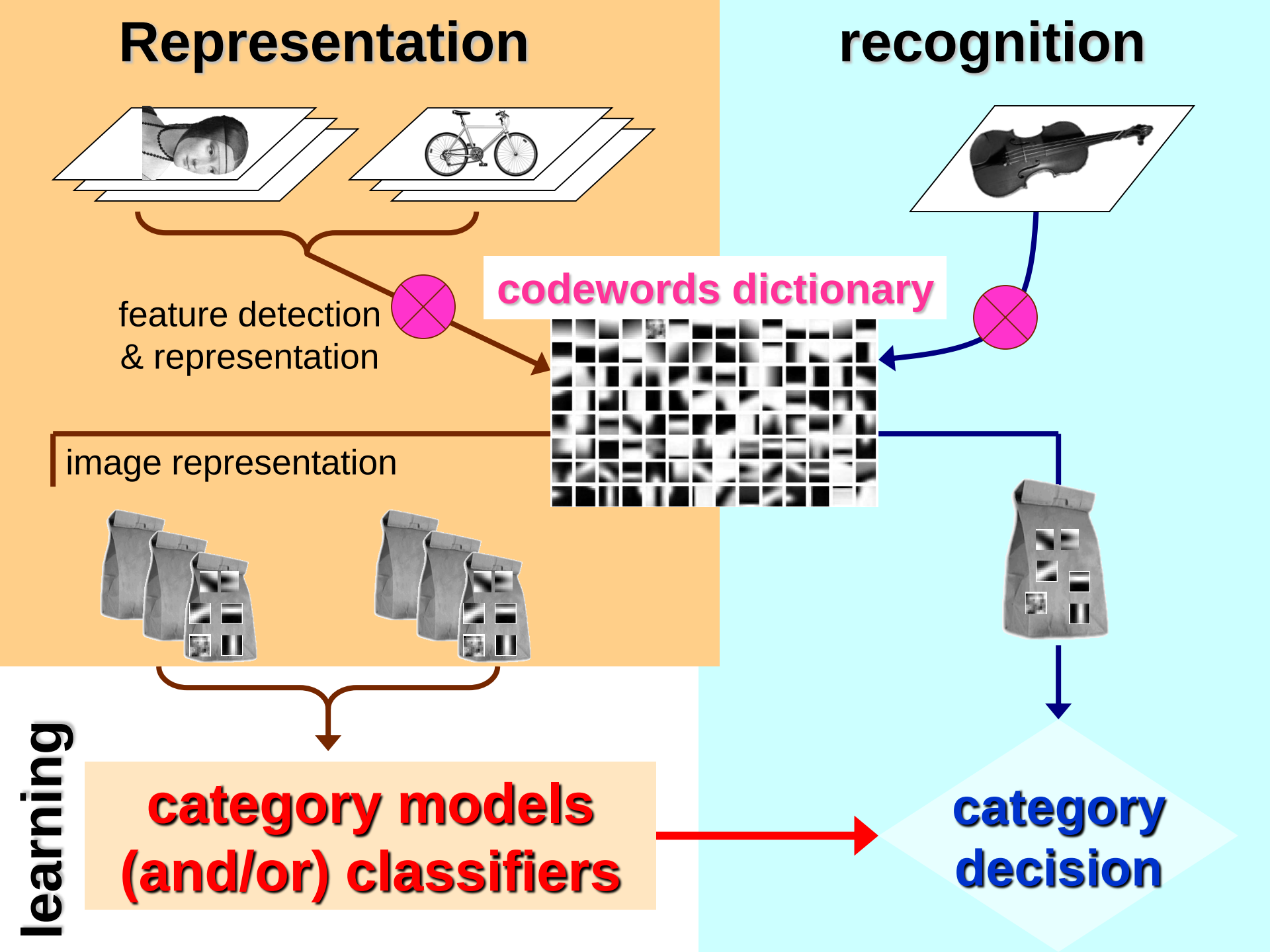
– Histogram of visual words (codewords)







codewords dictionary

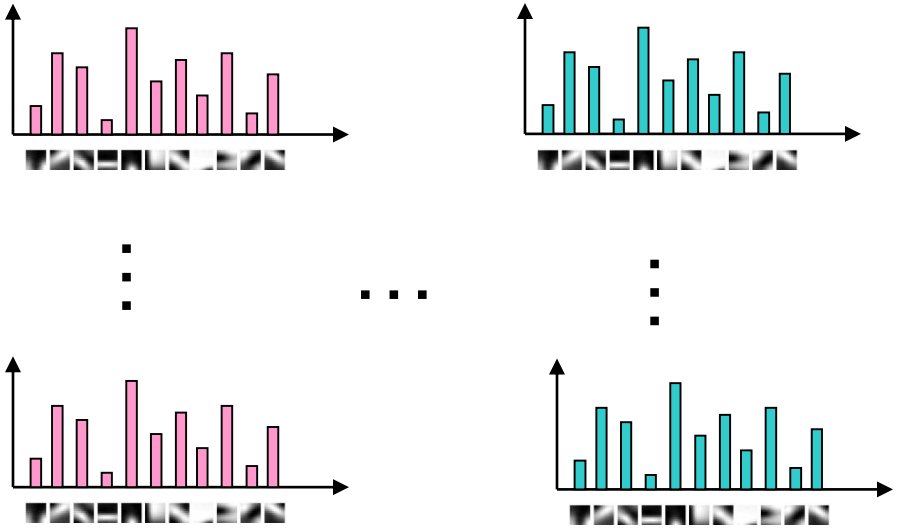

Representation
recognition
feature detection
& representation
codewords dictionary
image representation
learning
category models
(and/or) classifiers
category
decision

Classification

- Discriminative methods
  - Nearest neighbors
  - Linear classifier
  - SVM
- Generative methods

SVM classification

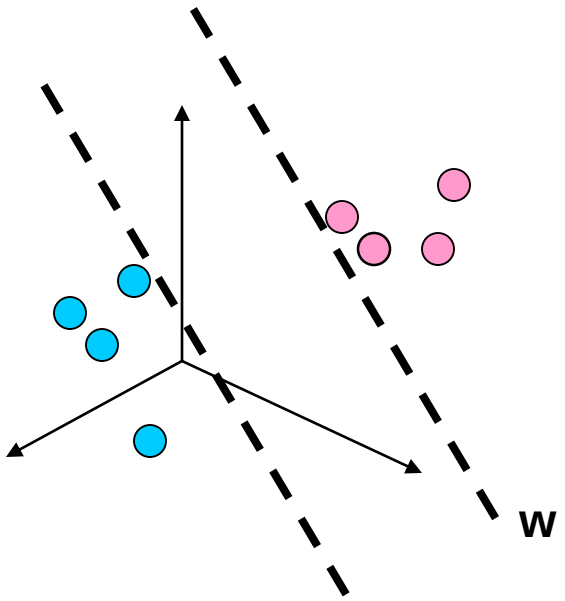
category models



Class 1

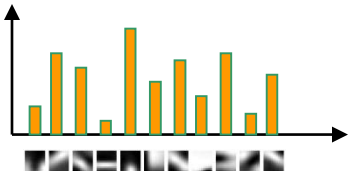
Class N

Model space



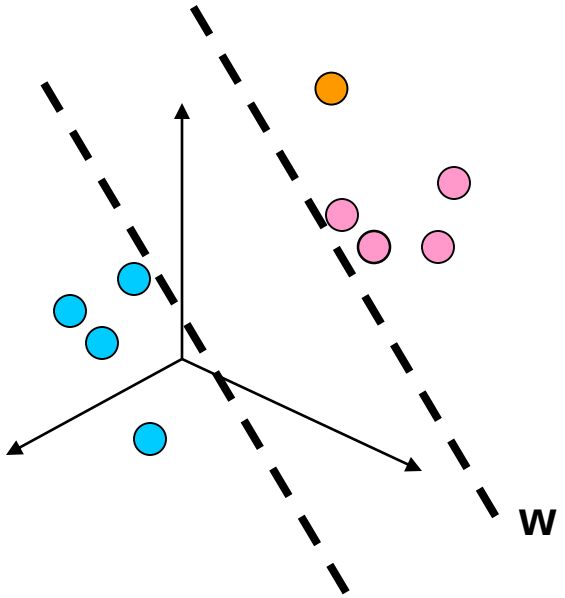
SVM classification

Query image



Winning class: pink

Model space

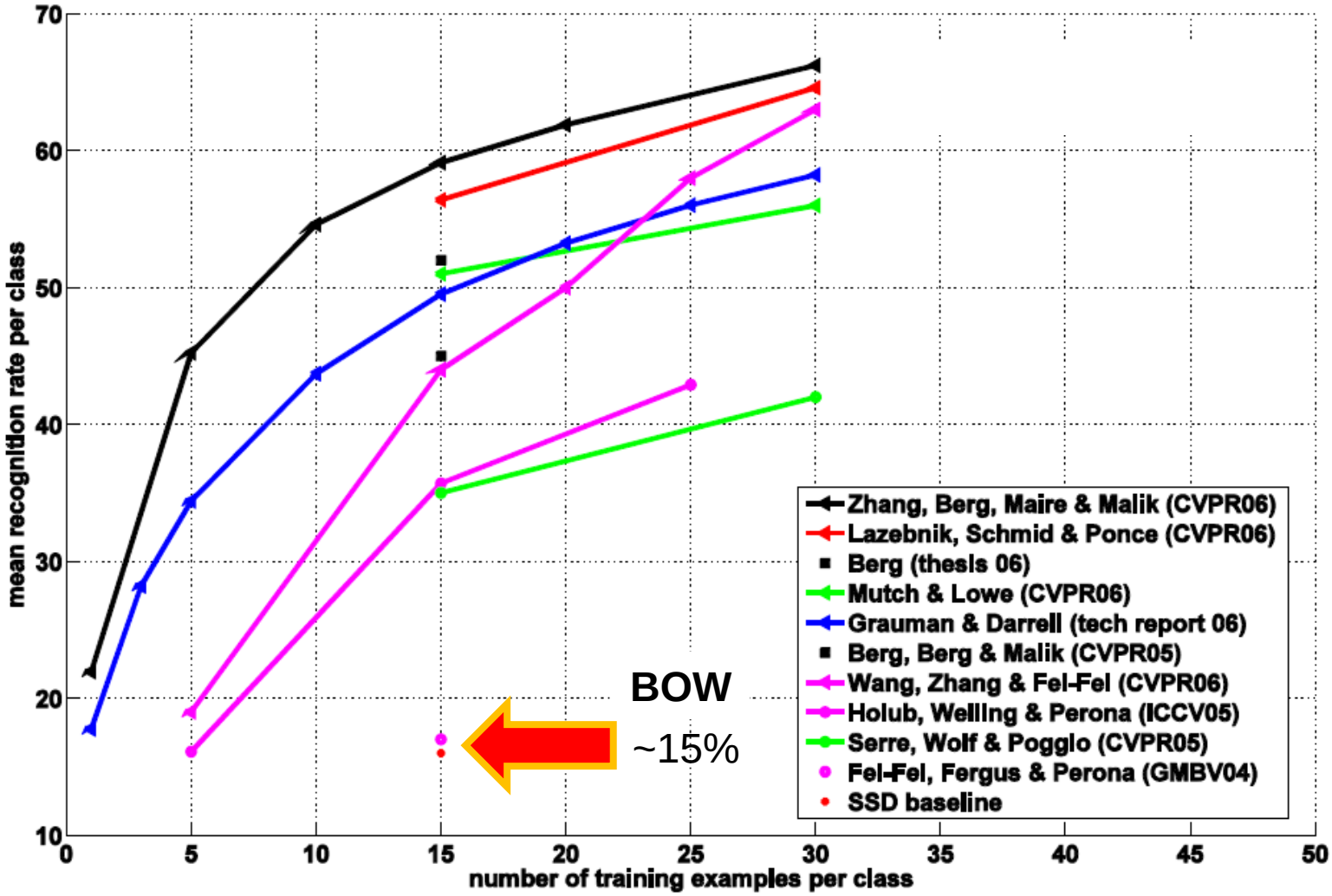


Caltech 101

Fei-Fei et al. (2004)

http://www.vision.caltech.edu/Image_Datasets/Caltech101/Caltech101.html

Caltech 101

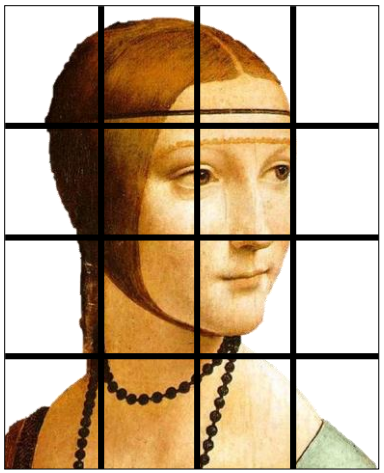

mean recognition rate per class
number of training examples per class
70
60
50
40
30
20
10
0
5
10
15
20
25
30
35
40
45
50
Zhang, Berg, Maire & Malik (CVPR06)
Lazebnik, Schmid & Ponce (CVPR06)
Berg (thesis 06)
Mutch & Lowe (CVPR06)
Grauman & Darrell (tech report 06)
Berg, Berg & Malik (CVPR05)
Wang, Zhang & Fei-Fei (CVPR06)
Holub, Welling & Perona (ICCV05)
Serre, Wolf & Poggio (CVPR05)
Fei-Fei, Fergus & Perona (GMBV04)
SSD baseline
BOW
~15%

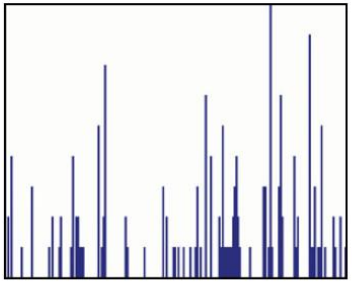
Major drawback of BOW models

Don’t capture spatial information!

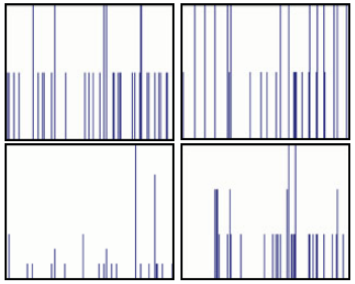
Spatial Pyramid Matching

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

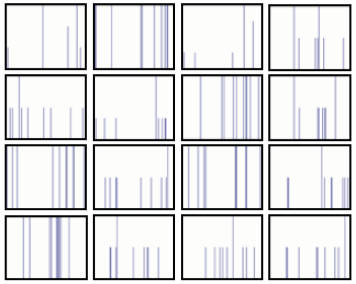




level 0



level 1



level 2

$$SPM(x_i, x) = \frac{1}{2^L} HIK_0(x_i, x) + \ldots + \frac{1}{L-l+1} HIK_l(x_i, x) + \ldots + HIK_L(x_i, x)$$

$$= I(h_1, h_2) + \frac{1}{2} I(h_1, h_2)$$

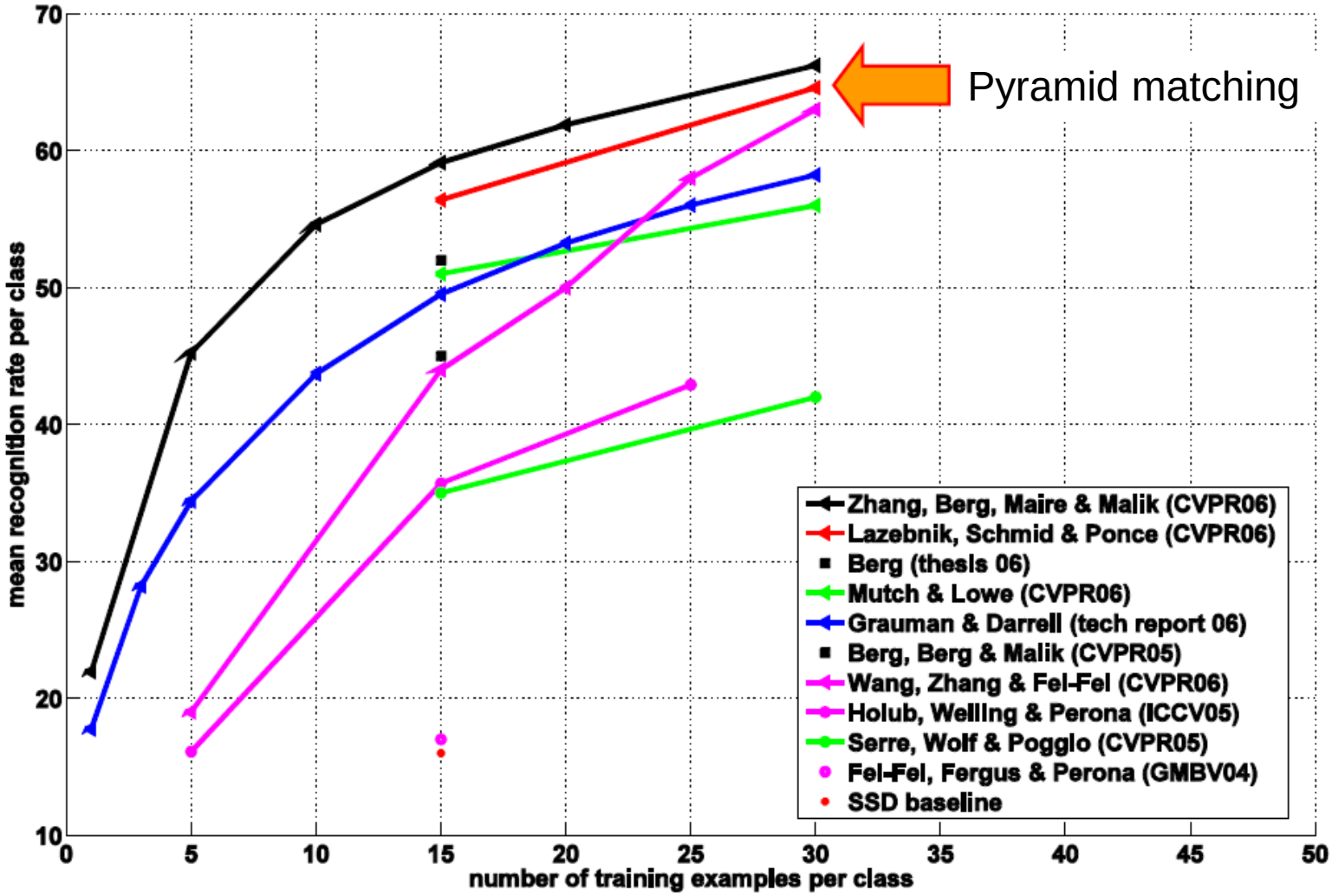
$$I(h_1, h_2) = \sum_{i=1}^{N} \min(h_1(i), h_2(i))$$

Caltech 101

Multi-class classification results (30 training images per class)

	Weak features (16)		Strong features (200)	
Level	Single-level	Pyramid	Single-level	Pyramid
0	15.5 ±0.9		41.2 ±1.2	
1	31.4 ±1.2	32.8 ±1.3	55.9 ±0.9	57.0 ±0.8
2	47.2 ±1.1	49.3 ±1.4	63.6 ±0.9	**64.6** ±0.8
3	52.2 ±0.8	**54.0** ±1.1	60.3 ±0.9	64.6 ±0.7

Caltech 101


Pyramid matching
mean recognition rate per class
number of training examples per class
70
60
50
40
30
20
10
0
5
10
15
20
25
30
35
40
45
50
Zhang, Berg, Maire & Malik (CVPR06)
Lazebnik, Schmid & Ponce (CVPR06)
Berg (thesis 06)
Mutch & Lowe (CVPR06)
Grauman & Darrell (tech report 06)
Berg, Berg & Malik (CVPR05)
Wang, Zhang & Fei-Fei (CVPR06)
Holub, Welling & Perona (ICCV05)
Serre, Wolf & Poggio (CVPR05)
Fei-Fei, Fergus & Perona (GMBV04)
SSD baseline

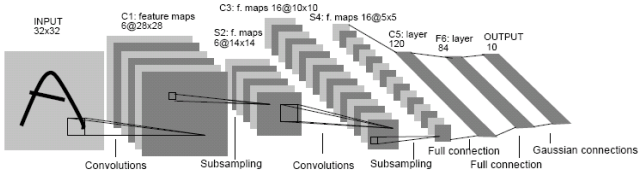
Discriminative models

Nearest neighbor

10^6 examples

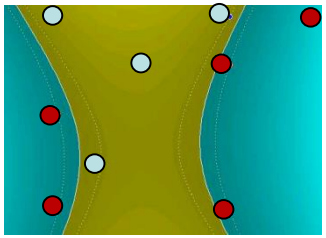
Shakhnarovich, Viola, Darrell 2003
Berg, Berg, Malik 2005...

Neural networks



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

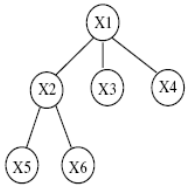
Support Vector Machines



Guyon, Vapnik, Heisele, Serre, Poggio…

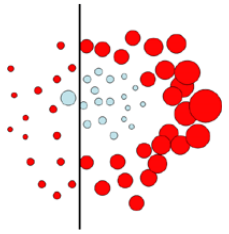
Latent SVM

Structural SVM



Felzenszwalb 00
Ramanan 03…

Boosting



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

Random forests

Courtesy of Vittorio Ferrari
Slide credit: Kristen Grauman

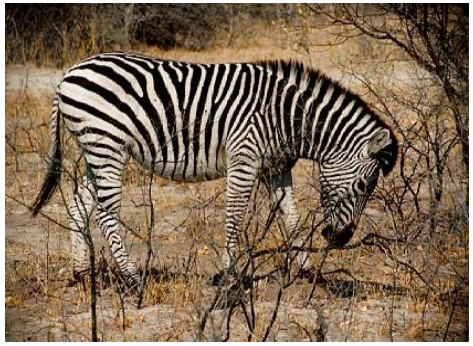
Slide adapted from Antonio Torralba

Lecture 13
Visual recognition

- Object classification bag of words models
  - Discriminative methods
  - Generative methods

- Object classification by PCA and FLD

Image classification



$$p(zebra \mid image)$$

vs.

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

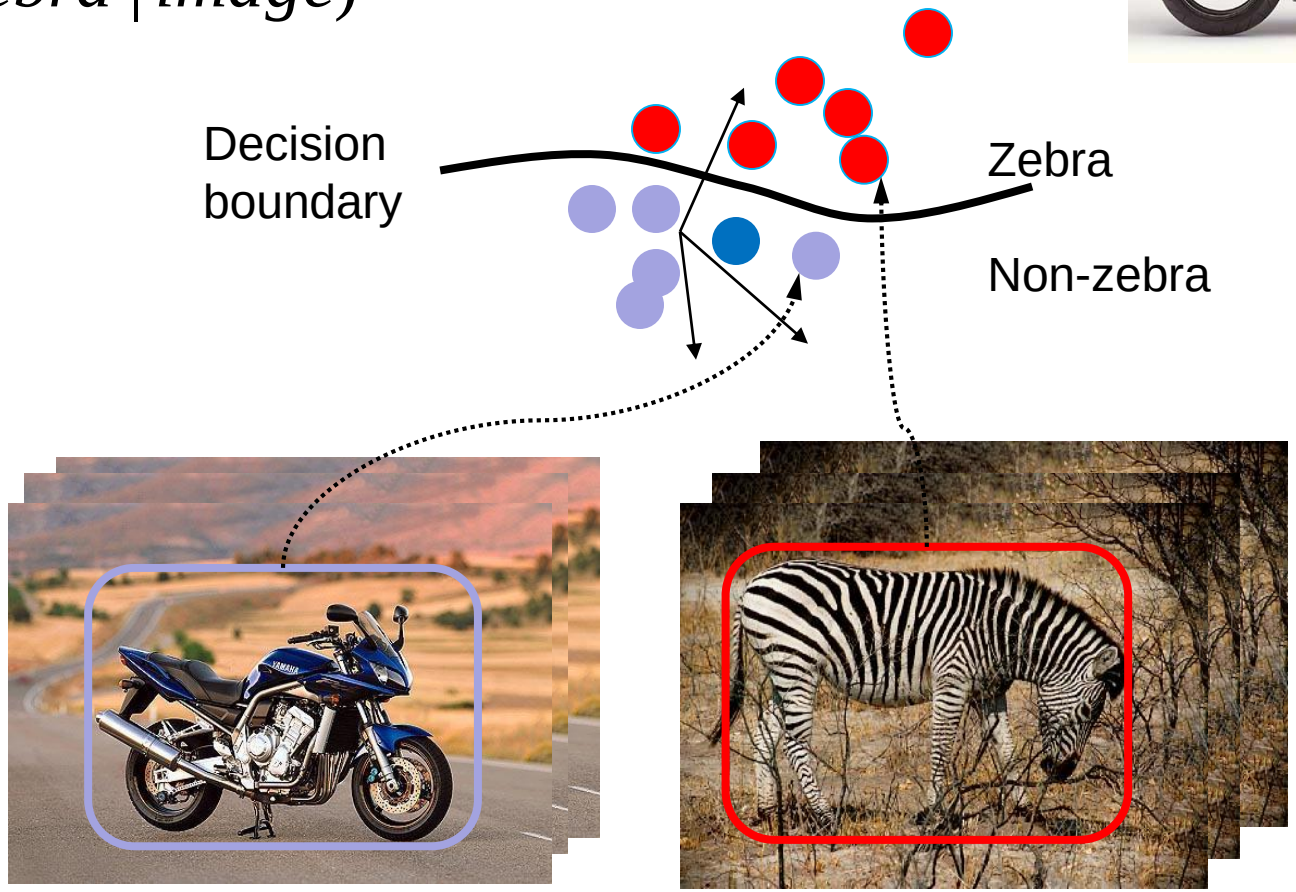
- Bayes rule:

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

Discriminative methods

- Modeling the posterior ratio:

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



Generative methods

$$p(zebra \mid image)$$

vs.

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

- Bayes rule:

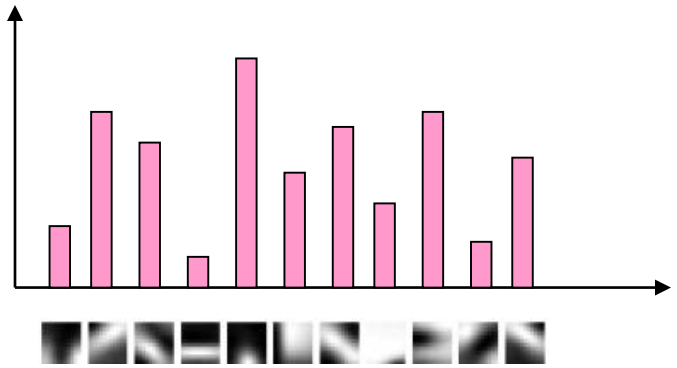
$$\underbrace{\frac{p(zebra \mid image)}{p(no\ zebra \mid image)}}_{\text{posterior ratio}} = \underbrace{\frac{p(image \mid zebra)}{p(image \mid no\ zebra)}}_{\text{likelihood ratio}} \cdot \underbrace{\frac{p(zebra)}{p(no\ zebra)}}_{\text{prior ratio}}$$

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

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

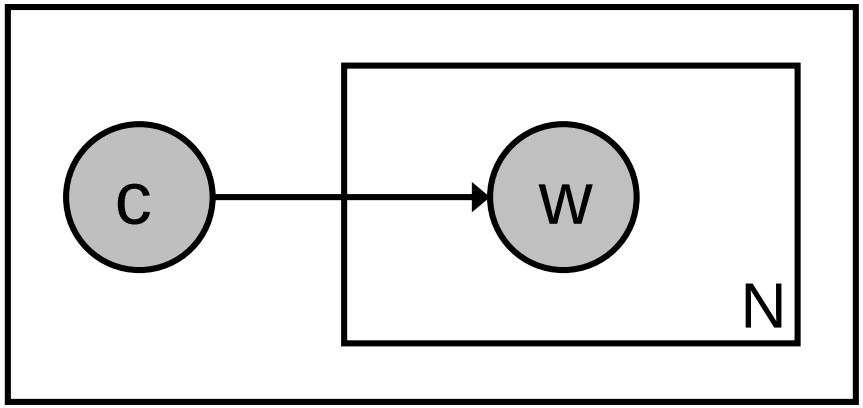
- c: category of the image

the Naïve Bayes model

$$p(c \mid w) \sim p(c)p(w \mid c) = p(c)\, p(w_1, \ldots, w_N \mid c)$$

Prior prob. of the object classes

Image likelihood given the class



the Naïve Bayes model

$$p(c \mid w) \sim p(c)p(w \mid c) = p(c)\, p(w_1, \ldots, w_N \mid c)$$

$$= p(c) \prod_{n=1}^{N} p(w_n \mid c)$$

Prior prob. of the object classes

Image likelihood given the class

Likelihood of n^{th} visual word given the class

- Assume that each feature (codewords) is conditionally independent *given the class*

$$p(w_1, \ldots, w_N \mid c) = \prod_{i=1}^{N} p(w_i \mid c)$$

the Naïve Bayes model

$$p(c \mid w) \sim p(c)p(w \mid c) = p(c)\, p(w_1, \ldots, w_N \mid c)$$

$$= p(c) \prod_{n=1}^{N} p(w_n \mid c)$$

Prior prob. of the object classes

Image likelihood given the class

Likelihood of nth visual word given the class

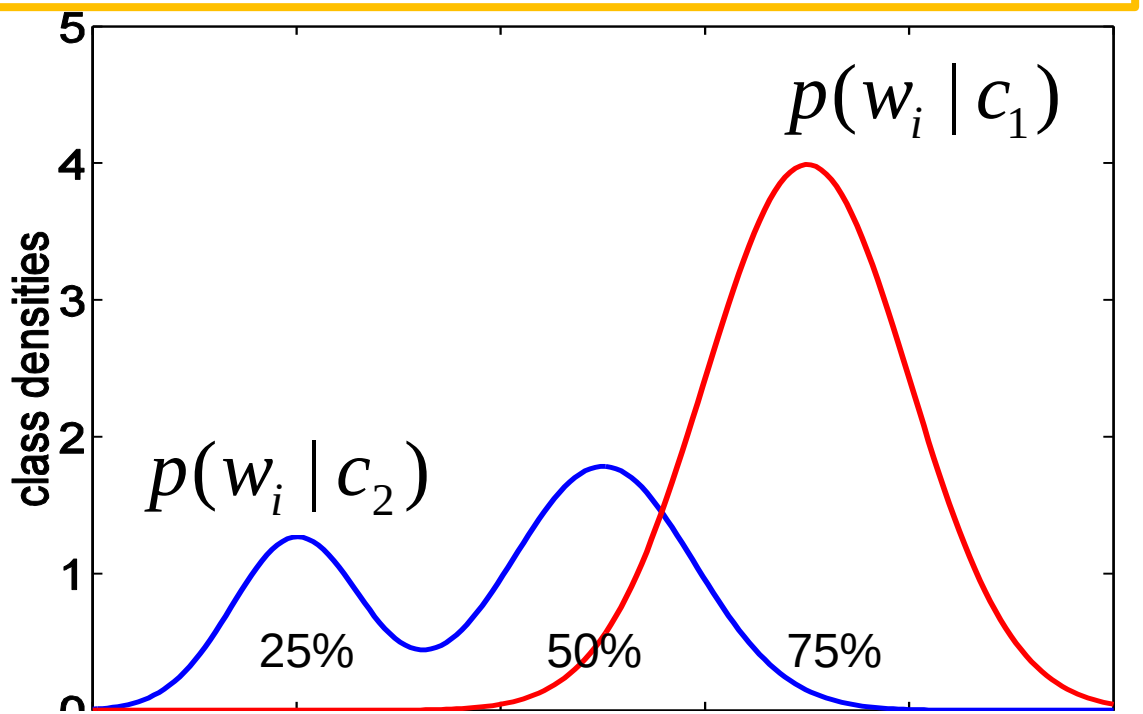
Example:

2 classes:
bananas vs oranges

Histogram of colors

Wi = number of pixels colored in yellow in the image

x-axis: percentage of pixel that are colored in yellow in the image



the Naïve Bayes model

$$p(c \mid w) \sim p(c)p(w \mid c) = p(c)\, p(w_1, \ldots, w_N \mid c)$$

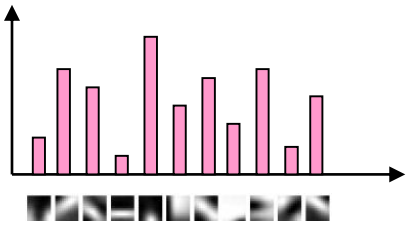
$$= p(c)\prod_{n=1}^{N} p(w_n \mid c)$$

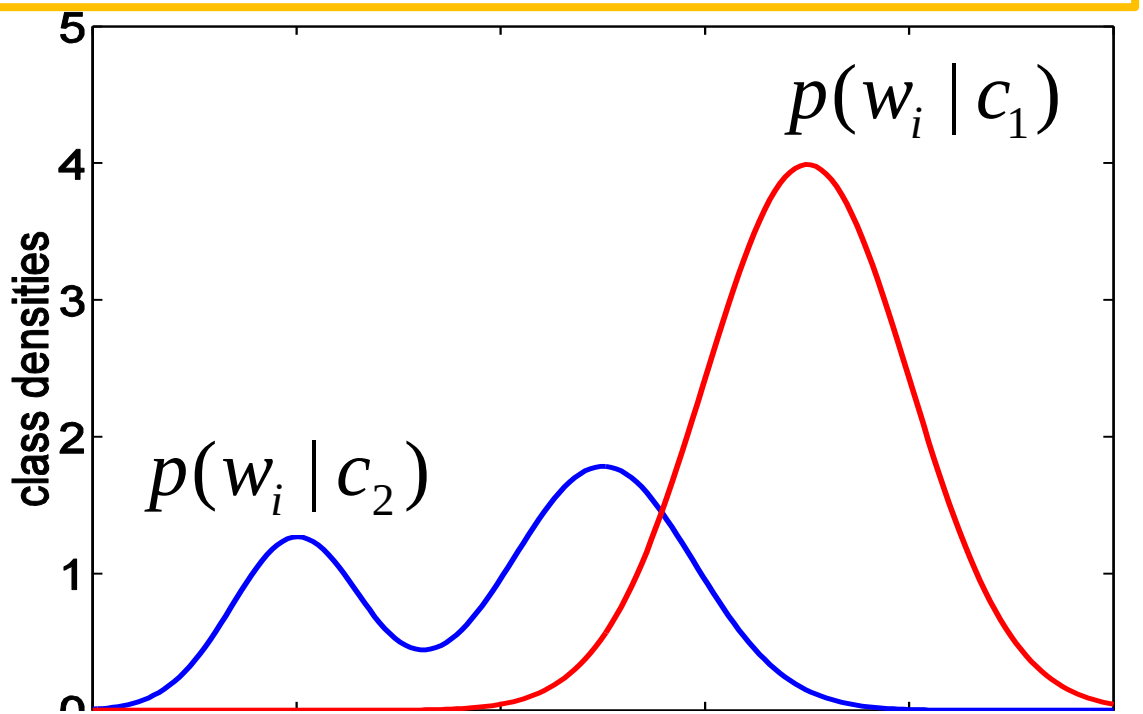
Prior prob. of the object classes

Image likelihood given the class

Likelihood of nth visual word given the class

- How do we learn $P(w_i|c_j)$?
- From empirical frequencies of code words in images from a given class





Classification/Recognition

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

Object class decision

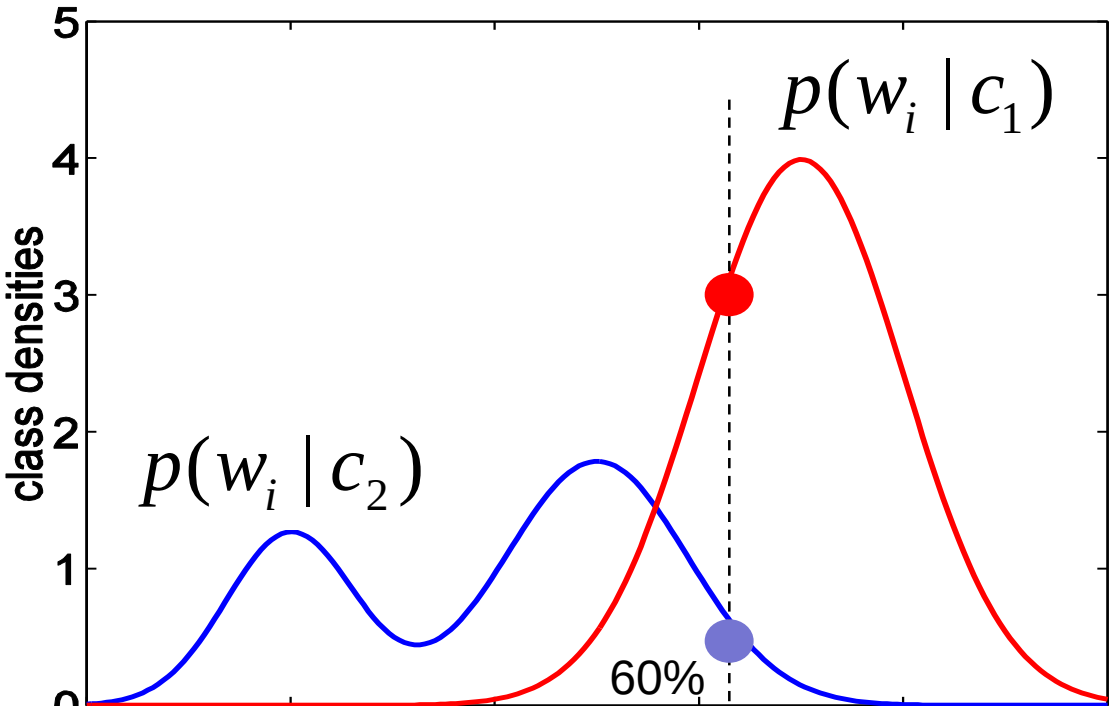
Example:

2 classes:
bananas vs oranges

Query image contains a banana

Look at how many pixels are yellow: say 60%

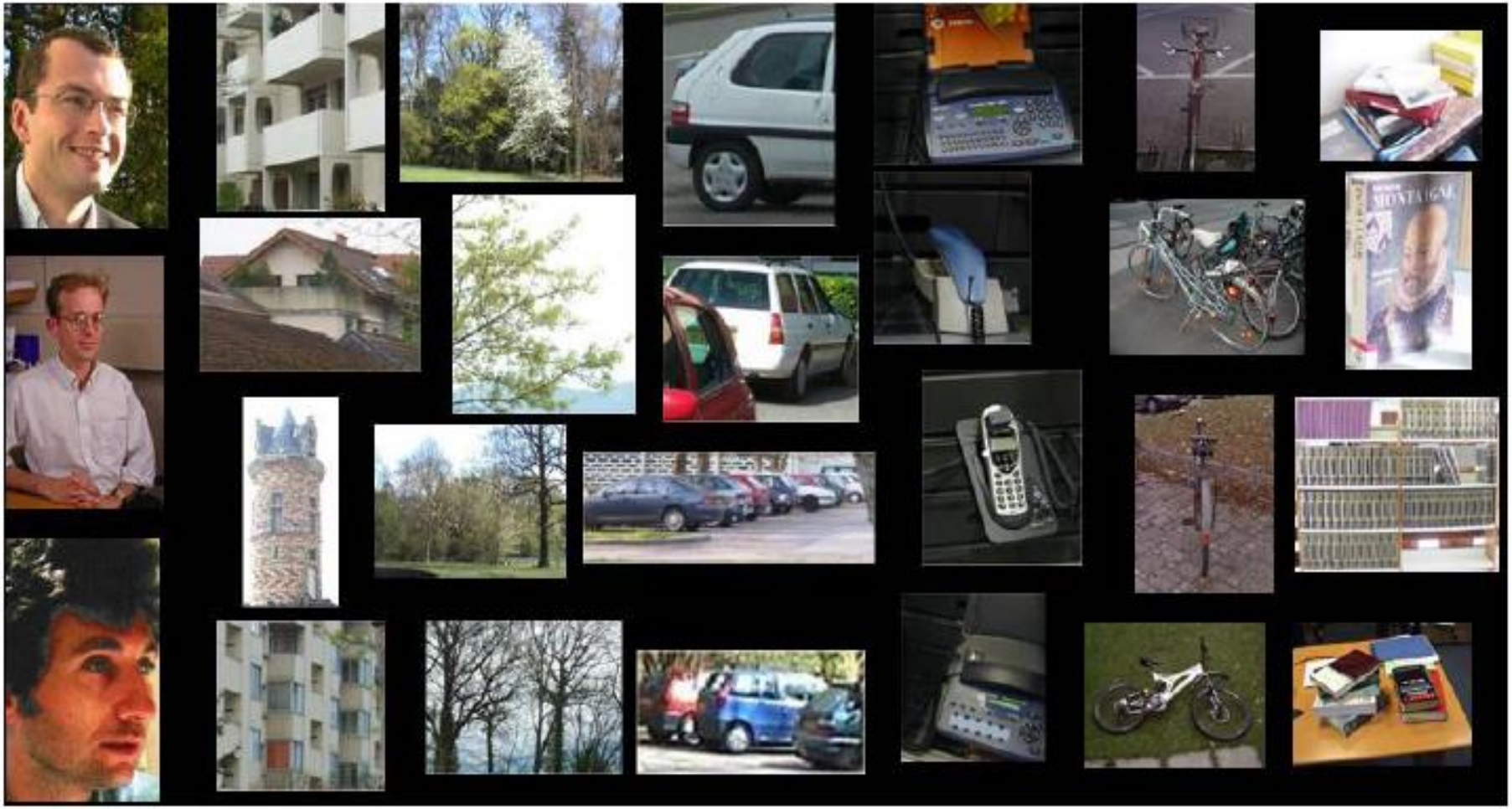
Look at corresponding likelihood values given the two class hypotheses → banana!



Summary: Generative models

- Naïve Bayes
  - *Unigram models* in document analysis
  - Assumes conditional independence of words given class
  - Parameter estimation: frequency counting

Csurka's dataset – 7 classes



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

E = 28%

Table 2. Confusion matrix and mean rank for SVM (*k=1000, linear kernel*).

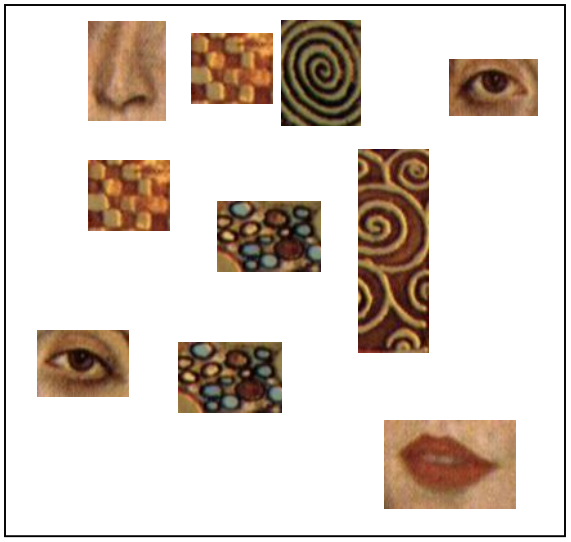
True classes →	*faces*	*buildings*	*trees*	*cars*	*phones*	*bikes*	*books*
faces	**98**	14	10	10	34	0	13
buildings	1	**63**	3	0	3	1	6
trees	1	10	**81**	1	0	6	0
cars	0	1	1	**85**	5	0	5
phones	0	5	4	3	**55**	2	3
bikes	0	4	1	0	1	**91**	0
books	0	3	0	1	2	0	**73**
Mean ranks	1.04	1.77	1.28	1.30	1.83	1.09	1.39

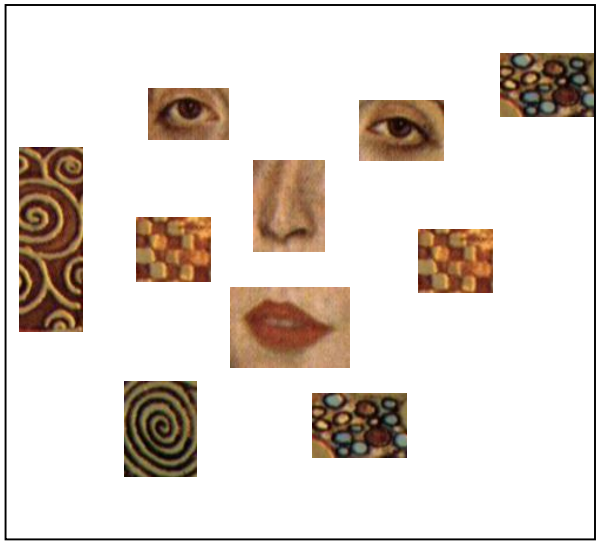
E = 15%

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





- All have equal probability for bag-of-words methods
- Location information is important
- No rigorous geometric information of the object components
- Segmentation and localization unclear

Lecture 13
Visual recognition

- Object classification bag of words models
  - Discriminative methods
  - Generative methods

- Object classification by PCA and FLD

Object classification by…

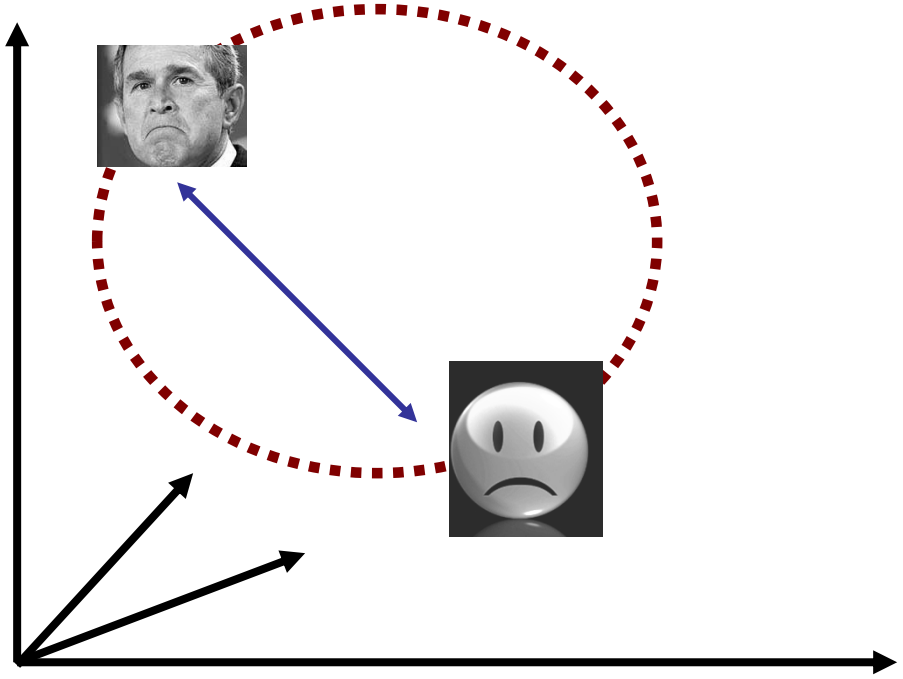
- Principle Component Analysis (PCA)
- Linear Discriminant Analysis (LDA)

Originally introduced for faces:

- Eigenfaces and Fisherfaces

Turk & Penland, 91
Belhumeur et al.,

The Space of images or histograms



- An image (or histogram) H is a point in a high dimensional space
  - An N x M image is a point in R^{NM}

[Thanks to Chuck Dyer, Steve Seitz, Nishino]

Key Idea

- H in the possible set $\chi = \{\hat{x}\}$ are highly correlated.
- So, compress them to a low-dimensional subspace that captures key appearance characteristics of the visual DOFs.

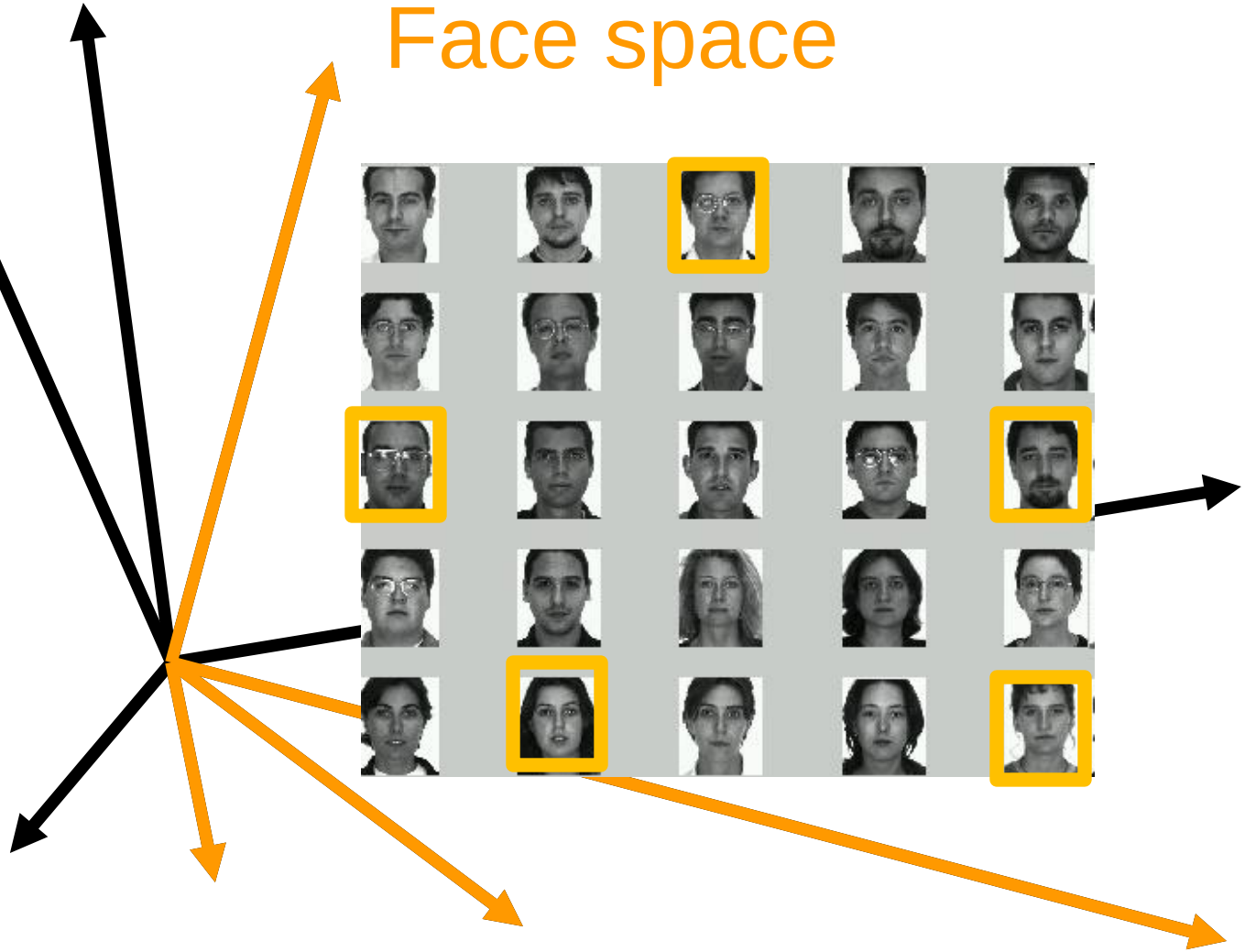
- USE PCA for estimating the sub-space (dimensionality reduction)

- Compare two objects by projecting the images into the subspace and measuring the EUCLIDEAN distance between them.

EIGENFACES: [Turk and Pentland 91]

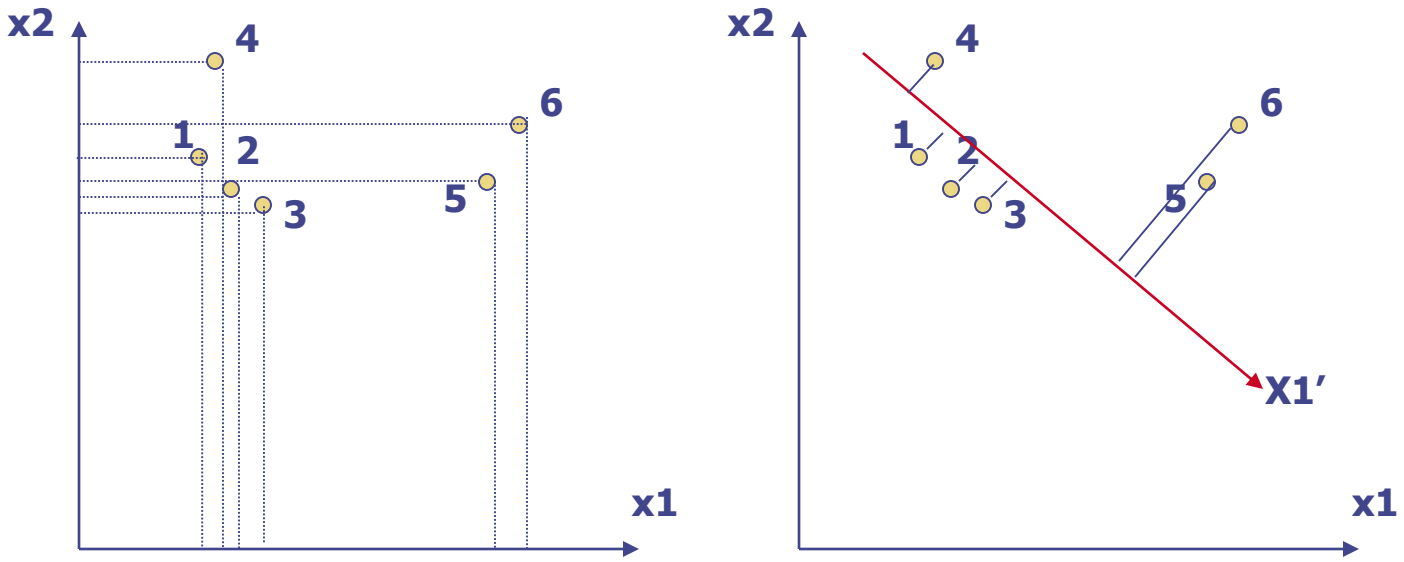
Image space

Face space



- Computes n-dim subspace such that the projection of the data points onto the subspace has the largest variance among all n-dim subspaces.
  - Maximize the scatter of the training images in face space

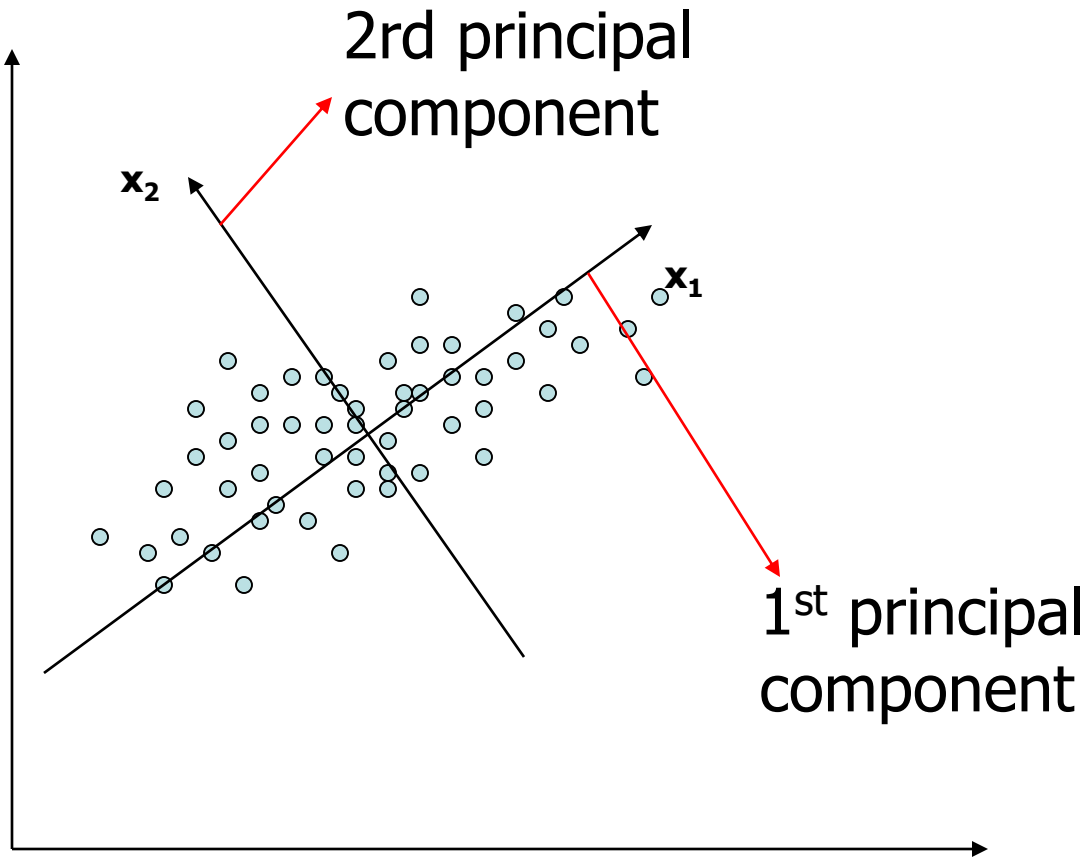
USE PCA for estimating the sub-space



PCA projection

• Computes n-dim subspace such that the projection of the data points onto the subspace has the largest variance among all n-dim subspaces.

USE PCA for estimating the sub-space



PCA Mathematical Formulation

PCA = eigenvalue decomposition of a data covariance matrix

Define a transformation, W,

$$y_j = W^T x_j \qquad j = 1, 2 \ldots N$$

m-dimensional | **Orthonormal** $W \in \mathbb{R}^{n \times m}$ | **n-dimensional**

$$S_T = \sum_{j=1}^{N} (x_j - \bar{x})(x_j - \bar{x})^T$$ = Data Scatter matrix

$$\tilde{S}_T = \sum_{j=1}^{N} (y_j - \bar{y})(y_j - \bar{y})^T = W^T S_T W$$ = Transf. data scatter matrix

Measure data scatter

Eigenvectors of S_T

$$W_{opt} = \arg\max_W |W^T S_T W| = [\mathbf{v}_1 \quad \mathbf{v}_2 \quad \ldots \quad \mathbf{v}_m]$$

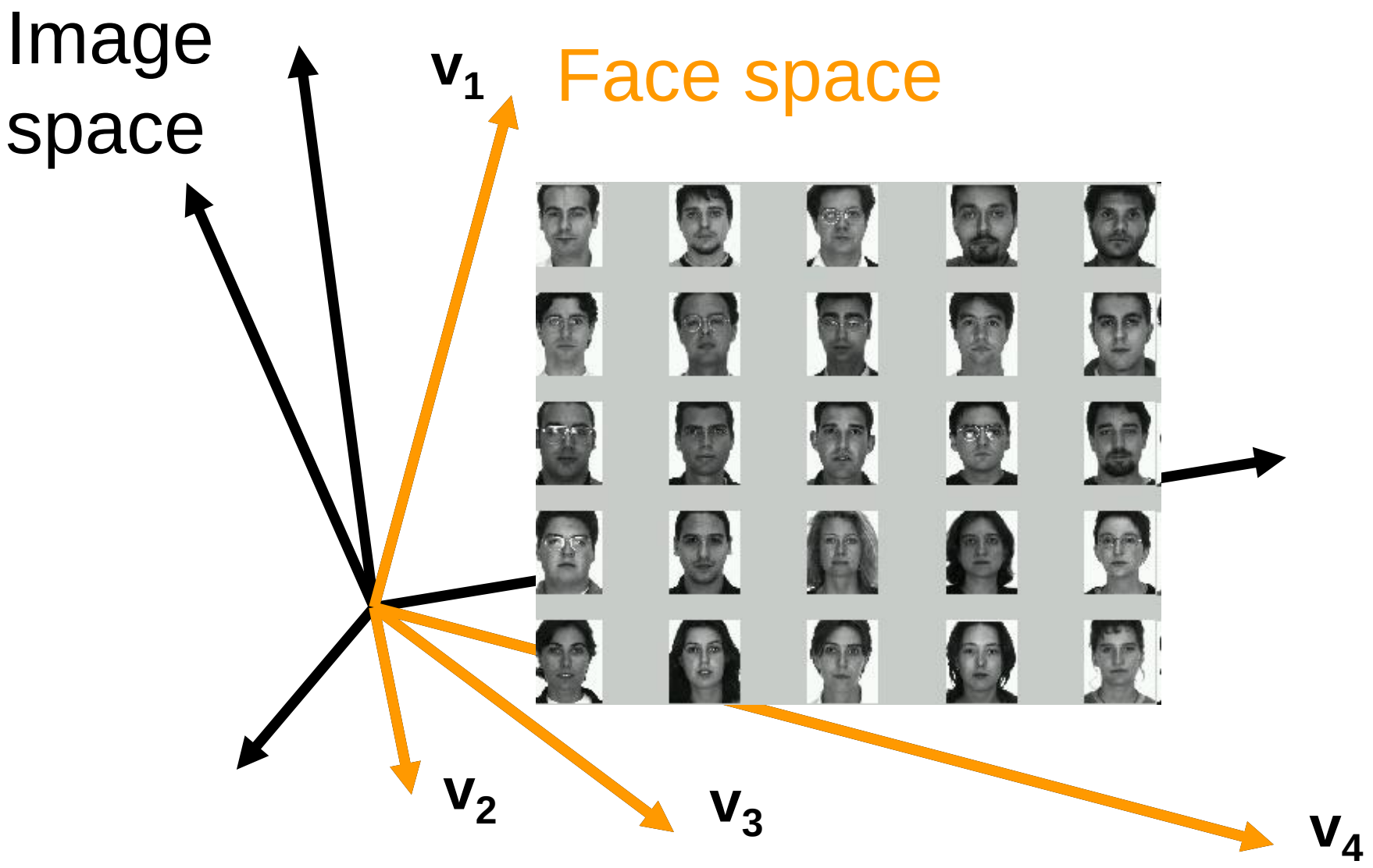


Image space
v_1
Face space
v_2
v_3
v_4

Projecting onto the Eigenfaces

- The eigenfaces $\mathbf{v_1}$, ..., $\mathbf{v_K}$ span the space of faces
  - A face is converted to eigenface coordinates by

$$\mathbf{x} \rightarrow (\underbrace{(\mathbf{x}-\overline{\mathbf{x}})\cdot\mathbf{v_1}}_{a_1}, \underbrace{(\mathbf{x}-\overline{\mathbf{x}})\cdot\mathbf{v_2}}_{a_2}, \ldots, \underbrace{(\mathbf{x}-\overline{\mathbf{x}})\cdot\mathbf{v_K}}_{a_K})$$

$$\mathbf{x} \approx \overline{\mathbf{x}} + a_1\mathbf{v_1} + a_2\mathbf{v_2} + \ldots + a_K\mathbf{v_K}$$

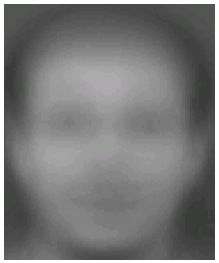
$\mathbf{x}$ $a_1\mathbf{v_1}$ $a_2\mathbf{v_2}$ $a_3\mathbf{v_3}$ $a_4\mathbf{v_4}$ $a_5\mathbf{v_5}$ $a_6\mathbf{v_6}$ $a_7\mathbf{v_7}$ $a_8\mathbf{v_8}$

Algorithm

Training

1. Align training images $x_1, x_2, \ldots, x_N$

Note that each image is formulated into a long vector!



2. Compute average face $\bar{x} = 1/N \Sigma x_i$

3. Compute the difference image $x_i - \bar{x}$

Algorithm

4. Compute the covariance matrix (total scatter matrix)

$$S_T = \sum_{j=1}^{N} (x_j - \bar{x})(x_j - \bar{x})^T$$

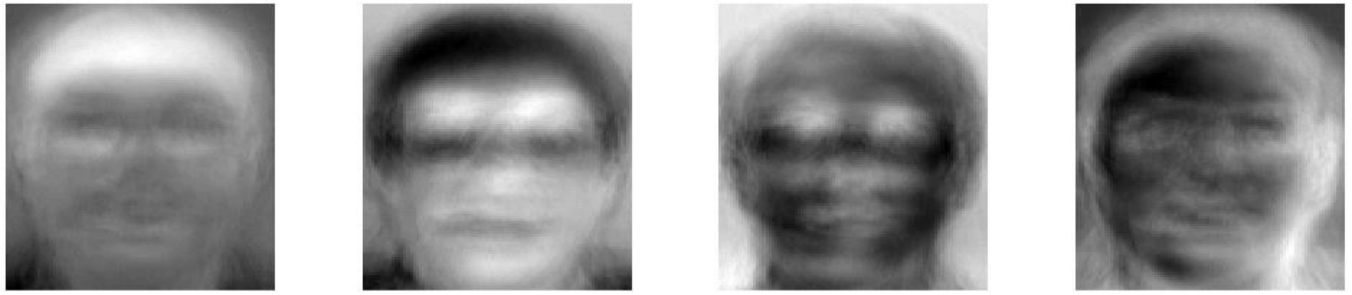
5. Compute the eigenvectors of the covariance matrix S_T
6. Compute training projections a_1, a_2... a_N

Testing

1. Take query image X
2. Project X into Eigenface space (W = {eigenfaces}) and compute projection ω_i
3. Compare projection ω_i with all training N projections a_i

Illustration of Eigenfaces

- The visualization of eigenvectors:



These are the first 4 eigenvectors from a training set of 400 images (ORL Face Database).

Eigenfaces look somewhat like generic faces.

Reconstruction and Errors

P = 4

P = 200

P = 400

- Only selecting the top P eigenfaces → reduces the dimensionality.
- Fewer eigenfaces result in more information loss, and hence less discrimination between faces.

Summary for Eigenface

Pros

- Non-iterative, globally optimal solution

Limitations

•PCA projection is **optimal for reconstruction** from a low dimensional basis, but **may NOT be optimal for discrimination…**

Extensions

- Generalized PCA:

R. Vidal, Y. Ma, and S. Sastry. Generalized Principal Component Analysis (GPCA). IEEE Transactions on Pattern Analysis and Machine Intelligence, volume 27, number 12, pages 1 - 15,

2005.

- Tensor Faces:

"Multilinear Analysis of Image Ensembles: TensorFaces," M.A.O. Vasilescu, D. Terzopoulos, *Proc. 7th European Conference on Computer Vision (ECCV'02)*, Copenhagen, Denmark, May, 2002

- PCA-SIFT

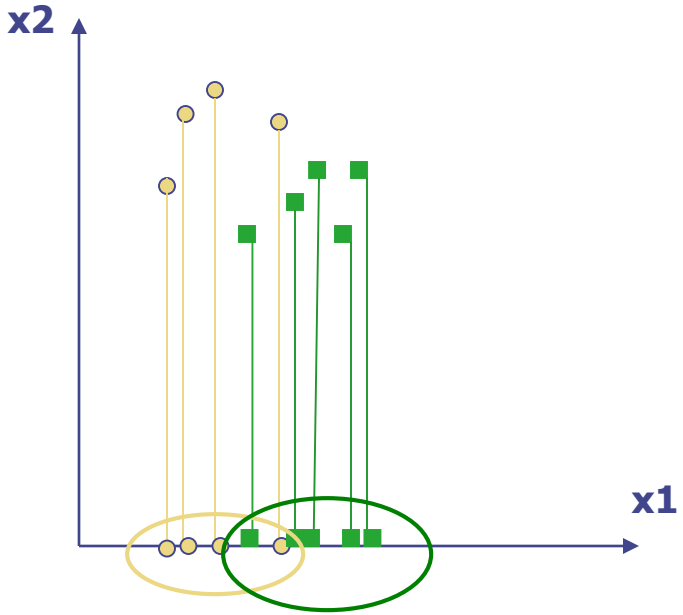
PCA-**SIFT**: A More Distinctive Representation for Local Image Descriptors -
Y Ke, R Sukthankar - IEEE CVPR 04

Linear Discriminant Analysis (LDA)
Fisher's Linear Discriminant (FLD)

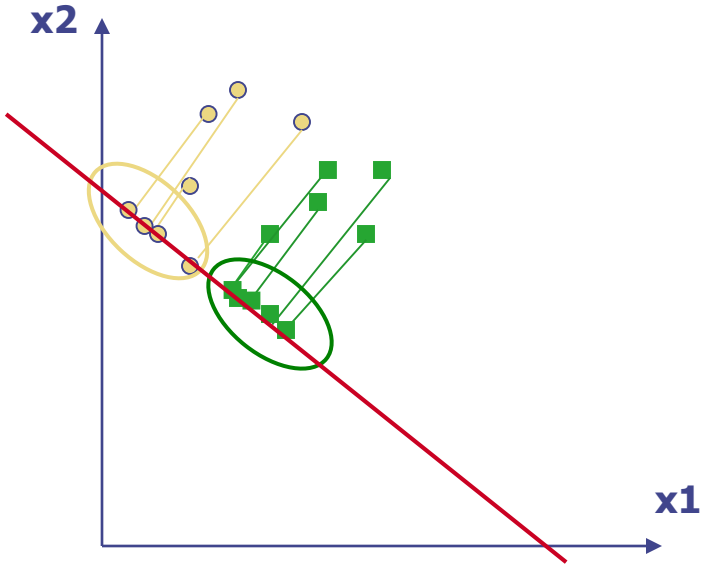
- Eigenfaces exploit the max scatter of the training images in face space
- Fisherfaces attempt to maximise the **between class scatter**, while minimising the **within class scatter**.

Illustration of the Projection

- Using two classes as example:



Poor Projection



Good

Variables

- N Sample images: $\{x_1, \cdots, x_N\}$
- c classes: $\{\chi_1, \cdots, \chi_c\}$
- Average of each class: $\mu_i = \frac{1}{N_i} \sum_{x_k \in \chi_i} x_k$
- Total average: $\mu = \frac{1}{N} \sum_{k=1}^{N} x_k$

Scatters

- Scatter of class i:

$$S_i = \sum_{x_k \in \chi_i} (x_k - \mu_i)(x_k - \mu_i)^T$$

- Within class scatter:

$$S_W = \sum_{i=1}^{c} S_i$$

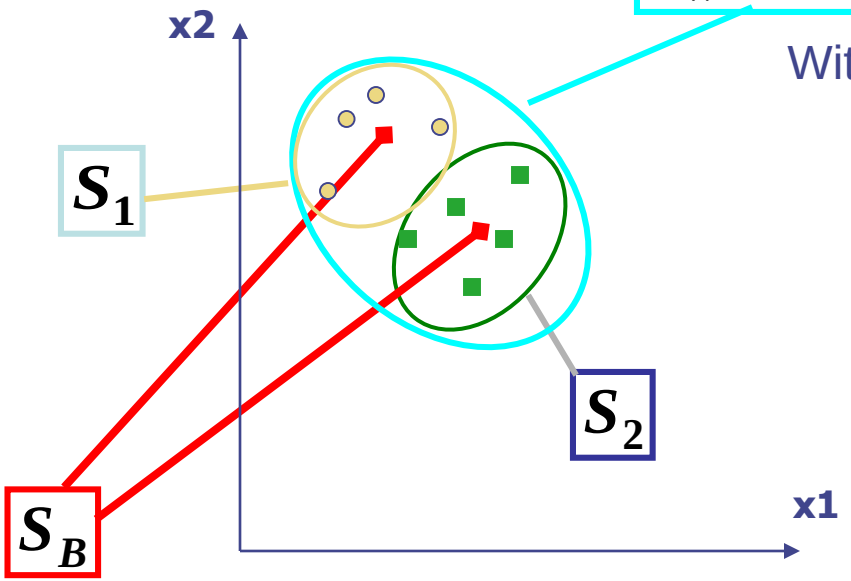
- Between class scatter:

$$S_B = \sum_{i=1}^{c} |\chi_i| (\mu_i - \mu)(\mu_i - \mu)^T$$

- Total scatter:

$$S_T = S_W + S_B$$

Illustration



$$S_i = \sum_{x_k \in \chi_i} (x_k - \mu_i)(x_k - \mu_i)^T$$

$$S_W = \sum_{i=1}^{c} S_i$$

$$S_B = \sum_{i=1}^{c} |\chi_i| (\mu_i - \mu)(\mu_i - \mu)^T$$

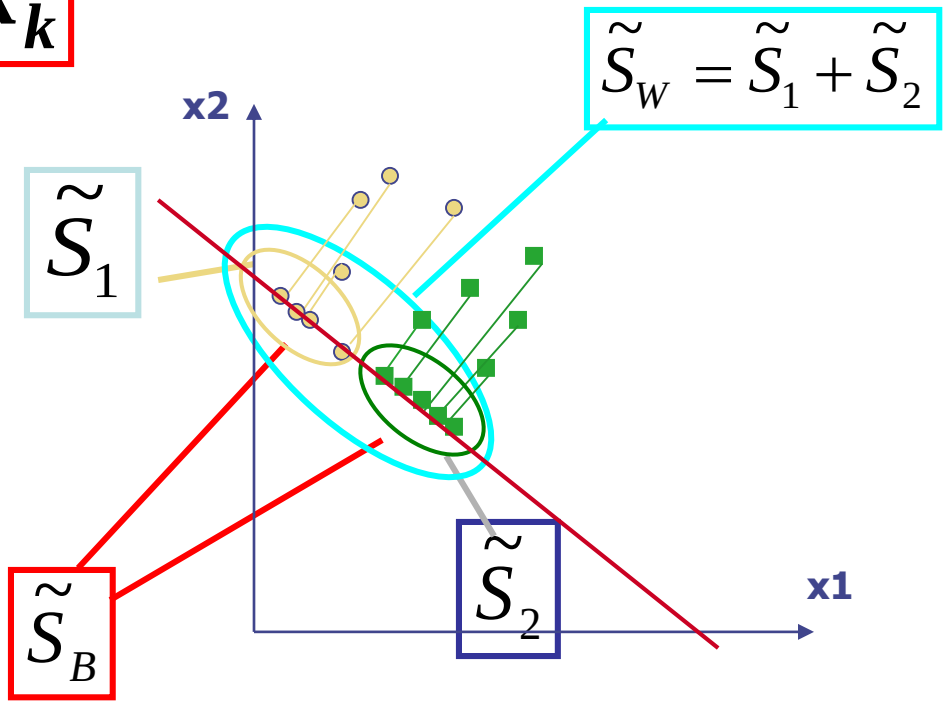
Mathematical Formulation (1)

- After projection: $y_k = W^T x_k$

- Between class scatter (of y's): $\tilde{S}_B = W^T S_B W$
- Within class scatter (of y's): $\tilde{S}_W = W^T S_W W$

Illustration

$$y_k = W^T x_k$$



$$S_W = \sum_{i=1}^{c} S_i$$

$$S_B = \sum_{i=1}^{c} |\chi_i| (\mu_i - \mu)(\mu_i - \mu)^T$$

$$\tilde{S}_W = W^T S_W W$$

$$\tilde{S}_B = W^T S_B W$$

Mathematical Formulation

- The desired projection:

$$W_{opt} = \arg\max_{\mathbf{W}} \frac{\left|\tilde{S}_B\right|}{\left|\tilde{S}_W\right|} = \arg\max_{\mathbf{W}} \frac{\left|W^T S_B W\right|}{\left|W^T S_W W\right|}$$

- How is it found ? → Generalized Eigenvectors

- If S_w has full rank, the generalized eigenvectors are eigenvectors of $S_W^{-1} S_B$ with largest eigen-values

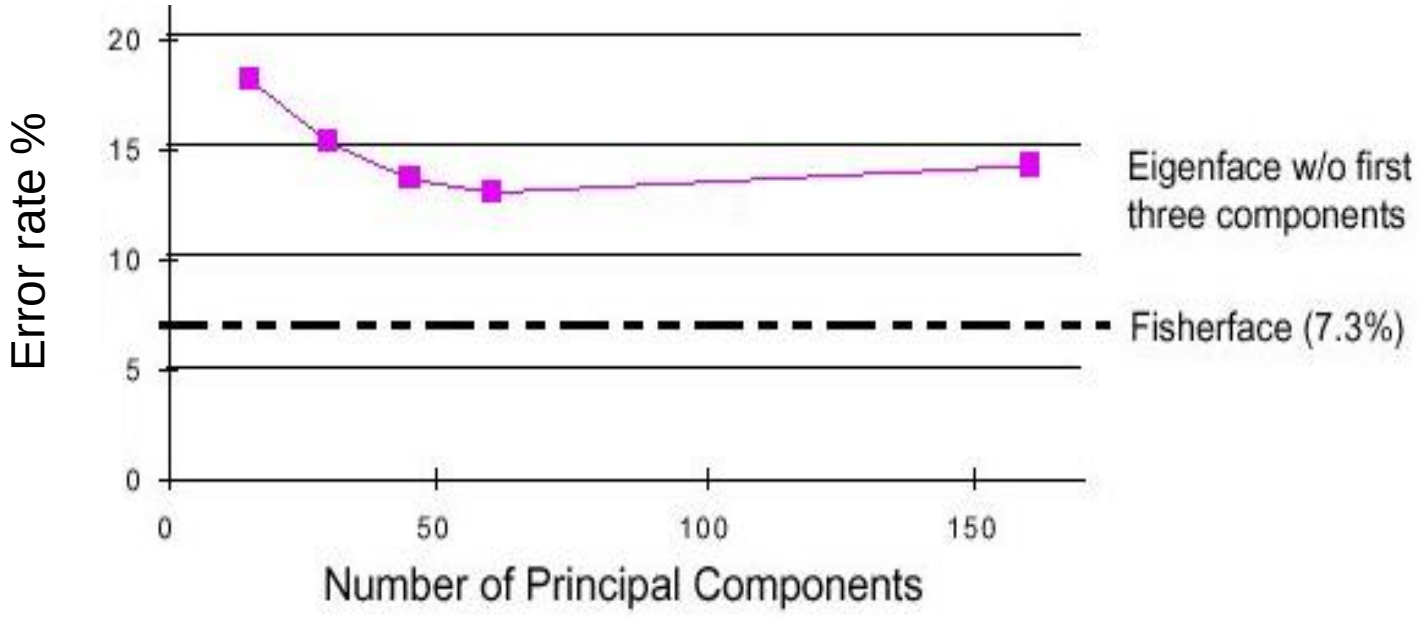
$$S_B w_i = \lambda_i S_W w_i \qquad i = 1, \ldots, m$$

Results: Eigenface vs. Fisherface

- Input: 160 images of 16 people
- Train: 159 images
- Test: 1 image

- Variation in Facial Expression, Eyewear, and Lighting

Results: Eigenface vs. Fisherface



Next lecture

- Object detection