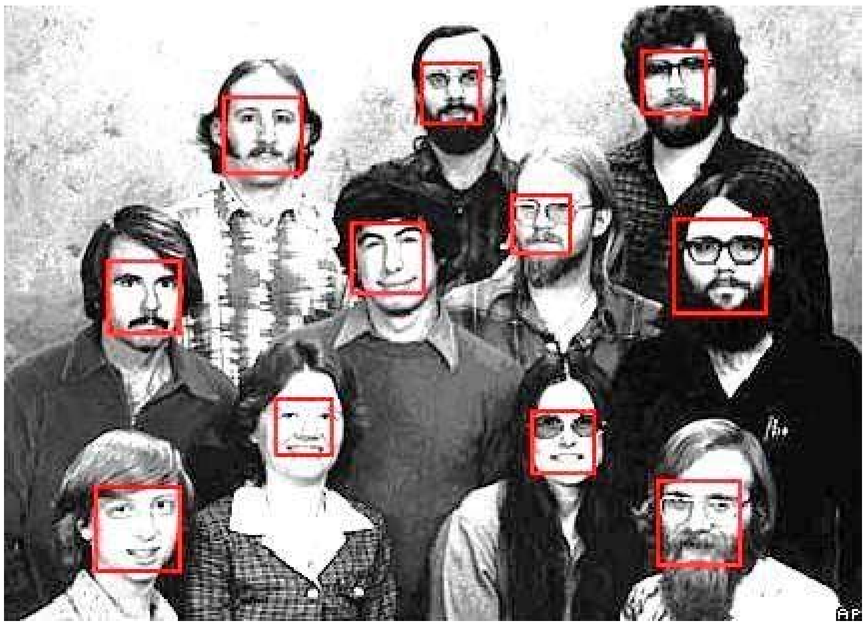
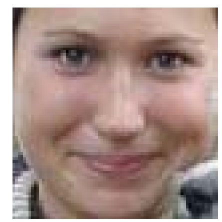
Face detection and recognition



Many slides adapted from K. Grauman and D. Lowe

Face detection and recognition

Detection



Recognition

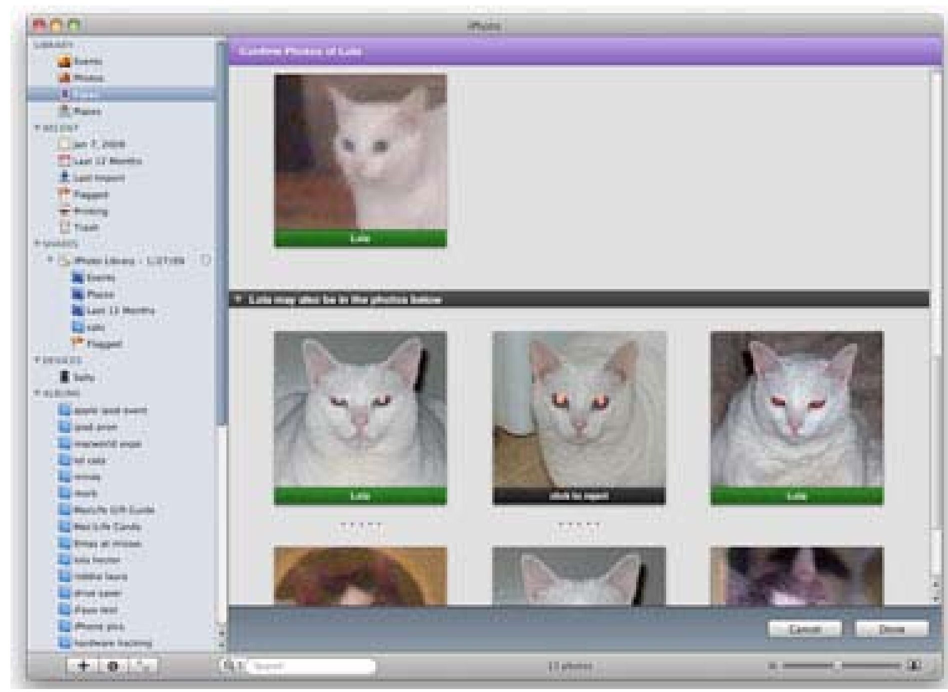
“Sally”

Consumer application: iPhoto 2009

http://www.apple.com/ilife/iphoto/

Consumer application: iPhoto 2009

Can be trained to recognize pets!



http://www.maclife.com/article/news/iphotos_faces_recognizes_cats

Consumer application: iPhoto 2009

Things iPhoto thinks are faces

Outline

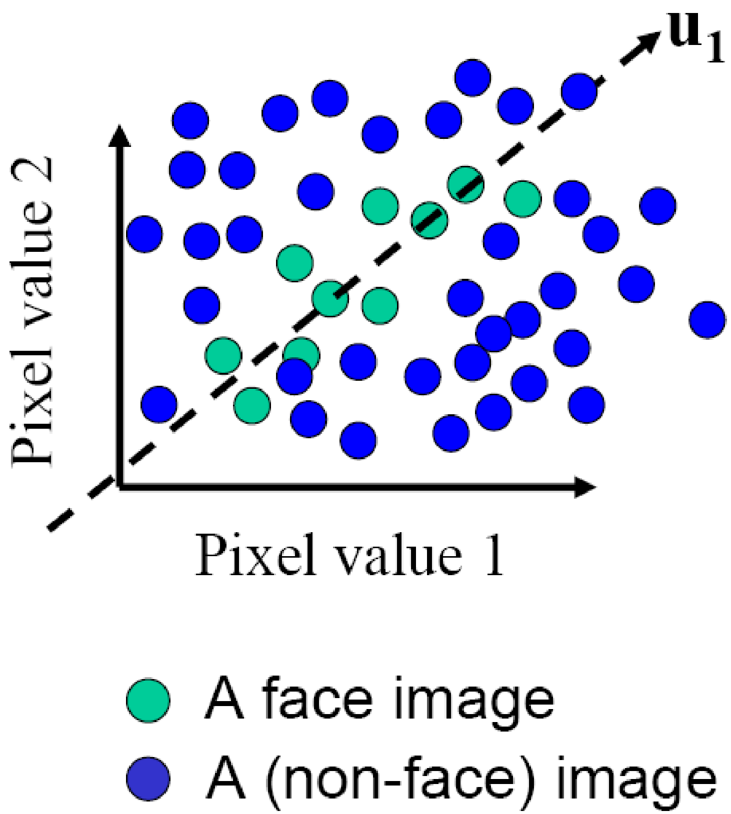
- Face recognition
  - Eigenfaces
- Face detection
  - The Viola & Jones system

The space of all face images

- When viewed as vectors of pixel values, face images are extremely high-dimensional
  - 100x100 image = 10,000 dimensions
- However, relatively few 10,000-dimensional vectors correspond to valid face images
- We want to effectively model the subspace of face images

The space of all face images

- We want to construct a low-dimensional linear subspace that best explains the variation in the set of face images



Principal Component Analysis

- Given: N data points $\mathbf{x}_1, \ldots, \mathbf{x}_N$ in R^d
- We want to find a new set of features that are linear combinations of original ones:

$$u(\mathbf{x}_i) = \mathbf{u}^T(\mathbf{x}_i - \boldsymbol{\mu})$$

($\boldsymbol{\mu}$: mean of data points)

- What unit vector $\mathbf{u}$ in R^d captures the most variance of the data?

Forsyth & Ponce, Sec. 22.3.1, 22.3.2

Principal Component Analysis

- Direction that maximizes the variance of the projected data:

$$var(u) = \frac{1}{N}\sum_{i=1}^{N} \underbrace{\mathbf{u}^{\mathrm{T}}(\mathbf{x}_i - \mu)}_{\text{Projection of data point}}(\mathbf{u}^{\mathrm{T}}(\mathbf{x}_i - \mu))^{\mathrm{T}}$$

$$= \mathbf{u}^{\mathrm{T}}\underbrace{\left[\sum_{i=1}^{N}(\mathbf{x}_i - \mu)(\mathbf{x}_i - \mu)^{\mathrm{T}}\right]}_{\text{Covariance matrix of data}}\mathbf{u}$$

$$= \mathbf{u}^{\mathrm{T}}\Sigma\mathbf{u}$$

The direction that maximizes the variance is the eigenvector associated with the largest eigenvalue of Σ

Principal component analysis

- The direction that captures the maximum covariance of the data is the eigenvector corresponding to the largest eigenvalue of the data covariance matrix

- Furthermore, the top *k* orthogonal directions that capture the most variance of the data are the *k* eigenvectors corresponding to the *k* largest eigenvalues

Eigenfaces: Key idea

- Assume that most face images lie on a low-dimensional subspace determined by the first k ($k<d$) directions of maximum variance

- Use PCA to determine the vectors or "eigenfaces" $\mathbf{u}_1, \ldots \mathbf{u}_k$ that span that subspace

- Represent all face images in the dataset as linear combinations of eigenfaces

M. Turk and A. Pentland, Face Recognition using Eigenfaces, CVPR 1991

Eigenfaces example

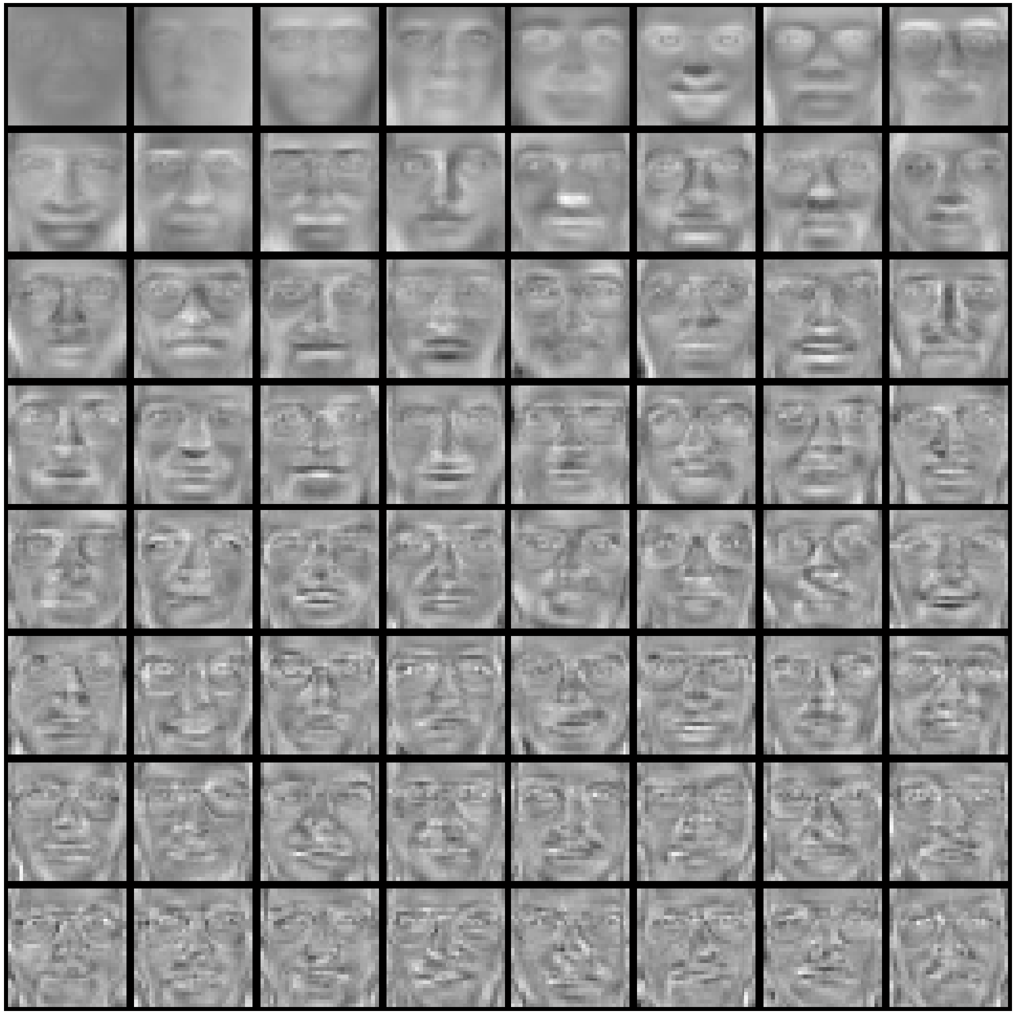
Training images

$\mathbf{x}_1, \ldots, \mathbf{x}_N$

Eigenfaces example

Mean: $\mathbf{\mu}$

Top eigenvectors: $\mathbf{u}_1, \ldots \mathbf{u}_k$



Eigenfaces example

Principal component (eigenvector) u_k

$\mu + 3\sigma_k u_k$

$\mu - 3\sigma_k u_k$

Eigenfaces example

- Face **x** in "face space" coordinates:

$$\mathbf{x} \rightarrow [\mathbf{u}_1^{\mathrm{T}}(\mathbf{x} - \mu), \ldots, \mathbf{u}_k^{\mathrm{T}}(\mathbf{x} - \mu)]$$

$$= \quad w_1, \ldots, w_k$$

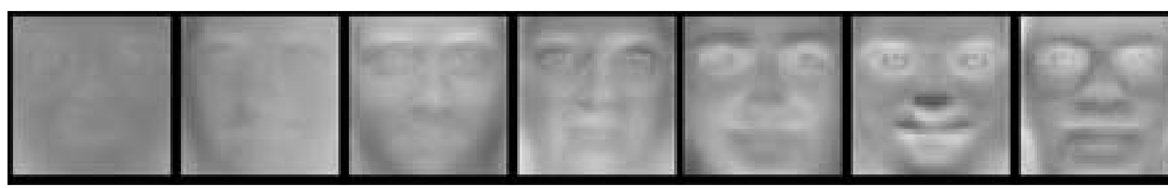
Eigenfaces example

- Face **x** in "face space" coordinates:

$$\mathbf{x} \rightarrow [\mathbf{u}_1^{\mathrm{T}}(\mathbf{x} - \mu), \ldots, \mathbf{u}_k^{\mathrm{T}}(\mathbf{x} - \mu)]$$

$$= \quad w_1, \ldots, w_k$$

- Reconstruction:

 =  + 

$$\hat{\mathbf{x}} = \boldsymbol{\mu} + w_1\mathbf{u}_1 + w_2\mathbf{u}_2 + w_3\mathbf{u}_3 + w_4\mathbf{u}_4 + \ldots$$

Reconstruction demo

Recognition with eigenfaces

Process labeled training images:

- Find mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$
- Find k principal components (eigenvectors of $\boldsymbol{\Sigma}$) $\mathbf{u}_1,\ldots\mathbf{u}_k$
- Project each training image $\mathbf{x}_i$ onto subspace spanned by principal components:
  $(w_{i1},\ldots,w_{ik}) = (\mathbf{u}_1^T(\mathbf{x}_i - \boldsymbol{\mu}), \ldots , \mathbf{u}_k^T(\mathbf{x}_i - \boldsymbol{\mu}))$

Given novel image $\mathbf{x}$:

- Project onto subspace:
  $(w_1,\ldots,w_k) = (\mathbf{u}_1^T(\mathbf{x} - \boldsymbol{\mu}), \ldots , \mathbf{u}_k^T(\mathbf{x} - \boldsymbol{\mu}))$
- Optional: check reconstruction error $\mathbf{x} - \hat{\mathbf{x}}$ to determine whether image is really a face
- Classify as closest training face in k-dimensional subspace

M. Turk and A. Pentland, Face Recognition using Eigenfaces, CVPR 1991

Recognition demo

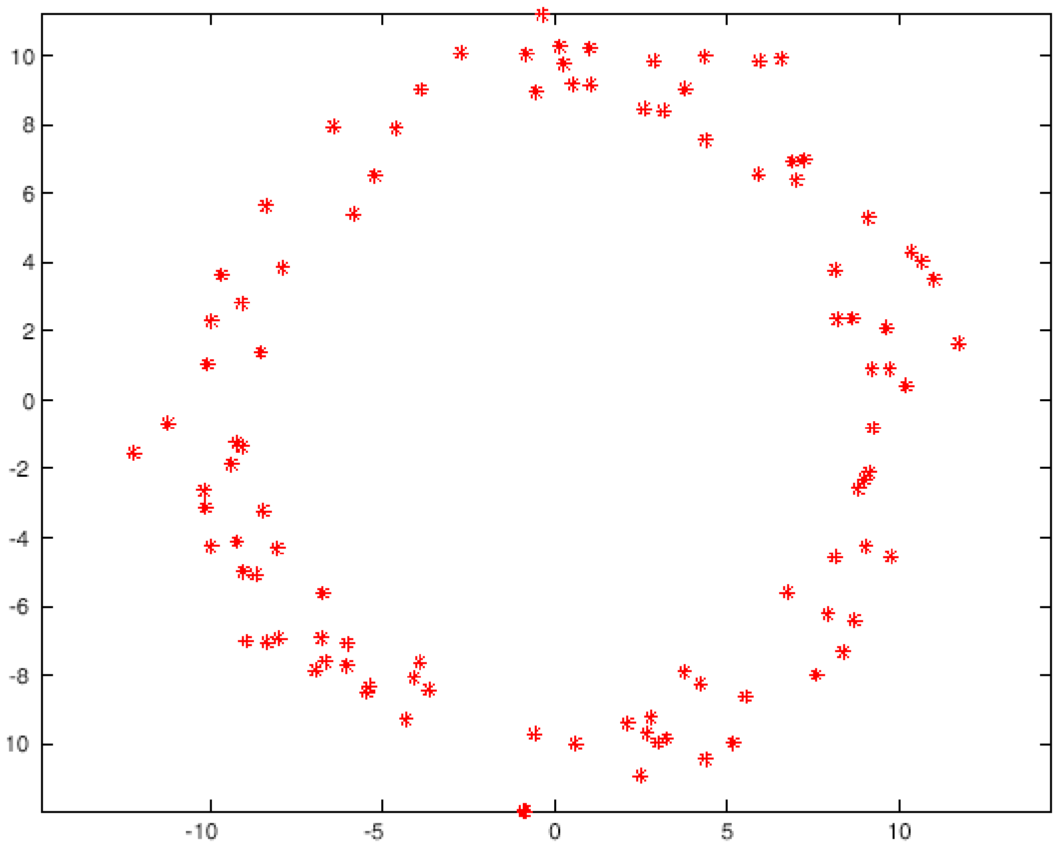
Limitations

- Global appearance method: not robust to misalignment, background variation

Limitations

- PCA assumes that the data has a Gaussian distribution (mean μ, covariance matrix Σ)



The shape of this dataset is not well described by its principal components

Limitations

- The direction of maximum variance is not always good for classification