Eigenfaces and Fisherfaces

Presenter: Harry Chao
aMMAI 2010

Outline

1. Introduction:
   Face recognition & Dimensionality reduction
2. Principal Component Analysis
3. Linear Discriminant Analysis
4. Other methods
5. Conclusion
6. Reference
I. Introduction

What & Why is face recognition?

The definition:
Identify or verify a person based on face

The motivation:
- Remarkable face recognition capability of human visual system
- Numerous important application:
  ex: Surveillance & face ID

Community involved:
Neuroscience, Psychology, pattern recognition, computer vision, machine learning……
General techniques

Observation:
Images, videos, and 3D images

Based on images:
- Holistic-based methods (appearance)
- Feature-based methods (landmark)

Funny idea form psychology

- Thatcher Illusion
Challenge of face recognition

Challenge:
- Feature + Classifier
- The distance between different faces is not obvious!!

Distortion:
- illumination, pose, affine transform, expression, occlusion, noise

Dimensionality reduction

Why?
- The curse of dimensionality
- Intrinsic dimensionality may be smaller
- Some feature are not relevant

Idea:
- Reduce the feature dimension while preserving as much information as possible
- Decorrelation
- Extract the real distribution of the population

Methods:
- Feature selection & Feature reduction
- Supervised (LDA) & Unsupervised (PCA)
Meet with face recognition

Face recognition is a special case:
- Many classes but a few samples of each class
- KNN or other distance-based measurements may perform better than classifiers

For holistic-based:
Dimension reduction performs like data-driven features

For feature-based:
Dimension reduction is based on domain knowledge

2. PCA: Eigenfaces
General idea

**Objective:**
- Look for a few linear combinations, which can be used to summarize the data and loses in data as little as possible (want to preserve the variance)

**For face recognition:**
- A 256x256 face image is equivalent to a 665536-dim vector
- We want to reduce the dimension based on the database
- The new dimensionality depends on the number of images in the database

**PCA is also known as:**
- Karhunen-Loeve methods

**Covariance matrix**

- The covariance matrix is symmetric with variances on the diagonal; assuming D dimensions (or variables)

\[
\Sigma = [\sigma_{ij}]_{D \times D} = \begin{bmatrix}
\sigma_{0,0} & \sigma_{0,1} & \sigma_{0,D-1} \\
\sigma_{1,0} & \sigma_{1,1} & \sigma_{1,D-1} \\
\vdots & \vdots & \vdots \\
\sigma_{D-1,0} & \sigma_{D-1,1} & \sigma_{D-1,D-1}
\end{bmatrix}
\]

- and covariance of two random variables (dimensions)

\[
\sigma_{x,y}^2 = E((x-\mu_x)(y-\mu_y)) = \frac{1}{N} \sum_{i=0}^{N-1} (x_i-\mu_x)(y_i-\mu_y)
\]

- Diagonal elements are individual variances in each dimension
- Off-diagonal elements are covariance indicating data dependency between variables (dimensions in histogram)
Procedure for PCA

Linear projection:
- Originally N points in D-dim: \( \{x_i\}_{i=1}^{N} \in \mathbb{R}^D \)
- A set of basis for projection: \( \{u_i\}_{i=1}^{M} \in \mathbb{R}^D \)
- These basis are orthonormal, and generally we have \( M<<D \)
- Preserve the reconstruction error as well as variance

Procedure:
- Find the mean vector \( \Psi \) (D-by-1)
- Subtract each vector by \( \Psi \) and get \( \Phi_i \)
- Calculate the covariance matrix \( \Sigma \) of \( \Phi_i \) (D-by-D)
- Calculate the set of eigenvectors of \( \Sigma \) (D-by-N matrix)
- Preserve the \( M \) largest eigenvalues (D-by-M matrix \( U \))
- \( U' \Phi_i \) is the eigenfaces of the \( i \)th face (M-by-1)

Let’s see an example

Face database


Mean face

Eigenvectors
**Formula of PCA**

- Assume we have Subtract each vector by $\Psi$ and get $\Phi_i$, we want to find a projection vector $b$ to minimize:

  \[ E[||bb^T\Phi_i - \Phi_i||^2] = E[||bb^T - I||^2] = E[((bb^T - I)\Phi_i)^T(bb^T - I)\Phi_i] \]

- **Important tools**
  - $\text{trace}(scale) = \text{scale}$
  - $\text{trace}(ABC) = \text{trace}(CAB) = \text{trace}(BCA)$
  - $\text{trace}(E) = E(\text{trace})$

- Then we can rewrite the formula

  \[
  E[(bb^T - I)\Phi_i^T(bb^T - I)\Phi_i] = E[(bb^T - I)^T(bb^T - I)\Phi_i \Phi_i^T]
  \]

- **Now we want to maximize:** (Using **Language multiplier**)

  \[ tr(b^T\Sigma b) = b^T\Sigma b \text{ with } b^Tb = 1 \]

**Formula for eigenvectors**

No we want to get the eigenvectors of $\Sigma$:

- Problem: $\Sigma$ is of size 65536-by-65536 for 256-by-256 images

Solution:

\[ \Sigma = E[\phi_i \phi_i^T] = \text{constant} * \Phi \Phi^T \quad (\Phi \text{ is of } D\text{-by-}N) \]

We can first solve $\Phi^T\Phi x = \lambda x$

then do $\Phi \Phi^T (\Phi x) = \lambda (\Phi x)$

where $\Phi \Phi^T$ is of $D\text{-by-}D$ and $\Phi^T\Phi$ of $N\text{-by-}N$

\[
\Phi : \begin{bmatrix} \Phi_1 \\ \Phi_2 \\ \vdots \\ \Phi_D \end{bmatrix}_D \quad \Phi^T : \begin{bmatrix} \Phi_1^T \\ \Phi_2^T \\ \vdots \\ \Phi_D^T \end{bmatrix}_D \quad \Phi \Phi^T : \begin{bmatrix} \Phi_1 \Phi_1^T & \ldots & \Phi_1 \Phi_D^T \\ \vdots & \ddots & \vdots \\ \Phi_D \Phi_1^T & \ldots & \Phi_D \Phi_D^T \end{bmatrix}_D \quad \Sigma = E[\phi_i \phi_i^T] : \begin{bmatrix} \Sigma_{11} & \ldots & \Sigma_{1D} \\ \vdots & \ddots & \vdots \\ \Sigma_{D1} & \ldots & \Sigma_{DD} \end{bmatrix}_D
\]
Covariance matrix

- The covariance matrix is symmetric with variances on the diagonal; assuming D dimensions (or variables)
\[
\Sigma = \begin{bmatrix}
\sigma^2_{0,0} & \sigma^2_{0,1} & \cdots & \sigma^2_{0,D-1} \\
\sigma^2_{1,0} & \sigma^2_{1,1} & \cdots & \sigma^2_{1,D-1} \\
\vdots & \vdots & \ddots & \vdots \\
\sigma^2_{D-1,0} & \sigma^2_{D-1,1} & \cdots & \sigma^2_{D-1,D-1}
\end{bmatrix}
\]

- and covariance of two random variables (dimensions) \(x, y\):
\[
\sigma^2_{x,y} = \text{E}[(x-\mu_x)(y-\mu_y)] = \frac{1}{N}\sum_{i=0}^{N-1} (x_i-\mu_x)(y_i-\mu_y)
\]

- Diagonal elements are individual variances in each dimension
- Off-diagonal elements are covariance indicating data dependency between variables (dimensions in histogram)

Example of face reconstruction

Reconstruction procedure
**Eigenfaces for face recognition**

- Input image $I$
- Mean image $\Psi$
- Identity $\Phi$
- Projected image $\Phi_t$
- Distance measure $\Omega$

**Example of character recognition**

- Original database
  - a e i o u
  - d g h l r
  - v w x y z
  - ! @ # $ %

- Eigenvectors

- Result 1
  - WORLD
  - WORLD
  - WORLD

- Result 2
  - good!
  - good!
  - good#i
Good properties of PCA

- Good for dealing with random noise, but not good for rotation-scaling-translation (RST) distortion. It could minimize the distance between projection space and data space, and really reduce the redundancy!


3. LDA: Fisherfaces
General idea (I)

Objective:
- Look for dimension reduction based on discrimination purpose

For face recognition:
- The variance among faces in the database may come from distortions such as illumination, facial expression, and pose variation. And sometimes, these variations are larger than variations among standard faces!!
- The images of a particular face, under varying illumination but fixed pose, lie in a 3D linear subspace of the high dimensional image space. (without shadowing)

General idea (II)

Idea:
- Try to find a basis for projection that minimize the intra-class variation but preserve the inter-class variation.
- Rather than explicitly modeling this deviation, we linearly project the image into a subspace in a manner which discount those regions of the face with large deviation

Fisher linear discriminant

inter-class: \[ \left| \bar{m}_i - \bar{m}_j \right| = \left| w^T (m_i - m_j) \right| \]

intra-class: \[ \hat{s}^2 = \sum_{y \in D_i} (y - \bar{m}_i)^2 \]

want to maximize: \[ J(w) = \frac{\left| \bar{m}_1 - \bar{m}_2 \right|^2}{\hat{s}^2 + \hat{s}^2_2} \]

\[ x, w, m_1, m_2 \]

\[ X \in \mathbb{R}^{D} \]

\[ S_{w}, S_{b} \]

**Multiple discriminant analysis**

\[ S_{b} = \sum_{i=1}^{c} N_i (m_i - m)(m_i - m)^T \]

\[ S_{w} = \sum_{i=1}^{c} \sum_{x \in D_i} (x - m_i)(x - m_i)^T \]

want to maximize: \[ J(W) = \frac{|W^T S_b W|}{|W^T S_w W|} \]

with \[ W = [w_1, w_2, \ldots, w_m] \]

\[ S_{b} w_i = \lambda_i S_{w} w_i \]

\[ m \leq c - 1 \]

Fisherface solution:

\[ W_{PCA} = \arg \max_{W} |W^T S_w W| \] where \[ S_{w} = \sum_{x} (x - m)(x - m)^T \]

\[ W_{FLD} = \arg \max_{W} |W^T S_B W| \]

\[ S_{w} \] is called the total scatter matrix
PCA vs. LDA (I)

- PCA:
  - The performance is weaker than correlation

- LDA:
  - LDA can be used for any kinds of classification problems
  - Ex. Glasses recognition

Experimental types:
- Extrapolation & Interpolation
- Leaving-one-out

PCA vs. LDA (II)

4. Other methods

- The combination of PCA & LDA: [Zhao, 1998]
  - Use PCA for noise cleaning and generalization when only a few samples in each class
- The use of 2-D PCA: [Yang, 2004]
  \[
  \Sigma = E[(A - E[A])^T(A - E[A])], \quad A: \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}
  \]
- Laplacianfaces: [He, 2005]
  - Extract the low-dimensional manifold structure
- Robust face recognition: [Wright, 2007]
  - Involved compressive sensing, sparse representation, and L1 minimization
  - Feature extraction is no longer important
Laplacianfaces

Robust face recognition

• Robust for occlusion, and the feature extraction is no longer important!

[Wright, 2007]
[Baraniuk, 2007]
[Wright, 2007]
5. Conclusion

PCA vs. LDA

- PCA is an unsupervised dimension reduction algorithm, while LDA is supervised
- PCA is good at outlier cleaning, and LDA could learn the within-class deviation
- These two methods only extract 1\textsuperscript{st} and 2\textsuperscript{nd} statistical moments
- The combination of PCA & LDA could enhance the performance
- PCA serves as the first-step processing of several kinds of face recognition technique
- Techniques of dimension reduction are frequently used in face recognition
Database

- FERET database

- Yale database (suitable for LDA)


Reference


