Eigenfaces (2)
Plan of the lecture

- PCA – repeated
- Back projection
- Feature vectors comparison
- Methods based on the Eigenfaces
Eigenfaces

- Feature extraction method
- Redundant information reduction (dimensionality reduction)
- Two stages:
  - training
  - projection (feature extraction)
- Possibility of back projection
Eigenfaces: training

\[
\begin{pmatrix}
C_{00} & \ldots & C_{0n} \\
\ldots & \ldots & \ldots \\
C_{n0} & \ldots & C_{nn}
\end{pmatrix}
\]

Normalised images → Covariance matrix → Eigenfaces
Eigenfaces: laboratory

- **Input data:**
  - normalised images, number and size

- **Implementation:**
  - covariance matrix
  - eigenvalues and eigenvectors (OpenCV)
  - output buffer – eigenvectors / eigenfaces

- **Testing:**
  - dimensionality reduction
Eigenfaces: feature extraction

Scalar products between normalised image and eigenvectors

Feature vector

K1
K2
K3
...

Face Recognition and Biometric Systems
Eigenfaces: feature extraction

- $\psi$ matrix can be cut to reduce dimensions

\[
\begin{pmatrix}
\psi \\
\psi'
\end{pmatrix}
\rightarrow
\begin{pmatrix}
\psi' \\
\psi''
\end{pmatrix}
\]

- Feature vector element is a scalar product:

\[
W_i = v_i^T \cdot x \\
W = \psi'^T \cdot x
\]

- Feature vector - cut projected vector $W$
Example

- **2-dimensional space:**
  - eigenvectors:
    
    \[
    \begin{bmatrix}
    \frac{\sqrt{2}}{2} \\
    \frac{\sqrt{2}}{2}
    \end{bmatrix}
    \quad \quad \quad \quad \quad
    \begin{bmatrix}
    -\frac{\sqrt{2}}{2} \\
    \frac{\sqrt{2}}{2}
    \end{bmatrix}
    \]
  - average vector \([0, 0]\)

- **Vectors projection:**
  - \([3; 1], [-2; -2], [10, 9]\)

- **Back projection**
Back projection

Feature vector -> face image

\[ x_p = \sum_{i=1}^{N'} w_i \cdot v_i + \mu \]

Projection error – difference between original and recovered image

\[ \epsilon = \| x - x_p \| \]
Back projection: 2D
Back projection: 2D
Back projection: 2D
Back projection: 2D
Back projection: face image

- Feature vector – face description
  - information reduction
- Back projection: face image recovered from feature vector
  - reduced information are lost
- Projection error:
  - depends on similarity to the training set
  - 2D example
  - face images
Back projection: detection

- Back projection of images:
  - face -> slightly modified face image
  - flower -> image similar to a face
- Back projection error is higher for non-face images
- Can be used as a verifier
  - threshold of accepted projection error
Feature vectors comparison

Similarity based on distance metric

\[ S(w_1, w_2) = \frac{1}{1 + \text{dist}(w_1, w_2)} \]

- Euclidean distance (norm L2)
- Mahalanobis distance
- Angle between vectors

Classifier-based similarity

- SVM, ANN
Feature vectors comparison

- Euclidean distance (L2 norm)
  - distance between two points in Euclidean space

\[ \text{dist}(\mathbf{w}_1, \mathbf{w}_2) = \sqrt{\sum_{i=1}^{N'} (w_{1i} - w_{2i})^2} \]
Feature vectors comparison

**Mahalanobis distance**
- variance normalised in all directions (a.k.a. whitening)

\[
\text{dist}(\mathbf{w}_1, \mathbf{w}_2) = \sqrt{\sum_{i=1}^{N'} \frac{(w'_{1i} - w'_{2i})^2}{\lambda_i}}
\]

\(\lambda\) - eigenvalue
Feature vectors comparison

- **Weak whitening:**

\[
\text{dist}(\mathbf{w}_1, \mathbf{w}_2) = \sqrt{\sum_{i=1}^{N'} \frac{(w_{1i} - w_{2i})^2}{\sqrt{\lambda_i}}}
\]

- **Eigenvalue filter:**

\[
\text{dist}(\mathbf{w}_1, \mathbf{w}_2) = \sqrt{\sum_{i=1}^{N'} (w_{1i} - w_{2i})^2 \frac{\lambda_i}{(\lambda_i + \lambda_{N'})^2}}
\]
Feature vectors comparison

Based on cosine of the angle

\[
dist(w_1, w_2) = \frac{w_1 \cdot w_2}{\|w_1\| \|w_2\|}
\]

- feature vector length not taken into consideration
Feature vectors comparison

- Classifiers
  - single vector can be classified
  - two or more classes
  - long training stage
- One person – one class
  - training necessary when gallery is changed
- Similarity between any two vectors
  - universal training
  - two vectors -> one vector
\[
\begin{align*}
\{ & K_{11} \\
& K_{12} \\
& \ldots \\
& K_{1n} \\
\} \\
\{ & K_{21} \\
& K_{22} \\
& \ldots \\
& K_{2n} \\
\}
\end{align*}
\]

\text{SVM}

\text{The same class}

\text{Different classes}
The same class

Different classes

\[
\{ \begin{align*}
K_{11} - K_{21} \\
K_{12} - K_{22} \\
\vdots \\
K_{1n} - K_{2n}
\end{align*} \}
\]

SVM
Feature vectors comparison

Training set for classifiers:
1. classified samples
2. intra-personal pairs
3. extra-personal pairs
4. differences within each pair
5. training with two sets: intra-personal and extra-personal

Face Recognition and Biometric Systems
Feature vectors comparison

- A pair of feature vectors:
  - many metrics, various results
  - metric as a separate feature extraction method
- Metrics fusion
  - weighted mean of single results
  - classifiers again
- Testing necessary
Eigenfaces – drawbacks

- holistic method
- face topology not taken into account
- statistical analysis of differences between images in the training set
- character of differences not taken into account
Example: PCA
Example: PCA not helpful
Example:
Linear Discriminant Analysis
Fisherfaces

- PCA finds main directions of variance
  - class identity not utilised

- Methods based on PCA which utilise class identity:
  - Linear Discriminant Analysis (LDA)
  - Fisherfaces
Fisherfaces

- Principal Component Analysis:
  - training set $\rightarrow$ covariance matrix

- Linear Discriminant Analysis:
  - classified training set $\rightarrow$ two covar. matrices
    - within-class covariance matrix
    - between-class covariance matrix
  - orthogonal basis from two matrices
Fisherfaces

**Between-class matrix**

\[ C_B = \sum_{i=1}^{C} M_i (\mu_i - \mu)(\mu_i - \mu)^T \]

- **\( C_B \)** - between-class covariance matrix
- **\( C \)** - number of classes
- **\( M_i \)** - number of images in \( i \)-th class
- **\( \mu \)** - average image
- **\( \mu_i \)** - average image of \( i \)-th class

Face Recognition and Biometric Systems
Fisherfaces

Within-class covariance matrix

\[ C_W = \sum_{i=1}^{C} \sum_{x_k \in X_i} (x_k - \mu_i)(x_k - \mu_i)^T \]

- \( C_W \) – within-class covariance matrix
- \( C \) – number of classes
- \( X_i \) – set of images which belong to \( i \)-th class
- \( x_k \) – \( k \)-th image which belongs to \( i \)-th class
- \( \mu_i \) – average image of \( i \)-th class
**Fisherfaces**

**PCA:**
\[
\psi = \arg \max_{\psi} \left| \psi^T C \psi \right|
\]
\[
C \cdot \nu = \lambda \cdot \nu
\]

\(\psi\) - eigenvectors matrix (vectors in columns)

**LDA:**
\[
\psi = \arg \max_{\psi} \left| \psi^T C_B \psi \right| / \left| \psi^T C_W \psi \right|
\]
\[
C_B \cdot \nu = \lambda \cdot C_W \cdot \nu \quad C_W^{-1} \cdot C_B \cdot \nu = \lambda \cdot \nu
\]

Face Recognition and Biometric Systems
Fisherfaces

- LDA – hard to find inverse matrix
- Fisherfaces – improved approach:
  - PCA for dimensionality reduction
  - LDA for finding optimal orthogonal basis

\[
\psi = \arg\max \psi^T \psi'_{PCA} C_B \psi'_{PCA} \psi - \psi^T \psi'_{PCA} C_W \psi'_{PCA} \psi
\]
Fisherfaces

Feature extraction in the Fisherfaces:

1. Feature vector calculated by PCA
   - normalised image as an input
   - dimensionality reduction

2. Feature vector calculated by LDA
   - PCA feature vector as an input
   - rotation of feature vector
   - no dimensionality reduction
Bayesian Matching

Vectors similarity based on probability of their difference classification

\[ S(I_1, I_2) = P(\Delta \in \Omega_I) = P(\Omega_I \mid \Delta) \]

\[ \Delta = I_1 - I_2 \]

- \( \Omega_I \) – set of intra-personal pairs
- \( \Omega_E \) – set of extra-personal pairs
Bayesian Matching

\[ P(\Omega_I \mid \Delta) = \frac{P(\Delta \mid \Omega_I)P(\Omega_I)}{P(\Delta \mid \Omega_I)P(\Omega_I) + P(\Delta \mid \Omega_E)P(\Omega_E)} \]

\[ P(\Delta \mid \Omega) \rightarrow \text{probability of observing a given difference in a defined set of differences} \]
- function of PCA back projection error \(- \varepsilon(\Delta)\)

\[ P(\Delta \mid \Omega) \sim e^{-\varepsilon(\Delta)^2} \]
Bayesian Matching

- Two classes of image pairs
  - intra- and extra-personal

- Differences generated from pairs
  - two classes of pairs

- PCA used for both classes separately
  - two image spaces

Face Recognition and Biometric Systems
Face Recognition and Biometric Systems
Bayesian Matching

- Image difference recognition
  - Dual Eigenfaces
- Difference distance from two image spaces
- Bayesian Matching – a slow method
  - image difference calculated for every comparison
  - possibility of applying other method for selecting candidates (n most similar images)
Local PCA

- Based on detected features
  - eyes, nose, mouth
- PCA for features
  - small part of face image
  - analysis of small images (eigeneyes, eigennoses, etc.)
- Less dimensional spaces
- Lower effectiveness, but supports the Eigenfaces
Local PCA

K1
K2
K3
K4

Face Recognition and Biometric Systems
Other methods

- Local Feature Analysis
- 2D PCA, 2D LDA
- Independent Component Analysis
Thank you for your attention!