Seminar Modellierung und Simulation biologischer Systeme

Hauptkomponentenanalyse: automatische Gesichtserkennung

Prof. Dr. Arne Nägel

Malaz al Mahdi
Yasser Saadaoui

Einleitung

Die Gesichtserkennung hat sich zu einem beliebten Forschungsbereich in der Computer Vision entwickelt und ist eine der erfolgreichsten Anwendungen der Bildanalyse und des Bildverständnisses. Sie umfasst zwei Hauptaufgaben:

•**Gesichtsidentifikation:** Bei einem gegebenen Gesichtsbild, das einer Person in einer Datenbank zugeordnet ist, muss bestimmt werden, wessen Bild es ist.

•**Gesichtsverifikation:** Bei einem Gesichtsbild, das möglicherweise nicht zur Datenbank gehört, muss überprüft werden, ob es tatsächlich von der Person stammt, als die es angegeben wird.

Frage

Was ermöglicht uns die effektive Durchführung dieser beiden Hauptaufgaben von der Gesichtserkennung?

Die ***Hauptkomponentenanalyse*** (PCA) ist ein Werkzeug zur ***Dimensionsreduktion***, das verwendet werden kann, um eine große Menge von Variablen auf eine kleine Menge zu reduzieren, die immer noch den ***größten Teil der Informationen*** aus der großen Menge enthält.

Vorgehen

1. Datenbank einrichten
2. Berechnung der Eigengesichter aus der Datenbank
3. Projektion der Gesichtsbilder auf den Gesichtsraum

Datenbank einrichten

Erwerb einer Anfangsmenge von Gesichtsbildern, die als Trainingsset dienen:

- Alle Bilder müssen auf dieselbe Auflösung geschnitten werden (M x N)
- Der vorherige Punkt sollte so bewerkstelligt werden, sodass das Gesicht möglichst das ganze Bild ausfüllt
- Alle Bilder müssen in schwarz-weiß Bild konvertiert werden

Berechnung der Eigengesichter

1. ***Alle Bilder von der Datenbank als Vektor repräsentieren***
   (Γ1, Γ2, . . . , ΓM)

$$= \begin{pmatrix} a_1 \\ a_2 \\ \mathrm{M} \\ a_{N^2} \end{pmatrix}$$

= Γ1

$$= \begin{pmatrix} b_1 \\ b_2 \\ \mathrm{M} \\ b_{N^2} \end{pmatrix}$$

= Γ2

$$= \begin{pmatrix} c_1 \\ c_2 \\ \mathrm{M} \\ c_{N^2} \end{pmatrix}$$

= Γ3

Berechnung der Eigengesichter

2. Durchschnittsgesicht

$$\Psi = \frac{1}{M} \sum_{n=1}^{M} \Gamma_n , \quad M = \text{Anzahl der Bilder}$$

3. Differenzgesichter

Jedes Bild von der Anfangsmenge unterscheidet sich von dem Durchschnittsgesicht durch

$$\Phi_n = \Gamma_n - \Psi \text{,, n = 1,..,M}$$

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

4. Kovarianzmatrix

Die Matrix wird durch die Formel berechnet

$$C = \frac{1}{M} \sum_{n=1}^{M} \Phi_n \Phi_n^T = AA^T$$

Wobei $A = [\Phi_1, \Phi_2, \ldots\ldots, \Phi_M]$

$\Rightarrow$ ***Die Matrix ist sehr schwer zu berechnen***

Berechnung der Eigengesichter (Tafel Berechnung)

Bestimmung der Eigenvektoren und Eigenwerte der Kovarianzmatrix

Angenommen $L = A^T A$ mit V_n: Eigenvektoren
und λ_n: Eigenwerte.

V_n Eigenvektor von $L \Rightarrow L \cdot V_n = \lambda_n V_n$

$L \cdot V_n \Rightarrow \lambda_n V_n \Rightarrow A^T A \cdot V_n = \lambda_n V_n$

$\Rightarrow A \cdot A^T A V_n = \lambda_n A \cdot V_n$

$\Rightarrow A \cdot A^T (A \cdot V_n) = \lambda_n (A \cdot V_n)$

Unsere Kovarianzmatrix $C = A \cdot A^T$
dann:

$C \cdot A V_n = \lambda_n (A \cdot V_n)$ (Da $A \cdot V_n$ ein Vektor ist)

$\Rightarrow A \cdot V_n$ sind Eigenvektoren von C und λ_n sind die Eigenwerte von C.

Warum ist die Matrix L einfacher zu berechnen als C (die Kovarianzmatrix)?

Warum ist $L = A^T A$ einfacher zu berechnen als $C = A \cdot A^T$?

Nehmen wir an, das wir 100 Bilder mit Auflösung von 100 x 100 Pixel

- A hat die Dimension 10.000 x 100
- A^T hat die Dimension 100 x 10.000

$\Rightarrow$ $A \cdot A^T$ ist eine 10.000 x 10.000 Matrix
$A^T \cdot A$ ist eine 100 x 100 Matrix

Bestimmung der Eigenvektoren und Eigenwerte der Kovarianzmatrix

Angenommen L = ATA , mit den Eigenvektoren. v_n und den Eigenwerten λ_n

$$A^T A v_n = \lambda_n v_n \Rightarrow AA^T A v_n = \lambda_n A v_n$$

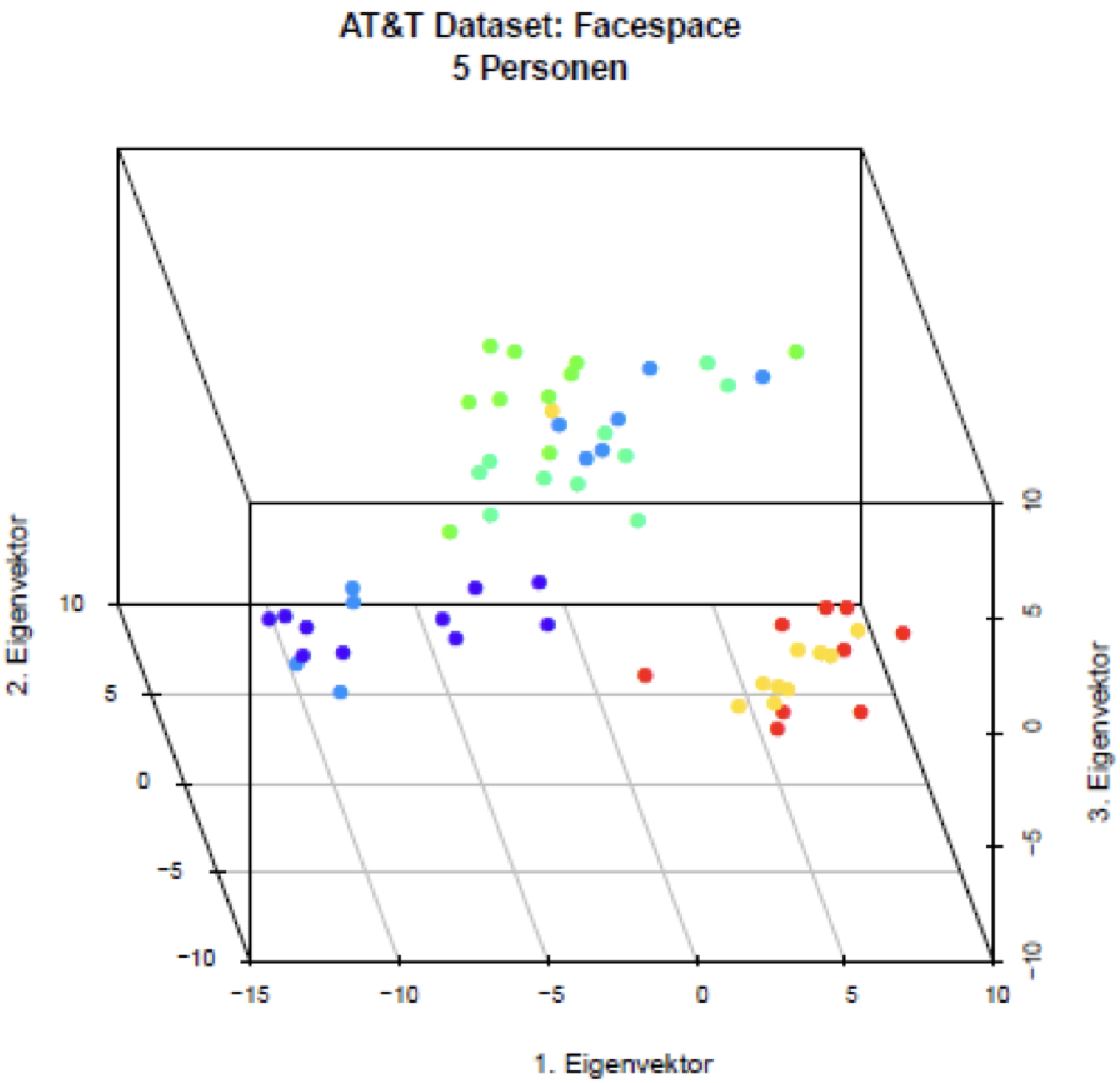
⇨ Av_n sind die Eigenvektoren von C. Daher $\mu_n = \sum_{k=1}^{M} v_{nk} \Phi_k = Av_n$

Gesichtsraum

- Die k besten Eigenvektoren (mit den k größten Eigenwerte)
- Bilder bekannter Personen werden durch eine einfache Operation
  $\omega_{ik} = \mu_k^T (\Gamma_i - \Psi)$ in den "Gesichtsraum" projiziert.
- Jedes Bild kann durch einen Mustervektor repräsentiert.

$$\Omega_i = [\omega_{i1}, \omega_{i2}, \ldots\ldots, \omega_{iM'}]$$

Gesichtsraum



Frage

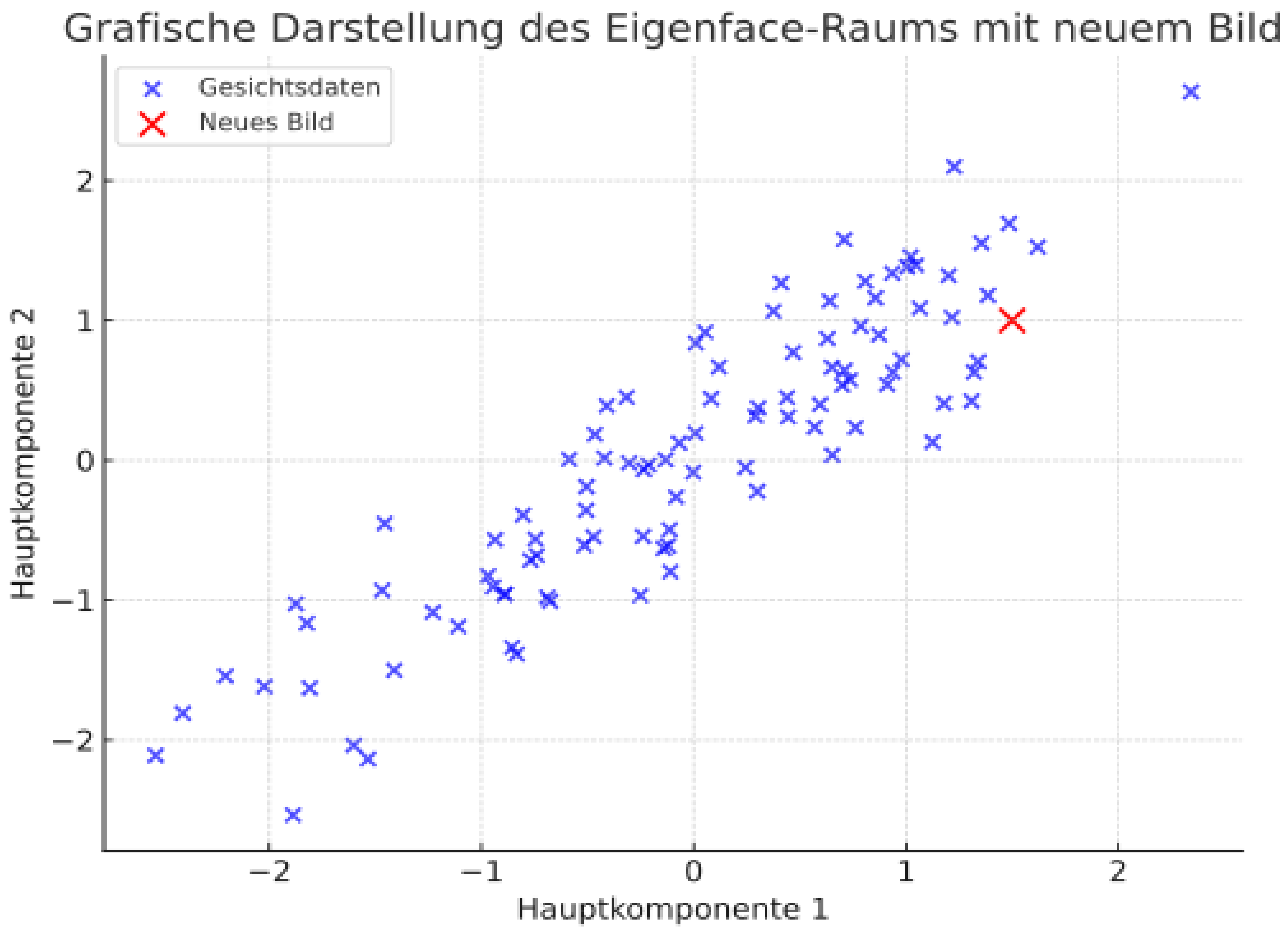
Gehört ein neu erhaltenes Gesichtsbild zu einer der Personen in unserer bestehenden Datenbank?

Neues Bild

- **Projektion:** Das neue Bild wird auf den Eigenface-Raum projiziert, um seinen Mustervektor zu erhalten

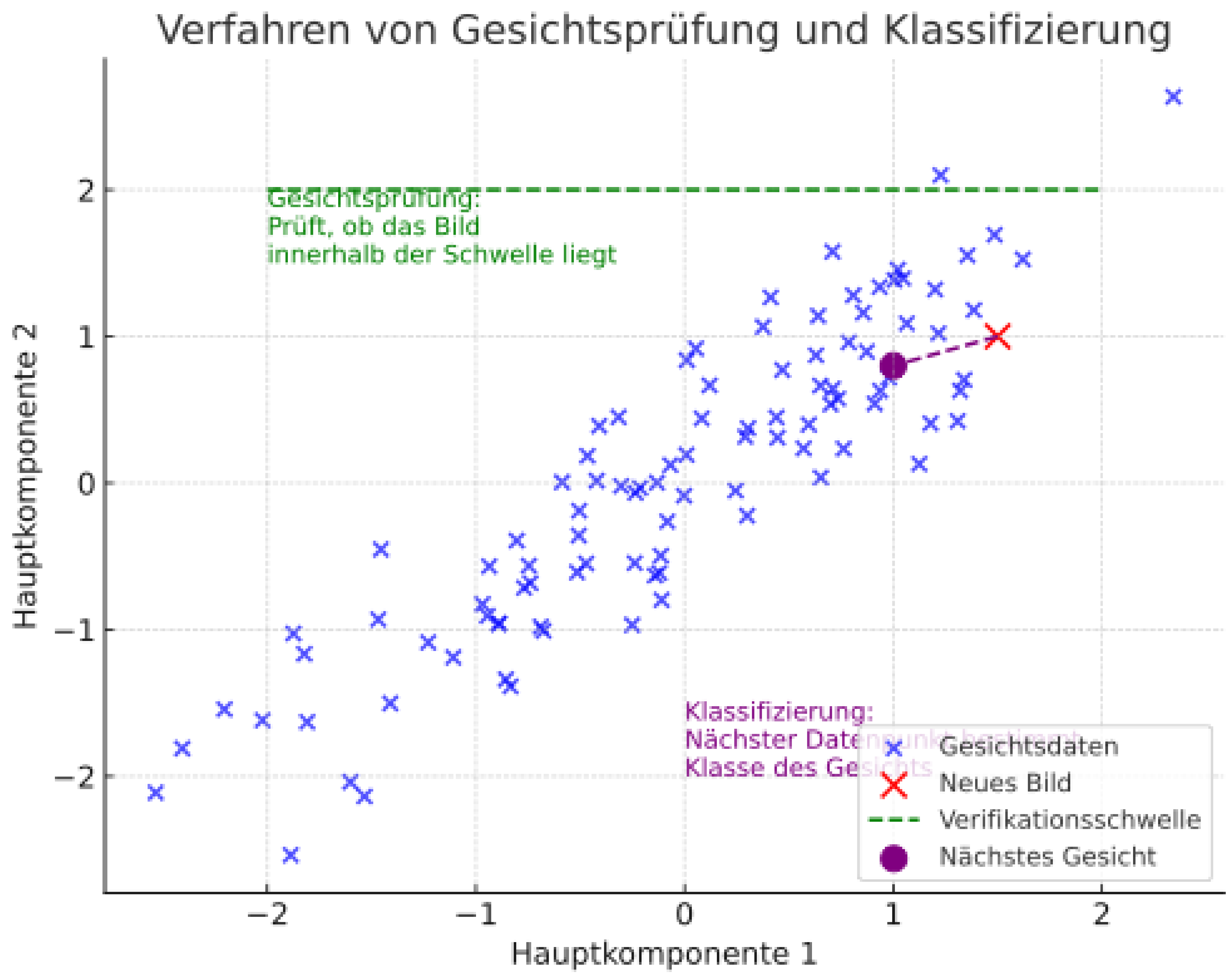
$$\Omega = [\omega_1, \omega_2, \ldots\ldots, \omega_{M'}]$$

- Wobei $\omega_{ik} = \mu_k^T (\Gamma_i - \Psi)$



Gesichtserkennung mit Eigengesichtern
Neues Bild

Gesichtsprüfung: Es wird geprüft, ob das Bild ein Gesicht darstellt, indem die Distanz zu einem vordefinierten Schwellenwert verglichen wird.



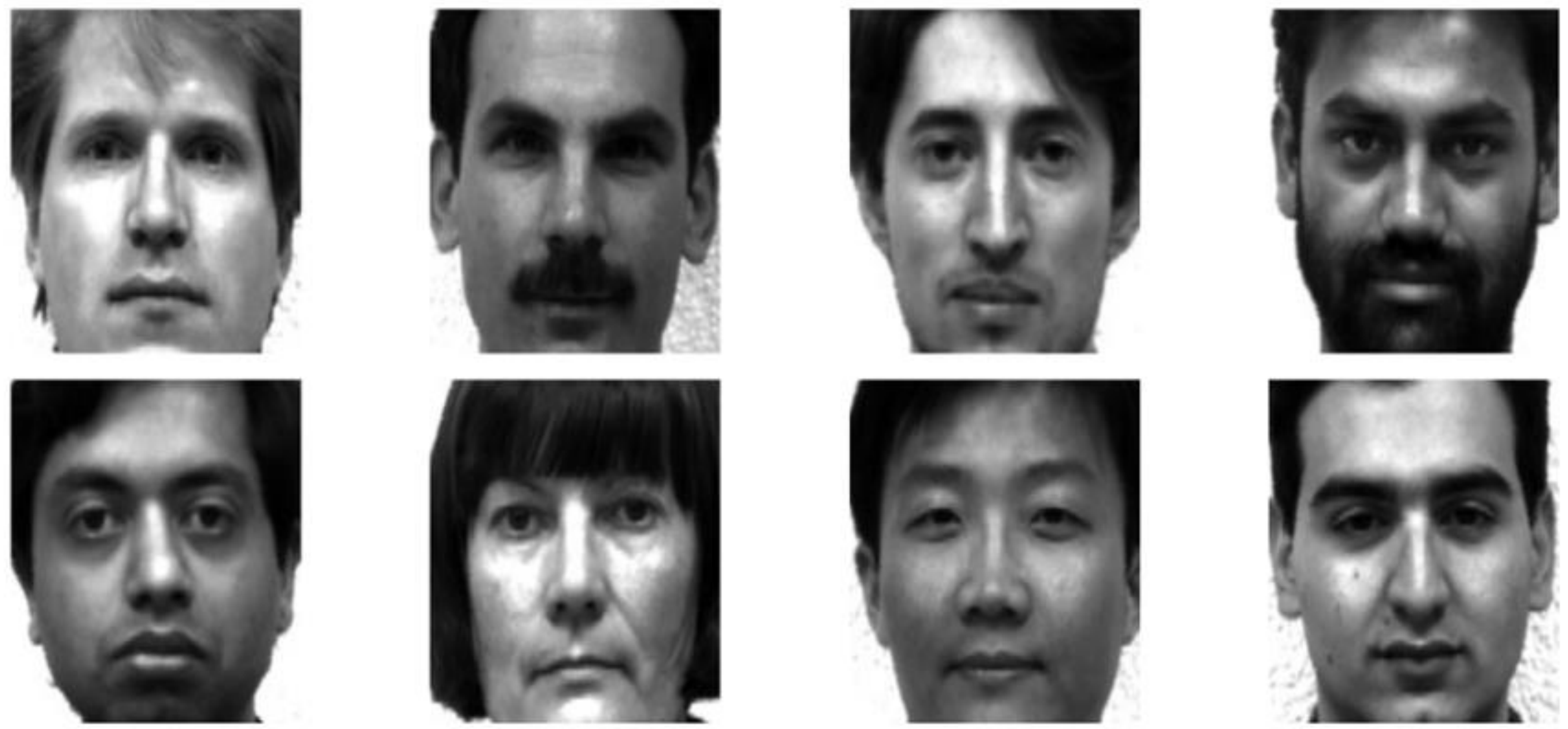
- **Klassifizierung:** Das Bild wird als bekanntes Individuum oder als unbekannt klassifiziert, basierend auf der minimalen Distanz zu den Mustervektoren bekannter Individuen.

$$\varepsilon_k^2 = \left\| (\Omega - \Omega_k)^2 \right\|$$

- Falls argmin(ei) < Schwellpunkt, ist das neue Gesicht erkannt. Sonst, ist es als unbekannt klassifiziert.

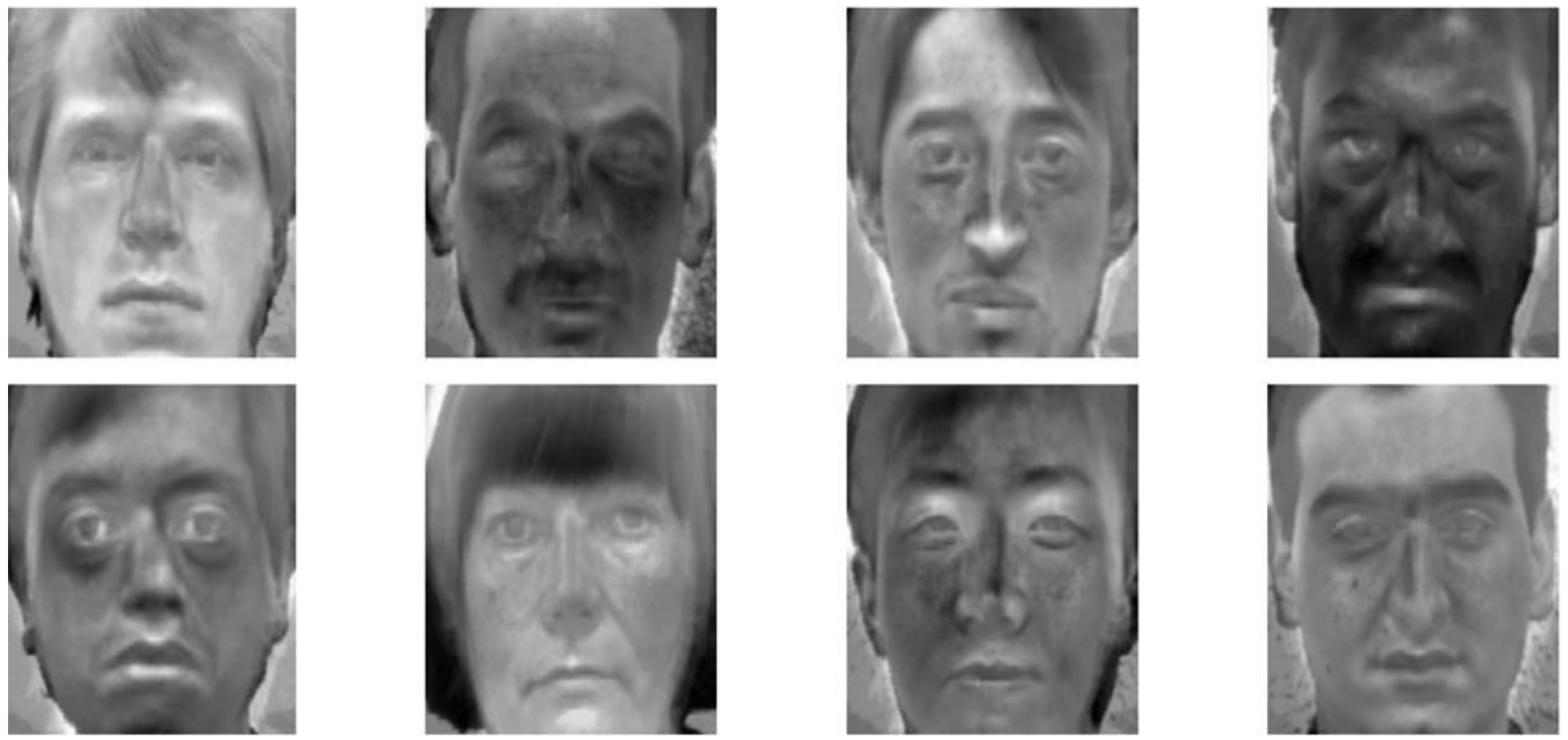
Implementierung

Trainingsbilder

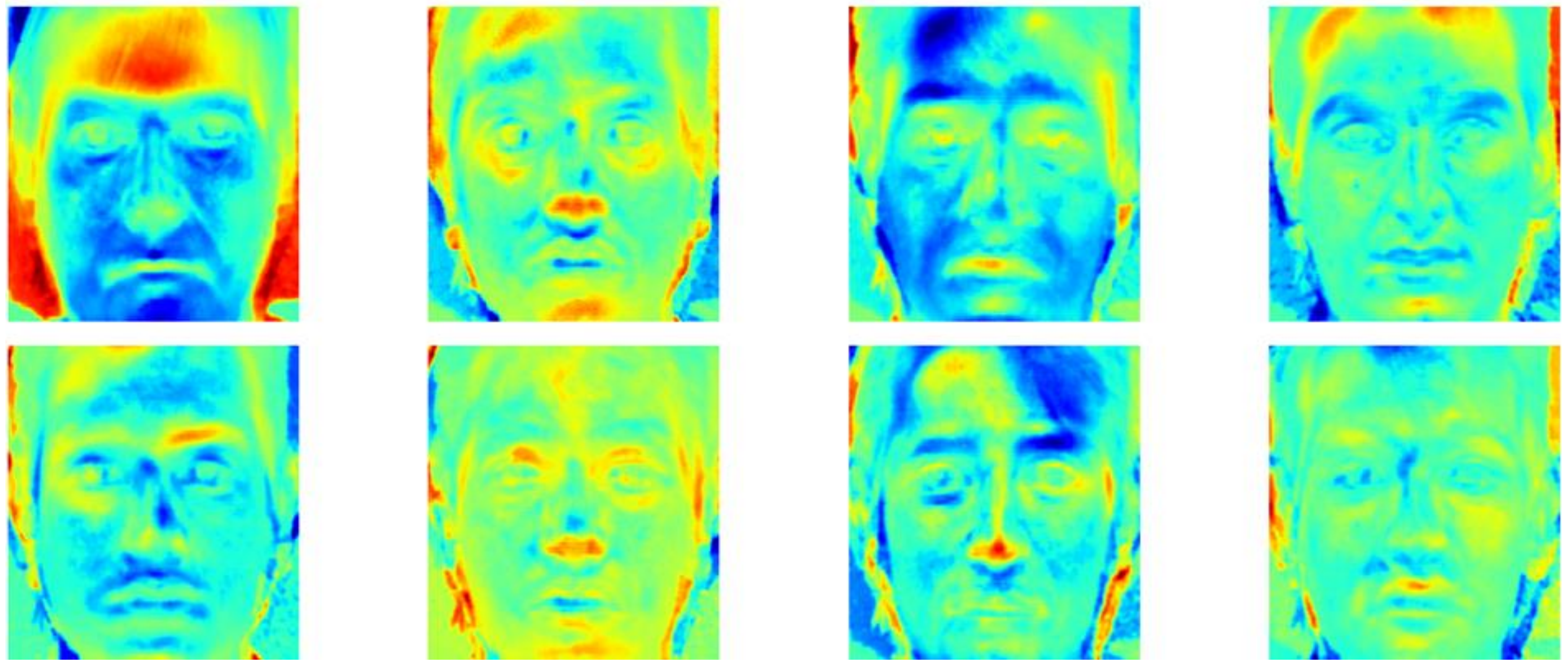


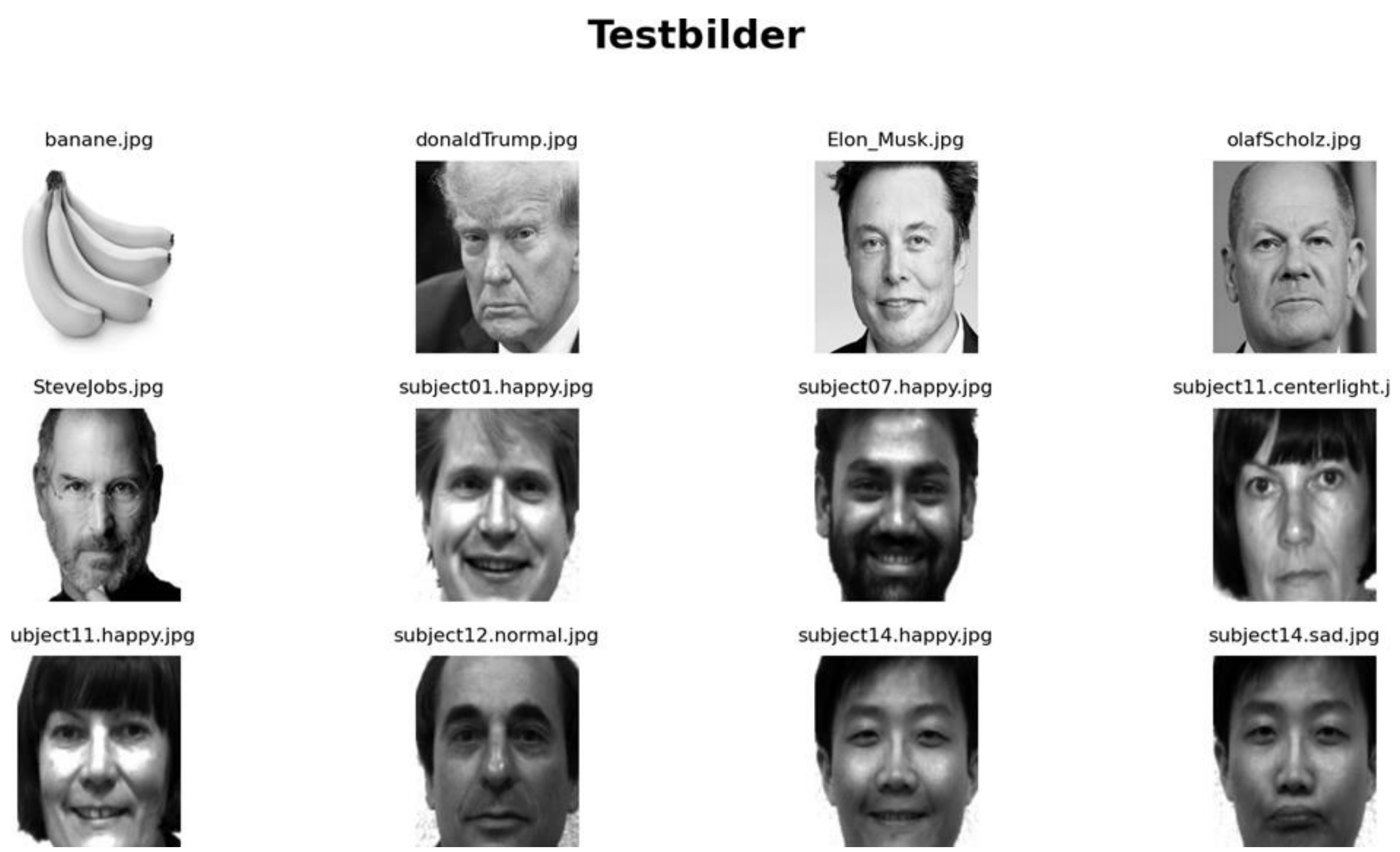
Durchschnittsgesicht

Normalisierte Gesichter

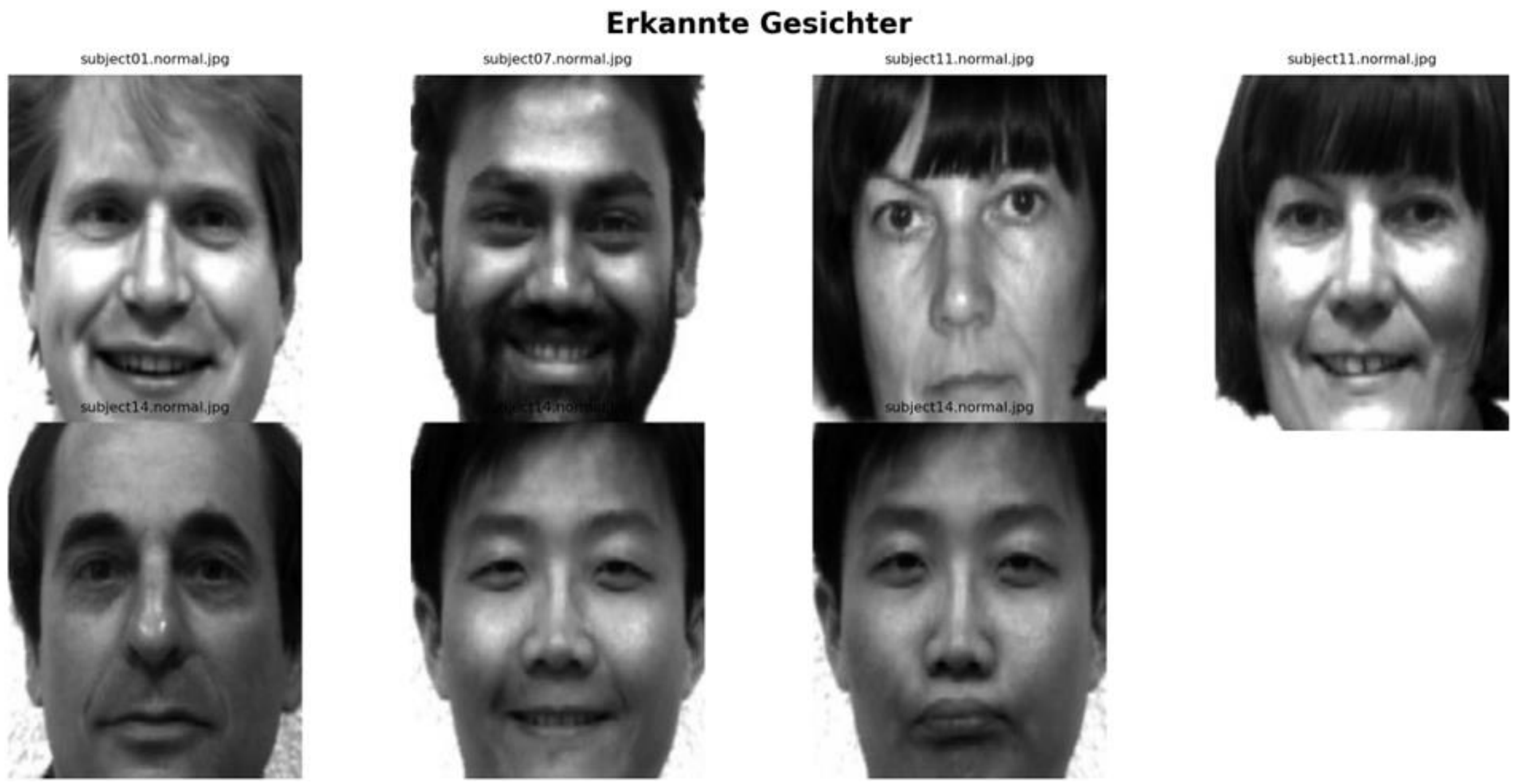


Eigenfaces




Testbilder
banane.jpg
donaldTrump.jpg
Elon_Musk.jpg
olafScholz.jpg
SteveJobs.jpg
subject01.happy.jpg
subject07.happy.jpg
subject11.centerlight.j
ubject11.happy.jpg
subject12.normal.jpg
subject14.happy.jpg
subject14.sad.jpg


Unbekannte oder kein Gesicht
Kein Gesicht:
banane.jpg
Unbekannt:
donaldTrump.jpg
Dist=99704566.41
Unbekannt:
Elon_Musk.jpg
Dist=71369427.56
Unbekannt:
olafScholz.jpg
Dist=85741523.02
Unbekannt:
SteveJobs.jpg
Dist=79180476.37


Erkannte Gesichter
subject01.normal.jpg
subject07.normal.jpg
subject11.normal.jpg
subject11.normal.jpg
subject14.normal.jpg
subject14.normal.jpg

Anwendung

- ***Sicherheits- und Überwachungssysteme***
- ***Personalisierte Benutzererfahrung***
- ***Forensik und Kriminalitätsbekämpfung***
  Smartphone und Computer
  Personalisierte Anzeigen
- ***Medizinische Anwendung***
  Diagnose genetischer Erkrankungen
  Psychiatrie und Neurologie

Danke für Ihre Aufmerksamkeit und Ihr Zuhören

Bei Fragen stehen wir gerne zur Verfügung

Quellen

Verwendete Quellen für die Gesichtserkennung:

- OpenCV Dokumentation für Gesichtserkennung und Zuschneiden:
  - 📌 https://github.com/opencv/opencv
- Verschiedene Implementierungen von Eigenfaces zur Gesichtserkennung:
  - 📌 https://github.com/agyorev/Eigenfaces/tree/master
  - 📌 https://github.com/vutsalsinghal/EigenFace/tree/master/Dataset
  - 📌 https://github.com/zwChan/Face-recognition-using-eigenfaces/blob/master/eigenFace.py

Quellen für die mathematische Theorie (PCA & Eigenfaces):

- **Bücher & wissenschaftliche Arbeiten**:
  - 📖 *Turk & Pentland (1991)* - "Eigenfaces for Recognition"
  - 📖 *Sirovich & Kirby (1987)* - "Low-dimensional Procedure for the Characterization of Human Faces
  - 📌 https://www.cs.hs-rm.de/~ulges/teaching/15MLSEM/files/folien/schmidt.pdf
  - 📌 *https://de.slideshare.net/slideshow/eigenface-for-face-recognition-presentation/758807?_gl=1*kkpo6c*_gcl_au*MTI3MTc1MDYwNS4xNzM4MDAwMDQz*

Datenbank für Trainingsbilder:

- 📌 https://github.com/vutsalsinghal/EigenFace/tree/master/Dataset