

Gesichtsdetektierung und Gesichtserkennung



Stefan Güttel
stefan@guettel.com

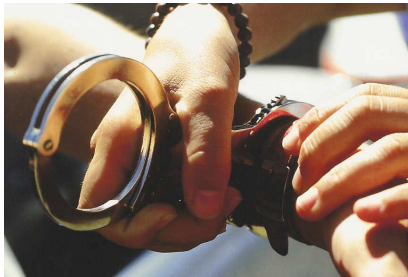
Fakultät für Mathematik und Informatik
Institut für Numerische Mathematik und Optimierung

Spezialistenlager des GCG Halle, 15.01.2009

Achim bricht bei Oma Frieda in die Gartenlaube ein...



Doch dummerweise ruft der Laubennachbar die Polizei...



Auf dem Präsidium wird Achim fotografiert. Sein Foto wird in einer riesigen Datenbank der Polizei gespeichert, in der schon Tausende andere vorbestrafte Personen eingetragen sind.

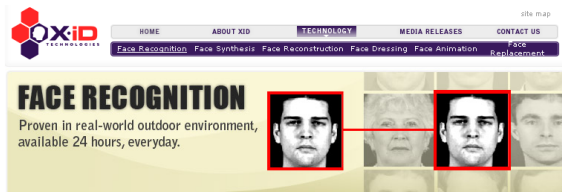


3 Monate später: Einbruch im Supermarkt.
Die Überwachungskamera registriert einen Tatverdächtigen.



Gesichtserkennung könnte helfen, die Polizei-Datenbank daraufhin zu prüfen, ob der Tatverdächtige darin enthalten ist.

Weitere Anwendung: Zugangskontrolle



HOME ABOUT XID TECHNOLOGY MEDIA RELEASES CONTACT US

Face Recognition Face Synthesis Face Reconstruction Face Dressing Face Animation Face Replacement

FACE RECOGNITION

Proven in real-world outdoor environment, available 24 hours, everyday.

Access Control

Logon Control

Civil ID

Tag and Face

For more info please [contact us](#), or email us directly at: info@xidtech.com

SDKs are now available for licensing. Kindly contact us here for more information.

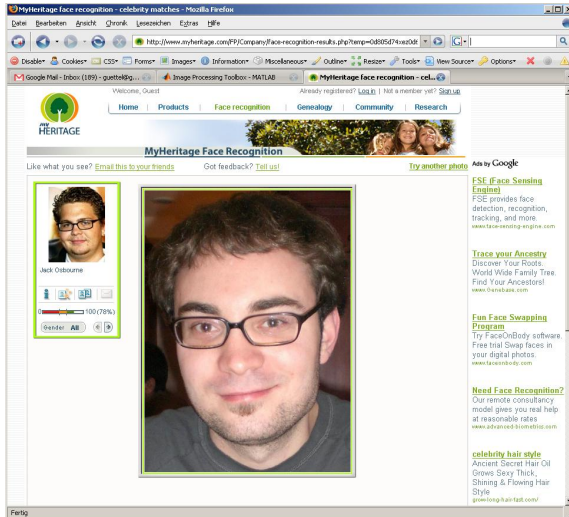
XID has deployed the world's largest face processing system onto the most complex environment known today.

"XID Technologies is a Singapore-based face recognition vendor that has recently deployed a facial recognition system at an immigrant workers dormitory in Singapore, controlling access for 6,000 blue-collar workers, outdoors, 24/7." Homeland Security Asia Magazine – October 2004

Demonstration of the real world face recognition deployment



Weitere Anwendung: "Ahnenforschung"



The screenshot shows a web browser window displaying the MyHeritage face recognition interface. The page title is "MyHeritage face recognition - celebrity matches - Mozilla Firefox". The URL is "http://www.myheritage.com/FP/Company/face-recognition-results.php?temp=04805d74ae0d8". The page content includes a navigation menu with "Home", "Products", "Face recognition", "Genealogy", "Community", and "Research". The main heading is "MyHeritage Face Recognition". Below this, there are links for "Like what you see? Email this to your friends", "Got feedback? Tell us!", and "Try another photo". The central part of the page features a small reference photo of Jack Osbourne on the left and a larger photo of a man with glasses on the right. Below the reference photo, there is a "Gender" indicator showing "AB" and a "100 (78%)" match percentage. On the right side, there are several advertisements: "FSE (Face Sensing Engine)", "Trace your Ancestry", "Fun Face Swapping Program", "Need Face Recognition?", and "celebrity hair style".

Weitere Anwendung: Klassifizierung in Bilddatenbanken


Erweiterte Bildsuche
[Hilfe zu Bildern](#) | [Über Google](#)

Ergebnisse finden mit **allen** Wörtern
 mit der **genauen** Wortgruppe
 mit **irgendeinem** der Wörter
ohne die Wörter

Content-Typen Suche nach Bildern mit Beliebiger Content Nachrichten-Content Gesichter

Größe Ausgabe von Bildern mit der Größe

Dateiarten Nur Bilder ausgeben, die formatiert sind als

Farben Ausgabe von Bildern nur in

Domains Ausgabe von Bildern folgender Website oder Domain

SafeSearch Kein Filter Moderate Filterung verwenden Strikte Filterung verwenden

©2008 Google

Weitere Anwendung: Gesichtsdetektierung in Digicams



Grundsätzlich verschieden:

Gesichtsdetektierung Wo auf dem Bild ist ein Gesicht zu sehen?

Gesichtserkennung Wer ist auf dem Bild zu sehen?



Jahr 2002:



Airport anti-terror systems flub tests

By Richard Willing, USA TODAY

Camera technology designed to spot potential terrorists by their facial characteristics at airports failed its first major test, a report from the airport that tested the technology shows.

Last year, two separate face-recognition systems at Boston's Logan Airport failed 96 times to detect volunteers who played potential terrorists as they passed security checkpoints during a three-month test period, the airport's analysis says. The systems correctly detected them 153 times.

The airport's report calls the rate of inaccuracy "excessive." The report was completed in July 2002 but not made public. The American Civil Liberties Union obtained a copy last month through a Freedom of Information Act request.

Logan is where 10 of the 19 terrorists boarded the flights that were later hijacked Sept. 11, 2001.

The airport is now testing other security technology, including infrared cameras and eyeball scans,

Face recognition works by matching faces picked up by surveillance cameras with pictures stored

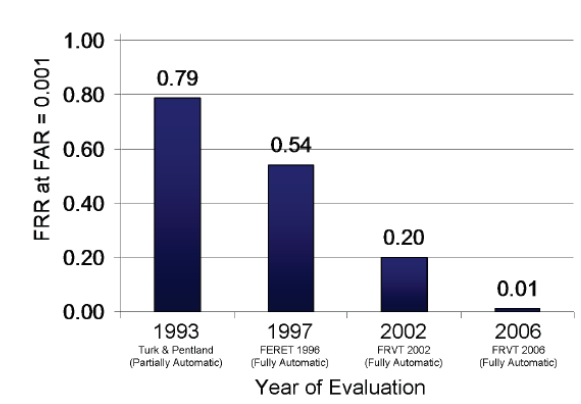


Figure 1: THE REDUCTION IN ERROR RATE FOR STATE-OF-THE-ART FACE RECOGNITION ALGORITHMS AS DOCUMENTED THROUGH THE FERET, THE FRVT 2002, AND THE FRVT 2006 EVALUATIONS.

Bevor es richtig losgeht...

... sind einige Grundbegriffe aus der *Linearen Algebra* notwendig:

Eine *Matrix* ist eine Anordnung von Zahlen in *Zeilen* und *Spalten*, z.B.

$$A = \begin{bmatrix} 2 & 0 & 3 \\ -1 & 7 & 2 \end{bmatrix}.$$

Die Matrix A hat 2 Zeilen und 3 Spalten. Wir schreiben $A \in \mathbb{R}^{2 \times 3}$.
Matrizen werden mit Großbuchstaben bezeichnet.

Ein Sonderfall von Matrizen sind die *Vektoren*:

Ein Vektor ist eine Matrix mit genau einer Spalte, z.B.

$$\mathbf{b} = \begin{bmatrix} -2 \\ 1 \\ -3 \end{bmatrix}.$$

Dieser Vektor hat die *Dimension* 3.

Für Vektoren verwenden wir **fette** Kleinbuchstaben.

Ein Sonderfall von Matrizen sind die *Vektoren*:

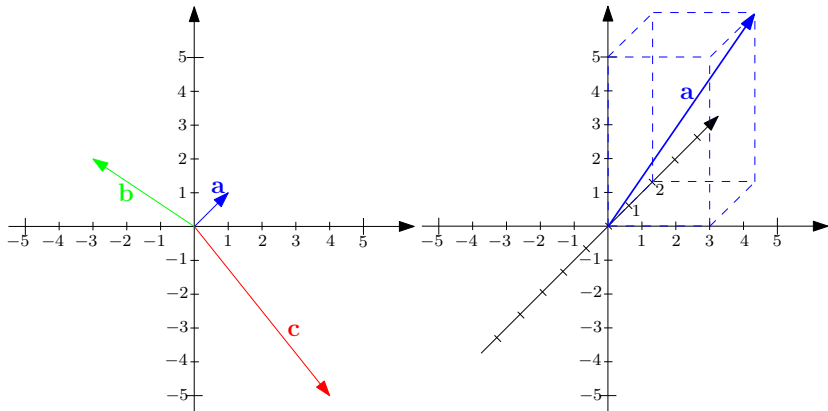
Ein Vektor ist eine Matrix mit genau einer Spalte, z.B.

$$\mathbf{b} = \begin{bmatrix} -2 \\ 1 \\ -3 \end{bmatrix}.$$

Dieser Vektor hat die *Dimension* 3.

Für Vektoren verwenden wir **fette** Kleinbuchstaben.

Vektoren bis zur Dimension 3 kann man sich gut in einem Koordinatensystem veranschaulichen.



Matrizen (und Vektoren) **mit gleicher Zeilen- und Spaltenzahl** kann man *addieren* genauso wie man Zahlen addiert. Dabei addiert man einfach elementweise:

$$\begin{bmatrix} -2 & 1 & 3 \\ 4 & -3 & 0 \end{bmatrix} + \begin{bmatrix} 1 & 2 & -3 \\ 2 & 2 & 1 \end{bmatrix} = \begin{bmatrix} -1 & 3 & 0 \\ 6 & -1 & 1 \end{bmatrix}.$$

Subtrahieren geht analog:

$$\begin{bmatrix} -2 & 1 & 3 \\ 4 & -3 & 0 \end{bmatrix} - \begin{bmatrix} 1 & 2 & -3 \\ 2 & 2 & 1 \end{bmatrix} = \begin{bmatrix} -3 & -1 & 6 \\ 2 & -5 & -1 \end{bmatrix}.$$

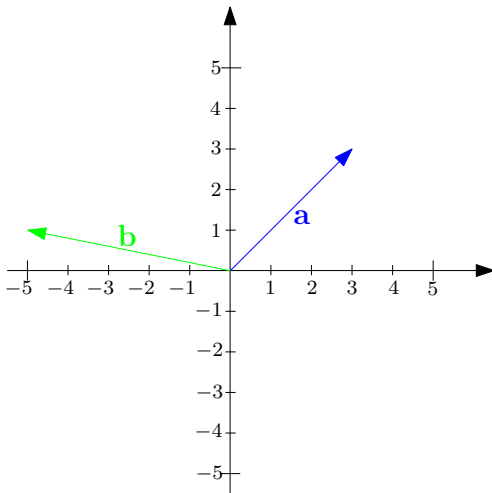
Matrizen (und Vektoren) **mit gleicher Zeilen- und Spaltenzahl** kann man *addieren* genauso wie man Zahlen addiert. Dabei addiert man einfach elementweise:

$$\begin{bmatrix} -2 & 1 & 3 \\ 4 & -3 & 0 \end{bmatrix} + \begin{bmatrix} 1 & 2 & -3 \\ 2 & 2 & 1 \end{bmatrix} = \begin{bmatrix} -1 & 3 & 0 \\ 6 & -1 & 1 \end{bmatrix}.$$

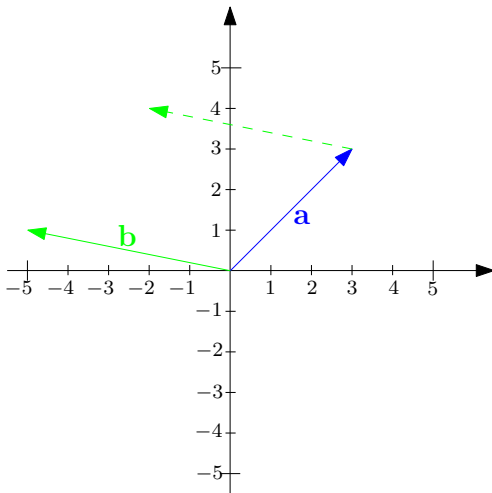
Subtrahieren geht analog:

$$\begin{bmatrix} -2 & 1 & 3 \\ 4 & -3 & 0 \end{bmatrix} - \begin{bmatrix} 1 & 2 & -3 \\ 2 & 2 & 1 \end{bmatrix} = \begin{bmatrix} -3 & -1 & 6 \\ 2 & -5 & -1 \end{bmatrix}.$$

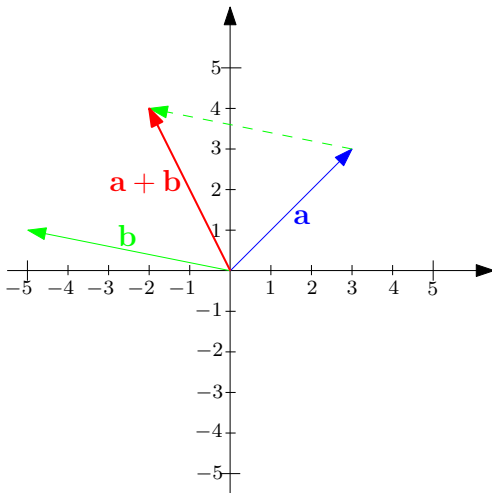
Die Addition von Vektoren lässt sich geometrisch interpretieren:



Die Addition von Vektoren lässt sich geometrisch interpretieren:



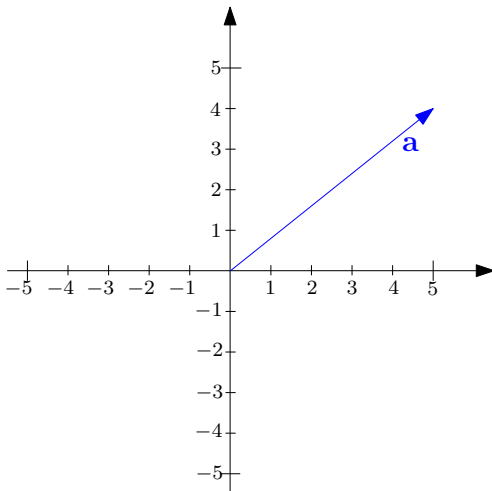
Die Addition von Vektoren lässt sich geometrisch interpretieren:



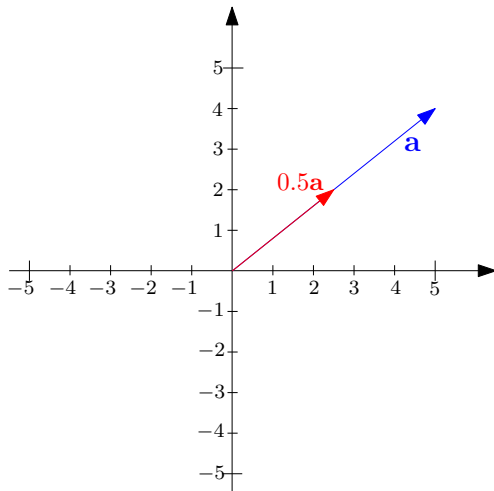
Vielfache von Matrizen (und Vektoren) bildet man, indem man jeden Eintrag der Matrix vervielfacht:

$$2 * \begin{bmatrix} -2 & 1 & 3 \\ 4 & -3 & 0 \end{bmatrix} = \begin{bmatrix} -4 & 2 & 6 \\ 8 & -6 & 0 \end{bmatrix}$$

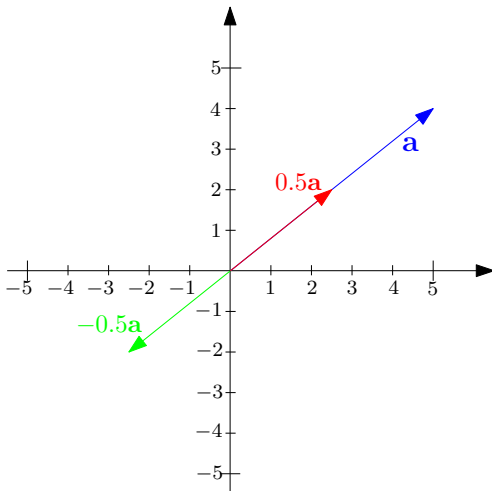
Bei Vielfachen von Vektoren ändert sich nur deren Länge und/oder Orientierung, nicht aber die Richtung:



Bei Vielfachen von Vektoren ändert sich nur deren Länge und/oder Orientierung, nicht aber die Richtung:



Bei Vielfachen von Vektoren ändert sich nur deren Länge und/oder Orientierung, nicht aber die Richtung:



Wir definieren die (*Frobenius-*) Norm $\| \cdot \|$ einer Matrix

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \cdots & a_{m,n} \end{bmatrix}$$

als

$$\|A\| = \sqrt{\sum_{j=1}^n \sum_{i=1}^m a_{i,j}^2},$$

also "Quadriere alle Einträge, summiere sie und berechne die Quadratwurzel der Summe."

Die Norm ist ein Maß dafür, wie weit eine Matrix von der Nullmatrix O (eine Matrix in der alle Einträge = 0 sind) "entfernt" ist.

Beispiel: Sei

$$A = \begin{bmatrix} 1 & 0 & -2 \\ 3 & -1 & 0 \end{bmatrix},$$

dann ist die Norm von A

$$\|A\| = \sqrt{1^2 + 3^2 + 0^2 + (-1)^2 + (-2)^2 + 0^2} = \sqrt{15}.$$

Frage: Wie groß ist $\|A - A\|$?

Für jede Matrix A gilt:

- $\|A\| \geq 0$,
- $\|A\| = 0$ genau dann, wenn $A = O$,
- für jede Zahl α gilt $\|\alpha A\| = |\alpha| \|A\|$.

Beispiel: Sei

$$A = \begin{bmatrix} 1 & 0 & -2 \\ 3 & -1 & 0 \end{bmatrix},$$

dann ist die Norm von A

$$\|A\| = \sqrt{1^2 + 3^2 + 0^2 + (-1)^2 + (-2)^2 + 0^2} = \sqrt{15}.$$

Frage: Wie groß ist $\|A - A\|$?

Für jede Matrix A gilt:

- $\|A\| \geq 0$,
- $\|A\| = 0$ genau dann, wenn $A = O$,
- für jede Zahl α gilt $\|\alpha A\| = |\alpha| \|A\|$.

Beispiel: Sei

$$A = \begin{bmatrix} 1 & 0 & -2 \\ 3 & -1 & 0 \end{bmatrix},$$

dann ist die Norm von A

$$\|A\| = \sqrt{1^2 + 3^2 + 0^2 + (-1)^2 + (-2)^2 + 0^2} = \sqrt{15}.$$

Frage: Wie groß ist $\|A - A\|$?

Für jede Matrix A gilt:

- $\|A\| \geq 0$,
- $\|A\| = 0$ genau dann, wenn $A = O$,
- für jede Zahl α gilt $\|\alpha A\| = |\alpha| \|A\|$.

Für zwei Vektoren

$$\mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{bmatrix} \quad \text{und} \quad \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}$$

ist das *Skalarprodukt* $\langle \mathbf{a}, \mathbf{b} \rangle$ definiert:

$$\langle \mathbf{a}, \mathbf{b} \rangle = \sum_{i=1}^m a_i b_i.$$

Beispiel: Sei

$$\mathbf{a} = \begin{bmatrix} 2 \\ -1 \\ 3 \end{bmatrix} \quad \text{und} \quad \mathbf{b} = \begin{bmatrix} 1 \\ 0 \\ -2 \end{bmatrix},$$

dann

$$\langle \mathbf{a}, \mathbf{b} \rangle = 2 * 1 + (-1) * 0 + 3 * (-2) = -4.$$

Für zwei Vektoren

$$\mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{bmatrix} \quad \text{und} \quad \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}$$

ist das *Skalarprodukt* $\langle \mathbf{a}, \mathbf{b} \rangle$ definiert:

$$\langle \mathbf{a}, \mathbf{b} \rangle = \sum_{i=1}^m a_i b_i.$$

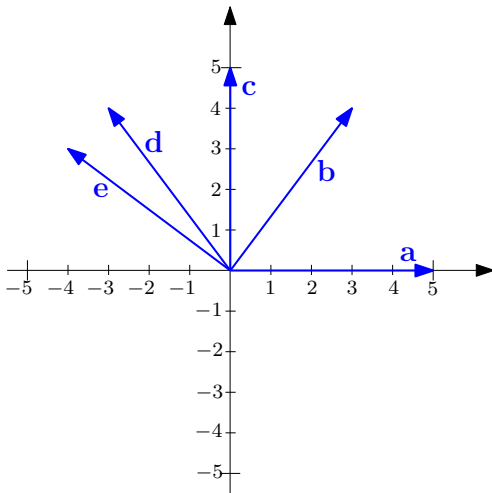
Beispiel: Sei

$$\mathbf{a} = \begin{bmatrix} 2 \\ -1 \\ 3 \end{bmatrix} \quad \text{und} \quad \mathbf{b} = \begin{bmatrix} 1 \\ 0 \\ -2 \end{bmatrix},$$

dann

$$\langle \mathbf{a}, \mathbf{b} \rangle = 2 * 1 + (-1) * 0 + 3 * (-2) = -4.$$

Berechne Skalarprodukte zwischen diesen Vektoren:



Das Skalarprodukt ist ein Maß für den Winkel α zwischen zwei Vektoren \mathbf{a} , \mathbf{b} . Genauer:

$$\cos \alpha = \left\langle \frac{\mathbf{a}}{\|\mathbf{a}\|}, \frac{\mathbf{b}}{\|\mathbf{b}\|} \right\rangle.$$

Folgerung: Das Skalarprodukt zweier aufeinander senkrechter Vektoren ist gleich 0.

Definition

Die Vektoren \mathbf{a} und \mathbf{b} sind *orthogonal*, wenn $\langle \mathbf{a}, \mathbf{b} \rangle = 0$.

Das Skalarprodukt ist ein Maß für den Winkel α zwischen zwei Vektoren \mathbf{a} , \mathbf{b} . Genauer:

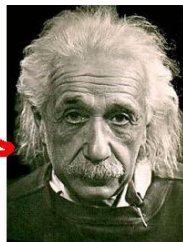
$$\cos \alpha = \left\langle \frac{\mathbf{a}}{\|\mathbf{a}\|}, \frac{\mathbf{b}}{\|\mathbf{b}\|} \right\rangle.$$

Folgerung: Das Skalarprodukt zweier aufeinander senkrechter Vektoren ist gleich 0.

Definition

Die Vektoren \mathbf{a} und \mathbf{b} sind *orthogonal*, wenn $\langle \mathbf{a}, \mathbf{b} \rangle = 0$.

Mir qualmt
schon der Kopf



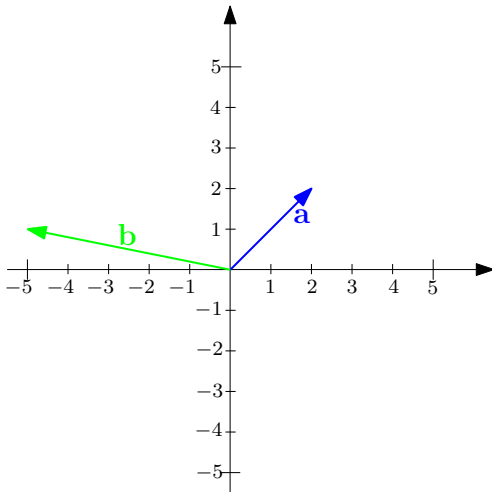
Wir brauchen nur noch den Begriff der *Linearkombination* von Vektoren.

Definition

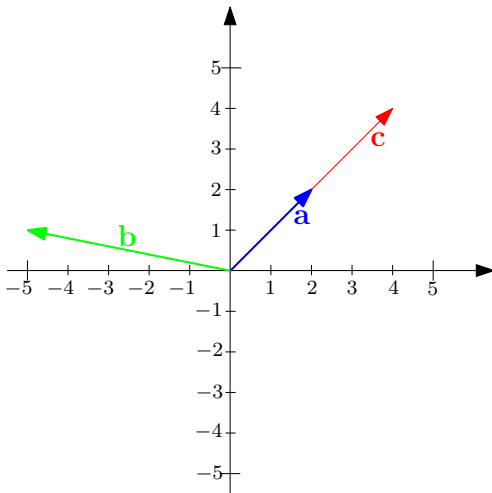
Ein Vektor \mathbf{a} ist eine *Linearkombination* von Vektoren $\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_n$ wenn es Zahlen $\beta_1, \beta_2, \dots, \beta_n$ gibt, so dass

$$\mathbf{a} = \beta_1 \mathbf{b}_1 + \beta_2 \mathbf{b}_2 + \dots + \beta_n \mathbf{b}_n.$$

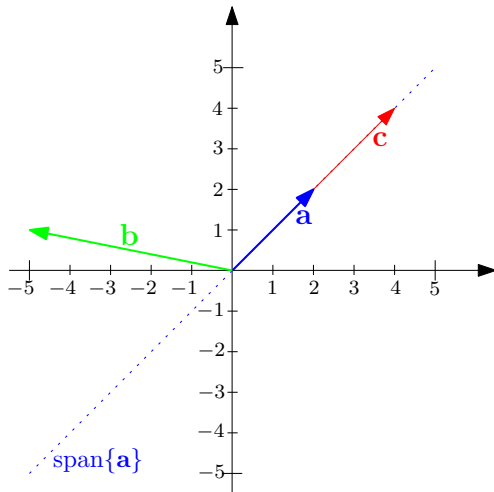
Beispiel: Gegeben seien $\mathbf{a}, \mathbf{b} \in \mathbb{R}^2$.



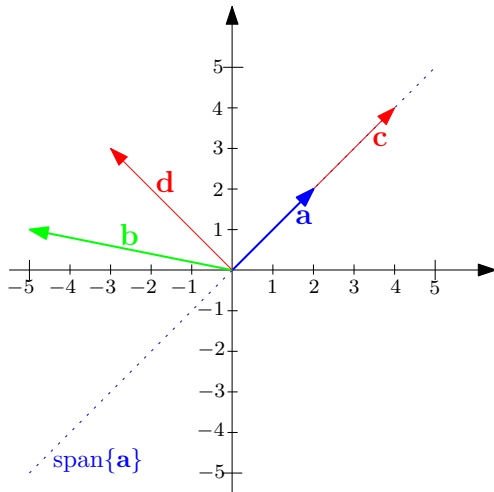
Außerdem sei $\mathbf{c} \in \mathbb{R}^2$ gegeben.



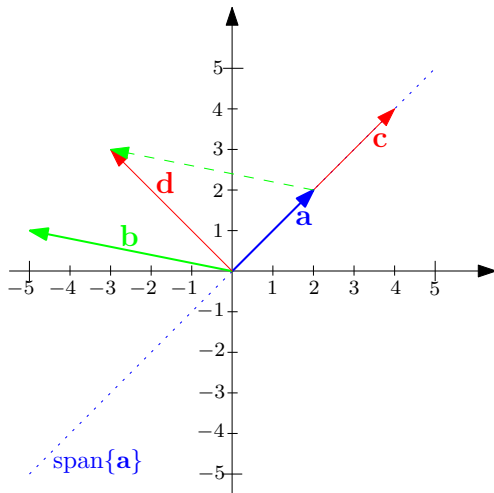
Offenbar ist \mathbf{c} eine Linearkombination von \mathbf{a} , denn $\mathbf{c} = 2\mathbf{a}$.



Allerdings gilt nicht: $\mathbf{d} \in \text{span}\{\mathbf{a}\}$.



Aber es gilt: $\mathbf{d} \in \text{span}\{\mathbf{a}, \mathbf{b}\}$.



Und was hat nun Gesichtserkennung mit Matrizen zu tun?

Gesichtserkennung hat etwas mit Bildern zu tun.

Und Bilder werden im Computer als Matrizen gespeichert.

Und was hat nun Gesichtserkennung mit Matrizen zu tun?

Gesichtserkennung hat etwas mit Bildern zu tun.

Und Bilder werden im Computer als Matrizen gespeichert.

Was ist ein Bild?

Bilder (genauer: Rastergrafiken) bestehen aus einer rasterförmigen Anordnung von Bildpunkten (Pixel), denen jeweils eine Farbe (bzw. Graustufe) zugeordnet ist. Dabei gibt es verschiedene Farbmodelle, um die Farben zu repräsentieren.

Beispiel (RGB-Farbmodell)

Jede Farbe wird aus den Farben Rot, Grün und Blau gemischt. Darstellung als Tripel (R, G, B), wobei jede Position Werte zwischen 0 und 1 annehmen kann.

■ (0, 0, 0) ■ (1, 0, 0) ■ (0.5, 0, 0) ■ (1, 1, 0) ■ (1, 0, 1)
■ (0, 0.3, 0.7) ■ (0, 0.5, 0.5) ■ (0, 1, 1) ■ (0.6, 0.6, 0.6) ■

Beispiel (Graustufen-Modell)

Jeder Grauton wird durch eine Zahl zwischen 0 (schwarz) und 1 (weiß) repräsentiert.

■ (0) ■ (0.2) ■ (0.4) ■ (0.6) ■ (0.8) ■

Was ist ein Bild?

Bilder (genauer: Rastergrafiken) bestehen aus einer rasterförmigen Anordnung von Bildpunkten (Pixel), denen jeweils eine Farbe (bzw. Graustufe) zugeordnet ist. Dabei gibt es verschiedene Farbmodelle, um die Farben zu repräsentieren.

Beispiel (RGB-Farbmodell)

*Jede Farbe wird aus den Farben **Rot**, **Grün** und **Blau** gemischt. Darstellung als Tripel (**R**, **G**, **B**), wobei jede Position Werte zwischen 0 und 1 annehmen kann.*

■ (0, 0, 0) ■ (1, 0, 0) ■ (0.5, 0, 0) ■ (1, 1, 0) ■ (1, 0, 1)
■ (0, 0.3, 0.7) ■ (0, 0.5, 0.5) ■ (0, 1, 1) ■ (0.6, 0.6, 0.6) ■ (1, 1, 1)

Beispiel (Graustufen-Modell)

Jeder Grauton wird durch eine Zahl zwischen 0 (schwarz) und 1 (weiß) repräsentiert.

■ (0) ■ (0.2) ■ (0.4) ■ (0.6) ■ (0.8) ■

Was ist ein Bild?

Bilder (genauer: Rastergrafiken) bestehen aus einer rasterförmigen Anordnung von Bildpunkten (Pixel), denen jeweils eine Farbe (bzw. Graustufe) zugeordnet ist. Dabei gibt es verschiedene Farbmodelle, um die Farben zu repräsentieren.

Beispiel (RGB-Farbmodell)

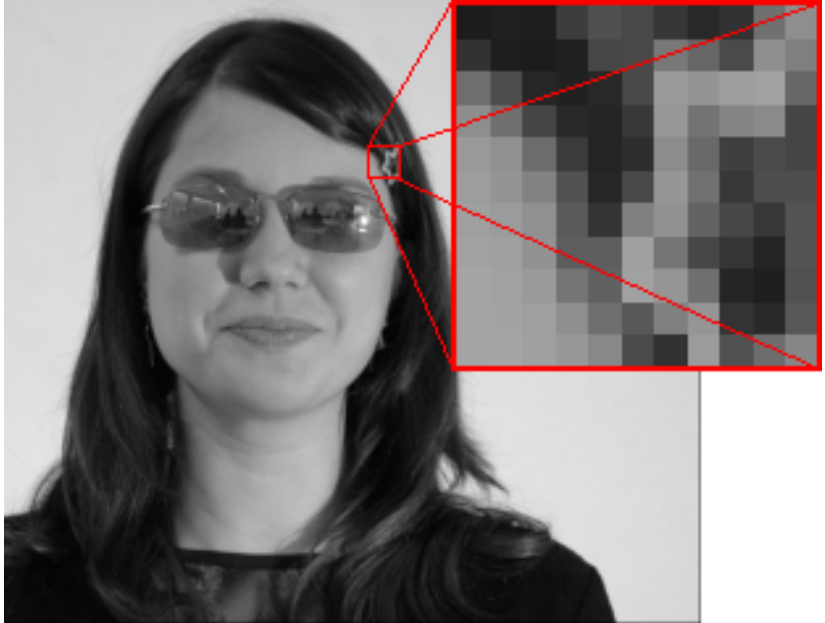
Jede Farbe wird aus den Farben *Rot*, *Grün* und *Blau* gemischt. Darstellung als Tripel (*R*, *G*, *B*), wobei jede Position Werte zwischen 0 und 1 annehmen kann.

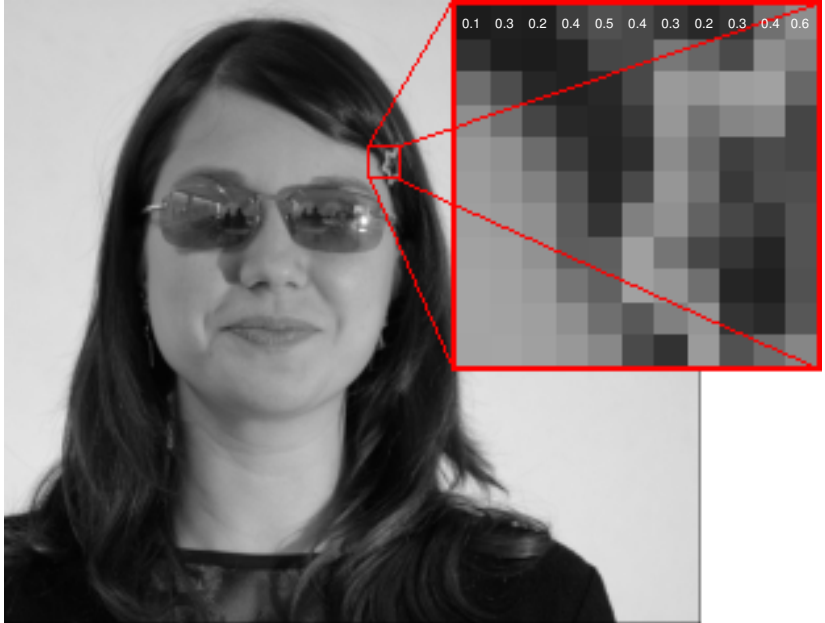
■ (0, 0, 0) ■ (1, 0, 0) ■ (0.5, 0, 0) ■ (1, 1, 0) ■ (1, 0, 1)
■ (0, 0.3, 0.7) ■ (0, 0.5, 0.5) ■ (0, 1, 1) ■ (0.6, 0.6, 0.6) ■ (1, 1, 1)

Beispiel (Graustufen-Modell)

Jeder Grauton wird durch eine Zahl zwischen 0 (schwarz) und 1 (weiß) repräsentiert.

■ (0) ■ (0.2) ■ (0.4) ■ (0.6) ■ (0.8) ■ (1)





Also: Graustufenbilder sind Matrizen $B \in \mathbb{R}^{m \times n}$ mit Einträgen $0 \leq B(i, j) \leq 1$.

Gesichtsdetektierung

In der *Gesichtsdetektierung* ist man daran interessiert, auf einem Bild bestimmte Merkmale (features) eines Gesichtes, wie z.B. Nase oder Augen, zu lokalisieren.

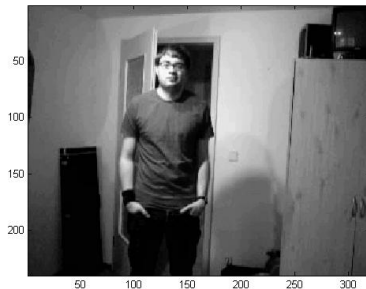
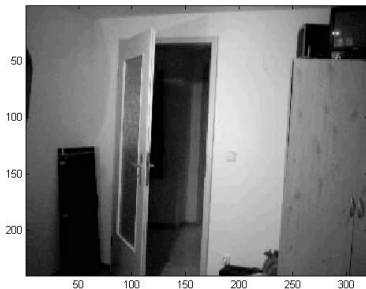
Rechnen mit Bildern (=Matrizen), Beispiel 1

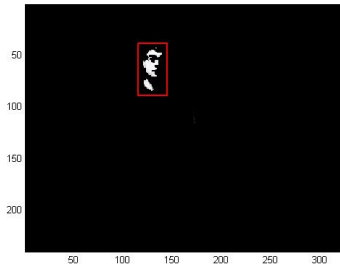
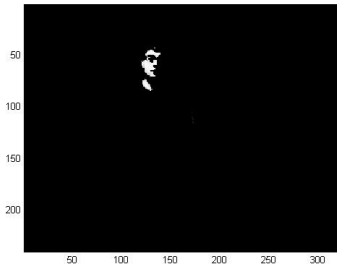
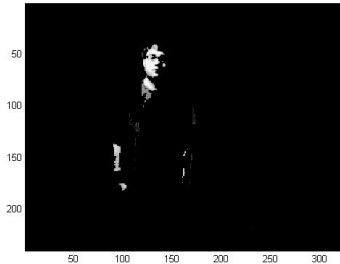
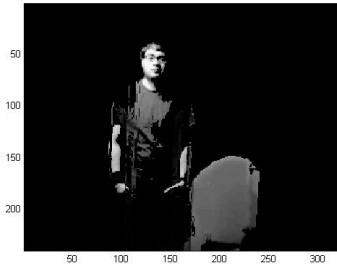
Entferne das Standbild S *pixelweise* aus einem Bild B :
 Berechne *elementweise*

$$D_\alpha := \text{tresh}_\alpha(\text{abs}(B - S)) .* B$$

mit

$$\text{tresh}_\alpha(x) = \begin{cases} 0, & x < \alpha; \\ 1, & x \geq \alpha. \end{cases}$$



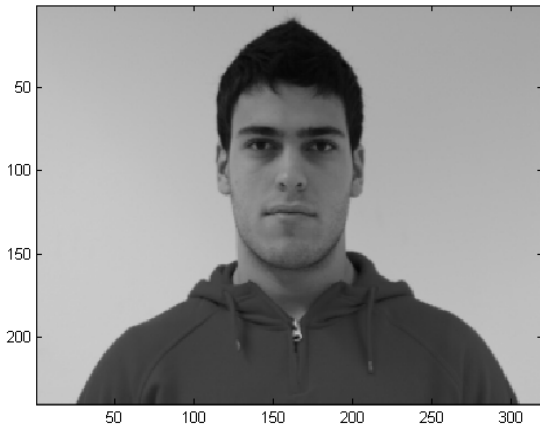


Rechnen mit Bildern (=Matrizen), Beispiel 2

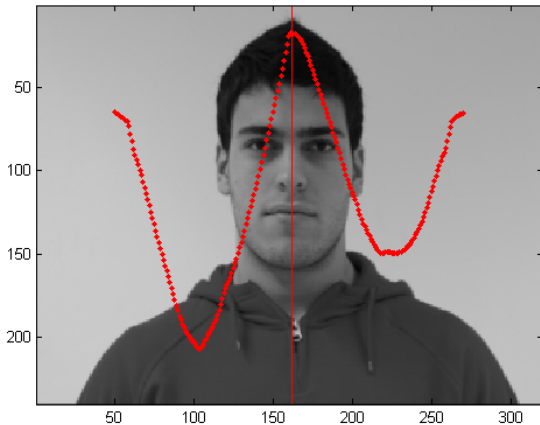
Vergleiche zwei Bilder A und B mit Hilfe einer Matrixnorm $\|B - A\|$,
z.B. *Frobeniusnorm*

$$\|M\| = \sqrt{\sum_{j=1}^n \sum_{i=1}^m M(i, j)^2}.$$

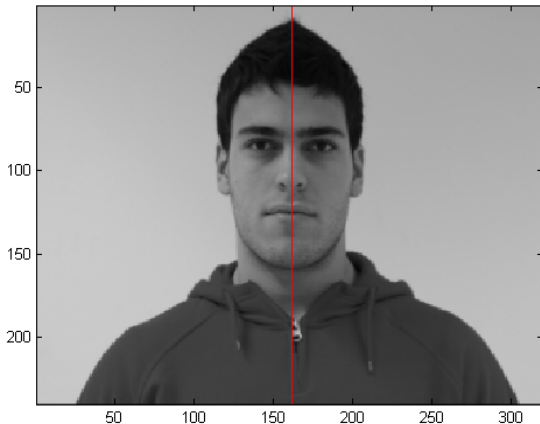
Untersuche ein Bild *pixelweise* auf Symmetrien und Ähnlichkeiten.



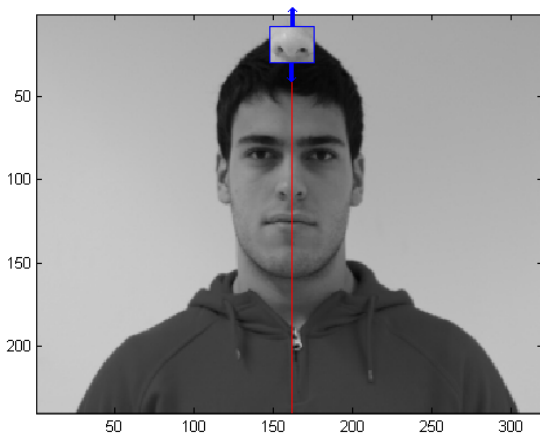
Untersuche ein Bild *pixelweise* auf Symmetrien und Ähnlichkeiten.



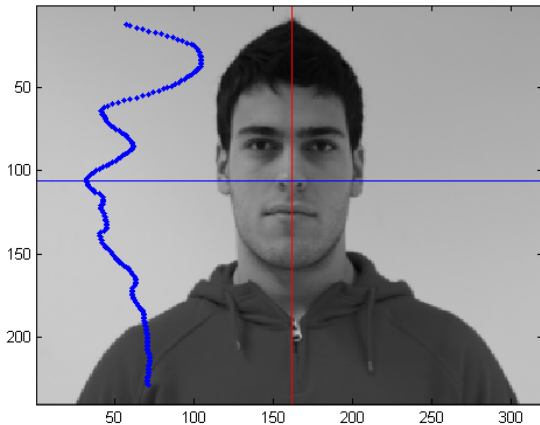
Untersuche ein Bild *pixelweise* auf Symmetrien und Ähnlichkeiten.



Untersuche ein Bild *pixelweise* auf Symmetrien und Ähnlichkeiten.

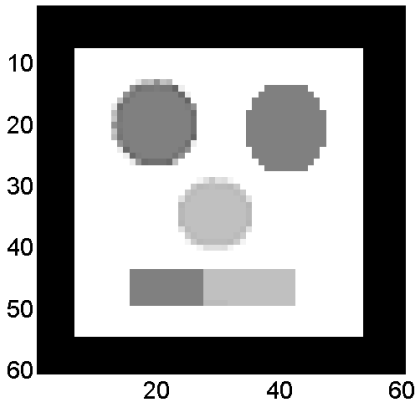


Untersuche ein Bild *pixelweise* auf Symmetrien und Ähnlichkeiten.

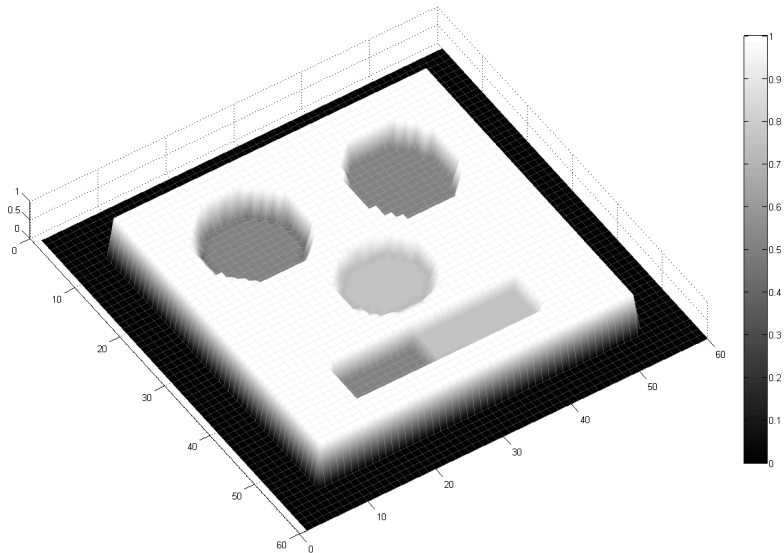


Rechnen mit Bildern (=Matrizen), Beispiel 3

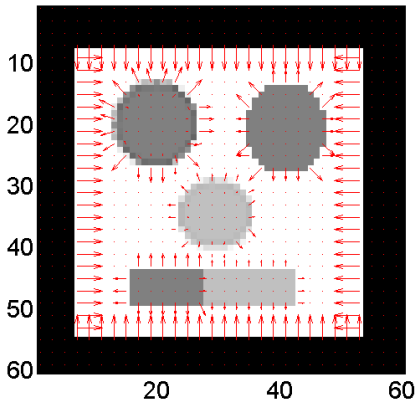
Untersuche ein Bild B auf Kanten:



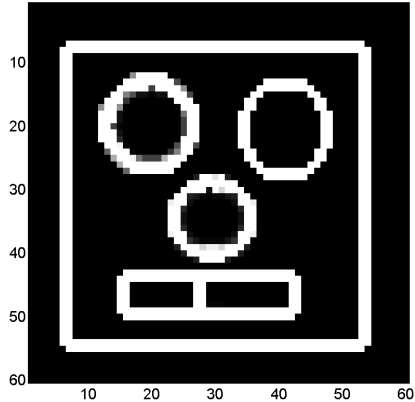
Interpretiere B als differenzierbare Funktion $b(x, y)$ auf $[1, m] \times [1, n]$:



Approximiere Gradienten $\nabla b(x, y)$ mit finiten Differenzen:



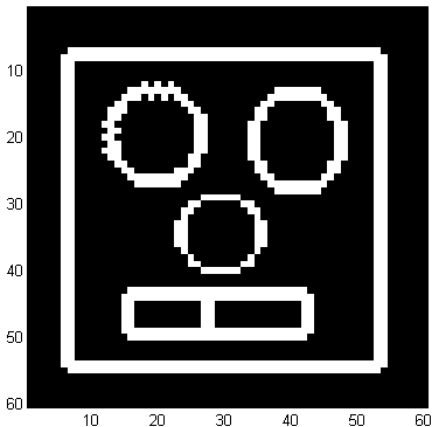
Gradientenbild $\|\nabla b(x, y)\|$:



Exkurs: Hough-Transformation

- robustes globales Verfahren zur Erkennung von parametrisierbaren geometrischen Figuren in Schwarz/Weiß-Bildern,
- Brute-Force-Ansatz, effiziente Implementierungen sind aber bekannt (Fast HT),
- 1962 von Paul V.C. Hough patentiert: "Method and Means for Recognizing Complex Patterns".

Beispiel: Finde weiße Kreise mit Radius r in Bild B :



Algorithmus:

Initialisiere Akkumlatormatrix $A_r = \text{zeros}(m, n)$.

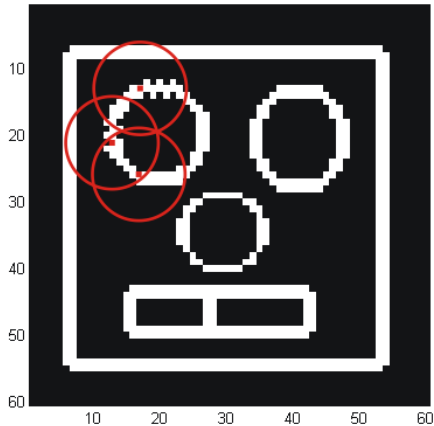
Für jedes *weiße* Pixel (i, j) in B ,

für jedes Pixel (k, ℓ) mit $|(k - i)^2 + (\ell - j)^2 - r^2| \leq 1$,

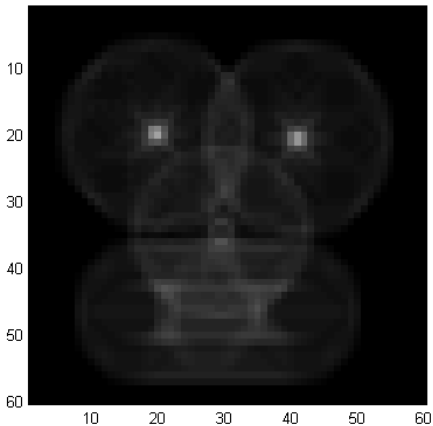
inkrementiere $A_r(k, \ell) = A_r(k, \ell) + 1$.

Kreismittelpunkte sind "große" Einträge in A_r .

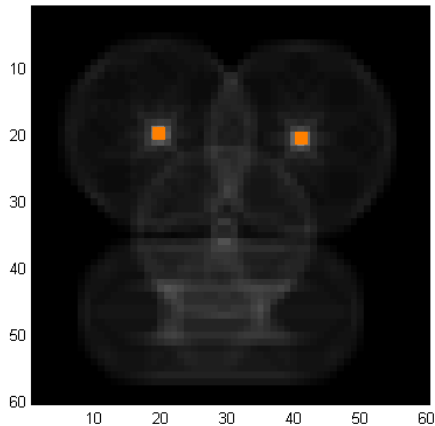
Prinzip der HT:



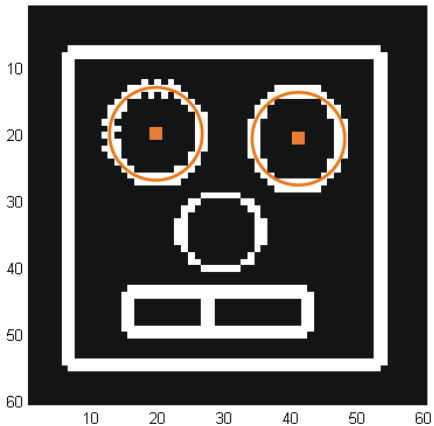
Einträge von A_7 :



Einträge von A_7 :

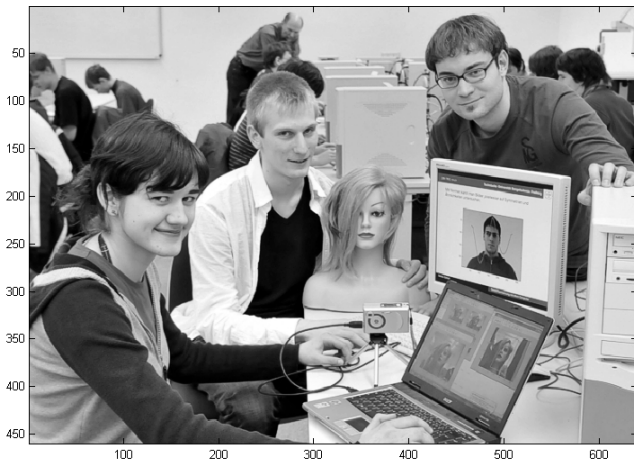


Erkennungsergebnis:



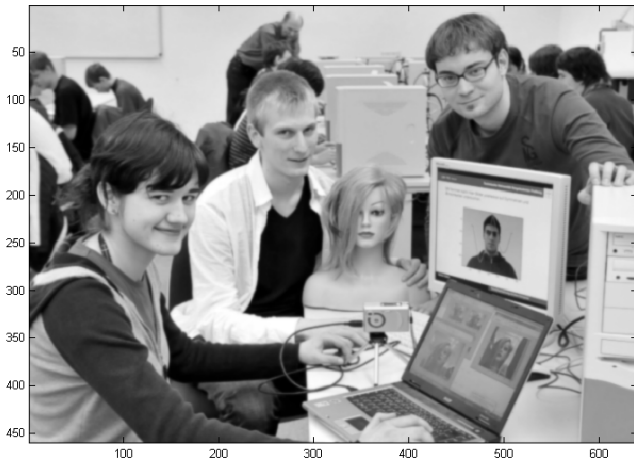
Rechnen mit Bildern (=Matrizen), Beispiel 5

Löse $\partial_t b(x, y, t) = \Delta b(x, y, t)$ mit $b(x, y, 0) = b(x, y)$:



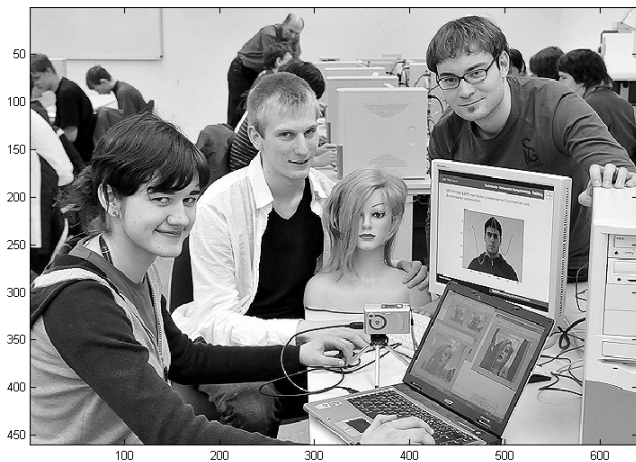
Rechnen mit Bildern (=Matrizen), Beispiel 5

Löse $\partial_t b(x, y, t) = \Delta b(x, y, t)$ mit $b(x, y, 0) = b(x, y)$:



Rechnen mit Bildern (=Matrizen), Beispiel 5

Löse $\partial_t b(x, y, t) = \Delta b(x, y, t)$ mit $b(x, y, 0) = b(x, y)$:



In der Praxis: Kombination mehrerer Methoden

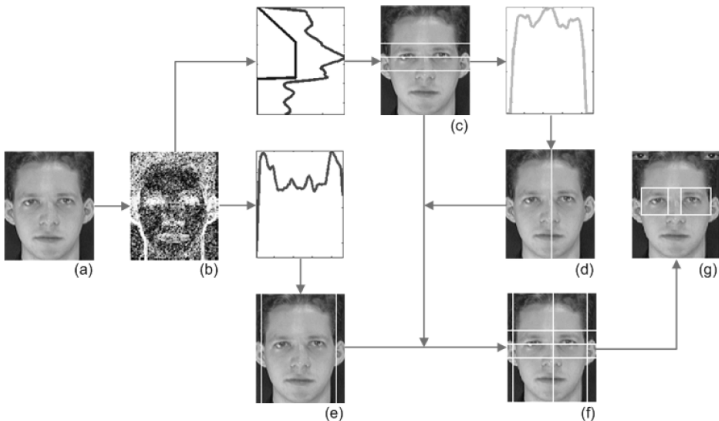


Fig. 2. Detection of eyes' regions

K. Peng, L. Chen, S. Ruan, G. Kukharev: A Robust Algorithm for Eye Detection on Gray Intensity Face without Spectacles, JCS&T, 2005.



Fig. 3. Template matching



Fig. 4. Examples for faces without spectacles



Fig. 5. Examples for faces with spectacles

Emotionserkennung

"...the young and the old of widely different races, both with man and animals, express the same state of mind by the same movements."
(Charles Darwin)

Facial Expression Classification System (FACS):

01 Heben der Augenbraue innen

04 Stirnrunzeln, "Sorgenfalte"

05 Heben des oberen Lides wie beim erstaunten Blick

06 Heben der Wange wie beim drohenden Blick

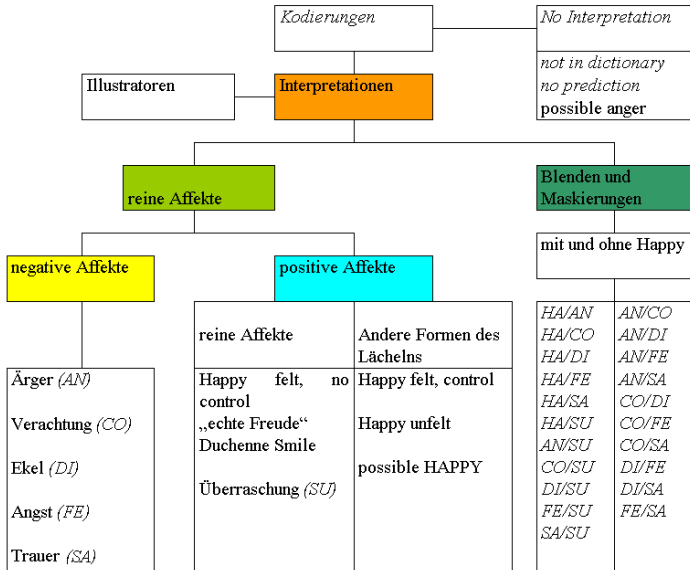
07 Zusammenpressen der Lider wie beim Biss in eine Zitrone

42 Lider zum Schlitz verengen

Stärke des Ausdrucks: A (sehr schwach) bis E (sehr stark)

Beispiel: 06D und 42C → Wut.

EmFACS-Klassifikation



Bis jetzt...

- Anwendungen
- Bilder als Matrizen
- Gesichtsdetektierung
- Einfache *feature-extraction*
 - Normen von Teilbildern (template matching)
 - Gradientenbild
 - Hough-Transformation
 - Weichzeichnen / Schärfen
- Bildverstehen: EmFACS

Ab jetzt: Gesichtserkennung.

Die Eigenface-Methode

- Verfahren zur Gesichtserkennung,
- basiert auf Hauptkomponentenanalyse der Bilddaten (PCA),
- 1987 L. Sirovich and M. Kirby: "Low-Dimensional procedure for the characterization of human faces",
- 1991 Begriff "Eigenface" erstmals von M.A. Turk and A.P. Pentland gebraucht: "Face recognition using eigenfaces".

Gegeben sind $p = 20$ Matrizen G_1, G_2, \dots, G_p (Gesichter):



Gegeben ist außerdem eine Matrix T , das *Testgesicht*. Beispiel:



Welches Gesicht G_j ist von derselben Person wie Testgesicht T ?

Damit Daten verglichen werden dürfen, müssen sie zentriert sein.

Beispiel (Noten der letzten Mathe-Klausur)

Name	Punkte	Punkte - $\bar{0}$
Achim	10	3.2
Bernd	7	0.2
Claudi	13	6.2
Doris	1	-5.8
Erwin	3	-3.8
	$\bar{0}$ 6.8	

Ist Bernd ein guter Mathematiker?

Bernd ist immerhin besser als der Durchschnitt!

Der reine Punktwert ist für die Klassifizierung von Schülerleistungen nicht aussagekräftig, dafür aber die Abweichung vom Mittel.

Damit Daten verglichen werden dürfen, müssen sie zentriert sein.

Beispiel (Noten der letzten Mathe-Klausur)

Name	Punkte	Punkte - $\bar{Ø}$
Achim	10	3.2
Bernd	7	0.2
Claudi	13	6.2
Doris	1	-5.8
Erwin	3	-3.8
	$\bar{Ø}$ 6.8	

*Ist Bernd ein guter
Mathematiker?*

*Bernd ist immerhin besser
als der Durchschnitt!*

*Der reine Punktwert ist für die
Klassifizierung von Schülerleistungen
nicht aussagekräftig, dafür aber die
Abweichung vom Mittel.*

Damit Daten verglichen werden dürfen, müssen sie zentriert sein.

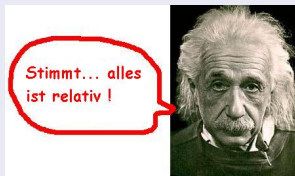
Beispiel (Noten der letzten Mathe-Klausur)

Name	Punkte	Punkte - $\bar{\emptyset}$
Achim	10	3.2
Bernd	7	0.2
Claudi	13	6.2
Doris	1	-5.8
Erwin	3	-3.8
	$\bar{\emptyset}$ 6.8	

Ist Bernd ein guter Mathematiker?

Bernd ist immerhin besser als der Durchschnitt!

Der reine Punktwert ist für die Klassifizierung von Schülerleistungen nicht aussagekräftig, dafür aber die Abweichung vom Mittel.

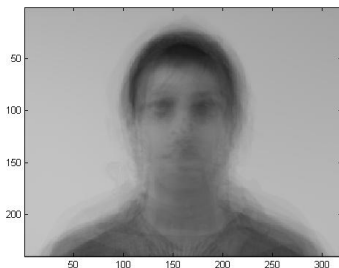


Frage: Sind Bilddaten relevant, die auf allen Bildern gleich sind (zum Beispiel der graue Hintergrund)?

Antwort: Sicher nein. (Der graue Hintergrund trägt für uns keine Information über das Gesicht auf dem Bild.)

Wir berechnen das *Durchschnittsgesicht*

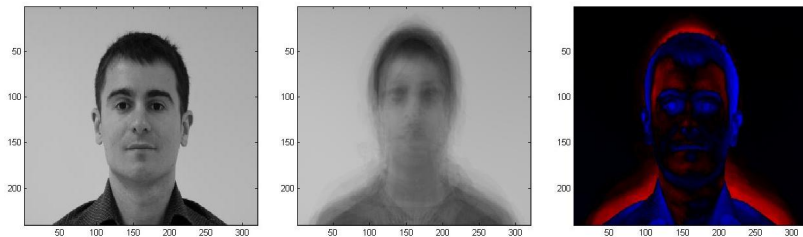
$$\tilde{G} := \frac{1}{p} \sum_{j=1}^p G_j.$$



Für jedes Gesicht G_j berechnen wir das *Differenzgesicht*

$$D_j := G_j - \tilde{G},$$

d.h., D_j stellt die Abweichung des Gesichts G_j vom Mittel dar.



(Rechtes Bild: positive Einträge sind rot, negative blau und Nulleinträge schwarz gekennzeichnet.)

Analog zentrieren wir auch das Testgesicht:

$$T_0 = T - \tilde{G}.$$

Die Gesichtsdifferenzen D_1, D_2, \dots, D_p sowie das zentrierte Testgesicht T_0 (alle haben Größe $m \times n$) werden als Vektoren $\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_p$ und \mathbf{t}_0 der Länge $N = mn$ aufgefasst, indem man die Spalten der Matrizen untereinander anordnet.

$$\begin{bmatrix} .1 & .4 \\ .2 & .5 \\ .3 & .6 \end{bmatrix} \Rightarrow \begin{bmatrix} .1 \\ .2 \\ .3 \\ .4 \\ .5 \\ .6 \end{bmatrix}$$

Bestimmung der Eigenfaces

Für $k = 1, 2, \dots, N$ berechne *orthonormale* Vektoren $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N \in \mathbb{R}^{mn}$ (genannt: *Eigenfaces*), so dass

$$\lambda_k = \frac{1}{p-1} \sum_{j=1}^p \langle \mathbf{v}_k, \mathbf{d}_j \rangle^2 \rightarrow \max.$$

Wir fordern außerdem: $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_N \geq 0$.

Interpretation: Die obige Summe wird groß, wenn die Richtung der Eigenfaces \mathbf{v}_k "möglichst gut" mit allen Bildern \mathbf{d}_j übereinstimmt.

Bestimmung der Eigenfaces

Für $k = 1, 2, \dots, N$ berechne *orthonormale* Vektoren $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N \in \mathbb{R}^{mn}$ (genannt: *Eigenfaces*), so dass

$$\lambda_k = \frac{1}{p-1} \sum_{j=1}^p \langle \mathbf{v}_k, \mathbf{d}_j \rangle^2 \rightarrow \max.$$

Wir fordern außerdem: $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_N \geq 0$.

Interpretation: Die obige Summe wird groß, wenn die Richtung der Eigenfaces \mathbf{v}_k "möglichst gut" mit allen Bildern \mathbf{d}_j übereinstimmt.

Jedes Bild \mathbf{d}_j lässt sich eindeutig als Linearkombination der Eigenfaces $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N$ darstellen, d.h. es gibt Zahlen $\beta_1^{(j)}, \beta_2^{(j)}, \dots, \beta_N^{(j)}$, so dass

$$\mathbf{d}_j = \beta_1^{(j)} \mathbf{v}_1 + \beta_2^{(j)} \mathbf{v}_2 + \dots + \beta_N^{(j)} \mathbf{v}_N.$$

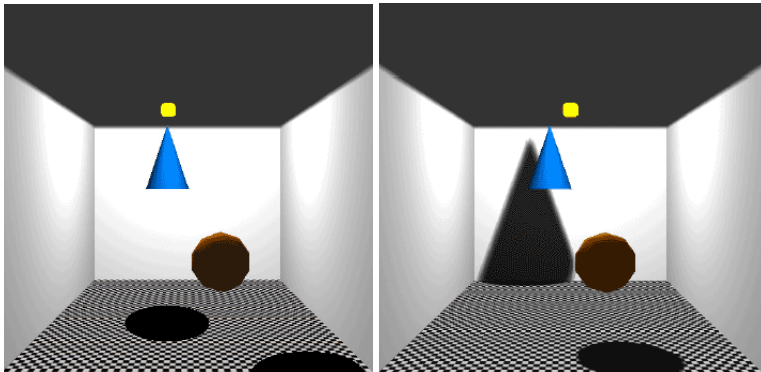
$\beta_k^{(j)}$ gibt an, wieviel Anteil das Eigenface \mathbf{v}_k am Bild \mathbf{d}_j hat.

Betrachtet man die Bilder $\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_N$ als (0-zentrierte) Punktwolke im \mathbb{R}^{mn} , dann ist \mathbf{v}_k die Richtung durch diese Punktwolke, entlang derer die *Varianz* λ_k größtmöglich ist. Die Varianz ist ein Maß für den Informationsgehalt einer Zufallsvariablen (jedes Bild ist nämlich die Realisierung einer mn -dimensionalen Zufallsvariablen).

Je mehr Information man über ein Bild besitzt, desto besser kann man es von anderen Bildern unterscheiden. Die Komponente $\beta_1^{(j)}$ speichert also (in einer Zahl!) die größtmögliche Information über das Bild \mathbf{d}_j .

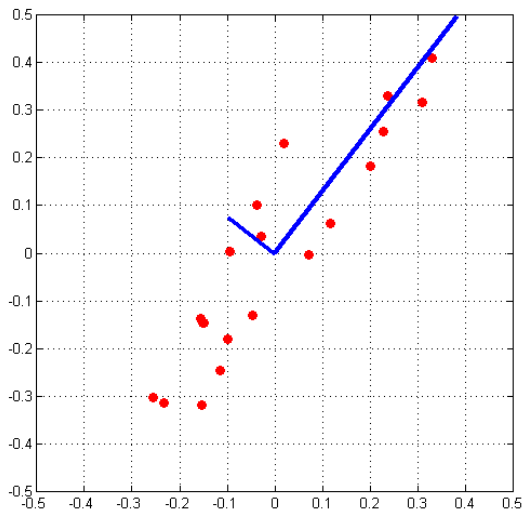
Anschaulich ist das in etwa so, als würde man versuchen, aus dem Schatten (die Komponenten $\beta_k^{(j)}$) verschieden platzierter Lichtquellen (die Vektoren \mathbf{v}_k) die Form eines Gegenstandes (das Bild \mathbf{d}_j) zu erraten.

Unsere Lichtquellen (die \mathbf{v}_k) sind so gut im Raum positioniert, dass die Schatten der \mathbf{d}_j (die Projektionen der \mathbf{d}_j auf $\text{span}\{\mathbf{v}_k\}$) größtmöglichen Informationsgehalt haben.

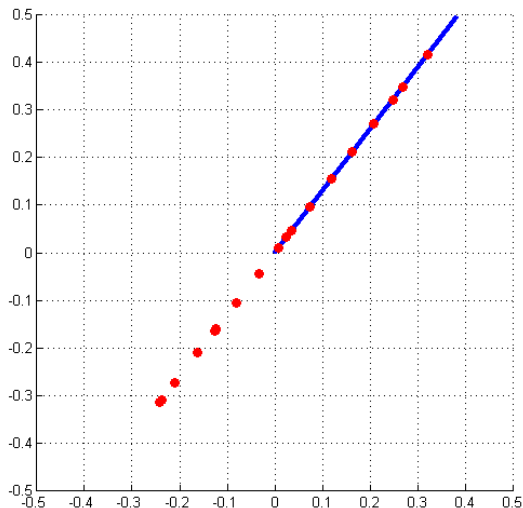


Der Schatten im rechten Bild trägt mehr Information über die unterschiedliche Geometrie der beiden Gegenstände als der Schatten im linken Bild.

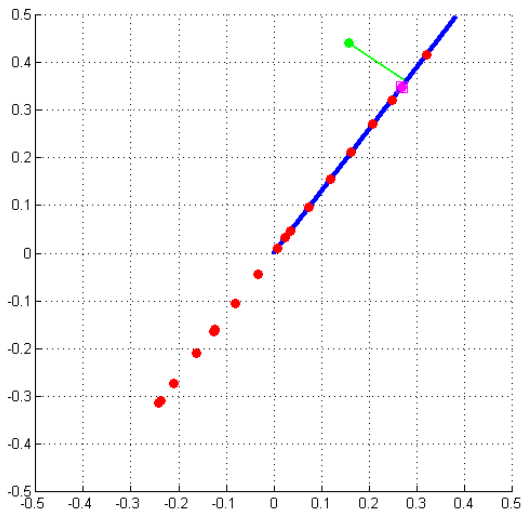
Beispiel: 20 Bilder der Größe 2×1 und ihre Eigenfaces:



Beispiel: Projektion der 20 Bilder auf ihre erste Hauptkomponente:



Beispiel: Projektion der 20 Bilder und des Testgesichtes:



Beispiel: Wir betrachten 3 Graustufen-Bilder der Größe 2×1 :

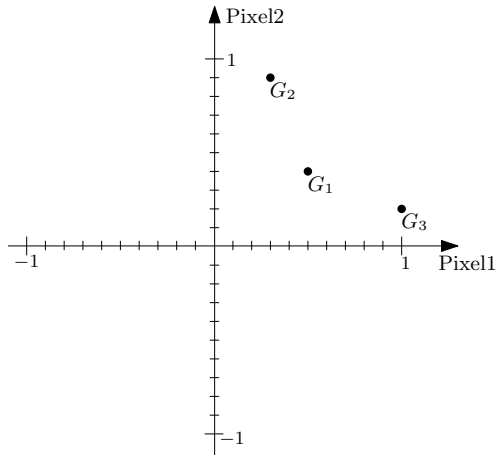
$$G_1 = \begin{bmatrix} 0.5 \\ 0.4 \end{bmatrix}, G_2 = \begin{bmatrix} 0.3 \\ 0.9 \end{bmatrix}, G_3 = \begin{bmatrix} 1.0 \\ 0.2 \end{bmatrix}.$$

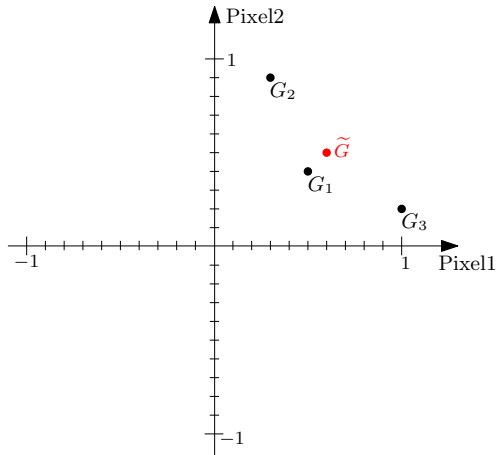
Das Durchschnittsgesicht ist

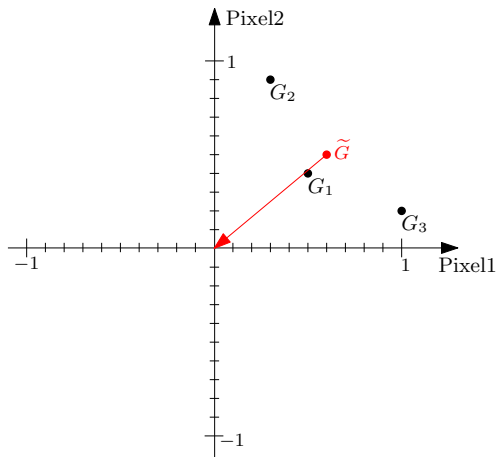
$$\tilde{G} := \frac{1}{3}(G_1 + G_2 + G_3) = \begin{pmatrix} 0.6 \\ 0.5 \end{pmatrix}$$

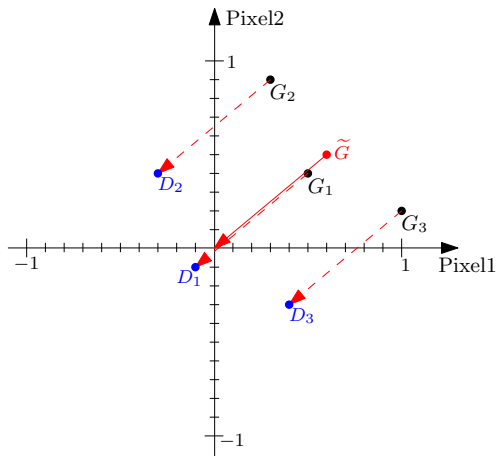
und die Differenzgesichter $D_j = G_j - \tilde{G}$ sind

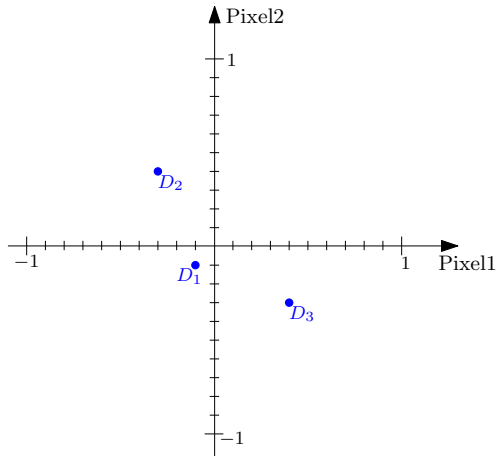
$$D_1 = \mathbf{d}_1 = \begin{bmatrix} -0.1 \\ -0.1 \end{bmatrix}, D_2 = \mathbf{d}_2 = \begin{bmatrix} -0.3 \\ 0.4 \end{bmatrix}, D_3 = \mathbf{d}_3 = \begin{bmatrix} 0.4 \\ -0.3 \end{bmatrix}.$$

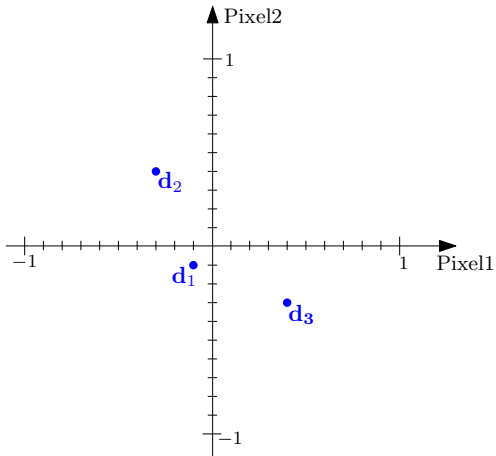


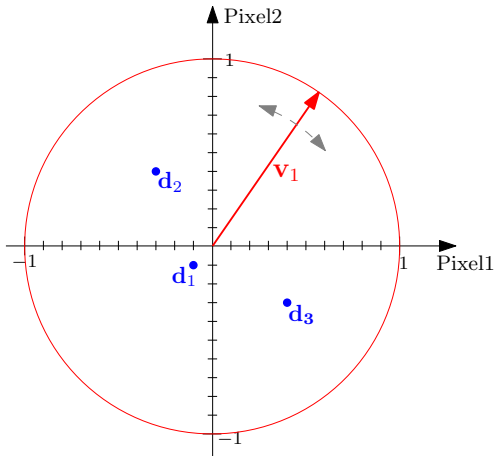


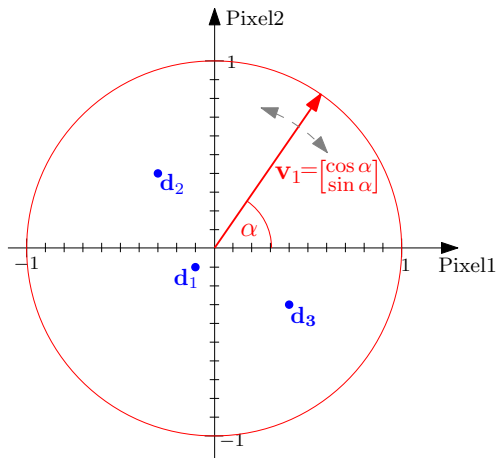


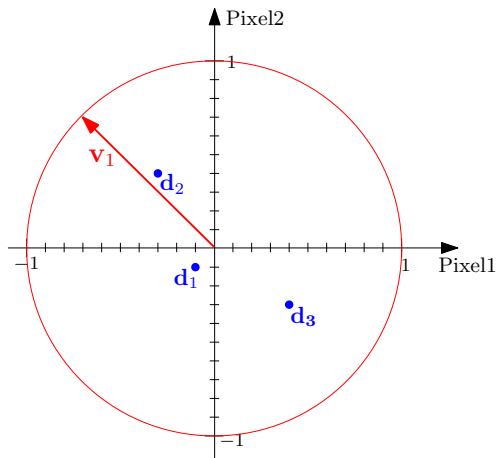


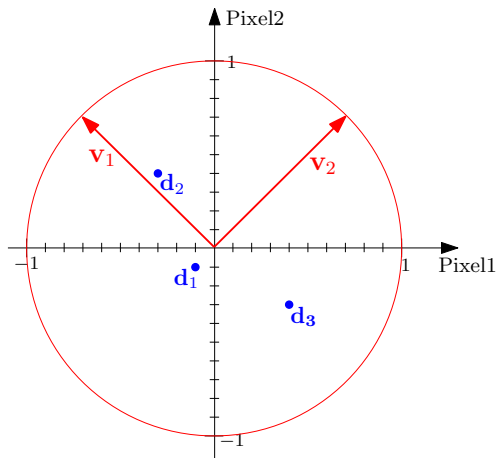










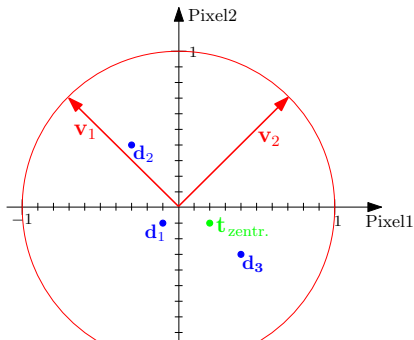


Die beiden Eigenfaces sind

$$\mathbf{v}_1 = \sqrt{\frac{1}{2}} \begin{bmatrix} -1 \\ 1 \end{bmatrix} \quad \text{und} \quad \mathbf{v}_2 = \sqrt{\frac{1}{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix}.$$

Es sei nun ein zentriertes Testgesicht $\mathbf{t}_{\text{zentr.}}$ gegeben, das wir auf Ähnlichkeit zu \mathbf{d}_1 , \mathbf{d}_2 und \mathbf{d}_3 prüfen wollen:

$$\mathbf{t}_{\text{zentr.}} = \begin{bmatrix} 0.2 \\ -0.1 \end{bmatrix}.$$



Dazu stellen wir \mathbf{d}_1 , \mathbf{d}_2 , \mathbf{d}_3 sowie das zentrierte Testgesicht $\mathbf{t}_{\text{zentr.}}$ als Linearkombination von \mathbf{v}_1 und \mathbf{v}_2 dar (gerundet):

$$\begin{aligned}\mathbf{d}_1 &= 0\mathbf{v}_1 + (-0.141)\mathbf{v}_2 \\ \mathbf{d}_2 &= 0.495\mathbf{v}_1 + 0.071\mathbf{v}_2 \\ \mathbf{d}_3 &= -0.495\mathbf{v}_1 + 0.071\mathbf{v}_2 \\ \mathbf{t}_{\text{zentr.}} &= -0.212\mathbf{v}_1 + 0.071\mathbf{v}_2\end{aligned}$$

Nun vergleicht man die **Koeffizienten** der Eigenfaces in den Differenzgesichtern mit den **Koeffizienten** der Eigenfaces im zentrierten Testgesicht, z.B. mittels Differenzenquadrate:

$$\text{Vergleich } \mathbf{d}_1 - \mathbf{t}_{\text{zentr.}}: (0 - -0.212)^2 + (-0.141 - 0.071)^2 = 0.090$$

$$\text{Vergleich } \mathbf{d}_2 - \mathbf{t}_{\text{zentr.}}: (0.495 - -0.212)^2 + (0.071 - 0.071)^2 = 0.500$$

$$\text{Vergleich } \mathbf{d}_3 - \mathbf{t}_{\text{zentr.}}: (-0.495 - -0.212)^2 + (0.071 - 0.071)^2 = 0.080$$

Dazu stellen wir \mathbf{d}_1 , \mathbf{d}_2 , \mathbf{d}_3 sowie das zentrierte Testgesicht $\mathbf{t}_{\text{zentr.}}$ als Linearkombination von \mathbf{v}_1 und \mathbf{v}_2 dar (gerundet):

$$\begin{aligned}\mathbf{d}_1 &= 0\mathbf{v}_1 + (-0.141)\mathbf{v}_2 \\ \mathbf{d}_2 &= 0.495\mathbf{v}_1 + 0.071\mathbf{v}_2 \\ \mathbf{d}_3 &= -0.495\mathbf{v}_1 + 0.071\mathbf{v}_2 \\ \mathbf{t}_{\text{zentr.}} &= -0.212\mathbf{v}_1 + 0.071\mathbf{v}_2\end{aligned}$$

Nun vergleicht man die **Koeffizienten** der Eigenfaces in den Differenzgesichtern mit den **Koeffizienten** der Eigenfaces im zentrierten Testgesicht, z.B. mittels Differenzenquadrate:

$$\text{Vergleich } \mathbf{d}_1 - \mathbf{t}_{\text{zentr.}}: (0 - -0.212)^2 + (-0.141 - 0.071)^2 = 0.090$$

$$\text{Vergleich } \mathbf{d}_2 - \mathbf{t}_{\text{zentr.}}: (0.495 - -0.212)^2 + (0.071 - 0.071)^2 = 0.500$$

$$\text{Vergleich } \mathbf{d}_3 - \mathbf{t}_{\text{zentr.}}: (-0.495 - -0.212)^2 + (0.071 - 0.071)^2 = 0.080$$

Dazu stellen wir \mathbf{d}_1 , \mathbf{d}_2 , \mathbf{d}_3 sowie das zentrierte Testgesicht $\mathbf{t}_{\text{zentr.}}$ als Linearkombination von \mathbf{v}_1 und \mathbf{v}_2 dar (gerundet):

$$\begin{aligned}\mathbf{d}_1 &= 0\mathbf{v}_1 + (-0.141)\mathbf{v}_2 \\ \mathbf{d}_2 &= 0.495\mathbf{v}_1 + 0.071\mathbf{v}_2 \\ \mathbf{d}_3 &= -0.495\mathbf{v}_1 + 0.071\mathbf{v}_2 \\ \mathbf{t}_{\text{zentr.}} &= -0.212\mathbf{v}_1 + 0.071\mathbf{v}_2\end{aligned}$$

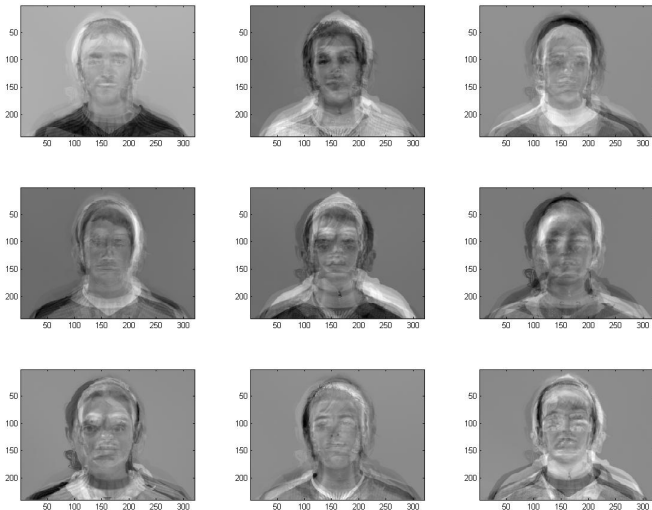
Nun vergleicht man die **Koeffizienten** der Eigenfaces in den Differenzgesichtern mit den **Koeffizienten** der Eigenfaces im zentrierten Testgesicht, z.B. mittels Differenzenquadrate:



$$\text{Vergleich } \mathbf{d}_1 - \mathbf{t}_{\text{zentr.}}: (0 - -0.212)^2 + (-0.141 - 0.071)^2 = 0.090$$










$$\text{Vergleich } \mathbf{d}_2 - \mathbf{t}_{\text{zentr.}}: (0.495 - -0.212)^2 + (0.071 - 0.071)^2 = 0.500$$

$$\text{Vergleich } \mathbf{d}_3 - \mathbf{t}_{\text{zentr.}}: (-0.495 - -0.212)^2 + (0.071 - 0.071)^2 = \underline{0.080}$$

Und so sehen die ersten 9 Eigenfaces für unsere 20 Gesichter aus:



					$\pi(d_j) - \pi(t_0)^2$
	93	18	35	50	101301
	-112	-35	14	0	19116
	23	-29	-36	-37	67850
	21	-17	10	49	58650
	-181	-142	55	35	

					$\pi(d_j) - \pi(t_0)^2$
	93	18	35	50	101301
	-112	-35	14	0	<u>19116</u>
	23	-29	-36	-37	67850
	21	-17	10	49	58650
	-181	-142	55	35	

Danke für die Aufmerksamkeit!

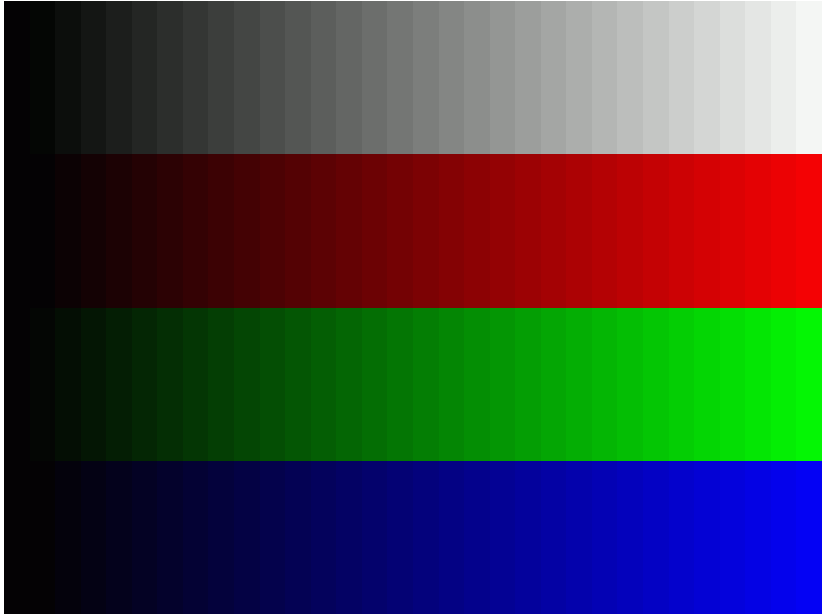
Sei $M := [\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_N]$, dann sind die \mathbf{v}_k Eigenvektoren der Kovarianzmatrix (symmetrisch, positiv semidefinit)

$$C := \frac{1}{N} MM^T.$$

C ist symmetrisch und positiv semidefinit. Mit $V := [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N]$ und $\Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_N)$ ist die Spektralzerlegung von C

$$C = V\Lambda V^T.$$

Das hier beschriebene Verfahren ist im Grunde eine *Hauptachsentransformation* auf Bilddaten. Dieses Verfahren findet v.a. in der Statistik zahlreiche weitere Anwendungen.



Weiterführende Literatur I



M.A. Turk, A.P. Pentland.
Face recognition using eigenfaces.
MIT, Cambridge, 1991.



Audio Visual Technologies Group.
UPC Face Database.
University of Catalonia, Spain.