

# Gesichtserkennung II

## *Eigenfaces*

Gerhard Schromm

Martin Halder

9. Januar 2002

### 1 Einleitung

Das Gesicht spielt eine wichtige Rolle im Umgang mit anderen Menschen. Im allgemeinen ist es das Gesicht und die Gestik, an der wir einen Menschen erkennen und seine momentane Gemütslage einzuschätzen vermögen. Daher ist für den Menschen im Allgemeinen sehr angenehm, von einem Computer durch Anschauen erkannt zu werden.

Der Prozess der Authentifizierung wird nicht mehr bewusst wahrgenommen. Für die Erkennung von Gesichtern existieren mehrere Verfahren, die sich in zwei Ansätze aufteilen lassen. Die einen verwenden markante Merkmale eines Gesichtes wie z. B. Augen, Nase, Ohren oder Kopfform, die anderen betrachten die Bilder als Ganzes.

### 2 Basisverfahren der *Eigenfaces* Methode

Die Methode der *Eigenfaces* basiert auf dem Verfahren der Karhunen-Loeve-Transformation (KLT), auch Principal-Component-Analysis (PCA) genannt. Dabei geht es um die effiziente Repräsentation von Bildern.

Mit einer Menge von Bildern wurde ein Koordinatensystem berechnet, wobei jede Koordinate ein Bild darstellt, das Eigenpicture genannt wurde. Eine Sammlung von Bildern kann komplett rekonstruiert werden, indem nur eine kleine Menge

von Gewichten und den dazugehörigen Standardbildern (Eigenpictures) speichert.

### 3 Das *Eigenface* Verfahren

Das Verfahren der *Eigenfaces* kann ebenso für die Erkennung anderer Bilder verwendet werden.

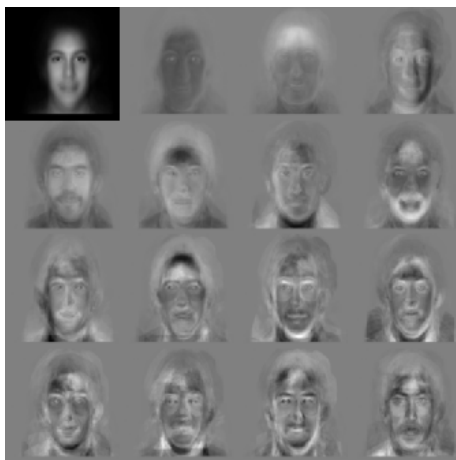
Ziel des Verfahrens ist es, die im Trainingsset der Gesichter enthaltenen charakteristischen Informationen zu extrahieren und die Gesichter als gewichtete Summen dieser Informationen zu speichern.

Die Installation des Systems besteht aus zwei Schritten.

Der erste Schritt ist eine Lernphase, in der die als bekannt anzunehmenden Gesichter analysiert und die relevanten Informationen aus dem Bild extrahiert werden.

Der zweite Schritt ist die Erkennung eines gegebenen Gesichtes. Hier wird das Gesicht auf die während der Analyse erhaltenen Informationen projiziert und die erhaltenen Werte auf Übereinstimmung mit den gespeicherten Werten verglichen.

Die zweidimensionalen Gesichtsbilder werden als Vektoren in einem multidimensionalen Raum betrachtet. Diese Vektoren liegen nicht beliebig verstreut in diesem Raum, sondern häufen sich in einem bestimmten Bereich dieses Raumes. Diesen Unterraum kann man durch Eigenvektoren beschreiben.

Abbildung 1: *Eigenfaces*

Diese Eigenvektoren, die die wichtigsten Merkmale der Gesichter darstellen werden als *Eigenfaces* bezeichnet. Die Visualisierung dieser *Eigenfaces* ergibt gesichtähnliche Bilder (siehe Bild).

Jedes dem System präsentierte Gesicht kann als Linearkombination dieser Eigenvektoren dargestellt werden. Der durch die *Eigenfaces* aufgespannte Unterraum wird *Facespace* genannt. Das Verfahren der *Eigenfaces* kann auch dazu verwendet werden, in einem gegebenen Bild, z. B. einer Raumaufnahme, ein Gesicht zu erkennen.

### 3.1 Vorgang der Installation

1. Zusammenstellen der Trainingsmenge
2. Berechnung der *Eigenfaces* der Trainingsmenge. Die besten  $m$  *Eigenfaces* mit den grössten Eigenwerten werden behalten
3. Berechnung der dazugehörigen Verteilung im  $m$ -dimensionalen Gewichtsraum für jedes bekannte Gesicht durch Projektion der Gesichter auf den *Facespace*

### 3.2 Vorgang der Erkennung

1. Berechnung der  $m$  Gewichte des gegebenen Bildes
2. Bestimmen, ob das Bild ein Gesicht ist (grosse Übereinstimmung mit dem *Facespace*)
3. Falls das Bild ein Gesicht darstellt, die Gewichte bestimmen und als bekanntes oder unbekanntes Gesicht klassifizieren
4. Update der *Eigenfaces* und oder der Gewichte (Lernen)

## 4 Berechnung der *Eigenfaces*

Sei  $I(x, y)$  das (8 Bit Graustufen) Bild eines Gesichts der Größe  $N \times N$ . Ein Bild wird als ein Vektor der Dimension  $N^2$  betrachtet. Dadurch stellt jedes  $N \times N$ -Bild einen Vektor in einem Raum mit  $N^2$  Dimensionen dar.

Da Bilder von Gesichtern eine Menge Eigenschaften gemeinsam haben, werden sie nicht zufällig in diesem Raum verteilt sein, sondern sich in einem bestimmten Bereich sammeln. Dieser Bereich lässt sich als Subraum mit einer wesentlich geringeren Anzahl an Dimensionen darstellen.

Gegeben ist ein Satz von Bildern  $\Gamma_1, \Gamma_2, \Gamma_3, \dots, \Gamma_M$ . Das Durchschnittsgesicht aus diesem Satz ist als  $\Psi = \frac{1}{M} \sum_{n=1}^M \Gamma_n$  definiert.

Jedes Gesicht  $\Gamma_n$  unterscheidet sich um den Vektor  $\Phi_i = \Gamma_i - \Psi$  vom Durchschnitt.

Aus diesem Satz von Vektoren werden nun  $M$  orthogonale Vektoren berechnet, welche den Raum, in dem die Bilder ( $\Gamma_n, n = \{1, 2, 3, \dots, M\}$ ) liegen, beschreiben. Der  $k$ -te Vektor  $\mathbf{u}_k$  wird gewählt so daß

$$\lambda_k = \frac{1}{M} \sum_{n=1}^M (\mathbf{u}_k^T \Phi_n)^2 \quad (1)$$

ein Maximum in Bezug auf

$$\mathbf{u}_l^T \mathbf{u}_k = \delta_{lk} = \begin{cases} 1, & \text{wenn } l = k \\ 0, & \text{sonst} \end{cases} \quad (2)$$

ist.

Die Vektoren  $\mathbf{u}_k$  und die Skalare  $\lambda_k$  sind Eigenvektoren, bzw. Eigenwerte der Kovarianzmatrix (3).

$$C = \frac{1}{M} \sum_{n=1}^M \Phi_n \Phi_n^T \quad (3)$$

## 5 Gesichtserkennung mit *Eigenfaces*

Da die *Eigenfaces* eine Basis für den Unterraum darstellen, in dem sich die Bilder befinden, lassen sich Bilder als eine Linearkombination bezüglich dieser Basis darstellen. Diese Linearkombination läßt sich recht einfach mittels

$$\omega_k = \mathbf{u}_k^T (\Gamma - \Psi) \quad (4)$$

berechnen, wobei  $k = 1, \dots, M$ . Dieses beschreibt eine Menge von punktwisen Bildmultiplikationen.

Die einzelnen Gewichtungen  $\omega_k$  bilden einen Vektor  $\Omega^T = [\omega_1, \omega_2, \dots, \omega_M]$  die den Beitrag der einzelnen *Eigenfaces* zum Eingabebild darstellen.

Dieser Vektor wird dann verwendet, um festzustellen ob das Gesicht bekannt ist, oder nicht. Die einfachste Methode wäre, den Euklidischen Abstand (s. 5) zu minimieren.

$$\epsilon_k = \|(\Omega - \Omega_k)\|^2 \quad (5)$$

$\Omega_k$  stellt ein bekanntes Gesicht dar.

Das Gesicht gilt als bekannt, wenn  $\epsilon_k$  kleiner als ein vorher gewählter Schwellenwert  $\theta_\epsilon$  ist.

## 6 Auffinden von Gesichtern

Die bisherigen Methoden gingen von Bildern fester Größe aus, die ein Gesicht darstellen. Dies ist in einer praktischen Anwendung nicht ohne weiteres zu erreichen, da Gesichtserkennung nicht auf die genaue Ausrichtung des Gesichtes vor der Kamera angewiesen sein sollte.

Deshalb ist ein Verfahren nötig, um in einem Bild ein Gesicht zu finden und dann zu erkennen. Dieses Verfahren macht auch das Verfolgen des Gesichtes durch das Bild möglich.

Um ein Gesicht in einem Einzelbild zu finden, kann man auf denselben „Gesichtsraum“ zurückgreifen, der auch zur Erkennung verwendet wird. Zur Erkennung wird im Bild für jeden Punkt mit seiner Umgebung  $\epsilon$  für das lokale Teilbild berechnet und in eine „Gesichtskarte“  $\epsilon(x, y)$  eingetragen.

$\epsilon(x, y)$  enthält nun für jeden Punkt des Bildes den Euklidischen Abstand (s. 5) seiner Umgebung zum Durchschnittsgesicht. Das Minimum dieser „Karte“ zeigt an, wo sich in einem Bild ein Gesicht befindet.

## 7 Das System im Einsatz

### 7.1 Durchgeführte Experimente

Um die Sicherheit, Fehlertoleranz und die Geschwindigkeit dieser Art der Gesichtserkennung festzustellen, wurden Experimente mit gespeicherten Gesichtern durchgeführt und ein System zur Erkennung von Gesichtern in einer sich ändernden Umgebung erstellt.

Dafür wurde eine Datenbank von Gesichtern erstellt, die verwendet wurde, um das Verhalten des Systems unter kontrollierten Bedingungen zu ermitteln.

### 7.1.1 Die Gesichts-Datenbank

Die Datenbank besteht aus 2500 Bildern, die unter kontrollierten Bedingungen digitalisiert wurden. Sechzehn Menschen wurden in allen Kombinationen aus drei Kopfstellungen, drei Kopfgrößen und drei Belichtungen aufgenommen. Aus jedem Bild wurde eine sechsstufige Gaußpyramide erstellt, die von den Auflösungen  $512 \times 512$  bis zu  $16 \times 16$  gingen.

Im ersten Experiment wurde der Einfluß von Lichtveränderungen, Kopfstellung und -größe ermittelt, indem die komplette Datenbank verwendet wurde.

Um das System zu trainieren wurden jeweils 16 Gesichter der Datenbank mit den gleichen Bedingungen (Kopfstellung, Kopfgröße, Belichtung) verwendet. Die Schwelle für unbekannte Gesichter wurde auf  $\infty$  gesetzt und dadurch jedes der 16 Gesichter als bekannt akzeptiert.

Zur Klassifizierung wurden jeweils 7 Gesichter verwendet. Mit diesem Versuch wurden folgende Ergebnisse für die durchschnittliche Klassifikation erzielt:

1. 96% bei variabler Beleuchtung
2. 85% bei variabler Ausrichtung
3. 64% bei variabler Größe

An diesem Ergebnis lässt sich gut erkennen, welche Veränderungen der Umgebung leicht vom System ausgeglichen werden können. Durch ein Wechsel der Beleuchtung bleibt die Korrelation benachbarter Pixel nahezu identisch, während eine Änderung der Ausrichtung oder der Größe die Korrelationsinformation stark verfälscht.

## 7.2 Gesichtserkennung in Echtzeit

Das Ziel der Studie am MIT war ein System zu entwickeln, das nahezu in Echtzeit Gesichter aus einer unstrukturierten Umgebung erkennen kann.

Hierfür wird im einer Folge von Bildern ein bewegliches Objekt isoliert und überprüft, ob es sich um ein Gesicht handelt.

Falls es ein Gesicht ist, wird das Unterbild an einen anderen Prozess weitergereicht, der die Gesichtserkennung durchführt.

Die Lokalisierung des Gesichtes wurde auf einer Sun 3/160 durchgeführt, während die Erkennung auf einer Sun Sparcstation berechnet wurde. Auf dieser Hardware konnten zwei bis drei Erkennungsvorgänge pro Sekunde durchgeführt werden.

## 8 Schlussfolgerung

Der *Eigenface* Ansatz bietet eine sehr praktikable Lösung, die für die Gesichtserkennung gut geeignet ist. Durch die Einfachheit der Berechnungen kann ein sehr performantes System realisiert werden und einen Einsatz in Echtzeit mit kostengünstiger Hardware möglich machen.

Der Einsatz von neuronalen Netzen ist möglich und erweitert das Einsatzgebiet dieses Ansatzes, da das System, abhängig vom Einsatzgebiet, eigenständig neue Gesichter in die Datenbank aufnehmen kann.

Es werden auch schon Versuche unternommen, das Geschlecht des vorgegebenen Gesichtes mit Hilfe der *Eigenfaces* zu erkennen.

## Literatur

- [1] Pentland, Choudhury (2000). Face Recognition for Smart Environments. IEEE Computer, Volume 33, Number. 2, Seiten 50–55
- [2] Turk, Pentland (1991). Eigenfaces for Recognition. Journal of Cognitive Neuroscience Volume 3, Number 1, Seiten 71–86