

# DATA MINING

- Versuch 6: Gesichtserkennung -

## Teammitglieder

---

JanHorak

Ralf Palyov

Andreas Mayer

### 3.4 Aufgaben

#### 1 Ab welcher Anzahl K von verwendeten Eigenvektoren treten Fehlklassifikationen ein?

Bei der Verwendung von nur einem Eigenvektor als Eigenface wird nur das dritte und sechste Bild richtig zugeordnet. Sobald zwei oder mehr Eigenvektoren verwendet werden, funktioniert die Klassifikation für alle Testbilder. Um sicher zu gehen wird die Verwendung von drei Eigenvektoren empfohlen.

#### 2 Wie groß ist dann die Mindestdistanz zwischen Test- und nächstliegendem Trainingsbild?

Insgesamt sind bei der Verwendung von nur einem Eigenface fünf Bilder falsch zugeordnet worden:

Bild	ki_04_of_28	ki_07_of_28	ki_16_of_28	ki_20_of_28	ki_28_of_28
Distanz (zu falschem Bild)	0,52	34,48	4,96	18,69	2,11

Die anderen zwei wurden richtig zugeordnet:

Bild	ki_12_of_28	ki_24_of_28
Distanz (zu richtigem Bild)	1,09	35,54

Es lässt sich leider kein Zusammenhang zwischen der Distanz und der richtigen Klassifikation herstellen. Selbst Abbildung 1 wird mit einer Distanz von 32.52 klassifiziert, obwohl es keinen Menschen darstellt.



**Abbildung 1: Ziege. Test, ob die Distanz für Bilder die keinen Menschen darstellen signifikant größer wird.**

### 3 Wie ändert sich die Distanz zwischen Bildern, wenn die Anzahl der Eigenvektoren reduziert wird?

Grundsätzlich wird die Distanz kleiner wenn die Anzahl der Eigenvektoren, die als Eigenfaces genutzt werden, reduziert wird. Um dies zu testen, wurde das Bild ki\_24\_of\_28 mit verschiedenen Werten für K klassifiziert:

K	1	2	3	4	5
Distanz	35,54	35,51	37,03	70,67	107,97

Das Verhalten tritt bei dem verwendeten Bild allerdings erst ab  $K = 2$  zutage. Der Grund ist, dass das Bild im zwei dimensionaligen Eigenspace sehr nahe einer Achse liegt. Wird die Dimension weiter reduziert, verändert sich Distanz kaum. Bei einem anderen Bild (ki\_12\_of\_28) liegt die Distanz bei  $K = 1$  bei 1,09 und bei  $K = 2$  bei 11,16. Hier ist die Werteentwicklung schon bei sehr wenigen Dimensionen deutlich.

### 4 Wie könnte dieser Einfluss der Eigenvektor-Anzahl auf die Mindestdistanz reduziert werden?

Momentan wird die euklidische Distanz zwischen den Punkten ermittelt. Diese zeichnet sich dadurch aus, dass absolute Abstände ermittelt werden, die abhängig von der absoluten Entfernung der Punkte sind. Kommt eine Dimension hinzu, werden die absoluten Abstände größer.

Wie im dritten Versuch (recommender Systeme) bereits verwendet, könnte hier ein Ähnlichkeitsmaß zum Einsatz kommen, bei dem die Verhältnisse zwischen den Werten und nicht die absoluten Werte bewertet werden. Ein solches Maß ist beispielsweise die Pearson Korrelation oder das Cosinus Ähnlichkeitsmaß.

### 5 Nennen Sie zwei Algorithmus-unabhängige Parameter, die starken Einfluss auf die Rate korrekter Gesichtserkennungen haben.

Neben der Gesamtanzahl der Pixel haben auch die Beleuchtungskonditionen der Bilder einen starken Einfluss auf die Rate korrekter Gesichtserkennungen. Wenn beispielsweise die Gesichtserkennung mit Personalausweisbildern oder ähnlichen biometrischen Bildern trainiert wurde, zum Klassifizieren allerdings nur ein kleines, unscharfes, unterbelichtetes Bild einer Überwachungskamera vorhanden ist, bei dem die Person vielleicht sogar eine Grimasse zieht, kann die Erkennung sehr schwierig werden.

## Zusatzaufgabe

### Ausgabe der Eigenfaces

Aus Neugierde haben wir beschlossen auch die Eigenfaces auszugeben. Das Resultat ist im Ordner res/eigenvectors zu sehen. Jedes Eigenface beschreibt per Definition ein Merkmal aus dem Pool der Eingabebilder. Die Eingabebilder können also durch die Kombination der Eigenfaces rekonstruiert werden (mehr oder weniger, es werden schließlich nur die Eigenvektoren verwendet die den höchsten Eigenwert besitzen, außerdem berechnen wir ohnehin nur so viele Eigenvektoren wie Eingabebilder vorhanden sind). Dementsprechend sehen die Resultate aus wie erwartet: Wie eine Mischung aus den verschiedenen Gesichtern. Allerdings ist doch zu erkennen, dass nicht jedes Bild die selben Merkmale enthält: Manche zeigen gar keinen Ansatz eines Bartes, andere lassen keine Brille erkennen, usw.

Das Aufbereiten der Eigenfaces für die Darstellung ist nicht ganz trivial: Die Pixel der Bilder denen die Eigenfaces zu Grunde liegen wurden zuerst in den Wertebereich von null bis eins gepresst und anschließend normiert.

Dementsprechend liegen auch die Pixelwerte der Eigenfaces nicht in einer darstellbaren Form vor.

Darüber hinaus wird jedem Bild bei der Normierung ein Teil abgeschnitten. Die Ursache liegt darin, dass von jedem Pixel der entsprechende Pixelwert des Durchschnittsbildes subtrahiert wird. Das Resultat kann dabei kleiner als null werden. Diese negativen Werte werden aber einfach auf null gesetzt.

Um Bilder ausgeben zu können, müssen die Eigenfaces zuerst wieder mit dem Durchschnittsbild summiert werden. In

diesem Schritt wird die Normierung rückgängig gemacht.

Dadurch, dass die Eigenfaces nicht im Durchschnittsbild enthalten sind, können sich Werte ergeben, die außerhalb des Wertebereichs zwischen null und eins liegen. Die Ergebnisse müssen also wieder in diesen Wertebereich überführt werden. Dies geschieht durch das abschnitten zu großer Werte. Zu kleine Werte können nicht enthalten sein, da hier nur addiert wird und der ursprüngliche Wert größer gleich null ist.

Abschließend muss der Wertebereich erneut angepasst werden, denn: Die Image Klasse des Python Packages PIL erwartet für Bildinstanzen einen Wertebereich zwischen null und 255 für jeden Pixel und jede Farbe, in unserem Fall jedoch reicht ein Wert für jeden Pixel, da wir mit Schwarzweißbildern arbeiten.

Die Vergleichbarkeit der Eigenfaces mit den Trainingsbildern ist nicht uneingeschränkt gegeben:

Dadurch, dass den Trainingsbildern beim Normieren Teile abgeschnitten werden, sind diese Teile auch in den Eigenfaces nicht mehr vorhanden. Wir haben getestet, ob sich die Erkennungsrate ändert, wenn auch negative Werte erlaubt sind, also nichts abgeschnitten wird: Die Erkennung wird massiv schlechter.

Durch das Addieren des Durchschnittsbildes auf die Eigenfaces, decken diese in der Realität ein größeres Helligkeitsspektrum ab als die Eingabebilder. Um dem entgegen zu wirken müsste ein anderes Bildformat oder abgedunkelte Eingabebilder verwendet werden.