

Eberhard Karls Universität Tübingen

Mathematisch-Naturwissenschaftliche Fakultät

Wilhelm-Schickard-Institut für Informatik

Dokumentation Praktikum Bildverarbeitung

Automatisches ausschneiden von Bildern aus Fotoalben

Mariella Dreißig, Tom Eckardt, Stefan Bergmann

17. Dezember 2018

Betreuer

Andreas Karge, Prof. Andreas Schilling

Mariella Dreißig, Tom Eckardt, Stefan Bergmann:

Automatisches ausschneiden von Bildern aus Fotoalben

Dokumentation Praktikum Bildverarbeitung

Eberhard Karls Universität Tübingen

Zusammenfassung

Inhaltsverzeichnis

1	Einleitung	5
1.1	Problemstellung	5
1.2	Benutzte Technologien	5
1.3	Programmaufbau	5
2	Hintergrundentfernung	6
3	Randentfernung	7
4	Gesichtserkennung	8
4.1	Cascade Classifier	9
5	Auswertung	11
5.1	Metriken zur Auswertung	11
5.2	Hintergrundentfernung	11
5.3	Randentfernung	11
5.4	Gesichtserkennung	11
6	Fazit	16
7	Benutzung und Konfiguration	17
	Literaturverzeichnis	18

1 Einleitung

1.1 Problemstellung

1.2 Benutzte Technologien

1.3 Programmaufbau

2 Hintergrundentfernung

3 Randentfernung

4 Gesichtserkennung

Die Aufgabenstellung beinhaltet, in den extrahierten Fotos Gesichter zu erkennen. Hier muss die Gesichtserkennung (in der Fachliteratur "Face Detection") von der Wiedererkennung von bereits identifizierten Gesichtern (in der Fachliteratur "Face Recognition") unterschieden werden. In der vorliegenden Arbeit geht es ausschließlich darum, die Position von Gesichtern in den eingescannten Fotos zu erkennen und nicht, bereits erkannte Gesichter wiederzuerkennen oder zuzuordnen ¹.

In der klassischen Bildverarbeitung wurden lange einfache 2D-Verfahren verwendet, um Gesichter in Bildern zu erkennen ([1], [6]). Beim Template Matching werden beispielsweise Vorlagen (von Teilen) von Gesichtern erzeugt um diese dann im Anwendungsfall mit den Zielbildern zu vergleichen. Dieses Verfahren ist nicht sehr robust, besonders bei sich verändernden Lichtverhältnissen, aber gleichzeitig extrem rechenaufwändig.

Eine einfachere Methode der Gesichtserkennung in der klassischen Bildverarbeitung ist die Nutzung von geometrischen Merkmalen in den Gesichtern. Hierbei werden beispielsweise Grauwerte in einem bestimmten Bereich des Bildes aufsummiert oder Gradientenoperatoren zur horizontalen und vertikalen Kantenerkennung angewandt. So kann die Ausrichtung, die relative Lage und ähnliche Merkmale der Nase, der Augen und des Mundes identifiziert und zusammen als Gesicht klassifiziert werden.

Doch auch diese Methode ist rechen- und zeitintensiv. Deswegen wird häufig eher auf komplexere Methoden zur Gesichtserkennung gesetzt, wie beispielsweise das Elastic Bunch Graph Matching, das eine robuste Waveletanalyse nutzt ([7]) oder die auf der Hauptkomponentenanalyse basierende Eigenfaces-Methode ([4]). Weiterentwicklungen dieser beiden Methoden werden heute noch eingesetzt, obwohl die Ansätze schon vergleichsweise alt sind.

Im Folgenden wird ein populärer Ansatz vorgestellt, der in den vorliegenden Arbeit verwendet wurde.

¹ Aufgrund der geringen Datenmenge und der schlechten Qualität der alten Fotos ist es nicht möglich, Gesichter wiederzuerkennen.

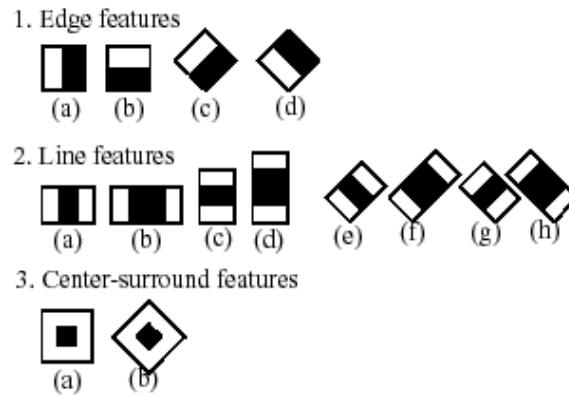


Abbildung 4.1: Alle vom Cascade Classifier Algorithmus genutzten Haar-ähnlichen Features zur Objekterkennung. Zur Gesichtserkennung werden nur fünf Features genutzt: 1(a), 1(b), 2(a), 2(c) und eine Kombination aus zwei 1(a) Features, die zusammen ein schachbrettartiges Muster ergeben.

4.1 Cascade Classifier

Die OpenCV Programmbibliothek beinhaltet vorimplementierte Funktionen zur Gesichtserkennung. Sie basiert auf einem 2001 entwickelten Ansatz zur Objekterkennung in Bildern ([5], [2]). Es ist ein trainierter Classifier, der einfache geometrische Merkmale im Bild erkennt und auf deren Basis die gewünschten Objekte erkennen kann ([3]).

Die Methode nutzt sogenannte Haar-ähnliche Features, mit denen grundlegende Strukturen (Helligkeitsunterschiede) im Bild erkannt werden. Im Bild 4.1 werden alle Features aufgelistet, die der Classifier nutzen kann. Für die Gesichtserkennung werden in der Praxis aber nicht so viele genutzt, sondern nur einige wenige. Mit diesen können allgemeine Eigenschaften von menschlichen Gesichtern repräsentiert werden. So ist zum Beispiel die Nase heller als die Bereiche um die Augen, das entspricht dem Feature 2(a) in Bild 4.1.

Nachdem die zu nutzenden Features ausgewählt wurden, wird ein Integralbild erzeugt, das in der Lage ist, die Features in konstanter Zeit auszuwerten. Dadurch wird eine relativ geringe Laufzeit erzielt.

In der Trainingsphase wird der Classifier mit Positiv- und Negativbeispielen von dem zu erkennenden Objekt trainiert, in diesem Fall mit Gesichtern². Dieser Vorgang wird mit verschiedenen Arten des AdaBoostings beschleunigt und verbessert. Dadurch werden die besten Features für die Klassifizierung identifiziert.

²In der OpenCV Bibliothek ist sowohl ein Trainer als auch ein Detector enthalten. Der Detector ist vortrainiert kann direkt zur Gesichtserkennung verwendet werden. Wenn man genug Daten zur Verfügung hat, kann man aber auch einen selbst trainieren.

4 Gesichtserkennung

Danach wird der Classifier in mehreren Stufen auf die Bilder angewandt ("cascade"), in denen das Objekt identifiziert werden soll. Jeder Classifier funktioniert dabei wie ein Filter, der die negativen Regionen aus dem Bild herausfiltert. Die verbleibenden Bereiche werden Stück für Stück durch immer komplexere Classifier gegeben, bis sie entweder aussortiert werden oder das Objekt (ein Gesicht) identifiziert wurde. Somit hat der Cascade Classifier die Struktur eines Entscheidungsbaumes und Rechenzeit wird gespart. Die Methode ist so robust, dass das Objekt in verschiedenen Größen in dem Bild erkennen kann, unabhängig von der Größe der Bilder, mit der der Classifier trainiert wurde.

Wenn die OpenCV Methode `detectMultiScale` ein Gesicht im Input-Bild erkennt, gibt es die Position in x- und y-Koordinaten und dessen Abmessungen (Höhe, Breite) zurück. In dem vorliegenden Programm werden zwei verschiedene Classifier auf die Bilder angewandt: einmal frontale Gesichter und einmal Gesichter im Profil. Die Rückgabewerte der Funktionen werden dafür genutzt, die Gesichter auszuschneiden und Dopplungen zu erkennen. Da es passieren kann, dass beide Classifier die gleichen Gesichter detektieren, wird durch eine Hilfsfunktion festgestellt, ob dieses Gesicht bereits in der Liste der erkannten Gesichter vorhanden ist. Dafür wird überprüft, ob der Mittelpunkt des kleineren Ausschnitts in dem größeren liegt. Wenn das der Fall ist, kann angenommen werden, dass es sich um das gleiche Gesicht handelt und eines davon wird verworfen. So wird sichergestellt, dass keine Gesichter verworfen werden, die lediglich nah beieinander liegen, so wie beispielsweise in einem Gruppenbild.

5 Auswertung

5.1 Metriken zur Auswertung

5.2 Hintergrundentfernung

5.3 Randentfernung

5.4 Gesichtserkennung

Die von OpenCV mitgelieferten Methoden zur Gesichtserkennung auf Bildern funktioniert grundsätzlich gut. Sie ist einfach anwendbar, hat eine vertretbare Laufzeit (der Algorithmus läuft nur einige Sekunden auf einer ganzen eingescannten Albumseite) und erkennt zumindest einen Großteil der Gesichter in den Fotos.

Um die Performanz der Classifier einzeln und in Kombination auszuwerten, wurden Wahrheitsmatrizen aufgestellt. Ausgewertet werden jeweils die korrekt erkannten Gesichter (True Positives (TP): positive - positive), die nicht erkannten Gesichter (False Negatives (FN): positive - negative) und die fälschlicherweise erkannten Gesichter (False Positives (FP): negative - positive). Korrekt nicht erkannte Gesichter (True Negatives (TN): negative - negative) gibt es in unserem Anwendungsfall nicht.

Tabellen 5.1 und 5.2 zeigen die Metriken in dem ersten und dem zweiten Datensatz wenn nur der frontal faces Classifier angewendet wird. Die daraus errechnete Genauigkeit ¹ beträgt im ersten Datensatz 0.75 und im zweiten 0.59.

Wenn der zweite Classifier zur Erkennung von Profilen mit dazu genommen wird, erhöht sich die Anzahl der erkannten Gesichtern ein wenig. Das ist in Tabellen 5.3 und 5.4 zu sehen. Die Gesichter, die bereits von dem frontal-Classifier erkannt wurden, werden automatisch heraus

¹Formel für die Berechnung der Genauigkeit ("Accuracy") ist: $\frac{TP+TN}{TP+FN+TN+FP}$, ($TN = 0$ in unserem Fall).

5 Auswertung

	Reference		
Prediction		Positive	Negative
	Positive	270	1
	Negative	88	-

Tabelle 5.1: Wahrheitsmatrix im ersten Datensatz, wenn nur der frontal-Classifer zum Einsatz kommt. Die Genauigkeit der Gesichtserkennung beträgt 0.75.

		Reference	
		Positive	Negative
Prediction	Positive	282	7
	Negative	172	-

Tabelle 5.2: Wahrheitsmatrix im zweiten Datensatz, wenn nur der frontal-Classifer zum Einsatz kommt. Die Genauigkeit der Gesichtserkennung beträgt 0.59.

gefiltert. Danach bleibt ein kleiner Mehrgewinn durch die Anwendung des profile-Classifiers übrig, die Performanz verbessert sich auf 0.76 und 0.65 jeweils.

		Reference	
		Positive	Negative
Prediction	Positive	274	2
	Negative	84	-

Tabelle 5.3: Wahrheitsmatrix im ersten Datensatz, wenn frontal- und profile-Classifer kombiniert werden. Die Genauigkeit der Gesichtserkennung beträgt 0.76.

Festlegung der Ground Truth

Die Referenz für die Auswertung der Gesichtserkennung festzulegen ist nicht trivial. Auf den alten Fotos sind aufgrund der schlechten Qualität der Aufnahmen viele Gesichter nicht tatsächlich zu erkennen. Vielmehr erschließt sich ein menschlicher Betrachter aus dem Kontext die Information, dass sich an gewissen Stellen ein Gesicht befinden muss. Das wurde bei der Festlegung der Ground Truth berücksichtigt. So wurden beispielsweise besonders kleine, verdeckte oder stark verschwommene Gesichter für die Berechnung der Metriken nicht berücksichtigt. Die Classifier haben außerdem Probleme, Gesichter zu erkennen, wenn starke Okklusionen auftreten, selbst wenn das Gesicht nur im Profil zu erkennen ist oder die Person eine Kopfbedeckung trägt. Diese wiederum wurden in die Gesamtsumme der Gesichter mit einberechnet. Im Folgenden werden exemplarisch einige Bilder gezeigt, die der Gesichtserkennungsalgorithmus nicht erkennen kann und die deswegen nicht in die Auswertung mit einbezogen wurden.

5 Auswertung

		Reference	
		Positive	Negative
Prediction	Positive	301	8
	Negative	153	-

Tabelle 5.4: Wahrheitsmatrix im zweiten Datensatz, wenn frontal- und profile-Classifer kombiniert werden. Die Genauigkeit der Gesichtserkennung beträgt 0.65.

Bild 5.1 zeigt ein kleines Mädchen, jedoch ist die Aufnahme sehr verschwommen. Jeder Mensch, der das Bild betrachtet, weiß, dass das Mädchen ein Gesicht haben muss. Wenn jedoch die Kontextinformationen entfernt werden würde, wäre es selbst für einen Menschen schwer, ein Gesicht zu identifizieren.

Auf Bild 5.2 ist eine ähnliche Situation zu sehen: alle abgebildeten Menschen haben Gesich-



Abbildung 5.1: Starke Bewegung im Bild verhindert, dass das Gesicht des Mädchens tatsächlich zu sehen ist, auch wenn menschliche Betrachter durchaus ihr Gesicht erkennen könnten.

ter, jedoch sind sie objektiv nicht wirklich zu sehen. Sie sind relativ klein, was normalerweise für den Classifier kein Problem darstellt, aber eine schlechte Beleuchtung und geringer Kontrast tragen dazu bei, dass die Gesichter nicht erkannt werden.

Um dennoch eine repräsentative Auswertung der Performanz der Gesichtserkennung zu gewähr-



Abbildung 5.2: Die Gesichtserkennung schlägt fehl, wenn die Gesichter zu schlecht beleuchtet sind und/oder zu wenig Kontrast aufweisen. Auch, wenn normalerweise auch Gesichter in dieser Größe von dem Algorithmus problemlos erkannt werden.

leisten, wurden viele nicht erkannte Gesichter mit in die Metriken einberechnet. Beispielsweise können die Classifier keine Gesichter mit Kopfbedeckungen erkennen. Ansonsten sind aber die Gesichter gut zu identifizieren, was in Bild 5.3 zu sehen ist.

Bild 5.4 zeigt einen problematischen Grenzfall aus dem zweiten Datensatz. In diesem Bild wurden insgesamt drei Gesichter erkannt (zwei vom frontal-Classifier (grün) und eins vom profile-Classifier (rot)). Da alle Gesichter in etwa die gleiche Größe und Schärfe haben, sollte angenommen werden, dass auch die restlichen Personen erkannt werden. Es ist nicht ersichtlich, warum auf diesem Bild beide Classifier keine weiteren Gesichter erkennen. Von daher fließen alle Gesichter auf diesem Foto mit in die Bewertung ein und erhöhen die Anzahl der False Negatives enorm. Besonders im zweiten Datensatz kommen solche Fälle häufiger vor, was die schlechte Performanz des Algorithmus' im Vergleich zum ersten Datensatz erklärt.

5 Auswertung



Abbildung 5.3: Kein Classifier erkennt Gesichter mit Hüten, auch wenn sie nur einen kleinen Teil vom Gesicht verdecken. Alle anderen Gesichter werden problemlos identifiziert.



Abbildung 5.4: In diesem Gruppenfoto werden nur wenige Gesichter erkannt, obwohl sie alle ähnlich aussehen. Das verschlechtert die Kennwerte für die Genauigkeit besonders im zweiten Datensatz stark.

6 Fazit

7 Benutzung und Konfiguration

Literaturverzeichnis

- [1] Kees, Benjamin: *Gesichtserkennung*, 2012. <https://www.algoropticon.de/Dateien/Seminararbeit{ }Gesichtserkennung.pdf>.
- [2] Lienhart, Rainer, Alexander Kuranov, Vadim Pisarevsky und M R L Technical Report: *Empirical Analysis of Detection Cascades of Boosted Classifiers for Rapid Object Detection*. Technischer Bericht, 2002. <http://www.multimedia-computing.de/mediawiki//images/5/52/MRL-TR-May02-revised-Dec02.pdf>.
- [3] OpenCV: *Haar Feature-based Cascade Classifier for Object Detection*. <https://docs.opencv.org/2.4/modules/objdetect/doc/cascade{ }classification.html{#}grouprectangles>, besucht: 2018-12-09.
- [4] Turk, Matthew und Alex Pentland: *Eigenfaces for Recognition*. Journal of Cognitive Neuroscience, 3(1):71–86, 1991. <https://doi.org/10.1162/jocn.1991.3.1.71>.
- [5] Viola, Paul: *Rapid Object Detection using a Boosted Cascade of Simple Features*. In: *Conference on Computer Vision and Pattern Recognition*, 2001. <http://wearables.cc.gatech.edu/paper{ }of{ }week/viola01rapid.pdf>.
- [6] Wächter, Carsten und Stefan Römer: *Gesichtserkennung I - Biometrische Systeme*, 2001. <http://www.informatik.uni-ulm.de/nl/Lehre/WS01/HS-Biometrische-Systeme/ausarbeitungen/Gesichtserkennung{ }1{ }block.pdf>.
- [7] Wiskott, Laurenz, Norbert Kr und Christoph Von Der Malsburg: *Face Recognition by Elastic Bunch Graph Matching*. Intelligent Biometric Techniques in Fingerprint and Face Detection, (11):355–396, 1999. <http://www.face-rec.org/algorithms/ebgm/wisfelkrue99-facerecognition-jainbook.pdf>.