

Face Detection

Youssef Mestiri : 11140185
Maher Mahouachi : 11143349

Modul: Bildbasierte Computergraphik
Wintersemester 2021/2022
Prof. Andreas Karge

Fakultät für Informatik und Ingenieurwissenschaften Studiengang
Master Medieninformatik
Technische Hochschule Köln
Gummersbach

Abstract

Gesichtserkennung “Face Detection” ist der Prozess, menschliche Gesichter in digitalen Bildern zu identifizieren .Doch bei Bildern von geringer Qualität können Probleme mit der Erkennung eines menschlichen Gesichtes auftreten, in diesem Paper wird mit Hilfe des referenzierten Papers ein face Detection System und die Präzision der Gesichtserkennung optimiert, indem verschiedene Bildverbesserungstechniken eingesetzt werden.

Inhaltsverzeichnis

Einleitung	4
Motivation	4
Anwendung und Techniken	5
Ergebnisse	6
Extraierungstest	6
Gesichtserkennung mit OpenCV	8
Gesichtserkennung mit MTCNN und Bildverbesserung	9
Ergebnisse	10
Probleme	11
Verbesserungsvorschläge	11
Fazit	12
Quellen	13

1. Einleitung

Die Gesichtserkennung ist ein Bereich der Computer Vision, bei dem es darum geht, ein menschliches Gesicht in einem digitalen Bild zu erkennen. Sie ist ein Spezialfall der Objekterkennung, bei der es darum geht, das Vorhandensein und den genauen Ort eines oder mehrerer Gesichter in einem Bild zu erkennen. Dies ist einer der am besten erforschten Bereiche der Computer Vision. Die starke Forschungsaktivität im Bereich der Gesichtserkennung hat auch zur Entstehung generischer Methoden zur Objekterkennung geführt.

Die Gesichtserkennung hat sehr viele direkte Anwendungen in den Bereichen Videoüberwachung, Biometrie, Robotik, Steuerung von Mensch-Maschine-Schnittstellen, Fotografie, Indexierung von Bildern und Videos, inhaltsbasierte Bildsuche etc.

2. Motivation

Obwohl Menschen Experten in der Gesichtserkennung sind, ist noch nicht klar wie diese Erkennung durchgeführt wird.

Es gibt viele Technologien, die es uns ermöglichen, Gesichter aus einem Bild zu erkennen. Unser Ziel ist es, diese Technologien zu nutzen, um ein besseres Ergebnis bei der Gesichtserkennung aus einem Fotoalbum zu erzielen.

3. Anwendung und Techniken

a. Anwendung

Die Anwendung ist ein Erkennungssystem, das Bilder extrahiert und optimiert um menschliche Gesichter zu erkennen. Es besteht aus drei Modulen. Jedes Modul stellt ein Feature der Anwendung dar.

- Extractor: Extrahierung der einzelnen Bilder aus einer Bilddatei.
- Enhancer: Preprocessing der Bilder vor der Erkennung, indem die Technik Histogram Equalization mit Hilfe der Bibliothek openCV angewandt werden.
- Detector: Erkennung der Gesichter in den extrahierten Bildern

b. FlowChart

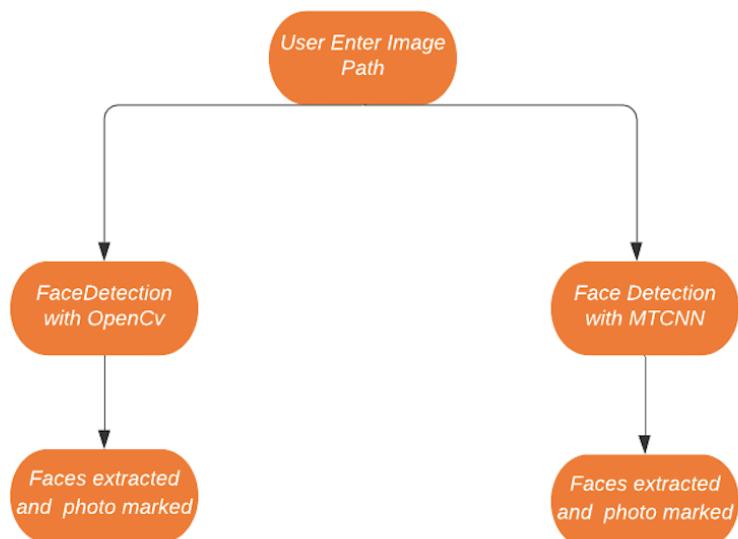


Abbildung 1: Ablaufdiagramm

c. Techniken

- Histogram Equalisation: ist eine Methode in der Bildverarbeitung zur Kontrastanpassung unter Verwendung des Histogramms des Bildes.
- Multi-Task Cascaded Convolutional Neural Network: Das Netzwerk verwendet eine Kaskadenstruktur mit drei Netzwerken; Zuerst wird das Bild auf verschiedene Größen neu skaliert (Bildpyramide genannt), dann schlägt das erste Modell (Proposal Network oder P-Net) Kandidaten für Gesichtsregionen vor, das zweite Modell (Refine Network oder R-Net) filtert die Begrenzungsrahmen , und das dritte Modell (Output Network oder O-Net) schlägt Gesichtsmarkierungen vor.

4. Ergebnisse

a. Extrahierungstest



Abbildung 2: 01.tif

Die Extraktion für das Foto (Abbildung 1) hat 5 Fotos erzeugt:

01_0.png, 01_1.png, 01_2.png, 01_3.png, 01_4.png



extrahiertes Foto 01_0.png



extrahiertes Foto 01_1.png



extrahiertes Foto 01_2.png



extrahiertes Foto 01_3.png



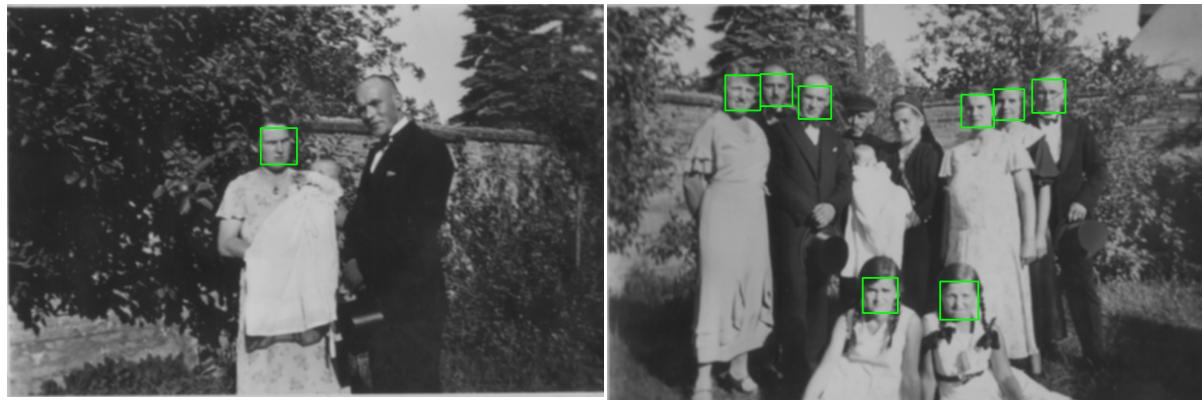
extrahiertes Foto 01_4.png

b. Gesichtserkennung mit OpenCV

Test der Gesichtserkennung: ohne Bildverbesserung



Gesichtserkennung 01_0.png Gesichtserkennung 01_1.png Gesichtserkennung 01_2.png



Gesichtserkennung 01_3.png

Gesichtserkennung 01_4.png

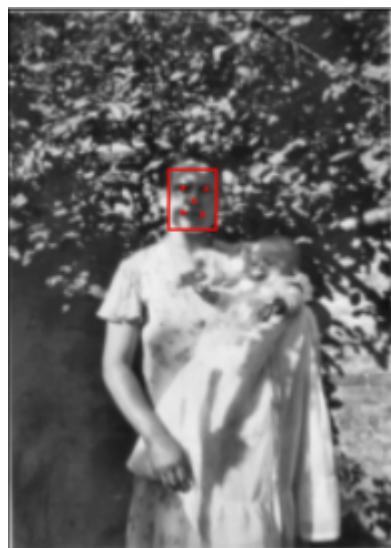
Das Testergebnis der Gesichtserkennung ohne Bildverbesserung erkannte nur 9 von 20 Gesichtern.

c. Gesichtserkennung mit MTCNN und Bildverbesserung

Bildverbesserung

Mit Hilfe der eingesetzten Technik Histogram-Equalization werden die Bilder vor der Gesichtserkennungsphase vorverarbeitet und verbessert.

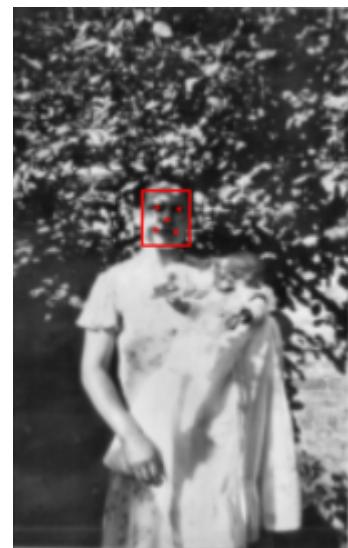
Ergebnis:



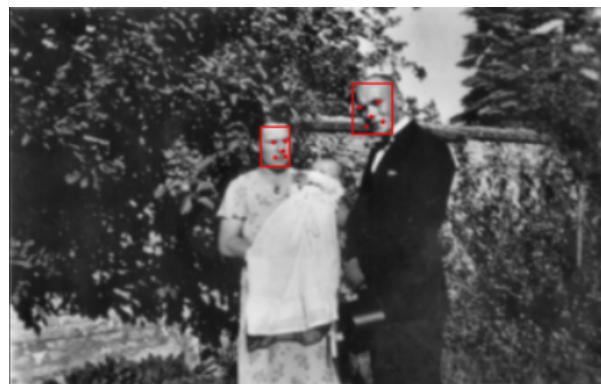
Verbessertes Foto mit
Gesichtserkennung 01_0.png



Verbessertes Foto mit
Gesichtserkennung 01_1.png



Verbessertes Foto mit
Gesichtserkennung 01_2.png



Verbessertes Foto mit Gesichtserkennung
01_3.png



Verbessertes Foto mit Gesichtserkennung
01_4.png

d. Quantitativ

Bild	Anzahl der Gesichter pro extrahiertes Bild	Ohne Bildoptimierung	Mit Histogramm Equalization und MTCNN	Verbesserung (%)
01.tif	19	9	15	60%
02.tif	16	9	14	64%
03.tif	11	7	10	42.86%
04.tif	7	1	5	400%
05.tif	ungültig	ungültig	ungültig	ungültig
06.tif	9	2	6	200%
07.tif	6	5	6	20%
08.tif	8	0	5	-
09.tif	8	7	7	0%
10.tif	8	3	8	166.67%

Tabelle 1: Gesichtserkennungsergebnisse

Die Tabelle 1 stellt die Ergebnisse der Gesichtserkennung mit und ohne Bildverbesserung dar.

Die Anwendung der Histogram-Equalization-Technik zeigt immer bessere Ergebnisse.

Bei der Histogram-Equalization (HE) wird der globale Kontrast eines Bildes durch Aktualisieren der Pixelintensitätsverteilung des Bildhistogramms angepasst. Dadurch können Bereiche mit geringem Kontrast einen höheren Kontrast im Ausgabebild erzielen.

Das Modell MTCNN wird als Multitask-Netzwerk bezeichnet, weil jedes der drei Modelle in der Kaskade (P-Net, R-Net und O-Net) auf drei Tasks trainiert wird, z.B drei Arten von Vorhersagen treffen, sie sind: Gesichtsklassifizierung, Bounding-Box-Regression und Lokalisierung von Gesichtsmerkmalen.

e. Qualitativ

Der Haar-Kaskaden-Klassifikatoren Detektor ist tendenziell am effektivsten für frontale Bilder des Gesichts. Profilbilder wurden mit Opencv2 meist nicht erkannt wegen der geringen Qualität der Bilder. Wogegen das MTCNN-Netzwerk das Bild auf verschiedene Maßstäbe skaliert, um eine Bild pyramide zu erstellen und somit dann die Frontal- und Profilbilder durch seine drei Aufgaben (P-Net, R-Net und O-Net) erkennt. Für bessere Ergebnisse ist das MTCNN einzusetzen. Der Vorteil der Verwendung von Haar-Kaskaden-Klassifikatoren ist hingegen, dass sie wesentlich schneller sind als MTCNN.

5. Performance

Basierend auf den Ergebnissen können wir feststellen, dass die Ausführung der MTCNN-Erkennung plus Bildverbesserung viermal langsamer ist als das vorhandenes Erkennungssystem (OpenCv) mit jeweils 2.40 s und 0.61 s.

Die Integration des MTCNN-Algorithmus mit der Bildverbesserung hat ihre Kosten bei der Leistung.

Da das Projektziel darin besteht, mehr Gesichter aus den Bildern zu extrahieren, spielt dieser Faktor keine große Rolle.

6. GUI

Die GUI, die es dem Benutzer ermöglicht, Gesichter aus Fotoalben oder Fotos zu erkennen und zu kommentieren, ist als Jupyter Notebook mit ipywidgets [4] implementiert. Die Widgets ermöglichen es dem Benutzer, mit dem Programm mit Schaltflächen, Textfeldern, Kontrollkästchen usw. zu interagieren. Gesichter aus dem Foto können durch Angabe eines Bildpfads extrahiert werden (siehe Abbildung 2).

Path:

Gesichter extrahieren

MTCNN Erkennung

Verbesserte Erkennung

Abbildung 3: GUI

Das Ergebnis dieses Prozesses ist eine Liste von den extrahierten Bildern mit markierten Gesichtern und eine Liste der extrahierten Gesichter.

7. Fazit

Das vorliegende Programm extrahiert einzelne Fotos aus eingescannten Albenseiten und erkennt Gesichter von den extrahierten Bildern. Das Programm wurde an 10 Bildern angewandt. Von insgesamt 92 Gesichtern wurden ohne Optimierung 44 Gesichter und mit Optimierung (HE, MTCNN) 76 Gesichter erkannt. Zusätzlich wurde durch die längere Verarbeitungszeit das Resultat verbessert.

8. Quellen

- [1] Ning Zhang, Junmin Luo, Wuqi Gao, "Research on Face Detection Technology Based on MTCNN", November 2020,
<https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9239720>
- [2] Akanksha Das, Ravi Kant Kumar,Dakshina Ranjan Kisku, "Heterogeneous Face Detection", March 2016, <https://dl.acm.org/doi/10.1145/2896387.2896417>
- [3] Md Khaled Hasan, Md. Shamim Ahsan, Abdullah-Al-Mamun, S. H. Shah Newaz and Gyu Myoung Lee, "Human Face Detection Techniques: A Comprehensive Review and Future Research Directions", <https://www.mdpi.com/2079-9292/10/19/2354/pdf>
- [4] Jupyter Widgets, 2017-2021, <https://ipywidgets.readthedocs.io/en/latest/>