

Eberhard Karls Universität Tübingen
Mathematisch-Naturwissenschaftliche Fakultät
Wilhelm-Schickard-Institut für Informatik

Masterarbeit Informatik

Exploring crowd gaze tracking techniques & applications: Measuring attention within a classroom

Falko Benezan

12. Juli 2017

Gutachter

Name Gutachter
Wilhelm-Schickard-Institut für Informatik
Universität Tübingen

Betreuer

Thomas Kübler
Adresse
Universität Tübingen

Benezan, Falko:

Exploring crowd gaze tracking techniques & applications:

Measuring attention within a classroom

Masterarbeit Informatik

Eberhard Karls Universität Tübingen

Bearbeitungszeitraum: 18.01.2017 – 18.07.2017

Zusammenfassung

Der übliche Aufbau zur Analyse von Gesichter ist, das verwenden eines Messgerät pro Person und Merkmal, wie Beispielsweise Eye-Tracking Brillen. Soll eine Auswertung auf mehreren Probanden gleichzeitig durchgeführt werden, so ist die Verwendung von weniger Geräten einfacher in der Handhabung.

Im Rahmen der Aufmerksamkeitsmessung im Unterricht, soll eine automatisierte Messung der Aufmerksamkeit einer ganzen Klassen durchgeführt werden. Dabei soll der Unterricht durch die Messinstrumente möglichst wenig beeinflusst werden.

Um Anhaltspunkte eines effizienten Aufbaus des eigentlichen Versuchs zu erhalten, sollen die Grenzen und Qualität bei der Gesichtsanalyse basierend auf Bildmaterial einer einzelnen Kamera aufgezeigt werden. Diese Kamera wird fest montiert um eine Frontalaufnahme aller Probanden gleichzeitig zu erhalten, wobei die gesamte Klasse im Fokus der Kamera liegt. Durch diesen Aufbau ergibt sich die Problematik von den sehr unterschiedlichen Distanzen zur Kamera und daraus resultierend die dargestellte Größe aller Probanden im Bild.

Um alle Probanden im Frame bewerten zu können, werden zuerst die einzelnen Gesichter im Bild gesucht, den Probanden zugeordnet und aufbereitet. Anschließend werden die Gesichter analysiert um ihre Position und Orientierung zu bestimmen. Die Augenregion wird zusätzlich behandelt, um genauere Ergebnisse bei der Bestimmung der Blickrichtung zu erhalten.

Die Versuche haben ergeben, das mit den heutigen HD Kameras eine gleichzeitige Analyse von mehreren Probanden im selben Frame möglich ist, die sich auf der Fläche eines üblichen Klassenzimmers verteilen. Für die Analyse kann meist nur auf den Gesichtern gearbeitet werden, da für die Bestimmung der Blickrichtung zu wenige Informationen in den kleinen Bildern vorhanden sind. Außerdem Können mit den verwendeten Verfahren nur Gesichter bis zu einem Winkel von 45° relativ zur Kamera erfasst werden.

Inhaltsverzeichnis

| | |
|---|-----------|
| 1 Einführung | 7 |
| 1.1 Intension | 7 |
| 1.2 Problemstellung | 8 |
| 2 Stand der Forschung | 11 |
| 2.1 Computer Vision Methoden zur Gesichtsanalyse | 11 |
| 2.1.1 Künstliches neuronales Netz | 11 |
| 2.1.2 Convolutional Neural Network (CNN) | 11 |
| 2.1.3 Constrained Local Model (CLM) | 12 |
| 2.1.4 Constrained Local Neural Fields (CLNF) | 13 |
| 2.1.5 Patch Experts | 13 |
| 2.1.6 Active Appearance Model (AAM) | 13 |
| 2.1.7 Non-maximum suppression (NMS) | 13 |
| 2.1.8 Point Distribution Model (PDM) & Generalized Adaptive View-based Appearance Model (GAVAM) | 14 |
| 2.2 Eye-Tracking in der Klassenzimmer-Umgebung | 14 |
| 3 Herangehensweise | 17 |
| 3.1 Ablauf der Implementierung | 17 |
| 3.2 Verwendete Eingabegeräte & Bibliotheken | 19 |
| 3.3 MTCNN Face Detection | 19 |
| 3.3.1 Die 3 Stufen der Verarbeitung | 19 |
| 3.3.2 Zuverlässigkeit bei der Detektion | 21 |
| 3.4 Aufbereitung der Bilder | 22 |
| 3.4.1 Bicubic-Skalierung | 22 |
| 3.4.2 Lanczos-Skalierung | 22 |
| 3.4.3 Linear-Skalierung | 23 |
| 3.4.4 Nearest-Neighbor-Skalierung | 23 |
| 3.5 OpenFace | 25 |
| 3.5.1 Bestimmung der Landmarks | 25 |
| 3.5.2 Bestimmung des Arbeitsbereiches | 27 |
| 3.5.3 Ergebnis bezüglich Verwendbarkeit | 33 |
| 3.6 Umwandlung von Farbbild nach Graubild | 34 |
| 3.6.1 Gleam-Verfahren | 34 |
| 3.6.2 Gleam-New-Verfahren | 35 |
| 3.6.3 Luminance-Verfahren | 35 |

| | | |
|----------|--|-----------|
| 3.6.4 | Min-Max-Verfahren | 35 |
| 3.6.5 | Quadrat-Verfahren | 36 |
| 3.6.6 | Normalisierung von Graubilder | 36 |
| 3.7 | Ellipse Selection for Robust Pupil Detection (ElSe) | 38 |
| 3.7.1 | Beschreibung von ElSe | 38 |
| 3.7.2 | Das Verfahren im Test | 40 |
| 3.7.3 | Ergebnis | 44 |
| 3.8 | Bestimmung des Ziels der Aufmerksamkeit | 45 |
| 3.8.1 | Bestimmung der Position & Orientierung des Gesichts | 45 |
| 3.8.2 | Größe und Genauigkeit | 47 |
| 3.8.3 | Bestimmung eines Punktes, auf der die Aufmerksamkeit liegt | 48 |
| 4 | Evaluation | 53 |
| 4.0.1 | Schulklassenvideo | 53 |
| 4.1 | Vorversuche | 53 |
| 4.1.1 | Arbeitsbereich der Verfahren - Versuch 1 | 53 |
| 4.1.2 | Arbeitsbereich der Verfahren - Versuch 2 | 56 |
| 4.1.3 | Auswertung der Augenpartie - Versuch 3 | 56 |
| 4.1.4 | Ergebnis der Vorversuche | 58 |
| 4.2 | Aufmerksamkeitsmessung - Versuch | 58 |
| 4.2.1 | Versuchsdurchführung | 58 |
| 4.2.2 | Fehleranalyse im Versuch | 63 |
| 4.3 | Fehleranalyse | 64 |
| 4.4 | Zusammenfassung | 64 |
| 5 | Diskussion | 67 |
| 5.1 | Verbesserung | 67 |
| 6 | Abbildungen | 69 |
| | Literaturverzeichnis | 81 |

1 Einführung

1.1 Intension

Die Grundlage für erfolgreiches Lernen ist die Aufmerksamkeit der Schüler. Daher ist sie für die Qualität des Unterrichtes ausschlaggebend. Das Verhalten eines Schülers kann eingeteilt werden in on-Task (aufmerksam bei der Sache) und off-Task (unaufmerksam). Allerdings ist das erfassen der Aufgabe zugewandten Aufmerksamkeit recht schwierig und verschiedene Erfassungsmethoden versuchen dies zu erreichen. Ein Vorschlag von Ehrhardt, Findeisen, Marinello und Reinhartz-Wenzel (1981) umfasst Beispielsweise die Parameter Blickrichtung, Körperhaltung und Tätigkeit.

Zur Erfassung werden z.B. Fragebögen eingesetzt, die Schüler und Lehrer selbst ausfüllen oder es gibt ein Beobachter der die Aufmerksamkeit einzelner Schüler anhand von Kriterien bewertet.

Für „Das Münchener Aufmerksamkeitsinventar (MAI)“[13] wurden beispielsweise die Kategorien „*ON-TASK, reaktiv/fremd-initiiert: der Schüler reagiert auf eine entsprechende Aufforderung oder Frage des Lehrers*“ oder „*OFF-TASK - aktiv, interagierend, störend: Der Schüler nimmt die Lerngelegenheit nicht nur nicht wahr, sondern ist erkennbar anderweitig engagiert*“, festgelegt. Um das Verhalten eines Schülers zu bewerten wird dieser 5s lange beobachtet und einer Kategorie zugeordnet.

Bei der „Videostudie zur Wirksamkeit des Unterrichtsprozesses“[1] wurden die Kriterien „*Blickkontakt zum legitimen Sprecher oder Objekt, Aktive Beteiligung an der Aufgabe, keine Ausübung anderer Tätigkeiten, keine Motorische Unruhe und keine themenferne Kommunikation*“, festgelegt. Dann wurde der Schüler in einem ein Minuten-Intervall beobachtet und bewertet. Sind drei oder mehr Kriterien erfüllt, gilt der Schüler als on-Task (Aufmerksam).

Bei dieser Art der Auswertung gibt es allerdings Interpretationsfreiheiten, die von jedem Beobachter anders ausgelegt werden können. So kann das Drehen des Stiftes als motorische Unruhe oder nur als Zeichen von Nervosität bewertet werden. Außerdem ist diese Art der Auswertung sehr zeitintensiv, alleine eine einzige Beurteilung jedes einzelnen Schülers einer Klasse, etwa 30 Personen nach Vorgabe der Klassenbildung [4], benötigt mindestens 30 Minuten. Somit kann eine Auswertung aller Schüler während einer Unterrichtsstunde schnell 15 und mehr Arbeitsstunden dauern. Um eine subjektive Bewertungen zu vermeiden, sollte außerdem ein beträchtlicher Teil der Daten von mindestens zwei Beobachtern parallel ausgewertet werden, um deren Übereinstimmung beurteilen zu können, was noch mehr Arbeit bedeutet.

Basiert die Auswertung auf wenigen Zeitintervalle um Arbeitszeit zu sparen, wird das gesamte Verhalten eines Schülers während des Unterrichts mit nur wenigen beobachteten Minuten beschrieben und entsprechend ungenau. Somit können sowohl quantitativ genaue, als auch

temporal hochauflösende Daten nicht erstellt werden.

So kann bei grob gewählten Auswertungsintervallen nur eine Aussage über den gesamten Unterricht gemacht werden und nicht beispielsweise über einzelne Übungen oder über einen einzelnen Schüler.

1.2 Problemstellung

Im Rahmen dieser Arbeit sollen die Grenzen aufgezeigt werden, wie weit es technisch möglich ist Filmmaterial einer einzigen Kamera im Bezug auf Blickrichtungen bzw. Ausrichtung des Gesichts Auszuwerten und mit welchen Einschränkungen und Genaugkeiten zu rechnen sind, wenn im Bild eine gesamte Klasse dargestellt ist.

Eine automatisierte Auswertung der Blickrichtung wäre erstrebenswert, da dies einer der wichtigsten Indikatoren für gerichtete Aufmerksamkeit ist. Ist dies nicht möglich, reicht eine Bestimmung der Kopforientierung aus, da diese in etwa der Blickrichtung entspricht.

Die Messung soll den Unterricht möglichst wenig beeinträchtigen, wodurch hierfür üblicherweise verwendete Geräte, wie z.B. Eye-Tracking Brillen, nicht verwendet werden können. Zum einen ist die Anschaffung einer großen Stückzahl dieser Geräte teuer und wurde bisher nur in wenigen speziell eingerichteten Laboratorien durchgeführt wie z.B. TüDiLab [35]. Zum anderen sind die Geräte entweder Ablenkend (Brillen) oder schränken den Aktionsradius ein (Remote Tracker).

Die hier bestimmten Grenzen ergeben Anhaltspunkte, wie das Setup (Anzahl und Position der Kameras und deren Auflösung) für ein größeres Experiment aussehen muss, um die Aufmerksamkeit einer ganzen Klasse zu erfassen. Wären man in der Lage, solch eine qualitativ hochwertige Auswertung mit nur wenigen Kamera durchführen zu können, so ist der Aufbau und die Aufnahmen der Daten auch für technische Laien durchführbar.

Eine Möglichkeit für das automatische Erfassen der Aufmerksamkeit wird in „Real time detection of driver attention“[11] vorgestellt. Bei diesem Verfahren ist eine Kamera direkt von vorn auf den Fahrer gerichtet und anhand der Kopf und Augenposition bewertet, ob dieser aktiv auf den Verkehr achtet.

Ein weiteres dazu passendes Verfahren wird in „AggreGaze“[39] präsentiert, dabei wird eine einzige Kamera fest auf einem Bildschirm montiert, um die Blickrichtung der Passanten auf den Bildschirm zu bestimmen, dieses Verfahren arbeitete allerdings nur auf einem recht begrenzten Bereich in dem sich die Probanden aufhalten dürfen und das Ziel der Blicke ist sehr nahe an der Kamera.

Um die Machbarkeit der Analyse zu untersuchen, wurden verschiedene Videoaufnahmen ausgewertet. Unter anderem zwei Originalaufnahmen eines Englischunterrichtes, diese zeigen die gesamte Klasse aus Richtung der Tafel und liefern Eindrücke über die verschiedenen Probleme. Allerdings besitzen sie nur eine sehr geringe Auflösung (640×480 Pixel).

Für die Vorversuche wurde eine Actioncam verwendet um erste Eindrücke, bezüglich der Auswirkung von Position und Zielpunkt auf die Auswertung zu erhalten.

Um mehr Messwerte für unterschiedlichen Zielpunkte zu erhalten wurde ein weiterer Video-Datensatz mit der Logitech-Webcam erstellt, bei der die Probanden ein bewegtes Ziel be-

1.2 Problemstellung

obachten sollten. Damit besser bewertet werden kann, wie mit dem Verfahren das Ziel der Aufmerksamkeit bestimmt werden kann.

Außerdem soll die Anwendung auch auf schon vorhanden Aufnahmen des Englischunterrichtes arbeiten.

2 Stand der Forschung

2.1 Computer Vision Methoden zur Gesichtsanalyse

Gesichtserkennung ist eine der fortschrittlichen Verfahren in der maschinellen Bildverarbeitung und wird ständig weiterentwickelt. Darunter fallen neben der Detektion des Gesichtes auch deren Analyse wie Orientierung, Übereinstimmungen oder das Erkennen von Mimik wie beispielsweise das Lächeln von Personen zum Auslösen einer Kameras.

Bei vielen Anwendungen ist der heutige Stand der Technik die Verwendung eines Neuronalen Netzes.

2.1.1 Künstliches neuronales Netz

Ein künstliches neuronales Netz besteht aus miteinander verknüpften künstlichen Neuronen. Jedes Neuron besitzt Eingangswerte und einen Ausgabewert.

Um die Ausgabe zu bestimmen, werden die einzelnen Eingangswerte des Neurons individuell Gewichtet, mit einer Übertragungsfunktion zusammengefasst und mittels einer Schwellenwertfunktion das Ergebnis bestimmt.

Um die Parameter (Gewichtung und Funktionen) des Neurons zu bestimmen, werden diese zufällig initialisiert und anschließend so angepasst, dass es zu einer gegebenen Eingabe das gewünschte Ergebnis liefert und der Fehler über dem gesamten Trainingsdatensatz minimal wird.

Soll ein gesamtes Netz trainiert werden, so wird jedes einzelne Neuron zufällig Initialisiert und anschließend so angepasst das der Fehler des Netzes auf dem Trainingsdatensatz minimal wird.

[16]

2.1.2 Convolutional Neural Network (CNN)

Die CNN definieren in vielen Anwendungsbereichen momentan den Stand der Technik. Sie sind eine Weiterentwicklung der neuronalen Netze und werden vor allem im Bereich Klassifizierung eingesetzt werden, unter anderem bei der Bild- und Spracherkennung. Der Unterschied liegt bei der Verwendung von gewichteten Faltungen der Eingabe.

Durch die Faltung werden die Information aus den umliegenden Punkten eines Bereiches zusammengefasst und komprimiert an die nächste Schicht weitergegeben, um in der untersten

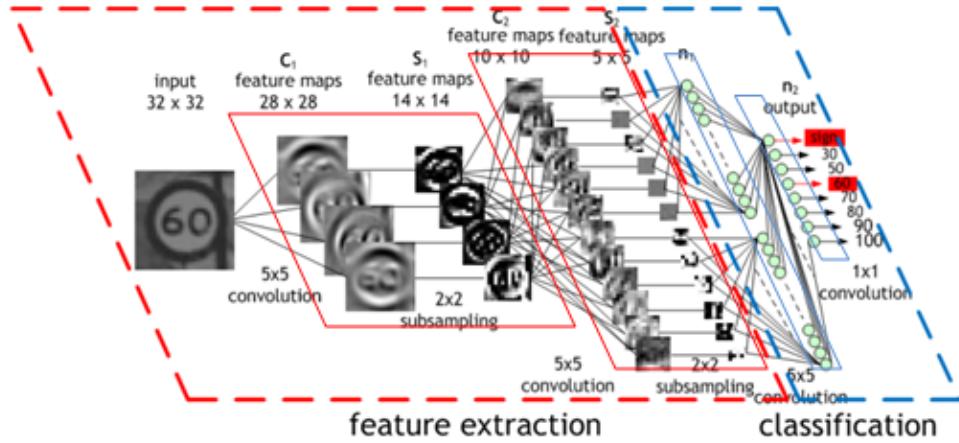


Abbildung 2.1: Beispiel für den Aufbau eines CNN zur Klassifizierung.

Zu sehen ist das Erkennen einer Zahl auf einem Straßenschild.[21]

Schicht alle vorhanden Informationen zusammenzuführen. Der Faltungskern kann je nach Anwendung beliebig gestaltet sein, so ist eine Glättung durch einen Gauß-Kernel oder Kanten-detection durch einen Kirsch-Operator möglich.

Ein CNN kann in zwei Bereiche aufgeteilt werden, Feature Extraktion und Klassifizierung. Bei der Feature Extraktion werden verschiedene Kernel und Komprimierung auf den Eingabe-informationen angewendet um sie für den zweiten Teil der Klassifizierung aufzubereiten. Dort wird nun die Eingabe ausgewertet um das Ergebnis zu erhalten.

Gelernt werden kann jeder einzelne Kernel für sich und die jeweiligen Bewertungen der Kernel und Neuronen.

[22][32]

2.1.3 Constrained Local Model (CLM)

Dies ist ein Verfahren um mehrere Punkte eines Objektes zu lokalisieren. Dabei wird eine Wahrscheinlichkeitskarte bezüglich der Position jeden einzelnen Punkt erstellt, wo dieser sich aufhalten kann, basierend auf einem Trainingsdatensatz. Auf dem Eingabebild wird nun eine weitere Wahrscheinlichkeitskarte für jeden Punkt erstellt, die die Ähnlichkeit der Darstellung angibt. Nun wird Versucht, für jeden Punkt den maximalen Wert zu erreichen zwischen passendem Farbverlauf und Wahrscheinlichkeit basierend auf der Position alle Punkte. Dieser Art der Bestimmung von positionsabhängigen Punkten ist ziemlich zuverlässig und dennoch dynamisch genug um auch mit kleinen Veränderungen klar zu kommen.

Dies ist Wichtig bei der Detektion von leicht verformbaren Objekten wie Beispielsweise Gesichter und ist zuverlässiger als das Active Appearance Model (AAM).

[7]

2.1.4 Constrained Local Neural Fields (CLNF)

Dabei handelt es sich um einen Gesichtsdetektor. Für die Detektion wird für jedes Merkmal ein eigener Detektor eingesetzt der auf einem Bildbereich arbeitet und eine Wahrscheinlichkeitskarte für dieses Merkmal erstellt.

Als nächster Schritt wird das Ergebnissen der Detektoren mit einer Karte der Position aller Landmarks mit ihren jeweiligen Abweichungen, kombiniert um somit die beste Position der Landmarks zu erhalten im Bezug auf den Farbverlauf und dem Verhältnis zu den anderen Landmarks.

[24]

2.1.5 Patch Experts

Das Patch Experts ist ein Bewertungsverfahren um die Wahrscheinlichkeit zu ermitteln, dass ein Landmark an einer bestimmten Stelle im Bild dargestellt wird. Für die Bestimmung wird ein ganzer Bereich um die Position ausgewertet um auch auf Teilen eines Pixels rechnen zu können.

[24]

2.1.6 Active Appearance Model (AAM)

Dies ist ein Verfahren der Bildverarbeitung um Übereinstimmungen zu einem Modell zu finden. Dazu wird aus dem Trainingsdatensatz eine typische einheitliche Form des Objektes generiert mit seinen signifikanten Landmarks.

Soll nun zu einem Eingabebild die Übereinstimmung ermittelt werden, wird zuerst versucht es bestmöglich mittels Transformation in die typische einheitliche Form zu überführen. Sind dennoch Unterschiede vorhanden, liegt diese an der Erscheinung des Objektes.

[27]

2.1.7 Non-maximum suppression (NMS)

Das NMS ist ein Verfahren um ein lokales Maximum zu bestimmen und kann z.B. in einem Bild eingesetzt werden um Kanten exakter zu bestimmen. Als Eingabe für das Verfahren im Beispiel, wird das Ergebnis eines Kantendetektor z.B. Kirsch-Operator verwendet. Dabei gibt die Höhe des Farbwertes eines Pixels an, wie nahe es an einer Kante im Originalbild liegen. Bei der Verarbeitung wird nun der Farbwert jedes einzelnen Pixels des Eingabebildes mit seinen umliegenden verglichen und sollte es nicht maximal sein auf Null gesetzt.

Auf diese Weise bleibt nur noch ein Kantenpixel übrig. Wird das Verfahren auf die Bestimmung von Boxen eingesetzt, so wird jene Fläche bestimmt die von allen am ehesten beschrieben wird.

[20][29]

2.1.8 Point Distribution Model (PDM) & Generalized Adaptive View-based Appearance Model (GAVAM)

Mit Point Distribution Model (PDM) können verformbare Objekte recht gut modelliert werden. Dabei wird die durchschnittliche Form \bar{X} des Objekts anhand der Eingabe bestimmt und eine Matrix P von Eigenvektoren ermittelt, um die möglichen Deformierungen darzustellen.

$$X = \bar{X} + P \cdot b$$

Somit kann durch einen Skalierungsvektor b alle möglichen Eingabeformen X des Objektes aus dem Durchschnittsmodell wiederhergestellt werden. Zur Vereinfachung reicht es, die signifikantesten Eigenvektoren in P aufzunehmen und dennoch X ausreichend genau beschreiben zu können.

Ist bekannt welche Art der Verformung durch den Eigenvektor dargestellt wird, z.B. eine bestimmte Orientierung, so kann anhand des Skalierungsvektors die Rotation der Eingabe bestimmt werden, siehe Generalized Adaptive View-based Appearance Model (GAVAM). Eine Problematik bei dieser Art der Rotationsbestimmung entsteht, wenn neben der Verschiebung der Landmarks durch die Rotation, auch eine Deformierung des Objektes stattgefunden hat und somit keine eindeutige Lösung gefunden werden kann. Dieses Problem tritt bei Berechnungen von Gesichtern auf, da immer eine Veränderung der Mundwinkel oder Augenlider vorhanden ist.

[18][19][34]

2.2 Eye-Tracking in der Klassenzimmer-Umgebung

Die Anwendung ist für den Unterricht ausgelegt, wie in der Abschnitt 1.2 beschrieben. Ein deutsches Klassenzimmer soll laut Baden-Württembergischen Schulbauempfehlungen eine Grundfläche von $54 - 66m^2$ aufweisen für maximal 28-32 Schüler [17].

Soll mit einer einzigen Kamera alle Schüler auf einmal beobachtet werden, dann muss diese den gesamten Bereich, in dem sich Schüler aufhalten können, erfassen. Abgeleitet aus der Grundfläche und Abzuglich der Bereiche für Tafel, Schränke und weitere Einrichtung beginnt dieser etwa $2,5m$ vor der Kamera und geht bis zu $8m$, auf einer Breite von $6m$, wenn sich die Kamera zentral an der Wand der Tafel befinden. Somit muss der Linsenwinkel mindestens 100° betragen mit entsprechender Schärfentiefe, damit ab einer Distanz von $2,5m$ ein Bereich von $6m$ Breite erfasst werden kann.

Der Unterricht soll durch die Messung möglichst wenig beeinflusst werden, somit ergeben sich folgende Randbedingungen:

- Brillen, Kontaktlinsen und ähnliches sind bei den Probanden erlaubt, ebenso beliebige Frisuren, Make-up usw.
- Die üblichen Bewegungen im Unterricht wie Sprechen, Kopfdrehungen usw. der Schüler sind gestattet.

2.2 Eye-Tracking in der Klassenzimmer-Umgebung

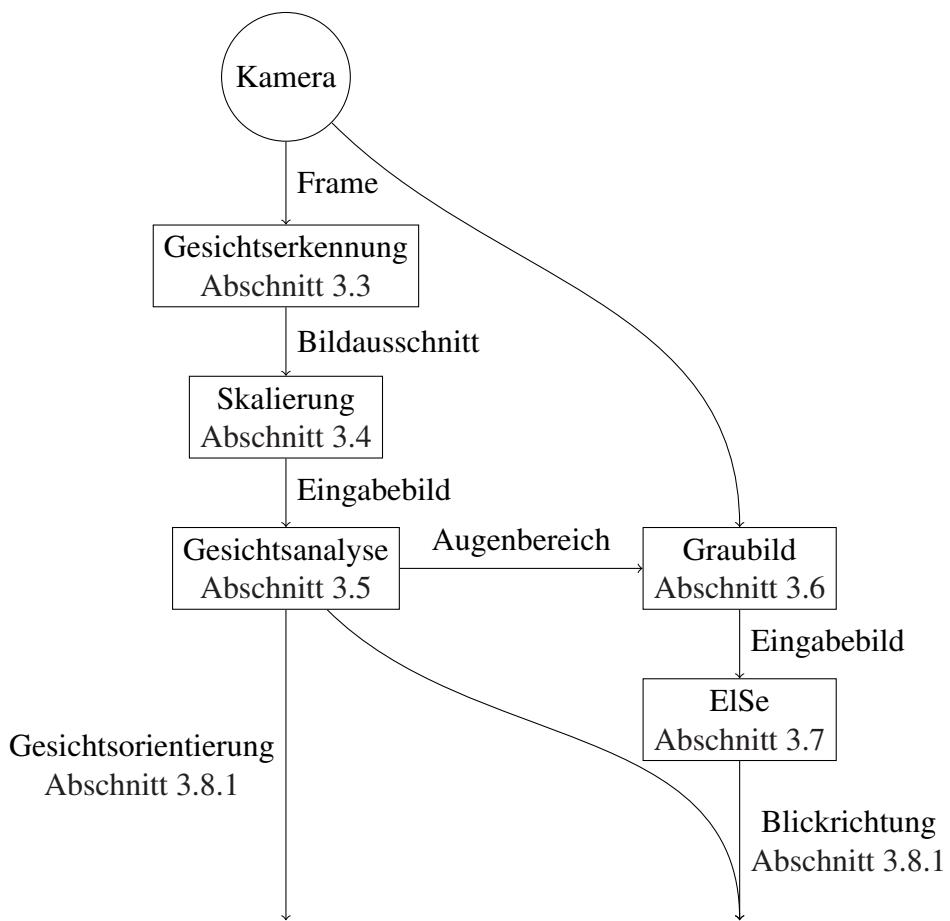
- Das Verfahren soll gleichzeitig auf Distanzen von $2,5 - 8m$ zur Kamera auf einer Breite von $6m$ funktionieren.
- Es werden keine Markierungen oder ähnliches an den Schülern angebracht, noch werden die Probanden exakt ausgemessen.

Für die Anwendung werden zusätzlich folgende Annahmen gemacht, die sich vor allem auf die Sitzordnung der Schüler sowie die Umgebung beziehen.

- Die Szene ist innerhalb eines Gebäudes, mit ausreichend gleichmäßiger Beleuchtung.
- Aus messtechnischer Sicht wäre die ideale Position der Kamera im Zentrum vor der Klasse, so dass die Hauptblickrichtung der Schüler in Richtung Kamera verläuft.
Diese Stelle kann jedoch nicht verwendet werden da diese Position für den Unterricht (Tafel/Lehrer) benötigt wird.
- Die Gesichter sind komplett sichtbar und nicht verdeckt durch andere Schüler oder von der Kamera abgewandt. Eine Sitzordnung, wie sie hauptsächlich im Frontalunterricht vorkommt.
- Möglichst alle Blickrichtungen bzw. die Gesichtsorientierung der Schüler sollen so exakt wie möglich erfasst werden.
- Die Überführung zwischen Welt- und Kamerakoordinatensystem ist bekannt.

3 Herangehensweise

3.1 Ablauf der Implementierung



Da nur eine einzige fest montierte Kamera ohne Zoom eingesetzt wird, muss diese eine entsprechend hohe Auflösung besitzen, damit alle Personen zu erkennen sind. Zur Bestimmung der Blickrichtung sowie Kopfposition und Orientierung wird ein mehrstufiges Verfahren eingesetzt um alle Teilprobleme zu lösen.

Am Anfang müssen alle Gesichter, die im aktuellen Frame vorhanden sind, detektiert werden da nur auf diesen eine Berechnung ausgeführt wird. Dabei machen die relevanten Bereiche nur einen sehr geringen Anteil des gesamten Bildes aus. Dazu wird die MTCNN Face Detection eingesetzt, siehe Abschnitt 3.3. Dieses Verfahren machte im Vorabtests auf Probebildern einen

3 Herangehensweise

sehr guten Eindruck und konnte die meisten Gesichtern mit verschiedenen Größen und Blickrichtungen finden.

Für die weiteren Berechnungen muss bekannt sein, welcher Bereich von einem Gesicht im Frame eingenommen wird, um die relevanten Bildausschnitte aufzubereiten. Dabei muss das gesamte Gesicht in der Box sein, weitere Besonderheiten gibt es nicht, da OpenFace einen eigenen Facedetector besitzt. Je nach verwendetem Trainingsdatensatz und darin enthaltener Annotation werden z.B. Kinn und Haaransatz noch als Gesichtsbereich oder schon als außerhalb betrachtet. So geben beiden Methoden (OpenFace und MTCNN-Face) Boxen aus, diese sind in ihren Ausmaßen allerdings nicht identisch. Da die folgende Verarbeitung eine OpenFace-skalierte Box erwartet, hat sich eine Vergrößerung der MTCNN-Face Box um 30% als sinnvoll erwiesen, um Ungenauigkeiten bezüglich der Position und Dimension des Kopfes im Bild entgegen zu wirken.

Sind mehrere Gesichter in mehreren Frames des Videos abgebildet, so muss auch eine Identitätszuordnung vorgenommen werden, damit bekannt ist welches Gesicht in Bild 1 welches in Bild 2 entspricht. Für die Zuordnung reicht es meist aus, jene Box zu wählen, die am ehesten den selben Bildausschnitt repräsentiert wie im vorigen Frame, da sich die Gesichter meist weder groß Bewegen noch sich die einzelnen Boxen der Probanden überlappen.

Damit sicher auf allen Gesichter gerechnet werden kann, ist eine semiautomatische Korrektur erforderlich um Falsch-Detektionen zu entfernen und fehlende Boxen der Gesichtern ergänzen zu können. Die gefundenen 5 Landmarks von MTCNN-Face Detection sind für die nachfolgende Berechnung nicht relevant, da sie gerade bei kleinen Gesichtern zu ungenau sind um sie zu verwenden. Daher können alle bisher unternommenen Schritte auch von anderen Verfahren übernommen werden, da es sich hierbei nur um ein Vorverarbeitungsschritt handelt und zur Beschleunigung sowie Stabilität des späteren Berechnung beitragen soll.

Damit das Verfahren im nächsten Schritt zuverlässig arbeiten kann, werden alle zu kleinen Bildbereiche hochskaliert, um die Gesichter auf eine Mindestgröße zu bringen, siehe Abschnitt 3.4. Diese Bildbereiche werden nun von OpenFace weiterverarbeitet um die Landmarks, die signifikanten Punkte eines Gesichtes, zu bestimmen. Durch die vorige Zuordnung der Gesichert kann das Verfahren gezielt auf den einzelnen Person arbeiten und ein entsprechend eingestelltes CLNF verwenden, um bessere Ergebnisse zu erzielen, siehe Abschnitt 3.5. Außerdem könne alle gefundenen Personen gleichzeitig (parallel) ausgewertet werden. Für dem im nächsten Schritt verwendetem ElSe Algorithmus, muss der Bildausschnitt des Auges in ein Graubild umgewandelt werden, siehe Abschnitt 3.6.

Um die Position der Pupille noch exakter zu ermitteln wird ElSe verwendet, da durch eine exakte Bestimmung der Pupillenposition, auch eine genaue Blickrichtungsbestimmung möglich ist, siehe Abschnitt 3.7.

Nun wird auf Basis der Landmarks und Kameraparameter die Position und Orientierung der Gesichter sowie die Blickrichtung bestimmt, siehe Abschnitt 3.8. Diese Ergebnisse können dann von weiteren Anwendungen verwendet werden.

3.2 Verwendete Eingabegeräte & Bibliotheken

Als Messinstrument für die Versuche wurden verschiedenen Farbkameras eingesetzt.

Für die Messungen im Versuch wurde die Explorer 4K Actioncam verwendet, sie besitzt ein 170° Weitwinkel-Linse mit großer Schärfentiefe. Mit ihrer 2.7K Einstellung wird ein 2688×1520 Video mit 30FPS aufgezeichnet. Leider ist das Bild stark von Pixelrauschen betroffen.

Außerdem die Logitech c920 HD Pro Webcam, diese liefert ein 15Fps Video mit einer Auflösung von 1600×896 Pixel. Die Kamera besitzt einen horizontalen Blickwinkel von etwa 70° .

Das Videomaterial der Schulkasse wurde mit einer unbekannten Videokamera aufgezeichnet, daher sind nur die Parameter des Filmes (640×480 Pixel mit 25Fps) bekannt.

Für die Umsetzung wurden Open Source Computer Vision (OpenCV 3.1) verwendet. Dies ist eine C/C++ Bibliothek von Algorithmen zur Bildverarbeitung in Echtzeit, veröffentlicht unter der BSD Lizenz (Berkeley Software Distribution)

[3][33]

3.3 MTCNN Face Detection

Multi-task Cascaded Convolutional Networks (MTCNN) ist ein Algorithmus zur Detektion von Gesichtern und Bestimmung von 5 Gesichts-Landmarks in Farbbildern. Dabei werden drei CNN auf einer Bildpyramide angewendet um zuverlässig Gesichter verschiedenster Größe zu erkennen. Außerdem wird für die Detektion der Gesichter auch deren Ausrichtung berücksichtigt, um bessere Ergebnis zu erzielen. Laut Beschreibung des Verfahrens sollen sogar recht kleine Gesichter mit 20×20 Pixeln erfassbar sein.

Sein Einsatzgebiet ist die Vorverarbeitung eines Frames für die spätere Auswertung. Somit soll dieser Schritt von einem möglichst robusten Verfahren durchgeführt werden. Dabei wird im aktuellen Fall auf einem hochauflösendem Bild gearbeitet mit verhältnismäßig kleinen, verschieden Großen und weit verteilten Gesichter.

3.3.1 Die 3 Stufen der Verarbeitung

Für die gute Detektionsqualität sorgt die dreistufige Verarbeitung mit verschiedenen CNN auf einer Bildpyramide. Bei der Bildpyramide handelt es sich um ein in verschiedenen Größen skaliertes Bild, damit der gesuchte Inhalt in der gewünschten Auflösung abgebildet ist, ohne etwas über den Inhalt zu wissen.

Dies ist von Vorteil, damit das CNN auf eine feste Größe von Gesichtern optimiert werden kann, um das Lernen nicht zusätzlich zu erschweren. So werden nur die Farbverläufe gelernt und nicht weite durch die Skalierung erschwert, wodurch das CNN auf seine jeweilige Aufgabe besser optimiert werden kann.

3 Herangehensweise

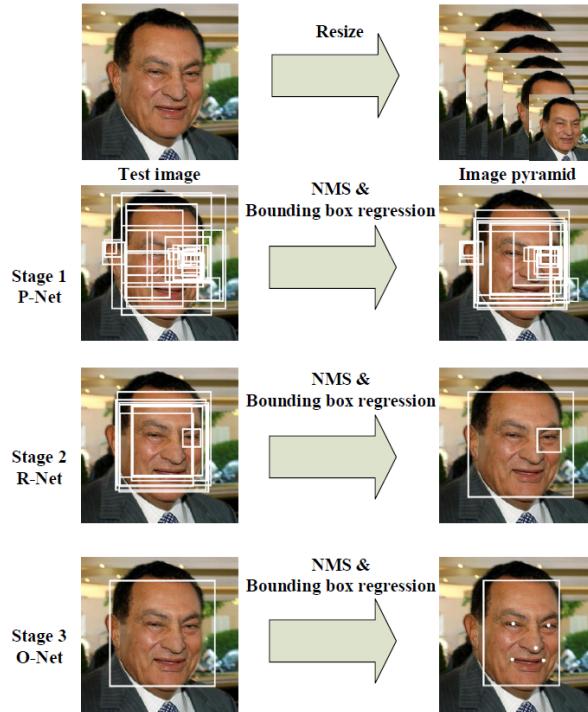


Abbildung 3.1: Darstellung des Funktionsablaufes von MTCNN[15]

Stufe 1

Beim ersten Verarbeitungsschritt werden alle Bereiche eines Bilds gesucht, in denen möglicherweise ein Gesicht zu erkennen ist. Dazu wird für die Detektion ein CNN eingesetzt, dem sogenannten Proposal Network (P-Net), das alle möglichen Bounding-Boxen ermittelt in denen ein Gesicht zu sehen sein könnte. Diese Bounding-Boxen werden anschließend mit einem NMS ausgedünnt, um die am stärksten überlappenden Boxen zusammen zu fassen. Dies ist notwendig, da dieses CNN zwar recht schnell arbeitet, allerdings auch mit einer sehr großen False-True-Fehlerrate (Erkennen trotz nicht vorhanden).

Stufe 2

Die möglichen Bereiche aus Stufe 1 werden anschließend mittels eines weiten CNN analysiert, damit alle Nicht-Gesichtsbereiche erkannt und entfernt werden können. Dies wird von dem Refine Network (R-Net) übernommen und anschließend die möglichen Bounding-Boxen mittels NMS noch weiter reduziert.

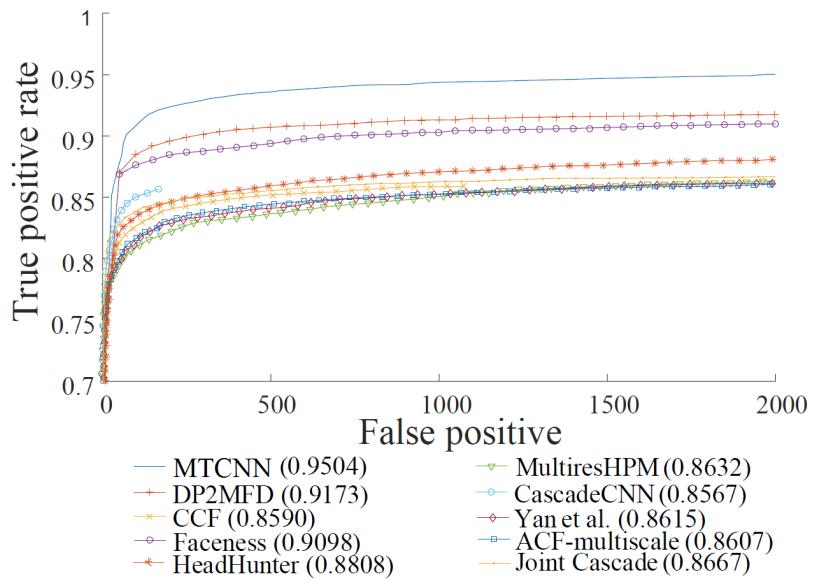


Abbildung 3.2: Qualität der Detektion der verschiedenen Verfahren im Vergleich [15]

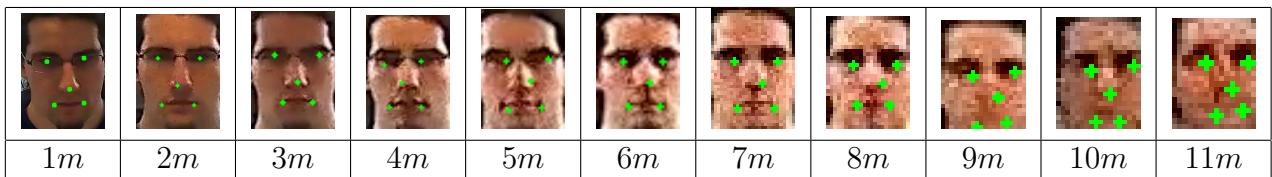


Abbildung 3.3: Dargestellt ist die Box und die 5 Landmarks von MTCNN-Face bei verschiedenen Distanzen des Probanden zur Actioncam

Stufe 3

Der letzte Schritt wird von einem deutlich genaueren CNN übernommen, um ein Gesicht zu detektieren, dem sogenannten Output Network (O-Net). Womit die resultierenden exakten Boxen mit ihren jeweiligen 5 Landmarks ermittelt werden.

3.3.2 Zuverlässigkeit bei der Detektion

MTCNN Face Detection ist bei der Zuverlässigkeit im Vergleich zu anderen bekannten Verfahren überlegen, siehe Abbildung 3.2, und zudem auf 640×480 großen Bilder Echtzeit fähig, dabei können auch sehr kleine Gesichter erfolgreich erkannt werden.

Somit sind alle Anforderungen erfüllt um mit diesem Verfahren den vorhanden Frame für die nachfolgenden Berechnungen vorzubereiten. Ein Test bestätigt diese Annahme, siehe Abbildung 3.3.

3.4 Aufbereitung der Bilder

OpenFace arbeitet laut Angabe im Paper [25] am besten auf Gesichtern mit einer Mindestgröße von 100 Pixel, daher werden die Bildbereiche auf diese Größe gebracht. Dies ist notwendig, da die Berechnungen meist auf recht kleinen Bildausschnitten ausgeführt werden müssen.

Dabei ist es wichtig, dass die Gesichtsmerkmale möglichst gut rekonstruiert werden, um die entsprechenden Landmarks zu bestimmen, dabei erhöht sich der Informationsgehalt der Bilder nicht, sie sind nur besser nutzbar, da sie dem Trainingsdatensatz stärker ähneln.

Die von MTCNN gelieferten und vergrößerten Boxen werden auf eine Breite von 130 Pixel gebracht (100 Pixel für den Kopf mit 30% Rand durch Vergrößerung), damit das beinhaltete Gesicht auf der gewünschte Größe dargestellt wird. Neben der Skalierung des Bildausschnittes muss bekannt sein, wie Punkte im skalierten Bildausschnitt in das Frame überführt werden können, damit dies bei späteren Berechnungen berücksichtigt wird.

Der Skalierungsfaktor ist für jeden Bildausschnitt individuell und kann sich über die Zeit ändern, wenn sich z.B. die Distanz zwischen Person und Kamera ändert. Von einer zu starken Vergrößerung ist abzuraten, da sich der Rechenaufwand pro Gesicht erhöht und die Zuverlässigkeit der Berechnungen von OpenFace sinkt, z.B. durch Falschdetektion.

3.4.1 Bicubic-Skalierung

Der neue Farbwert wird ermitteln, indem die umliegenden 4×4 Pixelwerte betrachtet werden um den Farbverlauf als eine Funktion 3. Grades zu bestimmen. Somit werden feinere Details besser dargestellt als beim linearen Verfahren und Kanten bleiben eher erhalten. Allerdings kann es durch den bestimmten Verlauf auch zum Überschwingen kommen, wodurch Fehlfarben entstehen können. Ein Beispiel als Ergebnis dieses Verfahrens ist in Abbildung 3.4 zu sehen. [28]

3.4.2 Lanczos-Skalierung

Dieser Filter basiert auf einem bewerteten Durchschnitt der umliegenden Pixel um den neuen Pixelwert zu erhalten. Die Bewertung der einzelnen Pixel wird durch eine Sinc-Funktion bestimmt, damit weiter entferntere Pixel schwächer bewertet werden als näher liegende, siehe Abbildung 3.5.

Außerdem wird durch den Kurvenverlauf der Bewertungsfunktion eine gewisse Bildschärfe erreicht. Die Funktion kann und wird für die Anwendung auf einen 8×8 Pixel großen Bereich begrenzt.

[30]

$$L(x) = \begin{cases} \frac{\sin(\pi x)}{\pi x} \cdot \frac{\sin(\pi \frac{x}{a})}{\pi \frac{x}{a}} & \text{wenn } -a < x < a, a \neq 0 \\ 1 & \text{wenn } x = 0 \\ 0 & \text{sonst} \end{cases}$$



Abbildung 3.4: Die ursprüngliche Abbildung von Lena betrug 100 Pixel Kantenlänge und beim Schachbrett 48 Pixel, beide wurden mittels bikubischem Verfahren auf 512 Pixel vergrößert und bei Lena die Differenz zum originalen Lena-Bild bestimmt, siehe mittleres Bild

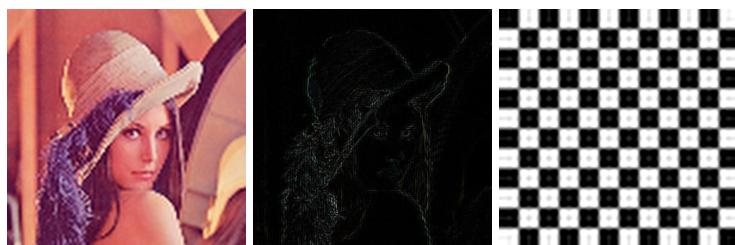


Abbildung 3.5: Die ursprüngliche Abbildung von Lena betrug 100 Pixel Kantenlänge und beim Schachbrett 48 Pixel, beide wurden mittels Lanczus-Verfahren auf 512 Pixel vergrößert und bei Lena die Differenz zum originalen Lena-Bild bestimmt, siehe mittleres Bild

3.4.3 Linear-Skalierung

Um den neuen Farbwert zu ermitteln, wird zwischen den nächstgelegenen umliegenden Pixel linear Interpoliert, wodurch weitere Farbwerte entstehen. Das Ergebnis ist gleichmäßiger als Neares Neighbor, und dennoch ein recht einfaches Verfahren. Die Kanten wirken allerdings unscharf, siehe Abbildung 3.6.

3.4.4 Nearest-Neighbor-Skalierung

Dieses Verfahren verwendet als neuer Farbwert, den gleichen Wert wie das nächstgelegene Pixel. Dadurch werden nur die ehemaligen Pixel größer und das Gesicht wirkt sehr Kantig, da keine neuen Farbwerte bestimmt werden, siehe Abbildung 3.7. Bei der Vergrößerung des Schachbretts sind kein Farbfehler aufgetreten, da nur zwei Farben vorhanden und Positionsabhängig sind.

3 Herangehensweise



Abbildung 3.6: Die ursprüngliche Abbildung von Lena betrug 100 Pixel Kantenlänge und beim Schachbrett 48 Pixel, beide wurden mittels linearer Interpolation auf 512 Pixel vergrößert und bei Lena die Differenz zum originalen Lena-Bild bestimmt, siehe mittleres Bild

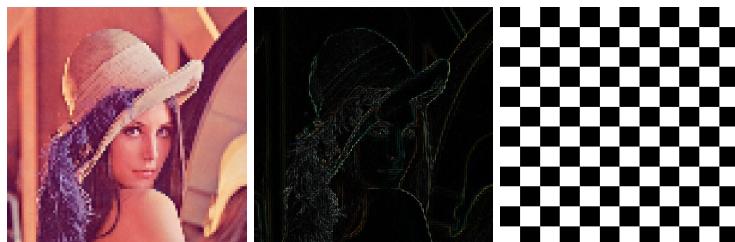


Abbildung 3.7: Die ursprüngliche Abbildung von Lena betrug 100 Pixel Kantenlänge und beim Schachbrett 48 Pixel, beide wurden mittels Nearest-Neighbor auf 512 Pixel vergrößert und bei Lena die Differenz zum originalen Lena-Bild bestimmt, siehe mittleres Bild

3.5 OpenFace

Ein Open-Source Echtzeitverfahren auf Basis von CLNF für die Bestimmung und Analyse von Gesichtsmerkmalen in Graubildern und Videos. Dabei stehen für diese Anwendung nur die Kameraparameter zur Verfügung und keinerlei Zusätze wie ein Tiefenbild (kann mitverwendet werden wenn es vorhanden ist) oder Infrarotbeleuchtung der Szene.

OpenFace kann 68 Landmarks ermitteln, die das Gesicht beschreiben, und mit deren Hilfe Position, Blickrichtung und Gesichtsmerkmale zu bestimmen. Sollte ein Video als Quelle fungieren, kann OpenFace auch lernen, wodurch eine zuverlässigere Verarbeitung erzielt werden kann.

Als Ergebnis ist die Kopfposition (Translation und Orientierung) sowie Blickrichtung von Interesse, da mit ihnen zurückrechnet werden kann woher die Person schaut.

Der Rechenaufwand zur Verarbeitung des Eingabebildes ist so ausgelegt, dass ein Webcam-Video in Echtzeit ausgewertet werden kann, dies ist im aktuellen Fall nicht notwendig, da es sich um eine nachträgliche Auswertung handelt, bei der es vor allem um Genauigkeit geht.

3.5.1 Bestimmung der Landmarks

Für die Bestimmung der Landmarks wird OpenFace auf den zuvor bestimmten Bildausschnitten eingesetzt. Dies bietet mehrere Vorteile, so wird nur auf Bildbereichen gearbeitet, in denen ein Gesicht zu sehen ist und unnötige Suche vermieden. Außerdem kann für jede Person die passende Initialisierung des CLNF, basierend auf dem letzten Ergebnis dieser Person, gewählt werden, auch für jene Personen die nur selten dargestellt sind. Auf diese Weise kann der Bildausschnitt möglichst exakt und gleichzeitig mit den anderen ausgewertet werden.

Für die eigentliche Bestimmung der Landmarks bietet OpenFace zwei verschiedene Methoden, die Berechnung auf Bildern und Videos. Der Hauptunterschied ist das Lernen, dass bei der Videoauswertung verwendet wird, wodurch sich der Toleranzbereich deutlich erhöht und bessere Ergebnisse liefert werden. Dies liegt an der Anpassung des Modells und dem möglichen Tracking der Landmarks.

Dies ist interessant für die spätere Anwendung, da somit auch Einzelbilder verwendet werden können, die eine deutlich höhere Auflösung besitzen als ein Video. Allerdings haben die Vorabtestes (Abschnitt 4.1), gezeigt, dass bei Verwendung von Einzelbilder der maximale Winkel relativ zur Kamera beträchtlich sinkt. Außerdem hat sich gezeigt das bei Verwendung eines Videos das Gesicht deutlich kleiner dargestellt sein kann bis keine Auswertung mehr möglich ist. Sollte ein Gesicht im aktuellen Frame erfolgreich detektiert werden, können auch die nachfolgenden Frames durch das Lernen ausgewertet werden.

Dennoch kann es passieren, dass trotz allem ein Gesicht falsch detektiert wird, wie z.B. das Erkennen eines sehr kleinen Gesichtes innerhalb einer Ohrmuschel. In solch einem Fall muss das CLNF zurückgesetzt werden, damit sich der Fehler nicht fortpflanzt.

3 Herangehensweise

Gesichts-Landmarks: Detektion und Verfolgung

Für die Bestimmung und Tracking der Landmarks wird ein Conditional Local Neural Fields (CLNF) eingesetzt. Dabei handelt es sich im Grunde um ein Constrained Local Model (CLM) nur mit verbesserter Patch Experts und Optimierungsfunktionen.

Die beiden Hauptkomponenten des CLNF von OpenFace ist das Point Distribution Model (PDM) zur Erfassung der Anordnung der Landmarks und Patch Experts zum Erfassen der Variante der einzelnen Landmarks.

Zu Beginn werden verschiedene initiale Hypothesen aus der dlib-Bibliothek verwendet und die Passende zur Eingabe ausgewählt. Bei den unterschiedlichen initial Hypothesen handelt es sich um die Darstellung verschiedener Gesichtsorientierungen auf denen unterschiedliche Netze trainiert wurden. Dieser Herangehensweise ist langsam, aber auch exakter als eine einfache Hypothese. Wird ein Tracing, das Verfolgen der Landmarks über mehrere Frames, durchgeführt wird als initiale Hypothese das Ergebnis aus der letzten Eingabe verwendet. Sollte das Tracing scheitern, wird das CNN reseted um Neu zu beginnen mit den ursprünglichen Hypothesen.

Auf diese Weise werden 68 Gesichts-Landmarks und weitere 28 pro Auge erfasst. Zur Berechnung auf den Gesichtern sollten diese laut Paper [25] eine Optimalgröße von 100 Pixeln für eine zuverlässige Detektion aufweisen.

Erkennen der Gesichtsmerkmale

Dieser Schritt kann von OpenFace ausgeführt werden, ist aber im aktuellen Fall nicht von Relevanz, da die Blickrichtung von Interesse ist und nicht die Mimik der Probanden.

Veröffentlichte Genauigkeit der Kopforientierung

Um die Qualität der Berechnung auf dem Kopf zu bewerten wurde im Paper [25] der „Biwi Kinect head pose“[10], „ICT-3DHP“[2] und „BU Datensatz“[5] ausgewertet. Dabei handelt es sich um Portrait-Fotos von Probanden, deren Körper in Richtung Kamera ausgerichtet sind und ihren Kopf in eine beliebige Richtung drehen. Für die Genauigkeit der Kopfposition haben sich Werte ergeben in Grad, siehe Abbildung 3.8.

Für die Qualität zur Bestimmung der Blickrichtung wurde der Augendatensatz „Appearance-based gaze estimation in the wild“[38] zur Bestimmung der Blickrichtung verwendet und es ergab sich ein durchschnittlicher Fehler von 9,96 Grad.

| | Yaw | Pitch | Roll | Mean | Median |
|-------------|-----|-------|------|------|--------|
| Biwi Kinect | 7.9 | 5.6 | 4.5 | 6.0 | 2.6 |
| BU dataset | 2.8 | 3.3 | 2.3 | 2.8 | 2.0 |
| ICT-3DHP | 3.6 | 3.6 | 3.6 | 3.6 | - |

Abbildung 3.8: Veröffentlichte Abweichung von OpenFace auf verschiedenen Datensätzen.[25]

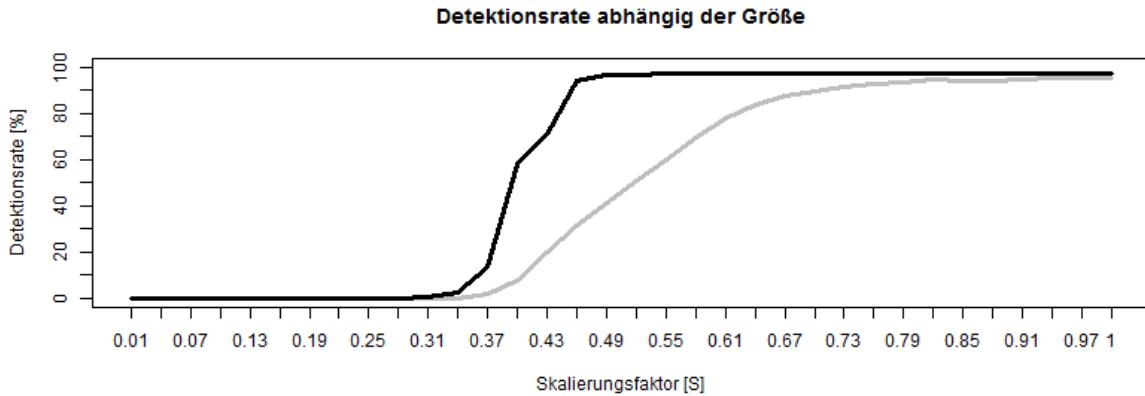


Abbildung 3.9: Die Bilder aus Labeled Faces in the Wild [14] (schwarz) und Biwi Kinect Head Pose Database [10] (grau) wurden mit den Faktor auf der X-Achse linear verkleinert und die Erkennungsrate Y-Achse abgebildet

3.5.2 Bestimmung des Arbeitsbereiches

Da mit diesem Verfahren die Landmarks bestimmt werden, aus denen die Gesichtsorientierung abgeleitet wird, sollen die Grenzen dieses Verfahrens ermittelt werden. Von Interesse ist die Bildqualität in der ein Gesicht dargestellt werden muss um dieses noch verarbeiten zu können und wie sehr diese Person von der Kamera abgewandt sein kann.

Auswirkung der Auflösung auf die Detektionsrate

Durch die Aufgabenstellung muss das Verfahren zuverlässig bezüglich der Distanzen bzw. Darstellungsgröße sein. Zur Messung wurde der Datensatz von Labeled Faces in the Wild [14] verwendet. In diesem Datensatz ergibt sich im Originalbild eine durchschnittliche Kopfbreite von 94 Pixel. Bei Random Forests for Real Time 3D Face Analysis [9] ist die durchschnittliche Breite 78 Pixel.

Zur Durchführung wurden die Bildgröße mit dem Skalierungsfaktor multipliziert und linear verkleinert um so kleinere, weiter entfernte Gesichter zu simulieren und anschließend mit dem Image-Detector von OpenFace verarbeitet. Das Ergebnis ist in Abbildung 3.9 dargestellt. Es ist zu erkennen, dass die Wahrscheinlichkeit auf eine erfolgreiche Detektion ab einer Skalierung von 0,6 bei BIWI auf dem, also Gesichert mit etwa 47 Pixel Breite, rapide abnimmt. Bei der in Abschnitt 3.2 beschriebenen Kamera entspricht dies einer Distanz von etwa 4,5m. Bei der maximalen Distanz auf der gearbeitet werden soll (8m) ergibt sich eine Gesichtsgröße von etwa 22 Pixel, das einer Skalierung von 0,25 entspricht. Bei dieser Bildgröße ist in der Standardanwendung ohne Skalierung keine Detektion möglich, siehe Abbildung 3.9.

3 Herangehensweise

Auswirkung der verschiedenen Skalierungsverfahren auf die Detektion

Um die Auswirkung der Skalierungsverfahren zu bestimmen, wurden verschiedene Gesichtsgrößen simuliert, indem sie um den angegebenen Faktor linear verkleinert wurden.

Beim Random Forest Datensatz [9] werden nur jene Bilder ausgewertet, in denen OpenFace bei einem Vorabtest ein Gesicht erkannte und nur der entsprechende Bildbereich ausgewertet. Als Zielgröße bei der Skalierung wurde das 1,3 fache der Originalgröße gesetzt, damit die abgebildeten Gesichter in etwa 100 Pixel groß sind für die Auswertung. Zur Berechnungen wurde eine Brennweite von 531,15 angenommen.

Bei dem Labeld Face in the Wild [14] Datensatz wurden alle Bilder im Orginal verwendet, um den angegebenen Skalierungsfaktor verkleinert und mit dem angegebenen Verfahren wieder auf die Orginalgröße gebracht.

Die Auswirkung der verschiedenen Skalierungsverfahren auf die Detektionswahrscheinlichkeit ist in Abbildung 3.10 dargestellt.

Es ist zu erkennen das die Detektionsrate über einen weiten Bereich, [1; 0,25] bei der Skalierung, nur sehr wenig abnimmt. Durch die Vergrößerung können somit jene Gesichter in Bereichen die normal nicht erkennbar sind, ausgewertet werden.

Erst bei den sehr kleinen Skalierungen ist ein wirklicher Unterschied zwischen den Verfahren zu erkennen. So nimmt die Detektionsrate bei Nearest-Neighbor (rot) deutlich früher ab als bei den anderen verfahren. Das Bicubic (blau) und Lanczos (grün) Verfahren haben die höchste Detektionsrate und fallen auch am spätestens ab, wobei Bicubic minimal besser ausfällt.

Es zeigt sich, dass durch die Skalierung die Anforderungen auf eine Detektion von Gesichtern mit 22 Pixel (Skalierung 0,25, 8m), von allen Verfahren erfüllt.

Ausgehend vom Skalierungsfaktor des Bicubic-Verfahren ab welchem die Detektionsrate stark abfällt, wären Distanzen bis zu 14m möglich (Basierend auf der Auflösung der Actioncam). Allerdings ist das Bild durch die Verkleinerung deutlich besser als bei Originalaufnahmen, da Pixelrauschen und Ähnliches nicht vorhanden ist.

Arbeitsbereich bezüglich Rotation

In Abbildung 3.11 ist der Median der Differenz zwischen den berechneten Winkel von Pose-World und der Angabe im Datensatz dargestellt.

Bei der X-Rotation zeigt sich, dass das Bicubic-Verfahren im Vergleich zu den anderen 1 Grad schlechter abschneidet. Der Fehler von Lanczos, Linear und Nearest-Neighbor liegt bei etwa 19° bis zu einer Skalierung von 0,25.

Es Zeigt sich, dass der Median in den Abweichung auf der Y-Achse (nicken) sehr hoch ausfällt mit knapp über 25°. Dabei liefern alle vier Verfahren nahezu identische Ergebnisse, die auch konstant bleiben bezüglich der Skalierung.

Die Z-Rotation wird am besten bestimmt mit einer Abweichung von etwa 7,5°, dabei ist aber auch zu beachten Wertebereich geringer als bei den anderen beiden Rotationen. Hier liefert Bicubic das Beste Ergebnis, wobei der Unterschied weniger als ein halbes Grad beträgt.

Für alle Berechnungen zeigt sich, dass der Fehler konstant bleibt, bis zu der Skalierung von 0.25, bei der auch Detektion scheitert.

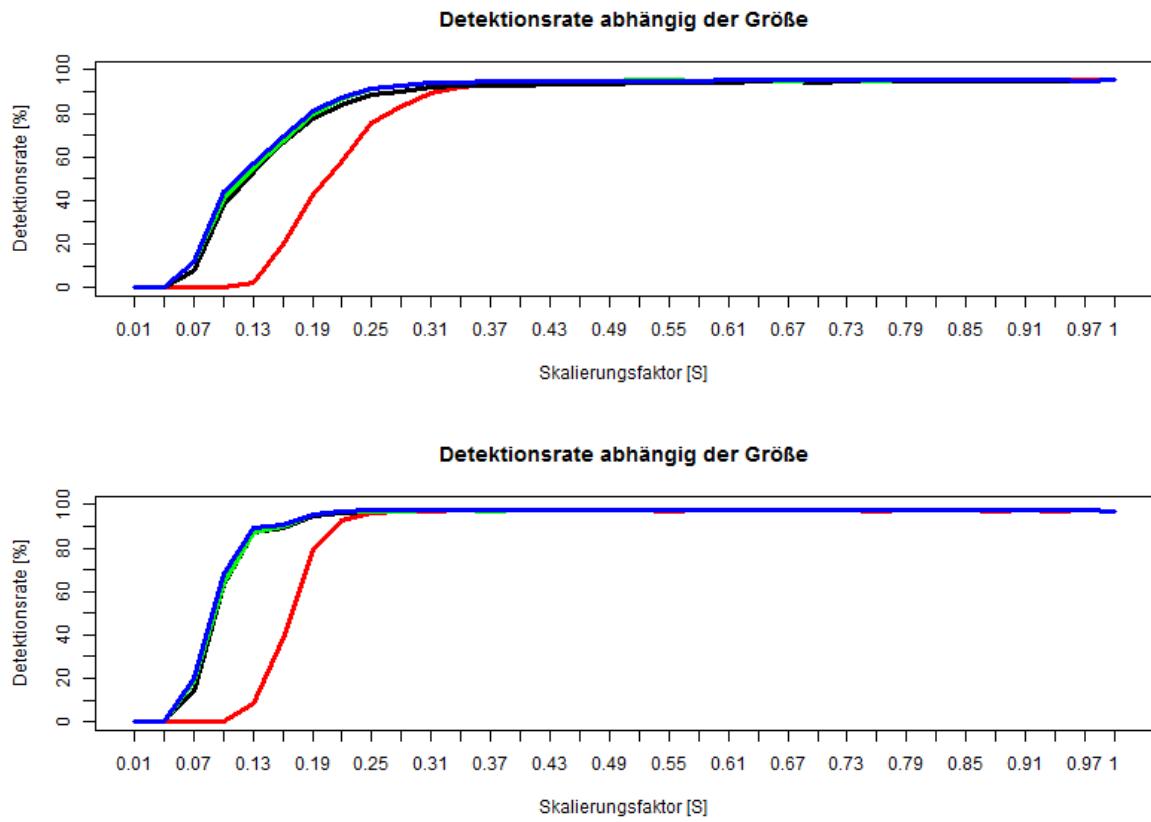


Abbildung 3.10: Die Bilder wurden mit den Faktor auf der X-Achse linear verkleinert und mit den verschiedenen Verfahren wieder vergrößert Abschnitt 3.4. Aufgetragen gegen die Y-Achse ist die Detektionswahrscheinlichkeit.

Bicubic (blau), Lanczos (grün), Linear (schwarz), Nearest-Neighbor (rot)

Oben: Biwi Kinect Head Pose Database [10]

Unten: Labeled Faces in the Wild[14]

3 Herangehensweise

Neben der Qualität von den bestimmten Winkel, ist auch der Arbeitsbereich von Interesse in dem Gesichter bei verschiedenen Skalierungen noch erkannt werden können. Da ein Gesicht das außerhalb dieser Bereiche liegt nicht erkannt und ausgewertet werden kann.

In Abbildung 6.1 sind die Quantile bei 50%; 80%; 99,5% und der Maximalwert, von den Rotationswinkel der Bilder aus dem Biwi Kinect Head Pose Database [10] ein Gesicht erkannt wurde, abgebildet. Durch den großen Unterschied zwischen den 80%-Wert, ;99,5%-Wert und dem Maximalwert legt die Vermutung nahe, dass es sich bei diesen Werten um falsch detektierte Bilder handelt, aber eine Rotation des Kopfes von 45% in alle Richtungen erkannt und ausgewertet werden kann.

Eine genaue Darstellung der Messung ist in Kapitel 6 abgebildet, für die X-Rotation Abbildung 6.5, Y-Rotation Abbildung 6.6 und Z-Rotation Abbildung 6.7.

Auswirkung der Skalierungsverfahren auf die Positionsbestimmung

Für eine zuverlässige Auswertung ist auch die Bestimmung der Position von Interesse. Im Biwi Kinect Head Pose Database [10] ist die Durchschnittliche Distanz zwischen Kamera und Proband bei etwas 90cm . Der Median der Differenz zwischen Datensatz und Rechnung ist in Abbildung 3.12 dargestellt. Bei sehr kleinen Skalierungen existieren durchaus auch sehr große Fehler, diese wurden allerdings bei der Darstellung abgeschnitten, da bei dieser Größe die Detektionsrate so klein ist, dass sie nahezu irrelevant werden.

Es zeigt sich, dass die Position in horizontaler und vertikaler Richtung auf etwa $6,5\text{cm}$ genau bestimmt werden kann, die Distanz (Tiefe) auf 9cm und auch auf sehr klein Skalierten Bildern. Nearest-Neighbor hat bei der Berechnung der X-Position die geringste Abweichung zu den anderen getesteten Verfahren mit $6,37\text{cm}$ bei der Originalgröße.

Bei der Bestimmung der Y-Position, liefern alle Skalierungsverfahren sehr ähnlich Ergebnisse mit einem Fehler von $6,5\text{cm}$, wobei das lineare Verfahren minimal besser ausfällt bei kleinen Skalierungen.

Die größte Ungenauigkeit liegt bei der Z-Position (Tiefe). Das Lineare Verfahren liefert auch hier das beste Ergebnis, mit einer Abweichung von $8,92\text{cm}$, dabei ist der Unterschied zu den anderen Verfahren minimal.

Eine ausführliche Darstellung der Messung ist in Abbildung 6.2, Abbildung 6.3 und Abbildung 6.4 dargestellt

Auswirkung von Pixelrauschen auf die Detektion

Mit diesem Test soll geprüft werden, welches der Verfahren auch stabil gegenüber Rauschen ist.

Um Pixelrauschen zu simulieren, wurden die Bilder aus Labeled Faces in the Wild [14] entsprechend verkleinert, mit Rauschen versehen um sie anschließend mit den unterschiedlichen Verfahren zu vergrößern.

Das Rauschen wird für jedes Pixel simuliert, indem eine Wahrscheinlichkeit von 50% besteht auf eine gleich verteilte Abweichung von $\pm 10\%$ des Originalen Farbwertes. Dieser Vorgang

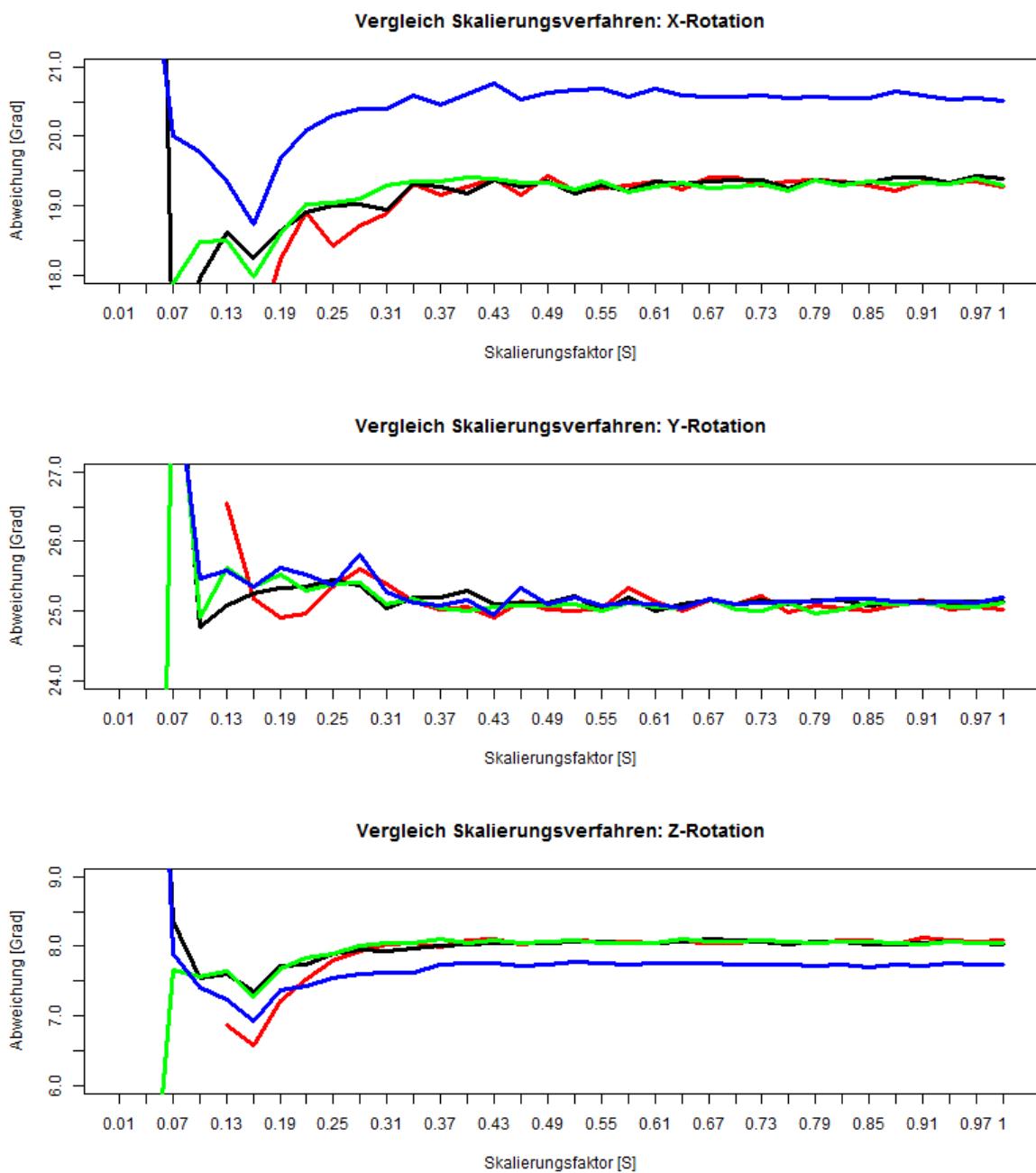


Abbildung 3.11: Dargestellt ist der Median der Abweichung zwischen der Berechneten Drehung und der des Datensatzes.

Bicubic (blau), Lanczos (grün), Linear (schwarz), Nearest-Neighbor (rot)

Oben: X-Rotation Wertebereich: $[18; 21]^\circ$

Mitte: Y-Rotation Wertebereich: $[24; 27]^\circ$

Unten: Z-Rotation Wertebereich: $[6; 9]^\circ$

3 Herangehensweise

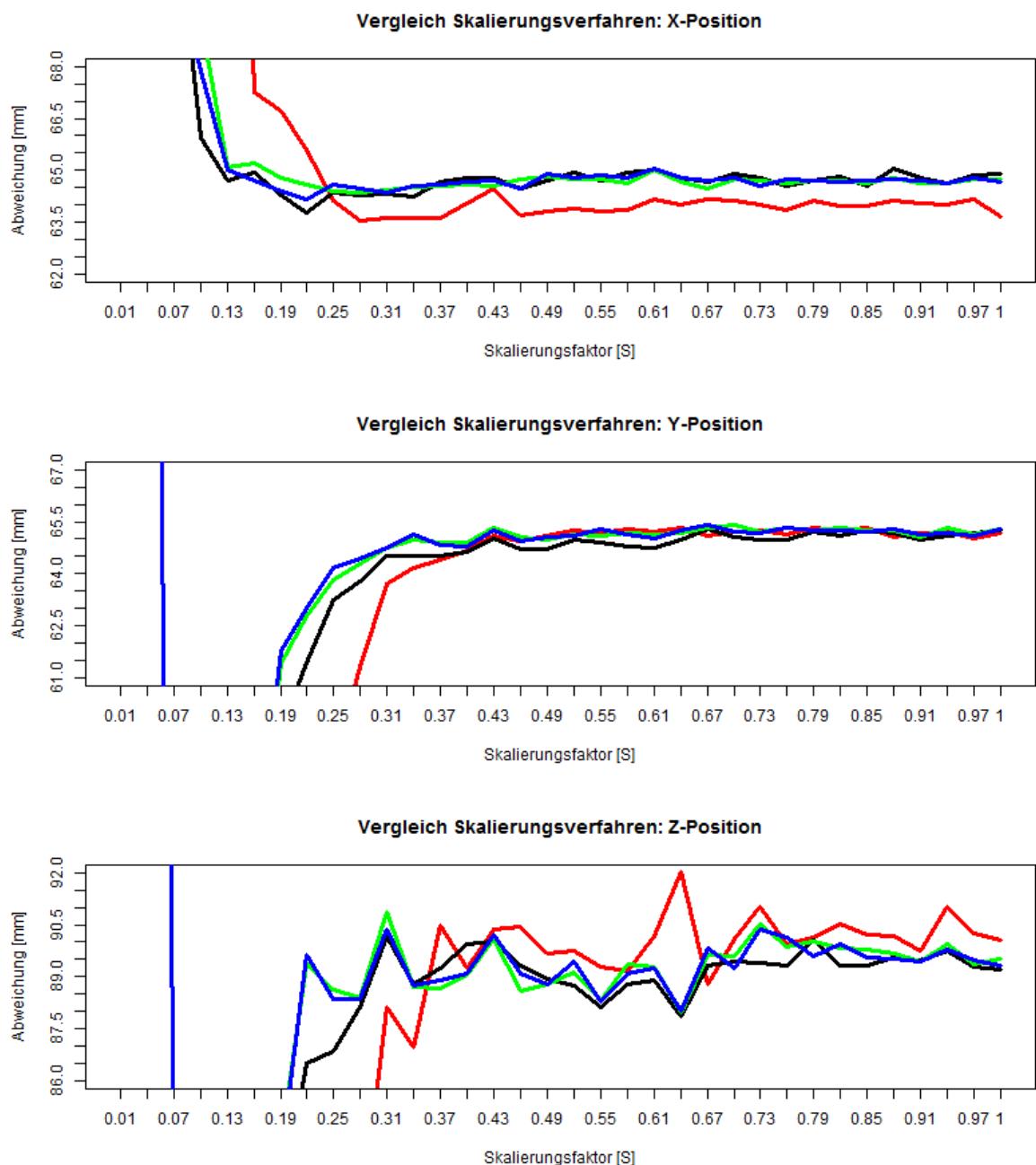


Abbildung 3.12: Dargestellt ist der Median der Abweichung zwischen der Berechneten Dreieckung und der des Datensatzes.
 Bicubic (blau), Lanczos (grün), Linear (schwarz), Nearest-Neighbor (rot)
 Oben: X-Position Abweichung: 62 – 68mm
 Mitte: Y-Position Abweichung: 58 – 68mm
 Unten: Z-Position Abweichung: 66 – 90mm

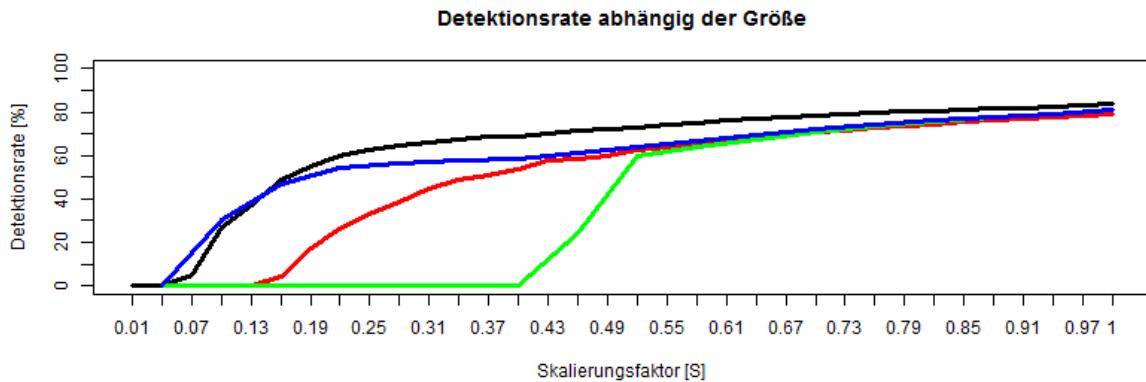


Abbildung 3.13: Bilder aus Labeled Faces in the Wild [14], mit dem X-Faktor verkleinert, um jedes Pixel mit 50% Wahrscheinlichkeit auf $\pm 10\%$ Gleichverteilung der Abweichung

wurde für jedes Bild viermal wiederholt um Zufälligkeiten bei der Rauschsimulation zu vermeiden.

Wie zu erwarten ist Nearest-Neighbor am schlechtesten, aber auch zwischen den anderen Verfahren sind nun Unterschiede zu erkennen, siehe Abbildung 3.13. Die gesamte Erkrankungsrate ist signifikant kleiner als ohne Rauschen, wobei die Position (0.15), ab welcher die Erkennungsrate rapide abfällt, beibehalten wird.

3.5.3 Ergebnis bezüglich Verwendbarkeit

Anhand der Detektionsrate abhängig von der Skalierung, siehe Abbildung 3.9, kann entnommen werden, das Gesichter unter 50 Pixel Größe nicht mehr sinnvoll erkannt werden können.

Werden sie hingegen hochskaliert, können sogar Gesichter mit einer Größe von 25 Pixel gefunden werden. Dies bedeutet das mit diesem Trick auch mit der Hälfte des Informationsgehaltes noch gearbeitet werden, wenn sie dadurch dem Trainingsdatensatz eher entsprechen.

Die Lanczos-Skalierung hat im Test mit einfach verkleinerten Bildern am Besten abgeschnitten mit einer der höchsten Detektionsrate bei den Skalierungen. Auch bei der Bestimmung der Rotation gehört es zu den besten Verfahren und kann auch bei der Positionsbestimmung überzeugen. Alle Anderen Verfahren zeigen Schwächen, wie die deutlich früheren Abfall der Detektionsrate von Nearest-Neighbor, die im Verhältnis zu den anderen starken Abweichung der X-Rotation von Bicubic oder das in allen Bereichen leicht schlechter abzuschneidende Lineare-Verfahren als das jeweilig Beste.

Der Test mit dem Pixelrauschen sollen etwaige Bildfehler simulieren, wie es bei schlechten Kameras der Fall sein kann und die Auswertung auf kleinen Bildausschnitten erschwert. Somit kann auch gezeigt werden, dass dieser Trick mit der Vergrößerung auch sehr wahrscheinlich in der späteren Anwendung funktionieren wird.

Es zeigt sich auch, dass für die Skalierung das Lineare Verfahren verwendet werden soll, da

3 Herangehensweise



Abbildung 3.14: Dies sind die Eingabebilder der verschiedenen Konverter von Farbe nach Grau. Links eine Farbpalette, Mitte Lena und Rechts ein Augenausschnitt aus dem Augendatensatz [37]

mit diesem im Verrauschen Bild die höchste Detektionsrate erreicht wird. Da die Unterschiede zwischen den einzelnen Verfahren aber recht gering ausfallen kann die Wahl des Skalierungsverfahren durch andere Kriterien abhängig gemacht werden wie z.B. der Rechenzeit, wobei vom Nearest-Neighbor abgeraten werden kann.

3.6 Umwandlung von Farbbild nach Graubild

Da die Berechnungen von ElSe auf Graubildern arbeitet und das Eingabebild in Farbe ist, muss es in ein Graubild umgewandelt werden.

Die Wahl des Verfahrens beruht auf der Anforderung, dass vor allem der Farbunterschied zwischen Pupille und der Umgebung maximal sein soll. Die Pupille möglichst dunkel und das restliche Auge hell. Die Farbe der Iris erschwert die Differenzierung zusätzlich, wenn diese recht dunkel ausfällt ist auch der Unterschied zur Pupille entsprechend gering in den Grauwerten. Außerdem ist das Erkennen der Pupille bei sehr kleinen Bildern schwierig bis unmöglich wodurch auf der Iris gerechnet werden muss, und daher diese weiterhin erhalten bleiben sollte.

Nach der Umwandlung wird für die Anwendung das Graubild noch normiert, damit Mindestens ein schwarzes und ein weißes Pixel vorhanden ist.

3.6.1 Gleam-Verfahren

Bei dem Gleam-Verfahren wird jede Farbe (Rot, Gelb und Grün) gleich stark bewertet allerdings wird jeder Farbwert mittels einer Gamma-Korrektur verändert und das Bild wirkt heller als bei dem Luminance-Verfahren.

Durch die Gamma-Korrektur wird vor allem der helle Bereich weiter erhöht, somit wird der Farbunterschied zwischen Iris und Auge vermindert, wodurch die Pupille der einzige dunkle Bereich wird.

Allerdings wird auch dieser Farbwert erhöht und sollte die Pupille nicht schwarz sein, wird sie eher ins Graue überführt, siehe Abbildung 3.15.

Dieses Verfahren wurde gewählt, da es im Vergleich zu den anderen Verfahren im Test von

„Color-to-Grayscale: Does the Method Matter in Image Recognition?“[6] am besten abgeschnitten hat.

$$G_{Gleam} = \frac{R^{\frac{1}{2.2}} + G^{\frac{1}{2.2}} + B^{\frac{1}{2.2}}}{3}$$

3.6.2 Gleam-New-Verfahren

Dies ist eine Variante von Gleam bei dem zuerst das gesamte Bild analysiert wird um die Parameter für die jeweilige Gamma-Korrektur zu ermitteln. Dies ist etwas aufwendiger, aber für die kleinen Bilder hinnehmbar.

Durch die individuelle Veränderung der Farbkanäle, werden Farbunterschiede minimiert und somit alle stark farbigen Bereiche ebenfalls dunkel dargestellt. Der Kontrast zwischen der farbigen Iris und dem weißen Auge wird verbessert, siehe Abbildung 3.16.

Da allerdings alle Farben dunkel werden, entstehen weitere dunkle Bereiche die die Detektion der Pupille beeinträchtigen können.

$$G_{GleamNew} = \frac{R^r + G^g + B^b}{3}$$

Wobei gilt $\{r, g, b\} = \frac{\log(V_{\max})}{\log(\{R, G, B\}_{\max})}$ mit V_{\max} als maximal möglicher Farbwert und R_{\max} als maximal Vorhandener Rot-Wert, G_{\max} und B_{\max} äquivalent.

3.6.3 Luminance-Verfahren

Dies ist ein lineares Verfahren, das der menschlichen Farbwahrnehmung entspricht und oft der Standard bei der Umwandlung von Farbbild nach Graubilder darstellt. Somit entsteht ein natürlicher Farbverlauf, bei dem der Farbunterschied zwischen Pupille, Iris und Auge auf einem mittleren Niveau bleibt, siehe Abbildung 3.17.

Eine Gamma-Korrektur wird bei der Umwandlung nicht verwendet.

$$G_{Luminance} = 0.299 \cdot R + 0.587 \cdot G + 0.114 \cdot B$$

3.6.4 Min-Max-Verfahren

Dabei handelt es sich eigentlich um zwei verschiedene Varianten, allerdings funktionieren beide nach dem selben Prinzip. Als Grauwert wird der jeweilige Extremwert aus den einzelnen Farbkanälen des Pixels gewählt.

Durch Verwendung der Extremwerte, ist nur noch der Wert von Relevanz, nicht die eigentliche Farbe, wodurch das gesamte Bild deutlich heller bzw. dunkler.

Bei dem Max-Verfahren werden alle farbigen und helle Bereiche hell dargestellt und nur gleichmäßig dunkel Bereiche bleiben dunkel wie es bei schwarz der Fall ist. Wenn der Minimalwert

3 Herangehensweise

anstelle verwendet wird, bleiben nur gleichmäßig helle Bereiche hell, alle anderen werden abgedunkelt.

$$G_{Max} = \max(R, G, B)$$
$$G_{Min} = \min(R, G, B)$$

3.6.5 Quadrat-Verfahren

Dies ist ein Verfahren, dass das Eingabebild verdunkelt und vom Aufbau dem Inversen von Gleam entspricht. Somit ist das gesamte Bild dunkler als bei dem Luminance-Verfahren, siehe Abbildung 3.19.

Durch die Abdunklung werden kleine Farbänderungen in den dunklen Bereichen reduziert, wodurch die Pupille sehr dunkel und der Farbunterschied zur Iris geringer ausfällt.

$$G_{Quadrat} = \frac{R^2 + G^2 + B^2}{3}$$

3.6.6 Normalisierung von Graubildern

Um ein Graubild zu erhalten, das das volle Spektrum der möglichen Grauwerte erfüllt, wird das Eingabebild normalisiert. Dazu wird der Maximale G_{max} und Minimale G_{min} Grauwert im Bild gesucht. Anschließend wird der neue Grau-Wert G_{new} wie folgt bestimmt:

$$G_{new} = (G - G_{min}) \cdot \frac{V_{max}}{G_{max} - G_{min}}$$

Dabei ist V_{max} der maximale mögliche Wert in der Ausgabe und G der aktuelle Grauwert im Bild.

Da für die Anwendung ein schwarzer Bereich gegen einen hellen Hintergrund gesucht wird, wird für die Bestimmung der Extremwerte nicht das originale Bild verwendet, sonder ein Gauß-gefiltertes.

Dies hat den Vorteil, das einzelne lokal auftretende Werte, z.B. Reflexionen, nicht als Extremwert verwendet werden, wodurch die Pupille gleichmäßiger dunkler und das gesamte Bild stärker aufgehellt wird.

Auswirkung des Gauß-Filters

Dies ist ein Tiefpassfilter und wird verwendet um das Eingangssignal zu glätten. Dies hat in der Bildverarbeitung den Effekt, dass Details im Bild verschwinden und das Bild unscharf wird. Die einzelnen Werte werden ihrer Umgebung angepasst, wodurch lokal auftretende Extrema verschwinden bzw. abgeschwächt werden und ähnliche Farbwerte zu ihrer Umgebung erhalten bleiben.

3.6 Umwandlung von Farbbild nach Graubild



Abbildung 3.15: Ergebnis der Umwandlung von Farb- nach Grauwert mittels Gleam-Verfahren



Abbildung 3.16: Ergebnis der Umwandlung von Farb- nach Grauwert mittels Gleam-New-Verfahren



Abbildung 3.17: Ergebnis der Umwandlung von Farb- nach Grauwert mittels Luminance-Verfahren

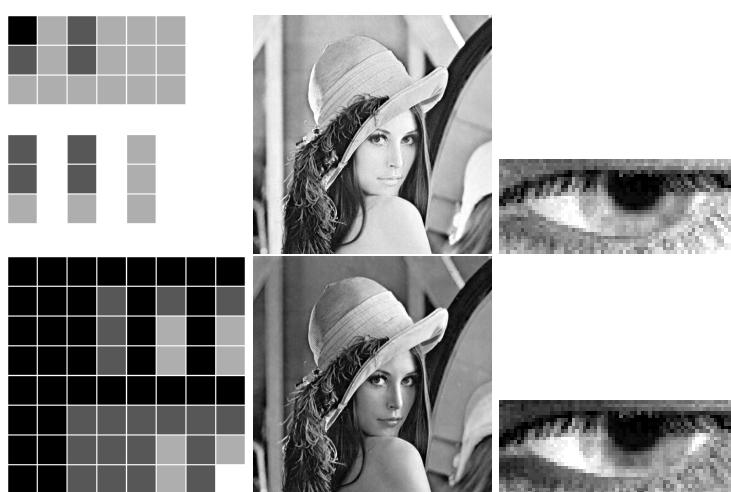


Abbildung 3.18: Ergebnis der Umwandlung von Farb- nach Grauwert mittels Extremwert-Verfahren. Oben: Max-Verfahren, Unten: Min-Verfahren

3 Herangehensweise



Abbildung 3.19: Ergebnis der Umwandlung von Farb- nach Grauwert mittels Quadrat-Verfahren

3.7 Ellipse Selection for Robust Pupil Detection (ElSe)

Zur Bestimmung der Blickrichtung ist die Augenregion natürlich von besonderer Bedeutung. Aus diesem Grund werden die Landmarks der Augenregion nochmals gesondert betrachtet. Aufgrund der besonderen Bedeutung existiert eine große Anzahl an Algorithmen, die speziell auf eine hochgenaue Bestimmung von Augenmerkmalen optimiert sind, wie Beispielsweise ElSe [36], Goutam [12], Starburst [8] oder Swirski [23].

Daher bestimmt OpenFace zusätzlich zu den 64 Landmarks, die das Gesicht beschreiben, weitere 28 Landmarks pro Auge, aus denen die Blickrichtung ermittelt wird. Um diese Augen-Landmarks zu bestimmen kommt ein weiteres CLNF zum Einsatz, das dafür trainiert wurde. Dabei zeigten die Vorabtests, dass die Detektion bei den getesteten kleinen Gesichtern unzureichend genau ausfällt.

Für die Bestimmung der Blickrichtung ist vor allem das Zentrum der Pupille bzw. Iris ausschlaggebend. Das Zentrum ergibt sich aus dem Umrissen (Landmarks) der Pupille bzw. Iris und muss möglichst exakt bestimmt sein., daher müssen diese aus dem Ergebnis von ElSe abgeleitet werden.

Um die Position der Landmarks zu verbessern, kann auf den Augenbereichen der ElSe-Algorithmus eingesetzt werden. Dieser Algorithmus basiert auf einer rechnerischen Ansatz und nicht auf Neuronen um die Umrisse der Pupille zu berechnen. Dieses Verfahren wurde gewählt, da es im Test [36] am besten abgeschnitten hat und direkt das Zentrum der Pupille liefert.

3.7.1 Beschreibung von ElSe

Bei realen Aufnahmen sind Bildfehler unvermeidlich, so können Reflektionen (Brille, Kontaktlinse usw.), Make-Up und körperliche Eigenschaften wie Augenfarbe die Detektion erschweren. Der Ursprüngliche ElSe-Algorithmus ist für Graubilder einer Eye-Tracking-Brille ausgelegt und optimiert, zudem ist es auf diesen Bildern zu einer Echtzeitauswertung in der Lage. Dieser Anwendungsbereich betrifft vor allem die hohe Qualität der Aufnahme im Bezug auf die Auflösung und die Infrarotbeleuchtung des Bildes. Die Infrarotbeleuchtung wird verwendet, damit das Auge ausreichend beleuchtet ist ohne den Probanden zu blenden.

Diese Voraussetzung führen dazu, dass die Detektionsleistung bei niedriger auflösenden Bildern rasch ab nimmt. Da die Berechnung unabhängig der Landmarks ausgeführt wird, empfiehlt sich

3.7 Ellipse Selection for Robust Pupil Detection (ElSe)

das Ergebnis zu überprüfen, damit die bestimmten Landmarks auch innerhalb der Augenhöhle liegt und grobe Fehler vermieden wird.

Für die Anwendung wurde der ursprüngliche ElSe-Algorithmus angepasst, um auf Farbbilder die nach Grau konvertiert wurden arbeiten zu können.

Als Ergebnis liefert ElSe eine Ellipse, die den Umriss der Pupille im Bild beschreibt. Aus dieser Ellipse können die Landmarks der Pupille abgeleitet werden. Ein Problem das schon im Test aufgetreten ist, entsteht wenn der Farbunterschied zwischen Iris und Pupille recht gering ausfällt oder durch Reflektionen der Kantenverlauf gestört wird.

Pupille bestimmen mit Kantendetektion

Da die Pupille als schwarzen Fleck im Bild dargestellt ist und die Iris einen helleren Farbton aufweist, wird ein Kantendetektor verwendet, der alle Pixel markiert, bei denen eine starke Farbänderung auftritt. Bei ElSe wird ein morphologischen Ansatz eingesetzt, von Relevanz sind nur zusammenhängende Kantenpixel um die Kante zwischen Pupille und Iris zu finden. Alle anderen Pixel können ignoriert werden, wobei jedes Kantenpixel als Startpunkt der Berechnung dienen kann.

Um jene Kantenpixel zu erhalten, die die Pupille beschreiben, wird versucht fortlaufende Kanten zu finden, die eine Ellipse bilden. Jene die nicht diesen Anforderung entsprechen, können recht schnell ignoriert werden. Anschließend werden auch alle offenen Ellipsenverläufe und jene Kantenpixel die am meisten vom bestimmten Verlauf abweichen, verworfen.

Das beste Ergebnis aller bestimmten Ellipsen wird als Lösung verwendet.

Grobe Bestimmung der Pupille

Sollte die Bestimmung der Ellipse, wie im letzten Kapitel beschreiben, scheitern, so wird das Zentrum des dunkelsten Kreises ermittelt. So ein Punkt kann immer gefunden werden, ist aber nicht zwingend die Pupille.

Auf einem verkleinerten Bild Abbildung 3.20 (1) wird ein kreisförmiger Mean-Filter eingesetzt mit Ergebnis in Abbildung 3.20 (3). Zur zweiten Faltung wird der Durchschnitt über ein Quadrat ohne inneren Kreis eingesetzt mit Ergebnis in Abbildung 3.20 (2), wobei bei beiden Kreisen der selbe Radius verwendet wird.

Nun wird das Ergebnis des Quadratischen Mean-Filters invertiert Abbildung 3.20 (4) und mittels Punkt-Multiplikation mit dem anderen Meanfilter zusammengebracht Abbildung 3.20 (5). Im resultierendem Bild wird nun der höchste Wert gesucht, da dies das Zentrum des dunkelsten kreisförmigen Ortes im Bild ist.

Ergebnis des Beispiels ist als Kreuz in Abbildung 3.20 (6) markiert.

3 Herangehensweise

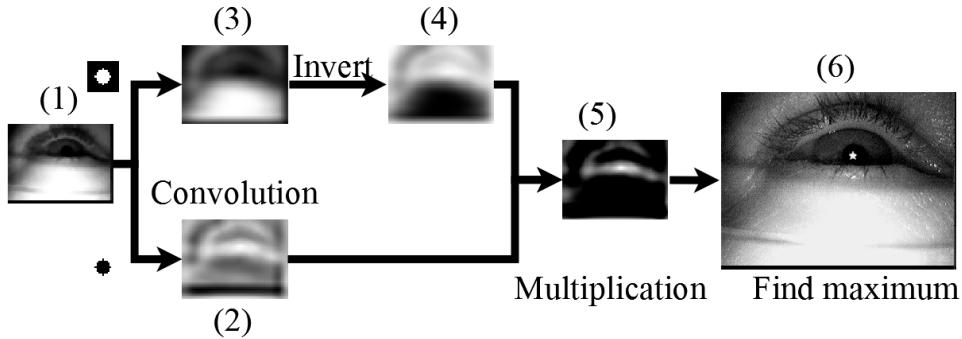


Abbildung 3.20: Ablauf der alternativen Berechnung zur Pupillen-Detektion von [36]

Veröffentlichte Ergebnisse

Für den Test, wurden Bilder von 384×288 Pixel Größe verwendet. Im Vergleich zu den anderen Verfahren, ist ElSe in den meisten Fällen überlegen, mit einer Verbesserung der Erkennungsrate um 14.53% auf dem verwendeten Datensatz [36].

3.7.2 Das Verfahren im Test

Der Ursprüngliche ElSe-Algorithmus wurde für Eye-Tracking Brillen entwickelt worden, daher soll geprüft werden in wieweit es in dieser Anwendung eingesetzt werden kann.

Um die einzelnen Grau-Verfahren besser vergleichen zu können, wurden künstliche Augen aus dem Datensatz [37] verwendet damit die exakte Position der Landmarks bekannt ist.

Ein gutes Verfahren muss stabil gegenüber der Skalierung sein, damit es auch auf kleinen Bereichen zuverlässig arbeitet. Da für die spätere Anwendung vor allem das Zentrum der Pupille von Interesse ist, wird der euklidische Abstand zum Zentrum als Qualitätsmaß verwendet.

Da ElSe für Eye-Tracking Brillen entwickelt wurde, also für ein Qualitativ hochwertiges Bild eines Auges, wurde der Bildbereich soweit verkleinert das nur noch alle Landmarks des Auges mit etwas Rand dargestellt werden, um diesen Anforderungen entsprechend nahe zu kommen. Somit sind die Bildausschnitte im Datensatz auf denen gerechnet wird etwa 64 auf 29 Pixel groß und werden für die Verarbeitung auf eine Breite von 384 Pixeln vergrößert, die Auflösung, wofür ElSe entwickelt wurde. Da durch die Skalierung allerdings keine zusätzlichen Informationen entstehen, ist vor allem die grobe Bestimmung der Ellipse, beschreiben in Abschnitt 3.7.1, von Interesse. Diese Auswahl des Bildbereiches kann auch in der späteren Anwendung eingesetzt werden, da der Augenbereich durch eigene Landmarks in der Gesichtsanalyse, relativ genau bestimmt ist.

Um die Qualität der Berechnung bei verschiedenen Größen zu ermittelt, wurde das Bild linear verkleinert.

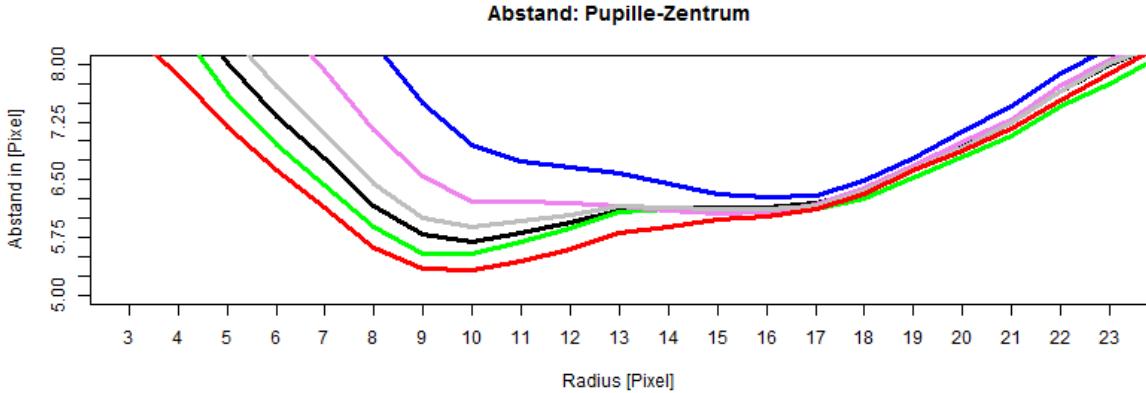


Abbildung 3.21: Median-Abstand in Pixel des Zentrums der Pupille gegen die Veränderung des Radius des Filters.

Verfahren: Gleam (rot), Luminance (schwarz), Max (grün), Min (violett), New-Gleam (grau), Quadrat (blau)

Auswirkung der verschiedenen Graubild-Verfahren

Es zeigt sich, dass die Verfahren, um den Farbwert in einen Grauwert zu überführen, durchaus Auswirkungen auf die Qualität der Berechnung hat.

Als Kriterien wird die Differenz zwischen den berechneten Radien der den Datensatzes von der Pupille und Iris. Außerdem der euklidischen Abstand zum berechneten Zentrum der Pupille. Der minimale Abstand der berechneten Zentren ergibt sich bei dem Gleam-Verfahren mit 5.327 Pixel als Median, siehe Abbildung 3.21. Der Beste Radius für den Filter ist für die Position der Iris bei 10 Pixel

Ein Unterschied zwischen den Verfahren konnte bei der Bestimmung des Radius der Pupille nicht gefunden werden, siehe Abbildung 3.22 links. Der beste Radius für den Filter ist im Test bei 8 Pixel und ergibt eine Abweichung von 1,555 Pixel.

Für die Bestimmung der Iris hat das quadratische Verfahren die geringste mittlere Abweichung mit 2,488 Pixel, nur etwas genauer als Min-Verfahren (2,49 Pixel). Für diese Berechnung ist ein Radius des Filters von 18 Pixel am besten gewählt.

Somit wurde drei Verfahren ausgewählt um diese näher zu untersuchen, Gleam mit der geringsten Abweichung des Zentrums, Quadrat als bestes Resultat bei der Iris und Luminance da es ein Standardverfahren ist. Alle mit den Radien 8/10/18 für die Berechnung der Pupille/Zentrum/Iris.

Es zeigt sich, dass der Fehler in den Berechnungen über einen großen Skalierungsbereich gleich bleibt. Der Fehler bei der Bestimmung des Zentrums ist bis zu einer Skalierung von 0,4 bei 5,4 Pixel, siehe Abbildung 3.23.

Auch bei der Berechnung der Pupille ist weiterhin kein Unterschied zu erkennen, siehe Abbildung 3.24 oben.

Die Bestimmung der Pupille und Iris bleibt auf dem gleichen Niveau bis zu einer Skalierung von 0,15. Auch bleibt die Unterschiede der Verfahren bleiben immer ähnlich.

3 Herangehensweise

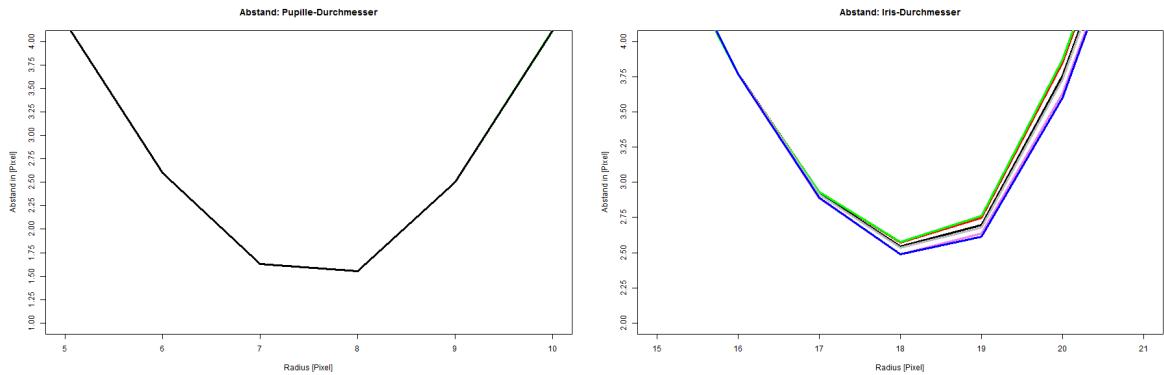


Abbildung 3.22: Differenz zwischen den Radien gegen die Veränderung des Radius des Filters von Pupille (links) und Iris (rechts)

Verfahren: Gleam (rot), Luminance (schwarz), Max (grün), Min (violett), New-Gleam (grau), Quadrat (blau)

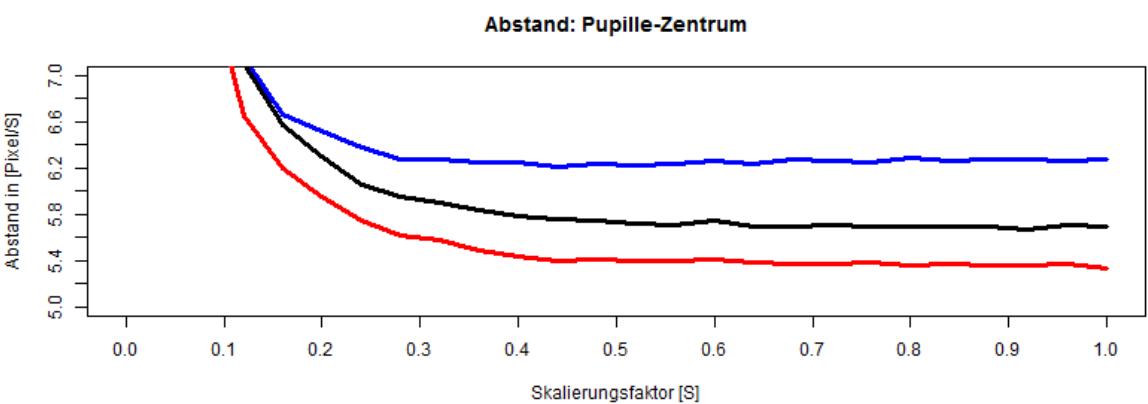


Abbildung 3.23: Euklidischer Abstand in Pixel zwischen dem berechneten Zentrum der Pupille und dem des Datensatzes gegen die Veränderung des Radius des Filters.

Verfahren: Gleam (rot), Luminance (schwarz), Quadrat (blau)

3.7 Ellipse Selection for Robust Pupil Detection (ElSe)

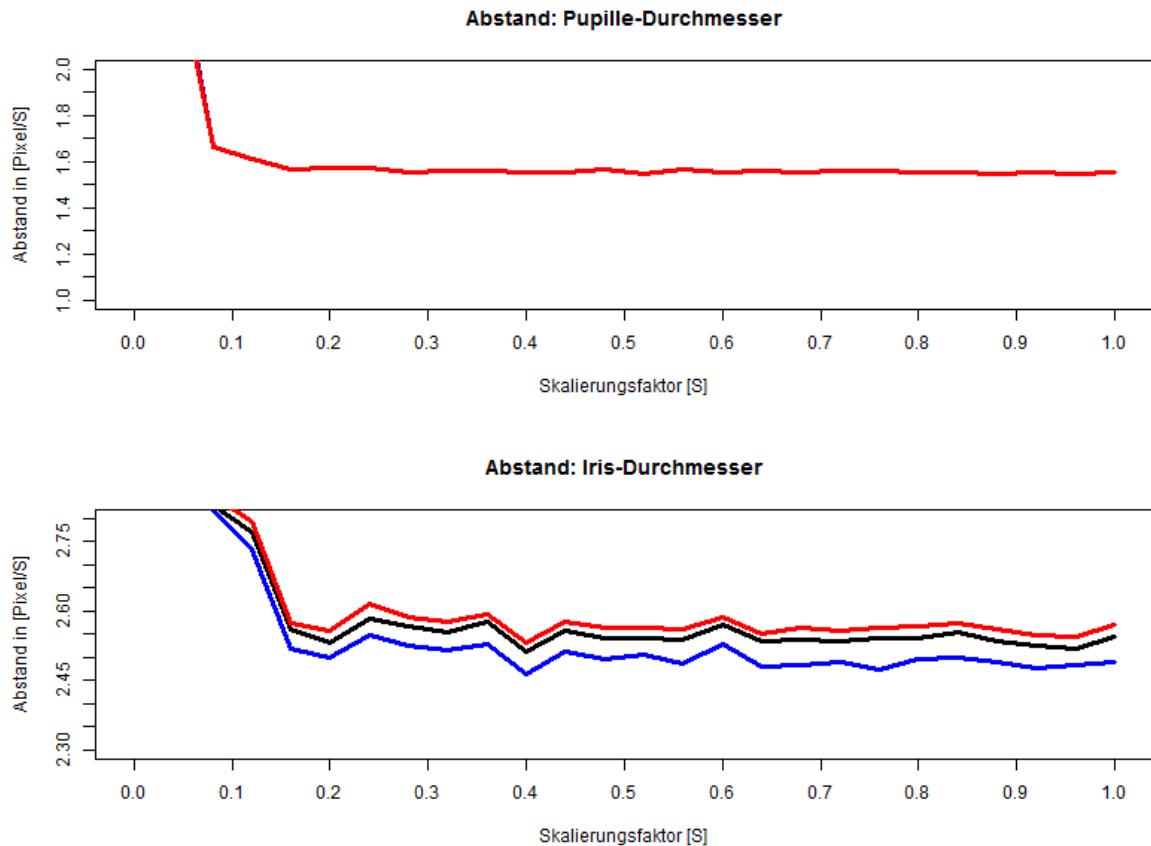


Abbildung 3.24: Differenz in Pixel zwischen den Radien der Berechnung und dem des Datensatzes gegen die Veränderung des Radius des Filters. Oben: Pupille, Unten Iris
Verfahren: Gleam (rot), Luminance (schwarz), Quadrat (blau)

Auswirkung des Filterradius

Ein wichtiger Parameter des ElSe-Verfahrens ist der Radius des Filters. Um den besten Parameter zu bestimmen wurde der Augen-Datensatz [37] verwendet und die Augenpartie ausgeschnitten. Im Datensatz besitzen die abgebildeten Augen durchschnittlich 15 Pixel Breite Pupille und eine Iris von 34 Pixel Durchmesser.

In Abbildung 3.22 ist zu erkennen, dass der Radius signifikant für die Qualität der Berechnung ist. Da für die spätere Anwendung vor allem das Zentrum der Pupille von Interesse ist, vgl. Abschnitt 3.8.1, muss ElSe in diesem Aspekt zuverlässig Ergebnisse liefern.

Im Versuch hat sich ein Radius von etwa einem Zwölftel des zu erwartetem Durchmesser der Iris bzw. Pupille als sinnvoll erwiesen, um deren Ausmaße möglichst exakt zu bestimmen. Im Versuch entspricht dies 8 und 18 Pixel.

Um die Position des Zentrums der Iris und der Pupille möglichst gut zu bestimmen, erwies sich ein Radius von 10 am besten, siehe Abbildung 3.21, wobei dieser Fehler nicht so sehr steigt bei Veränderung des Radius, als bei der Größenbestimmung von Pupille und Iris.

Vergleich zu OpenFace

Als Referenz wird das Ergebnis von OpenFace, für die zusätzlich bestimmten Landmarks der Augen, verwendet. Dies wurde auch auf dem Augendatensatz [37] angewendet, um vergleichbare Ergebnisse zu erhalten.

In Abbildung 3.25 ist zu erkennen dass dieses Verfahren im Schnitt oft schlechtere Ergebnisse liefert als das ElSe, allerdings ohne großen Fehler und auch öfters genauere Ergebnisse.

Da die hohe Qualität von ElSe nur erreicht werden kann, wenn es auf passenden Bildausschnitt angewendet wird, ist auch die Detektion des Auge von Interesse.

Nach Abbildung 6.12 ist zu entnehmen, dass der Bereich des Auges zwar nicht so exakt bestimmt wird, allerdings überdeckt er den relevanten Bereich ausreichend genau. Somit liegen die Landmarks der Augen im Bildausschnitt, wodurch diese Ausschnitt als Eingabe von ElSe verwendet werden kann.

3.7.3 Ergebnis

Im Test ist bei allen Skalierungen im durchschnitt ElSe den Ergebnisse von OpenFace überlegen, durch die Verteilung ist allerdings eine Kombination beider Verfahren sinnvoll, so kann das Ergebnis von OpenFace bei Bilder in denen die Iris größer als 21 Pixel ist direkt als Lösung verwendet werden, da der mögliche Fehler von OpenFace geringer ist als der von ElSe.

Im Bereich zwischen 21 und 15 Pixel können beide Ergebnisse Kombiniert werden, da sie ungefähr gleich gute Ergebnisse liefern.

Sollte die Iris im Originalbild noch kleiner sein, so ist ElSe deutlich genauer, da es noch bis zu einer Irisgröße von 3 Pixel stabil funktioniert.

Es ist zu erkennen, das ElSe zwar oft bessere Ergebnisse liefert als OpenFace, wobei auch grobe Fehler auftreten können, was an der Verteilung zu erkennen ist wenn Abbildung 3.25

3.8 Bestimmung des Ziels der Aufmerksamkeit

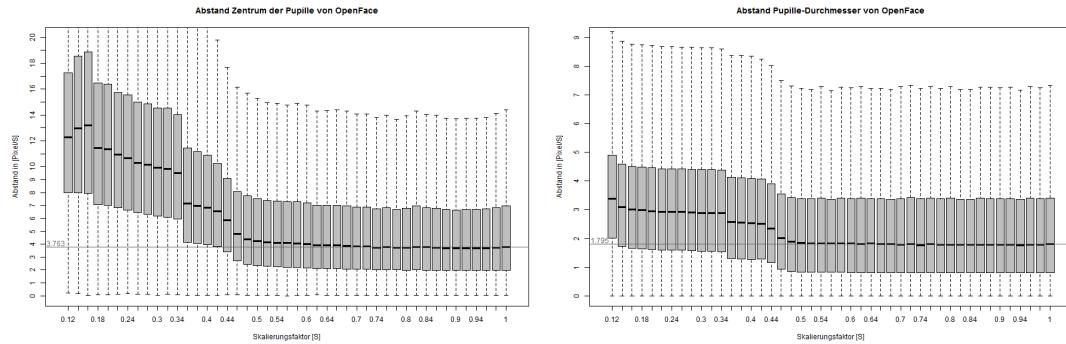


Abbildung 3.25: Auswirkung der Bildgröße auf die Qualität der Augendetektion von OpenFace.
Aufgetragen ist die Abweichung [Pixel/Skalierung] gegen den Skalierungsfaktor.

und Abbildung 6.11 verglichen werden. Eine genauere Darstellung der Messergebnisse ist in Kapitel 6 dargestellt. Die Auswirkung der Radien und der verschiedenen Verfahren auf die Pupille ist in Abbildung 6.9, auf die Iris in Abbildung 6.10 und auf die Bestimmung des Zentrums in Abbildung 6.8. Die Auswirkung der Skalierung in Abbildung 6.11.

3.8 Bestimmung des Ziels der Aufmerksamkeit

Um das Ziel der Aufmerksamkeit einer Person zu bestimmen, muss die reale Position ermittelt werden. Die Orientierung des Gesichtes und die Blickrichtung können als Verlauf einer Ursprungsgerade betrachtet werden, mit dem Ursprung an der Position des Gesichtes im Raum. Ist der Ursprung und die Gerade bekannt, so kann ermittelt werden, ob sie durch bestimmte Punkte im Raum verläuft. Ist dies der Fall, so wird dieser Punkt wahrscheinlich betrachtet und ist Ziel der Aufmerksamkeit der Person.

3.8.1 Bestimmung der Position & Orientierung des Gesichts

Wird ein weiter entfernter Punkt von beiden Augen fokussiert, so kann die Blickrichtung beider Augen als parallel angenommen werden, da der Unterschied zwischen Beiden minimal ausfällt. Um den Fehler zu minimieren wird als Ergebnis die durchschnittliche Blickrichtung beider Augen verwendet. Da die Berechnung für jedes Auge unabhängig vom anderen ausgeführt, wodurch Messungenauigkeiten dazu führen, dass die berechnete Blickrichtung der beiden Augen in verschiedene Richtung verlaufen.

Zur Bestimmung der Translation und Orientierung des Gesichtes wird ein CLNF bzw. PDM eingesetzt. Dabei wurde es mit der Kameraabbildung von 3D-Landmarks eines normierten Kopfes in verschiedenen Ausrichtungen initialisiert. Das normierte Ergebnis kann mit den passenden Kameraparameter von der Aufnahme angepasst werden um die reale Position und Orientierung zu bestimmen.

3 Herangehensweise

Abschätzen der Kameraparameter

Sind keine Kameraparameter bekannt, so können diese anhand der Bildauflösung grob geschätzt werden. Bei der Schätzung der Brennweite für ein Bild mit einer Dimension $I_x \times I_y$ wird das Standardobjektiv mit einer Auflösung von 640×480 Pixel angenommen, somit ergeben sich die Brennweiten f_x und f_y wie folgt:

$$f_x = 500 \cdot \frac{I_x}{640}$$

$$f_y = 500 \cdot \frac{I_y}{480}$$

Position & Orientierung

Zur Bestimmung der Kopfposition $P = (X_{avg} \ Y_{avg} \ Z_{avg})^t$ im Kamerakoordinaten wird die Größe, ein Skalierungsfaktor der normierten Kopfgröße S_G , im Bild verwendet.

Bei der Abbildung von Welt- nach Bild-Koordinaten gilt: $x = f \cdot \frac{X}{Z}$ und $y = f \cdot \frac{Y}{Z}$, damit kann die Tiefe wie folgt abgeschätzt werden.

Sei $P_1 = (X_1 \ Y_1 \ Z_1)^t$, $P_2 = (X_2 \ Y_2 \ Z_2)$ die Beschreibung der Größe G eines Kopfes mit:

$$a = \frac{\sqrt{(X_1 - X_2)^2 + (Y_1 - Y_2)^2}}{\frac{|Z_1 - Z_2|}{2}} = \frac{G}{Z_{avg}}$$

$$S = \frac{S_G}{G}$$

$$\Rightarrow a \cdot f = f \cdot \frac{G}{Z_{avg}} = S_G$$

$$Z_{avg} = \frac{f}{S_G} \cdot G = \frac{f}{S}$$

$$X_{avg} = \frac{x \cdot Z_{avg}}{f}$$

$$Y_{avg} = \frac{y \cdot Z_{avg}}{f}$$

Dies beschreibt allerdings nur eine Annäherung an die tatsächliche Position, da die Distanz mit Hilfe einer durchschnittlichen Kopfgröße geschätzt wird.

[25]

Bestimmung der Blickrichtung

Für möglichst genaue Ergebnisse wird für die Augenpartie ein weiteres CNN eingesetzt das nur auf diesem Bildaufschnitt arbeitet und weitere 28 Landmarks bestimmt. Durch diese werden

die Lider, Iris und Pupille dargestellt und für jedes Auge separat bestimmt.

Zur Bestimmung der Blickrichtung wird wie folgt vorgegangen: Zuerst wird der Strahl bestimmt der, ausgehend vom Zentrum der Kamera, durch das Zentrum der Pupille verläuft. Nun wird der Schnittpunkt zwischen diesem Strahl und einer Sphäre bestimmt, die das Auge repräsentiert. Anschließend wird ein Strahl bestimmt der vom Zentrum der Sphäre ausgehend durch den berechneten Schnittpunkt verläuft, dies ist die resultierende Blickrichtung.

Zusammenhang von Bildposition & Weltposition

Als Ausgangspunkt werden die Ergebnisse des CNN verwendet um die Position zu bestimmen. Zur Bestimmung der Orientierung R liefert auch das CNN ein Ergebnis R_{CNN} . Allerdings stimmt es nur im Zentrum des Bildes, da am Rand immer mehr die Orientierung der einzelnen Pixel mit berücksichtigt werden muss.

$$\begin{aligned} \text{euler}_x &= \tan^{-1}\left(\frac{\sqrt{X^2 + Z^2}}{Z^2}\right) \\ \text{euler}_y &= \tan^{-1}\left(\frac{\sqrt{Y^2 + Z^2}}{Z^2}\right) \end{aligned}$$

$R_{pos} = R(\text{euler}_x, \text{euler}_y, 0)$ Umwandlung zur Rotationsmatrix

$$R = R_{CNN} \cdot R_{pos}$$

Eine weitere Verbesserung kann erreicht werden, indem die gefundenen 2D-Landmarks mit Hilfe des PDM in 3D zu überführen. Um anschließend die Überführung von 2D nach 3D-Koordinaten erneut zu bestimmen um die Orientierung und Position zu ermitteln. Auch bei diesem Verfahren muss die Pixelorientierung beachtet werden. Allerdings ist auch ein Tiefenbild nötig, da ansonsten die Fehler weiter verstärkt werden. Daher ist es in der aktuellen Anwendung nicht sinnvoll einsetzbar.

3.8.2 Größe und Genauigkeit

Um die Qualität der Berechnung auf verschiedenen Distanzen zu ermitteln, wurde der Datensatz Forests for Real Time 3D Face Analysis [9] verwendet, da für jedes Gesicht die Position und Orientierung bekannt ist. Die durchschnittliche Distanz zwischen Kamera und Kopf beträgt ca 70cm bei einer Kopfbreite von 78 Pixel. Um die verschiedenen Distanzen zwischen Probanden und Kamera zu simulieren, wurden die Bilder mit dem angegebene Skalierungsfaktor (X-Achse) linear verkleinert.

Da verschiedene Verfahren zur Bestimmung der Position und Orientierung zur Verfügung stehen, sollen diese miteinander verglichen werden. Zur Bestimmung wurde nur das RGB-Bild verwendet und nicht zusätzlich die Tiefeinaufnahme, da dies in der Anwendung auch nicht vorhanden sind.

3 Herangehensweise

Position

Zur Bestimmung der Position gibt es zwei Verfahren, die direkte mittels Brennweite und Skalierung oder Überführungsmatrix von 3D zu 2D Landmarks arbeiten.

Die Funktionen PoseCamera und PoseWorld verwenden die einfache Bestimmung mittels Skalierung und CorrectPoseCamera und CorrectPoseWorld die Überführung von 3D und 2D Landmarks, daher überlagern sich die Linien in Abbildung 3.26, da die jeweiligen Verfahren nach dem selben Prinzip rechnen.

Der schnelle Abfall der Genauigkeit bei der Skalierung 0,25 ist an der selben Stelle an der auch die Detektionsrate stark absinkt, siehe Abschnitt 3.5.2. Somit kann das Verfahren bis zu seiner Grenze eingesetzt werden und erst, wenn die Detektion schwierig wird steigt auch der Fehler.

Orientierung

Bei der Rotation Zeigen sich nun Unterscheiden zwischen den einzelnen Verfahren, da bei PoseWorld und CorrectPoseWorld auch die Position im Kamerabild berücksichtigt wird.

Es Zeigt sich in Abbildung 3.27, das die zusätzliche Korrektur das Ergebnis weiter verbessern, vor allem wenn sich das Gesicht weit außerhalb des Zentrums befindet.

Ergebnis

Es zeigt sich, dass CorrectPoseWorld, also die komplexe Bestimmung der Position mittels 2D/3D Landmarks und zusätzlicher Korrektur der Winkel die besten Ergebnisse liefert im Test. Im Test ist die Überführung von 3D ud 2D Landmarks am besten (CorrectPoseCamera und CorrectPoseWorld) kann sich allerdings auch ändern wenn die Kamera Parameter besser abgeschätzt sind, da ohne eine Tiefenaufnahme die korrekte Überführung nur geschätzt werden kann und sich Fehler fortpflanzen können.

3.8.3 Bestimmung eines Punktes, auf der die Aufmerksamkeit liegt

Von Interesse ist vor allem der Punkt auf den der Blick ruht bzw. das Gesicht ausgerichtet ist. Bestimmung des Richtungsvektors V aus der Rotationsmatrix

$$V = R \cdot (0, 0, -1)^T$$

Aus der Blickrichtung mehrerer Probanden kann auch der reale Punkt der Aufmerksamkeit ermittelt werden. Dazu wird die Blickrichtung als Linie $L_i = s \cdot n_i + p_i$ beschrieben mit $s \in \mathbb{R}$ und $n_i, p_i \in \mathbb{R}^3$ verwendet.

$$c = \left(\sum_i I - n_i n_i^T \right)^{-1} \left(\sum_i (I - n_i n_i^T) \cdot p_i \right)$$

3.8 Bestimmung des Ziels der Aufmerksamkeit

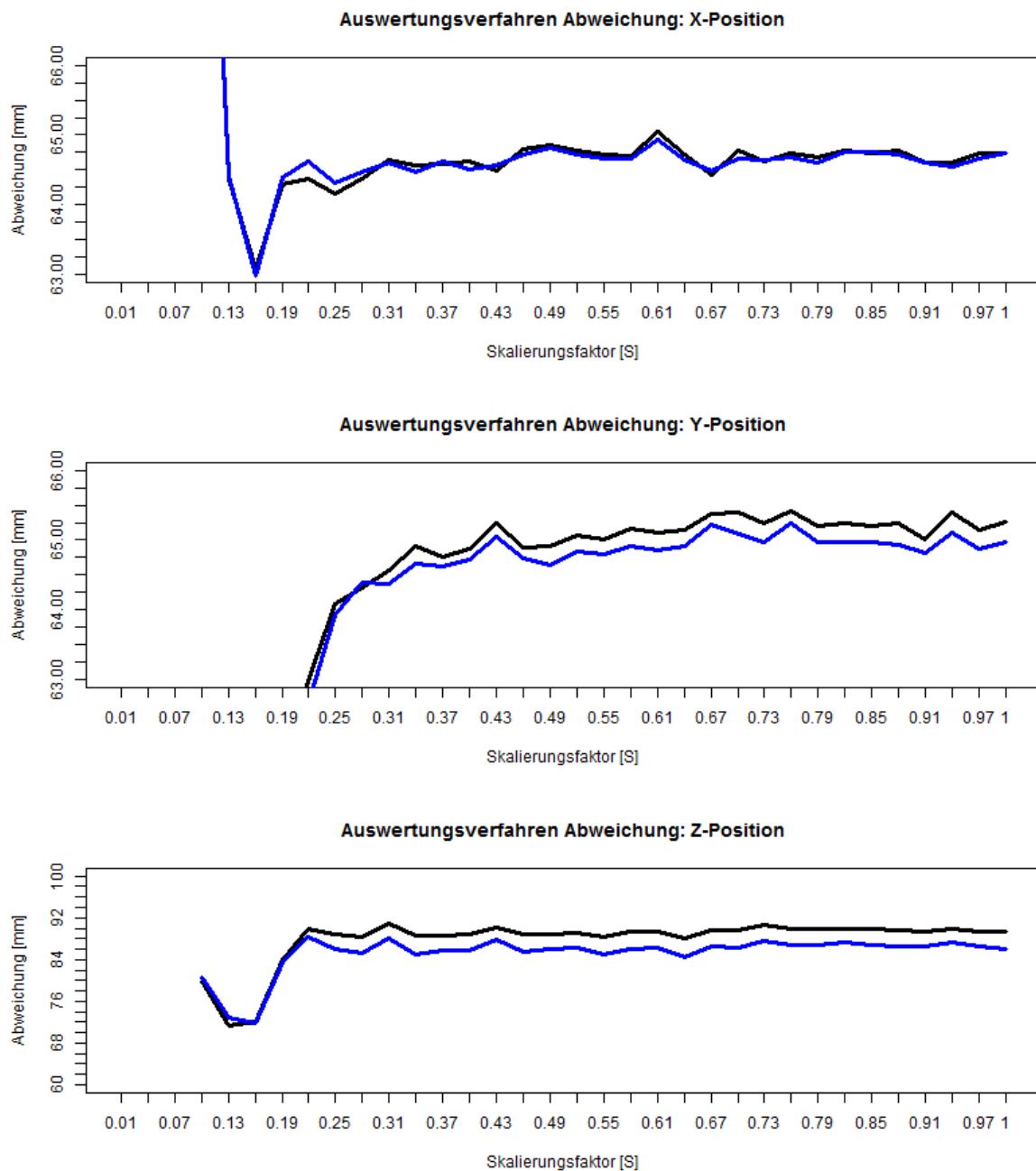


Abbildung 3.26: Dargestellt ist der Median der Abweichung in Millimeter der Positionsbestimmung auf Bilder die mit Lanczos skaliert wurden.
 PoseWorld (schwarz), PoseCamera (rot), CorrectPoseCamera (grün) und PoseWorld (blau)
 Oben: X-Position, Mitte: Y-Position, Unten: Z-Position

3 Herangehensweise

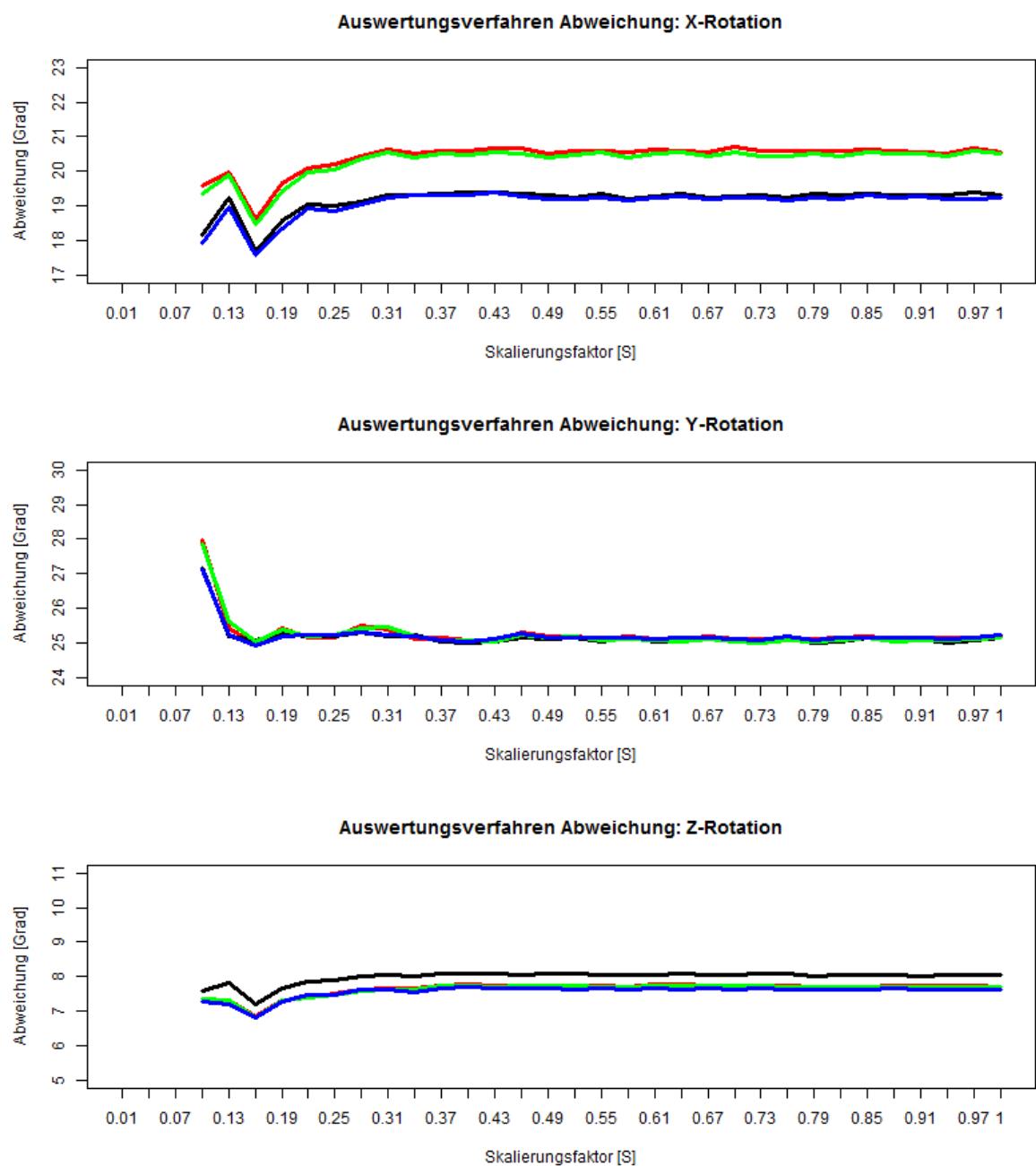


Abbildung 3.27: Dargestellt ist der Median der Abweichung in Grad der Positionsbestimmung auf Bilder die mit Lanczos skaliert wurden.
PoseWorld (schwarz), PoseCamera (rot), CorrectPoseCamera (grün) und CorrectPoseWorld (blau)
Oben: X-Rotation, Mitte: Y-Rotation, Unten: Z-Rotation

3.8 Bestimmung des Ziels der Aufmerksamkeit

Bei Verwendung der Gesichtsorientierung ergibt sich das Problem den konkreten Blickpunkt zu ermitteln, da die Augenbewegung nicht erfasst werden kann. So muss ein Kegel, der den üblichen Bereich der Augenbewegung umfasst, um die Orientierung berücksichtigt werden als Fehlertoleranz und der gesamte Bereich kommt als Lösungen in Frage. Außerdem liegt der Punkt der Aufmerksamkeit meist außerhalb des Bildbereiches der Kamera und muss entsprechend von einer Anwendung interpretiert werden.

Soll die Position des Ziels auf nahezu parallel verlaufende oder stark verrausche Ergebnisse berechnet werden, so ist die Bestimmung des Schnittpunkts nach dem obigen Verfahren nicht möglich.

Eine einfache Variante ist das Verwenden des durchschnittlichen Richtungsvektors V_{avg} und Position P_{avg} der Probanden. Die Tiefe a muss nun geschätzt werden um das Ziel $P = V \cdot a$ zu bestimmen.

4 Evaluation

4.0.1 Schulklassenvideo

ToDo:

- Anonymer Skreenshot
- Erfolge beschreiben
- Problematik beschreiben

4.1 Vorversuche

Um einen Eindruck über den zu erwartenden Datensatz und Schwierigkeiten zu erhalten, wurde einige Testdatensätze mit der Actioncam erstellt.

4.1.1 Arbeitsbereich der Verfahren - Versuch 1

Mit diesem Versuch soll der Zusammenhang zwischen Standort der Probanden und Targets untersucht werden. Dazu wird ein Klassenzimmer simuliert mit weit verteilten Schülern, die den gesamten frontalen Betrachten.

Versuchsaufbau

In einem Raum wurde die Kamera in $2.06m$ Höhe $31cm$ hinter den Targets so montiert, dass der gesamte Raum im Fokus liegt. Als Targets wurden 9 Punkte auf einer Ebene markiert mit der Kamera im Zentrum. Die Anordnung der Targets ist in Abbildung 4.1 dargestellt. Als Position der Probanden wurde ein Rastfeld mit $1m$ Kantenlänge im Raum eingezeichnet auf einer Fläche von $7 \times 11m$. Die Probanden stellten sich auf diesen Positionen auf um nacheinander alle Targets zu betrachten.

4 Evaluation

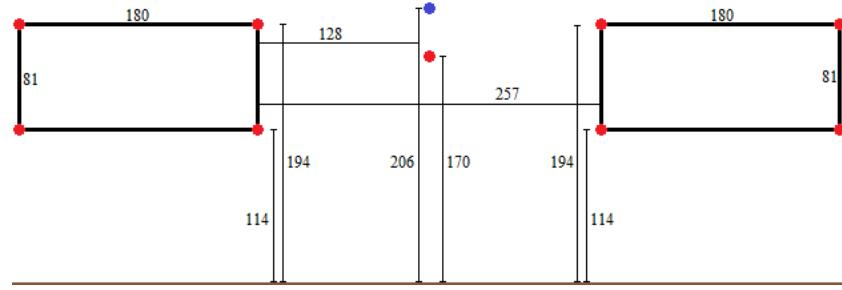


Abbildung 4.1: Aufbau der Targets im Vorversuch, alle Angaben gerundet in Zentimeter
rote Punkte: Target, blauer Punkt: Kamera

Detektion mit MTCNN

Um die Detektionswahrscheinlichkeit der MTCNN-Face Detektor zu testen wurden dieses Videos analysiert.

Es zeigt sich, das auf allen Positionen die Probanden erfolgreich erkannt wurden und die Boxen das Gesicht recht gut beschreibt. Allerdings ist zu erkennen, das die Landmarks unzureichend genau sind. Sie sollten die Mundwinkel, Nasenspitze und beide Augen markieren, liegen aber schon bei recht großen Bildern weit daneben, siehe Abbildung 3.3

Auswertung

Für die Analyse wurde aus dem Video jene Frames ausgewählt in denen ein Target fokussiert wurde und analysiert.

Für eine Analyse wurde zuerst die Einzelbildauswertung von OpenFace auf die Frames angewendet und jene resultierende Kopfrotationen markiert, an denen eine erfolgreich ein Gesicht erkannt wurde. In Abbildung 4.2 ist der horizontale Wertebereich dargestellt in denen an der jeweiligen Position ein Gesicht erfolgreich erkannt wurde.

Im zweiten Teil wurden die selben Frames für die Messung verwendet, dieses mal allerdings wurde das gesamte Video analysiert. Der Winkelbereich in denen auf der horizontalen Achse an den entsprechenden Positionen ein Gesicht erkannt wurde, ist in Abbildung 4.3 dargestellt. Das Fehlen von Ergebnissen in Spalte $-3m$ liegt an der unzureichenden Detektion. Als Ursache kann die Überbeleuchtung durch das einfallende Licht der Fenster angenommen werden.

Ergebnis

Es zeigt sich, dass eine Auswertung auf einem Video deutlich zuverlässiger arbeitet als auf Einzelbilder, vor allem der größere Rotationsbereich ist von Vorteil.

Durch die Verwendung des Weitwinkelobjektivs, kann die gesamte Breite eines Klassenzimmers erfasse werden und der Winkelbereich für eine erfolgreiche Detektion ist breit genug um Schüler erfassen zu können, die selbst die fordern Eckpunkte eines Klassenzimmers betrachten.

Bei der Distanz zur Kamera (Tiefe) ist Handlungsbedarf, als Ziel wurde 8m angesetzt und das

| | | | | | | |
|-----|-----|-----|--------|-----|-----|--|
| +5m | | | | | | |
| +4m | | | | | | |
| +3m | | | | | | |
| +2m | / | | | | | |
| +1m | | | | | | |
| | -2m | -1m | Kamera | +1m | +2m | |

Abbildung 4.2: Dargestellt ist der horizontale Winkebereich in dem mit der Image-Verarbeitung ein Gesicht erkannt wurden.

| | | | | | | |
|-----|-----|-----|--------|-----|-----|-----|
| +5m | ▲ | ▲ | ▲ | ▲ | ▲ | ↗ |
| +4m | ▲ | ▲ | ▲ | ▲ | ▲ | ↖ |
| +3m | ▲ | ▲ | ▲ | ▲ | ▲ | ↙ |
| +2m | | ▲ | ▲ | ▲ | ▲ | ↘ |
| +1m | | ▲ | ▲ | ▲ | | |
| | -2m | -1m | Kamera | +1m | +2m | +3m |

Abbildung 4.3: Dargestellt ist der horizontale Winkebereich in dem mit der Video-Verarbeitung ein Gesicht erkannt wurde.

4 Evaluation

aktuelle Verfahren endet bei *5m*.

Eine signifikante Aussage bezüglich des vertikalen Winkel kann aus diesem Aufbau nicht getroffen werden, da die Neigungswinkel zu ähnlich ausfallen bei stehenden Personen und beide einem geradeaus Blick ähneln.

4.1.2 Arbeitsbereich der Verfahren - Versuch 2

Da ein aufmerksamer Schüler durchaus auch auf den Tisch blicken kann, z.B. beim Schreiben, so soll getestet werden wie weit die Analyse in solchen Situationen funktioniert.

Versuchsaufbau

Für diesen Versuch wurde die Kamera auf *1.88m* Höhe und *3m* vor den vordersten Standort der Probanden aufgestellt.

Als Standorte wurde eine Markierung mit einem Meter Abstand zueinander auf eine Gerade bei *3m* und *9m* verwendet.

Als Target diente die Kamera, ein Punkt *78cm* unterhalb der Kamera und einer *40cm* über dem Boden und *50cm* vor der Kamera. Alle anderen Targets befinden sich *1m* vor den Standorten. Diesmal war das Versuchsgelände draußen an einem bedeckten Tag, wodurch eine helle schattenlose Szene entsteht.

Auswertung

ToDo

Ergebnisse

Es zeigt sich, dass eine Videoanalyse auch bei starke Neigung nach unten möglich ist. Die Einzelbildauswertung liefert erneut deutlich schlechter Ergebnisse als des Videos.

Dabei funktioniert das Traking nur, wenn die Versuchsperson zuerst in die Kamera geschaut hat, um es zu beginnen. Auch die stärkere gleichmäßige Beleuchtung ist hilfreich, da die Problematik mit Gegenlicht und Schatten entsteht.

4.1.3 Auswertung der Augenpartie - Versuch 3

Um einen Eindruck von ElSe mit hochauflösenden Aufnahmen zu erhalten, wurde mit einer Fotokamera (Sony ILCE-6000, Farbbild 6000×4000 Pixel, Brennweite *16mm*) an den selben Positionen wie in Versuch 1 ein weiterer Datensatz von Einzelbilder erstellt, dabei wurden nur Aufnahmen mit der Kamera als Target gemacht. Von Interesse ist die Augenpartie und die

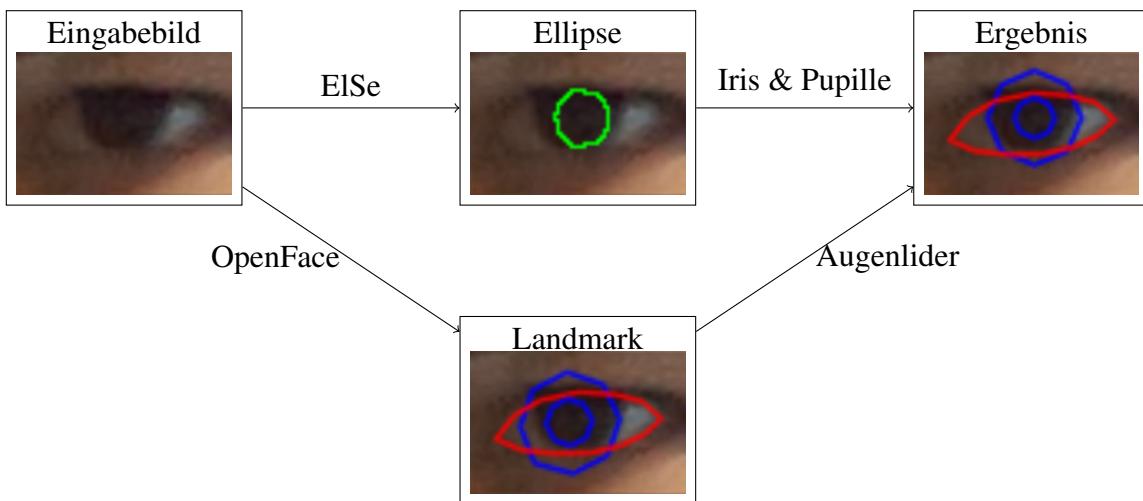


Abbildung 4.4: Dargestellt sind der Ablauf, um die Landmarks des Auges zu verbessern

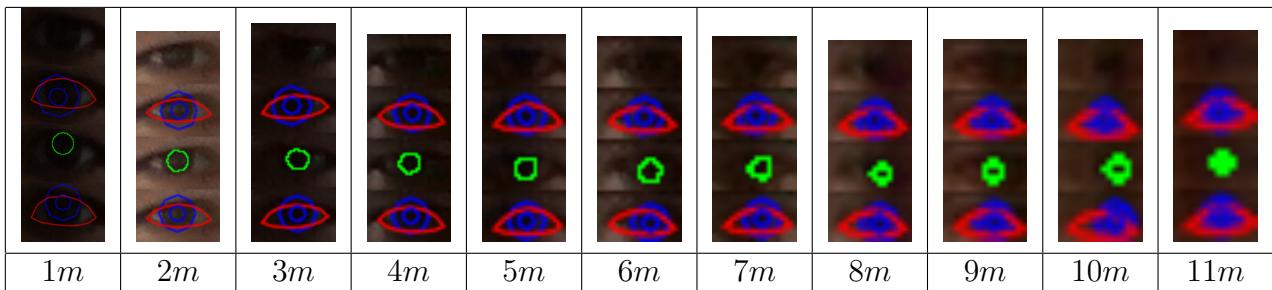


Abbildung 4.5: Ergebnisse von OpenFace und ElSe bei verschiedenen Distanz.

Von Oben nach Unten: Augenparie, Ergebnis OpenFace, Ergebnis ElSe, ElSe Ergebnis als Landmarks

Ergebnisse des OpenFace Eye-Detektor im Vergleich zu ElSe.

Dabei wurde ElSe in der Basis Konfiguration eingesetzt, dies bedeutet das Luminance-Verfahren, siehe Unterabschnitt 3.6.3 als Graukonvertierer und einem Radius der Maske von 12 Pixel.

Auswertung

Für die Analyse wurde zuerst mit OpenFace das Gesicht soweit analysiert um die Augenpartie als Eingabebild zu bestimmen, siehe Abbildung 4.4 und ein Beispiel in Abbildung 4.5 oben. Auf diesem Eingabebild wird nun der ElSe-Algorithmus angewendet um die Ellipse zu bestimmen, dargestellt in grün. Im Vergleich sind die zusätzlichen 28 Landmarks der Augen von OpenFace auch in Abbildung 4.5 Mitte oben. Als Ergebnis wurde aus den berechneten Ellipse von ElSe die Landmarks der Pupille und Iris abgeleitet und im selben Farbschema dargestellt.

Die einzelnen Augenpaare stammen von der selben Person, die sich bei der angegebenen Distanz frontal vor der Kamera befand. Es ist zu erkennen, dass selbst bei einer hohen Auflösung die Augenpartie sehr klein ausfällt und nur schwierig auszuwerten ist.

Ergebnis

Es zeigt sich das trotz einer hohen Bildauflösung der Informationsgehalt auf größere Distanzen deutlich abnimmt, wenn mit einer einzigen Kamera der gesamte Bereich einer Klasse erfasst werden soll. Außerdem ist auch gut zu erkennen das eine ausreichende Beleuchtung gebracht wird, da die Augenregion sehr dunkel ausfällt.

4.1.4 Ergebnis der Vorversuche

Es zeigt sich das der Arbeitsbereich in Hinblick auf Rotationen ausreichend ist um alle üblichen Bewegungen eines Schülers zu erfassen. Auch die Fläche auf dem sich die Schüler verteilen können ist vielversprechend, nur die Distanz muss noch verbessert werden.

Auch MTCNN-Face ist als Detektor geeignet, er findet zuverlässige alle Gesichter im Frame, unabhängig ihrer Größe und Orientierung. Sogar jene die von OpenFace auch bei der Videoanalyse nicht verwendbar sind. Einzige Anmerkung ist die etwas ungenaue Box, dies kann aber mit einer einfachen Verschiebung der Boxränder korrigiert werden.

4.2 Aufmerksamkeitsmessung - Versuch

Für den Versuch wurde ein Video verwendet, welches ein bewegtes Kreuz zeigt, das als Ziel der Aufmerksamkeit dient. Dieses Kreuz sollten die Probanden normal im Auge behalten, damit für jeden Zeitpunkt bekannt ist wo das Ziel der Aufmerksamkeit liegt.

Die Anordnung der Eckpunkte des bewegten Ziels sind in Abbildung 4.7 dargestellt und wurden mittels eines Projektors auf eine Größe von $2.88 \times 1.49m$ gebracht.

Das Ziel welches betrachtet werden soll (Target) beginnt immer in der Mitte und bleibt dort 1s stehen, bewegt sich innerhalb von 4 Sekunden zu einen der Randpunkte, verweilt dort für eine Sekunde und begibt sich in 4s zu einem nächstgelegenen Randpunkt, bleibt dort 1s und geht zurück zum Zentrum, dies wiederholt sich für alle Eckpunkte. Ein gesamter Durchlauf dauert 2min und 1s.

Die Versuchspersonen befinden sich etwa 1.5m vor der Leinwand, die Kamera befand sich 24cm unterhalb und 12.5cm vor dem zentralen Punkt des Targets mit Blickrichtung zum Projektor und Personen, siehe Abbildung 4.6.

4.2.1 Versuchsdurchführung

Um die ungefähre Position des Kopfes relativ zur Leinwand zu bestimmen, wurde die Distanz zwischen der Stirn am Nasenrücken und den 4 Eckpunkten durch einen Laserdistanzmessers bestimmt und trianguliert. Während der Aufnahme wurde auf weitere Messung der exakten Position verzichtet.

Es wurden 8 Videos von 6 Probanden (5 Männlich, 1 Weiblich, 3 mit Brille und 5 ohne Brille)

4.2 Aufmerksamkeitsmessung - Versuch



Abbildung 4.6: Foto der Versuchsdurchführung

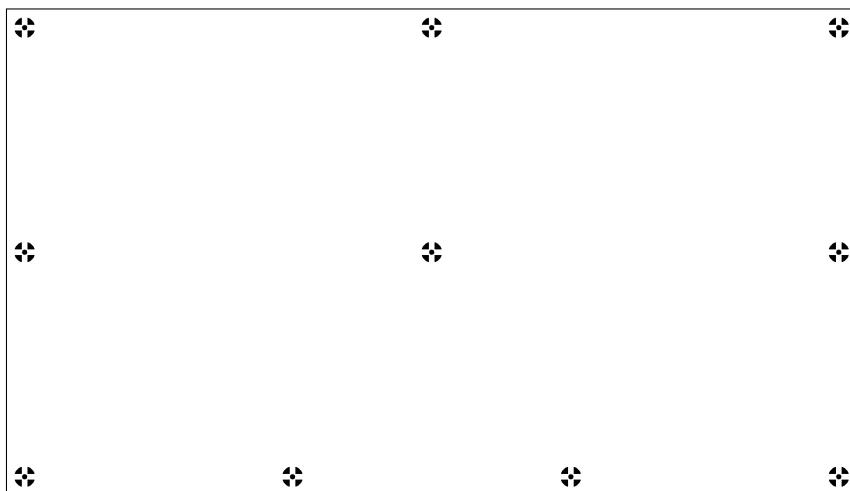


Abbildung 4.7: Eckpositionen des Bewegten Zieles bei der Videoaufnahme

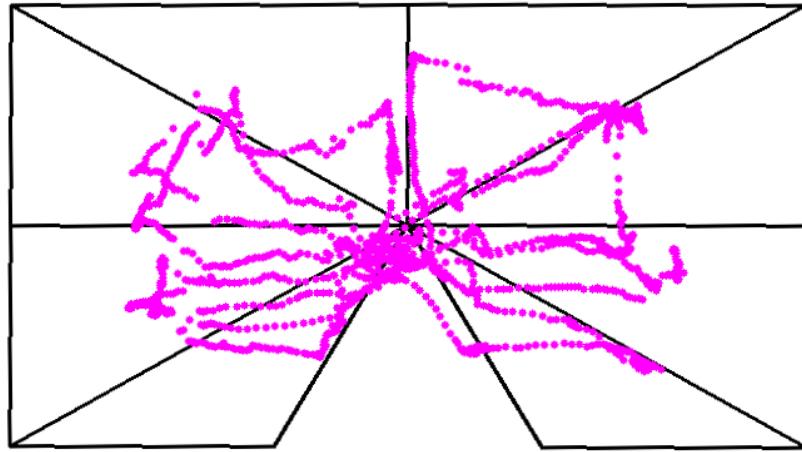


Abbildung 4.8: Dargestellt sind alle gemessene Auftreffpunkte der Gesichtsorientierung auf die Leinwand (Rosa) und des Targets (Schwarz)

erstellt.

Um die Bewegung des Targets mit der Aufzeichnung der Kopfbewegung zu synchronisieren, war im Kamerabild der duplizierte Bildschirm zum Projektorbild zusehen.

Die Aufnahmen wurden mit der Logitech-Webcam Abschnitt 3.2 erstellt.

Erster Eindruck

Dargestellt in Abbildung 4.8 sind alle Auftreffpunkte der Blickrichtung auf die Leinwand während der gesamten Aufnahme.

Es ist zu erkennen, dass die eigentlichen Kopfbewegungen sichtbar sind, es aber vor allem in den Randbereichen zu einer großen Differenz kommt.

Qualität

Durch die begrenzte Auflösung der Kamera und dem großen Distanzbereich auf dem gearbeitet werden muss, ist vor allem die Stabilität bezüglich Skalierung wichtig.

Bei der Bestimmung des horizontalen Winkels der Kopforientierung zeigt sich das die berechneten Werte im Schnitt etwas zu gering ausfallen. Die Orientierung in Richtung Kamera kann zuverlässig bestimmt werden, ebenso wenn der Proband seinen Kopf in eine Richtung dreht. Dabei wird der Fehler um so stärker je größer der zu messende Winkel wird. Betrachtet man in der Originalgröße die jeweiligen Quartale (Abbildung 4.9), so sind diese etwa 5° auseinander. Genug um einzelne Bereiche differenzieren zu können.

Bei der Bestimmung des vertikalen Winkels zeigt sich, das dieser Wert nur sehr ungenau bestimmt werden konnte, vor allem der Winkel nach Oben ist fast nicht messbar. Jener Richtung Boden wird besser erfasst, allerdings ist, bedingt durch den Versuchsausbau, der Wertebereich

4.2 Aufmerksamkeitsmessung - Versuch

recht gering.

Die bestimmte Blickrichtung ist trotz Verbesserung durch ElSe und Mittlung beider Augen, schon in der Originalgröße nur begrenzt verwendbar. Die Mittelwerte liegen selbst bei den Maximal Werten sehr eng beieinander und die Bereiche überschneiden sich stark. Die Differenz der Mittelwerte zwischen den Extremar sind nur etwa 20° weit auseinander, dabei liegen diese Punkte im Original etwa 90° weit auseinander, mit dieser Verteilung ist eine mittlere Abweichung von $17,5^\circ$.

Die Auswirkung der Skalierung ist hinnehmbar gering, allgemein steigt die Abweichung und der Bereich einer erfolgreichen Detektion sinkt. Bei einem Skalierungsfaktor von 0.01 können die einzelnen Bereiche noch gut getrennt werden, siehe Abbildung 4.9, dies entspricht eine Distanz von etwa $14m$. Auf der horizontalen Achse liegt der Abstand der Quartale etwa 9° weit auseinander, nur 4° mehr als im Original. Bei der Bestimmung des vertikalen Winkels ergibt sich ein ähnliches Verhalten, wobei vor allem der Wertebereich auf 30° sinkt.

Das Ergebnis der Blickrichtung kann bei der 0.01 Skalierung nicht verwendet werden, da die Differenz zwischen dem Rechten und Linken Maximalwert nur 8° beträgt und die Quartale sich fast vollständig überschneiden.

Überraschend ist das Ergebnis bei dem Skalierungsfaktor von 0.05 (ca $24m$). Die Ausrichtungen sind, zumindest horizontal, noch erkennbar und soweit differenzierbar um grobe Richtungsänderungen zu erkennen. Allerdings ist die Detektionsrate sehr gering und kann als Obergrenze angenommen werden.

Die Auswertung des Versuches hat die Erwartungen und Problematiken aus den Vorversuchen bestätigt. Eine Verarbeitung des Videomaterials ist sogar bei sehr niedriger Auflösung noch möglich, wobei die Ergebnisse besser sein könnte. Die Abweichung der Einzelnen Messungen ist in Abbildung 6.13 dargestellt.

4 Evaluation

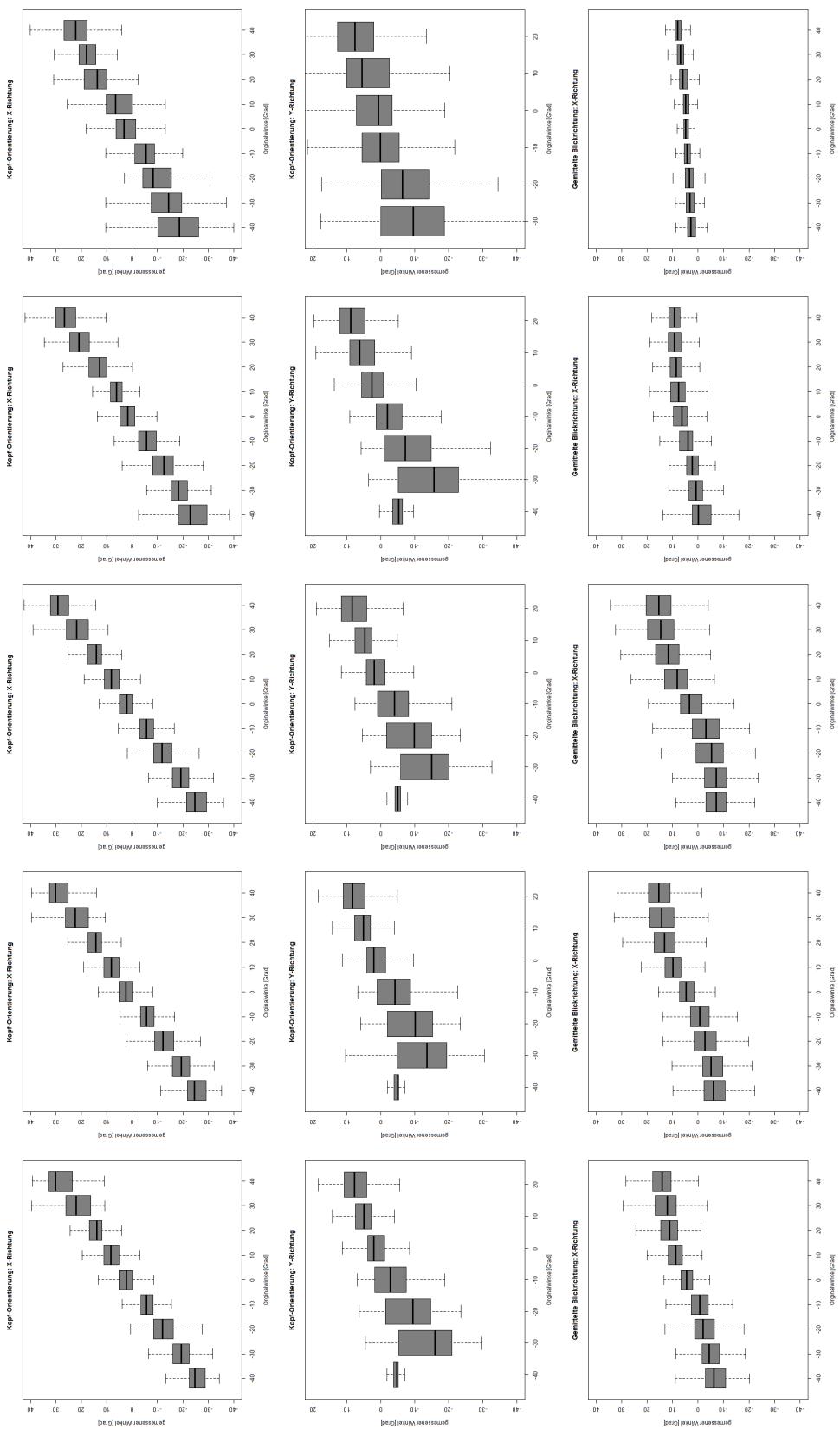


Abbildung 4.9: Auswertung der Videoaufnahme mit der Kopfausrichtung Horizontal (Oben), Kopforientierung Vertikal (Mitte) und die X-Ausrichtung der Augen (Unten)
Skalierungsfaktor von links nach rechts (1/0.5/0.25/0.1/0.05), Y-Achse: $[0 - 35]^\circ$

4.2.2 Fehleranalyse im Versuch

Eine Betrachtung der Fehlerquellen die bei der Messung entstanden sind bzw. die durch den Aufbau entstehen, sowie bei der Berechnung.

Da nur der Unterschied zwischen Target und Auftreffpunkt der gemessenen Gesichtsorientierung aufgezeigt werden kann, kommt es zu verschiedenen Fehlern, vor allem wird das Target mit den Augen gefolgt. So wird zu Beginn der Bewegung, dem Target nur mit den Augen gefolgt, bis sich der Kopf in Bewegung setzt. Dies wird so lange fortgeführt, bis die Kopfdehnung unangenehm und das Ende der Bewegung absehbar wird. So wird der letzte Teil der Bewegung nur noch von den Augen verfolgt.

Messung

Die erste Ungenauigkeit liegt bei der Distanz zur Leinwand, diese wurde nur vor der eigentlichen Aufnahme bestimmt. Somit entsteht eine Abweichung, da die Kopfbewegung während der Aufnahme nicht erfasst wird.

Die eigentliche Messung der Distanz vom Kopf der Personen zur Leinwand ist ebenfalls ungenau, da sie eine Abweichung von etwa 1cm in alle Richtungen abweicht. Außerdem liegt der Ursprung des Kopfes in der Anwendung etwas Tiefer und weiter Hinten als der ausgemessene Nasenrücken.

Auch die Parameter für der Überführungsmatrix von Welt- nach Kamerakoordinaten sowie die Brennweite wurden zwar sorgsam bestimmt, sind aber dennoch nicht perfekt.

Bedingt durch den Aufbau und der verwendeten Hardware, musste die Kamera in Richtung des Projektors ausgerichtet werden, wodurch diese vor dem direkten Licht geschützt werden muss. Somit konnte sich die Kamera nicht im Zentrum der Messpunkte befinden.

Da Kamera und Leinwand fest montiert sind, ergibt sich auch die Problematik das der Kopf der Probanden nicht im Zentrum des Kamerabildes sind und somit hat die Kamera immer einen Blickwinkel von unten auf das Gesicht.

Da die Probanden ebenfalls zwischen der Leinwand und dem Projektor standen, verdeckten diese das Bild, wodurch es manchmal passierte das der Zielpunkt im Schatten verschwand und keine zentrale Messung mit Blickrichtung nach unten möglich ist.

Umgebung

Bei der Aufzeichnung hat sich vor allem das Problem mit der ungleichmäßigen Beleuchtung bzw. dem Gegenlicht ergeben. Diesem wurde durch Abdunkeln der Fenster und Verwendung der Tafelbeleuchtung entgegengewirkt, damit das Gesicht gut erkennbar ist. Ein Problem das auch in der realen Anwendung auftreten wird.

Ein weiteres allgemeines Problem ist die Anzahl der Bildpunkte des Gesichtes im Bild, somit ist eine Berechnung auf dem Gesicht zwar möglich, auf den Augen allerdings nur bedingt. Außerdem ergibt sich ein weiteres Problem mir der Augenbewegung, die nicht erfasst wird.

4 Evaluation

Die allgemeine Exkursionen beträgt etwa 20° [31], der Winkelbereich der üblichen Augenbewegungen, und kann daher recht stark von der Kopforientierung abweichen.

Ein weiteres nicht zu verachtendes Problem ist die Reflektion, vor allem auf den Brillen, die die Pupille überdecken, von den starken Lichtquellen wie z.B. Fenster, Projektor und dessen Bild, sowie Lampen usw. Auch Schatten gerade in den Augenhöhlen erschweren die Auswertung.

4.3 Fehleranalyse

Mit entsprechend hochauflösenden Kameras, können auch bessere Resultate auf größeren Distanzen erzielt werden. Gerade die Bestimmung der Blickrichtung auf großer Distanz ist meist nicht möglich, da die Augenpartie viel zu klein für eine Berechnung ist. So bleibt meist nur die Gesichtsorientierung mit ihrer natürlichen Ungenauigkeit.

Da Bewegung erlaubt ist, passiert es immer wieder, dass Teile des Gesichtes verdeckt werden, durch Hände beim Melden, andere Schüler oder dem Lehrer selbst, der vor der Kamera steht oder sich der Kopf zu weit wegdreht und das Tracking scheitert. Aber auch die Frisuren spielen eine Rolle, da dadurch diese einige Landmarks verdeckt werden können, wie z.B. die Augenbrauen, und das Gesicht nicht erkannt wird .

4.4 Zusammenfassung

Für die Analyse der Gesichter in einem Video, wurden zuerst die einzelnen Gesichtern mittels MTCNN-Face Detection (Abschnitt 3.3) in allen vorhanden Frames gesucht. Dieses Verfahren ist Robust genug, das es auch kleinste Gesichter im Bild erkennen kann und auch recht stabil bezüglich der Rotation. Somit ist es als Gesichtsdetektor geeignet um aus einem Frame die Gesichter zu finden.

Anschließend wird jede Einzelperson unterscheiden und alle gefunden Bildbereiche der jeweiligen Person zugeordnet. Diese Bildbereiche werden nun auf eine Mindestgröße gebracht (Abschnitt 3.4), damit sie den Trainingsdatensatz des nächsten Schrittes stärker ähneln. Dazu wurde die Auswirkung der verschiedenen Skalierungsverfahren auf die nachfolgende Analyse untersucht.

Nun werden die einzelnen Bildbereiche Ausgewertet (Abschnitt 3.5) und die Gesichtsorientierung kann bestimmt werden. Um die Bereiche zu simulieren in denen das Verfahren eingesetzt werden kann, wurde durch lineare Skalierung die Bild des Trainingsdatensatzes verkleinert um die verschiedenen Distanzen zu simulieren.

Um die Detektion der Pupille zu verbessern wurde ElSe (Abschnitt 3.7) verwendet, mit dem Ziel, die Blickrichtung exakter zu ermitteln. Dazu wurde die Auswirkung der verschiedenen Farbbild nach Graubild Konvertierer (Abschnitt 3.6) untersucht, sowie die Veränderung des Radius der Maske.

Abschließend wurde getestet, wie zuverlässig das gesamte Verfahren auf Videos eingesetzt werden kann, um die Aufmerksamkeit zu ermitteln, siehe Abschnitt 4.2. Dazu wurde ein

4.4 Zusammenfassung

Versuch durchgeführt, bei dem die Probanden ein Ziel verfolgen sollten und ermittelt wie exakt das ziel der Aufmerksamkeit bestimmt werden kann.

Durch den nachgewiesen Wertebereich in dem eine Auswertung möglich ist, kann das gesamte Klassenzimmer mit nur einer Kamera erfasst werden. Bei Probanden deren Blickrichtung recht stark von der Kamera abweicht, ist das Erfassen zwar möglich, allerdings starker Fehlerbehaftet.

5 Diskussion

5.1 Verbesserung

Die größte Problematik bei der Auswertung einer ganzen Schulklasse, ist, dass immer wieder Teile der Gesichter verdeckt werden, sei es durch den Arm eines anderen Schülers oder der Frisur oder völlig verdeckt durch den Lehrer und ähnliches.

Dieser Problematik kann entgegen gewirkt werden, indem mehrere Kameras verwendet werden die beispielsweise an der Seite der Tafel platziert sind. Dies bietet neben der Möglichkeit einer 3D-Rekonstruktion der Szene auch die Chance das Gesicht vollständig erfasst wird.

Durch den großen Bereich in dem das Verfahren funktioniert ist die Positionswahl der Kameras recht frei und kann so gewählt werden, dass sie die gesamte Klasse erfassen, selten verdeckt und den Unterricht wenig beeinflusst.

Für die hinteren Reihen ist der Einsatz von zusätzlichen Kameras zu empfehlen, da diese Schüler recht klein dargestellt und oft durch die vorderen Reihen verdeckt werden, sollen sie von einer Kamera erfasst werden, die vor der Klasse aufgestellt wurde.

Für eine Auswertung der Aufmerksamkeit ist die erreichte Genauigkeit ausreichend, die Tendenzen sind klar erkennbar und können entsprechend interpretiert werden.

Da der große Erfassungsbereich nur auf Videos erreicht wird, wäre es von Vorteil die Detektion und Tracking soweit zu ergänzen, dass auf Profilbilder gearbeitet werden kann um Landmarks zu erkennen. Somit kann das Tracking auch begonnen werden, wenn die Probanden nicht grob in Richtung Kamera blicken. Das ist gegenüber einer Drehung robuster.

Auch der Einsatz von Weitwinkelobjektive kann nicht empfohlen werden, da zwar mit ihrer Hilfe die gesamte Klasse erfasst werden kann, aber sehr viele Bereiche im Kamerabild nur Umgebung zeigen und die Schüler entsprechend klein dargestellt sind. Eine fokussiertere Kamera würde zwar weniger Schüler erfassen, diese wären allerdings deutlich größer darstellen und die Kamera kann passend zur Position der Schüler aufgestellt werden.

6 Abbildungen

In diesem Abschnitt werden weitere Diagramme dargestellt um einen besseren Eindruck über die Messergebnisse zu erhalten.

Boxplot

Folgende Angaben gelten für alle dargestellten Boxplots.

- Die schwarze Mittellinie in der Box zeigt den Median der Messwerte an.
- Die Box beschreibt das Ober und untere Quartal der Messwerte, also jene Stellen an denen 25% der Messwerte größer bzw. kleiner sind als der gewählte dargestellte Wert.
- „Die Whiskers (gestrichelte Linie) zeigen das Maximum bzw. Minimum einer Verteilung, sofern diese nicht mehr als das 1,5-fache des Interquartilabstands vom Median abweichen“[26]
- Alle Ausreißer wurden zwecks Übersichtlichkeit weggelassen
- Die eingezeichnete horizontale Linie stellt den Median der Messwerte aus Skalierung 1 dar, die Beschriftung gibt den Median an.

Anzahl der Messwerte

Um eine Übersicht über die Anzahl der Messwerte zu erhalten ein Überblick:

Biwi Kinect Head Pose Database

Alle Darstellungen und Auswertungen bezüglich der verschiedenen Skalierungsverfahren haben folgende Anzahl an Messwerten bei den angegebenen Skalierungen.

[Abbildung 6.1, Abbildung 6.2, Abbildung 6.3, Abbildung 6.4, Abbildung 6.5, Abbildung 6.6, Abbildung 6.7]

| | 0,04 | 0,07 | 0,1 | 0,13 | 0,16 | 0,19 | 0,22 | 0,25 | 0,28 | 0,31-1 |
|------------------|------|------|------|------|------|------|------|------|------|-------------|
| Bicubic | 3 | 1190 | 4545 | 5888 | 7147 | 8329 | 8991 | 9439 | 9561 | 9600 – 9800 |
| Lanczos | 3 | 1224 | 4206 | 5696 | 6941 | 8224 | 8958 | 9400 | 9548 | 9700 – 9800 |
| Linear | 1 | 776 | 3935 | 5439 | 6851 | 8019 | 8625 | 9107 | 9313 | 9400 – 9800 |
| Nearest-Neighbor | 0 | 0 | 0 | 193 | 2081 | 4374 | 5976 | 7825 | 8595 | 9200 – 9800 |

6 Abbildungen

Augen-Datensatz [37]

Die einzelnen abgebildeten Boxplots basierend auf dem Augen-Datensatz [37] besitze mindestens 10000 Messwerte.

[Abbildung 6.8, Abbildung 6.9, Abbildung 6.10, Abbildung 6.11]

Aufmerksamkeitsmessung - Werte im Versuch

Für diese Auswertung ergeben sich folgende Werteverteilung (Abbildung 4.9, Abbildung 6.13)

Skalierung 1:

X-Mean-Error = 8,971; Y-Mean-Error = 10.08; EyeAVG-X-Mean-Error = 17,49

| Winkel | -40 | -30 | -20 | -10 | 0 | 10 | 20 | 30 | 40 |
|--------------------|------|------|------|------|------|------|------|------|------|
| X-Rotation: Anzahl | 3115 | 1363 | 1278 | 1297 | 4142 | 1189 | 1343 | 1304 | 3449 |
| Y-Rotation: Anzahl | 444 | 3328 | 1920 | 5692 | 2215 | 1804 | 3077 | | |

Skalierung 0.5:

X-Mean-Error = 8,927; Y-Mean-Error = 10,07; EyeAVG-X-Mean-Error = 17,07

| Winkel | -40 | -30 | -20 | -10 | 0 | 10 | 20 | 30 | 40 |
|--------------------|------|------|------|------|------|------|------|------|------|
| X-Rotation: Anzahl | 2420 | 1068 | 1002 | 1002 | 3217 | 932 | 1070 | 1023 | 2720 |
| Y-Rotation: Anzahl | 222 | 2649 | 1506 | 4372 | 1819 | 1415 | 2471 | | |

Skalierung 0.25:

X-Mean-Error = 8,742; Y-Mean-Error = 9,772; EyeAVG-X-Mean-Error = 17,07

| Winkel | -40 | -30 | -20 | -10 | 0 | 10 | 20 | 30 | 40 |
|--------------------|------|------|------|------|------|------|------|------|------|
| X-Rotation: Anzahl | 2471 | 1074 | 1012 | 1018 | 3283 | 950 | 1077 | 1047 | 2749 |
| Y-Rotation: Anzahl | 222 | 2753 | 1536 | 4452 | 1831 | 1417 | 2470 | | |

Skalierung 0.1:

X-Mean-Error = 9,899; Y-Mean-Error = 10,14; EyeAVG-X-Mean-Error = 21.31

| Winkel | -40 | -30 | -20 | -10 | 0 | 10 | 20 | 30 | 40 |
|--------------------|------|------|------|------|------|------|------|------|------|
| X-Rotation: Anzahl | 2466 | 1064 | 1002 | 1018 | 3283 | 950 | 1074 | 1047 | 2727 |
| Y-Rotation: Anzahl | 222 | 2734 | 1517 | 4451 | 1829 | 1417 | 2461 | | |

Skalierung 0.05:

X-Mean-Error = 12.48 Y-Mean-Error = 12.48; EyeAVG-X-Mean-Error = 21.48

| Winkel | -40 | -30 | -20 | -10 | 0 | 10 | 20 | 30 | 40 |
|--------------------|------|------|-----|------|------|-----|------|-----|------|
| X-Rotation: Anzahl | 1231 | 575 | 550 | 589 | 1859 | 557 | 607 | 530 | 1335 |
| Y-Rotation: Anzahl | | 1151 | 661 | 2419 | 1217 | 817 | 1568 | | |

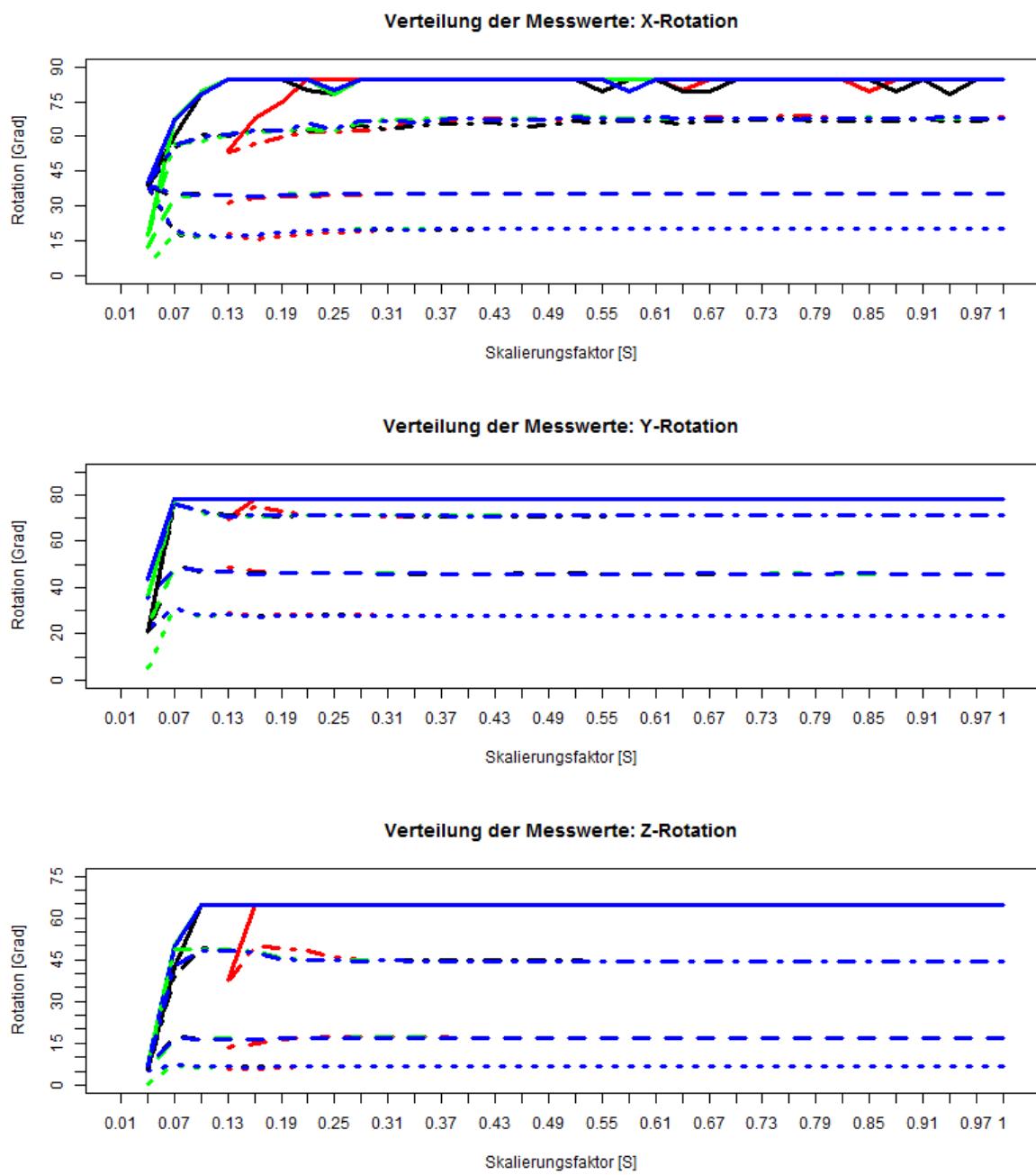


Abbildung 6.1: Dargestellt ist der Bereich in denen im Biwi Kinect Head Pose Database [10] ein Gesicht erkannt wurde.

Bicubic (blau), Lanczos (grün), Linear (schwarz), Nearest-Neighbor (rot)

Maximal erreichter Wert: _____

99,5% Quantile der Messwerte: _____

80% Quantile der Messwerte: _____

Median aus den Messwerten: _____

6 Abbildungen

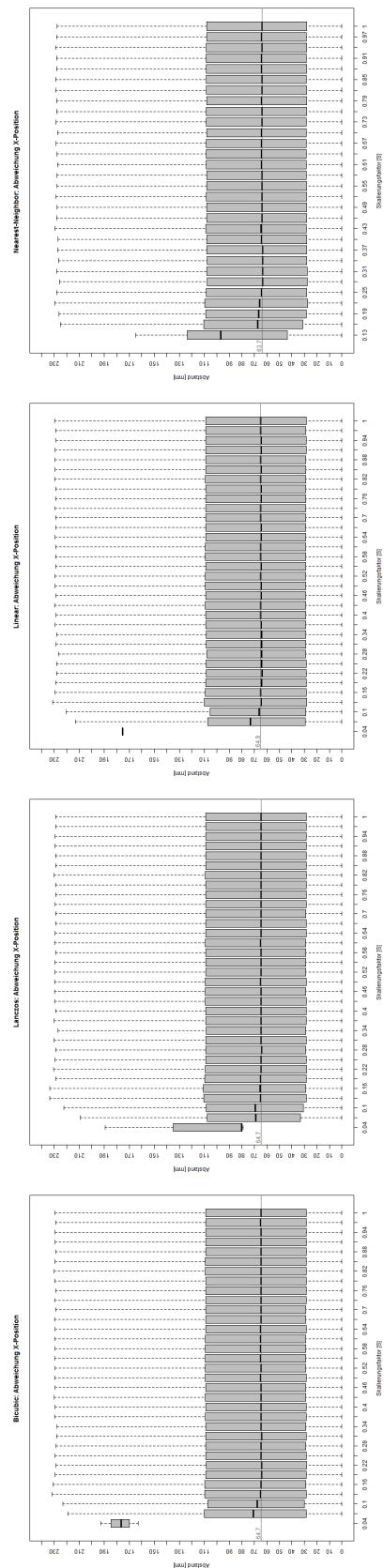


Abbildung 6.2: Zusammenhang zwischen der Skalierung und der Abweichung in X-Richtung.
Von rechts nach links: Bicubic, Lanczos, Linear, Nearest-Neighbor

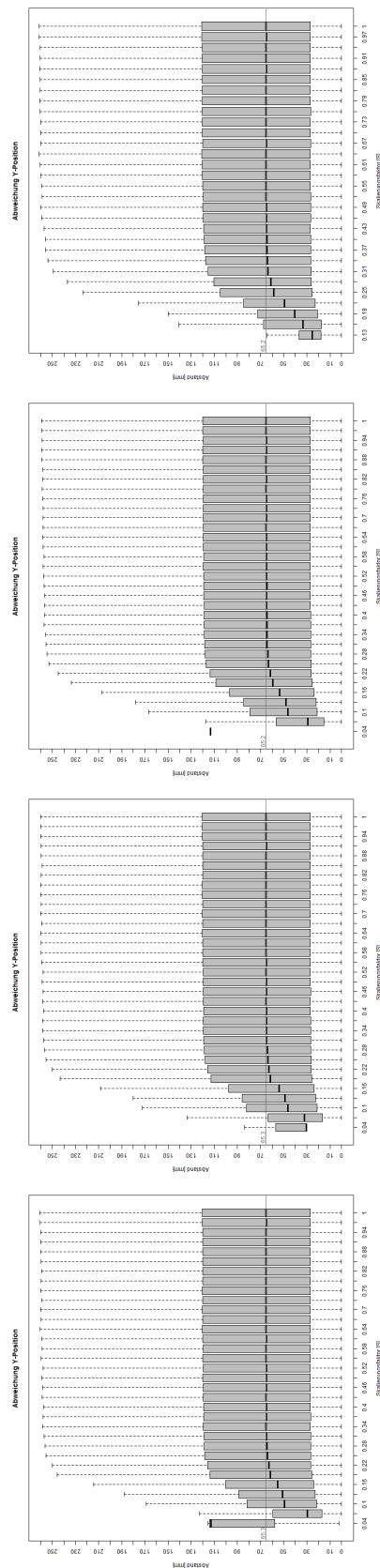


Abbildung 6.3: Zusammenhang zwischen der Skalierung und der Abweichung in Y-Richtung.
Von rechts nach links: Bicubic, Lanczos, Linear, Nearest-Neighbor

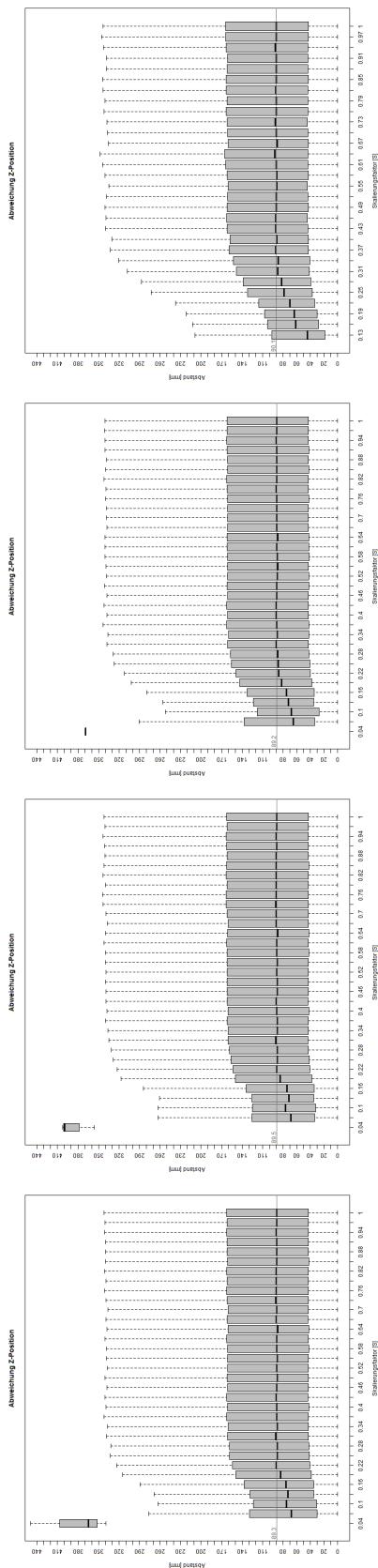


Abbildung 6.4: Zusammenhang zwischen der Skalierung und der Abweichung in Z-Richtung in Millimeter.
Von rechts nach links: Bicubic, Lanczos, Linear, Nearest-Neighbor

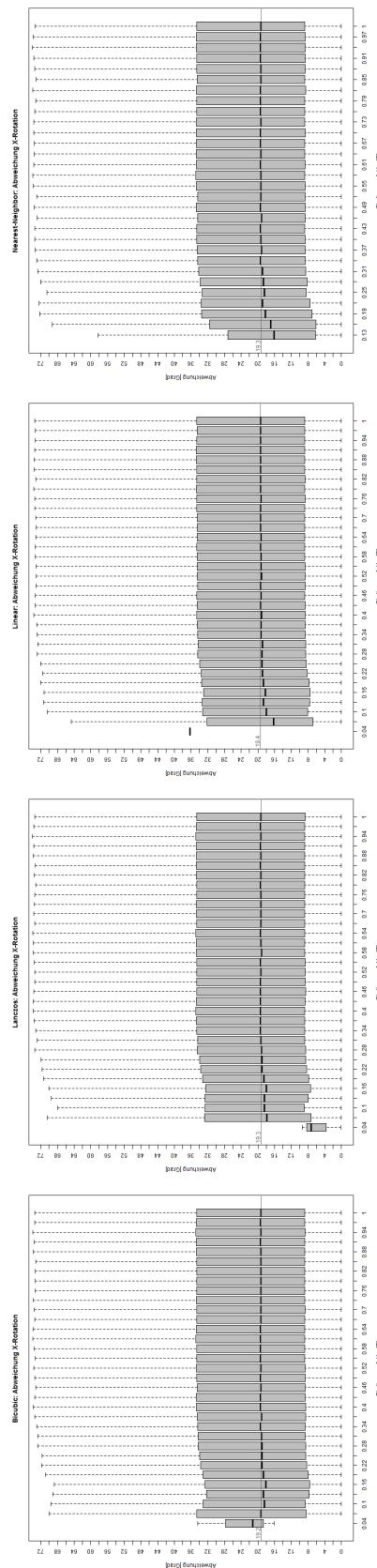


Abbildung 6.5: Zusammenhang zwischen der Skalierung und der Abweichung des Winkels in X-Richtung, Angabe in Bogenmaß.
Von rechts nach links: Bicubic, Lanczos, Linear, Nearest-Neighbor

6 Abbildungen

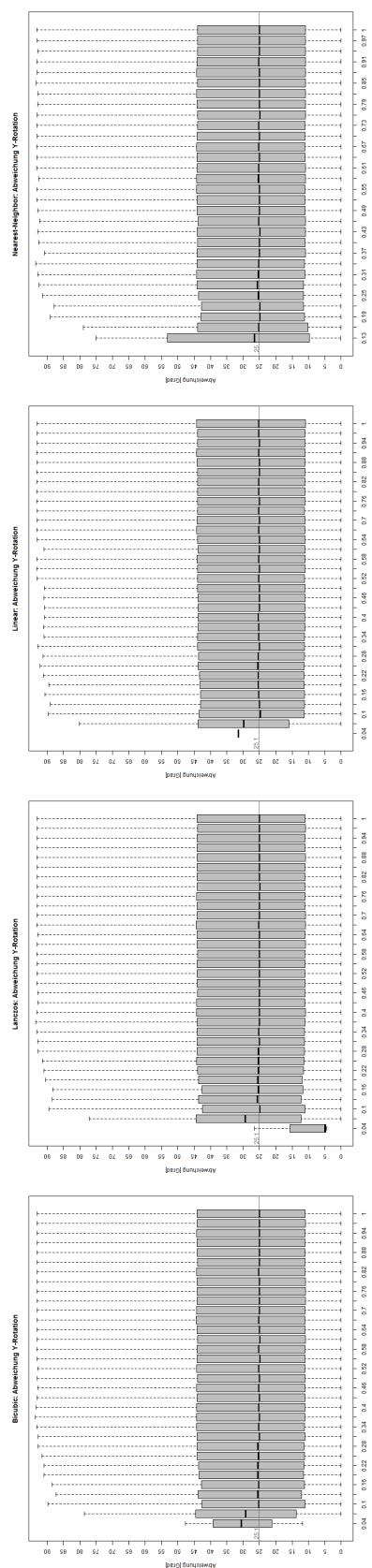


Abbildung 6.6: Zusammenhang zwischen der Skalierung (X-Achse) und der Abweichung des Winkels in X-Richtung, Angabe in Bogenmaß.
Von rechts nach links: Bicubic, Lanczos, Linear, Nearest-Neighbor

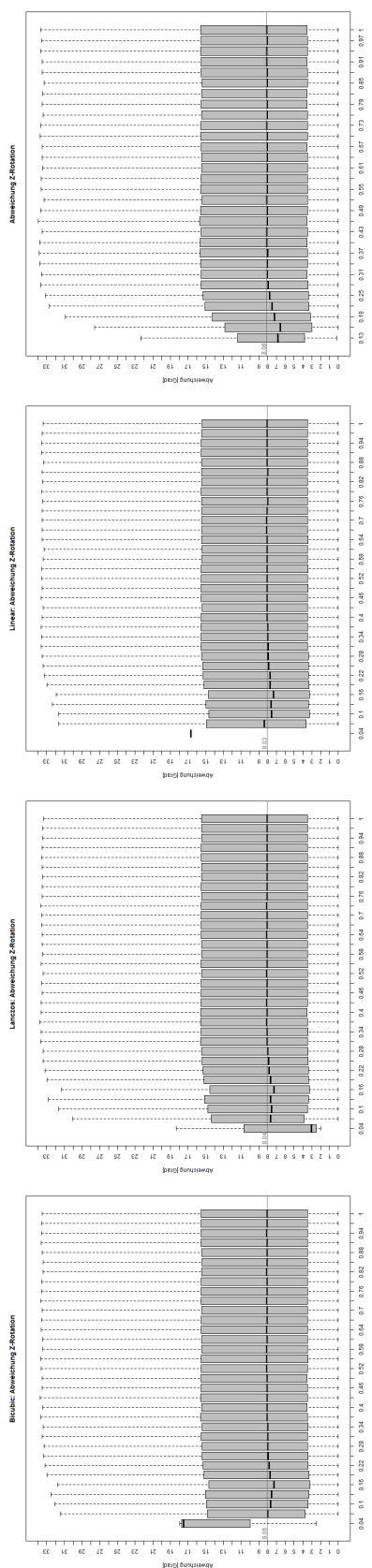


Abbildung 6.7: Zusammenhang zwischen der Skalierung (X-Achse) und der Abweichung des Winkels in Y-Richtung, Angabe in Bogenmaß.
Von rechts nach links: Bicubic, Lanczos, Linear, Nearest-Neighbor

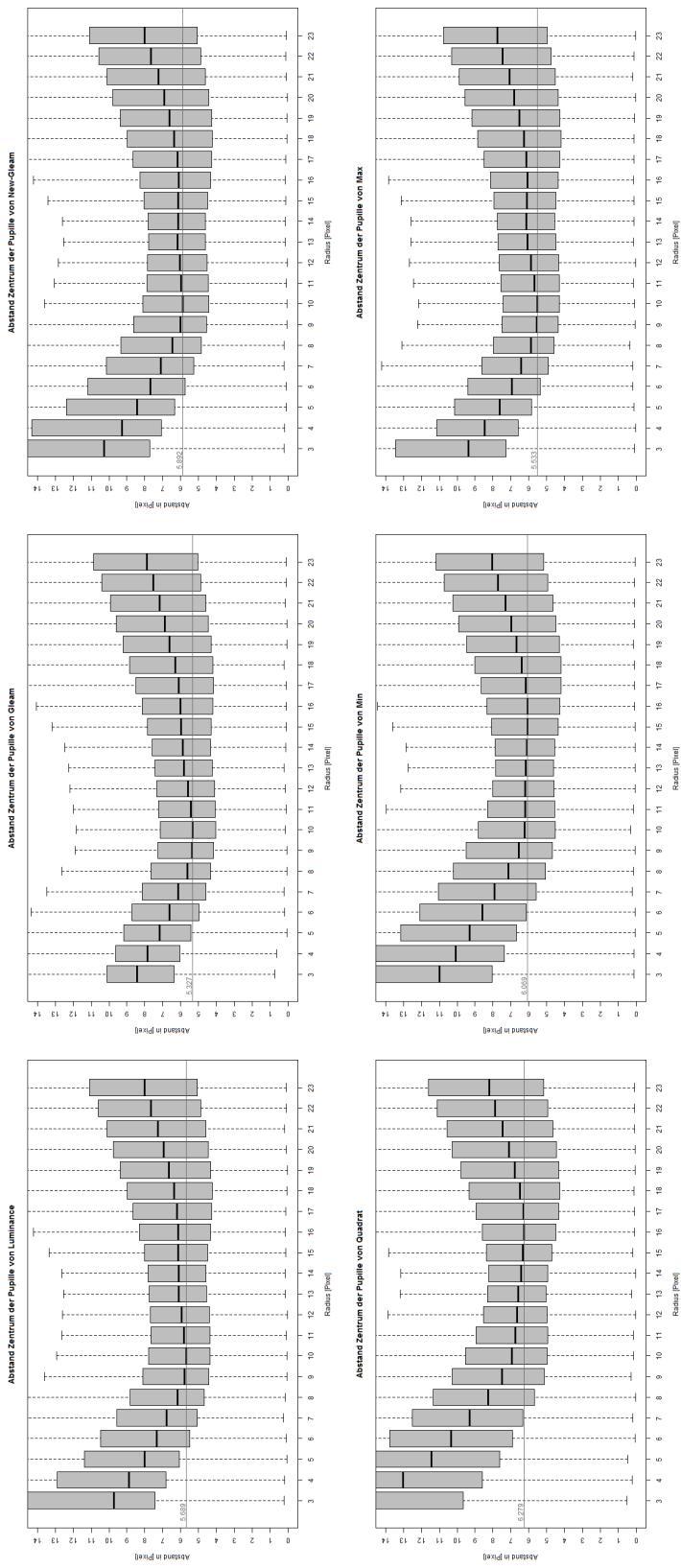


Abbildung 6.8: Abstand des Zentrums der Landmark-Pupille und der berechneten Ellipse in [Pixel] gegen den Radius-Größe des Filters.

Oben-Links: Luminance, Oben-Mitte: Gleam, Oben-Rechts: Gleam New,
Unten-Links: Quadrat, Unten-Mitte: Min-Mitte, Unten-Rechts: Max-Wert

6 Abbildungen

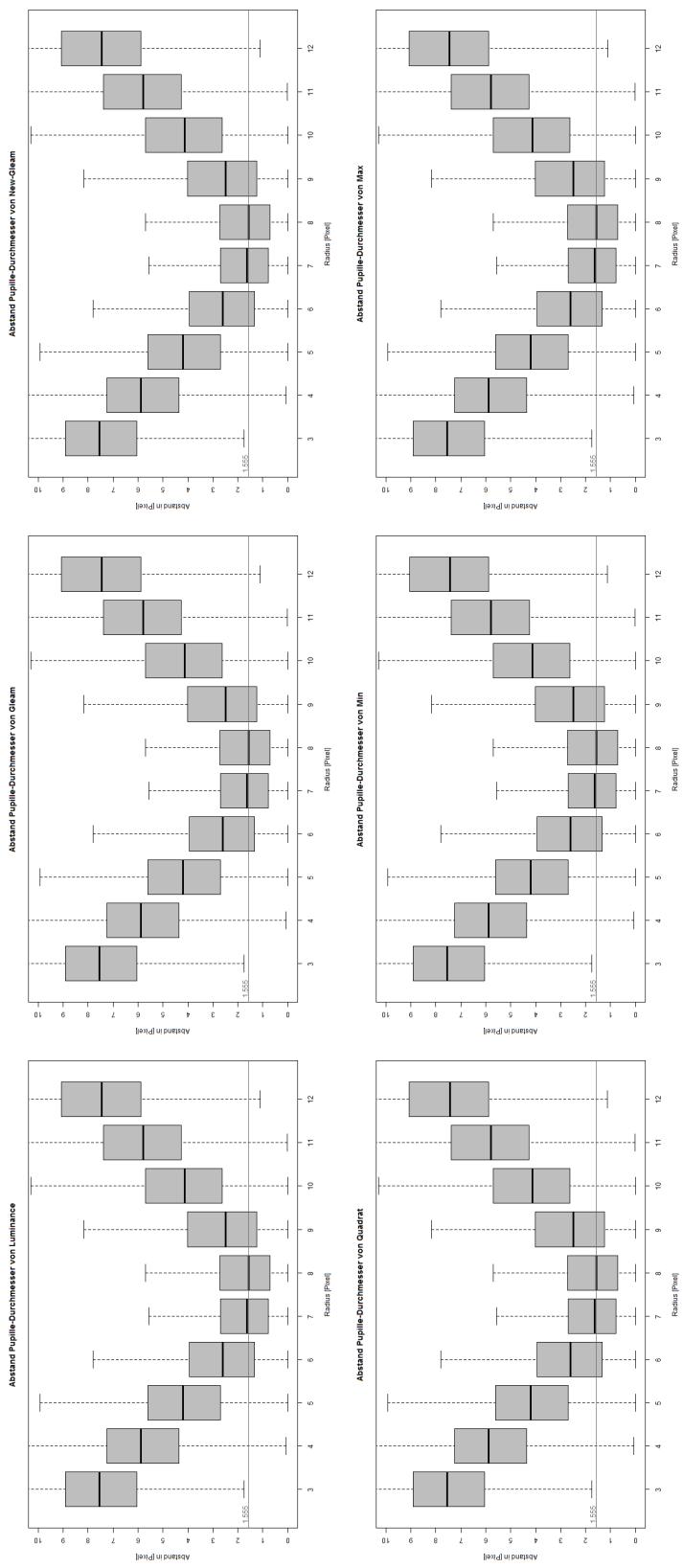


Abbildung 6.9: Unterschied Zwischen den Radien der Landmark-Pupille und der Berechneten Ellipse in [Pixel] gegen den Radius-Größe des Filters
 Oben-Links: Luminance, Oben-Mitte: Gleam, Oben-Rechts: Gleam New,
 Unten-Links: Quadrat, Unten-Mitte: Min-Wert, Unten-Rechts: Max-Wert

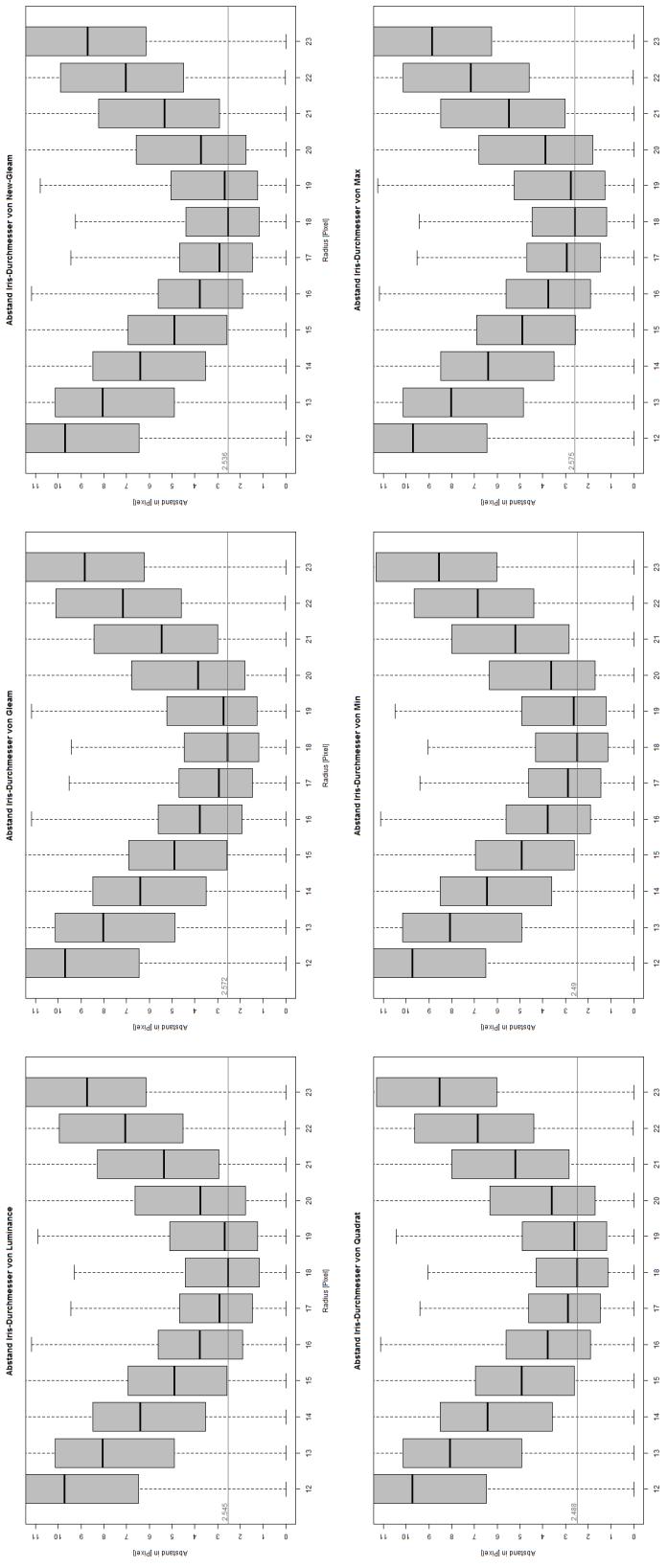


Abbildung 6.10: Unterschied Zwischen den Radien der Landmark-Iris und der Berechneten Ellipse in [Pixel] gegen den Radius-Größe des Filters.

Oben-Links: Luminance, Oben-Mitte: Gleam, Oben-Rechts: Gleam New,
Unten-Links: Quadrat, Unten-Mitte: Min-Wert, Unten-Rechts: Max-Wert

6 Abbildungen

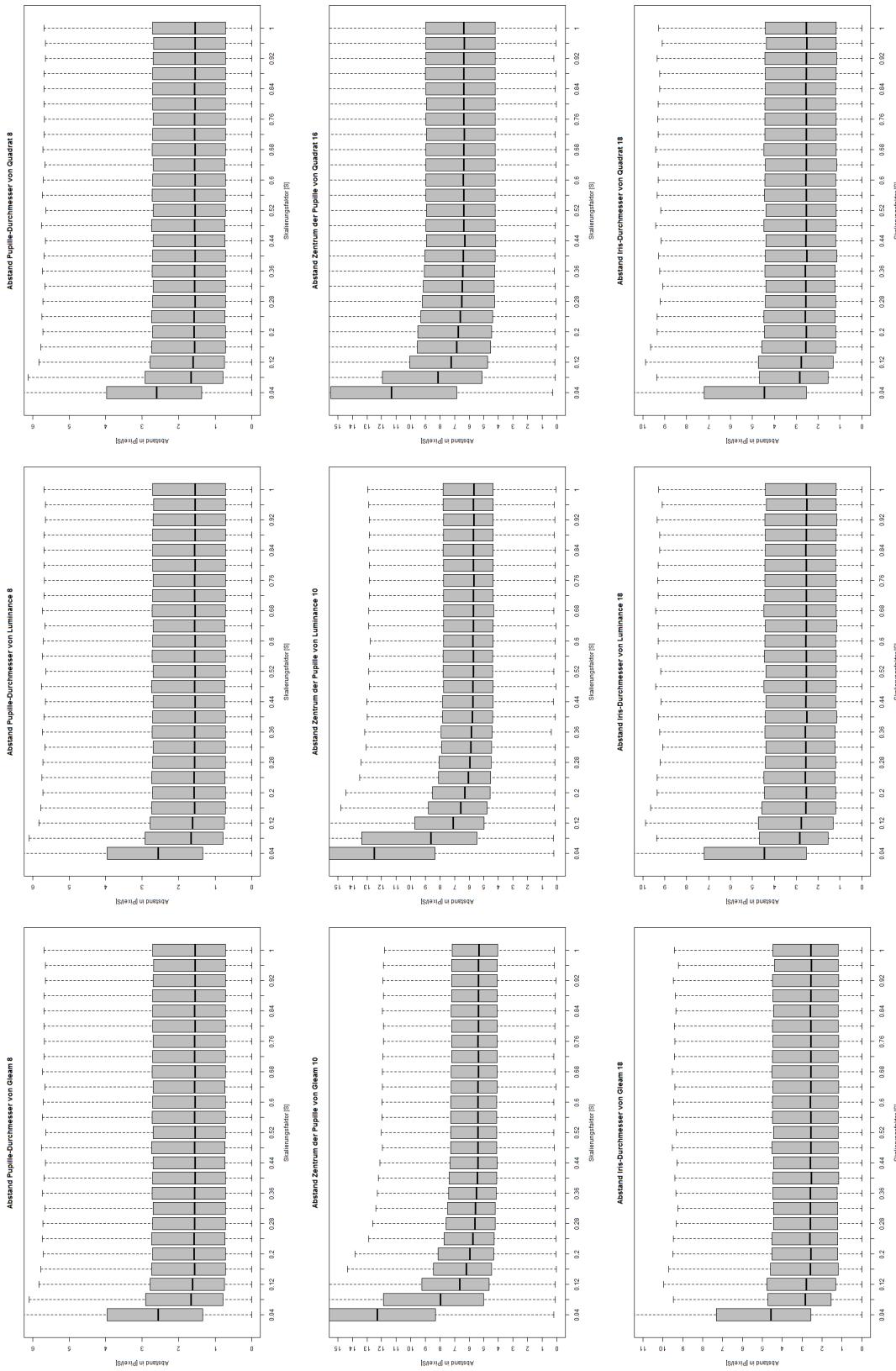


Abbildung 6.11: Auswirkung von der Bildgröße auf die Qualität der Berechnung. Aufgetragen ist die Abweichung [Pixel/Skalierung] gegen den Skalierungsfaktor. Oben: Pupille-Durchmesser, Mitte Abweichung Zentrum, Iris-Durchmesser
 Links: Gleam, Mitte: Luminance, Rechts Quadrat

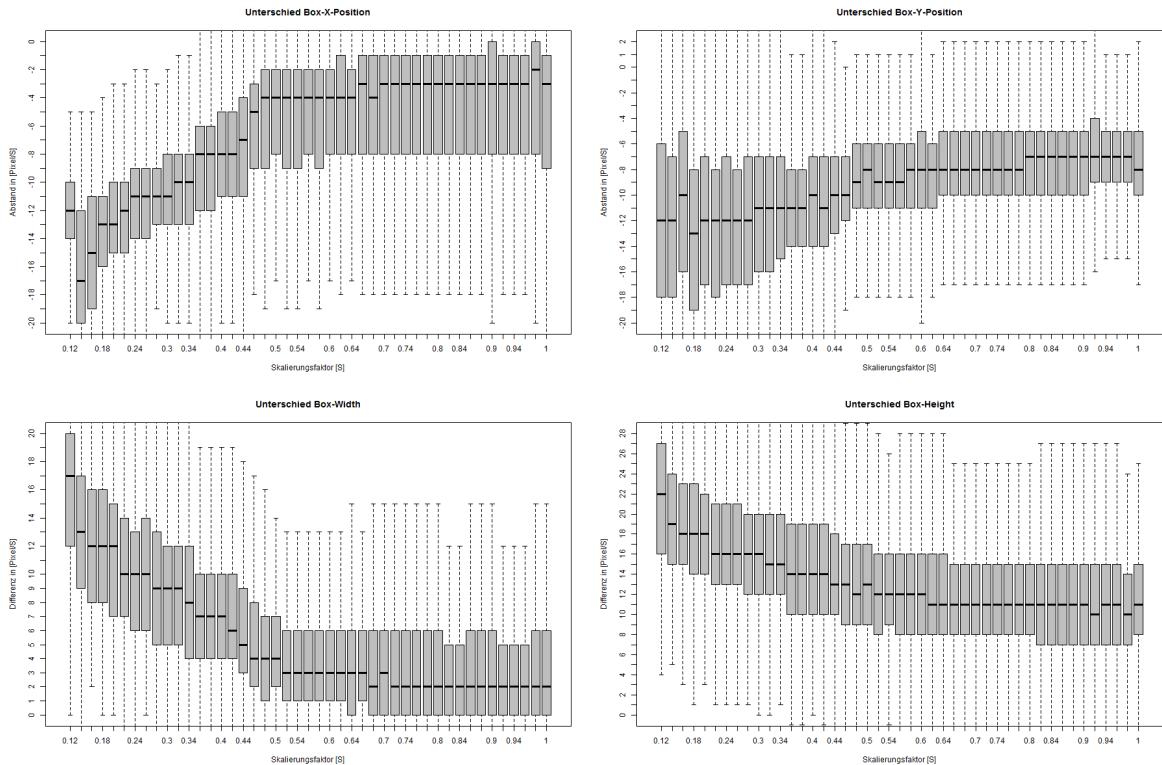


Abbildung 6.12: Bestimmung der Box ums Auge abhängig von der Bildgröße. Aufgetragen ist die Abweichung [Pixel/Skalierung] gegen den Skalierungsfaktor.

Dargestellt sind Koordinaten, X- und Y-Position in Pixel sowie die Ausdehnung der Box (Width und Height) ebenfalls in Pixel relativ zur umschließenden Box der Landmarks.

6 Abbildungen

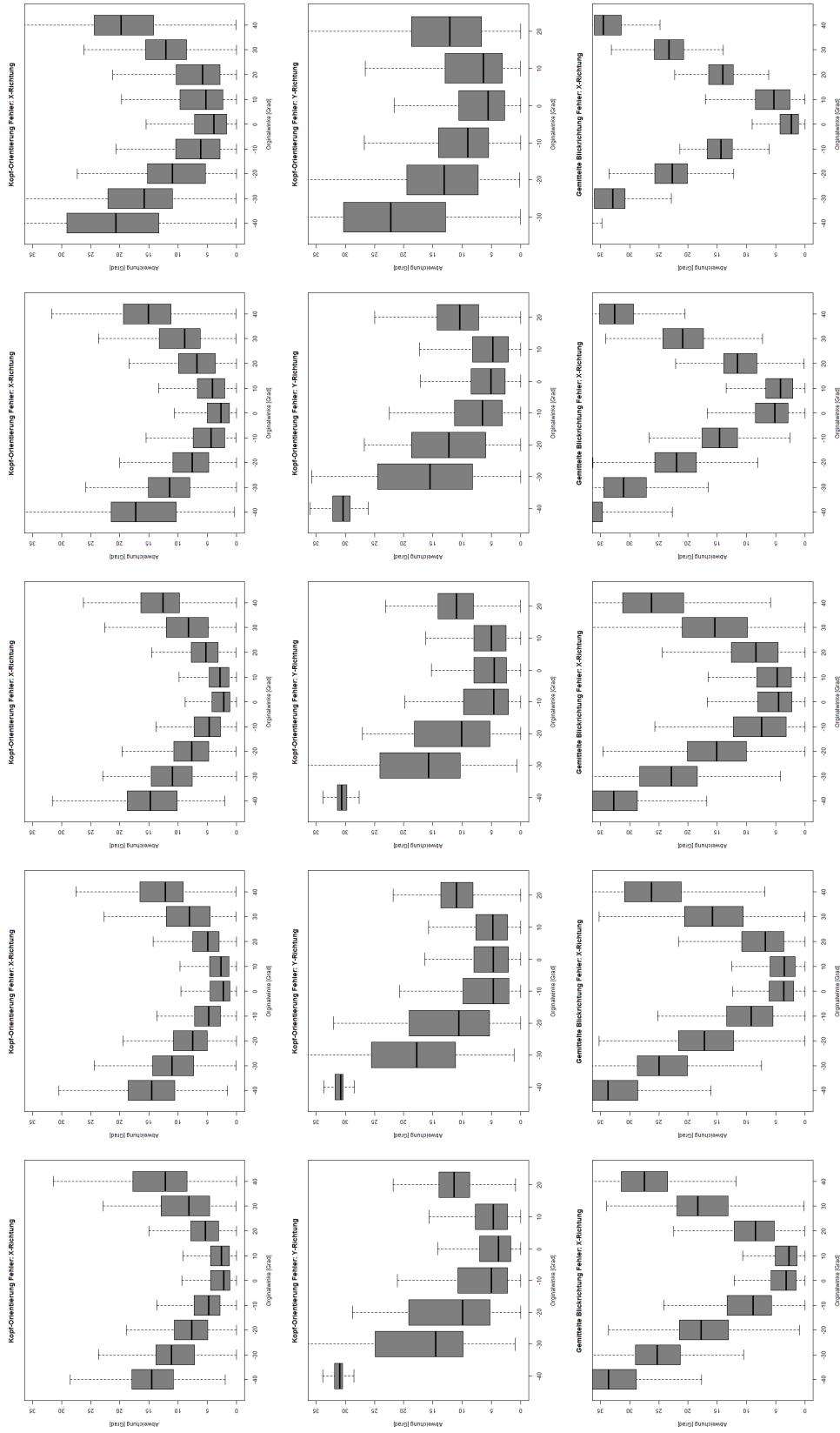


Abbildung 6.13: Abweichung der Videoaufnahme von der Kopfausrichtung Horizontal (Oben), Kopforientierung Vertikal (Mitte) und die X-Ausrichtung der Augen (Unten)
Skalierungsfaktor von links nach rechts (1/0.5/0.25/0.1/0.05), Y-Achse: $[0 - 35]^\circ$

Literaturverzeichnis

- [1] Appel, Johannes: *Die Bedeutung der Aufgaben für das Beteiligungsverhalten der Schüler : eine Videostudie zur Wirksamkeit des Unterrichtsprozesses*, 2015.
- [2] Baltrušaitis, Tadas, Peter Robinson und Louis Philippe Morency: *3D Constrained Local Model for Rigid and Non-Rigid Facial Tracking*. In: *Computer Vision and Pattern Recognition (CVPR 2012)*, Providence, RI, Juni 2012. <http://ict.usc.edu/pubs/3D%20Constrained%20Local%20Model%20for%20Rigid%20and%20Non-Rigid%20Facial%20Tracking.pdf>.
- [3] Bradski, Gary und Adrian Kaehler: *Learning OpenCV*. O'Reilly Media Inc., 2008. <http://oreilly.com/catalog/9780596516130>.
- [4] Bundesrepublik Deutschland, Sekretariat der Ständigen Konferenz der Kultusminister der Länder in der: *Vorgaben für die Klassenbildung - Schuljahr 2016/2017*, August 2016. https://www.kmk.org/fileadmin/Dateien/pdf/Statistik/Klassenbildung_2016.pdf.
- [5] Cascia, Marco La, Stan Sclaroff und Vassilis Athitsos: *Fast, Reliable Head Tracking under Varying Illumination: An Approach Based on Registration of Texture-Mapped 3D Models*. IEEE Trans. Pattern Anal. Mach. Intell., 22(4):322–336, 2000. <http://dblp.uni-trier.de/db/journals/pami/pami22.html#CasciaSA00>.
- [6] Christopher Kanan, Garrison W. Cottrell: *Color-to-Grayscale: Does the Method Matter in Image Recognition?*, 2012. <https://doi.org/10.1371/journal.pone.0029740>.
- [7] Cristinacce, David und Tim Cootes: *Feature Detection and Tracking with Constrained Local Models*, 2006.
- [8] Dongheng Li, David Winfield, Derrick J. Parkhurst, 2005.
- [9] Fanelli, Gabriele, Matthias Dantone, Juergen Gall, Andrea Fossati und Luc Van Gool: *Random Forests for Real Time 3D Face Analysis*. Int. J. Comput. Vision, 101(3):437–458, February 2013.
- [10] Fanelli, Gabriele, Juergen Gall und Luc J. Van Gool: *Real time head pose estimation with random regression forests*. In: *CVPR*, Seiten 617–624. IEEE Computer Society, 2011, ISBN 978-1-4577-0394-2. <http://dblp.uni-trier.de/db/conf/cvpr/cvpr2011.html#FanelliGG11>.

Literaturverzeichnis

- [11] G.L. Masala, E. Gross: *Real time detection of driver attention: Emerging solutions based on robust iconic classifiers and dictionary of poses*, 2014.
- [12] Goutam Majumder, Mrinal Kanti Bhowmik, Debotosh Bhattacharjee, 2013.
- [13] Helmke, Andreas und Alexander Renkl: *Das Muenchener Aufmerksamkeitsinventar (MAI): Ein Instrument zur systematischen Verhaltensbeobachtung der Schueleraufmerksamkeit im Unterricht*. Diagnostica, 38(2):130–141, 1992.
- [14] Huang, Gary B., Marwan Mattar, Honglak Lee und Erik Learned-Miller: *Learning to Align from Scratch*. In: *NIPS*, 2012.
- [15] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li Yu Qiao: *Joint Face Detection and Alignment using Multi-task Cascaded Convolutional Networks*, 2015.
- [16] Kinnebrock, Werner: *Neuronale Netze: Grundlagen, Anwendungen, Beispiele*. Oldenbourg, 1994, ISBN 9783486229479.
- [17] Kultus, Jugend und Sport Baden Württemberg Ministeriums für: *Empfehlungen für einen zeitgemäßen Schulhausbau in Baden-Württemberg*, 2012/2013. http://www.schulentwicklung-net.de/images/stories/Anlagen/510%20schulhausbau_BW_2013.pdf.
- [18] Kybic, Jan: *Point Distribution Models*, 2007. <http://cmp.felk.cvut.cz/cmp/courses/33DZOzima2007/slidy/pointdistributionmodels.pdf>.
- [19] Morency, Louis Philippe, Jacob Whitehill und Javier Movellan: *Generalized Adaptive View-based Appearance Model: Integrated Framework for Monocular Head Pose Estimation*. In: *8th International Conference on Automatic Face and Gesture Recognition*, Amsterdam, The Netherlands, 2008. <http://ict.usc.edu/pubs/Generalized%20Adaptive%20View-based%20Appearance%20Model-%20Integrated%20Framework%20for%20Monocular%20Head%20Pose%20Estimation.pdf>.
- [20] Neubeck, Alexander und Luc Van Gool: *Efficient Non-Maximum Suppression*. In: *Proceedings of the 18th International Conference on Pattern Recognition - Volume 03*, ICPR '06, Seiten 850–855, Washington, DC, USA, 2006. IEEE Computer Society, ISBN 0-7695-2521-0. <http://dx.doi.org/10.1109/ICPR.2006.479>.
- [21] Peemen, Maurice. <http://parse.ele.tue.nl/mpeemen>.
- [22] Stepanov, Vitalij: *Analyse komplexer Szenen mit Hilfe von Convolutional Neural Networks*, 2012.
- [23] Świrski, Lech, Andreas Bulling und Neil A. Dodgson: *Robust real-time pupil tracking in highly off-axis images*. In: *Proceedings of ETRA*, März 2012. <http://www.cl.cam.ac.uk/research/rainbow/projects/pupiltracking/>.
- [24] Tadas Baltrušaitis, Peter Robinson, Louis Philippe Morency: *Constrained Local Neural Fields for robust facial landmark detection in the wild*, 2013.

- [25] Tadas Baltrušaitis, Peter Robinson, Louis Philippe Morency: *OpenFace: an open source facial behavior analysis toolkit*, 2016.
- [26] Wikibooks: *GNU R: boxplot* — Wikibooks, Die freie Bibliothek, 2012. https://de.wikibooks.org/w/index.php?title=GNU_R:_boxplot&oldid=641628, [Online; abgerufen am 10. Juli 2017].
- [27] Wikipedia: *Active Appearance Model* — Wikipedia, Die freie Enzyklopädie, 2014. https://de.wikipedia.org/w/index.php?title=Active_Appearance_Model&oldid=135641554, [Online; Stand 16. Juni 2017].
- [28] Wikipedia: *Bicubic interpolation* — Wikipedia, The Free Encyclopedia, 2016. https://en.wikipedia.org/w/index.php?title=Bicubic_interpolation&oldid=751879378, [Online; accessed 6-May-2017].
- [29] Wikipedia: *Canny-Algorithmus* — Wikipedia, Die freie Enzyklopädie, 2016. <https://de.wikipedia.org/w/index.php?title=Canny-Algorithmus&oldid=156854550>, [Online; Stand 28. Juni 2017].
- [30] Wikipedia: *Lanczos-Filter* — Wikipedia, Die freie Enzyklopädie, 2016. <https://de.wikipedia.org/w/index.php?title=Lanczos-Filter&oldid=150175121>, [Online; Stand 6. Mai 2017].
- [31] Wikipedia: *Augenbewegung* — Wikipedia, Die freie Enzyklopädie, 2017. <https://de.wikipedia.org/w/index.php?title=Augenbewegung&oldid=166073779>, [Online; Stand 13. Juni 2017].
- [32] Wikipedia: *Convolutional Neural Network* — Wikipedia, Die freie Enzyklopädie, 2017. https://de.wikipedia.org/w/index.php?title=Convolutional_Neural_Network&oldid=166523646, [Online; Stand 29. Juni 2017].
- [33] Wikipedia: *OpenCV* — Wikipedia, Die freie Enzyklopädie, 2017. <https://de.wikipedia.org/w/index.php?title=OpenCV&oldid=166087629>, [Online; Stand 16. Juni 2017].
- [34] Wikipedia: *Point distribution model* — Wikipedia, The Free Encyclopedia, 2017. https://en.wikipedia.org/w/index.php?title=Point_distribution_model&oldid=759054014, [Online; accessed 9-May-2017].
- [35] Wissensmedien (IWM), Leibniz Institut für: *Tübingen Digital Teaching Lab (TüDiLab)*. <https://www.tuedilab-tuebingen.de/>.
- [36] Wolfgang Fuhl, Thiago C. Santini, Thomas Kübler Enkelejda Kasneci: *ElSe: Ellipse Selection for Robust Pupil Detection in Real-World Environments*, 2016. <http://dx.doi.org/10.1145/2857491.2857505>.
- [37] Wood, Erroll, Tadas Baltrusaitis, Xucong Zhang, Yusuke Sugano, Peter Robinson und Andreas Bulling: *Rendering of Eyes for Eye-Shape Registration and Gaze Estimation*. In: *Proc. of the IEEE International Conference on Computer Vision (ICCV 2015)*, 2015.

Literaturverzeichnis

- [38] Xucong Zhang, Yusuke Sugano, Mario Fritz Andreas Bulling: *Appearance-Based Gaze Estimation in the Wild*, 2015.
- [39] Yusuke Sugano, Xucong Zhang, Andreas Bulling: *AggreGaze: Collective Estimation of Audience Attention on Public Displays*, 2016.

Erklärung

Hiermit erkläre ich, dass ich diese schriftliche Abschlussarbeit selbständig verfasst habe, keine anderen als die angegebenen Hilfsmittel und Quellen benutzt habe und alle wörtlich oder sinngemäß aus anderen Werken übernommenen Aussagen als solche gekennzeichnet habe.

Ort, Datum

Unterschrift