

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

# Masterarbeit Informatik

## **Untersuchung zur Machbarkeit von simultanem Eye-Tracking bei Menschengruppen mit Anwendungen im Klassenzimmer**

Falko Benezan

17. Juli 2017

### **Erstprüfer**

**Jun.-Prof. Enkelejda Kasneci**  
Wilhelm-Schickard-Institut für Informatik  
Universität Tübingen

### **Betreuer**

**Prof. Ulrich Trautwein**  
Hector-Institut für Empirische Bildungsforschung  
Universität Tübingen

**Benezan, Falko:**

*Untersuchung zur Machbarkeit von simultanem Eye-Tracking bei Menschengruppen mit Anwendungen im Klassenzimmer*

Masterarbeit Informatik

Eberhard Karls Universität Tübingen

Bearbeitungszeitraum: 18.01.2017 – 18.07.2017

Betreuer: Thomas Kübler

## Zusammenfassung

Aufmerksamkeit ist einer der Grundvoraussetzungen für erfolgreiches Lernen in der Schule. Eine objektive Quantifizierung der Aufmerksamkeit eines Schülers oder einer ganzen Klasse könnte dabei helfen Lern- und Lehrprozesse besser zu verstehen und zu verbessern.

Eine technische Messung der Aufmerksamkeit ist nur indirekt möglich, z.B. per Eye-Tracking Brille um das Blickverhaltens zu beobachten. Mit dem momentanen Stand der Technik muss für jeden Schüler ein einzelnes Gerät eingesetzt werden. Dieser Prozess ist extrem teuer, störend für die Probanden und in der Auswertung aufwendig.

Diese Arbeit untersucht die technische Machbarkeit eines effizienten Aufbaus zur Aufmerksamkeitsanalyse einer Menschengruppe. Dazu werden die Grenzen und möglichen Genauigkeiten einer Gesichtsanalyse basierend auf Bildmaterial einer einzelnen Kamera ausgelotet.

Durch diesen Aufbau ergibt sich die Problematik sehr unterschiedlicher Distanzen zwischen den zu messender Person zur Kamera und, daraus resultierend, unterschiedliche Abbildungseigenschaften, wie z.B. die Anzahl an Pixel die das Gesicht der Person im Bild darstellen.

Um alle Probanden in einem Kamerabild bewerten zu können, werden zuerst die einzelnen Gesichter im Bild detektiert, eindeutig einem Probanden zugeordnet und dann aufbereitet. Durch eine folgende Analyse des abgebildeten Gesichts lassen sich dessen Position und Orientierung im Raum bestimmen. Die Augenregion ist für die gerichtete Aufmerksamkeit besonders aussagekräftig und wird deshalb zusätzlich gesondert behandelt, um genauere Ergebnisse bei der Bestimmung der Blickrichtung zu erhalten.

Die Versuche haben ergeben, dass mit den heutigen hochauflösenden Kameras eine gleichzeitige Analyse mehrerer Probanden möglich ist, wenn sie sich auf der Fläche eines üblichen Klassenzimmers verteilen.

Für die Analyse kann meist nur auf der Kopforientierung gearbeitet werden, da für die Bestimmung der Blickrichtung zu wenige Informationen in den kleinen Bildern vorhanden sind. Abgeleitet aus den Ergebnissen der einzelnen Verfahren sollte eine Auswertung der Augen auf einer Distanz von 4m möglich sein, konnte im Test unter Realbedingungen allerdings bei weitem nicht erreicht werden. Die verwendeten Verfahren zur Gesichtsanalyse (Landmarkenbestimmung und Positionserrechnung) sind auf einen Winkel von 45° relativ zur Kamera beschränkt.

# Inhaltsverzeichnis

<b>1</b>	<b>Einführung</b>	<b>7</b>
<b>2</b>	<b>Stand der Forschung</b>	<b>9</b>
2.1	Erfassung von ON/OFF-Task . . . . .	9
2.2	Automatisierte Messverfahren von Aufmerksamkeitsparametern . . . . .	10
2.3	Computer Vision Methoden zur Gesichtsanalyse . . . . .	10
2.3.1	Künstliches neuronales Netz . . . . .	11
2.3.2	Convolutional Neural Network (CNN) . . . . .	11
2.3.3	Constrained Local Model (CLM) . . . . .	11
2.3.4	Constrained Local Neural Fields (CLNF) . . . . .	13
2.3.5	Active Appearance Model (AAM) . . . . .	13
2.3.6	Patch Experts . . . . .	13
2.3.7	Non-maximum suppression (NMS) . . . . .	13
2.3.8	Point Distribution Model (PDM) & Generalized Adaptive View-based Appearance Model (GAVAM) . . . . .	14
2.4	Gesichtserkennung . . . . .	14
2.4.1	Die 3 Stufen der Verarbeitung . . . . .	15
2.4.2	Zuverlässigkeit bei der Detektion . . . . .	16
2.5	Aufbereitung der Bilder . . . . .	17
2.5.1	Bicubic-Skalierung . . . . .	17
2.5.2	Lanczos-Skalierung . . . . .	17
2.5.3	Linear-Skalierung . . . . .	19
2.5.4	Nearest-Neighbor-Skalierung . . . . .	19
2.6	Gesichtsanalyse mit OpenFace . . . . .	19
2.6.1	Gesichts-Landmarks: Detektion und Verfolgung . . . . .	20
2.6.2	Veröffentlichte Genauigkeit der Kopforientierung . . . . .	20
2.7	Graukonvertierung: Farbbild nach Graubild . . . . .	21
2.7.1	Gleam-Verfahren . . . . .	22
2.7.2	Gleam-New-Verfahren . . . . .	22
2.7.3	Luminance-Verfahren . . . . .	22
2.7.4	Min-Max-Verfahren . . . . .	23
2.7.5	Quadrat-Verfahren . . . . .	23
2.7.6	Normalisierung der Graubilder . . . . .	23

## Inhaltsverzeichnis

2.8	Augenanalyse mit Ellipse Selection for Robust Pupil Detection (ElSe) . . . . .	24
2.8.1	Pupille bestimmen mit Kantendetektion . . . . .	26
2.8.2	Grobe Bestimmung der Pupille . . . . .	26
2.9	Bestimmung des Ziels der Aufmerksamkeit . . . . .	27
2.9.1	Bestimmung der Position & Orientierung des Gesichts . . . . .	27
2.9.2	Bestimmung eines Punktes, auf dem die Aufmerksamkeit liegt . . . . .	30
2.10	Verwendete Bibliothek . . . . .	30
<b>3</b>	<b>Herangehensweise</b>	<b>31</b>
3.1	Eye-Tracking in der Klassenzimmer-Umgebung . . . . .	33
3.2	Schulklassenvideo . . . . .	34
<b>4</b>	<b>Evaluation</b>	<b>37</b>
4.1	OpenFace im Test . . . . .	37
4.1.1	Auswirkung der Auflösung auf die Detektionsrate . . . . .	37
4.1.2	Auswirkung der verschiedenen Skalierungsverfahren auf die Detektion . . . . .	38
4.1.3	Auswirkung der verschiedenen Skalierungsverfahren auf den Arbeitsbereich bezüglich Rotation . . . . .	38
4.1.4	Auswirkung der Skalierungsverfahren auf die Positionsbestimmung . . . . .	40
4.1.5	Auswirkung von Pixelrauschen auf die Detektion . . . . .	42
4.1.6	Verschiedene Rechenverfahren zur Bestimmung der Position . . . . .	42
4.1.7	Ergebnis bezüglich Verwendbarkeit . . . . .	47
4.2	ElSe im Test . . . . .	48
4.2.1	Auswirkung des Filterradius . . . . .	49
4.2.2	Auswirkung der verschiedenen Graubild-Verfahren . . . . .	49
4.2.3	Vergleich zu OpenFace . . . . .	51
4.2.4	Ergebnis . . . . .	51
4.3	Versuch 1 - Arbeitsbereich der Verfahren . . . . .	53
4.3.1	Versuchsaufbau . . . . .	53
4.3.2	Detektion mit MTCNN . . . . .	54
4.3.3	Auswertung der Aufnahme . . . . .	55
4.3.4	Ergebnis . . . . .	55
4.4	Versuch 2 - Arbeitsbereich bezüglich der Neigung des Kopfes . . . . .	56
4.4.1	Versuchsaufbau . . . . .	56
4.4.2	Auswertung . . . . .	57
4.4.3	Ergebnisse . . . . .	57
4.5	Versuch 3 - Berechnung auf der Augenpartie . . . . .	58
4.5.1	Versuchsaufbau . . . . .	58
4.5.2	Auswertung . . . . .	58

## *Inhaltsverzeichnis*

4.5.3	Ergebnis . . . . .	58
4.6	Versuch 4 - Aufmerksamkeitsmessung . . . . .	60
4.6.1	Versuchsdurchführung . . . . .	60
4.6.2	Fehleranalyse im Versuch . . . . .	63
4.7	Zusammenfassung . . . . .	66
<b>5</b>	<b>Diskussion</b>	<b>69</b>
<b>6</b>	<b>Abbildungen</b>	<b>71</b>
6.0.1	Boxplot . . . . .	71
6.0.2	Anzahl der Messwerte . . . . .	71
6.1	Graphen . . . . .	73
	<b>Literaturverzeichnis</b>	<b>86</b>

# 1 Einführung

Die Grundlage für erfolgreiches Lernen ist die Aufmerksamkeit der Schüler. Sie ist ein wichtiger Indikator für die momentane Aufnahmefähigkeit des Schülers und die Qualität des Unterrichts. Das Verhalten eines Schülers kann stark vereinfacht eingeteilt werden in *ON-Task* (aufmerksam bei der Sache) und *OFF-Task* (unaufmerksam). Allerdings ist das Erfassen von *einer Aufgabe zugewandten Aufmerksamkeit* technisch schwierig, da es sich um einen kognitiven Prozess handelt der nur indirekt beobachtet werden kann. Entsprechend existieren verschiedene Erfassungsmethoden; Ein Vorschlag von Ehrhardt, Findeisen, Marinello und Reinhartz-Wenzel (1981) umfasst beispielsweise die Beurteilung von Blickrichtung, Körperhaltung und Tätigkeit. Zur Erfassung werden z.B. Fragebögen eingesetzt, die Schüler und/oder Lehrer selbst ausfüllen oder ein unabhängiger Beobachter bewertet die Aufmerksamkeit einzelner Schüler anhand festgelegter Kriterien.

Die Zuwendung von Aufmerksamkeit kann indirekt z.B. durch eine Blickzuwendung gemessen werden (auch wenn nicht mit jeder Blickzuwendung zwangsweise eine Aufmerksamkeitszuwendung einhergehen muss, ist dies eine oftmals hinreichende Annäherung). Während eine Blickrichtungsbestimmung erstrebenswert wäre, kann auch bereits die Bestimmung der Kopforientierung als Richtungsindikator verwendet werden.

Im Rahmen dieser Arbeit soll untersucht werden, wie weit es technisch möglich ist Filmmaterial einer Unterrichtsstunde im Bezug auf Blickrichtungen auszuwerten und mit welchen Einschränkungen und Genauigkeiten zu rechnen ist. Daraus lassen sich Anhaltspunkte sowohl über die Auswertbarkeit existierender Daten als auch über einen optimalen Versuchsaufbau ableiten.

Gängige Methoden zur Bestimmung der Blickrichtung, wie beispielsweise Eye-Tracking Brillen, sind für diesen Zweck nur eingeschränkt geeignet. Zum einen ist die Anschaffung einer großen Stückzahl dieser Geräte teuer und wurde bisher nur in wenigen speziell eingerichteten Laboratorien durchgeführt wie z.B. dem TüDiLab [36]. Zum anderen sind die Geräte entweder intrusiv und haben damit ein Ablenkungspotential (Brillen) oder schränken den Aktionsradius ein (Remote Tracker mit ihrer Head-Box von üblicherweise weniger als 30x30 cm).

In dieser Arbeit werden die Grenzen der momentan zur Verfügung stehenden Algorithmen bestimmt. Dies liefert Anhaltspunkte für ein optimales Setup für ein größeres Experiment, um die Aufmerksamkeit einer ganzen Klasse erfassen zu können.

## *1 Einführung*

Dies betrifft vor allem die Anzahl und Position der Kameras und deren Auflösung, damit der gesamte Bereich eines Klassenzimmers voll abgedeckt und trotzdem möglichst einfach ist.

Wäre man in der Lage, solch eine qualitativ hochwertige Auswertung mit nur wenigen Kameras durchführen zu können, so ist der Aufbau und die Aufnahmen der Daten auch für technische Laien durchführbar.

Müssen dagegen viele Kameras verwendet werden, so ergeben sich verschiedene Problematiken: Alle Kameras müssen synchronisiert werden im Bezug auf die Zeit und ihre Ausrichtung zueinander, um die Ergebnisse basierend auf den einzelnen Videos miteinander abgleichen zu können. Diese Synchronisation ist bei wenigen Kameras deutlich einfacher. Außerdem müssen alle Aufzeichnungen in Echtzeit stattfinden, womit die Limitierung der Bandbreite bei Vernetzungen ebenfalls berücksichtigt werden muss und somit die Anzahl begrenzen kann.

Die Interpretation der Ergebnisse dieser Arbeit orientiert sich an Originalaufnahmen eines Englischunterrichtes. Diese zeigen die gesamte Klasse aus Sichtrichtung der Tafel.

Da für diese Aufnahmen keine Ground-Truth Daten (exakte Position der Schüler/Kamera usw.) bekannt sind, wird eine Reihe von Versuchen durchgeführt, um die einzelne Aspekte und Problemstellungen der Datenanalyse dieser Videos genauer zu untersuchen.

In den ersten Versuchen wurden verschiedene Aufnahmen verwendet, um die Auswirkung von Position und Blickziel relativ zur Kamera zu testen. Für den anschließenden Versuch wurde ein bewegliches Blickziel erstellt, um eine kontinuierliche Messwerterfassung zu testen.

## 2 Stand der Forschung

Eine Übersicht von bereits existierenden Verfahren die ähnlich gelagert sind oder für die aktuelle Anwendung verwendet werden. In Abschnitt 2.1 ist ein Überblick über Auswertungsverfahren aus der Verhaltensforschung und in Abschnitt 2.2 technische Messverfahren mit ähnlicher Anwendung.

Die grundlegenden Verfahren der Bildverarbeitung die verwendet werden sind in Abschnitt 2.3 kurz erklärt und Abschnitt 2.4 bis Abschnitt 2.10 befassen sich mit den verwendeten Algorithmen und Anwendungen.

### 2.1 Erfassung von ON/OFF-Task

„Das Münchener Aufmerksamkeitsinventar (MAI)“[13] ist ein Verfahren zur Bestimmung der Aufgabenbezogenen Aufmerksamkeit. Es unterscheidet unter anderem zwischen „*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 Lernelegenheit nicht nur nicht wahr, sondern ist erkennbar anderweitig engagiert*“. Um das Verhalten eines Schülers zu bewerten wird dieser 5s lang beobachtet und einer dieser binären Kategorien 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*“, unterschieden. Ein Schüler wird in einem Ein-Minuten-Intervall beobachtet und bewertet. Sind drei oder mehr Kriterien erfüllt, gilt das Intervall als ON-Task.

Bei dieser Art der Auswertung gibt es allerdings Interpretationsfreiheiten, die von jedem Beobachter anders ausgelegt werden können. So kann z.B. das Spielen mit einem Stift als motorische Unruhe oder als Zeichen von Nervosität bewertet werden. Außerdem ist diese Art der Auswertung sehr zeitintensiv, alleine die Beurteilung eines einzelnen Schülers einer Klasse benötigt mindestens 30 Minuten. Somit kann eine Auswertung aller Schüler (etwa 30 Personen nach Vorgabe der Klassenbildung [4]) für eine einzelne Unterrichtsstunde schnell 15 und mehr Arbeitsstunden dauern. Um subjektive Bewertungen zu vermeiden sollte außerdem ein beträchtlicher Teil der Daten

## *2 Stand der Forschung*

von mindestens zwei Beobachtern parallel ausgewertet werden, um deren Übereinstimmung beurteilen zu können.

Basiert die Auswertung auf wenigen ausschnittsweisen Zeitintervalle um Arbeitszeit zu sparen, wird das gesamte Verhalten eines Schülers während des Unterrichts mit nur wenigen beobachteten Minuten beschrieben. Somit können sowohl quantitativ genaue, als auch temporal hochauflösende Daten nicht erstellt werden.

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

## **2.2 Automatisierte Messverfahren von Aufmerksamkeitsparametern**

Eine Möglichkeit zur automatischen Erfassung 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. Es wird anhand der Kopf und Augenposition bewertet, ob dieser aktiv auf den Verkehr achtet.

Für die aktuelle Problemstellung ist dieses Verfahren allerdings nicht geeignet, da es auf einen einzigen Probanden in nächster Nähe zur Kamera ausgelegt ist. Außerdem wird die Zuordnung anhand von festen Regeln bezüglich der Kopfposition durchgeführt, und ist daher nicht dynamisch genug für die Auswertung einer ganzen Schulkasse.

Ein weiteres ähnliches Verfahren wird in „AggreGaze“[40] präsentiert. Dabei wird eine einzige Kamera fest auf einem Bildschirm montiert, um die Blickrichtung der Passanten auf den Bildschirm zu bestimmen. Dieses Verfahren arbeitet 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. Dies entspricht zwar dem grundsätzlichen Aufbau, allerdings muss der Erfassungsbereich deutlich größer ausfallen.

## **2.3 Computer Vision Methoden zur Gesichtsanalyse**

Gesichtserkennung ist eine der am weitesten fortgeschrittenen Verfahren in der maschinellen Bildverarbeitung und wird ständig weiterentwickelt. Neben dem Erkennen eines Gesichts fällt darunter auch dessen Analyse wie beispielsweise die Orientierung, Übereinstimmung oder das Erkennen von Mimik.

Eine Standardmethode bei der Gesichtserkennung ist die Verwendung eines neuronalen Netzes.

## 2.3 Computer Vision Methoden zur Gesichtsanalyse

### 2.3.1 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 trainiert, 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, dass der Fehler des Netzes auf dem Trainingsdatensatz minimal wird. [17]

### 2.3.2 Convolutional Neural Network (CNN)

CNNs definieren in vielen Anwendungsbereichen den Stand der Technik. Sie sind eine Weiterentwicklung der neuronalen Netze und werden unter anderem bei der Bild- und Spracherkennung eingesetzt. Als Erweiterung wird eine gewichtete Faltungen der Eingabe als Eingabeparameter für das neuronale Netz verwendet.

Ein CNN kann in zwei Bereiche aufgeteilt werden, Feature Extraktion und Klassifizierung.

Bei der Feature Extraktion werden verschiedene Faltungskern und Komprimierungen auf den Eingabeinformationen angewendet um die Informationen für den zweiten Teil der Klassifizierung aufzubereiten. Durch die Faltung werden die Information aus den umliegenden Punkten eines Bereiches zusammengefasst und komprimiert an die nächste Schicht weitergegeben. Der Faltungskern kann je nach Anwendung beliebig gestaltet sein, so ist z.B. eine Glättung durch einen Gauß-Kernel oder Kantendetektion durch einen Kirsch-Operator möglich.

Im zweiten Teil, der Klassifizierung, werden nun alle vorhanden und zusammenführten Informationen ausgewertet, um das Ergebnis für die Eingabe zu erhalten.

Gelernt werden kann jeder einzelne Kernel für sich und die jeweiligen Bewertungen der Kernel und Neuronen. [23][33]

### 2.3.3 Constrained Local Model (CLM)

In Constrained Local Modellen wird die Erkennung eines Objektes in die Erkennung einzelner charakteristischer Teilpunkte, der sogenannten Landmarks, aufgespalten. Die-

## 2 Stand der Forschung

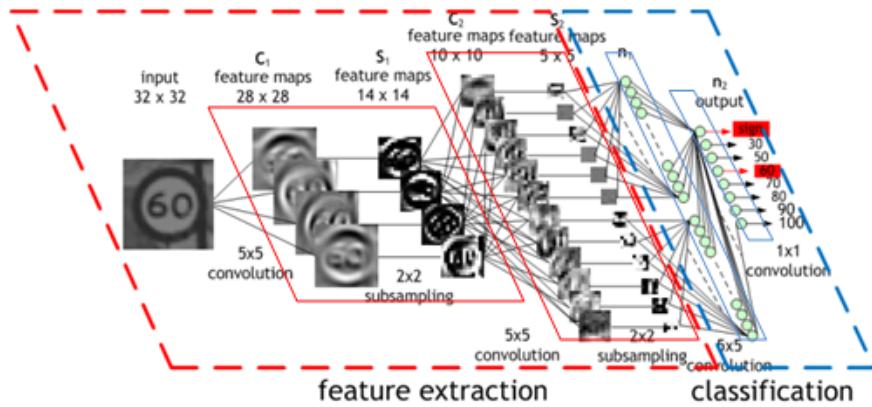


Abbildung 2.1: Beispiel für den Aufbau eines CNN zur Klassifizierung.

Zu sehen ist das Erkennen einer Zahl auf einem Straßenschild. Bild von [22]

ses Verfahren eignet sich deshalb besonders dazu deformierbare Objekte zu erkennen. Um mehrere Punkte eines Objektes zu lokalisieren wird eine Wahrscheinlichkeitskarte der einzelnen Teilpunkte relativ zueinander gelernt. Auf dem Eingabebild wird dann die Ähnlichkeit der Bildregionen mit den gesuchten Punkten quantifiziert, die die Ähnlichkeit der Darstellung angibt. Anschließend wird die optimale Kombination aus Bildähnlichkeit und der Lage aller Punkte zueinander bestimmt.  
Diese Art der Bestimmung von positionsabhängigen Punkten ist ziemlich zuverlässig und dennoch dynamisch genug um auch mit kleinen Veränderungen zurecht zu kommen.

Dies ist wichtig bei der Detektion von leicht verformbaren Objekten wie beispielsweise Gesichtern und ist zuverlässiger als das Active Appearance Model (AAM). [7]

### 2.3.4 Constrained Local Neural Fields (CLNF)

Bei CLNF 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 werden die Ergebnisse der Detektoren mit einer Karte der Position aller Landmarks, entsprechend dem Vorgehen eines CLM, verglichen. [25]

### 2.3.5 Active Appearance Model (AAM)

Dies ist ein Verfahren der Bildverarbeitung um Übereinstimmungen zu einem Modell zu finden. Dazu wird aus dem Trainingsdatensatz eine typische „durchschnittliche“ Form eines Objektes, sowie die Faktoren der wichtigsten möglichen Veränderungen der Form ermittelt. So können beispielsweise alle Merkmale, die ein Gesicht als männlich oder weiblich charakterisieren in einem Abweichungsvektor zusammengefasst und durch einen einzigen Gewichtungsparameter beschrieben werden.

Soll nun zu einem Eingabebild die Übereinstimmung ermittelt werden, so müssen nur die wichtigsten Veränderungsfaktoren angepasst werden. Dies ist ein bedeutend kleinerer Parameterraum als alle Landmarks einzeln anzupassen. Sind dennoch Unterschiede zur Eingabe vorhanden, liegen diese an der Erscheinung des Objektes. [28]

### 2.3.6 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 diese Position herum ausgewertet, um Ergebnisse auf ein Teilpixels genau zu bestimmen. [25]

### 2.3.7 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 zur exakten Bestimmung einer Kante, 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 dieser Wert nicht maximal sein auf Null

## 2 Stand der Forschung

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 beschreiben wird. [21][30]

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

Mit einem Point Distribution Model (PDM) können verformbare Objekte 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 Deformationen darzustellen.

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

Somit können 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 die Lösung nicht eindeutig ist. Dies kann daran liegen, dass unter Umständen durch verschiedene Gewichtungskombinationen die selbe Darstellung erzielt wird oder dass eine nicht erfasste Verformung des Objektes stattgefunden hat, wodurch immer eine Abweichung zu allen Kombinationen entsteht.

Dieses Problem der Verformung tritt bei Berechnungen von Gesichtern auf, da immer eine Veränderung z.B. der Mundwinkel oder Augenlider vorhanden ist. [19][20][35]

## 2.4 Gesichtserkennung

Multi-task Cascaded Convolutional Networks (MTCNN) ist ein Algorithmus zur Detektion von Gesichtern und Bestimmung von 5 Gesichts-Landmarks in Farbbildern. Es machte im Vorabtests auf Probebildern einen sehr guten Eindruck und konnte die meisten Gesichter mit verschiedenen Größen und Blickrichtungen finden.

Für die Detektion werden drei CNN auf einer Bildpyramide angewendet um zuverlässig Gesichter verschiedenster Größe zu erkennen. Außerdem wird für die Detektion der

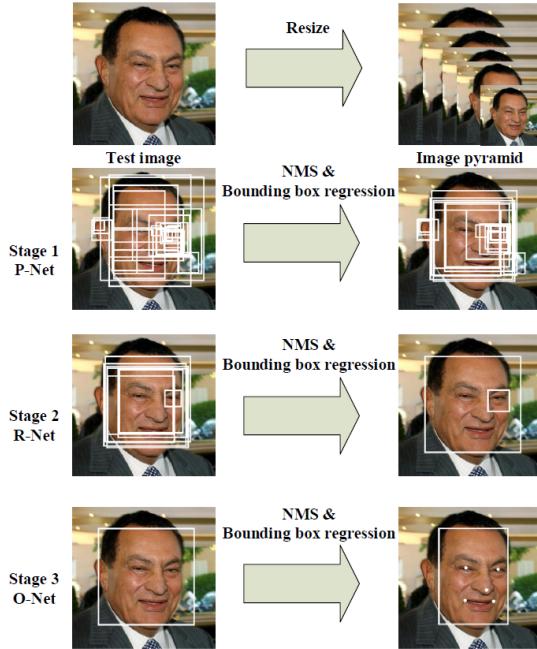


Abbildung 2.2: Darstellung des Funktionsablaufes von MTCNN, Abbildung aus [16]

Gesichter auch deren Ausrichtung berücksichtigt, um bessere Ergebnisse zu erzielen. Laut Beschreibung des Verfahrens sollen somit sogar recht kleine Gesichter mit  $20 \times 20$  Pixeln erfassbar sein.

Sein Einsatzgebiet in der aktuellen Anwendung 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 hochauflösenden Bild gearbeitet mit verhältnismäßig kleinen, verschieden großen und weit verteilten Gesichtern.

### 2.4.1 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 weiter durch die Skalierung erschwert, wodurch das CNN auf seine jeweilige Aufgabe besser optimiert werden kann.

## *2 Stand der Forschung*

### **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 weiteren 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.

### **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.

#### **2.4.2 Zuverlässigkeit bei der Detektion**

MTCNN Face Detection ist bei der Zuverlässigkeit im Vergleich zu anderen bekannten Verfahren laut dem dazugehörigen Paper [16] überlegen. Zudem auf  $640 \times 480$  großen Bildern echtzeitfähig und kann vor allem auch sehr kleine Gesichter erfolgreich erkennen.

Somit sind alle Anforderungen erfüllt um mit diesem Verfahren den vorhanden Frame für die nachfolgenden Berechnungen vorzubereiten. Ein Test bestätigt diese Annahme, siehe Abbildung 4.14. Die 5 bestimmten Landmarks von MTCNN-Face sind allerdings gerade bei sehr kleinen Bildbereichen zu ungenau und deshalb im Folgenden nicht weiter verwendbar.

## 2.5 Aufbereitung der Bilder

Für die Gesichtsanalyse wird OpenFace verwendet, dieses Verfahren arbeitet laut Angabe im Paper [26] am besten auf Gesichtern mit einer Mindestgröße von 100 Pixel, daher werden die Bildbereiche auf diese Größe gebracht. Dies ist notwendig, da die Berechnung meist auf recht kleinen Bildausschnitten ausgeführt werden muss.

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.

Für die Abbildung 2.3 bis 2.6 wurde die Abbildung von Lena mit einer Kantenlänge von 100 Pixel und ein Schachbrettmuster mit 48 Pixel verwendet und auf eine Größe mit den angegebenen Verfahren auf 512 Pixel gebracht. Um den Unterschied zwischen der vergrößerten Lena-Darstellung und der Originalen zu zeigen, wurde das Differenzbild ebenfalls dargestellt.

Wird ein Bild vergrößert, müssen weitere Pixel dem Bild hinzugefügt werden. Zur Bestimmung des Farbwertes des Pixels gibt es verschiedene Verfahren, ein Teil wurde ausgewählt und ihre Auswirkung im folgenden getestet.

### 2.5.1 Bicubic-Skalierung

Der Farbwerte eines Pixel wird ermittelt, indem die umliegenden  $4 \times 4$  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 2.3 zu sehen. [29]

### 2.5.2 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 entfernte Pixel schwächer bewertet werden als näher liegende, siehe Abbildung 2.4.

Außerdem wird durch den Kurvenverlauf der Bewertungsfunktion eine gewisse Bildschärfe erreicht. Die Funktion kann und wird für die Anwendung auf einen  $8 \times 8$  Pixel

## 2 Stand der Forschung



Abbildung 2.3: Vergrößerung der Bilder mittels bikubischem Verfahren.  
Links: Lena, Mitte; Differenz zum Original, Recht: Schachbrett



Abbildung 2.4: Vergrößerung der Bilder mittels Lanczus-Verfahren.  
Links: Lena, Mitte; Differenz zum Original, Recht: Schachbrett



Abbildung 2.5: Vergrößerung der Bilder mittels linearer Interpolation.  
Links: Lena, Mitte; Differenz zum Original, Recht: Schachbrett

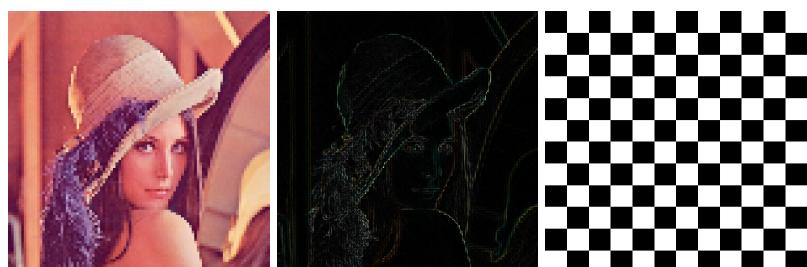


Abbildung 2.6: Vergrößerung der Bilder mittels Nearest-Neighbor.  
Links: Lena, Mitte; Differenz zum Original, Recht: Schachbrett

großen Bereich begrenzt. [31]

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

### 2.5.3 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 Nearest-Neighbor, und dennoch ein recht einfaches Verfahren. Die Kanten wirken allerdings unscharf, siehe Abbildung 2.5.

### 2.5.4 Nearest-Neighbor-Skalierung

Dieses Verfahren verwendet als neuen 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 2.6. Bei der Vergrößerung des Schachbretts sind kein Farbfehler aufgetreten, da nur zwei Farben vorhanden und diese positionsabhängig sind.

## 2.6 Gesichtsanalyse mit OpenFace

OpenFace ist ein Open-Source Echtzeitverfahren auf Basis von CLNF für die Bestimmung und Analyse von Gesichtsmerkmalen in Graubildern und Videos. Dieses Verfahren wurde gewählt, da es zum aktuellen Stand der Technik gehört und viele Probleme der Gesichtsanalyse löst. Dazu gehört neben der zuverlässigen Bestimmung der Landmarks, auch die Fähigkeit zur Bestimmung der 3D-Position, Ermittlung der Blickrichtung und die Fähigkeit zu lernen. Dabei stehen in der aktuellen Anwendung neben dem Farbbild 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 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

## 2 Stand der Forschung

von Interesse, da mit ihnen zurückrechnet werden kann wohin die Person schaut. Der Rechenaufwand zur Verarbeitung des Eingabebildes ist so ausgelegt, das 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.

### 2.6.1 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). Es wird das Point Distribution Model (PDM) zum Erfassen der Anordnung der Landmarks sowie ein verbessertes Patch Experts zum Erfassen der Variante der einzelnen Landmarks verwendet.

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. Diese Herangehensweise ist langsam, aber auch exakter als eine einfache Hypothese. Wird ein Tracking, das Verfolgen der Landmarks über mehrere Frames, durchgeführt, wird als initiale Hypothese das Ergebnis aus der letzten Eingabe verwendet. Sollte das Tracking 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 bestimmt. Für eine zuverlässige Detektion der Gesichter, sollten diese laut Paper [26] eine Größe von 100 Pixeln aufweisen.

### 2.6.2 Veröffentlichte Genauigkeit der Kopforientierung

Um die Qualität der Berechnung auf dem Kopf zu bewerten wurde im Paper [26] 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 bestimmten Kopfposition haben sich folgende Werte in Abbildung 2.7 ergeben, Angraben in Grad.

Für die Qualität zur Bestimmung der Blickrichtung wurde der Augendatensatz „Appearancebased gaze estimation in the wild“[39] zur Bestimmung der Blickrichtung verwendet und es ergab sich ein durchschnittlichen Fehler von 9,96 Grad.

## 2.7 Graukonvertierung: Farbbild nach Graubild

	Yaw	Pitch	Roll	Mean	Median
BIWI	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 2.7: Veröffentlichte Abweichung von OpenFace auf verschiedenen Datensätze, bestimmt von [26]



Abbildung 2.8: Dies sind die Eingabebilder der verschiedenen Konverter von Farbe nach Grau. Links eine Farbpalette, Mitte Lena und Rechts ein Augenausschnitt aus dem Augendatensatz [38]

## 2.7 Graukonvertierung: 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 von ElSe, 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.

Die Wahl von Gleam basiert auf den Ergebnissen von „Color-to-Grayscale: Does the Method Matter in Image Recognition?“[6] und New-Gleam als eine Umsetzung des dort veröffentlichtem Ausblicks. Luminance als Standart, Quadrat als gegenüber zu Gleam und Min/Max beruht auf der Idee die farbige Iris zu nutzen.

Das Eingabebild der Beispiele der einzelnen Graukonvertierungen ist in Abbildung 2.8 dargestellt. Eine Farbpalette, das Bildverarbeitungsbeispiel Lena sowie ein Augenbereich aus dem Augendatensatz [38].

## 2 Stand der Forschung

### 2.7.1 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, siehe Abbildung 2.9.

Allerdings wird auch dieser Farbwert erhöht und sollte die Pupille nicht schwarz sein, wird sie eher ins Graue überführt.[6]

$$G_{Gleam} = \frac{R^{\frac{1}{2,2}} + G^{\frac{1}{2,2}} + B^{\frac{1}{2,2}}}{3}$$

### 2.7.2 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 2.10. Da allerdings alle Farben dunkel werden, entstehen weitere dunkle Bereiche die die Detektion der Pupille beeinträchtigen können.

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

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.

### 2.7.3 Luminance-Verfahren

Dies ist ein lineares Verfahren, das der menschlichen Farbwahrnehmung entspricht und oft den Standard bei der Umwandlung von Farbbild nach Graubild darstellt. Somit entsteht ein natürlicher Farbverlauf, bei dem der Farbunterschied zwischen Pupille, Iris und Auge auf einem mittleren Niveau bleibt, siehe Abbildung 2.11.

Eine Gamma-Korrektur wird bei der Umwandlung nicht verwendet.[6]

$$G_{Luminance} = 0,299R + 0,587G + 0,114B$$

### 2.7.4 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 wird.

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

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

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

### 2.7.5 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 2.13.

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

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

### 2.7.6 Normalisierung der Graubilder

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.

## 2 Stand der Forschung



Abbildung 2.9: Ergebnis der Umwandlung von Farb- nach Grauwert mittels Gleam-Verfahren

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

Der Gauß-Filters 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 wirkt.

Die einzelnen Werte werden ihrer Umgebung angepasst, wodurch lokal auftretende Extremar verschwinden bzw. abgeschwächt werden und ähnliche Farbwerte zu ihrer Umgebung erhalten bleiben.

## 2.8 Augenanalyse mit 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 [37], Goutam [12], Starburst [8] oder Swirski [24]. Auch OpenFace verwendet ein zusätzlich dafür trainiertes CLNF, um weiter 28 LAndmarks pro Auge zu bestimmen, zusätzlich zu den 64 Landmarks das Gesicht. Aus diesen Augen-Landmarks, die Lider, Iris und Pupille darstellen kann die Blickrichtung ermittelt werden.

Um die Position der Landmarks zu verbessern, kann auf den Augenbereichen der ElSe-Algorithmus eingesetzt werden. Dieser Algorithmus basiert auf einem rechnerischen Ansatz und nicht auf Neuronen um die Umrisse der Pupille zu berechnen. Für die Verbesserung wurde das ElSe-Verfahren gewählt, da es im Test von [37] am besten abgeschnitten hat und direkt das Zentrum der Pupille liefert.

Der ursprüngliche ElSe-Algorithmus ist für Graubilder einer Eye-Tracking-Brille ausgelegt und optimiert, zudem ist es auf diesen Bildern zu einer Echtzeitauswertung

## 2.8 Augenanalyse mit Ellipse Selection for Robust Pupil Detection (ElSe)



Abbildung 2.10: Ergebnis der Umwandlung von Farb- nach Grauwert mittels Gleam-New-Verfahren



Abbildung 2.11: Ergebnis der Umwandlung von Farb- nach Grauwert mittels Luminance-Verfahren



Abbildung 2.12: Ergebnis der Umwandlung von Farb- nach Grauwert mittels Extremwert-Verfahren. Oben: Max-Verfahren, Unten: Min-Verfahren

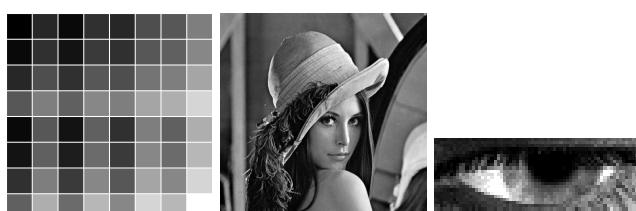


Abbildung 2.13: Ergebnis der Umwandlung von Farb- nach Grauwert mittels Quadrat-Verfahren

## 2 Stand der Forschung

in der Lage. Dieser Anwendungsbereich betrifft vor allem die Anforderung an eine 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 Voraussetzungen führen dazu, dass die Detektionsleistung bei niedriger auflösenden Bildern rasch abnimmt.

Als Ergebnis liefert ElSe eine Ellipse, die den Umriss der Pupille im Bild beschreibt. Aus dieser Ellipse können die Landmarks der Pupille abgeleitet werden.

Für den Test im Paper wurden Bilder von  $384 \times 288$  Pixel Größe verwendet und 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.[37]

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

### 2.8.1 Pupille bestimmen mit Kantendetektion

Da die Pupille als schwarzer Fleck im Bild dargestellt ist und die Iris einen helleren Farnton aufweist, wird ein Kantendetektor verwendet, der alle Pixel markiert, bei denen eine starke Farbänderung auftritt. Bei ElSe wird ein morphologischer 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 Anforderungen 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.

### 2.8.2 Grobe Bestimmung der Pupille

Sollte die Bestimmung der Ellipse, wie im vorigen 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 2.14 (1), wird ein kreisförmiger Mean-Filter eingesetzt mit Ergebnis in Abbildung 2.14 (3). Zur zweiten Faltung wird der Durchschnitt über ein Quadrat ohne inneren Kreis eingesetzt mit Ergebnis in Abbildung 2.14 (2), wobei für beide Kreise der selbe Radius verwendet wird.

Nun wird das Ergebnis des Quadratischen Mean-Filters invertiert, Abbildung 2.14 (4), und mittels Punkt-Multiplikation mit dem anderen Meanfilter zusammengebracht,

## 2.9 Bestimmung des Ziels der Aufmerksamkeit

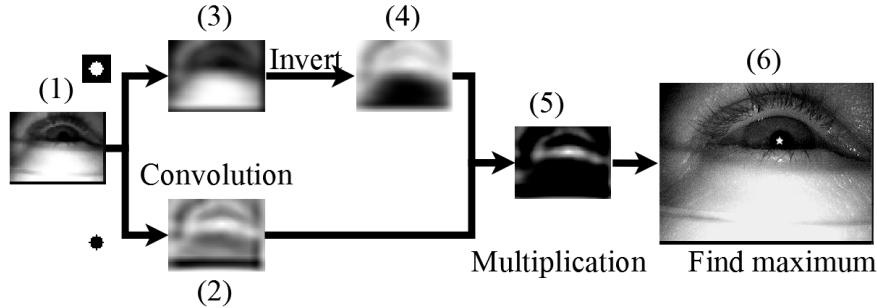


Abbildung 2.14: Ablauf der alternativen Berechnung zur Pupillen-Detektion von [37]

Abbildung 2.14 (5). Im resultierenden 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 2.14 (6) markiert.

## 2.9 Bestimmung des Ziels der Aufmerksamkeit

Um das Ziel der Aufmerksamkeit einer Person zu bestimmen, muss die 3D-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 die Gerade durch bestimmte Punkte im Raum verläuft. Ist dies der Fall, so wird dieser Punkt wahrscheinlich betrachtet und ist Ziel der Aufmerksamkeit der Person.

### 2.9.1 Bestimmung der Position & Orientierung des Gesichts

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 Kameraparametern der Aufnahme angepasst werden, um die reale Position und Orientierung zu bestimmen.

#### 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

## 2 Stand der Forschung

Dimension  $I_x \times I_y$  wird das Standardobjektiv mit einer Auflösung von  $640 \times 480$  Pixel angenommen, somit ergeben sich die Brennweiten  $f_x$  und  $f_y$  wie folgt [26]:

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

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

### Position & Orientierung

Zur Bestimmung der Kopfposition  $P = (X_{avg} \ Y_{avg} \ Z_{avg})^t$  im Kamerakoordinaten-system 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 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, somit ergibt sich die Position wie folgt:

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

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

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

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

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

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

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

Die Orientierung kann anhand der Position der Landmarks mit einem PDM bestimmt werden. [26]

### Bestimmung der Blickrichtung

Zur Bestimmung der Blickrichtung wird wie folgt vorgegangen: Zuerst wird der Strahl bestimmt der, ausgehend vom Zentrum der Kamera, durch das Zentrum der Pupille

## 2.9 Bestimmung des Ziels der Aufmerksamkeit

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.[26]

Da die Berechnung für jedes Auge unabhängig vom anderen ausgeführt wird, können Messungenauigkeiten dazu führen, dass die berechnete Blickrichtung der beiden Augen in verschiedene Richtung verlaufen.

Diesem kann entgegengewirkt werden, indem zwischen beiden Berechnungen (rechts und linkes Auge) eine Abhängigkeit formuliert wird, z.B. Durchschnitt.

Dieser Trick kann verwendet werden, da der Unterschied zwischen beiden Blickrichtungen minimal ausfällt (nahezu parallel), sollte ein weiter entfernter Punkt fokussiert werden.

### Auswirkung der Bildkoordinaten auf die Berechnung

Befinden sich die Bildpunkte nicht im Zentrum, so muss die Ausrichtung der Pixel beachtet werden, um diese mit in die Berechnung einfließen zu lassen. Dieser zusätzliche Winkel muss beachtet werden, da die Abweichung immer stärker wird, je weiter der Pixel vom Zentrum entfernt ist.

Als Ausgangspunkt werden die Ergebnisse des CNN verwendet um die Position zu erhalten. Zur Bestimmung der Orientierung  $R$  liefert auch das CNN ein Ergebnis  $R_{CNN}$ . Allerdings stimmt es nur im Zentrum des Bildes, da für die Berechnung die Anordnung der Landmarks relevant ist und die Position im Bild noch nicht beachtet wurde.

$$euler_x = \tan^{-1}\left(\frac{\sqrt{X^2 + Z^2}}{Z^2}\right)$$

$$euler_y = \tan^{-1}\left(\frac{\sqrt{Y^2 + Z^2}}{Z^2}\right)$$

$R_{pos} = R(euler_x, euler_y, 0)$  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 überführt werden. Anschließend werden die 3D nach 2D-Koordinaten wieder überführt um die Orientierung und Position zu ermitteln. Auch bei diesem Verfahren muss die Pixelorientierung beachtet werden. Allerdings ist die Verwendung eines Tiefenbildes von Vorteil, da ansonsten die Fehler weiter verstärkt werden. Daher ist es in der aktuellen Anwendung nicht sinnvoll einzusetzen.[26]

## 2 Stand der Forschung

### 2.9.2 Bestimmung eines Punktes, auf dem die Aufmerksamkeit liegt

Von Interesse ist vor allem der Punkt auf dem der Blick ruht bzw. auf den 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.

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

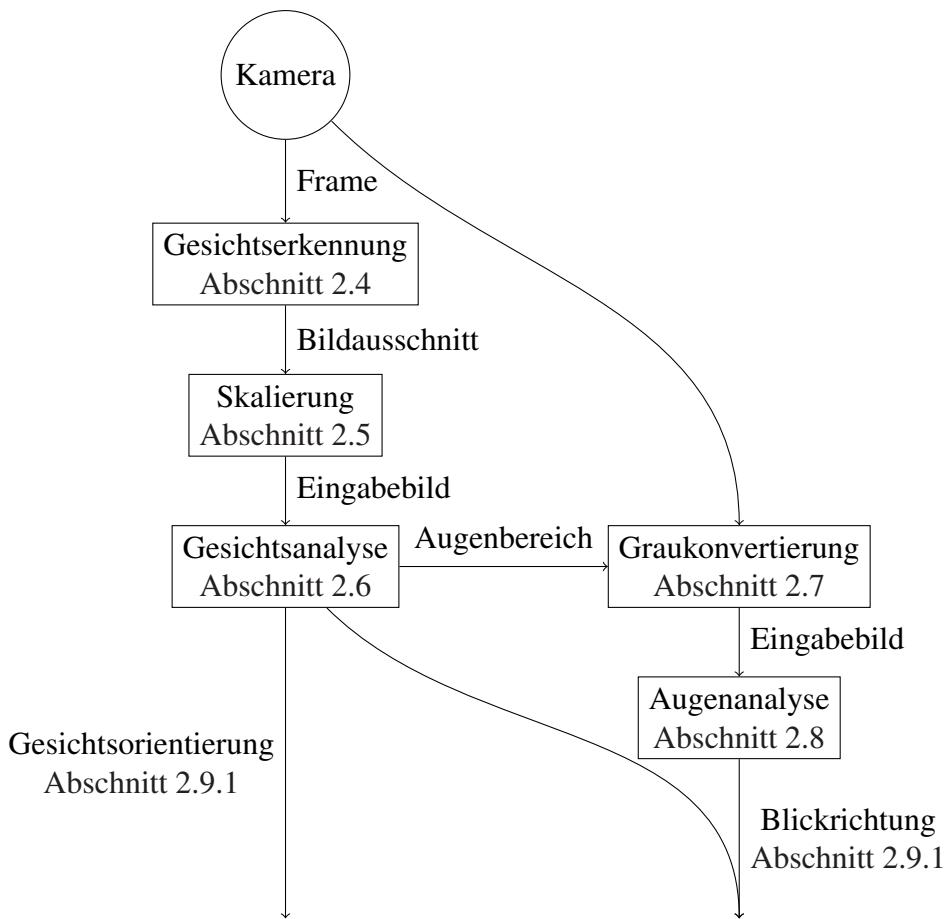
Bei Verwendung der Gesichtsorientierung ergibt sich das Problem den konkreten Blickpunkt zu ermitteln, da die Augenbewegung nicht erfasst werden kann. So muss ein Kegel, der den üblichen Bereich der Augenbewegung umfasst, um die Orientierung als Fehlertoleranz berücksichtigt werden und der gesamte Bereich 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 nur schwer 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.

## 2.10 Verwendete Bibliothek

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][34]

### 3 Herangehensweise



Zur Bestimmung der Blickrichtung sowie Kopfposition und Orientierung wird ein mehrstufiges Verfahren eingesetzt:

Zuerst müssen alle Gesichter, die im aktuellen Videobild (Frame) vorhanden sind, detektiert werden mit MTCNN-Face, siehe Abschnitt 2.4. Dabei machen die relevanten Bereiche nur einen sehr geringen Anteil des gesamten Bildes aus.

Ist ein Gesicht in mehreren Einzelbildern des Videos abgebildet, so muss auch eine Identitätszuordnung vorgenommen werden, damit dem Computerprogramm bekannt ist, welches Gesicht in Bild 1 welchem in Bild 2 entspricht. Für die Zuordnung reicht es

### 3 Herangehensweise

meist aus, jene Box zu wählen, die am ehesten den selben Bildausschnitt repräsentiert. Entweder über ähnliche Positionierung im Videobild oder über die Bildähnlichkeit, da ein Kopf sich zwischen zwei schnell aufeinanderfolgenden Einzelbildern nur limitiert bewegen kann.

Damit sicher auf allen Gesichtern gerechnet werden kann, ist eine semiautomatische Korrektur erforderlich um Falsch-Detektionen zu entfernen und fehlende Boxen der Gesichter ergänzen zu können. Daher können alle bisher unternommenen Schritte auch von anderen Verfahren übernommen werden, da es sich hierbei nur um einen Vorverarbeitungsschritt handelt und zur Beschleunigung sowie Stabilität der späterer Berechnung beitragen soll.

Die Bereiche müssen nicht exakt sein, 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 des Gesichtes betrachtet wodurch Gesichtserkennung und Landmark Erkennung auf unterschiedlichen großen Gesichtsbereichen arbeitet. 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, ist ein Zwischen-schritt zur Vergrößerung der erkannten Bereiche notwendig, damit das gesamte Gesicht abgebildet ist. Für die MTCNN-Face Box hat sich eine Vergrößerung um 30% als sinnvoll erwiesen, um Ungenauigkeiten bezüglich der Position und Dimension des Kopfes im Bild entgegen zu wirken.

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 2.5

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 gewünschten Größe dargestellt 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. 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.

Auf diese zuvor bestimmten und vergrößerten Bildausschnitten werden nun von OpenFace weiterverarbeitet um die Landmarks, die signifikanten Punkte eines Gesichtes zu bestimmen, siehe Abschnitt 2.6.

Durch die vorherige Identitätszuordnung der Gesichter kann das Verfahren gezielt auf einzelnen Personen arbeiten und ein entsprechend auf die Person eingestelltes CLNF verwenden, um bessere Ergebnisse zu erzielen, auch für jene Personen die

### *3.1 Eye-Tracking in der Klassenzimmer-Umgebung*

nur selten dargestellt sind. Außerdem können alle gefundenen Personen gleichzeitig (parallel) ausgewertet werden und durch die zuvor bestimmten Bildbereichen in denen ein Gesicht zu sehen ist, kann unnötige Suche vermieden werden.

Für die eigentliche Bestimmung der Landmarks bietet OpenFace zwei verschiedene Methoden, die Berechnung auf Bildern und Videos. Dies ist interessant für die spätere Anwendung, da somit auch Einzelbilder verwendet werden können, die eine deutlich höhere Auflösung besitzen als ein Video.

Der Hauptunterschied zwischen den beiden Verfahren ist das Lernen, dass bei der Videoauswertung verwendet wird, wodurch sich der Toleranzbereich deutlich erhöht und bessere Ergebnisse geliefert werden und, sollte ein Gesicht im aktuellen Frame erfolgreichen detektiert werden, können auch die nachfolgenden Frames durch das Lernen ausgewertet werden. Dies liegt an der Anpassung des Modells und dem möglichen Tracking der Landmarks.

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.

Für den im nächsten Schritt verwendeten ElSe Algorithmus muss der Bildausschnitt des Auges in ein Graubild umgewandelt werden, siehe Abschnitt 2.7.

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, siehe Abschnitt 2.8.

Da die Berechnung unabhängig der Landmarks ausgeführt wird, empfiehlt sich das Ergebnis zu überprüfen, damit die bestimmten Landmarks auch innerhalb der Augenhöhle liegen und grobe Fehler vermieden werden.

Nun wird auf Basis der Landmarks und Kameraparameter die Position und Orientierung der Gesichter sowie die Blickrichtung bestimmt, siehe Abschnitt 2.9.

## **3.1 Eye-Tracking in der Klassenzimmer-Umgebung**

Die Anwendung ist für den Unterricht ausgelegt, wie in Kapitel 1 beschrieben. Ein deutsches Klassenzimmer soll laut Baden-Württembergischen Schulbauempfehlungen eine Grundfläche von  $54 - 66m^2$  aufweisen und ist damit für maximal 28-32 Schüler geeignet [18].

Sollen mit einer einzigen Kamera alle Schüler auf einmal beobachtet werden, so lassen sich bereits hieraus Implikationen für die Kamera ableiten, da diese den kompletten Bereich erfassen muss, indem sich Schüler aufhalten können.

Abgeleitet aus der Grundfläche und abzüglich der Bereiche für Tafel, Schränke und weitere Einrichtung beginnt dieser etwa  $2,5m$  vor der Kamera und geht bis zu  $8m$  in

### 3 Herangehensweise

die Tiefe, bei einer Breite von  $6m$ , wenn sich die Kamera zentral an der Wand der Tafel befindet.

Somit muss der Linsenwinkel mindestens  $100^\circ$  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, womit sich folgende Randbedingungen ergeben:

- Brillen, Kontaktlinsen und Schmuck müssen nicht abgenommen werden, ebenso sind beliebige Frisuren, Make-up usw. möglich, solange sie das Gesicht nicht zu sehr verdecken.
- Die üblichen Bewegungen im Unterricht wie Sprechen, Kopfdrehungen usw. der Schüler sind möglich. Idealerweise ist eine freie Bewegung der Schüler im gesamten Klassenzimmer möglich.
- 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 einer aufwändigen Kalibrierung oder Vermessung unterzogen.

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

- Die Aufnahme erfolgt innerhalb eines Gebäudes, sodass einigermaßen kontrollierte Beleuchtungsbedingungen gewährleistet werden können.
- Die Gesichter der Schüler sind die meiste Zeit über komplett sichtbar und nicht verdeckt durch andere Schüler oder von der Kamera abgewandt.
- Blickrichtung und Gesichtsorientierung der Schüler sollen so exakt wie möglich erfasst werden.
- Die Überführung zwischen Welt- und Kamerakoordinatensystem ist bekannt. (Beispielsweise die Position der Kamera im Klassenzimmer und relativ zur Tafel)

## 3.2 Schulklassenvideo

Zur Visualisierung der Problematik der Daten, wurde ein anderes Bild verwendet, da aus Datenschutzgründen kann kein originales Bild des Unterrichtes veröffentlicht werden darf. Die Bildaufteilung, Kameraausrichtung und Auflösung des Datensatzes ist

### 3.2 Schulklassenvideo



Abbildung 3.1: Eine Screenshot des YouTube-Videos „Maxi Beister als Herr Müller überrascht eine Schulkasse“[14]

ähnlich zur Abbildung 3.1. Zur Verfügung steht nur ein einfaches Video der Schulkasse, ohne Ground-Truth Daten. Außerdem wurde es mit einer unbekannten Videokamera aufgezeichnet, daher sind nur die Parameter des Filmes ( $640 \times 480$  Pixel mit  $25Fps$ ) bekannt.

Die Hauptproblematik ist die Bildauflösung, sie ist sehr gering und die Gesichter sind nur durch entsprechend wenige Pixel dargestellt. Außerdem ist die Distanz zwischen den Schülern und Kamera sehr unterschiedlich wodurch verscheiden Größen entstehen.



# 4 Evaluation

Zuerst werden die einzelnen verwendeten Verfahren getestet bezüglich ihrer Stabilität bei sinkendem Informationsgehalt der Eingabe. Dazu wurden die einzelnen Verfahren auf verschiedenen skalierten Datensätzen getestet um anschließend ihre Verwendbarkeit unter realen Bedingungen zu testen.

## 4.1 OpenFace im Test

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.

Das Herunterskalieren von Bildern ist nicht das selbe wie eine Aufnahme auf großer Distanz, ist aber ähnlich genug um eine Aussage darüber treffen zu können.

Zur Messung wurde der Datensatz von Labeled Faces in the Wild LFW [15] und BIWI Random Forests for Real Time 3D Face Analysis [9] verwendet.

Der LFW Bilddatensatz enthält Abbildungen verschiedener Personen mit einer durchschnittlichen Abbildung der Breite des Kopf von 95 Pixeln. Bei BIWI beträgt die durchschnittliche Breite 78 Pixel und die durchschnittliche Distanz zwischen Kamera und Proband beträgt etwa 95cm.

### 4.1.1 Auswirkung der Auflösung auf die Detektionsrate

Durch die Aufgabenstellung muss das Verfahren zuverlässig bezüglich der Distanzen bzw. Darstellungsgröße sein.

Bei BIWIk wurde eine Selektion durchgeführt, da die Detektionsrate nur bei 63,4% liegt und durch die Darstellung die Veränderung des Verlauf nur schwer zu erkennen ist.

Um die Grenzen der Methode auszuloten wurde das Bild mit unterschiedlichen Faktoren linear skaliert, um so weiter entferntere Gesichter zu simulieren.

## 4 Evaluation

Um die Detektionsrate zu bestimmen, wurde der Image-Detector von OpenFace auf den skalierten Bildern angewendet und gezählt wie oft der Detektor ein Gesicht erkannt hat. Dabei wurde nicht geprüft ob es sich dabei um ein korrektes Gesicht handelt. Das Ergebnis dieser Messung ist in Abbildung 4.1 dargestellt. Es ist zu erkennen, dass die Wahrscheinlichkeit einer erfolgreichen Detektion ab einer Skalierung von 0,64 bei BIWI (Gesicherter mit etwa 50 Pixel Breite) und 0,46 bei LFW (Gesichter mit 44 Pixel Breite) rapide abnimmt. Bei der in Abschnitt 4.3 beschriebenen Kamera entspricht dies einer Distanz von etwa 4,5m.

### 4.1.2 Auswirkung der verschiedenen Skalierungsverfahren auf die Detektion

Um die Auswirkung der Skalierungsverfahren zu bestimmen, wurden verschiedene Gesichtsgrößen simuliert, indem alle Bilder von BIWI und LFW um den angegebenen Faktor linear verkleinert wurden und mit den angegebenen Verfahren wieder auf die Originalgröße gebracht.

Die Auswirkung der verschiedenen Skalierungsverfahren (Bicubic, Lanczos, Linear, Nearest-Neighbor) auf die Detektionswahrscheinlichkeit ist in Abbildung 4.2 dargestellt.

Es ist zu erkennen, dass die Detektionsrate über einen weiten Bereich, [1; 0,3] bei der Skalierung, nur sehr wenig abnimmt. Durch die Vergrößerung können somit jene Gesichter in Skalierungsbereichen ausgewertet werden, die ohne nicht erkennbar sind. 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 (BIWI 0,34/27Pixel und LFW 0,22/21Pixel) ab, als bei den anderen Verfahren (BIWI 0,22/17Pixel und 0,13/12Pixel). Das Bicubic (blau) und Lanczos (grün) Verfahren haben die höchste Detektionsrate und fallen zuletzt ab, wobei Bicubic minimal besser ausfällt.

### 4.1.3 Auswirkung der verschiedenen Skalierungsverfahren auf den Arbeitsbereich bezüglich Rotation

In Abbildung 4.3 ist der Median der Differenz zwischen per OpenFace berechnetem und im Datensatz angegebenem Kopforientierungswinkel aufgetragen.

Bei der X-Rotation zeigt sich, dass das Bicubic-Verfahren im Vergleich zu den anderen Verfahren etwa 2 Grad mehr abweicht, ein recht hoher Wert, zumal der Fehler von Lanczos, Linear und Nearest-Neighbor bei etwa  $5,8^\circ$  bis zu einer Skalierung von 0,25

#### 4.1 OpenFace im Test

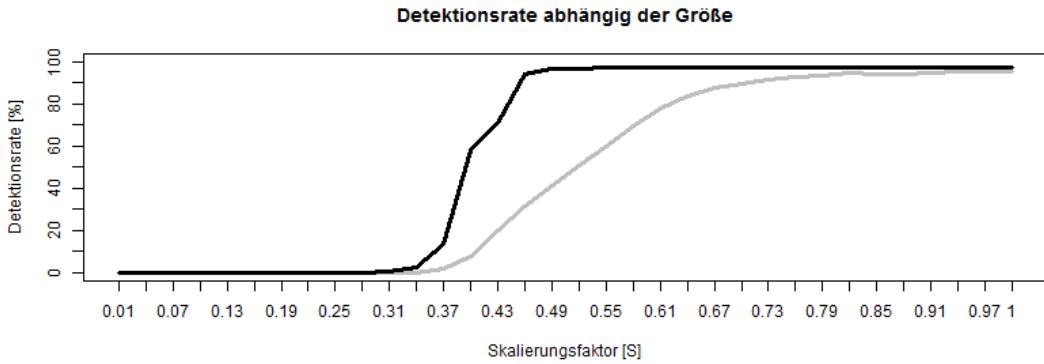


Abbildung 4.1: LFW [15] (schwarz) und BIWI [9] (grau) um den Skalierungsfaktor verkleinert und gegen die Erkennungsrate aufgetragen

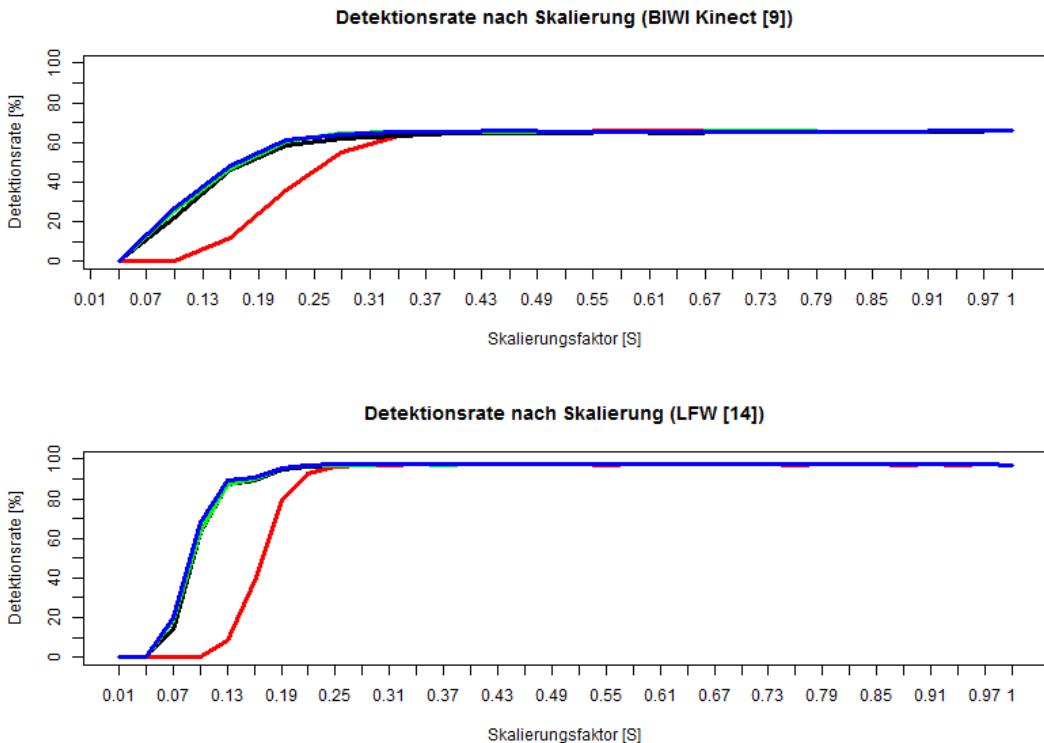


Abbildung 4.2: Bilder um den Skalierungsfaktor verkleinert und mit verschiedenen Verfahren wieder vergrößert.  
Bicubic (blau), Lanczos (grün), Linear (schwarz), Nearest-Neighbor (rot)

## 4 Evaluation

liegt.

Der Median der Fehler auf der Y-Achse (nicken) bleibt ebenfalls gering mit  $4,2^\circ$  vom Bicubic-Verfahren. Dabei liefern die anderen drei Verfahren nahezu identische Ergebnisse mit etwa  $5^\circ$ . Die jeweilige Qualität bleibt nahezu konstant bezüglich der Skalierung.

Die Z-Rotation wird am besten bestimmt mit einer Abweichung von etwa  $2^\circ$  von allen getesteten Verfahren. Dabei ist aber auch zu beachten, dass dieser Wertebereich im Datensatz deutlich geringer ausfällt, als bei den anderen beiden Rotationen.

Für alle Berechnungen zeigt sich, dass der Fehler konstant bleibt, bis zu der Skalierung von 0.25, bei der auch die Detektionsrate deutlich abfällt.

Neben der Qualität der bestimmten Winkel, ist auch der Arbeitsbereich von Interesse in dem Gesichter bei verschiedenen Skalierungen noch erkannt werden können. Ein Gesicht das außerhalb dieses Bereiches liegt kann nicht erkannt und ausgewertet werden.

In Abbildung 6.1 sind die Quantile bei 50%; 80%; 99% und der Maximalwert, von den Rotationswinkel der Bilder aus BIWI [9] dargestellt, in denen ein Gesicht erkannt wurde. Durch den großen Unterschied zwischen dem 80%-Wert, 99%-Wert und dem Maximalwert liegt die Vermutung nahe, dass es sich bei diesen Werten um falsch detektierte Bilder handelt Dennoch kann eine Rotation des Kopfes von 45% in alle Richtungen erkannt und ausgewertet werden.

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.

### 4.1.4 Auswirkung der Skalierungsverfahren auf die Positionsbestimmung

Für eine zuverlässige Auswertung ist auch die Bestimmung der Position von Interesse. Zur Berechnung der Position auf BIWI wurde eine Brennweite der Kinect-Kamera auf 531,15 geschätzt, da es keine Angabe für den Datensatz gab.

Der Median der Differenz zwischen Datensatz und Rechnung ist in Abbildung 4.4 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 die Ergebnisse nahezu irrelevant werden.

Aus der Abbildung ist zu entnehmen, dass die Position in horizontaler und vertikaler Richtung auf etwa 2,5cm genau bestimmt werden kann, die Distanz (Tiefe) auf etwa 2,8cm genau. Dies ist selbst bei sehr klein skalierten Bildern möglich.

Nearest-Neighbor hat bei der Berechnung der X-Position die geringste Abweichung zu den anderen getesteten Verfahren mit einem Unterschied von etwa 1mm.

Bei der Bestimmung der Y-Position hingegen, ist der Unterschied zwischen den vier

#### 4.1 OpenFace im Test

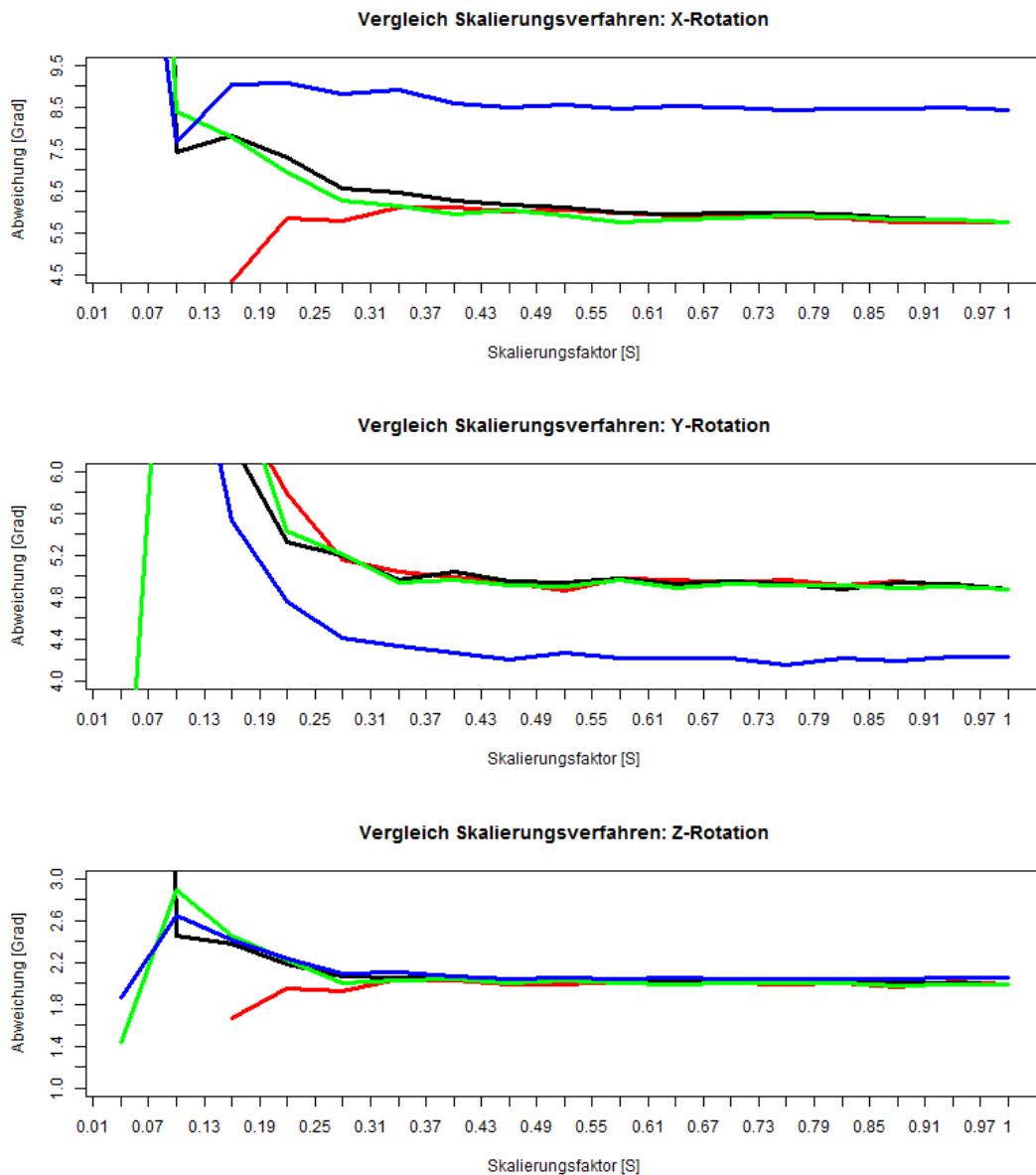


Abbildung 4.3: Dargestellt ist der Median der Abweichung zwischen der berechneten Drehung und der des Datensatzes.  
Bicubic (blau), Lanczos (grün), Linear (schwarz), Nearest-Neighbor (rot)

## 4 Evaluation

Verfahren minimal. Die Differenz beträgt weniger als  $1\text{mm}$  und keines der Verfahren zeugt einen aussagekräftigen Unterschied zu den anderen.

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

### 4.1.5 Auswirkung von Pixelrauschen auf die Detektion

Mit diesem Test soll geprüft werden, welches der Verfahren auch stabil gegenüber Rauschen ist um einen besseren Eindruck für die Verwendbarkeit bei realen Aufnahmen zu erhalten.

Das Rauschen wird simuliert, indem für jedes Pixel eine Wahrscheinlichkeit von 50% besteht auf eine gleichverteilte Abweichung von  $\pm 10\%$  des originalen Farbwertes. Die Bilder von LFW [15] wurden entsprechend verkleinert, mit Rauschen versehen um sie anschließend mit den unterschiedlichen Verfahren zu vergrößern. Dieser Vorgang 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 4.5. Die gesamte Erkennungsrate ist deutlich kleiner als ohne Rauschen, wobei der Skalierungsfaktor von 0.22, ab welcher die Erkennungsrate rapide abfällt, beibehalten wird.

### 4.1.6 Verschiedene Rechenverfahren zur Bestimmung der Position

Um die Qualität der Berechnung auf verschiedenen Distanzen zu ermitteln, wurde der BIWI Datensatz [9] verwendet, da für jedes Gesicht die Position und Orientierung bekannt ist. Um die verschiedenen Distanzen zwischen Probanden und Kamera zu simulieren, wurden die Bilder mit dem angegebene Skalierungsfaktor 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 Tiefenaufnahme, da diese in der Anwendung auch nicht vorhanden ist.

#### Position

Zur Bestimmung der Position gibt es zwei Verfahren, das eine arbeitet direkt mittels Brennweite und Skalierung (Pose) arbeiten oder zusätzlich eine Überführungsmatrix von 3D zu 2D Landmarks verwenden (CorrectPose)).

Die Funktionen PoseCamera und PoseWorld verwenden die einfache Bestimmung

#### 4.1 OpenFace im Test

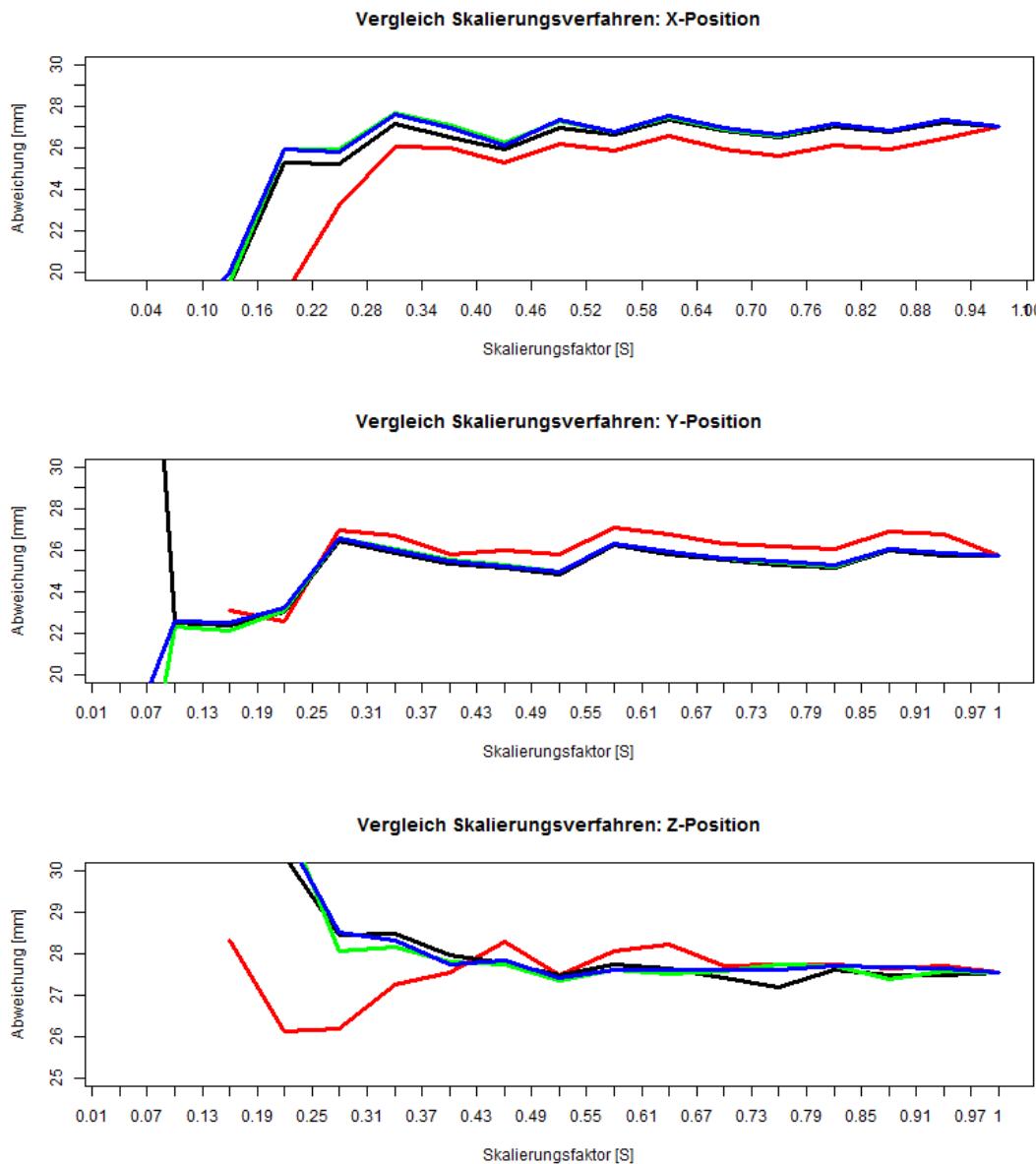


Abbildung 4.4: Dargestellt ist der Median der Abweichung zwischen der berechneten Drehung und der des Datensatzes.  
Bicubic (blau), Lanczos (grün), Linear (schwarz), Nearest-Neighbor (rot)

## 4 Evaluation

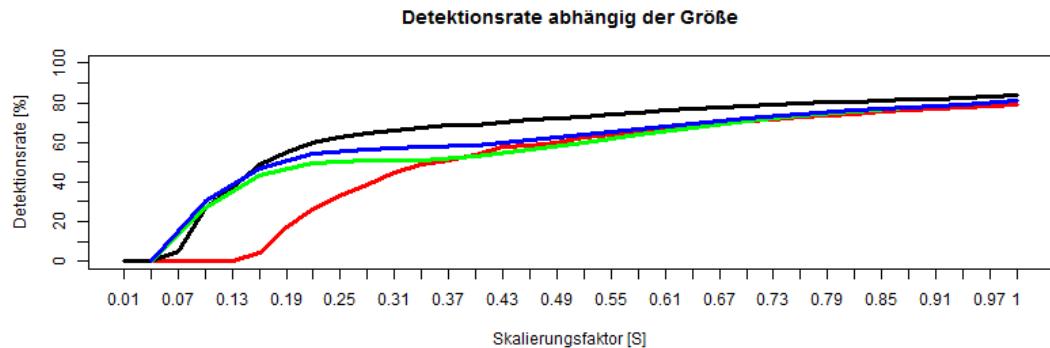


Abbildung 4.5: Bilder aus Labeled Faces in the Wild [15], mit dem X-Faktor verkleinert, um jedes Pixel mit 50% Wahrscheinlichkeit auf  $\pm 10\%$  Gleichverteilung der Abweichung  
Bicubic (blau), Lanczos (grün), Linear (schwarz), Nearest-Neighbor (rot)

mittels Skalierung und CorrectPoseCamera und CorrectPoseWorld die Überführung von 3D und 2D Landmarks, daher überlagern sich die Linien in Abbildung 4.6, 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 Unterabschnitt 4.1.2. 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 Unterschiede zwischen den einzelnen Verfahren, da bei PoseWorld und CorrectPoseWorld auch die Position im Kamerabild berücksichtigt wird.

Aus Abbildung 4.7 ist zu entnehmen, dass die zusätzliche Korrektur das Ergebnis weiter verbessert, wenn die Pixelorientierungen mit beachtet werden.

## Ergebnis

Es zeigt sich, dass PoseWorld, also die einfache Bestimmung der Position mittels Skalierung und Brennweite und zusätzlicher Korrektur der Winkel die besten Ergebnisse liefert im Test.

Im Test ist die Überführung von 3D zu 2D Landmarks in manchen Parametern (z.B.

#### 4.1 OpenFace im Test

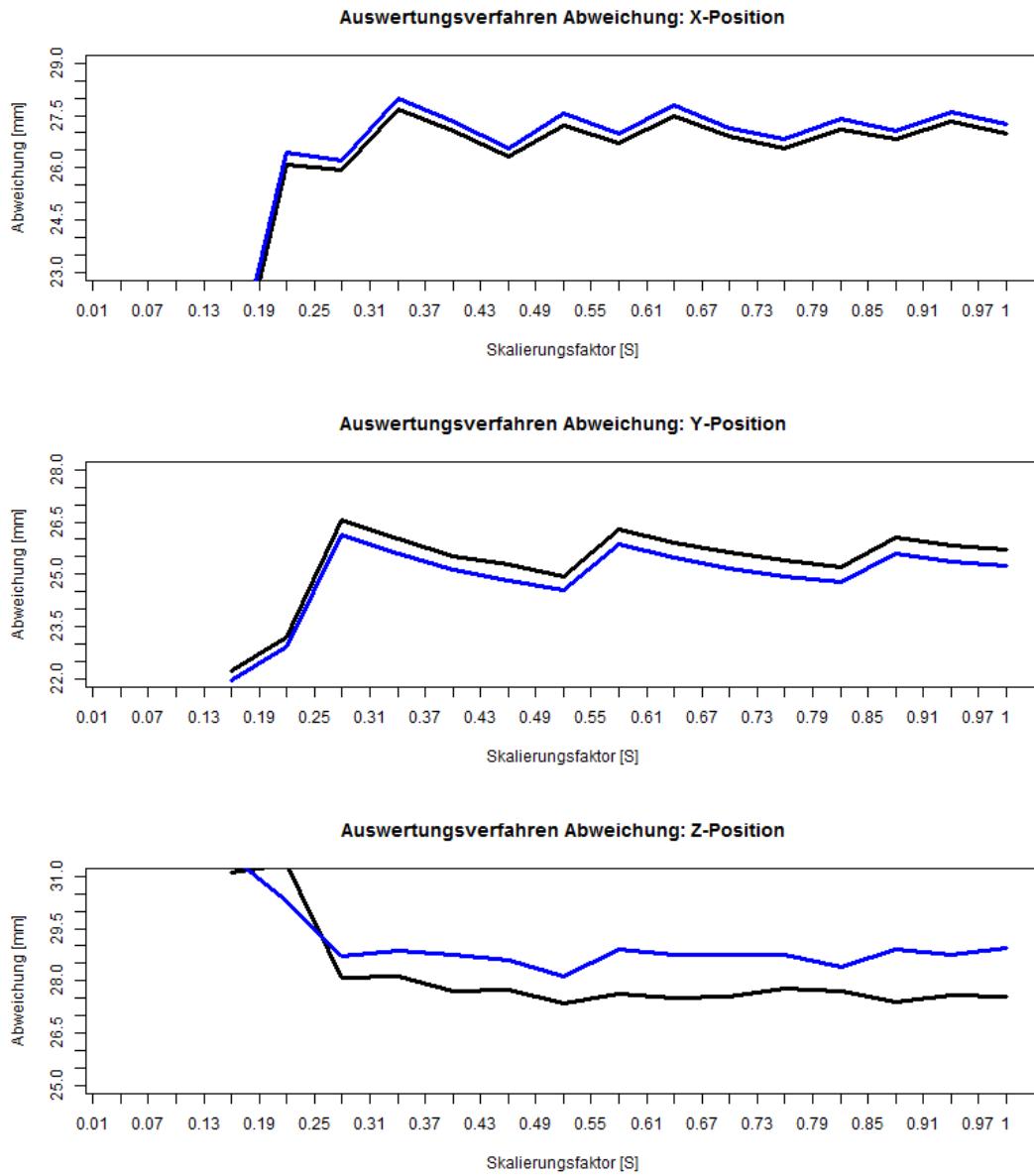


Abbildung 4.6: Median der Abweichung in Millimeter der Positionsbestimmung auf BIWI [9] die mit Lanczos skaliert wurden.  
 PoseWorld (schwarz), PoseCamera (rot, verdeckt von PW), CorrectPoseCamera (grün, verdeckt von CPW) und CorrectPoseWorld (blau)  
 Oben: X-Position, Mitte: Y-Position, Unten: Z-Position

## 4 Evaluation

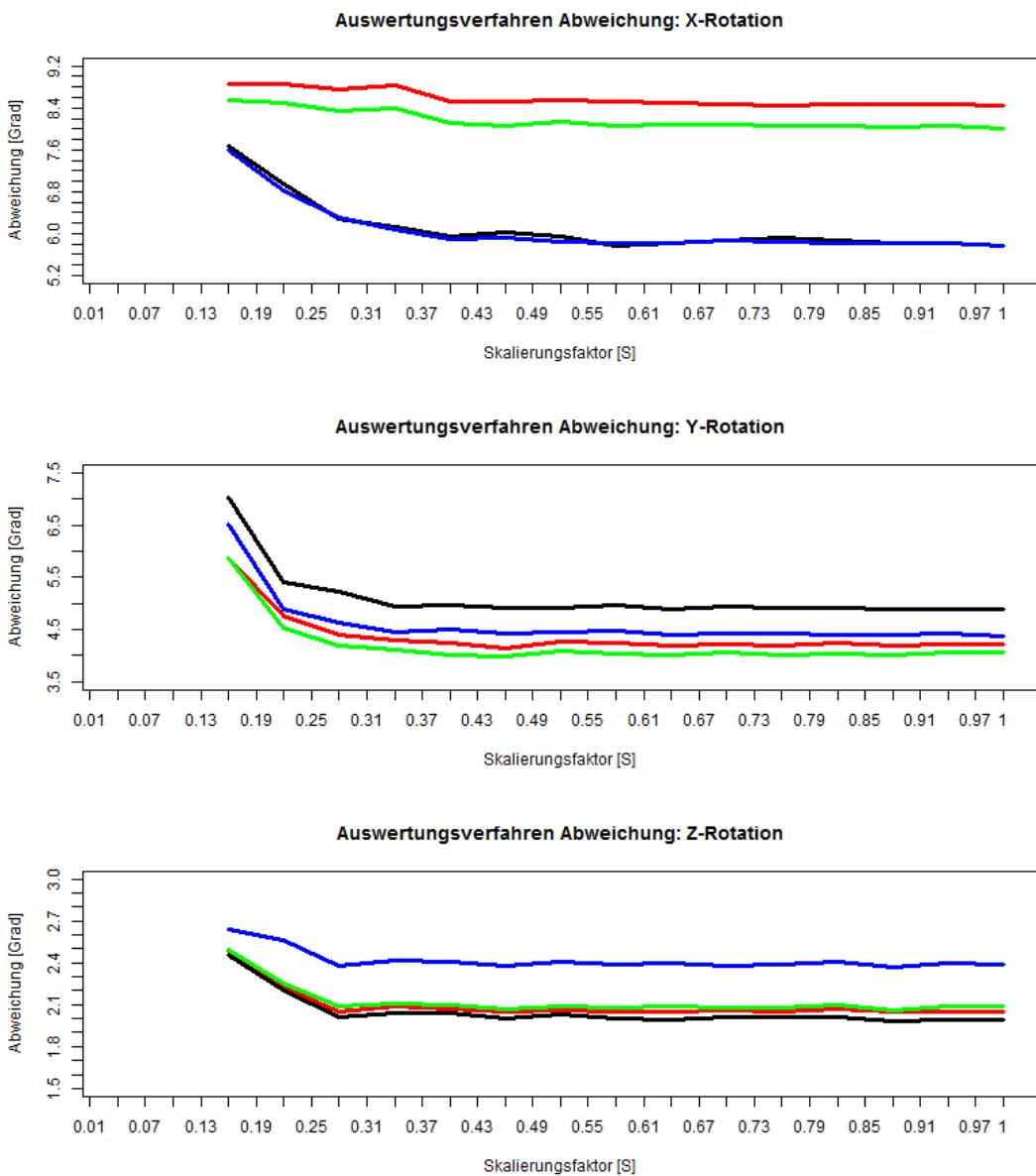


Abbildung 4.7: Dargestellt ist der Median der Abweichung in Grad der Positionsbestimmung auf Bilder die mit Lanczos skaliert wurden.  
 PoseWorld (schwarz), PoseCamera (rot), CorrectPoseCamera (grün) und CorrectPoseWorld (blau)  
 Oben: X-Rotation, Mitte: Y-Rotation, Unten: Z-Rotation

## 4.1 OpenFace im Test

Y-Rotation) überlegen, wobei der unterschied minimal ausfällt. Dies 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.

### 4.1.7 Ergebnis bezüglich Verwendbarkeit

Anhand der Detektionsrate abhängig von der Skalierung, siehe Abbildung 4.1, kann entnommen werden, das Gesichter unter 50 Pixel Größe nicht mehr sinnvoll erkannt werden können. Somit ergibt sich eine maximale Distanz von etwa 4,5m (basierend auf der Actioncam) für eine Analyse.

Da die maximale Distanz auf der gearbeitet werden soll jedoch (8m) beträgt, ergibt sich eine Gesichtsgröße von etwa 22 Pixel. Dies entspricht einer Skalierung von 0,28 für den BIWI-Datensatz. Bei dieser Bildgröße ist in der Standardanwendung ohne Skalierung keine Detektion möglich.

Werden die Bildbereiche hingegen hochskaliert, zeigt der Test, das sogar Gesichter mit einer Größe von unter 22 Pixel (Skalierung 0,25, 8m) gefunden und analysiert werden können. Dies bedeutet, dass mit diesem Trick auch mit der Hälfte der Bildinformationen noch gearbeitet werden kann, wenn die Eingabe dadurch dem Trainingsdatensatz eher entspricht.

Für eine erfolgreiche Analyse sind die Parameter Detektionsrate, Qualität der Rotation und Qualität der Position relevant. Daher wurden die verschiedenen Skalierungsverfahren in diesen Parametern bei unterschiedlichen Größen der Eingabebilder verglichen. Die höchste Detektionsrate bei den Skalierungen erreicht Bicubic, wobei der Unterschied zu Lanczos und Linear so minimal ausfällt, das sie als Gleichwertig in diesem Bereich betrachtet werden kann. Es zeigt sich auch die deutliche Schwäche von Nearest-Neighbor, die Detektionsrate nimmt deutlich früher ab als bei den anderen.

Bei der Bestimmung der Rotation kann nur ein geringer Unterschied zwischen den einzelnen Verfahren erkannt werden. Für die X-Rotation hat das Bicubic-Verfahren einen um etwa 2° größeren Fehler als die anderen, bei der der Y-Rotation hingegen 0,8° genauer ist. Bei der Z-Rotation ist kein klarer Unterschied zu erkennen, somit ist bei diesem Parameter die Wahl des Verfahrens egal.

Zur Bestimmung der Position ist das lineare Verfahren am besten geeignet, da es den kleinsten Fehler aufweist, wobei der Unterschied mit 1mm sehr gering ausfällt.

Der Test mit dem Pixelrauschen soll etwaige Bildfehler simulieren, wie es bei schlechten Kameras der Fall sein kann, was 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. In diesem Test erreicht das lineare Verfahren die höchste Detektionsrate, diesmal ist der Unterschied zwischen den einzelnen Verfahren deutlich besser erkennbar.

## 4 Evaluation

Somit erfüllt das lineare Verfahren die Parameter am besten, wobei der Unterschied zwischen den einzelnen recht gering ausfällt und die Wahl des Skalierungsverfahren von anderen Kriterien abhängig gemacht werden kann, wie z.B. von der Rechenzeit.. Dabei kann vom Nearest-Neighbor abgeraten werden wegen dem deutlich früheren Abfall der Detektionsrate.

Theoretisch wären sogar Distanzen bis zu 14m (12Pixel) möglich, basierend auf der hohen Auflösung der Actioncam für eine erfolgreiche Detektion.

Der Unterschied bei Skalierung 1 (Originalgröße) zu der Ergebnissen in Paper [26] kann an der Verwendung des Median anstelle des Durchschnitts als auch an der verwendung von verschiedenen Kameraparametern liegen.

## 4.2 ElSe im Test

Der Ursprüngliche ElSe-Algorithmus wurde für Eye-Tracking Brillen entwickelt, also für ein qualitativ hochwertiges Bild eines Auges, 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 [38] 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 ist der Augenbereich genauer bestimmt als im Datensatz enthalten, daher wurde der Bildbereich soweit verkleinert, dass 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, beschrieben in Unterabschnitt 2.8.2, 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.

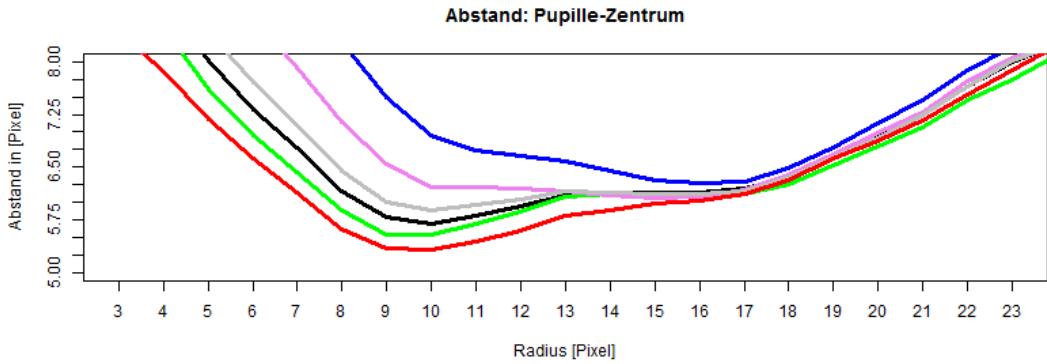


Abbildung 4.8: Median-Abstand in Pixel des Zentrums der Pupille gegen die Veränderung des Radius des Filters.

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

#### 4.2.1 Auswirkung des Filterradius

Ein wichtiger Parameter des ElSe-Verfahrens ist der Radius des Filters. Um den besten Parameter zu bestimmen wurde der Augen-Datensatz [38] verwendet und die Augenpartie ausgeschnitten. Im Datensatz besitzen die abgebildeten Augen durchschnittlich eine Pupille mit 15 Pixel und eine Iris von 34 Pixel Durchmesser.

In Abbildung 4.9 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 2.9.1, muss ElSe in diesem Aspekt zuverlässig Ergebnisse liefern.

Im Versuch hat sich ein Radius von etwa einem Zwölftel des zu erwartenden Durchmessers 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 Pixel am besten, siehe Abbildung 4.8, wobei dieser Fehler nicht so sehr steigt bei Veränderung des Radius, als bei der Größenbestimmung von Pupille und Iris.

#### 4.2.2 Auswirkung der verschiedenen Graubild-Verfahren

Es zeigt sich, dass die Verfahren mit denen der Farbwert in einen Grauwert überführt wird, durchaus Auswirkungen auf die Qualität der Berechnung haben.

Für die Bewertung der Verfahren werden folgende Kriterien verwendet: Die Differenz

## 4 Evaluation

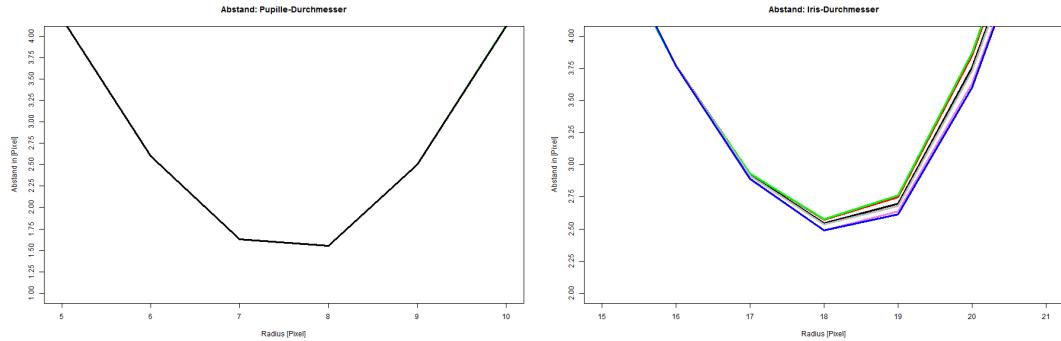


Abbildung 4.9: Differenz zwischen den Radien gegen die Veränderung des Radius des Filters von Pupille (links) und Iris (rechts)

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

zwischen den berechneten und tatsächlichen Radien von Pupille und Iris sowie die Abweichung des berechneten Zentrums der Pupille.

Der minimale Abstand der berechneten Zentren ergibt sich bei dem Gleam-Verfahren mit 5,327 Pixel als Median, siehe Abbildung 4.8. 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 4.9 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 wurden 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 Standartverfahren ist. Mit allen Verfahren wurde die Berechnung zur Pupille/Zentrum/Iris für verschiedene groß skalierte Bilder bestimmt mit ihren jeweiligen optimalen Filterradien.

Bei der Berechnung der Pupille auf den unterschiedlich großen Abbildungen ist weiterhin kein Unterschied zu erkennen, siehe Abbildung 4.11 oben.

Auch bleiben die Unterschiede der Verfahren erhalten und die Fehler auf dem gleichen Niveau bis zu einer Skalierung von 0,15, ab welcher die Berechnung bei allen Verfahren scheitert. So liefert das Gleam-Verfahren die besten Ergebnisse im Bezug auf das Zentrum, wo hingegen das Quatrat-Verfahren geeignet für die Bestimmung des Iris-Radius ist.

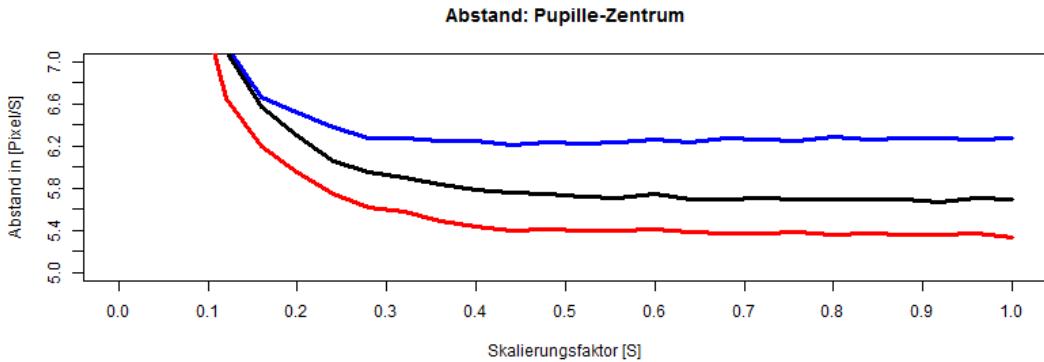


Abbildung 4.10: Euklidischer Abstand in Pixel zwischen dem berechneten Zentrum der Pupille und dem des Datensatzes gegen die Veränderung des Radius des Filters.  
Verfahren: Gleam (rot), Luminance (schwarz), Quadrat (blau)

### 4.2.3 Vergleich zu OpenFace

Als Referenz wird das Ergebnis von OpenFace, für die zusätzlich bestimmten Landmarks der Augen, verwendet. Dies wurde auch auf dem Augendatensatz [38] angewendet, um vergleichbare Ergebnisse zu erhalten.

Wird Abbildung 4.12 mit Abbildung 4.10 bzw. Abbildung 6.11 verglichen so ist zu erkennen, dass OpenFace im Mittel einen geringen Fehler bis zu einer Skalierung von 0,47 besitzt als ElSe. Ab diesem Wert hat ElSe einen geringen mittleren Fehler, da die Abweichung fast unverändert bis 0,12 beibehalten wird.

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

Aus 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 damit die Landmarks im Bildausschnitt liegen. Somit kann dieser Bildausschnitt als Eingabe von ElSe verwendet werden.

### 4.2.4 Ergebnis

Die Tests haben ergeben, dass ElSe mit einem Radius von 10 Pixel auf Bildern die mithilfe von Gleam ins Graue überführt wurde die besten Ergebnisse liefert. Dabei ist das Verfahren stabil gegenüber der Skalierung und kann die Iris bis zu einer Größe von 3 Pixel erkennen, das einer Distanz von etwa 4m entspricht (Basierend auf der Actioncam).

## 4 Evaluation

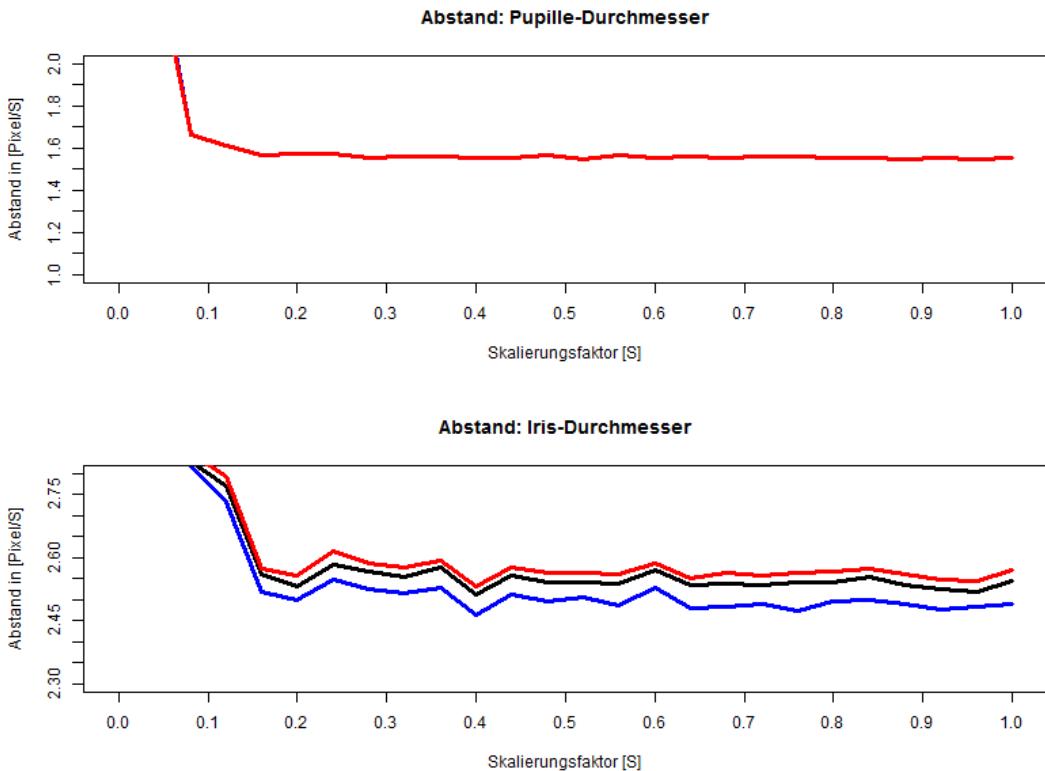


Abbildung 4.11: Differenz in Pixel zwischen den Radien der Berechnung und dem des Datensatzes gegen die Veränderung des Radius des Filters.  
 Oben: Pupille mit Filterradius 8, Unten: Iris mit Filterradius 18  
 Verfahren: Gleam (rot), Luminance (schwarz), Quadrat (blau)

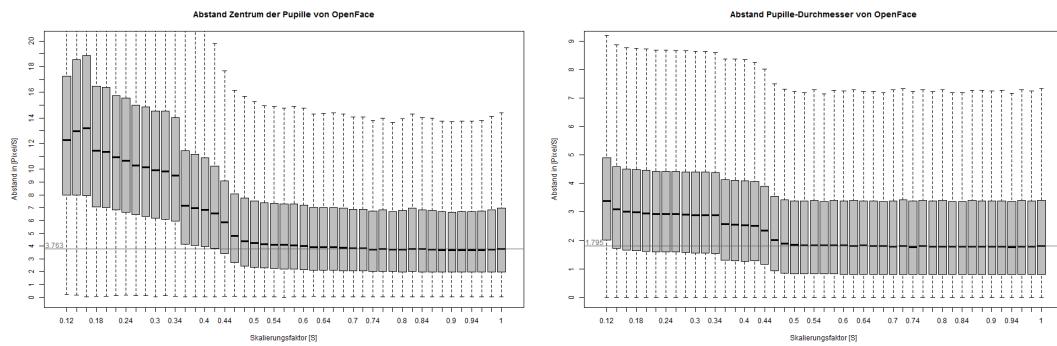


Abbildung 4.12: Auswirkung der Bildgröße auf die Qualität der Augendetektion von OpenFace. Aufgetragen ist die Abweichung [Pixel/Skalierung] gegen den Skalierungsfaktor.

### 4.3 Versuch 1 - Arbeitsbereich der Verfahren

Allerdings hat der Vergleich ergeben, das bis zu einer Skalierung von 0,47 ElSe schlechtere Ergebnisse liefert als das OpenFace-Augen-CNN.

So kann das Ergebnis von OpenFace bei Bildern 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 17 und 15 Pixel ( $0,5 - 0,44$ ) können beide Ergebnisse kombiniert werden, da sie ungefähr gleich gute Ergebnisse liefern um den Gesamtfehler zu minimieren, da die beiden Verfahren unabhängig voneinander arbeiten.

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.

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 dargestellt, die Auswirkung der Skalierung in Abbildung 6.11.

Bei realen Aufnahmen sind Bildfehler unvermeidlich, so können Reflexionen (Brille, Kontaktlinse usw.), Make-Up und körperliche Eigenschaften wie Augenfarbe die Detektion erschweren. Ein Problem das schon im originalen Test [37] aufgetreten ist, wenn der Farbunterschied zwischen Iris und Pupille recht gering ausfällt oder durch Reflexionen der Kantenverlauf gestört wird, wodurch die maximale Distanz in der eine Auswertung der augen möglich ist weiter sinkt.

## 4.3 Versuch 1 - Arbeitsbereich der Verfahren

Mit diesem Versuch soll der Zusammenhang zwischen Standort eines Probanden und Position des Blickziels (Targets) untersucht werden. Dazu wird eine Klassenzimmerumgebung simuliert, in der sowohl Standort als auch Blickziel relativ zur Kamera bekannt sind.

Als Messinstrument für die Versuche 1 und 2 wurde die Explorer 4K Actioncam verwendet, da sie eine hohe Auflösung bei ausreichend *FPS* und eine  $170^\circ$  Weitwinkel-Linse mit großer Schärfentiefe besitzt. Mit ihrer 2,7K Einstellung wird ein  $2688 \times 1520$  Farbvideo mit  $30FPS$  aufgezeichnet.

Allerdings ist die Bildqualität durch Pixelrauschen und Ähnliches deutlich schlechter als die Verkleinerung der Originalaufnahmen in den Datensätzen.

### 4.3.1 Versuchsaufbau

In einem Raum wurde die Kamera in  $2,06m$  Höhe  $31cm$  hinter den Targets so montiert, das der gesamte Raum im Fokus liegt. Als Targets wurden 9 Punkte auf einer Ebene

#### 4 Evaluation

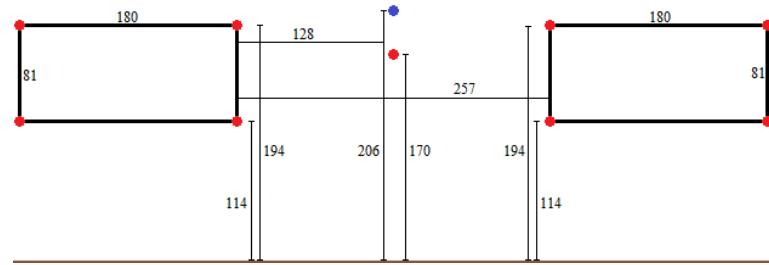


Abbildung 4.13: Aufbau der Targets im Vorversuch, alle Angaben gerundet in Zentimeter  
rote Punkte: Target, blauer Punkt: Kamera

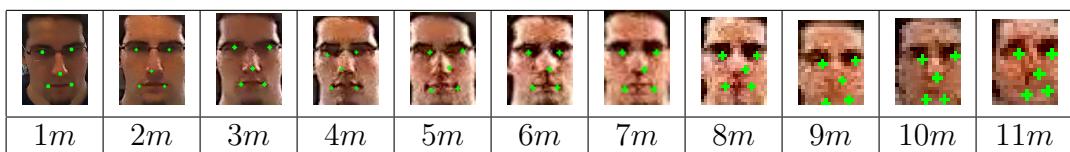


Abbildung 4.14: Dargestellt ist die Box und die 5 Landmarks von MTCNN-Face bei verschiedenen Distanzen des Probanden zur Actioncam

markiert mit der Kamera im Zentrum. Die Anordnung der Targets ist in Abbildung 4.13 dargestellt.

Als Position der Probanden wurde ein Rasterfeld mit 1m Kantenlänge im Raum eingezeichnet auf einer Fläche von  $7 \times 11\text{m}$ . Die Probanden stellten sich auf diesen Positionen auf um nacheinander alle Targets zu betrachten.

#### 4.3.2 Detektion mit MTCNN

Um die Detektionswahrscheinlichkeit des MTCNN-Face Detektors zu testen wurden diese Videos analysiert.

Es zeigt sich, das auf allen Positionen die Probanden erfolgreich erkannt wurden und die Boxen das Gesicht recht gut beschreiben. 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 4.14

### 4.3 Versuch 1 - Arbeitsbereich der Verfahren

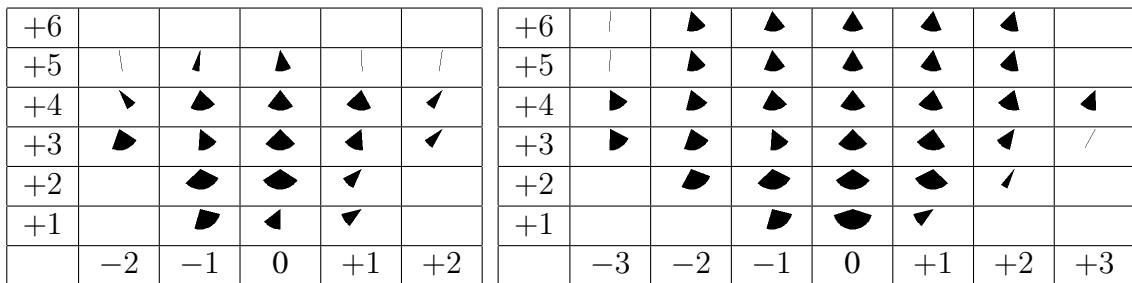


Abbildung 4.15: Dargestellt ist der horizontale Winkelbereich in denen ein Gesicht erkannt wurde. Angaben in Meter  
Links: Einzelbilder, Rechts: Video

#### 4.3.3 Auswertung der Aufnahme

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

Als erstes wurde die Einzelbildauswertung von OpenFace auf die Frames angewendet und jene Abbildungen der Kopfrotationen markiert, in denen erfolgreich ein Gesicht erkannt wurde. In Abbildung 4.15 links ist der horizontale Wertebereich dargestellt in dem 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 dem auf der horizontalen Achse an den entsprechenden Positionen ein Gesicht erkannt wurde, ist in Abbildung 4.15 rechts dargestellt.

Um alle Verbesserungen in einer realen Umgebung auszutesten, wurde wie in Kapitel 3 beschrieben vorgegangen und die relevanten Bildausschnitte linear vergrößert. Die Auswirkung auf den horizontalen Bereich ist in Abbildung 6.14 dargestellt. Durch diese Verbesserung wird die Distanz auf der gearbeitet werden kann mehr als verdoppelt bei der Video- und Einzelbild-Analyse.

#### 4.3.4 Ergebnis

Es zeigt sich, dass eine Auswertung von OpenFace auf einem Video deutlich zuverlässiger arbeitet als auf Einzelbildern, vor allem der größere Arbeitsbereich bezüglich der Rotation ist von Vorteil.

Durch die Verwendung des Weitwinkelobjektivs, kann die gesamte Breite eines Klassenzimmers erfasst werden und der Arbeitsbereich der Auswertung ist für eine erfolgreiche Detektion und Analyse breit genug um Schüler erfassen zu können, die selbst die vorderen Eckpunkte eines Klassenzimmers betrachten.

## 4 Evaluation

Bei der Distanz zur Kamera (Tiefe) besteht Handlungsbedarf, als Ziel wurde  $8m$  angesetzt und das aktuelle Verfahren endet bei  $5m$  in der einfachen Ausführung. Wird der Bildbereich allerdings verbessert, so verdoppelt sich die Distanz zwischen Kamera und Person, wodurch eine Abdeckung des gesamten Klassenzimmers erreicht wird. Der in Unterabschnitt 4.1.2 theoretisch bestimmte Detektionsabstand von  $14m$  konnte nicht erreicht werden, die erreichte Maximaldistanz liegt bei etwa  $10m$ , immer noch ausreichend für ein Klassenzimmer. Als Ursache kann das Pixelrauschen und die Überbeleuchtung durch das einfallende Licht der Fenster angenommen werden.

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

Eine signifikante Aussage bezüglich des vertikalen Winkel kann aus diesem Aufbau nicht getroffen werden, da die Neigungswinkel zwar differenziert werden können, sie allerdings zu ähnlich sind bei stehenden Personen (beides mal fast horizontal).

Von Interesse ist ob ein Schüler auch erkannt werden kann, wenn dieser auf den Tisch vor sich schaut, was einen sehr großen Neigungswinkel bedeutet. Um dies aus zu testen wurde Versuch 2 durchgeführt.

### 4.4 Versuch 2 - Arbeitsbereich bezüglich der Neigung des Kopfes

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

#### 4.4.1 Versuchsaufbau

Für diesen Versuch wurde die Kamera auf  $1,88m$  Höhe und  $3m$  vor dem vordersten Standort der Probanden aufgestellt.

Als Standorte wurde eine Markierung mit einem Meter Abstand zueinander auf einer 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.

#### 4.4 Versuch 2 - Arbeitsbereich bezüglich der Neigung des Kopfes

+3m								
	-3m	-2m	-1m	Kamera	+1m	+2m	+3m	+4m
+3m								
	-3m	-2m	-1m	Kamera	+1m	+2m	+3m	+4m

Abbildung 4.16: Dargestellt ist der horizontale Winkelbereich in denen ein Gesicht erkannt wurde.

Oben: Einzelbilder, Unten: Video

+9m								
+3m								
	-3m	-2m	-1m	Kamera	+1m	+2m	+3m	+4m
+9m								
+3m								
	-3m	-2m	-1m	Kamera	+1m	+2m	+3m	+4m

Abbildung 4.17: Dargestellt ist der horizontale Winkelbereich in denen ein Gesicht erkannt wird bei aufbereitetem Eingabebild.

Oben: Einzelbilder, Unten: Video

#### 4.4.2 Auswertung

Es wurde die selbe Auswertung wie in Versuch 1 verwendet, einzelne Frames mit bekanntem Target. In Abbildung 4.16 ist der vertikale Arbeitsbereich dargestellt, wiederum ist der Arbeitsbereich bei Verwendung der Video-Analyse etwas größer mit  $50^\circ$ , wobei ein Winkel von  $50^\circ$  nach unten immer erfasst werden kann.

Durch die Verbesserung der Eingabebilder kann auf einer Distanz von 9m gearbeitet werden, siehe Abbildung 4.17, mit dem selben Arbeitsbereich.

#### 4.4.3 Ergebnisse

Es zeigt sich, dass eine Videoanalyse auch bei starker Neigung nach unten möglich ist. Die Einzelbildauswertung liefert erneut etwas schlechtere Ergebnisse als die Videoauswertung.

Dabei funktioniert das Tracking nur, wenn die Versuchsperson zuerst in die Kamera geschaut hat, um es zu beginnen. Aus diesem Test lässt sich ableiten, dass auch eine

## *4 Evaluation*

Analyse erfolgen kann, wenn der Schüler sich auf den Tisch vor ihm konzentriert. Auch die stärkere gleichmäßige Beleuchtung ist hilfreich, da sie Probleme durch Gegenlicht und Schatten reduziert.

## **4.5 Versuch 3 - Berechnung auf der Augenpartie**

Um einen Eindruck von ElSe mit hochauflösenden Aufnahmen zu erhalten, wurde ein weiterer Versuch durchgeführt mit einer deutlich besseren Kamera. Dieser Versuch soll auch die Nutzbarkeit von Eye-Tracking in solch einem Szenario testen.

### **4.5.1 Versuchsaufbau**

Als Messinstrument wurde die Sony ILCE-6000 verwendet. Diese liefert ein  $6000 \times 4000$  Pixel großes Farbbild bei einer Brennweite von  $16mm$ . Als Standort der Probanden wurden die selben wie in Versuch 1 verwendet. Da es sich um eine Fotokamera handelt, wurde ein Datensatz von Einzelbildern erstellt. Dabei wurden nur Aufnahmen gemacht, bei denen die Probanden direkt in die Kamera schauten.

### **4.5.2 Auswertung**

Von Interesse ist die Augenpartie und die Ergebnisse des OpenFace Eye-Detektor im Vergleich zu ElSe. Dabei wurde ElSe in der Basiskonfiguration eingesetzt, dies bedeutet das Luminance-Verfahren, siehe Unterabschnitt 2.7.3 als Graukonvertierer und einem Radius der Maske von 12 Pixel. Für die Analyse wurde zuerst mit OpenFace das Gesicht soweit analysiert um die Augenpartie als Eingabebild zu bestimmen, siehe Abbildung 4.18 und ein Beispiel in Abbildung 4.19 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 4.19 Mitte oben. Als Ergebnis wurde aus den berechneten Ellipsen von ElSe die Landmarks der Pupille und Iris abgeleitet und im selben Farbschema dargestellt.

### **4.5.3 Ergebnis**

Die Augenpartie der Probanden an den verschiedenen Positionen ist in Abbildung 6.15 dargestellt, die sich bei der angegebenen Distanz frontal vor der Kamera befand. Es ist

#### 4.5 Versuch 3 - Berechnung auf der Augenpartie

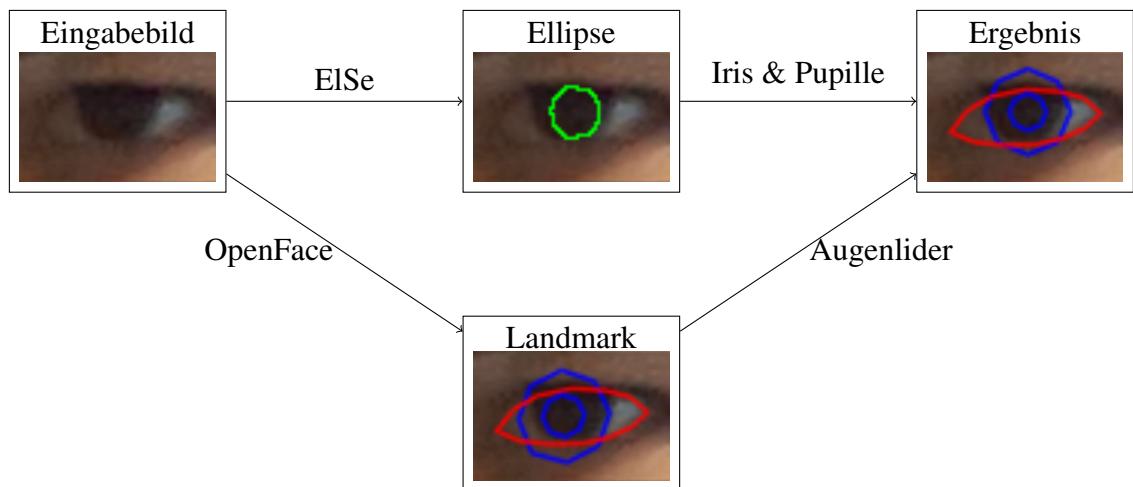


Abbildung 4.18: Dargestellt sind der Ablauf um die Landmarks des Auges zu verbessern

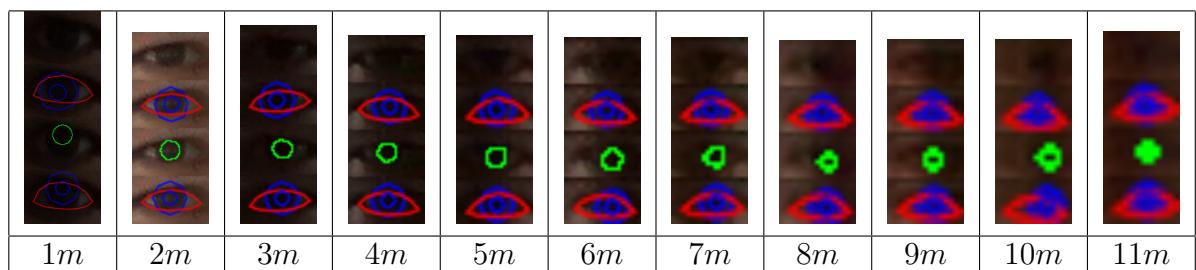


Abbildung 4.19: Ergebnisse von OpenFace und ElSe bei verschiedenen Distanz.  
Von Oben nach Unten: Augenparie, Ergebnis OpenFace, Ergebnis ElSe, ElSe Ergebnis als Landmarks

## 4 Evaluation

zu erkennen, das selbst bei einer hohen Auflösung die Augenpartie sehr klein ausfällt und nur schwierig auszuwerten ist.

Somit zeigt sich, dass 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, dass eine ausreichende Beleuchtung gebraucht wird, da die Augenregion sehr dunkel ausfällt.

## 4.6 Versuch 4 - Aufmerksamkeitsmessung

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 4.21 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 einem 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 4.20.

Als Aufnahmegerät wurde die Logitech c920 HD Pro Webcam verwendet, diese liefert ein 15FPS Video mit einer Auflösung von  $1600 \times 896$  Pixel und besitzt einen horizontalen Blickwinkel von etwa  $70^\circ$ .

### 4.6.1 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 eine weitere Messung der exakten Position verzichtet.

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

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

#### 4.6 Versuch 4 - Aufmerksamkeitsmessung



Abbildung 4.20: Foto der Versuchsdurchführung

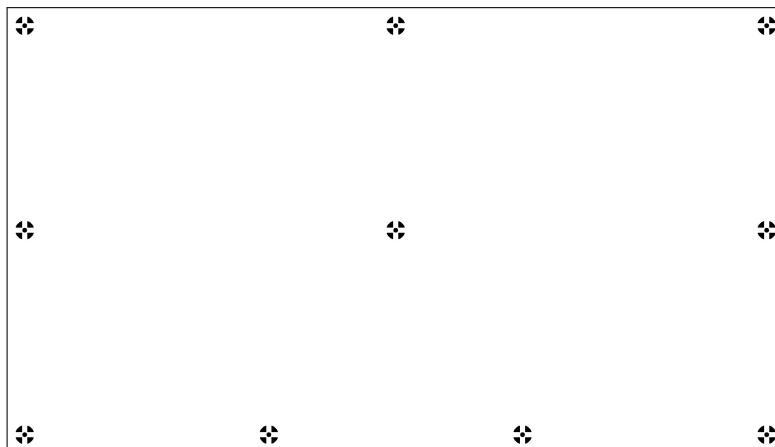


Abbildung 4.21: Eckpositionen des bewegten Ziels bei der Videoaufnahme

## 4 Evaluation

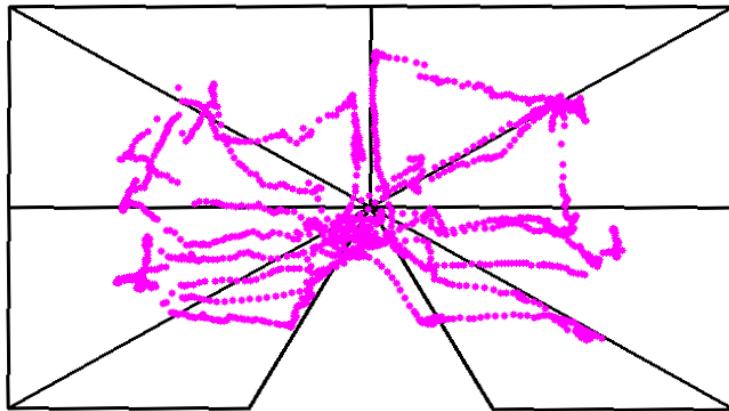


Abbildung 4.22: Dargestellt sind alle gemessene Auftreffpunkte der Gesichtsorientierung auf die Leinwand (rosa) und des Targets (schwarz)

### Erster Eindruck

Dargestellt in Abbildung 4.22 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 den 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, dass 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 4.23), so sind diese etwa  $5^\circ$  auseinander. Genug um einzelne Bereiche differenzieren zu können.

Bei der Bestimmung des vertikalen Winkels zeigt sich, dass 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 recht gering.

Die bestimmte Blickrichtung ist, trotz Verbesserung durch ElSe und Mittlung beider

## 4.6 Versuch 4 - Aufmerksamkeitsmessung

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  $20^\circ$  weit auseinander, dabei liegen diese Punkte im Original etwa  $90^\circ$  weit auseinander, mit dieser Verteilung ergibt sich 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 4.23, dies entspricht einer Distanz von etwa  $14m$ . Auf der horizontalen Achse liegt der Abstand der Quartale etwa  $9^\circ$  weit auseinander, nur  $4^\circ$  mehr als im Original. Bei der Bestimmung des vertikalen Winkels ergibt sich ein ähnliches Verhalten, wobei vor allem der Wertebereich auf  $30^\circ$  sinkt.

Das Ergebnis der Blickrichtung kann bei der 0.01 Skalierung nicht verwendet werden, da die Differenz zwischen dem rechten und linken Maximalwert nur  $8^\circ$  beträgt und die Quartale sich fast vollständig überschneiden.

Überraschend ist das Ergebnis bei dem Skalierungsfaktor von 0,05, da dies einer Distanz von ca.  $24m$  entspricht. 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 könnten. Die Abweichung der einzelnen Messungen ist in Abbildung 6.13 dargestellt.

### 4.6.2 Fehleranalyse im Versuch

Da nur der Unterschied zwischen Target und Auftreffpunkt der gemessenen Gesichtsorientierung aufgezeigt werden kann, kommt es zu verschiedenen Fehlern. Vor allem wird der Bewegung das Targets 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 Kopfdrehung unangenehm und das Ende der Bewegung des Targets absehbar wird. So wird der letzte Teil der Bewegung wieder nur von den Augen nachverfolgt.

Die allgemeine Exkursionen beträgt etwa  $20^\circ$ , der Winkelbereich der üblichen Augenbewegungen, und kann daher recht stark von der Kopforientierung abweichen.[32]

## 4 Evaluation

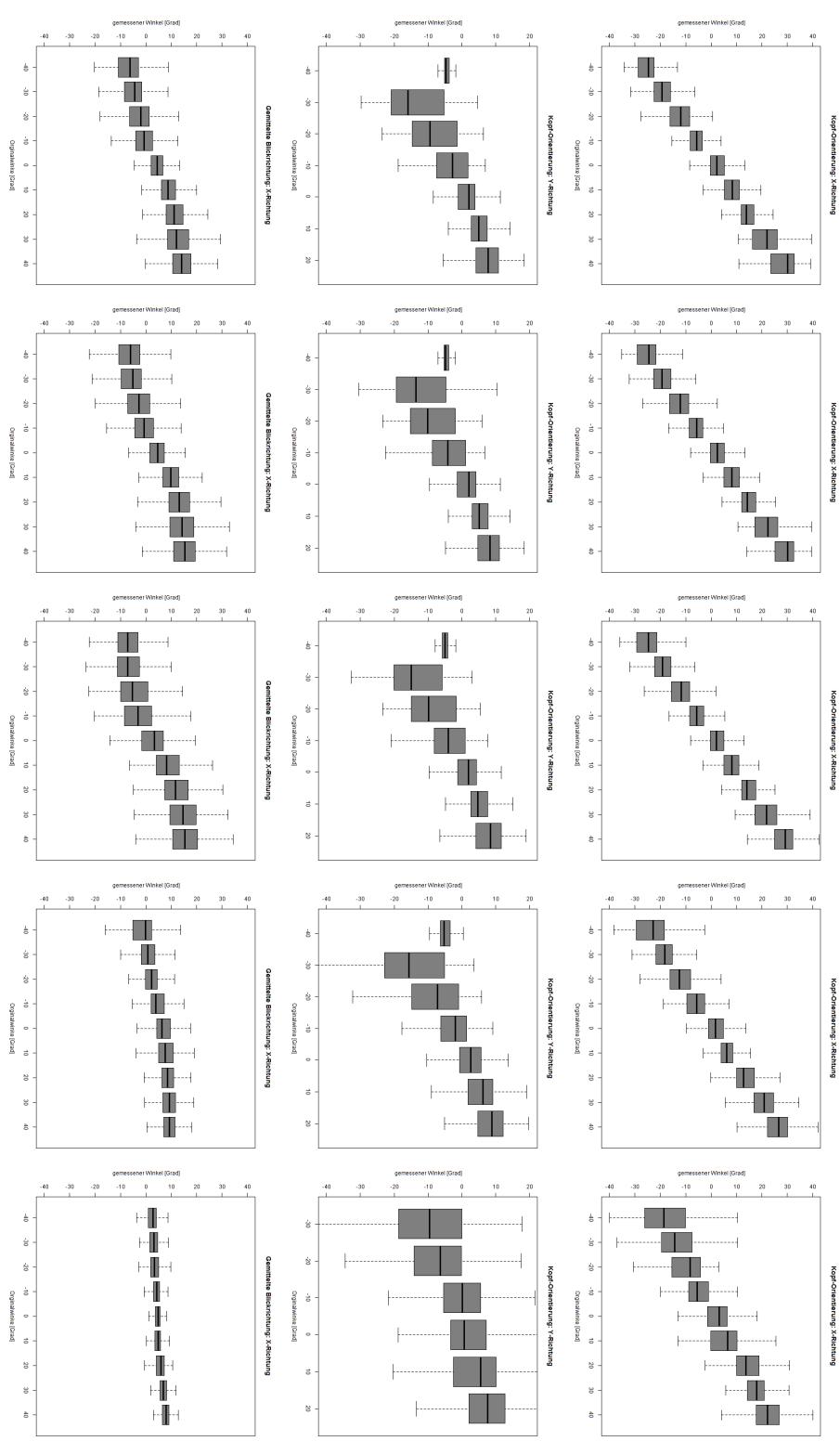


Abbildung 4.23: Auswertung der Videoaufnahme mit der Kopfausrichtung Horizontal (Oben), Kopforientierung Vertikal (Mitte) und die X-Ausrichtung der Augen (Unten)

Skalierungsfaktor von links nach rechts (1/0.5/0.25/0.1/0.05), Y-Achse:  $[0 - 35]^\circ$

## 4.6 Versuch 4 - Aufmerksamkeitsmessung

### Bei der 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  $1\text{cm}$  in alle Richtungen aufweist. Außerdem liegt der Ursprung des Kopfes in der Anwendung etwas tiefer und weiter hinten als der ausgemessene Nasenrücken.

Auch die Parameter für die 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. Dadurch musste diese vor dem direkten Licht geschützt des Projektors geschützt werden. Somit konnte sich die Kamera nicht im Zentrum der Messpunkte befinden.

Da Kamera und Leinwand fest montiert sind, ergibt sich auch die Problematik, dass der Kopf der Probanden nicht im Zentrum des Kamerabildes ist 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, dass 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 wird die Auswertung der Augen weiter erschwert durch die Reflektion von starken Lichtquellen (wie z.B. Fenster, Lampen, Projektorbild) auf den Brillen, die die Pupille überdecken kann. Auch Schatten gerade in den Augenhöhlen erschweren die Auswertung.

## 4.7 Zusammenfassung

Für die Analyse der Gesichter in einem Video wurden zuerst die einzelnen Gesichter mittels MTCNN-Face Detection (Abschnitt 2.4) in allen vorhanden Frames gesucht. Dieses Verfahren ist robust genug, dass auch kleinste Gesichter im Bild erkennen kannt werden können und auch recht stabil bezüglich der Rotation. Somit ist es als Gesichtsdetektor geeignet um in einem Frame die Gesichter zu finden.

Anschließend wird jede Einzelperson unterschieden und alle gefundenen Bildbereiche der jeweiligen Person zugeordnet. Diese Bildbereiche werden nun auf eine Mindestgröße gebracht (Abschnitt 2.5), damit sie dem Trainingsdatensatz des nächsten Schrittes stärker ähneln. Dazu wurde die Auswirkung der verschiedenen Skalierungsverfahren auf die nachfolgende Analyse untersucht und das lineare-Verfahren als das brauchbarste identifiziert.

Nun werden die einzelnen Bildbereiche ausgewertet (Abschnitt 2.6) und die Gesichtsorientierung kann bestimmt werden. Um die Bereiche zu simulieren in denen das Verfahren eingesetzt werden kann, wurden die Bilder des Trainingsdatensatzes durch lineare Skalierung verkleinert.

Um die Detektion der Pupille zu verbessern wurde ElSe (Abschnitt 2.8) verwendet, mit dem Ziel, die Blickrichtung exakter zu ermitteln. Um dieses Verfahren zu optimieren, wurde die Auswirkung der verschiedenen Farbbild nach Graubild Konvertierer (Abschnitt 2.7) untersucht, sowie die Veränderung des Radius der Maske und die Stabilität der Ergebnisse auf linear verkleinerten Eingabebildern. Die Messung hat ergeben, dass eine Verbesserung durch ElSe vor allem bei sehr kleinen Bildern möglich ist, der Augendetektor von OpenFace allerdings auch sehr gute Ergebnisse liefert.

Um eine Übersicht über den Arbeitsbereich zu erhalten, wurden verschiedene Versuche durchgeführt um die maximalen Kopfrotationen und Distanzen zu bestimmen. Dabei konnte gezeigt werden, dass der Wertebereich in dem eine Auswertung möglich ist, ausreicht um das gesamte Klassenzimmer mit nur einer Kamera zu erfassen. Dabei zeigte sich, dass gerade die Bestimmung der Blickrichtung auf großer Distanz meist nicht möglich ist, da die Augenpartie viel zu klein für eine Berechnung ist. So bleibt meist nur die Gesichtsorientierung mit ihr natürlichen Ungenauigkeit.

Abschließend wurde getestet, wie zuverlässig das gesamte Verfahren auf Videos unter realen Bedingungen eingesetzt werden kann, um die Aufmerksamkeit zu ermitteln, siehe Abschnitt 4.6. Dazu wurde ein Versuch durchgeführt, bei dem die Probanden ein Ziel verfolgen sollten und ermittelt wie exakt das Ziel der Aufmerksamkeit bestimmt werden kann. Es zeigte sich, dass schon in der Originalgröße eine Auswertung der Blickrichtung recht Fehlerbehaftet ist. Vor allem bei Probanden die dem Target vor allem mit den Auge gefolgt sind, zeigt sich die Problematik einer zuverlässigen Auswertung.

Da Bewegung erlaubt ist, passiert es immer wieder, dass Teile des Gesichtes verdeckt

#### *4.7 Zusammenfassung*

werden, durch Hände beim Melden, andere Schüler oder den 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. Eine Problematik die schon in den Versuchen aufgetreten ist und mit großer Sicherheit auch in einem Großversuch auftreten wird.

Des weiteren haben die Tests gezeigt, dass bei Verwendung von Einzelbildern der maximale Winkel relativ zur Kamera beträchtlich sinkt. Außerdem kann bei Verwendung eines Videos das Gesicht deutlich kleiner dargestellt sein bis keine Auswertung mehr möglich ist als für die Einzelbilder.



## 5 Diskussion

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

Dieser Problematik kann mit mehreren Kameras entgegen gewirkt 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 zu erfassen.

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 etwas verdeckt und der Unterricht dadurch wenig beeinflusst wird. 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.

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, wenn sie von einer Kamera erfasst werden, die vor der Klasse aufgestellt wurde. Mit entsprechend hochauflösenden Kameras können auch bessere Resultate auf größeren Distanzen erzielt werden, wobei ein großer Unterschied eher ausbleiben wird, sollte die gesamte Klasse auf einmal aufgenommen werden, da die Schüler recht weit verteilt sind und die Auflösung durch den großen Distanzunterschied begrenzt ist.

Als Messinstrument ist eine oder mehrere Videokameras sinnvoll, da die Qualität der Auswertung auf Videos die begrenzte Auflösung bei weitem wettmacht.

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 das Tracking soweit zu ergänzen, dass auf Profilbildern gearbeitet werden kann um Landmarks zu erkennen. Somit kann das Tracking auch begonnen werden, wenn die Probanden nicht grob in Richtung Kamera blicken und ist gegenüber Drehungen robuster.

Auch der Einsatz von Weitwinkelobjektiven 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 werden

## *5 Diskussion*

allerdings deutlich größer dargestellt und die Kamera kann passend zur Position der Schüler aufgestellt werden.

Abschließend kann die Aussage getroffen werden, dass es mit dem aktuellen Stand der Technik möglich ist, mehrere Personen mit nur einer einzigen Kamera soweit zu analysieren, dass eine Aussage über ihre Blickrichtung gemacht werden kann.

# 6 Abbildungen

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

## 6.0.1 Boxplot

Folgende Angaben gelten für alle dargestellten Boxplots.

- Die schwarze Mittellinie in der Box zeigt den Median der Messwerte an.
- Die Box beschreibt das obere 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“[27]
- 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.

## 6.0.2 Anzahl der Messwerte

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

### Biwi Random Forests for Real Time 3D Face Analysis [9]

Alle Darstellungen und Auswertungen bezüglich der verschiedenen Skalierungsverfahren haben folgende Anzahl an Messwerten bei den angegebenen Skalierungen, Maximum bei Skalierung 1 mit 9800.

Die geringe Anzahl der Messwerte bei 0,04 kann keine begründete Aussage gemacht

## 6 Abbildungen

werden, außerdem weichen die Werte so sehr von den anderen ab, dass sie als False-True erkannten Gesichtern angesehen werden können [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	$\geq 0,31$
Bicubic	3	1190	4545	5888	7147	8329	8991	9439	9561	> 9600
Lanczos	3	1224	4206	5696	6941	8224	8958	9400	9548	> 9700
Linear	1	776	3935	5439	6851	8019	8625	9107	9313	> 9400
Nearest-N.	0	0	0	193	2081	4374	5976	7825	8595	> 9200

## Augen-Datensatz [38]

Die einzelnen abgebildeten Boxplots basierend auf dem Augen-Datensatz [38] besitzen mindestens 10.000 Messwerte.

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

## Aufmerksamkeitsmessung - Werte im Versuch

Für diese Auswertung ergeben sich folgende Werteverteilungen, dabei wurde der durchschnittliche Fehler (Mean-Error) über den gesamten Datensatz bestimmt. Die Einteilung der Boxplots erfolgte durch das natürliche Runden auf die Zehnerstelle der wahren X bzw. Y Werte. Die ungleichmäßige Verteilung der Messwerte liegt am Versuchsaufbau und ein gewisser Teil des Rauschens kann auf den Wertebereich zurückgeführt werden, die in einer Box zusammengefasst wurden ( $\pm 5^\circ$ )  
Dies Angaben beziehen sich auch Abbildung 4.23 und Abbildung 6.13)

Skalierung 1:

X-Mean-Error =  $8,971^\circ$ ; Y-Mean-Error =  $10,08^\circ$ ; EyeAVG-X-Mean-Error =  $17,49^\circ$

Winkel [Grad]	-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 =  $8,927^\circ$ ; Y-Mean-Error =  $10,07^\circ$ ; EyeAVG-X-Mean-Error =  $17,07^\circ$

Winkel [Grad]	-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 =  $8,742^\circ$ ; Y-Mean-Error =  $9,772^\circ$ ; EyeAVG-X-Mean-Error =  $17,07^\circ$

## 6.1 Graphen

Winkel [Grad]	-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 [Grad]	-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 [Grad]	-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		

## 6.1 Graphen

## 6 Abbildungen

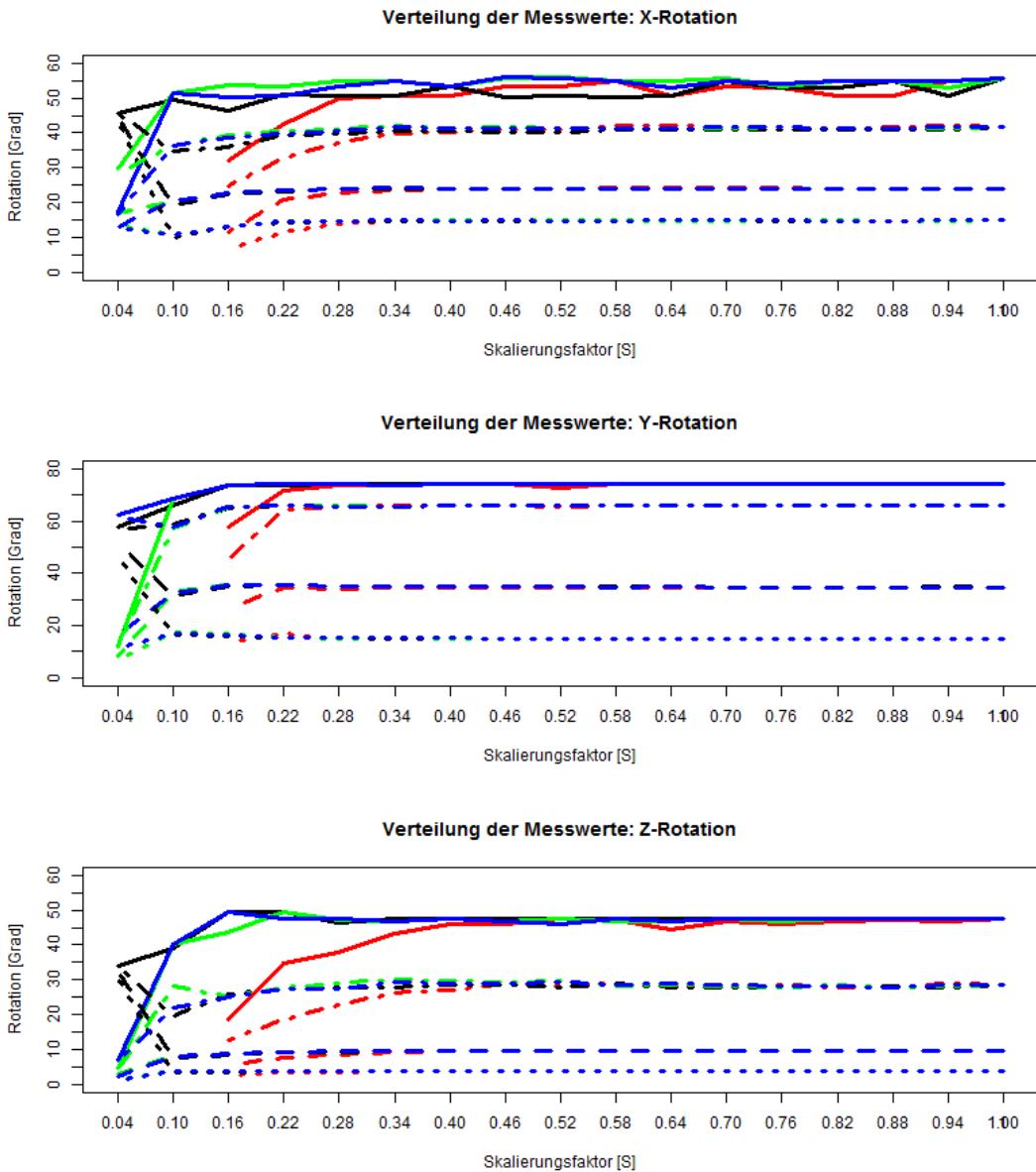


Abbildung 6.1: Dargestellt ist der Bereich in denen im BIWI [9] ein Gesicht erkannt wurde.

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

Maximal erreichter Wert: \_\_\_\_\_

99% Quantile der Messwerte: -----

80% Quantile der Messwerte:-----

Median aus den Messwerten: -----

## 6.1 Graphen

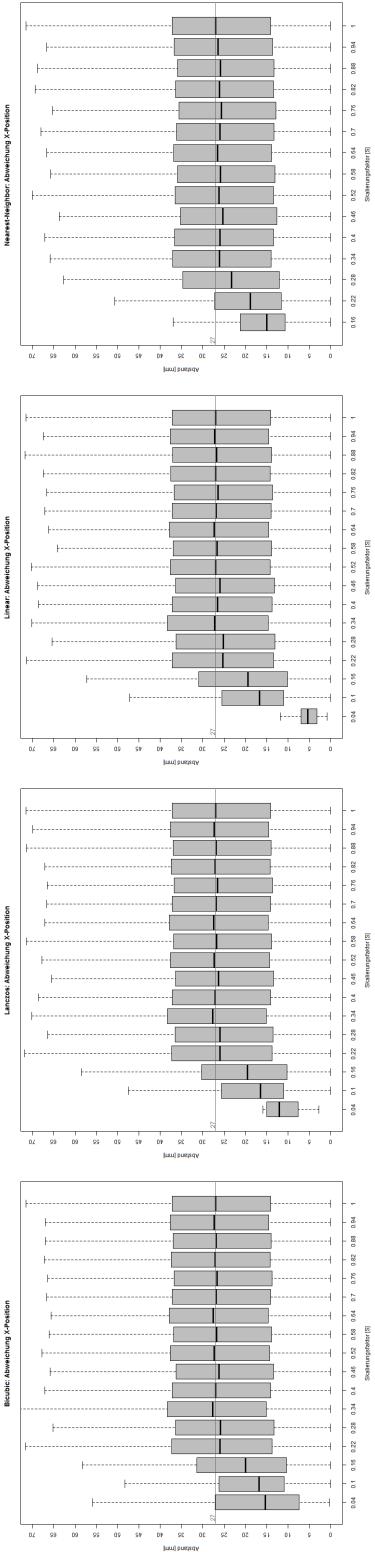


Abbildung 6.2: Zusammenhang zwischen der Skalierung und der Abweichung in X-Richtung in Millimeter.  
Von rechts nach links: Bicubic, Lanczos, Linear, Nearest-Neighbor

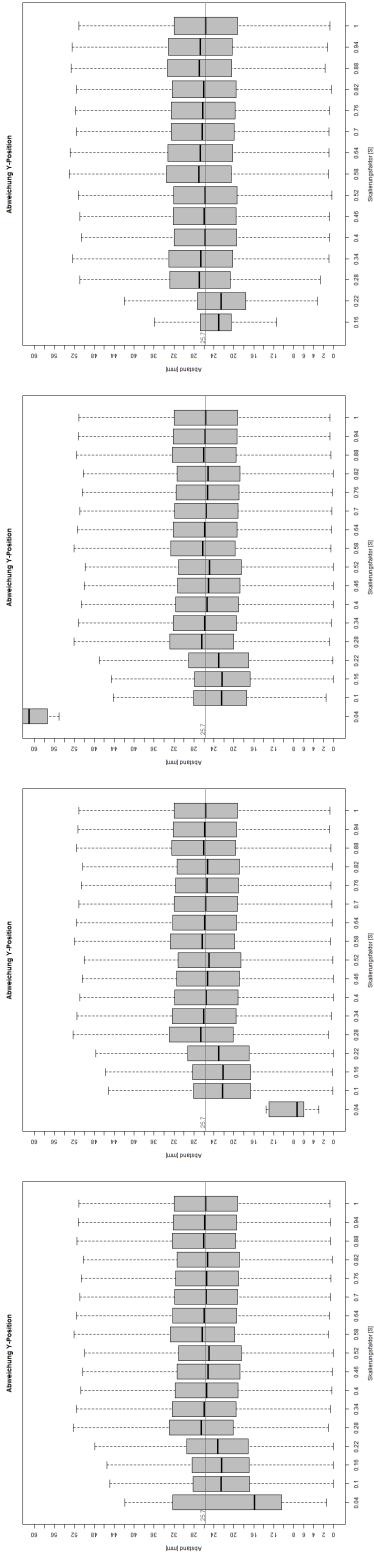


Abbildung 6.3: Zusammenhang zwischen der Skalierung und der Abweichung in Y-Richtung in Millimeter.  
Von rechts nach links: Bicubic, Lanczos, Linear, Nearest-Neighbor

## 6 Abbildungen

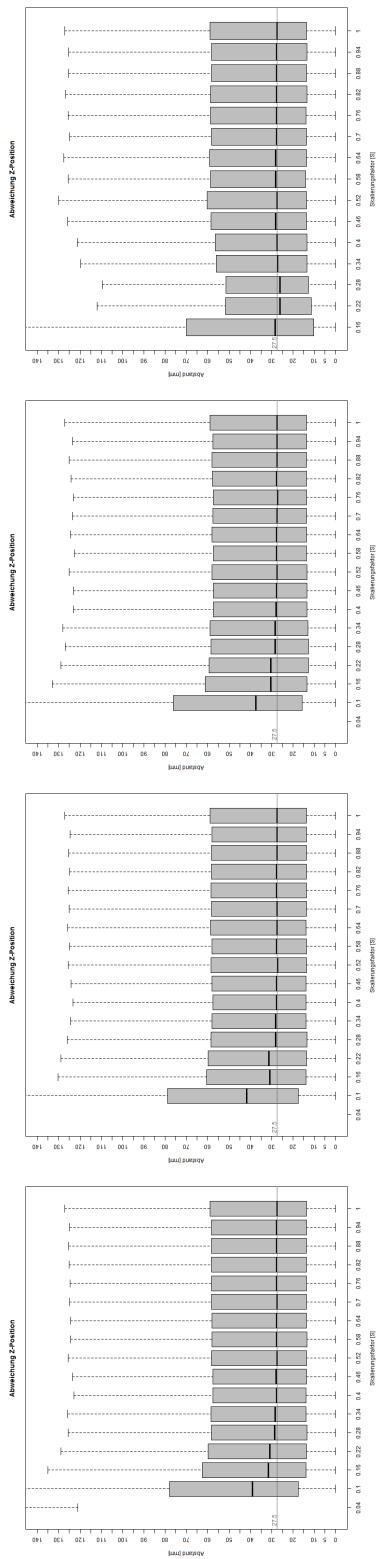


Abbildung 6.4: Zusammenhang zwischen der Skalierung und der Abweichung in Z-Richtung in Millimeter.  
Von rechts nach links: Bicubic, Lanczos, Linear, Nearest-Neighbor

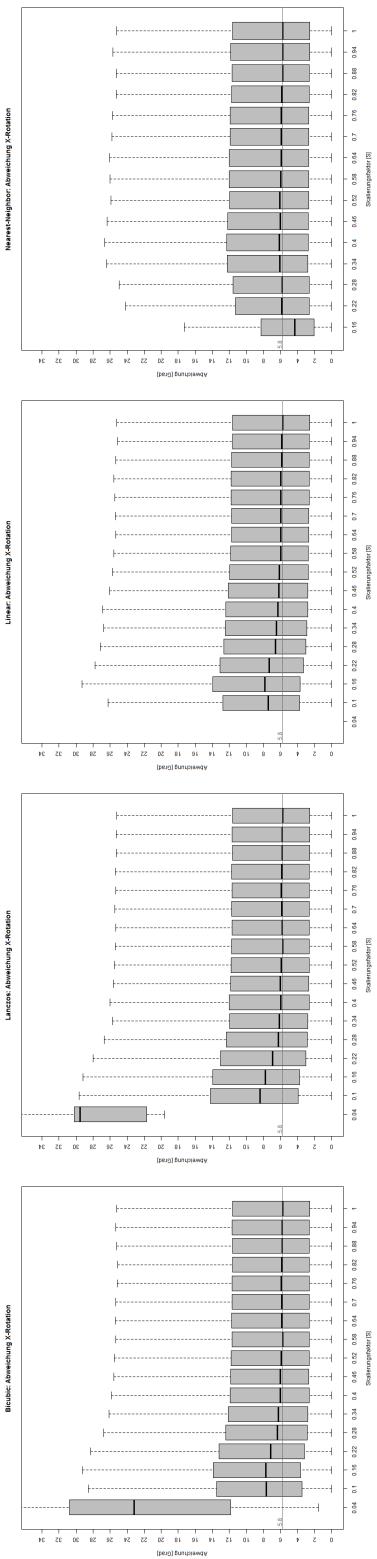


Abbildung 6.5: Zusammenhang zwischen der Skalierung und der Abweichung des Winkels in X-Richtung, Angabe in Bogenmaß.  
Von rechts nach links: Bicubic, Lanczos, Linear, Nearest-Neighbor

## 6.1 Graphen

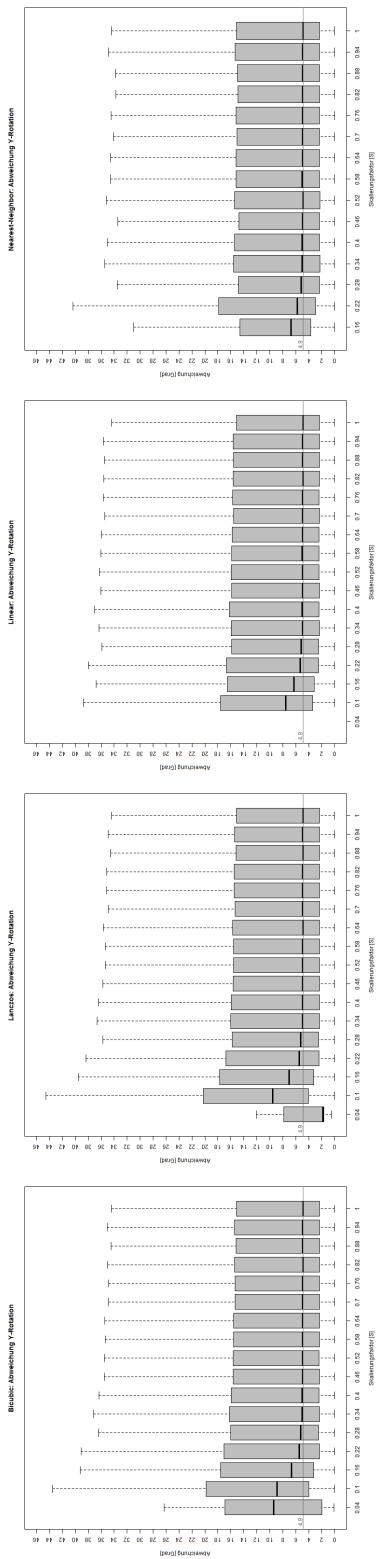


Abbildung 6.6: Zusammenhang zwischen der Skalierung (X-Achse) und der Abweichung des Winkels in X-Richtung, Angabe in Bogenmaß.  
Von rechts nach links: Bicubic, Lanczos, Linear, Nearest-Neighbor

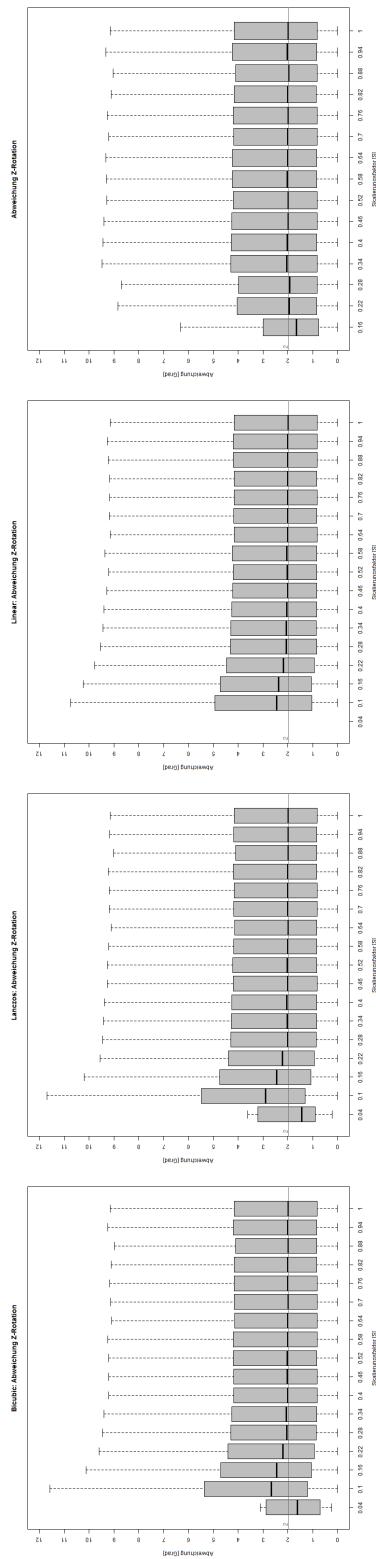


Abbildung 6.7: Zusammenhang zwischen der Skalierung (X-Achse) und der Abweichung des Winkels in Y-Richtung, Angabe in Bogenmaß.  
Von rechts nach links: Bicubic, Lanczos, Linear, Nearest-Neighbor

## 6 Abbildungen

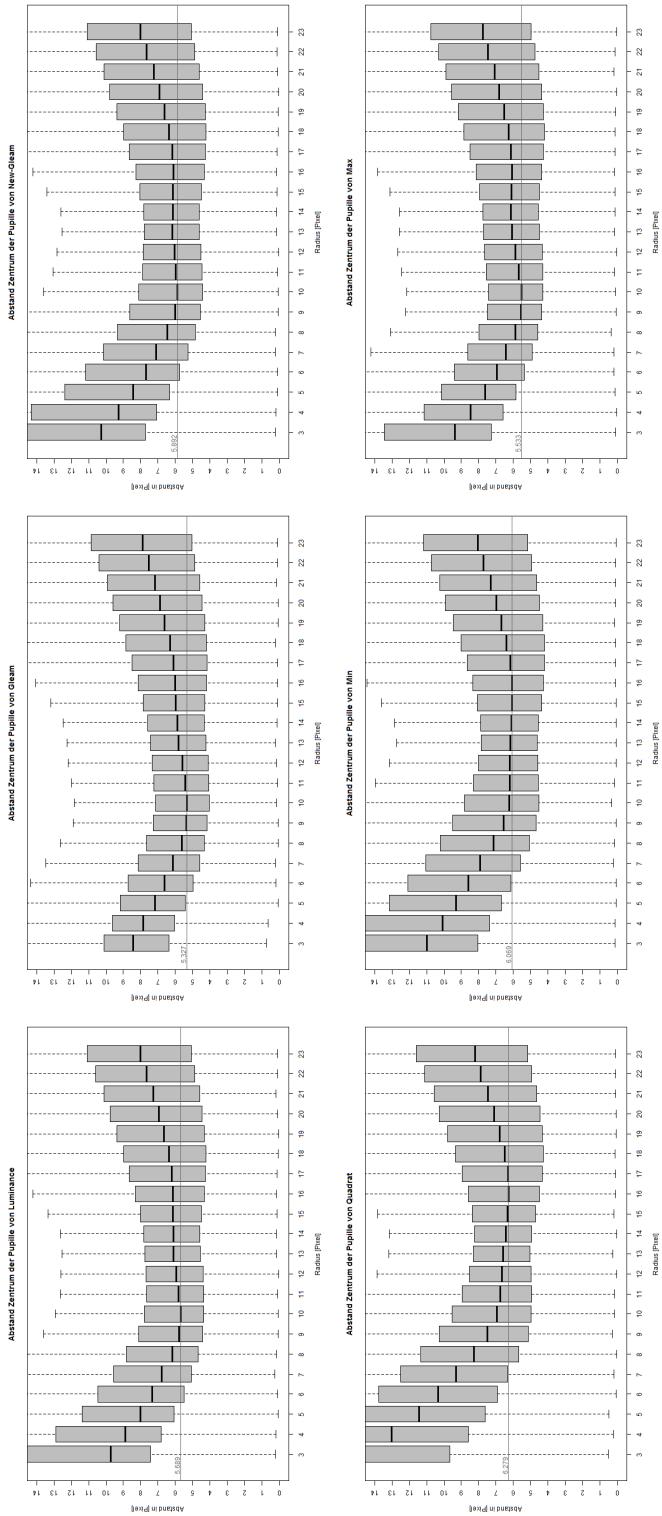


Abbildung 6.8: Abstand des Zentrums der Landmark-Pupille und der berechneten Ellipse in [Pixel] gegen den Radius-Größe des Filters.

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

## 6.1 Graphen

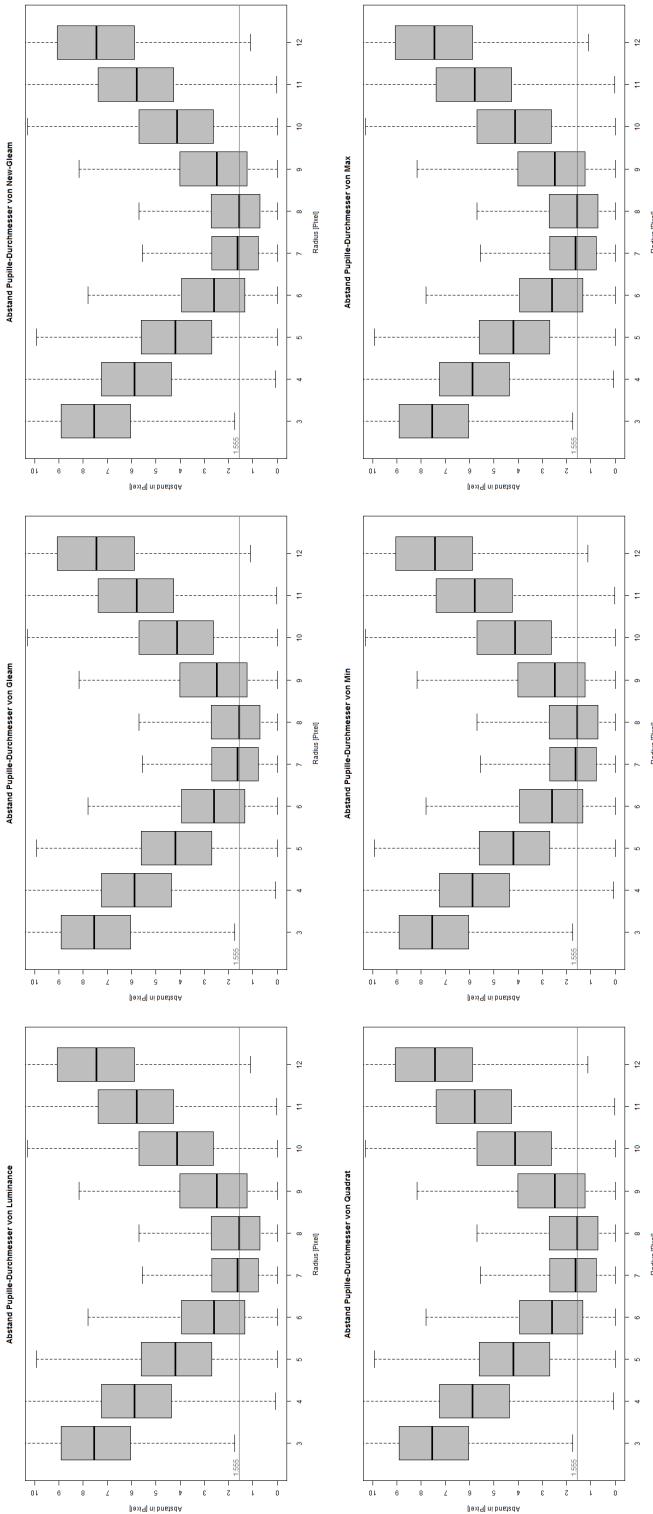


Abbildung 6.9: Unterschied Zwischen den Radien der Landmark-Pupille und der Berechneten Ellipse in [Pixel] gegen den Radius-Größe des Filters

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

## 6 Abbildungen

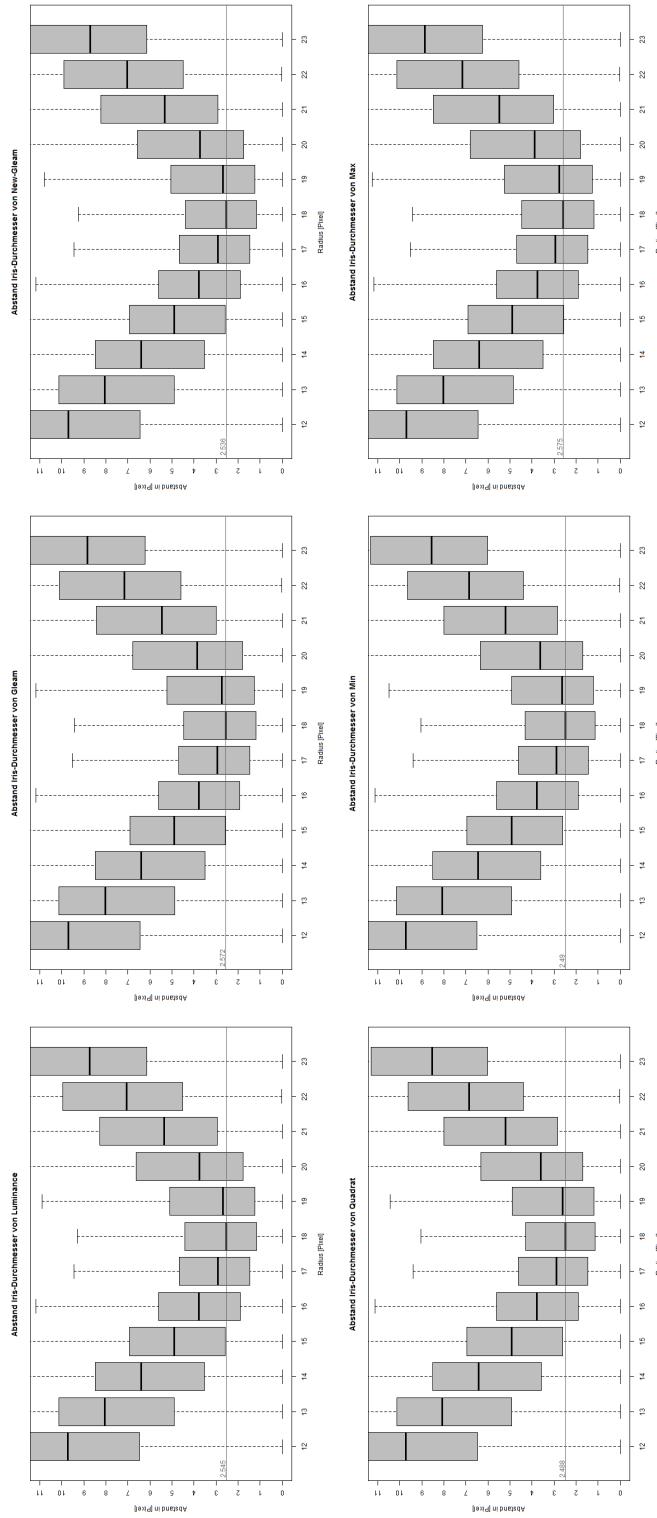


Abbildung 6.10: Unterschied Zwischen den Radien der Landmark-Iris und der Berechneten Ellipse in [Pixel] gegen den Radius-Größe des Filters.  
 Oben-Mitte: Gleam, Oben-Rechts: Gleam New,  
 Unten-Links: Luminance, Unten-Mitte: Min-Wert, Unten-Rechts: Max-Wert

## 6.1 Graphen

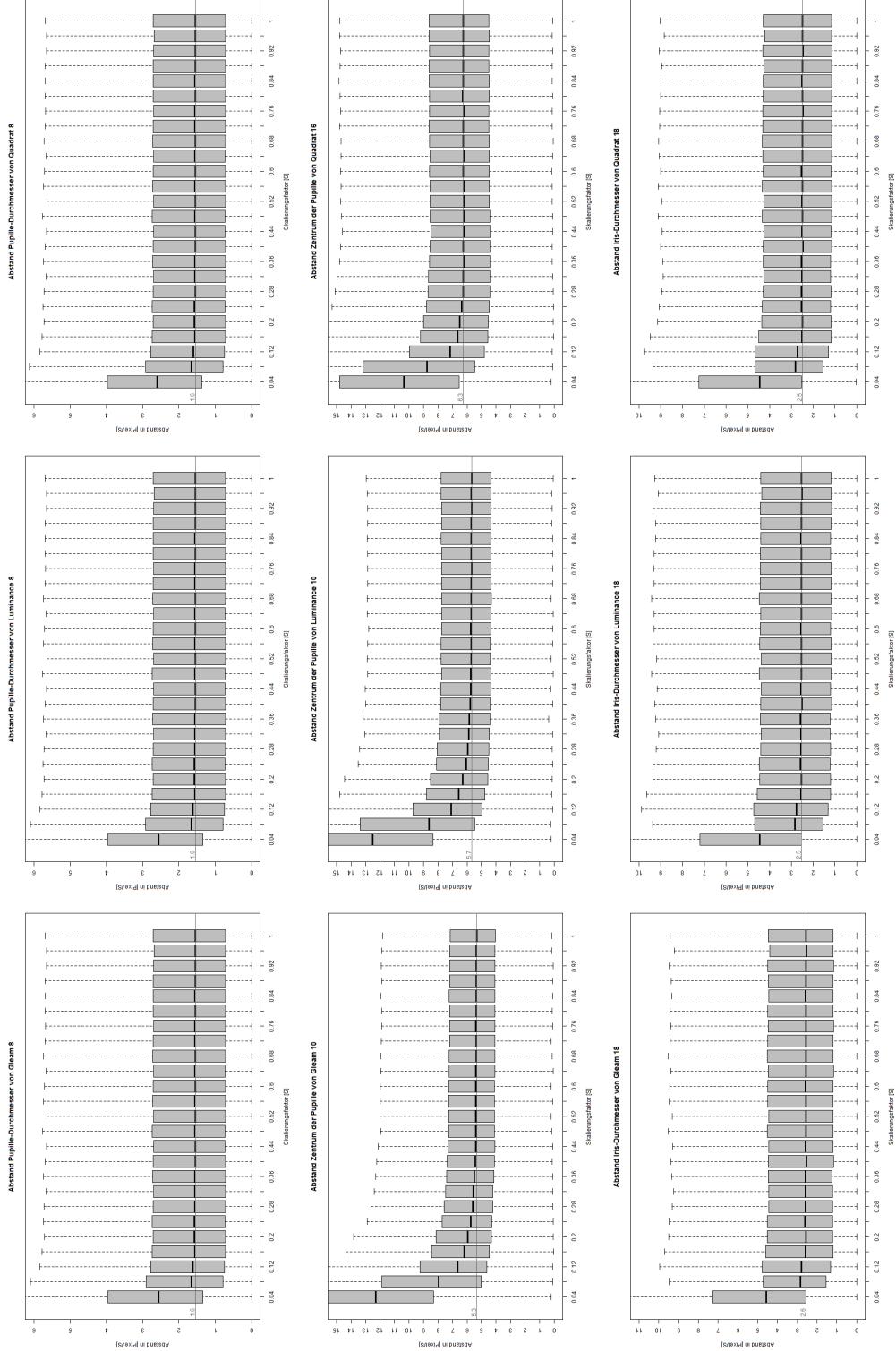


Abbildung 6.11: Auswirkung von der Bildgröße auf die Qualität der Berechnung. Aufgetragen ist die Abweichung [Pixel/Skalierung] gegen den Skalierungsfaktor. Oben: Pupille-Durchmesser, Mitte Abweichung Zentrum, Iris-Durchmesser  
Links: Gleam, Mitte: Luminance, Rechts Quadrat

## 6 Abbildungen

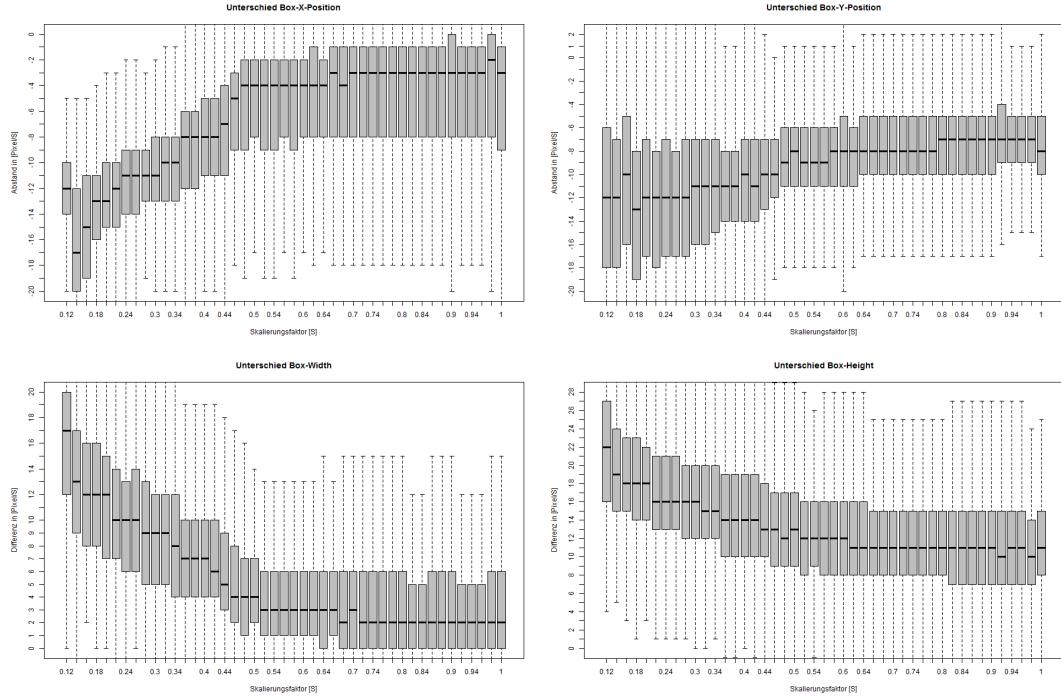


Abbildung 6.12: Bestimmung der Box ums Auge abhängig von der Bildgröße. Aufgetragen ist die Abweichung [Pixel/Skalierung] gegen den Skalierungsfaktor.

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.

## 6.1 Graphen

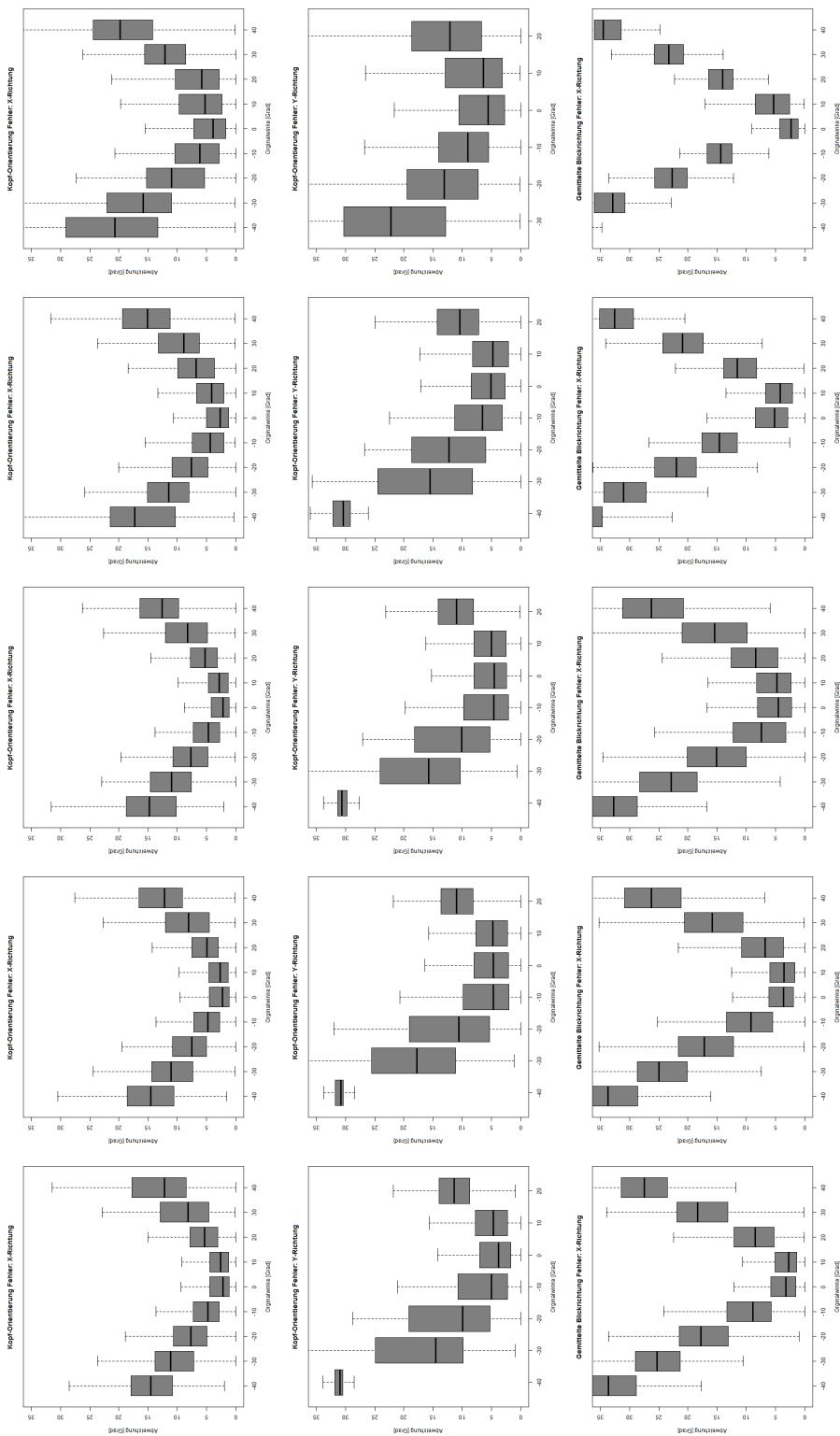


Abbildung 6.13: Abweichung der Videoaufnahme von der Kopfausrichtung Horizontal (Oben), Kopforientierung Vertikal (Mitte) und die X-Ausrichtung der Augen (Unten)  
Skalierungsfaktor von links nach rechts ( $1/0.5/0.1/0.05$ ), Y-Achse:  $[0 - 35]^\circ$

## 6 Abbildungen

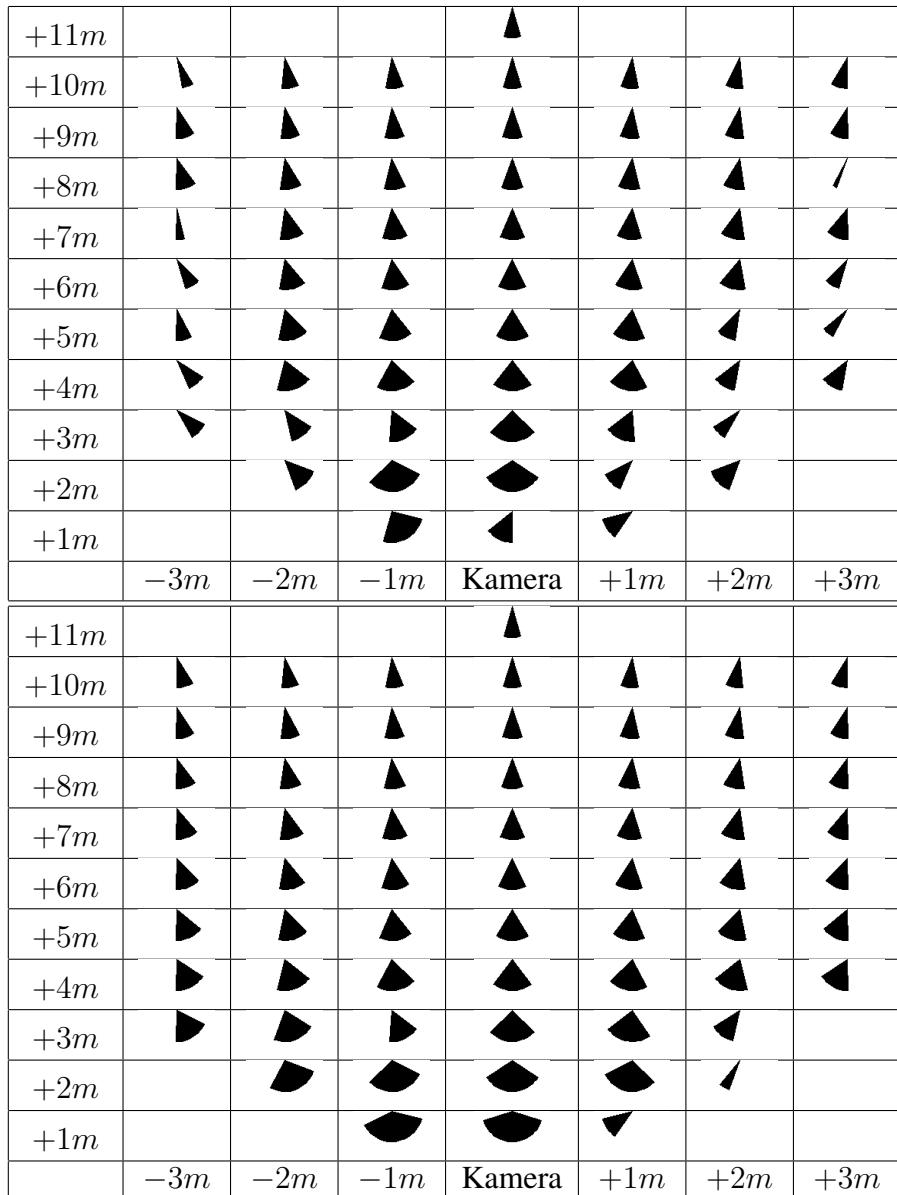


Abbildung 6.14: Dargestellt ist der horizontale Winkelbereich in denen ein Gesicht mit aufbereitetem Inhalt erkannt wurde.  
 Links: Einzelbilder, Rechts: Video

## 6.1 Graphen

	$+12m$	$+11m$	$+10m$	$+9m$	$+8m$	$+7m$	$+6m$	$+5m$	$+4m$	$+3m$	$+2m$	$+1m$	$-3m$	$-2m$	$-1m$	Kamera	$+1m$	$+2m$	$+3m$
																Kamera	$+1m$	$+2m$	$+3m$
																Kamera	$+1m$	$+2m$	$+3m$
																Kamera	$+1m$	$+2m$	$+3m$
																Kamera	$+1m$	$+2m$	$+3m$
																Kamera	$+1m$	$+2m$	$+3m$

Abbildung 6.15: Dies sind die Augenbereiche der Probanden aus Versuch 3

# 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, and Louis Philippe Morency: *3d Constrained Local Model for Rigid and Non-Rigid Facial Tracking*. In *Computer Vision and Pattern Recognition (CVPR 2012)*, Providence, RI, June 2012. <http://ict.usc.edu/pubs/3D%20Constrained%20Local%20Model%20for%20Rigid%20and%20Non-Rigid%20Facial%20Tracking.pdf>.
- [3] Bradski, Gary and 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](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 Consstrained 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.

## Literaturverzeichnis

- [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>.
- [11] G.L. Masala, E. Grossi: *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] HSV: *Maxi Beister als Herr Müller überrascht eine Schulkasse*. <https://www.youtube.com/watch?v=WqK-6ienapo>, [Online; abgerufen am 14. Juli 2017].
- [15] Huang, Gary B., Marwan Mattar, Honglak Lee und Erik Learned-Miller: *Learning to Align from Scratch*. In: *NIPS*, 2012.
- [16] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li Yu Qiao: *Joint Face Detection and Alignment using Multi-task Cascaded Convolutional Networks*, 2015.
- [17] Kinnebrock, Werner: *Neuronale Netze: Grundlagen, Anwendungen, Beispiele*. Oldenbourg, 1994, ISBN 9783486229479.
- [18] 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](http://www.schulentwicklung-net.de/images/stories/Anlagen/510%20schulhausbau_BW_2013.pdf).
- [19] Kybic, Jan: *Point Distribution Models*, 2007. <http://cmp.felk.cvut.cz/cmp/courses/33DZOzima2007/slidy/pointdistributionmodels.pdf>.
- [20] 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>.
- [21] Neubeck, Alexander und Luc Van Gool: *Efficient Non-Maximum Suppression*. In: *Proceedings of the 18th International Conference on Pattern Recognition - Volume 1*, 2006.

## Literaturverzeichnis

- me 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>.
- [22] Peemen, Maurice. <http://parse.ele.tue.nl/mpeemen>.
  - [23] Stepanov, Vitalij: *Analyse komplexer Szenen mit Hilfe von Convolutional Neural Networks*, 2012.
  - [24] 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/>.
  - [25] Tadas Baltrušaitis, Peter Robinson, Louis Philippe Morency: *Constrained Local Neural Fields for robust facial landmark detection in the wild*, 2013.
  - [26] Tadas Baltrušaitis, Peter Robinson, Louis Philippe Morency: *OpenFace: an open source facial behavior analysis toolkit*, 2016.
  - [27] Wikibooks: *GNU R: boxplot* — Wikibooks, Die freie Bibliothek, 2012. [https://de.wikibooks.org/w/index.php?title=GNU\\_R:\\_boxplot&oldid=641628](https://de.wikibooks.org/w/index.php?title=GNU_R:_boxplot&oldid=641628), [Online; abgerufen am 10. Juli 2017].
  - [28] Wikipedia: *Active Appearance Model* — Wikipedia, Die freie Enzyklopädie, 2014. [https://de.wikipedia.org/w/index.php?title=Active\\_Appearance\\_Model&oldid=135641554](https://de.wikipedia.org/w/index.php?title=Active_Appearance_Model&oldid=135641554), [Online; Stand 16. Juni 2017].
  - [29] Wikipedia: *Bicubic interpolation* — Wikipedia, The Free Encyclopedia, 2016. [https://en.wikipedia.org/w/index.php?title=Bicubic\\_interpolation&oldid=751879378](https://en.wikipedia.org/w/index.php?title=Bicubic_interpolation&oldid=751879378), [Online; accessed 6-May-2017].
  - [30] 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].
  - [31] 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].
  - [32] Wikipedia: *Augenbewegung* — Wikipedia, Die freie Enzyklopädie, 2017. <https://de.wikipedia.org/w/index.php?title=Augenbewegung&oldid=166073779>, [Online; Stand 13. Juni 2017].

- [33] Wikipedia: *Convolutional Neural Network — Wikipedia, Die freie Enzyklopädie*, 2017. [https://de.wikipedia.org/w/index.php?title=Convolutional\\_Neural\\_Network&oldid=166523646](https://de.wikipedia.org/w/index.php?title=Convolutional_Neural_Network&oldid=166523646), [Online; Stand 29. Juni 2017].
- [34] Wikipedia: *OpenCV — Wikipedia, Die freie Enzyklopädie*, 2017. <https://de.wikipedia.org/w/index.php?title=OpenCV&oldid=166087629>, [Online; Stand 16. Juni 2017].
- [35] Wikipedia: *Point distribution model — Wikipedia, The Free Encyclopedia*, 2017. [https://en.wikipedia.org/w/index.php?title=Point\\_distribution\\_model&oldid=759054014](https://en.wikipedia.org/w/index.php?title=Point_distribution_model&oldid=759054014), [Online; accessed 9-May-2017].
- [36] Wissensmedien (IWM), Leibniz Institut für: *Tübingen Digital Teaching Lab (TüDiLab)*. <https://www.tuedilab-tuebingen.de/>.
- [37] 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>.
- [38] 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.
- [39] Xucong Zhang, Yusuke Sugano, Mario Fritz Andreas Bulling: *Appearance-Based Gaze Estimation in the Wild*, 2015.
- [40] 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