Interaktionstechniken für mobile Augmented-Reality-Anwendungen basierend auf Blick- und Handbewegungen

zur Erlangung des akademischen Grades eines

Doktors der Ingenieurwissenschaften

von der KIT-Fakultät für Informatik
des Karlsruher Instituts für Technologie (KIT)

genehmigte

Dissertation

von

Dipl.-Inform.

Jan Hendrik Hammer

aus Lippstadt

Tag der mündlichen Prüfung:	07.12.2020
Erster Gutachter:	Prof. Dr.-Ing. habil. Jürgen Beyerer
Zweiter Gutachter:	Prof. Dr.-Ing. Michael Beigl

Abstract

Visual Augmented Reality has the potential to fundamentally change the way how humans interact with machines. Basic prerequisite therefore are comfortable binocular AR glasses with a large field of view for visual high contrast fade-ins of virtual elements, so that those are perceived as part of the real environment. Additionally, such AR systems need an intuitive interaction to be accepted by the user. Besides language, gaze and hand gestures form the interaction techniques of choice to enable interaction with virtual elements. The herein presented thesis discusses gaze analysis allowing for an implicit unconscious interaction. Furthermore it examines capture of hand movements for explicit interaction in mobile applications. One of the first methods for fully automatic real-time capable gaze analysis in three-dimensional environments using an example from the museum context is presented. Therefore, a 3D viewpoint calculation and a real-time gaze analysis of 3D gaze paths were realized, as this was impossible using any eye tracker including its associated software. Additionally, the Projected Gaussian method for the representation of three-dimensional gaze behavior is introduced, which allows for the realistic visualization of gaze paths in three-dimensional environments as heat maps in real time. This worldwide unique process projects the visual acuity of the human gaze into the scene and thus remains close to the physical process of perception. None of before presented procedures took account of occlusions or enabled a coloring of surfaces independently of their polygon structure. Both, the method for fully automatic gaze analysis as well as Projected Gaussians, are demonstrated with an example on real gaze data and hereby gained results are presented. For the explicit interaction with hands the focus of this work lies on the first step of hand gesture recognition in monocular color images: the determination of the hand region, meaning the detection of a bounding box containing the hand in an image. The developed

methods merge optical flow and segmentations of skin color in different ways. Furthermore, those methods use object classifiers and hand pose estimators for an optimized hand region determination. The latter is then fused with a publicly available 2D hand pose estimator. This fusion surpasses the current state of the art in 2D and 3D monocular hand pose estimation on the public data set EgoDexter regarding the evaluation of 2D hand pose estimations in case of small permitted deviations. The obtained results show a deficit in the use of previous hand pose estimates by current 3D hand pose estimators for monocular input images. The shown procedure for hand region determination allows combinations with any hand pose estimator.

Kurzfassung

Visuelle Augmented Reality hat das Potential, die Art und Weise, wie der Mensch mit Maschinen kommuniziert, grundlegend zu verändern. Grundvoraussetzung dafür sind angenehm zu tragende binokulare AR-Brillen mit einem großen Sichtfeld für visuelle Einblendungen mit hohem Kontrast, so dass virtuelle Elemente als Teil der realen Umgebung dargestellt und wahrgenommen werden können. Gleichzeitig bedürfen derartige AR-Systeme einer intuitiven Interaktion mit ihrem Benutzer, um akzeptiert zu werden. Blick und Handgesten bilden neben Sprache die Interaktionstechniken der Wahl, um mit virtuellen Elementen zu interagieren. Die vorliegende Arbeit beschäftigt sich mit der Analyse des Blickes für eine implizite unbewusste Interaktion und mit der Erfassung von Handgesten für die explizite Interaktion in mobilen Anwendungen. Es wird eines der ersten Verfahren zur vollautomatischen echtzeitfähigen Blickbewegungsanalyse in dreidimensionalen Umgebungen anhand eines Beispiels aus dem Museumskontext vorgestellt. Dafür wurde eine 3D-Blickpunktberechnung und eine darauf aufsetzende echtzeitfähige Blickanalyse von 3D-Blickpfaden realisiert, als dies mit anderen Blickmessgeräten inklusive zugehöriger Software nicht möglich war. Zusätzlich wird das Verfahren Projected Gaussians für die Darstellung dreidimensionalen Blickverhaltens vorgestellt, das in Echtzeit realistische Visualisierung von Heatmaps in dreidimensionalen Umgebungen erzeugt. Dieses Verfahren ist das weltweit einzige, das die visuelle Schärfe des menschlichen Blickes in die Szene projiziert und damit nah am physikalischen Prozess der Wahrnehmung bleibt. Kein zuvor vorgestelltes Verfahren berücksichtigte Verdeckungen oder ermöglichte eine von der Polygonstruktur unabhängige Einfärbung von Oberflächen. Sowohl das Verfahren zur vollautomatischen Blickanalyse als auch

Projected Gaussians wird anhand eines Beispiels auf echte Blickdaten angewendet und die Ergebnisse dieser Analyse werden präsentiert. Für die explizite Interaktion mit den Händen beschäftigt sich diese Arbeit mit dem ersten Schritt der Handgestenerkennung in monokularen Farbbildern: der Handregionsbestimmung. Bei dieser wird die Region der Hand in einem Kamerabild ermittelt. Die entwickelten Verfahren fusionieren auf unterschiedliche Weise optischen Fluss und Segmentierungen von Hautfarbe. Des Weiteren nutzen sie Objektklassifikatoren und Handposenschätzer für eine optimierte Handregionsbestimmung. Letztere wird anschließend mit einem öffentlich verfügbaren 2D-Handposenschätzer fusioniert. Diese Fusion übertrifft bei der 2D-Posenschätzung und geringen erlaubten Abweichungen auf dem öffentlichen Datensatz EgoDexter den aktuellen Stand der Technik der Handposenschätzung, obwohl zugehörige Verfahren trotz monokularen Eingabedaten ihre Schätzungen im dreidimensionalen Raum durchführen. Die Ergebnisse zeigen bei aktuellen 3D-Handposenschätzern für monokulare Eingabebilder ein Defizit bei der Wiederverwendung vorheriger Handposenschätzungen. Das hier vorgestellte Verfahren zur Handregionsbestimmung kann mit jedem Handposenschätzer kombiniert werden.

Danksagung

In erster Linie möchte ich mich bei Prof. Dr.-Ing. habil. Jürgen Beyerer für die Betreuung dieser Arbeit bedanken. Er stand immer mit Rat zur Seite und war für jedwede Diskussion offen. Des Weiteren möchte ich mich bei ihm für die hervorstechende Qualität seiner Lehre am Karlsruher Institut für Technologie (KIT) bedanken. Seine Vorlesung „Automatische Sichtprüfung und Bildverarbeitung“ war vermutlich die lehrreichste und wichtigste in meinem Studium und hat mich zur Bildverarbeitung geführt. Bei Prof. Dr.-Ing. Michael Beigl möchte ich mich für die unkomplizierte Übernahme des Korreferats sowie seine Ratschläge bedanken. Zusätzlich möchte ich mich bei Dr. Elisabeth Peinsipp-Byma, Dr.-Ing. Michael Voit, Dr.-Ing. Florian van de Camp, Manuel Martin, Jutta Hild und Vitali Henne für die vielen Diskussionen und Gespräche zu Themen der vorliegenden Arbeit als auch zur alltäglichen Projektarbeit bedanken. Weiterer Dank geht an meine ehemaligen Kollegen Dr.-Ing. Chengchao Qu und Klaus Jäger sowie meine ehemaligen Studenten Christian Lengenfelder, Michael Maurus, Carlos Garcia, Gerhard Kurz, Daniel Reichert, Daniel Secker, Matthias Horne, Stanislav Arnaudov, Ulrich Konrad, Leonard Graf, Andreas Bauer, Karin Schmidt und Daniel Tiefert für ihre Begeisterung für die unterschiedlichen Themen, die sie bearbeitet haben. Ein großer Dank geht an meine Eltern für ihre Unterstützung. Mein größter Dank geht an meine Frau Linda, die mir den Rücken nicht nur für die Erstellung dieser Arbeit freigehalten hat.

Inhaltsverzeichnis

1 Einleitung

1.1 Motivation

Erweiterte Realität (engl. Augmented Reality (AR)) beinhaltet die visuelle Erweiterung der Realität mit nicht realen sog. virtuellen über den Sehsinn wahrnehmbaren Elementen. Virtuelle und reale Welt können verschmelzen. Der wachsende Einfluss der erweiterten Realität auf unser alltägliches Leben ist bereits in einzelnen Anwendungen, z. B. beim Autofahren, erfahrbar. Sogenannte Head-Up-Displays (HUDs) visualisieren Informationen wie die aktuelle Geschwindigkeit, Geschwindigkeitslimit und Navigationshinweise direkt im Sichtfeld des Fahrers. Der Fahrer kann die Informationen mit kleineren Blickbewegungen erfassen, wodurch die Straße zentrierter im Sichtfeld bleibt, als wenn er auf die Mittelkonsole schauen muss, was von einer zusätzlichen Kopfbewegung begleitet, die vorausfahrenden Autos an den Rand des peripheren Sichtfelds wandern lässt. Durch AR kann der Mensch Informationen einer Maschine schneller erfassen, dadurch wird sein Situationsbewusstsein gefördert. Nach Endsley [End99] gliedert sich selbiges in drei Ebenen: Die erste Ebene ist die *Wahrnehmung der Umwelt*. Die zweite Ebene beinhaltet das *Erfassen der Situation* durch das Zusammenführen aller Informationen aus der ersten Ebene und deren Interpretation. Diese Interpretation besteht aus der gedanklichen Übertragung der gesehenen Informationen auf die Umwelt bzw. das gedankliche Abbild der Umwelt. Die Übertragung kann mit einem zeitaufwendigen und fehleranfälligen Kontextwechsel einhergehen, wenn die Informationen z. B. nicht ortsreferenziert dargestellt werden. Die dritte Ebene des Situationsbewusstseins beschreibt das *Propagieren* der aktuellen Situation

in die Zukunft für die Entscheidungsfindung. Dieses Ebenenmodell des Situationsbewusstseins ist ursprünglich auf Insassen eines Fluggerätes zugeschnitten worden, lässt sich allerdings ebenso auf den Fahrer eines am Boden befindlichen Fortbewegungsmittels als auch den Nutzer einer AR-Anwendung übertragen.

Die technologische Herausforderung für realisierbare AR begann mit der Entwicklung optischer Systeme, charakterisiert durch die Fähigkeit, visuelle Elemente im Sichtfeld des Nutzers einblenden zu können. Wie so oft in der Menschheitsgeschichte fand eine derart hoch technologische und damit kostspielige Entwicklung im militärischen Kontext ihren Anfang [Ras09a]. Nach Whitehouse et al. [WHI17] waren kopfgetragene transparente AR-Systeme in Fluggeräten im Einsatz, bevor der Begriff AR geprägt wurde. Die entwickelten optischen Systeme befinden sich derweil in der Kommerzialisierung mit für jedermann erhältlichen Resultaten in Form von HUDs im Automobil als auch sog. AR-Brillen, wobei letztere derzeit (Februar 2019) einen prototypischen Charakter aufweisen. Ihre Verbesserung für den alltäglichen Einsatz findet entweder in Entwicklungsabteilungen großer Firmen [Les16, Mei14, Chi17] oder in Unternehmen mit enormer finanzieller Unterstützung großer Investoren statt [Mer14]. Hunderte Millionen Dollar [Por17b] werden in die Entwicklung von AR-Brillen aus folgendem Grund investiert: AR steigert die Effizienz der Mensch-Maschine-Interaktion durch eine Verbesserung des Situationsbewusstseins und mit AR-Brillen lässt sich AR in vielen alltäglichen Anwendungen umsetzen, die Mensch-Maschine-Interaktion beinhalten. Somit lassen sich Zeit und Kosten bei unterschiedlichsten Anwendungen einsparen.

Für ein interaktives AR-System wird zusätzlich zur visuellen Erweiterung, der *Informationsanreicherung*, eine intuitive *Interaktion* benötigt. Hierfür bieten sich besonders die Hände als Manipulationswerkzeug, aber auch der menschliche Blick an. AR-Anwendungen benötigen folglich für die Informationsanreicherung und die Interaktion *AR-Techniken*, welche in der vorliegenden Arbeit mit Fokus auf die Interaktion, basierend auf Blickbewegungsanalyse und Handgestenerkennung, betrachtet werden.

1.2 Was ist Augmented Reality?

Die Antwort auf diese Frage liefert folgende eigene Definition:

Definition 1.1. *Augmented Reality (AR) bezeichnet die Einflussnahme auf die Realitätswahrnehmung des Menschen durch nicht in der Wirklichkeit vorhandene Elemente, welche auf einen Sinn wirken. In der vorliegenden Arbeit wird unter Augmented Reality speziell die optische Einflussnahme auf den Sehsinn verstanden.*

Bei dieser *visuellen* Augmented Reality werden virtuelle Elemente durch ein Anzeigesystem über den menschlichen Sehapparat wahrgenommen. Je nach Art der Visualisierung kann der Eindruck entstehen, dass sich diese Elemente in der Wirklichkeit befinden – virtuelle Elemente und echte Umgebung verschmelzen. Für die Darstellung existieren festinstallierte, mobile und kopfgetragene Anzeigesysteme, die für die Darstellung unterschiedliche Technologien und Anzeigemodi verwenden.

Für interaktive AR-Anwendungen ist es unabdingbar, zu wissen, was für eine Aktion der Nutzer zum jeweiligen Zeitpunkt durchführt. Liest er etwas? Sucht er etwas? Bedient er eine Nutzerschnittstelle? Um Antworten auf diese Fragen zu finden, ergeben sich erneut Fragen: Was schaut der Nutzer an? Wie ist sein Blickverhalten? Was macht der Nutzer mit seinen Händen? Zeigt er auf etwas? Bewegt er ein Objekt? Manipuliert er ein echtes oder virtuelles Element? Spricht er mit jemandem? Hier hilft die Erkennung *impliziter* und *expliziter* Interaktion, wobei sich die vorliegende Arbeit nicht mit Sprache als Kommunikationskanal beschäftigt. In den Bereich implizite Interaktion fällt die Blickanalyse, durch die Rückschlüsse auf die visuelle Aufmerksamkeit ermöglicht werden, welche vom AR-System als Eingabe für eine kontextbasierte Folgeaktion genutzt werden kann. Diese Art der Interaktion wird als implizit bezeichnet, weil der Nutzer sie unbewusst steuert. Bei der expliziten Interaktion hingegen führt der Nutzer eine bestimmte Aktion aus, um das System bewusst zu steuern. Diese Interaktion kann die Manipulation eines echten oder virtuellen Elements umfassen und wird meistens – getreu dem Ursprung des Wortes *Manipulation* – mit den Händen (lat. manus) durchgeführt. Den Blick

für die explizite Interaktion zu nutzen, ist im Bezug auf virtuelle Objekte nicht undenkbar.

1.2.1 Anwendungsfelder für die erweiterte Realität

Nachfolgend werden verschiedene Anwendungsfelder vorgestellt, in denen AR bereits genutzt oder für die AR einen deutlichen Vorteil haben wird, wenn sie ihren praktischen Einsatz erreicht. Laut einer Umfrage vom Harvard Business Review [Har17] liegen diese Anwendungsfelder im Bereich Maschinenbetrieb, -bedienung und -wartung durch AR-unterstützte Anleitungen, Inspektionen und (Fern-)Anweisungen. Im Bereich Produktion liegen die Hauptanwendungsfelder in der Qualitätssicherung und der manuellen Montage. Weitere Gebiete sind laut der Umfrage HUDs im Automobil, kollaborative Produktentwicklung oder auch medizinische Operationen. Als wichtigste Ziele werden eine höhere Qualität in der Produktion, kürzere Entwicklungszyklen oder eine bessere Benutzererfahrung genannt. Allen nachfolgend dargestellten Anwendungen liegt eine Erstellung von Inhalten zu Grunde, die für sich allein gesehen eine zeitaufwendige und komplexe Aufgabe ist, für die es Spezialisten benötigt.

1.2.1.1 Militär

Kampfflugzeuge waren zuerst mit HUDs ausgestattet, um die Bedienung im Cockpit zu verbessern. Mittlerweile verfügt der Helm des Piloten über die Fähigkeit, Elemente im Sichtfeld des Piloten zu visualisieren, egal in welche Richtung er schaut. Das Flugzeug muss folglich nicht mehr auf ein Objekt ausgerichtet werden, um das Sichtfeld visuell zu erweitern, sondern nur noch der Kopf. Hierdurch wird die Effizienz der Bedienung deutlich gesteigert. Auch wenn die Auswahl eines visuellen Elementes derweil durch die Kopfausrichtung realisiert wird, wäre es denkbar, diese Auswahl durch den Blick zu realisieren, da eine Ausrichtung der Augen auf ein Objekt weniger anstrengend ist als eine Kopfbewegung und der Blick der Kopfbewegung vorausgeht [Fre08], wird somit die schnellere Auswahl eines Objektes erlaubt.

1.2.1.2 Produktion

In der Produktion ist AR in unterschiedlichen Bereichen sinnvoll nutzbar: In der Produktion erzeugen Maschinen eine Unmenge von Daten, die überwachenden Mitarbeitern sinnvoll dargestellt werden müssen. In der manuellen Montage – besonders bei kleiner Losgröße – oder in der Qualitätssicherung sind unterschiedliche Arbeitsschritte durchzuführen, die den Mitarbeitern verständlich dargestellt werden müssen. Eine sinnvolle Informationsanreicherung erhöht das Verständnis und sorgt für eine effizientere Abarbeitung durch Einsparung von Zeit bei der Ausführung und erhöht die Vollständigkeit der Aufgabe. Montageschritte können dreidimensional und ortsreferenziert visualisiert werden, so dass der Mitarbeiter virtuell sieht, welche Einzelteile er wo findet, um sie dann in einer ganz bestimmten Weise zusammenzubauen. Dies ist besonders für das Anlernen neuer Mitarbeiter eine große Hilfe. Aber auch bei aufwendigen Prüfungen mit vielen Prüfpunkten oder seltenen Instandsetzungsarbeiten ist eine unterstützende Informationsanreicherung sinnvoll. So sind Zeiteinsparungen von 25 % bei Aufgaben in der Produktion keine Seltenheit [Por17c]. Für diese Aufgaben ist eine Interaktion mit dem AR-System notwendig, die z. B. handgestenbasiert [Por17a] realisiert werden kann.

1.2.1.3 Automobil

Wie oben bereits erwähnt, haben HUDs Einzug in das Automobil erhalten, um dem Fahrer wichtige Informationen mitzuteilen. Der Blick auf die Mittelkonsole entfällt. Eine Fokussierung des Armaturenbrettes ist für durch HUDs dargestellte Informationen ebenfalls nicht notwendig, weil die Informationen vor dem Auto auf der Straße liegend erscheinen. Das erneute Fokussieren des Geschehens auf der Straße entfällt. Die Aufmerksamkeit liegt folglich kontinuierlicher auf der Straße vor dem Auto, weil nicht so oft die Fokusebene gewechselt werden muss. Der Fahrer wird dadurch weniger beansprucht, was die Nutzererfahrung verbessert.

1.2.1.4 Medizin

Im medizinischen Umfeld kann Augmented Reality genutzt werden, um unterschiedlichste Informationen darzustellen. Es gibt Systeme, die die Venen von Patienten visuell darstellen, um Nadeln besser setzen zu können [Acc19]. Weiter wird an Systemen gearbeitet, die dabei helfen, Abläufe von Operationen zu visualisieren [Sur19], damit diese allen Beteiligten klar verständlich sind. Andere Systeme projizieren Markierungen in Echtzeit auf die Haut von Patienten, um eine präzisere Bestrahlung zu ermöglichen [Chy12]. Eine komplexe Aufgabe für Operateure ist häufig die Steuerung von Instrumenten bei minimal-invasiven Eingriffen. Die Position und Orientierung des Werkzeugs lassen sich dann nur über eine ebenfalls eingeführte Kamera erfassen. Um das Werkzeug zu steuern, müssen sowohl Position und Orientierung der Kamera als auch Position und Orientierung des Werkzeugs relativ zur Kamera berücksichtigt werden, was zwei Kontextwechseln entspricht. Ortsreferenzierte Darstellungen der Werkzeuge und des Situs über AR könnten dieses Problem vereinfachen, weil der Kontextwechsel entfällt. Im BMBF-Projekt *KonsensOP* [Kon15, Zie15, Här15] ist das Ziel ein intelligenter Operationsassistent, der ermittelt, in welcher Phase sich eine Operation befindet, um kontextsensitiv bei Anomalien behilflich sein zu können. Bei dieser Phasenermittlung werden neben Merkmalen, wie der Position der einzelnen Akteure des OP-Teams, Blick- und Handbewegungen des Operateurs als implizite Indikatoren genutzt. Weiter dienen positionsbasierte Handbewegungen dem Operateur zur Interaktion mit den dargestellten Informationen.

1.2.1.5 Kultur

Auch im kulturellen Bereich spielt Augmented Reality eine Rolle. Museen vermitteln per Smartphone oder Tablet Informationen über Exponate, die früher per Texttafel oder Audio-Guide übermittelt wurden. Andere Ansätze wie das EU-Projekt ARtSENSE [Dam12] gehen hier weiter: Durch das Tragen einer AR-Brille mit Blickbewegungsmessung kann die visuelle Aufmerksamkeit eines Besuchers ermittelt und die Informationen über ein Exponat, angepasst an sein visuelles Interesse, übermittelt werden. Ohne es zu bemerken, wird

durch die Blickbewegungsanalyse implizit der Inhalt individuell aufbereitet, mit dem zusätzlich explizit per Handgesten interagiert werden kann.

1.3 Beitrag der Arbeit

Aufgrund des oben ermittelten Bedarfs an Interaktionstechniken beschäftigt sich die vorliegende Arbeit mit dem Thema Blickanalyse und Handbewegungserfassung für mobile kopfgetragene AR-Systeme.

Im Bereich der Blickanalyse in Kapitel 3 wird eines der ersten Verfahren zur vollautomatischen echtzeitfähigen Blickbewegungsanalyse in dreidimensionalen Umgebungen anhand eines Beispiels aus dem Museumskontext vorgestellt, siehe Abschnitt 3.1. Zusätzlich wird in Abschnitt 3.2 das erste echtzeitfähige Verfahren zur realistischen Visualisierung von Heatmaps in dreidimensionalen Umgebungen beschrieben.

Für die explizite Interaktion sind die Hände die Modalität der Wahl. Der erste Schritt der Handgestenerkennung ist die Handregionsbestimmung, bei der die Region der Hand in einem Kamerabild ermittelt wird. Im zugehörigen Teil der Arbeit, in Kapitel 4, werden unterschiedliche Verfahren zur monokularen Handregionsbestimmung beschrieben und auf einem dafür erstellten Datensatz evaluiert. In einer ausgeklügelten Kombination mit einem Verfahren zur 2D-Handposenerkennung auf Handregionsbildern entstand in dieser Arbeit ein Handposenschätzer, der den Stand der Technik bzgl. Genauigkeit bei der 2D-Handposenerkennung auf dem bekannten öffentlichen Datensatz *EgoDexter* übertrifft.

Das direkt nachfolgende Kapitel 2 führt den Leser in die verwandten Arbeiten zur Interaktion in mobilen AR-Anwendungen basierend auf Blick- und Handbewegungen ein und gibt der Vollständigkeit halber zu Beginn einen Einblick in den Stand der Technik zur Informationsanreicherung.

2 Verwandte Arbeiten

In diesem Kapitel werden verwandte Arbeiten zur *Informationsanreicherung* und *Interaktion* bis zum Zeitpunkt Ende Januar 2019 beschrieben. Abschnitt 2.1 beschäftigt sich mit dem Stand der Technik zur Informationsanreicherung, um einen Einblick in die Technologien und Konzepte von Anzeigesystemen zu bekommen. In Abschnitt 2.2 werden die relevanten Themen der Blickanalyse beschrieben. Der Stand der Technik zur handgestenbasierten Interaktion wird in Abschnitt 2.3 behandelt.

2.1 Informationsanreicherung

Kontextbasierte Informationsanreicherung für unmittelbares Verständnis ist eine der Grundvoraussetzungen der erweiterten Realität für eine effizientere Bedienung von Systemen. Die Informationen müssen über nachfolgend *visuelle Elemente* genannte virtuelle Objekte übermittelt werden. Visuelle Elemente werden in Abschnitt 2.1.1 einer Kategorisierung unterzogen. Für die Anzeige von Informationen werden unterschiedliche Anzeigesysteme, wie festinstallierte Displays, festinstallierte HUDs, mobile Displays, kopfgetragene Systeme (engl. Head-Mounted-Displays (HMDs)) oder auch Projektoren genutzt. Zur Bewertung der Bildqualität von Anzeigesystemen sind die wichtigsten Bewertungskriterien: Auflösung, Kontrast und Verzerrung [Ras09a]. Bei kopfgetragenen AR-Systemen hängt je nach Anwendungen die Akzeptanz zusätzlich vom Gewicht des Gerätes selbst als auch von der Größe des Sichtfeldes ab. Die Abschnitte 2.1.2 bis 2.1.5 befassen sich mit den unterschiedlichen Anzeigesystemen und aktuellen Entwicklungen. Besonderes Augenmerk wird dabei in Abschnitt 2.1.3 auf kopfgetragene Systeme gelegt, da diese die Basis für mobile AR-Anwendungen sind.

2.1.1 Kategorien von visuellen Elementen

Informationen, wie z. B. die Geschwindigkeit oder Höhe über Normalnull eines Fortbewegungsmittels können Insassen als *ortsunabhängige* visuelle Elemente dargestellt werden. Diese können dementsprechend beliebig in der erweiterten Realität platziert werden. Informationen wie Wegpunkte, Navigationshilfen, Objekte oder Spaltmaße besitzen einen Ortsbezug und müssen *ortsreferenziert* dargestellt werden, um ihrer Bedeutung einen Sinn zu geben. Der Bezug ortsreferenzierter visueller Elemente können die Welt, der Nutzer selbst, sein Fortbewegungsmittel oder andere Objekte sein. Die Informationen werden durch anwendungsspezifische Symbolik dargestellt, die vom Nutzer erlernt werden muss, damit Informationen und Welt verschmelzen und die Aufmerksamkeit nicht abgelenkt wird [Ras09a]. Visuelle Elemente können das Situationsbewusstsein und damit die Reaktionsgeschwindigkeit mindern mit gefährlichen Konsequenzen für den Nutzer [Pri04]. Zusätzlich zur örtlichen spielt die zeitliche Art und Weise der Darstellung von visuellen Elementen eine große Rolle. Dies bezieht sich besonders auf das Einblenden von Warnungen und deren Priorisierung bei gleichzeitigem Auftreten. Eine Studie von Sun et al. [Sun15] beschreibt die verminderte Reaktionsgeschwindigkeit unter gleichzeitiger Durchführung zweier Aufgaben mit gleichem Fokus der visuellen Aufmerksamkeit. Dies unterstreicht die besondere Bedeutung eines wohldurchdachten örtlichen und zeitlichen Arrangements von visuellen Elementen.

2.1.2 Festinstallierte Anzeigesysteme

Werden Informationen auf einem im Cockpit eines Fortbewegungsmittels festinstallierten Anzeigesystem dargestellt, entspricht die Zeit, die der Nutzer zum Erfassen der Situation benötigt, dem Wenden des Blickes auf das Anzeigesystem, dem Wahrnehmen der visuellen Elemente und der Interpretation derselben. Die Interpretation kann durch den Einbezug einer relativ zur Umwelt berechneten Pose (Position und Orientierung) des Fortbewegungsmittels gefördert werden. Beispiele hierfür sind Parkassistenten mit Rundumblick, die Seitenansichten und verschiedene Draufsichten auf das

Auto bieten, oder die Fahrzeug-zentrierte Sicht bei Navigationsgeräten. Diese erleichtern den Kontextwechsel zwischen dargestellter Information auf z. B. einem Display und der Realität. Unabhängig davon ist für den Blick auf das Anzeigesystem eine zusätzliche Kopfbewegung notwendig. Dieser Nachteil sog. *Head-Down-Displays* führte zur Entwicklung von transparenten HUDs, welche in bemannten Flugobjekten für eine effizientere Wahrnehmung der Informationen sorgen und damit ein steigendes Situationsbewusstsein mit sich bringen [End99, Pri04]. Trotz dieses positiven Effekts wurden Unfälle im Bereich der zivilen Luftfahrt, bei denen die Aufmerksamkeit des Piloten durch Informationen im HUD von der Realität abgelenkt war, beobachtet [Pri04]. HUDs weisen zusätzlich den Nachteil auf, dass der durch sie mit Informationen erweiterbare Bereich ein kleines Sichtfeld aufweist und dieses an die Ausrichtung des Fortbewegungsmittels gekoppelt ist. Um die Wahrnehmung von Informationen weiter zu steigern, wurde die Entwicklung kopfgetragener Systeme vorangetrieben [End99].

2.1.3 Kopfgetragene Anzeigesysteme

Bei kopfgetragenen Anzeigesystemen wird insbesondere die Dauer verkürzt, um die Informationen erblicken zu können, da keine Kopfbewegung durchgeführt werden muss, um auf das Anzeigesystem zu schauen [The13]. Die Wahrnehmung der Informationen ist also schneller. Zusätzlich kann durch den Einbezug der Pose des kopfgetragenen Systems relativ zur Umwelt (Fortbewegungsmittel oder Umgebung) die Interpretation vieler Informationen durch ortsreferenzierte visuelle Elemente gesteigert werden. Die zweite Ebene des Situationsbewusstseins nach Endsley [End99] wird folglich erleichtert.

Rash et al. [Ras09a] definieren die Komponenten eines allgemeinen kopfgetragenen AR-Systems als:

- Plattform
- Bildquelle
- Optik
- Kopfposenschätzung

Die Plattform kann z. B. ein Helm, aber auch eine Brille sein. An ihr wird die Optik zur Informationsdarstellung angebracht. Die Bildquelle liefert die darzustellenden Informationen, die mit der Optik für den Nutzer sichtbar gemacht werden. Die Bildquelle kann ein Nachtsichtgerät am Flugobjekt sein, aber auch die RGB-Kamera, die in eine AR-Brille integriert ist. Die darzustellenden Informationen können die Rohdaten eines Sensors, aber auch Bildverarbeitungsresultate der Sensorrohdaten oder daraus abgeleitete Erkenntnisse sein, die als visuelle Elemente im Sichtfeld dargestellt werden und einer Kategorie aus Abschnitt 2.1.1 zugeordnet werden können. Die Bildquelle kann demnach eine Sensordatenauswertung beinhalten.

Da das, was Rash et al. als *Optik* bezeichnen, aus mehreren Optiken bzw. optischen Systemen bestehen kann, wird es im Folgenden zur Klarheit *Anzeigesystem* genannt. In Anzeigesystemen verwendete Optiken als auch unterschiedliche Anzeigemodi werden nachfolgend beschrieben.

Ein Anzeigesystem besteht aus mindestens einer Kombination aus bildgebendem Modul und Optik. Das bildgebende Modul strahlt das Bild der Bildquelle aus und ist in Form eines Displays oder Projektors umgesetzt. Die Optik unterscheidet zwischen undurchsichtiger und transparenter Anzeige visueller Elemente. Auf diese wird in den Abschnitten 2.1.4 und 2.1.5 genauer eingegangen.

Der Anzeigemodus des Anzeigesystems kopfgetragener Systeme lässt sich *monokular*, *biokular* und *binokular* realisieren. Bei monokularem Anzeigemodus sieht nur ein Auge die dargestellte Information und das Anzeigesystem besteht aus einer Kombination aus bildgebendem Modul und Optik. Wird beiden Augen der gleiche Inhalt aus gleicher Perspektive präsentiert, wird dies als biokular bezeichnet. Das Anzeigesystem hat folglich für jedes Auge eine Kombination aus bildgebendem Teil und Optik. Durch binokulare Präsentation wird beiden Augen eine unterschiedliche Perspektive dargestellt, so dass perspektivische Wahrnehmung möglich wird [Ras09a]. Aktuelle Systeme für militärische Luftfahrzeuge besitzen Anzeigesysteme, bei denen das Sichtfeld häufig aus einem zentralen überlappenden Bereich mit binokularer Wahrnehmung und zwei monokularen Bereichen an den Rändern des Sichtfeldes besteht. Hierdurch wird das Sichtfeld erweitert und ähnelt dem des Menschen.

Nachfolgend wird ausschließlich von monokularen und binokularen Anzeigesystemen gesprochen. Denn monokulare Systeme implizieren einen einäugigen Aufbau und können nur monokular Informationen präsentieren. Binokulare Systeme implizieren einen zweiäugigen Aufbau und besitzen sowohl die Fähigkeit monokular, biokular als auch binokular Informationen darzustellen. Zu den drei wichtigsten Kriterien für die Bildqualität bei Anzeigesystemen, Auflösung, Kontrast und Verzerrung, gehört bei kopfgetragenen Geräten zusätzlich die Ausdehnung des Sichtfeldes (Field of View, FoV). Nach Zuckerman [Zuc54] ist das Sichtfeld eines menschlichen Auges oval, vertikal 120° und horizontal 150° ausgedehnt. Für beide Augen zusammen beträgt die Ausdehnung des Sichtfeldes horizontal 200°. Der binokular überlappende Bereich liegt bei ca. 90°. Nach Cuervo entsteht ein immersiver Eindruck, wenn das Sichtfeld horizontal mindestens 80° bemisst [Cue17].

2.1.4 Video see-through Augmented Reality und Virtual Reality

Realisiert ein Anzeigesystem eine intransparente Anzeige, kann der Nutzer im durch das Anzeigesystem überdeckten Bereich seines Sichtfeldes ausschließlich sehen, was die Bildquelle liefert. In Kombination mit einer die aktuelle Umwelt des Nutzers darstellenden Bildquelle wird dies als *Video see-through* AR bezeichnet. Die Umwelt wird indirekt über ein Anzeigesystem betrachtet. Hier existieren u.a. Anwendungen für nicht kopfgetragene monokulare Anzeigesysteme wie Smartphones und Tablets, um die reale Sicht mit Informationen anzureichern. Die größte Herausforderung für derartige Systeme ist die korrekte Darstellung ortsreferenzierter visueller Elemente, siehe Abschnitt 2.1.1, wofür die aktuelle Pose relativ zur Umgebung korrekt geschätzt werden muss.

Eine das gesamte Sichtfeld des Menschen vereinnahmende und kopfgetragene intransparente Anzeige fällt in den Bereich der Virtual Reality (VR). Die Optik einer VR-Brille sorgt dafür, dass der Nutzer die Informationen, die auf dem sehr nah vor den Augen getragenen Display dargestellt werden, dreidimensional wahrnehmen kann. Der Nutzer kann mit VR-Brillen virtuell komplett

in echte oder fiktive Welten eintauchen. Stellt die Bildquelle eines VR-Systems die Umwelt des Nutzers dar, nennt man dies ebenfalls Video see-through AR. Große Herausforderungen sind hier die unverzögerte Darstellung der Umgebung in der virtuellen Welt, wie auch die Posenschätzung des HMD zur Darstellung aus der Egoperspektive. Zusätzlich gehören zu den Herausforderungen die Visualisierung des Körpers des Nutzers in der virtuellen Welt und die Interaktion mit virtuellen Objekten.

Aufgrund der derzeitigen Limitierung optisch transparenter Anzeigesysteme bzgl. der Größe des Sichtfeldes lassen sich VR-Brillen, als Video see-through System realisiert, mit ihren mehr als 100° bemessenen Sichtfeldern nutzen, um AR-Brillen mit großem Sichtfeld zu simulieren. So ist es möglich prototypische AR-Anwendungen umzusetzen, deren Informationsanreicherung große, das gesamte Sichtfeld überdeckende visuelle Elemente nutzt. Ein Beispiel hierfür ist das vom Autor der vorliegenden Arbeit behandelte Thema transparenter Fahrerhäuser und Cockpits [Rei17, Sec18].

2.1.5 Optical see-through Augmented Reality

Transparente Anzeigesysteme werden als *optical see-through* bezeichnet. Bei optical see-through Augmented Reality wird die Welt folglich direkt betrachtet, ohne dass ein intransparentes Anzeigesystem die Sicht verdeckt. Um eine transparente Anzeige zu realisieren sind komplexe Arrangements von optischen Elementen notwendig. Ein Überblick über solche optischen Systeme bestehend aus Linsen, (halbtransparenten) Spiegeln und Prismen findet sich in Cakmakci und Rolland [Cak06]. Ausprägungen dieser Systeme finden sich z. B. im *iStar*-Prototyp [Bau12], in der *Google Glass* [Ols13] oder in diversen Helmsystemen für den militärischen Einsatz [Li13b, Cam15]. Diesen Optiken überlegen bzgl. Größe des Sichtfeldes, Größe des Augenbereichs (Bereich, in dem das Auge platziert sein muss, damit die visuellen Elemente erfasst werden können), Ausmaßen der optischen Elemente, Leistungsverbrauch und Darstellungsfähigkeiten sind Optiken, welche holographische optische Elemente nutzen. Letztere bestehen aus Lichtwellenleitern und diffraktiven optischen

Elementen, um ein kleines Eingabebild, welches vom bildgebenden Modul erzeugt wird, derart umzuleiten und zu vergrößern, dass eine große Austrittspupille entsteht [Muk09, Cam12, Li13b, Kre13]. Ausprägungen finden sich für den militärischen Bereich in Form von Helmsystemen [Cam15] oder für den kommerziellen Bereich in Form der *HoloLens* [Kre17] oder der *Magic Leap One* [Yeo17].

Die reale Umgebung wird bei optical see-through Augmented Reality direkt betrachtet und um visuelle Elemente erweitert. Durch kopfgetragene Systeme mit binokularem transparenten Anzeigesystem wird eine stereoskopische Darstellung von visuellen Elementen möglich, wodurch ortsbezogene Informationen an Ort und Stelle in der Realität dargestellt werden können, wenn die Pose des Anzeigesystems relativ zur Umgebung bekannt ist. Der große Vorteil dieser Verschmelzung ist der Wegfall eines möglicherweise fehlerbehafteten Kontextwechsels, also des gedanklichen Überführungsprozesses der durch die visuellen Elemente übertragenen Informationen auf die reale Umgebung. Dies resultiert in einer besseren Interpretation ortsbezogener Informationen und führt zu einer höheren Leistungsfähigkeit [Ras09a, Har05, Man07]. Dadurch, dass dieses Anzeigekonzept aufgabenspezifische Informationen direkt im Blickfeld darstellen kann, entfallen zusätzliche Kopfbewegungen in Richtung festinstallierter Anzeigesysteme, deren Anzahl je nach Anwendung immens reduziert werden kann oder sie ganz obsolet macht.

2.1.6 Posenschätzung

Für die stereoskopische Darstellung ortsbezogener visueller Elemente, sowohl in AR als auch VR, wird die Pose des Anzeigesystems benötigt. Die Schätzung dieser Pose muss robust und schnell sein, weil das Bild erst nach ihrer Berechnung für die durch sie definierte Perspektive gezeichnet und dargestellt werden kann. Um Unwohlsein und Übelkeit zu vermeiden, werden Latenzen von unter 20 Millisekunden (ms) angestrebt [Elb18].

In Cockpits von Fluggeräten werden für das Tracking des Helmes unterschiedliche Methoden verwendet. Ein Überblick über prinzipielle Herangehensweisen wurde von Ferrin erstellt [Fer91]. Er unterscheidet zwischen auf Ultraschall basierenden, magnetischen und optischen Technologien.

Verfahren, die auf Ultraschall basieren, um die Helmpose zu bestimmen, haben mehrere Ultraschall-Sender am Helm. Das Empfangsmodul besteht aus einzelnen Empfängern, um die Distanz zu jedem Sender ermitteln zu können. Diese Verfahren sind anfällig gegenüber Luftströmen und Turbulenzen sowie anderen Ultraschallquellen und Reflexionen im Cockpit.

Bei magnetischen Tracking-Verfahren wird ein Magnetfeld durch einen im Cockpit angebrachten Sender erzeugt. Der Empfänger wird am Helm angebracht. Die erreichbare Genauigkeit des Trackings nimmt mit größer werdendem Abstand zum Sensor rapide ab, der resultierende Bereich, in dem die Helmpose bestimmt werden kann, wird von Foxlin et al. [Fox04] mit einem Radius von 30 cm um den Sender beziffert. Des Weiteren wird das Magnetfeld von metallischen Objekten verzerrt. Diese Störungen können zwar gemessen und kompensiert werden, der Prozess hierfür ist aber zeitaufwendig und muss nach jeder Änderung der Sitzposition wiederholt werden.

Optische Posenschätzung ist auf verschiedene Weisen realisierbar. Gemein haben alle optischen Verfahren für die Posenschätzung, dass sie lichtempfindliche Sensoren (Kameras oder einzelne Fotorezeptoren) und lichtausstrahlende Sender benutzen. Die Sender müssen das Licht nicht selbst erzeugen, sondern können das Umgebungslicht reflektieren, das allerdings künstlich durch eine zusätzliche Lichtquelle erzeugt werden kann. Meistens handelt es sich hierbei um eingebrachtes Infrarot-Licht, da dieses für den Menschen nicht sichtbar ist. Sender können künstlich eingebrachte optische Referenzmarkierungen wie z. B. retroreflektierende Kugeln, Leuchtdioden (engl. Light-Emitting Diodes (LEDs)), aber auch Marken in Form binärer Muster oder auch die Umgebung selbst sein.

2.1.6.1 Optische Posenschätzung mit Referenzmarkierungen

Optisches Tracking mit Infrarot-LEDs wird bei der Helmposenschätzung genutzt. Einige Systeme nutzen das Tracking von am Helm angebrachten Infrarot-LEDs durch eine im Cockpit montierte Sensoreinheit [Cam13a]. Dies wird auch als *outside-in* Tracking bezeichnet, weil die Empfänger (die Kameras) in der Umgebung (Cockpit) und die Sender (LEDs) am kopfgetragenen System angebracht sind. Sind andersherum die Sensoren am Objekt, dessen Pose es zu schätzen gilt, befestigt und die Sender in der Umgebung angebracht, wird dies als *inside-out* Tracking bezeichnet. Optische Systeme haben generell den Nachteil, dass sie von anderen Infrarotquellen wie dem Sonnenlicht oder Reflexionen im Cockpit gestört werden können, sowie eine unverdeckte Sicht der Kameras auf die LEDs oder optischen Referenzmarkierungen benötigen.

Ein von Larsson und Blomqvist beschriebener Ansatz [Lar08] nutzt im Cockpit angebrachte Kameras, die mit 180 Hz Bilder erfassen. Am Helm sind insgesamt 16 LED-Cluster mit jeweils vier LEDs angebracht. Um die Helmpose bei Kopfdrehungen und Bewegungen im Cockpit schätzen zu können, werden drei bildgebende Sensoren verwendet. Zwei Sensoren schauen von links und rechts schräg hinter dem Sitz und einer von vorne aus dem Cockpit auf den Helm. Zur Berechnung der Helmpose wird zu jedem Zeitpunkt nur ein Sensor und ein LED-Cluster, also 4 LEDs verwendet.

Durch die Entwicklung mikroelektronischer mechanischer Systeme (MEMS) wurden miniaturisierte inertiale Messeinheiten (IMUs) realisiert. Diese Sensoren weisen im Gegensatz zu akustischen, magnetischen oder optischen Sensoren keine Reichweitenbegrenzung auf und es gibt kein Risiko für Störsignale, die das Nutzsignal überlagern. Zusätzlich können sie mit hoher Rate ausgelesen werden. Ein Problem ist allerdings, dass eine ausschließlich auf ihren Messwerten basierende Posenschätzung über die Zeit eine hohe Abweichung aufweist. Deshalb werden IMUs oft in Kombination mit anderen der oben bereits erwähnten Verfahren zur Posenschätzung fusioniert, um diese Abweichung zu korrigieren.

Foxlin [Fox00] entwickelte für Flugsimulatoren eine Kombination aus IMU und akustischem Tracking. Für Cockpits folgte von Foxlin et al. [Fox04] eine Kombination aus IMU und optischem Tracking. Letzteres nutzt sowohl ein optisches outside-in als auch inside-out Tracking, welches als *inside-outside-in* Tracking bezeichnet wird. Es benötigt wenige LEDs auf dem Helm, allerdings auch LEDs im Cockpit und neben Kameras im Cockpit ebenfalls eine auf dem Helm. Später folgte von Atac et al. [Ata14] eine abgewandelte Version ohne Sensoren, die im Cockpit angebracht werden müssen. Diese nutzt nur den inside-out Teil des oben erwähnten inside-outside-in Trackings, allerdings werden als Sender keine lichtausstrahlenden LEDs, sondern binäre kreisförmige Muster in Form von Aufklebern mit Durchmessern zwischen sechs und 25 Millimetern im Cockpit angebracht und während der Belichtungszeit der Kamera mit Infrarotlicht bestrahlt, um sie robust zu erkennen. Dies reduziert die Komplexität der Hardware ungemein, da keine LEDs im Cockpit angebracht werden müssen und deren Stromversorgung wegfällt. Zusätzlich sind die Aufkleber flexibel positionierbar.

Optisches Tracking wird nicht nur im Cockpit von Fluggeräten verwendet, sondern in weiteren Anwendungsfällen wie bspw. bei der Bewegungserfassung von Personen [Kur02], zum Tracking von Objekten bspw. Werkzeugen, um manuelle Montageprozesse zu überwachen oder zu dokumentieren [Len18], oder VR-Brillen und VR-Eingabegeräten bei VR-Anwendungen. Zu den bekanntesten VR-Brillen zählen derzeit (Februar 2019) die Oculus Rift und HTC Vive [Cue17]. Die Posenschätzung der Oculus Rift nutzt ein outside-in Tracking. An der Brille sind Infrarot-LEDs angebracht, die über eine oder mehrere externe Kameras erfasst werden müssen [Ran17]. Die Posenschätzung der HTC Vive, das sog. Lighthouse-Tracking [Dey15], zählt ebenfalls zu den optischen Verfahren, arbeitet aber mit rotierenden Infrarot-Linienlasern, ähnlich einem bereits 1991 von Ferrin [Fer91] beschriebenen Verfahren zur Bestimmung der Helmposition in Fluggeräten. Die Sensoren sind einzelne an der Brille angebrachte Photodioden. Im Sender arbeiten zwei rechtwinklig angeordnete, rotierende Linienlaser. Die Linien fahren den Raum ab und jede einzelne wird zu bestimmten Zeitpunkten von den Empfängern registriert. Aus Kenntnis der zu den jeweiligen Zeitpunkten korrespondierenden Orientierung der rotierenden Linienlaser lässt sich die

Position der Sensoren relativ zum Sender im Raum ermitteln. Dazu müssen vier Sensoren, deren relative Position zueinander bekannt sein muss, von der Sendeeinheit bestrahlt worden sein.

2.1.6.2 Optische Posenschätzung ohne Referenzmarkierungen

Trägt der Nutzer keinen Helm oder keine VR-Brille, sondern nur eine AR-Brille, die wenig Fläche zur Anbringung von LED-Konstellationen oder anderen optischen Referenzmarkierungen bietet, oder wenn die Anwendung es nicht ermöglicht, in der Umgebung Sender oder Empfänger anzubringen, bietet sich in diesen Fällen reines inside-out Tracking an, das die Umgebung als Referenz nutzt und zusätzlich mit IMUs kombiniert sein kann. Je nach im kopfgetragenen Gerät eingebauter Sensorik bieten sich unterschiedliche Verfahren aus dem Bereich der visuellen simultanen Positionsbestimmung und Kartenerstellung (engl. Simultaneous Localization and Mapping (SLAM)) an. Ein Überblick über SLAM-Verfahren findet sich in Jin and Yang [Jin18]. Diese gibt es für Systeme mit monokularer Sicht [Eng14, Mur15, Eng18], Systeme, die eine Stereokamera verwenden [Eng15], oder Verfahren, die Farb- und Tiefendaten nutzen [Hen12, Mur17].

2.1.7 Realitäts-Virtualitäts-Kontinuum

Milgram et al. [Mil95] führten 1995 das Realitäts-Virtualitäts-Kontinuum (engl. Reality-Virtuality Continuum) zur Einordnung von Darstellungsformen ein. An einem Ende des Kontinuums befindet sich die Realität, in der alles, was man sieht, in Wirklichkeit vorhanden ist. Am anderen Ende liegt die Virtualität, in der nichts von dem, was man sieht, echt ist.

Zwischen beiden Extremen befinden sich gemischte Darstellungsformen. Diese werden allesamt mit gemischter Realität (engl. *Mixed Reality*) bezeichnet. So ist Augmented Reality ein Spezialfall der Mixed Reality, bei dem *primär* die Wirklichkeit, erweitert um virtuelle Elemente, dargestellt wird. Wird primär eine computergenerierte Welt dargestellt und um visuelle Elemente aus der Wirklichkeit erweitert, bezeichnet man dies als erweiterte Virtualität (engl.

Augmented Virtuality, AV). Visuelle Elemente aus der Wirklichkeit können bspw. die Hände des Benutzers oder Hindernisse der realen Umgebung sein, die eingeblendet werden, damit der Nutzer nicht mit diesen zusammenstößt, wenn er sich in der virtuellen Realität bewegt.

In den Bereich der Realität fallen alle Darstellungsformen, die primär eine Umgebung der Wirklichkeit und ggfs. sekundär virtuelle, nicht in der Wirklichkeit vorhandene Elemente abbilden. Die wirkliche Umgebung muss nicht notwendigerweise die des Nutzers sein, sondern kann von einem anderen Ort stammen wie z. B. bei der Telepräsenz, die unter Video see-through AR einzuordnen ist.

Das Spektrum der Virtualität umfasst alle Darstellungsformen, die primär eine computergenerierte, nicht in der Wirklichkeit vorhandene Umgebung und ggfs. sekundär echte, in der Wirklichkeit vorkommende Elemente darstellen. Zu diesen nicht in der Wirklichkeit vorhandenen Umgebungen zählen auch Abbilder der realen Welt, wenn die Bildquelle ihre Sensordaten (Bildern inkl. möglicher Bildverarbeitung) in eine andere Repräsentationsform umwandelt, z. B. in ein Computer-Aided Design (CAD)-Modell oder eine 3D-Punktwolke, deren Ansicht vom Computer erst erzeugt werden muss, um sie betrachten zu können. Die Anwendung Google Earth VR ist deshalb dem Bereich VR zuzuweisen, obwohl in der Wirklichkeit vorhandene Umgebungen dargestellt werden. Interessant für die Einordnung ist das für die transparenten Fahrerhäuser [Rei17, Sec18] genutzte System. Dieses visualisiert primär die Live-Bilder der Stereokamera, erweitert um sekundäre 3D-Punktwolken, die aus Teilbereichen der Live-Bilder gewonnen werden.

Kress und Cummings [Kre17] führten mehr als 20 Jahre nach Milgram et al. [Mil95] eine weitere aber abweichende Klassifikation von kopfgetragenen Geräten ein. Nach ihnen beinhaltet die Klasse *Smart Glasses* monokulare transparente und monokulare intransparente Anzeigesysteme mit Sichtfeldern zwischen 10° und 20°, die möglicherweise eine auf einer IMU basierende Posenschätzung haben, aber hauptsächlich für die nicht ortsreferenzierte Darstellung von visuellen Elementen verwendet werden. Hierzu zählen folglich Vertreter wie die Google Glass [Ols13]. Für VR-Brillen werden die Klassen *mobile VR-Headsets* und *PC-gebundene VR-Headsets* genannt. Erstere

zeichnen sich dadurch aus, dass ihr Display das eines Smartphones ist. Die Posenschätzung wird allein über inside-out Posenschätzung mit einer möglichen Kombination aus IMU und Szenekamera bestimmt. Die Berechnungen finden auf dem Smartphone statt. Zu den *PC-gebundenen VR-Headsets* zählen Vertreter wie die Oculus Rift und HTC Vive [Ran17], die an einen PC angeschlossen sein müssen, bestimmte Anforderungen an die Grafikkarte haben und eine Posenschätzung verwenden, die zusätzliche Sensoren oder optische Referenzmarkierungen im Raum benötigt, siehe Abschnitt 2.1.6.1. Werden derartige VR-Brillen mit einer Stereokamera versehen und die Bilder der Stereokamera auf dem Display der Brille angezeigt und dadurch die reale Umgebung sichtbar, bezeichnen Kress und Cummings diese Klasse von Geräten als *Merged Reality* HMDs. Dieser Begriff ist bei Milgram et al. [Mil95] nicht zu finden. Die Einschränkung des Sichtfeldes durch den Öffnungswinkel der Kamera sorgt hierbei für einen mehr oder weniger großen Tunneleffekt. Die geringe Winkelauflösung der VR-Brillen und die Verzögerung der Darstellung durch das Display sind weitere Unterschiede zu optisch transparenten Anzeigesystemen. Letztere stellen die Realität direkt dar und haben deshalb nicht mit geringen Auflösungen oder Verzögerungen der Umgebungsdarstellung zu kämpfen. Binokulare Varianten dieser optisch transparenten Systeme fallen nach Kress und Cummings in die Klasse *Augmented Reality HMDs*, zu deren wichtigsten Eigenschaften, wie in Abschnitt 2.1.3 erwähnt, Auflösung, Kontrast, Größe des Sichtfeldes und Gewicht gehören. Die *HoloLens* [Kre17] ist ein Vertreter der von Kress und Cummings zusätzlich eingeführten Klasse *Mixed Reality HMDs*, welche im Vergleich zu AR-HMDs eine weitere Evolutionsstufe darstellen, weil sie eine akkurate Verortung in der Umwelt (engl. *3D World Locking*) durch präzise inside-out Posenschätzung aufweisen, ohne dafür externe Hardware oder optische Referenzmarkierungen zu benötigen.

Es ist festzustellen, dass besonders der Begriff *Mixed Reality* von Kress und Cummings anders verwendet wird als bei Milgram et al. [Mil95], bei denen er für alles zwischen den Extremen Realität und Virtualität steht. Bei Kress und Cummings hingegen sind Mixed-Reality-HMDs kopfgetragene binokulare optical see-through Geräte mit inside-out Posenschätzung. Aufgrund der weiteren Verbreitung der Begrifflichkeiten von Milgram et al., werden diese im Folgenden benutzt und wird bei Abweichungen darauf hingewiesen.

Die Klassen von Kress und Cummings können wie folgt in das Realitäts-Virtualitäts-Kontinuum von Milgram et al. [Mil95] eingeordnet werden: *Smart Glasses* gehören zur Realität. Ihre Anzeigesysteme sind wie kleine, am Kopf befestige Monitore zur Darstellung kontextbezogener, aber nicht ortsreferenzierter visueller Elemente. Von Kress und Cummings werden die Anzeigesysteme deshalb auch als *see-around* Displays bezeichnet. Die Klassen der VR-Headsets, unabhängig davon, ob sie mobil oder PC-gebunden sind, gehören je nach Anwendung zur Darstellungsform Virtualität oder erweiterten Virtualität. Die Klasse *Merged Reality HMDs* fällt in den Bereich der Video see-through Augmented Reality wie das System der transparenten Fahrerhäuser. Die Klassen *AR-HMDS* und *Mixed Reality HMDs* gehören zum Bereich der optical see-through Augmented Reality.

2.2 Blickanalyse

Während der Betrachtung der Umwelt nutzt der Mensch im Wesentlichen zwei Augenbewegungen: Fixationen und Sakkaden. Fixationen sind die Verweilzeiten des Auges, während derer sich das Auge kaum bewegt. Sakkaden sind ballistische, schnelle Augenbewegungen zwischen Fixationen. Die Umwelt wird kontinuierlich durch das optische System des Auges auf die Netzhaut (Retina) abgebildet, siehe Abbildung 2.1. Die visuelle Wahrnehmung findet allerdings nur während Fixationen statt [Tob10].

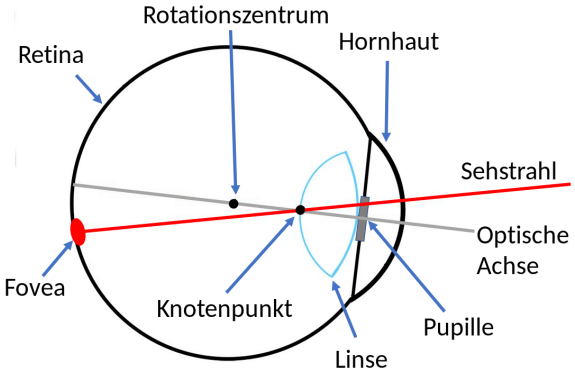


Abbildung 2.1: Aufbau des menschlichen Auges

Obwohl das menschliche Sichtfeld einen Öffnungswinkel von ca. 200° horizontal und 120° vertikal aufweist [Zuc54], nimmt die Schärfe des Sehens mit größer werdendem Abstand zum Schnittpunkt von Netzhaut und Sehstrahl des Auges ab. Der kleine Bereich mit einem Öffnungswinkel von 2° um diesen Schnittpunkt herum wird *Fovea* genannt [Ray98]. Er weist die höchste Dichte an Zapfen auf, welche das Farbsehen ermöglichen, und ist der Bereich des schärfsten Sehens. Die Fovea ist bis zu einem Öffnungswinkel von 5° von der Para Fovea umgeben. Die verbleibende Region wird peripheres Sichtfeld genannt. Auf der Netzhaut befinden sich in diesem Bereich hauptsächlich Stäbchen, welche es uns ermöglichen, auch bei schwachem Licht etwas zu erkennen.

Um Objekte, deren Projektion auf die Netzhaut (Retinabild) einen größeren Öffnungswinkel als 2° hat, visuell erfassen zu können, muss der Blick über das Objekt bewegt und müssen die visuellen Eindrücke aus unterschiedlichen Retinabildern zusammengesetzt werden. Durch die Analyse von Blickbewegungen ist es möglich, die visuelle Aufmerksamkeit eines Menschen zu erfassen, wobei berücksichtigt werden muss, dass die visuelle Aufmerksamkeit und gedankliche Aufmerksamkeit nicht übereinstimmen müssen. Unsere Aufmerksamkeit kann auf etwas gelenkt sein, das wir nicht fokussieren und sich z. B. im peripheren Sichtfeld befindet. Dies wird als verdeckte oder verborgene Aufmerksamkeit (engl. *covert attention*) bezeichnet. Wenn unser Blick Änderungen der Aufmerksamkeit folgt, wird von offener oder offenkundiger Aufmerksamkeit (engl. *overt attention*) gesprochen. Interessanterweise gibt es Situationen, in denen wir eine Szene betrachten, aber blind für Änderungen in der Szene sind, die offensichtlich zu erkennen gewesen wären. Dieses Phänomen wird Veränderungsblindheit (engl. *change blindness*) genannt [Sim05]. Die fehlende Achtsamkeit kann durch komplett andere Erwartungen an mögliche Änderungen in der Szene erklärt werden.

In mobilen AR-Anwendungen ist ein wesentlicher Beitrag der Blickanalyse die Erkennung unbewusst bekundeten Interesses durch die Schätzung der visuellen Aufmerksamkeit. Das Wissen über die visuelle Aufmerksamkeit kann je nach Anwendung auf unterschiedliche Weise weiterverarbeitet werden und zu einer Reaktion des AR-Systems führen. Verstanden wird dieser Vorgang als *implizite Interaktion*, weil der Nutzer nicht explizit interagiert. Wird von der visuellen Aufmerksamkeit auf sein Interesse geschlossen, wird angenommen, dass die oben erwähnte offenkundige Aufmerksamkeit vorliegt. Dies ist in vielen Situationen der Fall, da von Findlay [Fin05] herausgefunden wurde, dass verdeckte Aufmerksamkeit den aktiven Sehprozess begleitet, aber für gewöhnlich nicht ersetzt. Dennoch können sich mentale Vorgänge mit etwas anderem befassen, als dem, was gerade angeschaut wird. Dies ist besonders der Fall, wenn sich die Person in einem müden, gelangweilten oder erregten Zustand befindet. Aus genannten Gründen kann die visuelle Aufmerksamkeit nur zur Erkennung visuell relevanter Bereiche genutzt werden, die als Hinweis auf das übergreifende, gesamte Interesse genutzt werden

können, wodurch sich interessante Anwendungsmöglichkeiten besonders für AR-Anwendungen bieten.

In den nachfolgenden Unterabschnitten wird zuerst der generelle Aufbau mobiler Blickmessgeräte und die mit ihnen berechneten Daten beschrieben (Abschnitt 2.2.1). Es folgen die drei Schritte der Blickanalyse von der Blickmessung in Abschnitt 2.2.2 über die Blickbewegungsanalyse in Abschnitt 2.2.3 zur Blickverhaltensanalyse in Abschnitt 2.2.4. Danach zeigt Abschnitt 2.2.5 Anwendungsbeispiele der Blickanalyse für die implizite Interaktion auf. Abschnitt 2.2.6 behandelt die Nutzung des Blickes für die explizite Interaktion. Auf die manuelle Blickanalyse mit Fokus auf Heatmap-Visualisierungstechniken wird in Abschnitt 2.2.7 eingegangen.

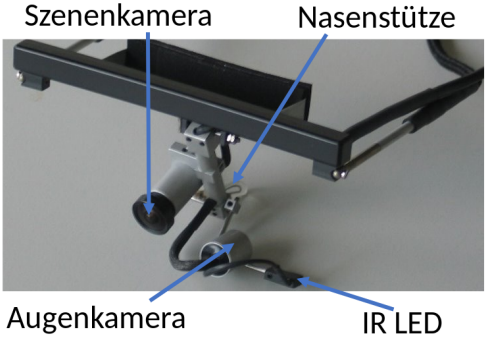


Abbildung 2.2: Mobiles Blickmessgerät *Dikablis* von Ergoneers aus dem Jahr 2010 [Erg11]. Es ist zu erwähnen, dass Ergoneers aktuelles Blickmessgerät binokulare Blickmessung durchführt und unter dem Namen *Dikablis Glasses 3* geführt wird [Erg19].

2.2.1 Mobile Blickmessgeräte

Ältere mobile Blickmessgeräte wie der *Dikablis* aus dem Jahr 2010 besitzen, so in Abbildung 2.2 zu sehen, eine Szenenkamera und eine Augenkamera. Die Blickmessung wird monokular durchgeführt. Die von der zugehörigen Software in Echtzeit bereitgestellten Daten umfassen einen zweidimensionalen Blickpunkt im Szenenkamerabild, welcher der Projektion des Schnittpunktes des Sehstrahls mit der Kalibrierebene entspricht. Weiter unterstützt das

Dikablis System zweidimensionale Referenzmarkierungen, die in der Szene angebracht werden können. Jede Referenzmarkierung spannt im dreidimensionalen Raum eine Ebene auf und der Schnittpunkt des Sehstrahls mit dieser Ebene wird für jede erkannte Referenzmarkierung berechnet und ebenfalls in Echtzeit zur Verfügung gestellt. Die Eckpunkte der Referenzmarkierungen im Szenenkamerabild werden zusätzlich übermittelt. Den Sehstrahl lieferte das System im Jahr 2010 nicht. Aktuellere Versionen des Dikablis Blickmessgerätes und auch die von anderen Herstellern führen eine binokulare Blickmessung durch und liefern ebenfalls den Sehstrahl, was für mobile Anwendungen wichtig ist, siehe Abschnitt 2.2.2, um einen 3D-Blickpunkt in der Szene berechnen zu können.

Während mobile Blickmessgeräte von namhaften Herstellern lange Zeit nicht für unter 15.000 Euro zu kaufen waren, kam mit dem mobilen Blickmessgerät *pupil* [Kas14], siehe Abbildung 2.3, von *pupil labs* ein vergleichsweise kostengünstiges binokulares Blickmessgerät auf den Markt, welches für beide Augen die Sehstrahlen in Echtzeit liefert und dessen zugehörige Software quelloffen ist. Es wurde im BMBF-Projekt KonsensOP [Kon15] vom Operateur getragen und berechnet u. a. einen 2D-Blickpunkt im Szenenkamerabild als auch einen Sehstrahl pro Auge und einen 3D-Blickpunkt relativ zur Szenenkamera.

Abbildung 2.3: Mobiles Blickmessgerät *pupil* von pupil labs [Kas14]

2.2.2 Blickmessung

Die Blickmessung (engl. eye tracking) besteht aus der Pupillendetektion, der Sehstrahlberechnung pro Auge und der anschließenden Blickpunktberechnung. Die Blickpunkte, egal ob zwei- oder dreidimensional bestimmt, bilden einen Blickpfad. Ein Überblick über verschiedene Blickmessverfahren und Blickbewegungsmessgeräte findet sich in Hansen und Ji [Han10]. Die meisten Eye Tracker arbeiten für die Blickmessung mit aktiven Sensoren. Die Augenszene wird mit nahem Infrarot-Licht aus ein oder mehreren Punktlichtquellen beleuchtet und über die Augenkameras werden die Pupille sowie die Reflexionen der Lichtquellen auf der Hornhaut erfasst. Detektionsverfahren für die Pupille und die Kornea-Reflexionen liefern deren Positionen im Bild der Augenkamera. Bei stationären Blickmessgeräten werden zusätzlich Verfahren zur Gesichtserkennung vorgeschaltet, um zuvor die Augenpartien zu extrahieren.

Fortgeschrittenere Verfahren bilden intern ein dreidimensionales Augenmodell des Nutzers und berechnen aus den Positionen der Pupille und der Hornhautreflexionen (Kornea-Reflexionen) zusammen mit dem Augenmodell den Sehstrahl, der die Fovea sowie den Knotenpunkt des optischen Systems des Auges schneidet. Ein mathematisches Modell für die Berechnung des Sehstrahls aus Pupillenposition und Kornea-Reflexionen im Bild ist in Guestrin and Eizenman [Gue06] zu finden.

Zur Blickpunktberechnung wird bei monokularen Verfahren der Sehstrahl mit der virtuellen Kalibrierebene geschnitten und dann in die Szenenkamera des Blickmessgerätes projiziert. Das Resultat ist ein zweidimensionaler Blickpunkt in Pixelkoordinaten, für dessen Berechnung ein Sehstrahl ausreicht [Erg11]. Dies funktioniert solange die Kalibrierebene die Betrachtungsebene ist, z. B. wenn auf einen Monitor geschaut wird und der Monitor zum Kalibrieren genutzt wurde. Bewegt sich allerdings der Kopf des Nutzers, bewegt sich die Kalibrierebene mit und das Ergebnis der Blickpunktberechnung ist nicht mehr korrekt, weil der tatsächliche Blickpunkt nicht mehr auf der Kalibrierebene liegt. Dann liegt der in die Szenenkamera rückprojizierte Blickpunkt nicht an der korrekten Stelle des Blickpunktes im Bild.

Binokulares Eye Tracking löst dieses Problem. Dabei wird für jedes Auge ein Sehstrahl berechnet und der 3D-Blickpunkt dort angenommen, wo die zwei Sehstrahlen den kürzesten Abstand zueinander aufweisen. Dieser kann zusätzlich in die Szenenkamera des Eye Trackers projiziert werden und ist auch bei Kopfdrehungen korrekt berechenbar. Steht nur ein monokulares Blickmessgerät zur Verfügung, lassen sich trotz fehlendem zweiten Sehstrahl dreidimensionale Blickpunkte berechnen. Ein dafür verwendetes Verfahren wird von Thies [Pfe12b] *geometriebasiert* genannt und bedarf der Raumgeometrie der Umgebung als 3D-Modell. Der Sehstrahl wird dabei in der Umgebung rekonstruiert und mit der Umgebung geschnitten. Für die Rekonstruktion des Sehstrahls muss die Pose (Position und Orientierung) des mobilen Blickmessgerätes bekannt sein. Hierfür können Posenschätzungsverfahren aus Abschnitt 2.1.6 verwendet werden.

2.2.3 Blickbewegungsanalyse

Der zweite Schritt der Blickanalyse, die Blickbewegungsanalyse, transformiert den Blickpfad zu einer Blickbewegungsfolge aus Sakkaden und Fixationen. Diese bilden die zwei häufigsten Blickbewegungen. Während sich das Auge bei einer Fixation kaum bewegt, rotiert es bei einer Sakkade um eine beliebige Achse durch das Rotationszentrum des Auges. Der Winkel zwischen dem Sehstrahl vor und nach der Sakkade, wird als Amplitude bezeichnet.

Methoden zur Berechnung von Sakkaden und Fixationen aus 2D-Blickpunkten werden detailliert von Salvucci and Goldberg [Sal00] als auch Komogortsev et al. [Kom10] beschrieben. Letztere liefern zusätzlich eine Methodik zur Evaluation der Genauigkeit der berechneten Blickbewegungen. Die Verfahren mit der höchsten Genauigkeit werden *Velocity Threshold Identification (I-VT)* und *Kalman Filter Identification (I-KF)* genannt. Da diese Verfahren für die Analyse von zweidimensionalen Blickdaten entwickelt wurden, sind für den Einsatz in mobilen Anwendungen Adaptionen vorzunehmen.

2.2.4 Automatische Blickverhaltensanalyse

Der dritte Schritt der Blickanalyse, die Blickverhaltensanalyse, arbeitet mit unterschiedlichen Metriken und ggfs. mit vorab definierten interessanten Bereichen (engl. Areas of Interest (AOIs)) im Bild oder Raum.

Für die automatische Analyse von Blickdaten in mobilen Anwendungen gibt es zwei Herangehensweisen. Eine basiert auf der korrekten Projektion des Blickpunktes in das Szenenkamerabild des Eye Trackers. Dieser 2D-Blickpunkt wird dann mit im Bild erkannten Objekten verknüpft wie bei Toyama et al. [Toy12]. Voraussetzung hierfür ist eine korrekte Objekterkennung in den Bildern der Szenenkamera sowie Segmentierung des Bildes. Die zweite Herangehensweise ist die oben beschriebene geometriebasierte 3D-Blickpunktberechnung mit einem 3D-Modell der Umgebung, in dem zusätzlich bekannt ist, welche Teile der Geometrie zu welchem Objekt gehören. So lässt sich aus dem Schnittpunkt des Sehstrahls mit der Umgebung direkt das betrachtete Objekt ableiten.

Basierend auf Blickbewegungen wie Sakkaden und Fixationen gibt es eine Vielzahl von Metriken. Die verbreitetsten Maßzahlen werden in den Arbeiten von Goldberg und Kotval [Gol99] als auch Goldberg und Helfman [Gol10] beschrieben. Auf sie wird in den nachfolgenden Unterabschnitten inkl. Interpretationsmöglichkeiten genauer eingegangen.

2.2.4.1 Metriken der Blickpfadanalyse

Ein Blickpfad besteht aus einer alternierenden Abfolge von Fixationen und Sakkaden eines betrachteten Zeithorizontes. Die Gesamtanzahl an Sakkaden als auch die Summe der Amplituden der Sakkaden sind Metriken für die Länge eines Blickpfades. Die Länge eines Blickpfades ist ein Indikator für das Ausmaß der visuellen Betrachtung. Ein langer Blickpfad entsteht bei großflächigem Suchverhalten. Während Fixationen nimmt der Mensch seine Umwelt wahr. Statistiken über die Dauer von Fixationen sind ein Indikator für das Ausmaß an lokal stattfindender Informationsverarbeitung oder Verarbeitungskomplexität für den betrachteten Bereich. Die gesamte Dauer eines

Blickpfades für einen bestimmten Bereich ist eine Aussage über die Komplexität der für ihn benötigten gedanklichen Verarbeitung. Das Verhältnis von aufsummierten Fixations- und Sakkaden-Zeiträumen kann auf die folgenden Arten interpretiert werden: Ein kleines Verhältnis deutet auf ausgedehntes Suchverhalten hin, während ein großer Wert zeigt, dass häufiger visuelle Stimuli verarbeitet werden und Suche oder freies Betrachten reduziert sind. Zwischen aufeinanderfolgenden Sakkaden kann ein relativer Winkel berechnet werden, der die Änderung der Richtung der Sakkade bzw. die Änderung der Rotationsachse beschreibt. Die Summe der aufsummierten relativen Winkel ist ein Indikator für die Art der Betrachtung. Eine kleine Summe zeigt, dass der Blick entlang strukturell angeordneten visuellen Reizen, z. B. Kanten, verlaufen ist und so wenige Richtungsänderungen vorgenommen hat. Bei visuellem Stimulus ohne Struktur werden häufiger Richtungsänderungen vollzogen, wodurch die Summe der relativen Winkel größer wird, weil der Blick nicht geleitet wird.

2.2.4.2 Bereichsbasierte Metriken

Da die Blickanalyse für Aussagen über die visuelle Anziehungskraft oder Relevanz von Objekten genutzt werden soll, werden häufig oben genannte AOIs definiert, die Objekte oder Bereiche einschließen. Mit ihnen lassen sich Fixationen Objekten zuordnen. Es gibt zwei verschiedene Arten der Definition von AOIs: Die eine besteht aus der generischen Unterteilung des Bildes oder der Szene durch ein Raster, die andere aus der Anpassung von AOIs an die Objekte, die von Interesse sind. AOI-basierte Blickanalyse ermöglicht die Berechnung weiterer Metriken. Werden AOIs auf Basis eines generischen Rasters angenommen, ist die räumliche Dichte die Anzahl an Feldern, welche mindestens eine Fixation beinhalten, geteilt durch die Gesamtanzahl an Feldern des Rasters. Größere Werte deuten auf ein gleichmäßig verteiltes Blickverhalten hin, während kleinere Werte auf eine geringere Verteilung von Fixationen hindeuten, was wie beim oben beschriebenen aufsummierten relativen Winkel zwischen Sakkaden auf eine der Szene zugrundeliegenden Struktur hinweisen kann. Nachfolgende Metriken gelten ebenfalls für nicht generisch definierte AOIs. Die Zeit bis zur ersten Fixation einer AOI oder die Zeit nach der

jede AOI einer Untermenge aller AOIs mindestens einmal durch eine Fixation im Fokus der Aufmerksamkeit lag, gibt Einsicht in die Anziehungskraft der AOI oder Menge von AOIs. Diese Zeiten bis zur ersten Fixation können ebenfalls in Reihenfolge gebracht werden und es kann festgestellt werden, welche Bereiche früher den Blick auf sich ziehen als andere. Die Anzahl an Folgefixationen einer AOI auf ihre erste Fixation gibt Auskunft über die Relevanz des zugehörigen Objektes. Falls es bereits betrachtet wurde, aber nicht relevant bzw. von Interesse ist, wird die Anzahl an Folgefixationen gering bleiben. Der Anteil an Fixationen einer AOI oder der Anteil an kumulierter Fixationszeit einer AOI an der gesamten kumulierten Fixationszeit einer Szene gibt einen Überblick über die Verteilung der visuellen Aufmerksamkeit in dieser Szene.

2.2.4.3 Analyse von Übergangsmatrizen

Eine weitere Möglichkeit für die Analyse des Blickverhaltens ist die Erstellung einer quadratischen Übergangsmatrix **A**. Bei n AOIs enthält die Übergangsmatrix n^2 Einträge. Für zwei AOIs i und j gibt es einen Eintrag a_{ij} in Zeile i und Spalte j, der die Anzahl an Sakkaden enthält, die ihren Start in AOI i und ihr Ende in AOI j genommen haben. Der Übergang in entgegengesetzter Richtung findet sich in Eintrag a_{ji}. Die Übergangsdichte ist die Anzahl an Matrixelementen, die nicht null sind. Kleinere Werte deuten auf einen gerichteten Blickpfad hin. Die visuelle Relevanz einer AOI ist höher, je mehr Übergänge in ihr enden. Ein Beispiel für eine solche AOI könnte z. B. die Adresszeile in einem Browser sein.

2.2.5 Blickanalyse für die implizite Interaktion

2.2.5.1 Wahrnehmung von Bildern und Kunstwerken

Nachfolgend werden Studien vorgestellt, in denen oben beschriebene Metriken für die Blickanalyse bei der freien Betrachtung von Bildern und Kunstwerken angewendet wurden.

Kaspar und König [Kas11] analysierten, wie Personen mit verschiedenem Interesse Bilder aus unterschiedlichen Kategorien betrachteten. Sie fanden heraus, dass bei Personen, die sich für ein Bild interessierten, Fixationen kürzer wurden, mehr Sakkaden stattfanden und die Fixationen global über das Bild verstreut waren.

Buswell [Bus35] analysierte den Blick auf Kunstwerke. Die Personen durften sich diese ohne Anweisung anschauen. Er schlussfolgerte, dass die Augen Hauptlinien im Bild folgen und dass visuell komplexe Regionen längere Fixationen hervorrufen sowie Gesichter die Aufmerksamkeit auf sich ziehen. Diese Schlussfolgerung machten 75 Jahre später auch Engelbrecht et al. [Eng10].

Locher and Nodine [Loc08] beschreiben zwei Phasen während der Betrachtung von Kunstwerken. Die erste dient dazu, einen generellen Eindruck der Struktur des Bildes zu bekommen. Eine Art ästhetische Beurteilung findet ebenfalls beim ersten Blick auf das Kunstwerk statt. Die zweite Phase der Betrachtung beinhaltet nach Nodine et al. [Nod93] viele Überblicksfixationen von 200-300 Millisekunden (ms) Länge, um relevante Bereiche auszumachen. Zwischen diesen Überblicksfixationen finden Beobachtungsfixationen von 400 ms Länge statt, um ausgemachte relevante Bereiche genauer zu betrachten. Locher und Nodine betrachteten zusätzlich die Abdeckung der betrachteten Kunstwerke nach drei und sieben Sekunden sowie am Ende der Betrachtung. Eine wesentliche Beobachtung dabei ist, dass nach sieben Sekunden fast keine zusätzlichen Bereiche mehr die visuelle Aufmerksamkeit auf sich ziehen. Um den Übergang der ersten Phase, der Überblickgewinnung, zur zweiten Phase, der detaillierteren Betrachtung relevanter Bereiche, zu erkennen, kann das Verhältnis von Fixationsdauer zu Sakkadendauer genutzt werden. Dieses steigt beim Übergang der Phasen an.

Die Kenntnis der Entwicklung bestimmter Metriken kann genutzt werden, um z. B. zu unterscheiden, ob jemand in einer Phase freien Betrachtens ist oder Details eines relevanten Bereichs untersucht. Die Entwicklung verschiedener Metriken kann aber auch im Zusammenspiel betrachtet werden, wie in nachfolgendem Abschnitt beschrieben.

2.2.5.2 Erkennung relevanter Bereiche

In diesem Abschnitt werden Studien vorgestellt, die sich mit der Erkennung relevanter Bereiche beschäftigt haben. Die verwendeten Metriken werden zumeist zusammengefasst und Klassifikatoren trainiert, um die visuelle Relevanz von Bereichen zu bestimmen.

Klami et al. [Kla08] versuchten die Relevanz von Bildern in einem kontrollierten Szenario basierend auf Blickbewegungen zu erkennen. Vier Bilder wurden den Testpersonen gleichzeitig dargestellt und die Aufgabe war die Aussage zu treffen, ob eines der vier Bilder etwas mit Sport zu tun hat. Der genutzte Merkmalsvektor beinhaltete die gesamte Dauer der Fixationen, die Anzahl der Fixationen, die durchschnittliche Länge einer Fixation, die Anzahl an Übergängen zwischen den Bildern, die Anzahl an Bildern mit mindestens einer Fixation und die Anzahl an Fixationen pro Bild. Es konnte gezeigt werden, dass durch diese einfachen Metriken in Kombination mit einer linearen Diskriminanzanalyse anhand der Augenbewegungen größtenteils festgestellt werden konnte, ob eines der gezeigten Bilder und welches etwas mit Sport zu tun hatte.

Kandemir et al. [Kan10] testeten für ihre AR-Anwendung, ob aus Blickdaten die Relevanz von Objekten einer Szene bestimmt werden kann, um basierend auf der Relevanz zu entscheiden, welche Informationen als visuelle Elemente eingeblendet werden sollen. Ihre Definition von Relevanz ist das Interesse, mehr über ein Objekt durch visuelle Elemente zu erfahren. Die benutzten Merkmale waren die gesamte Fixationsdauer, die durchschnittliche Fixationsdauer auf allen zu einem Objekt gehörenden AOIs, die mittlere Distanz aller Fixationen zu dem Schwerpunkt der AOIs eines Objektes und die mittlere Länge an Übergangen zu den AOIs eines Objektes. Es zeigte sich, dass die Kombination aller Merkmale einzelnen Merkmalen gegenüber auch bei unterschiedlichen Zeitfenstern überlegen war. Zu diesen zählte auch ein Verfahren, dass ausschließlich die kumulierte Fixationszeit betrachtete.

Ajanki et al. [Aja10] beobachteten das unbewusst bekundete Interesse von Nutzern ihres mobilen AR-Systems, um zu entscheiden, was als nächstes angezeigt werden soll. Zur Erkennung relevanter Objekte nutzten sie sowohl einen Spracherkenner als auch eine Analyse des Blickverhaltens. Bei der Blickanalyse wurde die Betrachtungszeit eines Objektes bestimmt und in einem Zeitfenster entschieden, welches Objekt am längsten angeschaut wurde. Dieses AR-System wurde gegen ein Tablet-basiertes System getestet und erzielte leichte Vorteile bei der Nutzbarkeit, auch wenn der prototypische Charakter des Systems Schwächen bei der Darstellungsqualität der visuellen Elemente aufwies. Über die Sprache als relevant erachtete Objekte wurden stets solche vorgezogen, die über die Analyse des Blickes ermittelt worden waren.

2.2.6 Blickbasierte explizite Interaktion

Explizite Interessensbekundung ist ein weiterer wichtiger Interaktionsaspekt für mobile AR-Anwendungen und kommt zum Tragen, wenn visuelle Elemente im Sichtfeld dargestellt werden und der Nutzer mit diesen interagieren soll, z. B. etwas auswählen soll oder durch ein Menü manövrieren möchte. Handgesten sind optimal für diese Aufgabe, aber es gibt eine Menge Versuche, den Blick für die explizite Interaktion zu nutzen. Der Blick ist schneller und weniger beanspruchend als Armbewegungen, aber letzte Ansätze für die blickbasierte Auswahl erscheinen weniger intuitiv und frustrierender zu sein. Selektionsmethoden auf Basis des Blickes sind besonders wichtig in Anwendungen für Menschen mit Behinderungen, die nur noch den Blick als Kommunikationskanal mit der Außenwelt haben. Eine Selektion besteht aus zwei Schritten: Der erste Schritt ist das Anvisieren des auszuwählenden Ziels. Der zweite Schritt besteht in der Bestätigung des aktuell anvisierten Bereichs. Am normalen Desktop-Arbeitsplatz wird diese Aufgabe mit der Maus durchgeführt. Über die Bewegung der Maus wird der Cursor positioniert und durch die Betätigung einer Maustaste die Selektion vollendet. Das Anvisieren des Ziels durch den Blick ist sehr schnell und bereits Teil des Anvisierprozesses mit der Maus. Die visuelle Kontrolle, ob der Cursor am richtigen Platz ist, entfällt, wodurch ein intuitiveres Anvisieren entsteht. Den Tastendruck

durch den Blick zu ersetzen ist die Schwierigkeit. Meistens wird der Mausklick durch ein Verweilen auf dem Ziel für eine gewisse Dauer, der Verweilzeit (engl. dwell time), durchgeführt. Selektion per Verweilzeit resultiert im *Midas Touch* Problem [Jac90], denn die Selektion wird oft unbeabsichtigt ausgelöst, da Anvisieren und Bestätigen nicht durch orthogonale Modalitäten durchgeführt werden, sich also gegenseitig beeinflussen. Eine perfekte Verweilzeit kann es daher nicht geben und Nutzer passen sich an diesen Missstand an, indem sie den Blick vom Ziel abwenden und es sofort wieder anschauen, falls sie es betrachten wollen, ohne es zu selektieren [Zan10].

Eine Lösung des Problems bieten Zander et al. [Zan10]. Sie messen die Hirnströme des Nutzers um eine Gehirn-Computer-Schnittstelle zu realisieren, die die Bestätigung des Selektionsprozesses übernehmen soll, das Anvisieren wird weiterhin durch den Blick durchgeführt. Der Nutzer denkt dabei an etwas Bestimmtes, in diesem Fall das Auswringen eines Handtuchs, wenn er die Bestätigung durchführen will. Verglichen mit Selektion basierend auf der Nutzung von Verweilzeiten, war der Ansatz nicht schneller, aber genauer und von den Nutzern deutlich bevorzugt. Praxistauglich ist der Ansatz aufgrund des aufdringlichen Messgerätes für die Hirnströme nicht, die meistens in Form von Kappen aufgesetzt werden müssen und für exakte Messungen bestimmte Gels zwischen Kopfhaut und Sensoren benötigen.

Praxistauglicher ist hier der Einsatz von tragbaren Tastern. Barz et al. [Bar18] nutzten z. B. einen leichtgewichtigen Präsenter, der normalerweise zum Präsentieren von Folien gedacht ist. Weniger unaufdringlich sind Präsenter, die direkt auf einen Finger wie ein Ring aufgesteckt werden können.

Ohne Taster sind neben der auf Verweilzeiten basierenden Selektion – das Zwinkern und andere unnatürliche Augengesten für die Bestätigung der Selektion außer Acht lassend – Handgesten bereits die naheliegendste intuitive Alternative für mobile Anwendungen. Gerade deshalb sind Handgesten mit die wichtigste Modalität für die Interaktion mit den aktuell (Februar 2019) fortgeschrittensten kommerziellen AR-Systemen, der HoloLens [Kre17] und der *Magic Leap One* [Yeo17]. Kapitel 4 beschäftigt sich mit dem Thema Handregionsbestimmung, das essentiell für die Erkennung von Handgesten ist.

2.2.7 Manuelle Blickverhaltensanalyse

Neben der automatischen Analyse von Blickdaten und weil die Interpretation der Blickanalyseergebnisse weitestgehend Aufgabe des Menschen bleibt, sind besonders Visualisierungen des Blickverhaltens für die manuelle Analyse wichtig.

Bei Abtastraten des Blickes zwischen 30 und 120 Hz kommen pro Minute 1.800 bis 7.200 Blickpunkte zustande. Visualisierungstechniken, die Kreise bzw. Kugeln für Blickpunkte und Fixationen einzeichnen und Sakkaden als Linien darstellen, sind unpraktisch, da sich durch Verdeckungen die Dichte der Aufmerksamkeit nicht erfassen lässt. Intuitiv und schnell verständlich sind Heatmap-Visualisierungen wie in Duchowski et al. [Duc12], die Bereiche der betrachteten Szene je nach Dichte der vorhandenen Blickpunkte oder Fixationen wie bei einem Wärmebild einfärben. Eine die Blickdaten analysierende Person bekommt bei der Betrachtung einer Heatmap schnell einen Überblick darüber, welche Bereiche der betrachteten Szene am meisten Aufmerksamkeit bekommen haben. Heatmaps werden nach Pfeiffer und Memili [Pfe16] meist für den ersten Überblick über die gesammelten Blickdaten genutzt.

Duchowski et al. [Duc12] präsentierten einen Ansatz für die Zeichnung von Heatmaps für zweidimensionale Blickpfade. Für jeden Blickpunkt oder jede Fixation wird eine Gauß-Verteilung, zentriert an der jeweiligen Position, gebildet. Pro Datum entsteht eine Verteilung über das gesamte Bild. Alle Verteilungen des betrachteten Zeitraums werden aufsummiert und anschließend normalisiert. Jedes Pixel enthält anschließend eine Gewichtung, die per Transferfunktion in einen Farbwert umgewandelt wird. Durch die Nutzung von speziellen Funktionen auf Grafikkarten ist mit dieser Vorgehensweise ein echtzeitfähiges Verfahren für 2D-Blickpfade realisierbar.

Blickanalyse für statische nicht mobile Blickmessgeräte ist üblicherweise für Szenarien gedacht, in denen die Testpersonen Bilder oder Videos auf einem Bildschirm anschauen, z. B. zur Analyse von Werbevideos. Dabei soll herausgefunden werden, ob die Werbung ihren Effekt erzielt und bestimmte Schriftzüge oder Produkte ihre gewollte Aufmerksamkeit erhalten. Durch das Aufkommen von mobilen Blickmessgeräten sind dreidimensionale Szenarien wie

in Geschäften [Har13] oder im Sport [Cam13b] für die Blickanalyse umsetzbar geworden. Es bedarf daher einer Heatmap-Visualisierung für dreidimensionale Szenarien. Existierende Ansätze werden im Folgenden betrachtet.

Stellmach et al. [Ste10b] zeigen drei Ansätze für die Visualisierung von dreidimensionalen Blickdaten: Eine projizierte, eine objektbasierte und eine polygonbasierte Darstellung. Bei der projizierten Darstellung wird im Raum eine virtuelle Kamera platziert und eine gewöhnliche zweidimensionale Heatmap aus den Blickdaten erstellt. Diese Darstellung dient dem räumlichen Überblick der Blickverteilung aus einer bestimmten Sicht. Die objektbasierte Darstellung visualisiert die kumulative Fixationszeit pro Objekt durch Einfärben des gesamten Objektes in eine mit der Fixationszeit korrespondierenden Farbe. Dies kann verglichen werden mit einer AOI-basierten Analyse, bei der die Polygonmodelle der Objekte der AOI gleichen. Bei der polygonbasierten Heatmap-Visualisierung wird um jeden 3D-Blickpunkt eine dreidimensionale Gauß-Verteilung gelegt und jeder Eckpunkt (engl. Vertex) des Polygonmodells erhält eine Gewichtung entsprechend aller Gauß-Verteilungen des betrachteten Blickpfades. Über eine Transferfunktion wird jedem resultierenden Gewicht eine Farbe zugewiesen. Ein Polygon wird anschließend durch Interpolation der Farben seiner Eckpunkte texturiert. Aus diesem Grund hängt das Aussehen der resultierenden Heatmap von der Polygon-Struktur der 3D-Modelle ab und bedarf für eine detaillierte Blickvisualisierung auf einfachen 3D-Strukturen, die mit wenigen Polygonen dargestellt werden können, einer Repräsentation aus vielen kleinen Polygonen. Dieser Sachverhalt wird von Stellmach et al. in Abbildung 3 in [Ste10a] beschrieben und resultiert möglicherweise in einer inkorrekten Darstellung des Blickverhaltens. Ein weiterer Nachteil der polygonbasierten Visualisierung ist, dass die Richtung des einfallenden Sehstrahls nicht berücksichtigt wird. Es ist also kaum möglich, die Richtung, aus der betrachtet wurde, zu erkennen. Zusätzlich werden Verdeckungen nicht berücksichtigt. Bekommt bspw. die Rückseite eines Objektes hohe Gewichte, weil die Standardabweichung der Gauß-Verteilung vergleichsweise groß zu den Ausmaßen des Objektes ist, entsteht der Eindruck, die Rückseite des Objektes wurde ebenfalls angeschaut, obwohl dies nicht der Fall war. Dieses Problem ist in Abbildung 3 in [Ste10a] links bei der Ente zu sehen, deren Hals auch auf der Rückseite eingefärbt ist.

Beim Volumengrafik-Ansatz von Pfeiffer et al. [Pfe12a] werden Fixationen durch eine kontinuierliche dreidimensionale Gauß-Verteilung dargestellt. Die Verteilung repräsentiert dabei die visuelle Schärfe um den Sehstrahl herum. Zusätzlich wird die Verteilung rechtwinklig zum Sehstrahl in Abhängigkeit der Distanz der 3D-Position der Fixation zum Betrachter vergrößert. Die Gewichtung und damit auch die Einfärbung wird mit größer werdender Dauer einer Fixation intensiver gewählt. Durch den Volumengrafik-Ansatz können auf einfache Weise verschiedene Zeitspannen oder auch Blickpfade verschiedener Personen dargestellt werden, weil die Volumendaten nur integriert werden müssen. Die resultierende Visualisierung ist nicht realistisch, weil auch hier die visuelle Schärfe des menschlichen Blickes nicht in die Szene projiziert wird und keine Verdeckungen berücksichtigt werden. Tatsächlich ist dieser Ansatz ähnlich der Visualisierung transparenter, orientierter Ellipsoide an den Stellen der 3D-Blickpunkte oder Fixationen.

Oben erwähnte Verfahren bilden nach Blascheck et al. [Bla17] weiterhin den Stand der Technik der Heatmap-Visualisierung.

Hinzugefügt werden muss das Verfahren von Pfeiffer und Memili [Pfe16], welches dreidimensionale Heatmaps inkl. der Berücksichtigung von Verdeckungen erzeugt. Die Grundidee ist, dass für das gesamte Umgebungsmodell und einzelne möglicherweise dynamische Objekte dieser Umgebung eine zweite Textur erstellt wird, in der die Gewichte der Aufmerksamkeitsverteilungsfunktionen gespeichert werden. Die Aufmerksamkeitsverteilung entspricht einer dreidimensionalen Gauß-Verteilung an der Position des Blickpunktes. Die akkumulierten Gewichte ergeben sich entsprechend aus der Aufsummierung aller Verteilungen, die für jeden Blickpunkt konstruiert werden. Verdeckungen werden durch die Anwendung von *Shadow Mapping*, einer Methode der Computergrafik, um Schattenwurf zu berechnen, berücksichtigt. Durch die Speicherung der die Aufmerksamkeit repräsentierenden Gewichtungen in einer Textur ist es zusätzlich möglich, sich die Heatmap in gängigen 3D-Visualisierungswerkzeugen anzuschauen.

Zusammenfassend kann gesagt werden, das alle Verfahren zur Darstellung von dreidimensionalen Heatmaps, das von Pfeiffer und Memili [Pfe16] ausgenommen, keine korrekte Visualisierung der Szenenwahrnehmung darstellen,

weil Sie dreidimensionale Gauß-Verteilungen um Blickpunkte oder Fixationen legen und keine Verdeckungen berücksichtigen. Dies führt zu Situationen, in denen Bereiche eingefärbt werden können, obwohl sie nicht betrachtet wurden, wenn die Ausmaße der Geometrie im Vergleich zum gewählten Ausmaß der Gauß-Verteilung klein sind. Abschnitt 3.2 behandelt die in der vorliegenden Arbeit entwickelten Verbesserungen im Bereich der Visualisierung von 3D-Heatmaps für die Blickanalyse.

2.3 Handgestenbasierte Interaktion

Die Hände bieten sich in mobilen AR-Anwendungen besonders für die explizite Interaktion an, da der Mensch es gewöhnt ist, mit den Händen Objekte zu greifen und zu manipulieren oder mit Zeigegesten auf etwas zu zeigen.

Die Hand nimmt dabei unterschiedliche Positionen ein und die Finger können teils gestreckt oder gebeugt sein. Alle Parameter, die die Handposition und Konfiguration der Finger beschreiben, werden unter dem Begriff *Handpose* zusammengefasst. Handgesten für mobile AR-Anwendungen lassen sich aus der Handposition oder unterschiedlichen Handposen ableiten. Sie werden genutzt, um mit dem AR-System über visuelle Elemente zu interagieren. Faktoren, die dabei berücksichtigt werden müssen, sind die körperliche Beanspruchung zur Durchführung einer Geste, ihre Intuitivität im gegebenen Kontext sowie die Erkennungsgenauigkeit und -geschwindigkeit des Systems. Jede Anwendung benötigt folglich ein Gestenvokabular, das ausschließlich robust erkennbare Gesten nutzen und diese im gegebenen Kontext einer intuitiven Bedeutung zuweisen sollte, die falls nötig schnell gelernt und gut in Erinnerung behalten werden kann.

Stehen die Handposen durch die verwendete Sensorik und genutzten Verfahren nicht zur Weiterverarbeitung zur Verfügung, sondern nur die Handpositionen, können aus diesen positionsbasierte Handgesten oder Wischgesten konstruiert werden. Die Konstellation der Finger spielt dabei keine Rolle. Die Verfahren bestimmen entweder nur die Handgelenkposition oder die Handposition. In den Projekten ARtSENSE [Dam12] und KonsensOP [Kon15] sowie bei Spielen für Spielekonsolen mit Tiefensensoren zur Menscherfassung wird dieser Gestentyp genutzt, um durch Menüs zu navigieren oder zwischen Inhalten auszuwählen bzw. das Anvisieren des Selektionsprozesses, siehe Abschnitt 2.2.6, mit der Hand durchzuführen.

Eine Handpose kann für sich allein gesehen eine statische Handgeste sein wie in Fingeralphabeten, bei denen die Fingerkonstellation einen Buchstaben beschreibt [Bin05, Ras09b], oder direkt ein anwendungsspezifisches Kommando

auslösen [Bad09, Mai12]. Werden Abfolgen von Handposen als Geste definiert, wird von dynamischen Handgesten gesprochen. Als Beispiel kann man sich eine Geste zum Greifen von Objekten vorstellen, welche mit einer offenen Hand, Finger gestreckt, beginnt und mit einer geschlossenen Hand, einer lockeren Faust, endet. Diese Abfolge bestünde aus zwei Handposen, welche zeitlich nacheinander auftreten müssen. Solche Abfolgen von zwei aufeinanderfolgenden Handposen werden in den aktuell fortgeschrittensten kommerziellen AR-Systemen, der HoloLens [Kre17] und der *Magic Leap One* [Yeo17] genutzt.

Die Selektion bei der HoloLens funktioniert wie folgt [Hol18]: Die Kopfausrichtung wird für das Anvisieren genutzt und die Bestätigung mit einer *Air tap* genannten Handgeste durchgeführt. Die Start-Pose ist dabei die geschlossene Hand mit ausgestrecktem Zeigefinger, ähnlich der Handpose beim Zeigen auf ein Objekt, und die End-Pose die geschlossene Hand mit angewinkeltem Zeigefinger. Der Zeigefinger wird im Vergleich zur Start-Pose also nur gebeugt, was einem Mausklick in der Luft gleichkommt. Dabei muss diese Geste nur von den Sensoren der HoloLens erfassbar sein, der Ort ihrer Durchführung hat keinen Einfluss auf das Anvisieren. Die HoloLens erkennt eine weitere Geste namens *bloom*, welche zum Öffnen des System-Startmenüs dient.

Bei der Magic Leap One werden acht Handposen namens *Open Hand*, *Closed Fist*, *OK Sign*, *Thumbs Up*, *Open Pinch*, *Closed Pinch*, *Relaxed Point* und *Closed Point* erkannt [Mag18]. Die Positionen des Handrückens sowie die der Finger können zusätzlich für die Interaktion genutzt werden. Magic Leap schlägt den Entwicklern zum Selektieren bspw. folgende Kombinationen vor: Für die Selektion von Objekten soll die *Relaxed Point*-Pose zum Anvisieren genutzt werden, welche der Start-Pose der bei der HoloLens verwendeten *Air tap* Geste ähnelt, nur dass der Daumen ebenfalls ausgestreckt ist. Die Bestätigung des Selektionsprozesses von sich in Armreichweite befindlichen Objekten soll durch einfaches Berühren des Objektes mit dieser Geste realisiert werden. Bei Objekten, die nicht in Armreichweite sind, soll die Bestätigung durch die *Closed Point*-Handpose realisiert werden, bei der der Daumen angewinkelt wird, der Zeigefinger ausgestreckt bleibt. Das Greifen eines Objektes soll durch das

Berühren und Bilden einer Faust umgesetzt werden. Magic Leap weist Entwickler an, solche ein Objekt direkt berührende Gesten mit viel Spielraum für Ungenauigkeit umzusetzen. Dies ist aufgrund der fehlenden taktilen Rückmeldung durch virtuelle Objekte und Ungenauigkeiten bei der Handposenschätzung zu erklären.

Bei der Betrachtung von optischen Verfahren zur Handgestenerkennung ist die Handlokalisierung oder Handregionsbestimmung ein elementarer erster Schritt, um eine Bildregion zu bestimmen, welche die Hand beinhaltet. Dazu wird über Segmentierungs- oder Objektlokalisationsverfahren die Handregion bestimmt. Die Handregion wird in einem zweiten Schritt für die Schätzung von statischen Gesten oder Handposen weiterverarbeitet. Auch wenn die oben erwähnten Systeme HoloLens und Magic Leap One Tiefensensorik für die Handgestenerkennung nutzen, wird das Thema der auf monokularen Farbbildern basierenden Handposenschätzung im Bereich der Wissenschaft intensiviert. Die Gründe hierfür ergeben sich wie folgt: Für die breite Akzeptanz von AR-Brillen sind neben der Erfüllung der Anforderungen an die Bildqualität, Auflösung, Kontrast und Verzerrung sowie Ausdehnung des Sichtfeldes (siehe Abschnitt 2.1.3), das optische Erscheinungsbild der AR-Brille und deren Akkulaufzeit wichtige Faktoren. Deren Optimierung gestaltet sich umso schwieriger, desto größer und komplexer die Sensorik ist und desto mehr Leistung sie benötigt. Die Nutzung rein monokularer passiver Sensorik ist folglich ein klarer Vorteil. Da sich die monokulare Szenensegmentierung ohne Tiefeninformation jedoch als deutlich schwieriger gestaltet, ist die Bestimmung der Handregion hier bereits eine schwierigere Aufgabe, die als Grundlage für die Gestenerkennung robust gelöst werden muss. Die vorliegende Arbeit beschäftigt sich nachfolgend speziell mit dem Thema der Handregionsbestimmung auf monokularen Farbbildern, da ein Kernteil der vorliegenden Arbeit Verfahren für die Handregionsbestimmung sind. Zuvor wird der Stand der Technik im Bereich Handregionsbestimmung und Handposenschätzung, die auf Tiefenbildern bzw. Farbbildern arbeiten, behandelt. Abschnitt 2.3.1 stellt Verfahren auf Tiefenbildern vor und Abschnitt 2.3.2 Verfahren, die keine Tiefenbilder nutzen. Eine Zusammenfassung der genutzten Verfahren zur Handregionsbestimmung liefert Abschnitt 2.3.3. Die von den Verfahren

genutzten Datensätze werden ebenfalls behandelt. Abschnitt 2.3.4 behandelt die Datensätze mit und Abschnitt 2.3.5 die ohne Tiefenbilder.

2.3.1 Handregionsbestimmung und Posenschätzung auf Tiefenbildern

Mit der breiten Verfügbarkeit kostengünstiger Tiefensensoren wie der Microsoft Kinect [Zha12] standen Tiefenbilder für Bildverarbeitungsverfahren zur Verfügung, wodurch eine Szenensegmentierung und damit eine Trennung von im Bild erfassten Objekten deutlich erleichtert wurde. Zusätzlich veröffentlichten Shotton et al. [Sho11] ein Verfahren zur Körperposenschätzung, welches das Handgelenk und die Handposition aus frontaler Sicht auf eine Person bestimmt. Dazu werden beim Training zufällig spezielle Features definiert, pro Pixel berechnet und ein Random Decision Forest [Bre01] trainiert, um während der Klassifikation pro Pixel eine Körperteilzugehörigkeit zu berechnen. Das Verfahren fand Einzug in die Spielewelt von Microsofts Spielekonsole XBox.

Verfahren zur Handposenschätzung können in diskriminative und generative Ansätze unterteilt werden. Diskriminative Verfahren schließen aus der Beobachtung direkt auf eine Handpose. Generative Verfahren erzeugen aus einer Hypothese für die zu schätzenden Parameter eines Handmodells ein Erscheinungsbild der Hand, welches mit der aktuellen Beobachtung verglichen wird. Je weniger unterschiedlich der Vergleich ausfällt, desto besser ist die Hypothese. Frühe diskriminative Methoden alleine waren zu ungenau, weil der Satz an Posen, auf die sie schließen konnten, zu klein war, um hohe Genauigkeit zu erzielen. Da die Anzahl an zu testenden Hypothesen bei generativen Verfahren aufgrund der hohen Dimensionalität von mehr als 20 Parametern zu hoch ist, entstanden hybride Verfahren, die versuchten, den Parameterraum effizienter zu durchsuchen, indem sie diskriminative und generative Verfahren geschickt kombinieren. Nachfolgend wird zuerst auf Vertreter dieser Verfahren eingegangen.

Oikonomidis et al. [Oik11] nutzen eine Hautfarbenerkennung im Farbbild eines Tiefensensors, um die Region der Hand zu bestimmen. Diese wird

anschließend mit morphologischer Dilatation erweitert. Alle 3D-Punkte, die nicht weiter weg sind als 25 cm von der im vorangegangenen Einzelbild bestimmten Handposition, werden entfernt. Das verwendete dreidimensionale Handmodell setzt sich aus Ellipsoiden, Kugeln und Kegeln zusammen und die Handpose ist durch 27 Parameter vollständig definiert. Verschiedene Hypothesen für die Handpose werden über ein Energiefunktional bewertet, das sich aus dem Vergleich des beobachteten Tiefenbildes und eines zur Hypothese gehörenden gerenderten Tiefenbildes sowie eines Terms zusammensetzt, der Modell und zuvor berechnete Hautfarbensegmentierung vergleicht und entsprechend bewertet. Ein erweiterter *Particle Swarm Optimization (PSO)*-Ansatz [Yas10] wird genutzt, um die beste Hypothese im globalen Suchraum zu finden.

Tompson et al. [Tom14] erstellten den Datensatz *NYU Hand Pose* mit einer abgeänderten Methode von Oikonomidis et al. [Oik11]. Nach dem Ausführen von PSO wird bei ihnen zusätzlich eine lokale Optimierung mit dem Nelder-Mead-Verfahren [Tse99] durchgeführt, um das lokale Minimum an der mit PSO ermittelten Stelle im Parameterraum besser anzunähern. Dieser Datensatz wurde weiterverwendet, um ein Segmentierungsverfahren für die Hand, basierend auf Random Decision Forests, zu trainieren als auch einen Schätzer für die Handgelenkpositionen, basierend auf Convolutional Neural Networks (CNNs). Anstelle des Handmodells aus geometrischen Formen wurde ein parametrisches, realistisch anmutendes Handmodell [Šar11] genutzt. Dieses modelliert die Hand ursprünglich durch mehr als 70.000 Polygone. Tompson et al. reduzierten die Polygonanzahl auf knapp über 3.300, um schneller rendern zu können, wofür eine Textur aus Hautfarbe genutzt wird. Für den Vergleich des gerenderten Bildes mit dem aufgenommenen Bild wurden drei Kameraansichten genutzt, um Verdeckungen entgegenzuwirken. Das Handsegmentierungsverfahren arbeitet mit einem Random Decision Forest zum pixelweisen Segmentieren der Hand, ähnlich wie im oben beschriebenen Verfahren zur Körperposenschätzung von Shotton et al. [Sho11]. Tompson et al. waren die ersten, die CNNs zur Extraktion von Gelenkpositionen auf Tiefenbildern durch das Generieren sog. Heatmap-Merkmalsbilder nutzten. Der Grundgedanke bei diesen Merkmalsbildern ist, dass ein CNN aus dem gegebenen Eingangsbild, egal ob RGB- oder Tiefenbild, ein Bild erzeugt, das an der 2D-Position des

Gelenks eine zweidimensionale Gauß-Verteilung schätzt, die die ungefähre Position des Gelenks beschreibt. Aus den berechneten 2D-Gelenkpositionen können über die Tiefeninformation 3D-Gelenkpositionen ermittelt werden, welche in einem Energiefunktional mit denen des parametrisierten Handmodells verglichen werden. Ein erweitertes PSO-Verfahren [Yas10] wird genutzt, um die Parameter zu bestimmen. Die gesamte Berechnung benötigte 25 ms auf einem Vierkern-Intel-Prozessor und einer Nvidia GTX 580 GPU im Jahr 2014.

Ein Forscherteam bei Microsoft Research um Toby Sharp, Jamie Shotton und Andrew Fitzgibbon, die zuvor am artikularen Körperposentracking für die Kinect [Sho11] mitgearbeitet haben und derweil im HoloLens Team angesiedelt sind, zeigen in Sharp et al. [Sha15] den *Microsoft Research Synthetic Hand* Datensatz sowie ein Verfahren zur Handposenschätzung. Diese bestimmt in einem ersten Schritt die Körperpose mit dem Verfahren von Shotton et al. [Sho11]. Die Handregion wird nicht näher erläutert aus der Handposition des Skeletts abgeleitet. Weiter definieren sie sechs Basisposen, aus denen für das Training 100.000 Posen generiert wurden. Aus den Posen wurden anschließend synthetische Trainingsdaten erzeugt, in denen die Hände eine größere Entfernungen zur Kamera aufweisen als bei Tompson et al. [Tom14]. Die Bestimmung der Handpose wird über einen zweistufigen hierarchischen Prozess durchgeführt. Die erste Stufe dient der Bestimmung der Rotation der Hand. Die zweite Stufe enthält für 128 verschiedene Rotationen jeweils drei Schätzer zum Verbessern der Rotation, der Translation und zum Schätzen der Basispose. Anschließend wird mit PSO, ähnlich wie bei Oikonomidis et al. [Oik11], der Parametersuchraum weiter durchsucht, wobei die Veränderung der Parameter sich hauptsächlich auf die Finger konzentriert. Das Verfahren erreicht eine Rate von 30 Hz auf einer Nvidia GTX Titan Grafikkarte.

Tang et al. [Tan15] nutzen zur Bestimmung der Handpose ähnlich wie Sharp et al. [Sha15] ein hierarchisches Modell aus Schätzern zur Bestimmung von Untermengen der Posenparameter. Ihre Hierarchie berücksichtigt den kinematischen Aufbau der Hand und besitzt fünf Stufen, die bei der Handwurzel

beginnen und an den Fingerspitzen enden. Zusätzlich nutzen sie eine Zwischenbewertung generierter Hypothesen. Diese vergleicht generierte Gelenkpositionen mit den beobachteten Tiefenwerten, welche für eine gute Bewertung entsprechend klein sein sollen. Zusätzlich wird die 2D-Position der 3D-Gelenkposition in der Handregion bestimmt und eine Distanz zum Umriss der Hand berechnet, welche für in der Hand liegende Punkte den Wert Null und für außerhalb liegende einen positiven Wert annimmt. Mit dieser Art der Zwischenbewertung werden in jeder Ebene der Hierarchie eine Vielzahl von Zwischenhypothesen verworfen und der Ausgaberaum effizient durchsucht. Es ergibt sich eine bestimmte Anzahl an vollständigen Hypothesen. Für jede wird mit Hilfe eines parametrisierten Handmodells ein Tiefenbild gerendert, welches mit dem beobachteten Tiefenbild verglichen wird, um die beste Hypothese zu ermitteln.

Ye et al. [Ye16] erweitern den hierarchischen Ansatz von Tang et al. [Tan15] um die Vorverarbeitung der Eingabe jeder Ebene der Hierarchie. Nach der Bestimmung der Hypothesen jeder Ebene wird die Eingabe, die Handregion im Tiefenbild, durch die Berechnung einer ganz bestimmten Rotation und räumlichen Transformation für die Berechnungen auf der nächsten Ebene vorbereitet. Diese Normierung der Eingabe verkleinert den Raum aller möglichen Eingaben weiter.

Durch die Etablierung des Datensatzes *HIM2017* durch Yuan et al. [Yua17a] zur 3D Handposenschätzung auf Tiefenbildern wurden viele weitere Verfahren entwickelt, wie die Übersicht von Yuan et al. [Yua18] zu Ergebnissen auf HIM2017 zeigt, die sich mit dieser Thematik beschäftigen. Zu erkennen ist der Trend zu Verfahren, die komplett aus neuronalen Netzen bestehen und keine inverse Kinematik oder iterative Verfahren wie PSO nutzen.

Mueller et al. [Mue17] erzeugen ein auf ResNet50 [He15] basierendes CNN, welches sie HALNet (HAnd Localization Net) nennen, um die Region der Hand zu bestimmen. Als Eingabe bekommt es ein eingefärbtes Tiefenbild. Ein weiteres auf ResNet50 basierendes CNN berechnet die 3D-Handpose.

Moon et al. [Moo17] lieferten längere Zeit mit dem *V2V-PoseNet* das beste Verfahren auf HIM2017, welches auf Voxeln arbeitet, um das Problem der Verzerrung von Tiefenbildern zu vermeiden, und intern dreidimensionale CNNs nutzt.

Zhou et al. [Zho18] stellten erst kürzlich ein CNN vor, welches aus drei Zweigen bestehend die Handpose direkt berechnet. Während einige der oben vorgestellten Verfahren einen hierarchischen Ansatz wählen, der ab der zweiten Ebene alle Finger einbezieht und sich Ebene für Ebene zu den Fingerspitzen vorarbeitet, gruppieren Zhou et al. die Finger in drei Gruppen: Daumen, Zeigefinger, restliche Finger. Sie behaupten, diese Gruppierung sei sinnvoll, weil Daumen und Zeigefinger alleine für bestimmte Gesten genutzt werden können und die anderen drei Finger häufig aufgrund der Muskelstruktur zusammen bewegt werden. Jede dieser Gruppen wird von einem relativ kleinen CNN gehandhabt, das darauf trainiert wurde, nur seine zugehörigen Finger zu berechnen. Die resultierenden Merkmale werden konkateniert und in eine weitere Regressionsschicht gefüttert, die die gesamte Handpose berechnet. Verglichen mit dem V2V-PoseNet von Moon et al. [Moo17] besitzt das gesamte CNN deutlich weniger Parameter und ist deshalb mit weniger Daten in schnellerer Zeit derart trainierbar, dass es auf dem *SEEN*-Teil von HIM2017 die niedrigsten durchschnittlichen Abweichungen von knapp über fünf Millimetern erzielt und dabei mit über 600 Hz ca. 20 mal so schnell inferiert wie das V2V-PoseNet.

Bevor in Abschnitt 2.3.3 die Arten der Handregionsbestimmung zusammengefasst werden, erfolgt im nachfolgenden Abschnitt zunächst die Betrachtung der Verfahren zur monokularen Handregionsbestimmung und Posenschätzung.

2.3.2 Handregionsbestimmung und Posenschätzung auf monokularen Farbbildern

Wie oben bereits erwähnt ist die Segmentierung der Handregion ohne Tiefeninformation deutlich komplexer. In stationären Anwendungen zeigten Bader et al. [Bad09, Bad11] eine Gestenerkennung für statische Handposen. Eine

aktive Infrarot-Quelle beleuchtet dabei die Szene, welche mit einem Sensor mit Tageslichtsperrfilter erfasst wird. Durch die Infrarot-Beleuchtung und den statischen Hintergrund konnte die Hand durch eine Subtraktion eines eingelernten Hintergrundes nahezu perfekt segmentiert und ihre Kontur bestimmt werden. Über Konturmerkmale wurde eine Support Vector Machine (SVM) trainiert, um eine kleine Menge von statischen Handposen zu erkennen, welche in Anwendungen zur Interaktion auf großflächigen Oberflächen genutzt werden können, ohne dass diese berührungssensitiv sind [Mai12, Sch13]. Da eine Hintergrundsubtraktion bei sich änderndem Hintergrund nicht sinnvoll ist, um Vordergrund (der die Hand beinhaltet) und Hintergrund zu bestimmen, und eine Infrarot-Beleuchtung die Anwendung an den Innenraum bindet sowie zu sperrig ist, um getragen werden zu können, bietet sich dieser Ansatz nicht für mobile Anwendungen an.

Das Problem der Segmentierung wird von Wang und Popović [Wan09] mit einem texturierten Handschuh gelöst, mit dessen Hilfe sie sogar eine 3D-Handpose schätzen können. Mistry und Maes [Mis09] markierten die Fingerspitzen mit Markern, um sie robust erkennen zu können und die Hand für die Interaktion nutzen zu können.

Verfahren, die keine Objekte an den Händen anbringen, um die Segmentierung der Hand zu erleichtern, nutzten im Wesentlichen zwei Hinweise: Hautfarbe und Bewegungsinformation [Wac11]. Die Erkennung von Hautfarbe war elementar für erste Verfahren zur Gesichtslokalisierung, welche parametrische Modelle oder Histogramme zur Repräsentation von Hautfarbe nutzen, siehe Kakumanu et al. [Kak07] sowie Phung et al. [Phu05]. Bewegungsinformation wurde in statischen Anwendungen, aber auch mobilen Anwendungen durch die Subtraktion von aufeinanderfolgenden Einzelbildern wie bei Spruyt et al. [Spr10] oder durch die Berechnung des optischen Flusses ermittelt [Köl04]. Dynamische Wischgesten, bei denen nur die Trajektorie der Hand eine Rolle spielt, werden mit Hidden Markov Modellen oder Dynamic Time Warping erkannt [Kan04].

Ren und Gu [Ren10] erstellten auf dem *Intel Egocentric Object Recognition* Datensatz [Ren09] ein System, das aus der Egoperspektive mit Hilfe dichten optischen Flusses und Segmentierung desselben in Ebenen sowie Vorannahmen

über Objektpositionen und Farbinformationen eine Segmentierung vornimmt und zur Genauigkeit bei der Erkennung alltäglich genutzter Objekte des Datensatzes auf SIFT-Merkmalen [Low04] oder HOG-Merkmalen [Dal05] trainierte Klassifikatoren nutzt.

Fathi et al. [Fat11] erweiterten den Ansatz von Ren und Gu [Ren10]. Ihr Ansatz führt zuerst eine Segmentierung in Vorder- und Hintergrund aus. Dazu wird aus einer Menge von aufeinanderfolgenden Einzelbildern mit Hilfe dichten optischen Flusses ein Panorama erzeugt, das für diese Menge von Bildern den als statisch angenommenen Hintergrund beschreibt. Zusätzlich wird für dieses Panorama eine Kantenbestimmung durchgeführt. Jedes Einzelbild wird mit dem Panorama registriert und in Superpixel [Ren03] unterteilt. Dann wird ein Hintergrundmodell basierend auf Merkmalen wie Textur und Farbe gelernt, um eine Wahrscheinlichkeit für die Zugehörigkeit eines Superpixels zum Hintergrund schätzen zu können. Die Übereinstimmung der Kanten eines Superpixels mit den Kanten des Panorama-Kantenmodells liefert eine weitere Wahrscheinlichkeit für die Zugehörigkeit zum Hintergrund. Diese Wahrscheinlichkeiten werden sowohl räumlich als auch zeitlich in einem Markov Random Field (MRF) probabilistisch modelliert und das MRF mit der Methode des minimalen Netzwerkschnittes berechnet, um die Segmentierung in Hintergrund und Vordergrund zu erhalten.

Li und Kitani [Li13a] entwickelten ein Verfahren zur pixelweisen Hautfarbenerkennung, das gegenüber vorherigen Ansätzen zur Hautfarbenerkennung [Phu05, Kak07] folgende Innovationen besitzt: Die Merkmale zur Klassifikation bestehen nicht nur aus dem Farbwert, sondern werden um Texturmerkmale aus der lokalen Bildregion eines Pixels berechnet. Als Klassifikator wird ein Random Decision Forest anstelle von Histogrammen oder Gauß-Mischverteilungen genutzt. Durch die zusätzlichen Texturmerkmale sollten hautfarbige Bereiche von hölzernen Bereichen unterschieden werden können. Als weitere Innovation werden für verschiedene Umgebungsbedingungen verschiedene Klassifikatoren trainiert, die in ihrer jeweiligen Umgebung besser klassifizieren können. Von diesen wird zur Laufzeit nur eine Untermenge für die Klassifikation herangezogen. Diese Untermenge besteht

aus den Klassifikatoren mit den zur aktuellen Umgebung ähnlichsten Bedingungen. Zur Bestimmung dieser Ähnlichkeit werden aus den einzelnen Farbkanälen des Eingabebildes Histogramme gebildet und diese mit den Histogrammen der gelernten Umgebungen verglichen. Für jedes Pixel wird mit den gewählten Klassifikatoren eine Wahrscheinlichkeit für die Zugehörigkeit zur Klasse Hautfarbe berechnet. Die entstehende zweidimensionale Wahrscheinlichkeitsverteilung wird anschließend als Grauwertbild interpretiert und mit der Methode von Otsu [Ots79b] binarisiert.

Der Ansatz von Bambach et al. [Bam15] zielt darauf ab, sowohl die eigenen Hände als auch die Hände eines Gegenüber unterscheiden und segmentieren zu können. Nach ihrer Definition geht eine Hand bis zum Handgelenk und nicht wie z. B. bei Li und Kitani [Li13a] über das Handgelenk hinaus, wo auch der Unterarm in die Segmentierung eingeschlossen ist. Ihr Ansatz funktioniert ähnlich wie bei Objekterkennern: Aus dem Bild werden Kandidatenfenster bestimmt, z. B. durch einen *Sliding-Window*-Ansatz, welche einzeln vom Objekterkenner klassifiziert werden. Das Kandidatenfenster mit dem besten Klassifikationsergebnis bestimmt die Position des Objektes. Da der Sliding-Window-Ansatz das gesamte Bild abfährt, entsteht eine sehr hohe Anzahl an Kandidatenfenstern. Hier wirken Bambach et al. entgegen, indem sie Kandidatenfenster zufällig bestimmen und die Position und Größe eines Fensters durch eine Verteilung, welche das Vorkommen einer Hand bzgl. Position und Größe modelliert, bewerten. Diese Verteilung kann aus der Grundwahrheit berechnet werden. Hierbei nutzen sie folglich Eigenschaften der Handpositionen und -größe aus, die bei Aufnahmen aus der Egoperspektive entstehen. Die Farbe des zentralen Pixels eines Kandidatenfensters wird zudem mit einem nicht-parametrischen Hautfarbenmodell bewertet. Kandidatenfenster mit einer hohen Bewertung werden folglich öfter klassifiziert. Zu bemerken ist, dass die Abdeckung des Generierungsverfahrens von Kandidatenfenstern eine obere Schranke für die korrekte Erkennung der Hand zur Folge hat. Als Objektklassifikator nutzen sie das CNN *CaffeNet* [Jia14]. Nach der Bestimmung der besten Kandidatenfenster für die jeweilige Hand wird in jedem Fenster eine initiale Segmentierung der Hand über das nicht-parametrische Hautfarbenmodell durchgeführt, welche anschließend mit dem auf Markov

Random Fields basierenden Segmentierungsverfahren *GrabCut* [Rot04] verfeinert wird.

Auch im Bereich der Körperposenschätzung konnten durch CNNs große Verbesserungen erzielt werden. Wei et al. [Wei16] trainierten ein CNN, das aus dem Eingabebild mit einer zentrierten Person für jedes Gelenk, ähnlich wie bei Tompson et al. [Tom14], eine Heatmap generiert, aus der die Position des Gelenks ermittelt werden kann. Alle Gelenkpositionen zusammen ergeben dann das Skelett der Person. Das Netz lernt die Relation der Gelenkpositionen zueinander. Für den Input des CNNs wird ein Personenklassifikator, ebenfalls ein CNN, trainiert, der die Fenster eines Sliding-Window-Ansatzes bewertet.

Cao et al. [Cao17, Cao18] erweiterten diesen Ansatz um *Part Affinity Fields*. Mit diesen werden Entfernungsbeziehungen von Gelenken zueinander modelliert und im Bild gefundene Gelenke können dadurch unterschiedlichen Personen zugeordnet werden. Ein Vorteil ist, dass dadurch kein Personenerkenner wie bei Wei et al. [Wei16] benötigt wird und die Laufzeit nicht proportional zur Anzahl gefundener Personen ansteigt. In dem Verfahren wurde das Körperskelett zusätzlich um die Position der Nase, Augen und Ohren erweitert. Es steht unter dem Namen *OpenPose* [Cao18] für die Forschung öffentlich als Software zur Verfügung.

Simon et al. [Sim17] übertrugen den Ansatz von Wei et al. [Wei16] auf die 2D-Handposenerkennung mit einer abgewandelten Netzarchitektur. Ihr Verfahren ist als zusätzliches Modul in der Software OpenPose enthalten und erwartet als Input die Handregion und Angabe, ob eine linke oder rechte Hand darin enthalten ist. Für die Bestimmung der Handregion nutzen Simon et al. die Körperposenerkennung von OpenPose [Cao18] auf folgende Weise: Aus den Positionen des Ellbogen- und Handgelenks ergibt sich der Unterarm, welcher um 15 % seiner Länge über das Handgelenk hinaus verlängert wird, um die Position der Hand zu schätzen. Während des Trainings bildet das Verfahren ein alle Gelenkpositionen der Hand umschließendes Rechteck. Sei die längste Seitenlänge dieses Rechtecks b, so wird ein Quadrat der Seitenlänge $2{,}2b$ um dieses Rechteck gelegt, das die Handregion bildet. Während der Laufzeit wird dieses Quadrat in Abhängigkeit von der Kopfhöhe bestimmt. Da OpenPose die Hände aus der Egoperspektive nicht findet, weil der Großteil

des Körpers nicht im Bild zu sehen ist, funktioniert das Verfahren für diesen Anwendungsfall nur unter Nutzung eines weiteren Verfahrens zur Handregionsbestimmung. Für die Detektion der Gelenkpositionen der Hand wird ein CNN basierend auf *VGG-19* von Simonyan und Zisserman [Sim14] genutzt.

Zimmermann und Brox [Zim17] präsentierten ein Verfahren zur Schätzung von 3D-Handposen auf monokularen Farbbildern. Es besteht in einem ersten Schritt aus einem *HandSegNet* benannten CNN zur Handregionsbestimmung, das einen ähnlichen Aufbau besitzt wie der Personenerkenner von Wei et al. [Wei16] und als Ausgabe eine Segmentierung der im Bild vorkommenden Hände hat. Darauf aufbauend wird ein CNN zur Bestimmung der Gelenkpositionen genutzt, das ähnlich zu Tompson et al. [Tom14] und Wei et al. [Wei16] Heatmaps pro Gelenk bestimmt. Diese Heatmaps sind Eingabe für ein CNN, das eine normalisierte 3D-Handpose bestimmt.

Donoso et al. [Gom17] zeigten ein Verfahren zur Schätzung von 2D-Handposen auf monokularen Farbbildern. Zuerst wird mit einem CNN die Handregion bestimmt. Hierfür nutzen sie das Region Proposal Network *Faster R-CNN* von Ren et al. [Ren15], welches sie auf Nahaufnahmen von Händen trainierten. Anschließend wird ein CNN genutzt, um die Gelenkpositionen in Form von Heatmaps wie bei anderen Verfahren [Tom14, Sim17, Zim17] zu berechnen. Hierfür wird ein abgewandeltes ResNet50 [He15] trainiert. Die Regression der Pose auf Basis der Heatmaps in einem letzten Schritt wird ebenfalls über ein abgewandeltes ResNet50 realisiert.

Panteleris et al. [Pan18] stellten ein Verfahren zur Bestimmung von 3D-Handposen vor, welches die Handregionsbestimmung mit einem zur Erkennung von Händen trainierten *YOLO v2* [Red17] Objektdetektor durchführt. Dieser wurde gleichzeitig als Kopfdetektor trainiert. Zur Erstellung der Trainingsbilder für die Handregionserkennung nutzten sie den Körperposenschätzer von OpenPose [Cao18]. In ihrem Ansatz verwenden sie den 2D-Handposenschätzer aus OpenPose von Simon et al. [Sim17] für die Berechnung einer 2D-Handpose. Für die Berechnung der 3D-Koordinaten der Gelenke wird ein Energiefunktional gebildet, welches das Problem als inverses kinematisches Problem beschreibt. Aus den 27 Parametern für ein Handmodell können die

3D-Positionen berechnet werden. Nach einer Projektion derselben in das Kamerabild, sollen diese auf den vorher berechneten 2D-Gelenkpositionen zu Liegen kommen. Die Parameter für das Handmodell werden über die Minimierung des Energiefunktionals bestimmt. Panteleris et al. verglichen ihr Verfahren mit dem von Zimmermann und Brox [Zim17] und erzielten auf verschiedenen Datensätzen bessere Ergebnisse.

Mueller et al. [Mue18] präsentierten ein Verfahren zur 3D-Handposenschätzung auf monokularen Farbbildern, das zu Teilen auf ihrem auf Tiefendaten arbeitenden Verfahren [Mue17] basiert. Dieses nutzt ein CNN, *RegNet* genannt, das auf einer Handregion sowohl die 2D-Heatmaps als auch die 3D-Positionen der Gelenke der Hand berechnet. In diese Schätzungen wird ein Skelett der Hand eingepasst, welches nur plausible Handposen zulässt. Die benutzte Energiefunktion wird mit einem Gradientenabstiegsverfahren minimiert und beinhaltet u. a. auch einen Term für die Glattheit der Parameter der Handpose zur vorangegangenen Schätzung. Die Handregion wird initial auf das gesamte Bild gesetzt und mit einem Filter [Cas12] die Handposition im nächsten Einzelbild geschätzt. Mueller et al. verglichen ihr Verfahren u. a. mit dem von Zimmermann und Brox [Zim17] und erzielten auf dem *Stereo Handpose Dataset (SHD)* [Zha16] sowie *EgoDexter* Datensatz bessere Ergebnisse. Das Verfahren von Zimmermann und Brox [Zim17] hatte auf SHD zuvor die besten Ergebnisse erzielt. Für den Vergleich der 2D-Handpose auf EgoDexter gibt es nur die Ergebnisse von Mueller et al. [Mue18] und Zimmermann und Brox [Zim17], da dieser Datensatz vergleichsweise neu ist.

Dibra et al. [Dib18] zeigten ein weiteres Verfahren für die 3D-Handposenschätzung auf monokularen RGB-Daten. Für die Handregionsbestimmung wird in ihrem Verfahren *Faster R-CNN* von Ren et al. [Ren15] genutzt und anschließend die Hand mit einem adaptierten SegNet [Bad15] segmentiert, welches sie HandSegNet von Zimmermann und Brox [Zim17] gegenüberstellen. Basierend auf dieser Segmentierung wird direkt die Handpose geschätzt. Dibra et al. verglichen sich mit mehreren Verfahren auf dem *Stereo Handpose Dataset* (SHD) [Zha16]. Im Vergleich zu Zimmermann und Brox [Zim17] schneiden sie dabei schlechter ab.

Iqbal et al. [Iqb18] stellten ein Verfahren zur Schätzung der 3D-Handpose auf monokularen Farbbildern mit einem CNN vor, das implizit Tiefenbilder der Hand und Heatmaps der Gelenke erzeugt. In ihrem Verfahren verwenden sie eine 2,5D-Darstellung der Gelenkpositionen, aus der sie nach Schätzung eines Skalierungsfaktors die resultierenden 3D-Gelenkpositionen berechnen können. Der Vorteil ihrer 2,5D-Darstellung ist die Differenzierbarkeit, so dass die Heatmaps als latente Variablen vom CNN gelernt werden. Ein visueller Vergleich in Abbildung 2 von [Iqb18] mit wie sonst üblich manuell erzeugten Heatmaps zeigt deutlich die akkuratere Beschreibung der Gelenkposition durch Verteilungen, welche sich der Form der Hand im Bereich einer Gelenkposition besser anpassen. Die Handregionsbestimmung wird wie bei Panteleris et al. [Pan18] mit *YOLO v2* [Red17] im ersten Einzelbild und danach aus der vorangegangenen Pose bestimmt. Das Verfahren wurde mit dem Stand der Technik verglichen und zeigt die bis dato (Februar 2019) besten Resultate auf den Datensätzen *Stereo Handpose Dataset (SHD)* [Zha16] und *EgoDexter* und übertrifft damit die oben vorgestellten Verfahren von Zimmermann und Brox [Zim17], Panteleris et al. [Pan18] sowie Mueller et al. [Mue18] sowohl bei der 2D- als auch 3D-Handposenschätzung.

2.3.3 Zusammenfassung Handregionsbestimmung

Nach Betrachtung der obigen Verfahren zur Handposenschätzung sind folgende Ansätze zur Handregionsbestimmung zu finden:

- Die Körperpose wird mit dem Verfahren von Shotton et al. [Sho11] auf Basis von Tiefenbildern geschätzt und aus der groben Handposition und dem Tiefenbild eine Segmentierung der Hand erzeugt [Sha15].
- Es wird ein Handerkenner auf Tiefenbildern nach dem Vorbild von Shotton et al. [Sho11] gebaut [Tom14].
- Es wird ein Hautfarbenerkenner genutzt, um im Farbbild des Sensors die Hand zu segmentieren. Das Tiefenbild wird für die weitere Segmentierung genutzt [Oik11].

- Die Hand wird als nächstes Objekt zur frontal angebrachten Kamera angenommen [Obe17].
- Die Handregion wird als gegeben vorausgesetzt [Tan15, Ye16, Yua17a, Moo17, Wu18, Zho18].
- Die Hand wird mit einem Segmentierungsverfahren segmentiert [Wac11, Ren10, Fat11, Li13a, Zim17, Dib18].
- Es wird ein aktuelles Objektlokalisationsverfahren wie ResNet50 [He15], *Faster R-CNN* [Ren15] oder *YOLO v2* [Red17] für die Handlokalisierung trainiert [Gom17, Mue17, Pan18, Iqb18].
- Nach einer initialen Schätzung wird die Position der Handregion aus der vorangegangenen Pose abgeleitet [Mue18, Iqb18].

Das Problem der Handregionsbestimmung ist auf Tiefendaten einfacher anzugehen, da die Tiefeninformation eine Segmentierung der Szene auf einfache Weise ermöglicht. Verfahren, die keine Tiefenbilder verwenden, nutzten Hautfarbenerkenner und Bewegungsinformation. Modernere Verfahren nutzen unterschiedliche CNNs für die Segmentierung oder aktuelle, auf Hände trainierte Objekterkenner. Einige Verfahren nutzen die vorangegangene Handpose zu Schätzung der Handregion im nächsten Einzelbild. In Kapitel 4 wird ein in der vorliegenden Arbeit erstelltes Verfahren zur Bestimmung der Handregion vorgestellt, welches mit obigen Verfahren zur Handposenschätzung kombiniert werden kann und diese deutlich verbessert. Der Vollständigkeit halber folgen zwei Abschnitte über Datensätze für diesen Forschungsbereich.

2.3.4 Datensätze mit Tiefenbildern

Tang et al. [Tan14, Tan17] erstellten den Datensatz *Imperial College Vision Lab (ICVL)*. Dabei führten zehn verschiedene Personen mehr als 26 verschiedene Handposen mit frontaler Sicht eines Tiefensensors auf die Hand durch. Pro Sekunde wurden drei Tiefenbilder aus den Aufnahmen extrahiert und es entstanden 20.000 Tiefenbilder, deren Anzahl durch Rotation auf mehr als 180.000 Tiefenbilder erhöht wurde.

Der Datensatz *NYU Hand Pose* wurde von Tompson et al. [Tom14] erstellt. Er enthält 72.757 Farb- und Tiefenbilder mit annotierten Handposen aus drei Ansichten (Frontalansicht und zwei Seitenansichten). Als Sensor wurde die Microsoft Kinect genutzt. Zum Testen enthält der Datensatz zusätzlich 8.252 Testdaten. Die geschätzten 2D-Gelenkpositionen des genutzten CNNs sind ebenfalls verfügbar. Zusätzlich sind 6.736 Tiefenbilder auf Pixelebene bzgl. der Klassen „Hand“ und „nicht Hand“ annotiert.

Für den Datensatz *Microsoft Research Synthetic Hand* [Sha15] wurden sechs Basisposen gewählt. Aus diesen wurden 100.000 Posen generiert und synthetische Trainingsdaten erzeugt. Im Vergleich zum NYU Hand Pose Dataset von Tompson et al. [Tom14] weist der Microsoft Datensatz größere Entfernungen zwischen Sensor und Hand auf.

Zhang et al. [Zha16] konstruierten den Datensatz *Stereo Handpose Dataset* (SHD). Er besteht aus zwölf Sequenzen mit je 1.500 Einzelbildern, also zusammen 18.000 Einzelbildern. Für jedes Einzelbild gibt es sowohl das Tiefen- als auch Farbbild und die 3D-Handpose.

Yuan et al. [Yua17b] stellten den Datensatz *BigHand2.2M* vor. Dieser enthält 2,2 Millionen Tiefenbilder und zugehörige 3D-Gelenkpositionen von zehn Testpersonen, die jeweils über zwei Stunden aufgenommen wurden. Dafür wurden sechs Magnetsensoren auf Fingernägeln und Handrücken angebracht und über inverse Kinematik auf einem Handmodell mit 31 Freiheitsgraden die 21 Gelenkpositionen berechnet. Der Datensatz beinhaltet 290.000 Einzelbilder aus der Egoperspektive. Im Vergleich zu den NYU und ICVL Datensätzen weist BigHand2.2M deutlich mehr Varianz bei Kamera- und Handpose auf. Yuan et al. konnten auch zeigen, dass das *Holi*-Verfahren von Ye et al. [Ye16] auf dem Datensatz BigHand2.2M trainiert auf ICVL und NYU bessere Ergebnisse erzielt, als wenn es auf einem der beiden Datensätze alleine trainiert wurde.

Garcia-Hernando et al. [Gar18] erstellten den Datensatz *First-Person Hand Action Dataset* (FHAD) mit über 100.000 Tiefen- und Farbbildern, aufgenommen aus der Sicht eines auf der Schulter angebrachten Tiefensensors. Die Daten zeigen 45 verschiedene alltägliche Aktivitäten mit 26 Objekten durchgeführt

von sechs Testpersonen. Für die Annotation der 21 Gelenkpositionen des verwendeten Handmodells wurden sechs magnetische Tracker und inverse Kinematik ähnlich wie bei Yuan et al. [Yua17b] benutzt.

Yuan et al. [Yua17a] stellten den Datensatz *Hands in the Million Challenge on 3D Hand Pose Estimation* (HIM2017) vor [Yua17a]. Er kombiniert die Datensätze BigHand2.2M und FHAD. Von BigHand2.2M enthält er 99 Segmente mit 270 bis 330 aufeinanderfolgenden Einzelbildern und ein paar kurze Sequenzen mit 150 aufeinanderfolgenden Einzelbildern von FHAD. Für die Aufgabe der Handposenschätzung gibt es 295.000 Frames insgesamt. Der Datensatz beinhaltete mehrere Hände gleichzeitig, verschiedene Ansichten, Handposen und Verdeckungen der Hände durch Objekte. Einige Daten sind nicht aus der Egoperspektive aufgenommen, sondern aus frontaler Sicht. Bei Letzteren ist die Testperson manchmal zu sehen und manchmal nicht. Die zwei gestellten Herausforderungen sind die Handposenschätzung bei gegebener initialer Pose und die Handposenerkennung auf einzelnen Bildern. Zusätzlich kommt der Datensatz mit einer Evaluationsmethodik.

Zimmermann und Brox [Zim17] nutzten zum Trainieren ihres oben vorgestellten Verfahrens den extra dafür erstellten Datensatz *Rendered Handpose*. Er besteht aus 41.258 Training- und 2.728 Testdaten. Jedes Testdatum besteht aus einem RGB- und Tiefenbild sowie einer Segmentierungsmaske mit Klassen für den Hintergrund, die Handfläche sowie 15 weiteren Klassen für die Fingerglieder und 21 2D-Gelenkpositionen im RGB-Bild. Alle Daten sind synthetisch erstellt.

Der Datensatz *Large-scale Multiview 3D Hand Pose* wurde von Gomez et al. [Gom17] vorgestellt. Er beinhaltet insgesamt 20.500 Einzelbilder in 21 Sequenzen, die nicht aus der Egoperspektive aufgenommen wurden. Für jedes Einzelbild gibt es vier Kameraansichten, wodurch sich mehr als 80.000 Farbbilder ergeben. Die Grundwahrheit wurde mit der Leap Motion erstellt. Drei Aufgaben werden gestellt: Die Bestimmung der Handregion, die Schätzung der 2D- und der 3D-Handpose. Weiter stellen sie für die Ergebnisse der 2D-Posenschätzung die Ergebnisse eines Basisverfahrens vor. Bisher (Februar 2019) wurden keine weiteren Ergebnisse auf diesem Datensatz präsentiert.

Mueller et al. [Mue17] stellten den Datensatz *SynthHands* vor. Er enthält 220.000 Farbbilder aus der Egoperspektive mit 63.530 annotierten Einzelbildern. Für das Handmodell wurden zwölf verschiedene Haut-Texturen genutzt. Für den Hintergrund wurden zufällig 10.000 echte Bilder genutzt. In den öffentlichen Daten ist der Hintergrund allerdings nicht enthalten. Zusätzlich präsentierten Mueller et al. den Datensatz *EgoDexter* [Mue17]. Er besteht aus 3.190 Farb- und Tiefenbildern bestehend aus vier Sequenzen, die natürliche Handinteraktionen mit Objekten von vier verschiedenen Personen in komplexen Umgebungen aus der Egoperspektive teils in Bewegung zeigen. 1.485 Einzelbilder sind bzgl. der Fingerspitzen sowohl in 2D als auch 3D annotiert. Die Handpose wurde dazu manuell im Tiefenbild annotiert und die 3D-Position eines Gelenks über das Tiefenbild ermittelt. Verdeckte Fingerspitzen wurden nicht annotiert.

2.3.5 Datensätze ohne Tiefenbilder

Ren und Philipose [Ren09] erstellten den Datensatz *Intel Egocentric Object Recognition* zur Erkennung von 42 Objekten des täglichen Gebrauchs aus der Egoperspektive. Die Farbkamera wurde dafür auf der linken Schulter der Personen angebracht und auf den Bereich vor der Person ausgerichtet, um mit den Händen manipulierte Objekte erfassen zu können. Insgesamt wurden die Daten von zwei Personen in fünf unterschiedlichen Umgebungen mit unterschiedlichen Beleuchtungsbedingungen aufgenommen. Es entstand Datenmaterial von zwei Stunden Länge mit 100.000 Farbbildern, von denen 70 % zu erkennende Objekte beinhalten. 420 Bilder wurden bzgl. Objekten und Hintergrund segmentiert sowie 40 bzgl. Händen und Hintergrund. Für die Erkennung der Objekte wurden SIFT-Merkmale und eine Support Vector Machine (SVM) als Klassifikator genutzt. Hauptprobleme bei der Erkennung der Objekte sind den Autoren zufolge Bewegungsunschärfe und Verdeckung der Objekte durch die Hände der Person.

Fathi et al. [Fat11] erstellten den Datensatz *Georgia Tech Egocentric Activity* (GTEA), der sieben tägliche Aktivitäten aus der Egoperspektive, durchgeführt

von vier Personen, beinhaltet. Die Farbkamera wurde dafür auf einer Baseball-Mütze angebracht. Von den mit 30 Hz in 720p aufgenommenen Daten wurden 31.222 Einzelbilder erstellt. Die Grundwahrheit beinhaltet pro Einzelbild die Information der vorkommenden Objekte. Die Hände zählen hier nicht zu den Objekten. Ziel ihres entwickelten Verfahrens (siehe Abschnitt 2.3.2) ist die Erkennung der Aktivitäten und ist eine Weiterentwicklung des Verfahrens von Ren und Gu [Ren10].

Pirsiavash und Ramanan [Pir12] erstellten den Datensatz *Activities of Daily Living (ADL)*. Er beinhaltet Aufnahmen von alltäglich vorkommenden Aktivitäten aus der Sicht einer von 20 verschiedenen Personen in 20 verschiedenen Umgebungen auf Brusthöhe getragenen Kamera. Eine Million Farbbilder wurden in Form von Videos mit einer Gesamtlänge von zehn Stunden aufgenommen. Für einige Objekte sind ihre sie umgebenden Rechtecke annotiert, für die Hände nicht. Der Datensatz ist gedacht für die Erkennung alltäglicher Aktivitäten.

Li und Kitani [Li13a] erstellten einen Datensatz aus 600 auf Pixelebene annotierten Bildern mit Händen aus der Egoperspektive. Nicht nur die Hände, sondern auch die Arme wurden annotiert. Das Verfahren, welches zur Detektion von Händen in Bildern genutzt wird, entspricht aufgrund der vorgenommenen Annotation einem Verfahren zur Erkennung von Hautfarben. Die Videos beinhalten keinerlei soziale Interaktion.

Bambach et al. [Bam15] stellten den Datensatz *EgoHands* vor. Dieser enthält 48 Sequenzen mit ca. 130.000 Farbbildern aus der Egoperspektive in 720p Auflösung, von denen 4.800 annotiert sind. Jeweils ist die eigene linke, die eigene rechte Hand und die linke und rechte Hand des Gegenübers als Segmentierung annotiert. Die Videos stellen jeweils zwei Personen dar, die an verschiedenen Tischen sitzen und eine von vier sozialen Interaktionen wie z. B. Karten- oder Schachspielen durchführen. Ihr entwickeltes Verfahren zur Segmentierung und Unterscheidung der vorkommenden Hände wird in Abschnitt 2.3.2 beschrieben.

Ein weiterer Datensatz für Bilder aus der Egoperspektive heißt *Epic Kitchen* und stammt von Damen et al. [Dam18]. Er beinhaltet Videos von 32 Teilnehmern bei täglichen Arbeiten in über den Globus verteilten Küchen. Insgesamt kamen 55 Stunden an Videos mit über elf Millionen Einzelbildern zusammen. Diese wurden bzgl. verschiedener Aktionen in knapp 40.000 zeitliche Segmente unterteilt und über 450.000 Objekte durch ihr umgebendes Rechteck annotiert. Da der Fokus auch hier nicht auf der Analyse von Händen lag, wurden keine Hände annotiert.

3 Blickanalyse in mobilen Anwendungen

In diesem Kapitel wird in Abschnitt 3.1 eines der ersten Verfahren für die vollautomatische echtzeitfähige Blickanalyse in mobilen Anwendungen vorgestellt und anhand einzelner Ergebnisse einer Studie die sinnvolle Nutzung demonstriert. Es wurde in Hammer et al. [Ham13a] vorgestellt und zu einer Zeit entwickelt, als kein mobiles Blickmessgerät namhafter Hersteller, zu denen das Verwendete gehörte, den 3D-Vektor eines Sehstrahls weder zur Laufzeit noch im Nachhinein zur Verfügung stellte. Auch ermöglichte keine Analysesoftware dieser Hersteller eine automatische Analyse zur Laufzeit. Letzteres ist für interaktive mobile Anwendungen wie im Projekt ARtSENSE, siehe Abschnitt 1.2.1.5, oder KonsensOP, siehe Abschnitt 1.2.1.4, eine Grundvoraussetzung.

Durch die Berechnung und Verwendung von dreidimensionalen Blickdaten konnte festgestellt werden, dass keine realistische Visualisierung für 3D-Blickdaten in Form von 3D-Heatmaps existierte, die die menschliche visuelle Schärfe des Blickes noch Verdeckungen berücksichtigt, siehe Abschnitt 2.2.7. In Abschnitt 3.2 dieses Kapitels wird deshalb das Verfahren aus Maurus et al. [Mau14] für die echtzeitfähige Visualisierung von realistischen Heatmaps vorgestellt, das den Stand der Technik zur Zeit der Veröffentlichung übertraf und den aktuellen Stand der Technik direkt beeinflusst hat. Die in diesem Kapitel gezeigten Blickdaten wurden in einer Studie im Projekt ARtSENSE im Labor von Lavoisier (Musée des Arts et Métiers, Paris, Frankreich) sowie in der Valencianischen Küche (Museo Nacional de Artes Decorativas, Madrid, Spanien) aufgenommen. Die Umgebungen wurden manuell als 3D-Modelle nachgebildet, siehe Abbildungen 3.1 sowie 3.2.

Abbildung 3.1: 3D-Modell des Labors von Lavoisier mit Gerätschaften zur Untersuchung der Rolle von Sauerstoff bei Verbrennungen und Entwicklung des Masseerhaltungssatzes bei chemischen Reaktionen

Abbildung 3.2: 3D-Modell der Valencianischen Küche. Dargestellt sind zwei von vier Wänden, die eine Herrin und Angestellte bei verschiedenen Aufgaben in einer Küche in Valencia aus der Zeit um das Jahr 1780 zeigen.

3.1 Vollautomatische 3D-Blickanalyse

Für eine vollautomatische Blickanalyse in mobilen Anwendungen muss der Blick mit der Umwelt registriert werden. Hierfür wurde im vorliegenden Anwendungsfall der geometriebasierte Ansatz genutzt, siehe Abschnitt 2.2.2. Dafür muss der Sehstrahl in der Umgebung rekonstruiert werden. Wie dies mit dem verwendeten Dikablis Blickmessgerät durchgeführt wird, ist im Abschnitt 3.1.1 beschrieben. Abschnitt 3.1.2 zeigt das verwendete Verfahren für die Blickbewegungsanalyse. Eine komplette Blickanalyse wird an einem Beispiel in Abschnitt 3.1.4 vorgestellt.

3.1.1 Bestimmung der Position des Augapfels

Um den Sehstrahl in der Umgebung rekonstruieren zu können, muss dieser am Knotenpunkt des Auges angesetzt werden, siehe Abbildung 2.1. Der Knotenpunkt des Auges kann relativ zur Szenenkamera eines Blickmessgerätes definiert werden. Nutzt die Blickmessung ein 3D-Modell des Auges, wird der Knotenpunkt mitberechnet. Kennt man die Pose der Szenenkamera, die extrinsischen Parameter, so kann die 3D-Position des Knotenpunktes des optischen Systems des Auges sowie der Sehstrahl in das Referenzsystem transformiert werden.

Für die Posenschätzung kommen Schätzungsverfahren aus Abschnitt 2.1.6 in Frage. Die korrekte Bestimmung der Pose ist essentiell für die korrekte Verortung des Sehstrahls in der Umgebung. Grundsätzlich können dafür outside-in Posenschätzer genutzt werden. Dafür müssen am Blickmessgerät optische Referenzmarkierungen und in der Umgebung Sensorik zur Erfassung dieser Referenzmarkierungen angebracht werden. Um Bereiche von mehreren Quadratmetern Größe abzudecken, steigen die Kosten hierfür schnell in unattraktive Höhen, wenn dafür kommerzielle Systeme verwendet werden müssen. Die Alternative ist die inside-out Posenschätzung. Dafür werden optische Referenzmarkierungen in der Umgebung angebracht, siehe bspw. Abbildung 3.4a auf Seite 67 oder 3.7a auf Seite 70, und über die Bilder der Szenenkamera die optischen Markierungen durch Bildverarbeitungsverfahren lokalisiert

und die Posenschätzung durchgeführt. Der Vorteil von inside-out Tracking ist, dass es leichter in großflächigen Umgebungen genutzt werden kann, weil auf Papier ausgedruckte Marken einfacher anzubringen sind als ein Erfassungssystem aus mehreren Kameras. Der Nachteil bei der inside-out Posenschätzung mit Marken in der Umgebung ist, dass diese als visueller Reiz die Aufmerksamkeit ablenken können.

Wie in Abschnitt 2.2.1 beschrieben, beinhaltet das Dikablis Blickmessgerät ein Marken-Lokalisierungsverfahren im Bild der Szenenkamera. So ergeben sich Korrespondenzen von 2D-Positionen im Bild der Szenenkamera zu 3D-Positionen, die über die Anbringung der Referenzmarkierungen bekannt sind und über die mit der Lösung des PnP-Problems [Nak16] die extrinsischen Parameter geschätzt werden.

3.1.1.1 Rekonstruktion der Blickrichtung

Da die Blickrichtung in Form des 3D-Richtungsvektors des Sehstrahls nicht direkt zur Verfügung steht, das Dikablis System aber die Schnittpunkte mit den von den Referenzmarkierungen aufgespannten 2D-Ebenen im Raum und dem Sehstrahl berechnet, kann der Sehstrahl rekonstruiert werden. Abbildung 3.3 veranschaulicht diesen Vorgang. Die Marker sind als schwarz-weiß strukturierte Marken auf Stativen angebracht und so aufgestellt, dass bei den häufigsten Kopfausrichtungen während der Betrachtung der Szene mindestens ein Marker zu sehen ist. Werden Marker vom System im Szenenkamerabild erkannt, werden sie blau umrandet. In der Abbildung werden beim Blick auf den Kopf der Büste drei Marker erkannt. Die von jedem Marker aufgespannte zweidimensionale Ebene ist im Bild durch grüne Punkte dargestellt. Wird ein Marker an einer Wand angebracht, stimmen die virtuelle, vom Marker aufgespannte Ebene und Wand überein und der Schnittpunkt mit der Ebene ist direkt der Blickpunkt. Im vorliegenden Fall liegt der Schnittpunkt mit einer Ebene, dargestellt durch die gelben Kugeln im Bild, nicht auf einem Objekt, sondern befindet sich irgendwo in der Luft. Im Idealfall sollten diese gelben Kugeln alle auf einer Linie, dem Sehstrahl, liegen.

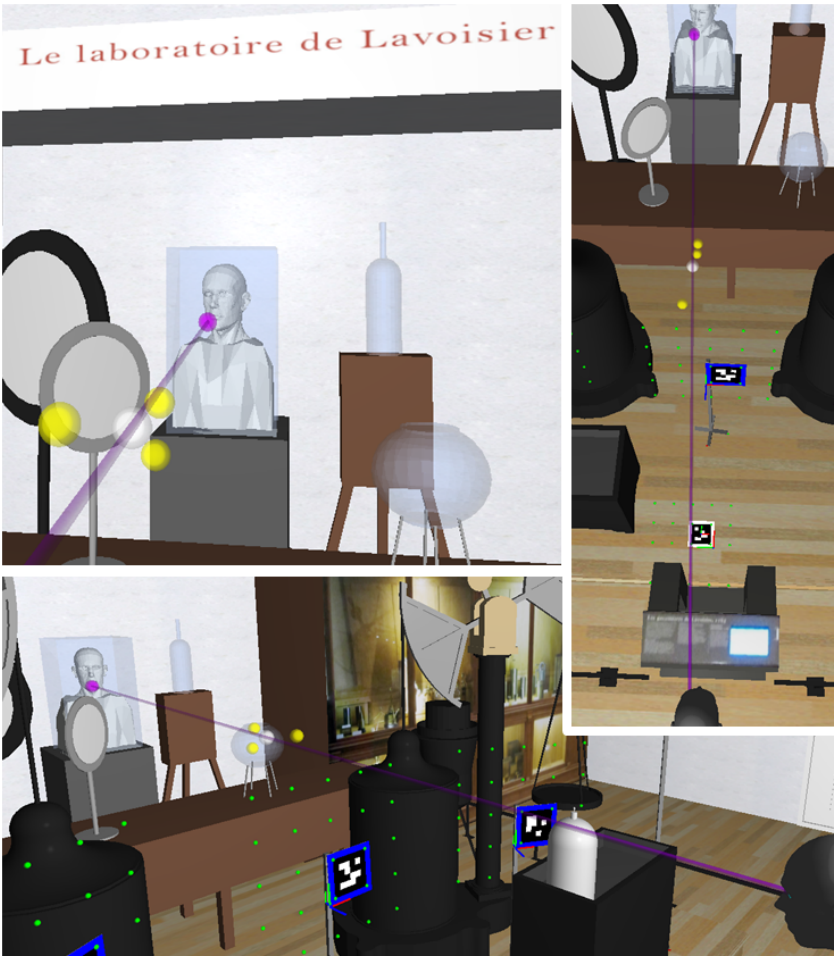


Abbildung 3.3: Berechnung des 3D-Blickpunktes (violette Kugel) durch die Schnittpunktberechnung von Sehstrahl mit 3D-Umgebungsmodell: Der Sehstrahl (violette Linie) ergibt sich aus der Position des Auges und dem Schwerpunkt (weißgrau) des vom Dikablis System berechneten Schnittpunktes (gelb) des Sehstrahls mit den von den Markern aufgespannten virtuellen Ebenen (grüne Punkte).

In der Praxis ist dies aufgrund von Ungenauigkeiten sowohl bei der Markerlokalisierung und Schnittpunktberechnung als auch bei der manuellen dreidimensionalen Positionierung und Ausrichtung der Marken nicht der Fall. Deshalb wird, wie in der Abbildung zu sehen, der Schwerpunkt der gelb gefärbten Schnittpunkte mit den Ebenen berechnet. Dieser ist in der Abbildung als weißgraue Kugel zu sehen. Die Position dieses Schwerpunktes sowie die zuvor berechnete Position des Auges bzw. ihres Knotenpunktes definieren den Sehstrahl. Der Sehstrahl wird in Richtung der Blickrichtung mit dem 3D-Modell der Umwelt geschnitten. Im Beispiel ist der Blickpunkt am Kinn des Kopfes der Büste berechnet und violett dargestellt.

3.1.2 Berechnung von Fixationen aus 3D-Blickpunkten

In der Einleitung zu Abschnitt 2.2 sowie in Abschnitt 2.2.3 wurde die Wichtigkeit der Blickbewegungsanalyse als Aufteilung des Blickpfades in Fixationen und Sakkaden dargelegt. Für die Blickbewegungsanalyse wird nachfolgend das dort bereits erwähnte I-VT-Verfahren [Sal00, Kom10], angepasst an 3D-Blickpunkte, beschrieben.

I-VT basiert auf Punkt-zu-Punkt-Geschwindigkeiten in Grad pro Sekunde gemessen. Sind die aktuelle Position des Auges sowie die zwei letzten Blickpunkte gegeben, können zwei Sehstrahlen vom Auge zu den Blickpunkten gebildet werden. Der von diesen Strahlen eingeschlossene Winkel, geteilt durch die vergangene Zeit, ergibt die Punkt-zu-Punkt-Geschwindigkeit. Diese wird mit einem räumlichen Schwellenwert von $50°/s$ verglichen. Ist die aktuelle Geschwindigkeit größer, gehört der letzte 3D-Blickpunkt zu einer Sakkade und ansonsten zu einer Fixation. Die Position einer Fixation berechnet sich als Schwerpunkt aller zugehörigen 3D-Blickpunkte. Das Verfahren I-VT lässt sich optional mit einer zeitlichen Mindestdauer für Fixationen versehen, welche nach Goldberg und Schryver [Gol95] auf 100 Millisekunden gesetzt werden sollte. Fixationen, die kürzer als diese Mindestdauer sind, werden entfernt und die Blickpunkte Sakkaden zugeordnet.

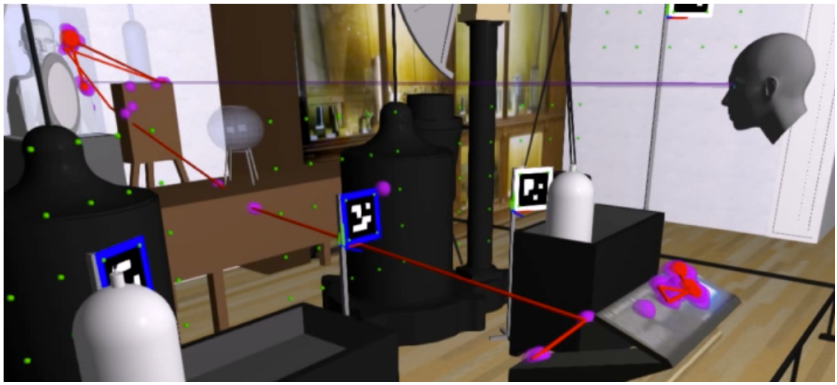
Abbildung 3.4a zeigt die Visualisierung der Blickpunkte eines Blickpfades von drei Sekunden Länge als violette Kugeln. Die Person liest gerade Erläuterungen zum Labor von Lavoisier. Der Blickpfad, umgewandelt in Blickbewegungen, ist in Abbildung 3.4b zu sehen, wo mehrere Fixationen als rote Kugeln dargestellt sind.

(a) Darstellung eines Blickpfades von drei Sekunden Länge, während die Person Informationen über das Labor von Lavoisier liest.

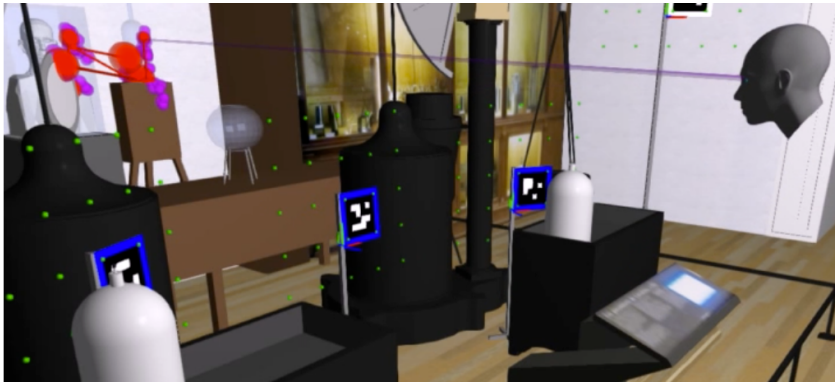
(b) Darstellung des Blickpfades durch Einblendung der Fixationen (rote Kugeln)

Abbildung 3.4: Visualisierung von Blickpunkten, Fixationen und Sakkaden (1/2).

Je länger eine Fixation dauert, desto größer ist der Durchmesser der zugehörigen roten Kugel. Blickpunkte, die zu Sakkaden gehören, werden durch rote Linien verbunden. Abbildung 3.5a zeigt dies beim Fokus auf die Büste. Es ist zu erkennen, dass Fixationen erst wieder entstehen, wenn der Blick auf der Büste verweilt. Nach der Büste wird das Objekt rechts daneben betrachtet, wie in Abbildung 3.5b zu sehen.



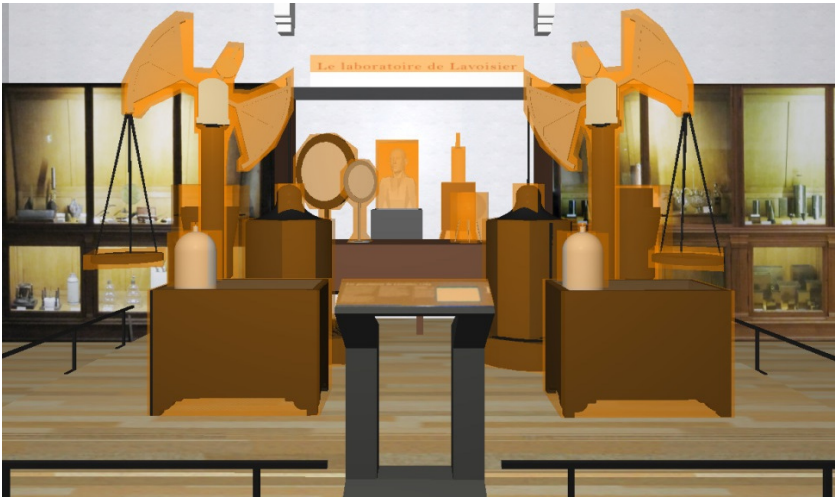
(a) Darstellung des Blickpfades während des Wechsels des Fokus der Aufmerksamkeit vom Informationstext auf die Büste. Sakkaden sind als rote Linien dargestellt.



(b) Der Fokus der Aufmerksamkeit wechselt von der Büste auf das Objekt rechts daneben.

Abbildung 3.5: Visualisierung von Blickpunkten, Fixationen und Sakkaden (2/2).

(a) AOIs in der Valencianischen Küche

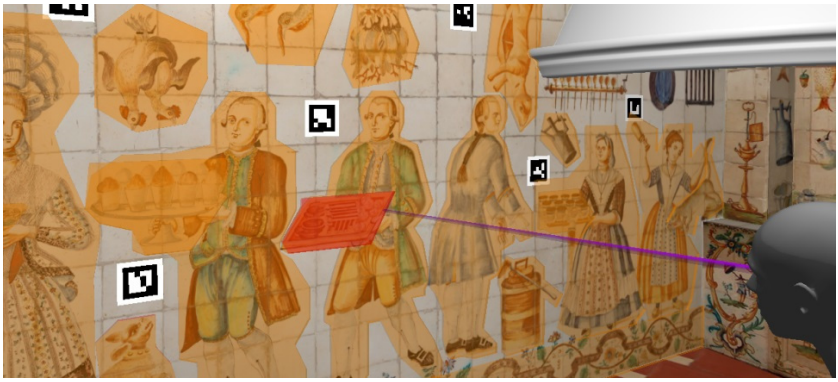


(b) AOIs im Labor von Lavoisier

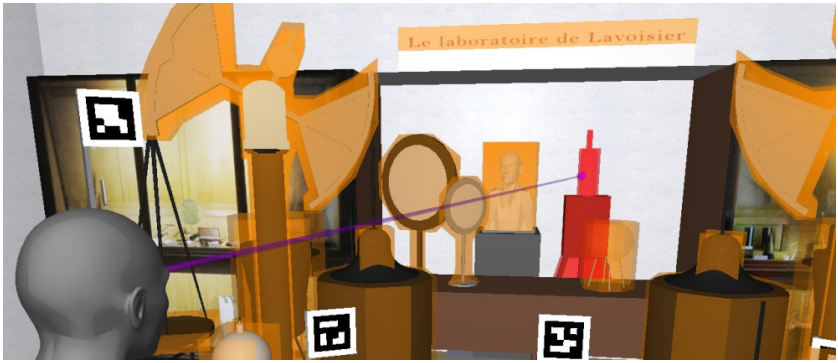
Abbildung 3.6: Definierte AOIs, dargestellt durch orange transparent gefärbte und einhüllende dreidimensionale Körper

3.1.3 Definition relevanter Bereiche

Für die automatische Analyse des Blickverhaltens müssen relevante Bereiche, sog. AOIs, siehe Abschnitt 2.2.4, definiert werden. Dies wurde für die Valencianische Küche und das Labor von Lavoisier durchgeführt, wie in Abbildung 3.6a und 3.6b zu sehen. Die Abbildungen 3.7a und 3.7b zeigen die Visualisierung betrachteter AOIs. Die aktuell betrachtete AOI wird dabei transparent in Rot hervorgehoben.



(a) Visualisierung einer vom Blick getroffenen AOI (rot) in der Küche



(b) Visualisierung einer vom Blick getroffenen AOI (rot) im Labor

Abbildung 3.7: Visualisierung betrachteter AOIs

3.1.4 Blickanalyse in der Valencianischen Küche

Das in dieser Arbeit entwickelte System von Hammer et al. [Ham13a] erlaubt die echtzeitfähige dreidimensionale Blickanalyse in mobilen Anwendungen. In den folgenden zwei Abschnitten wird demonstriert, wie das System für die Blickanalyse als auch die blickbasierte implizite Interaktion genutzt werden kann.

3.1.4.1 Manuelle Analyse des Blickverhaltens

In der Valencianischen Küche wurde eine Studie zum Blickverhalten durchgeführt. Die erste Aufgabe der Probanden war die freie Betrachtung der in Abbildung 3.2 links zu sehenden Wand, welche die Hausherrin sowie die Bediensteten darstellt. Das Blickverhalten eines Probanden während dieser freien Betrachtung ist in Abbildung 3.8 durch Blickpunkte dargestellt. Der Blickpfad besteht aus mehr als 7.221 Blickpunkten während einer Dauer von knapp fünf Minuten. Das Blickverhalten ist übersichtlicher zu betrachten, wenn wie in Abbildung 3.9 nur die Fixationen gezeigt werden. Festzustellen ist, dass die gesamte Wand betrachtet wurde. Sowohl der obere Teil mit unterschiedlichen Gerätschaften als auch die untere Hälfte der Wand mit den Personen wurde visuell inspiziert. Bei den Personen ist zu erkennen, dass u. a. ihre Gesichter Aufmerksamkeit auf sich gezogen haben. Dieses wurde auch in anderen Studien beobachtet [Bus35, Eng10].

Anschließend wurden die Probanden mit Kopfhörern ausgestattet und erhielten eine Audioführung. Sie sollten den Erklärungen zuhören und die Wand betrachten. Die rohen Blickdaten des gezeigten Blickverhaltens sind in Abbildung 3.10 zu sehen, während sie in Abbildung 3.11 als Fixationen dargestellt sind. Da Heatmaps eine intuitivere Ansicht des Blickverhaltens für die visuelle Interpretation sind, wird bereits hier das im später folgenden Abschnitt 3.2 beschriebene Verfahren zur Visualisierung von Heatmaps in dreidimensionalen Anwendungen für die Visualisierung genutzt. Abbildung 3.12 zeigt das Blickverhalten während der freien Betrachtung als Heatmap, während Abbildung 3.13 das Blickverhalten mit Audioführung zeigt.

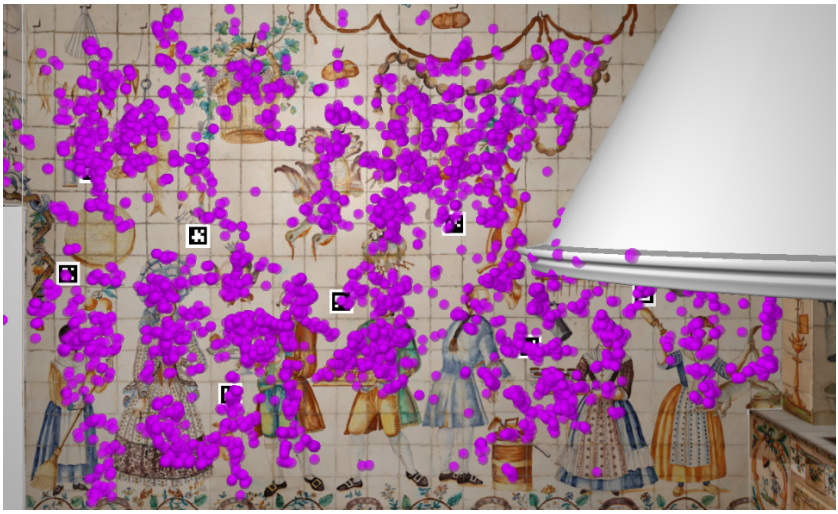


Abbildung 3.8: Visualisierung von 7.221 Blickpunkten (Dauer: ca. 5 Minuten) entstanden bei freier Betrachtung

Abbildung 3.9: Visualisierung des Blickverhaltens aus Abbildung 3.8 durch Fixationen in Form von 693 roten Kugeln

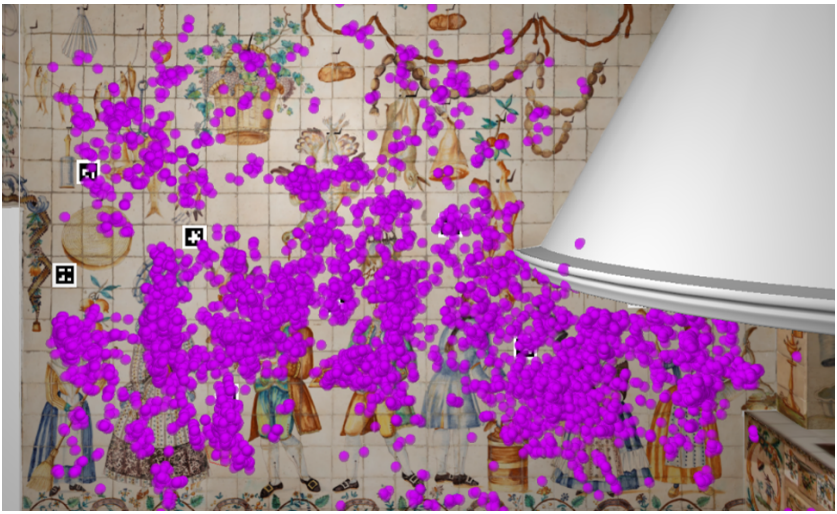


Abbildung 3.10: Visualisierung der Blickdaten entstanden bei Betrachtung mit Audioführung (ca. 6,5 Minuten)

Abbildung 3.11: Visualisierung des Blickverhaltens aus Abbildung 3.10 durch Fixationen in Form von roten Kugeln

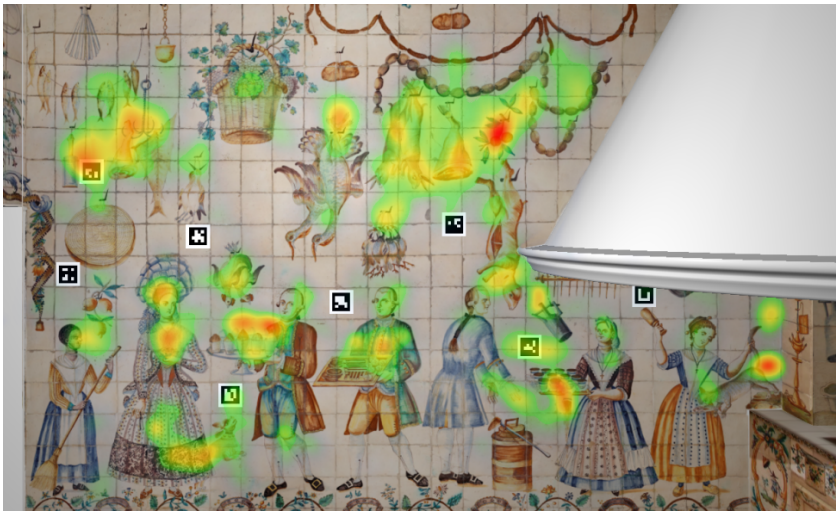


Abbildung 3.12: Darstellung einer normalisierten Heatmap der Blickdaten bei freier Betrachtung (ca. 5 Minuten)

Abbildung 3.13: Darstellung einer normalisierten Heatmap der Blickdaten unter Einfluss der Audioführung (ca. 6,5 Minuten)

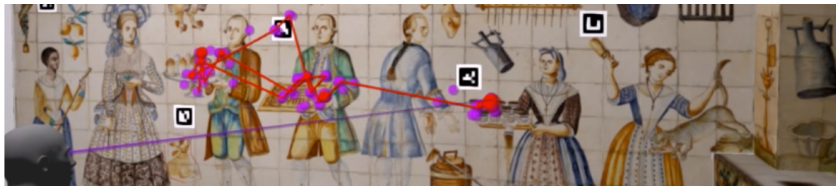
Vergleicht man das Blickverhalten während der freien Betrachtung mit dem unter Nutzung der Audioführung ist besonders beim Vergleich der Heatmap-Visualisierungen der Unterschied zu erkennen. Die Informationen der Audioführung haben die Aufmerksamkeit auf die untere Hälfte der Wand gelenkt, wo die Personen abgebildet sind sowie deren Kleidung und Tätigkeiten inklusive benutzter Objekte beschrieben werden. Besonders gut ist in Abbildung 3.13 zu sehen, dass im Fokus der Betrachtung die Gesichter stehen. Die Hausherrin ist die zweite Person von links. Rechts neben ihren Füßen befindet sich ein Hund, der ebenfalls betrachtet wird. Er ist Teil der Erläuterungen zur Wand. Weitere beschriebene Elemente sind die Katze ganz rechts in der Szene, die mit einem Küchenholz verjagt wird, die Tasse, die vom Tablett links neben der Katze fällt, sowie die Tabletts, die von den Angestellten getragen werden. Die Betrachtung der Tabletts wird in Abbildung 3.14 veranschaulicht. Der dargestellte Blickpfad hat eine Dauer von ungefähr drei Sekunden und wurde während einer Erzählung über die Angestellten mit ihren drei Tabletts als Teil der Audioführung aufgezeichnet. Es ist eindeutig zu erkennen, wie der Blick erst auf dem linken Tablett (Abbildung 3.14a) verweilt. Anschließend wandert er über die Gesichter der zwei männlichen Angestellten rechts neben der Herrin zum mittleren Tablett (Abbildung 3.14b). Auf diesem verweilt der Blick für längere Zeit (Abbildung 3.14c). Dann zieht das linke Tablett noch einmal die Aufmerksamkeit auf sich (Abbildung 3.14d), bevor der Blick über das mittlere Tablett auf das Tablett ganz rechts geht. Der beschriebene Blickpfad untermauert, dass der Betrachter die Erklärungen nicht nur akustisch gehört, sondern auch visuell nachvollzogen hat, was ein Verständnis des Gehörten und damit eine gedankliche Verarbeitung voraussetzt.

(a)

(b)

(c)

(d)



(e)

Abbildung 3.14: Blickverhalten bei den Tabletts

Ein weiteres Beispiel für das Nachvollziehen der Beschreibungen der Audioführung mit den Augen ist das Verjagen der Katze sowie der Vergleich der Kleidung zweier Angestellter, siehe Abbildung 3.15. Es ist ein Blickwechsel über die gesamte Szene von der Angestellten ganz rechts zur Angestellten ganz links und wieder zurück zu erkennen, während die Audioführung über die Kleidung der zwei Frauen berichtet. Zusätzlich wird beschrieben, wie die Angestellte ganz rechts die Katze verjagt, was dazu führt, dass bei der Angestellten links daneben die Tasse vom Tablett fällt.

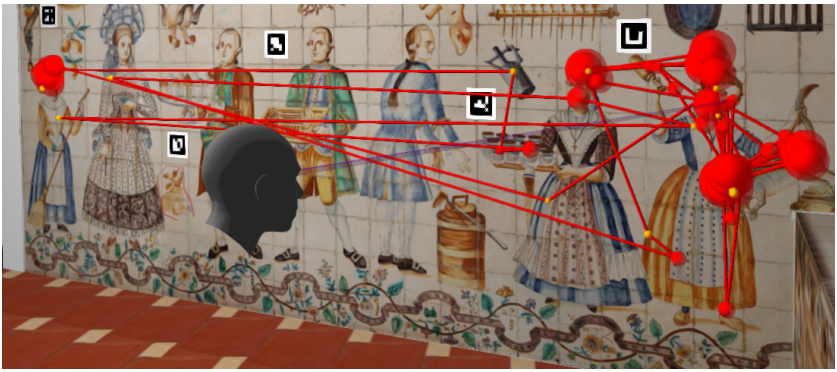


Abbildung 3.15: Darstellung des Blickwechsels von der Angestellten ganz rechts zur Angestellten ganz links sowie zurück zur Angestellten rechts, als sie die Katze verjagt.

3.1.4.2 Automatische Analyse des Blickverhaltens

Durch eine AOI-basierte automatische Blickanalyse kann die oben durchgeführte qualitative Bewertung des Blickverhaltens während des freien Betrachtens, Abbildung 3.12, und des Betrachtens mit Audioführung, Abbildung 3.13, quantitativ belegt werden. Die AOIs der Valencianischen Küche sind in Abbildung 3.6a dargestellt. Die kumulierte Fixationszeit, siehe Abschnitt 2.2.4.2 zu Metriken der Blickanalyse für AOIs, wurde für die AOIs berechnet und die Ergebnisse werden nachfolgend beschrieben.

Bei der freien Betrachtung kamen für den ausgewählten Probanden 294 Sekunden (s) an Blickdaten zusammen. Davon enthalten 264 s gültige Berechnungen, es konnte also die Blickmessung und Posenschätzung durchgeführt sowie der Blickpunkt berechnet werden. Von diesen verwendbaren Daten entfallen 142 s auf Fixationen, ein Anteil von 54 % der gesamten Zeit. Davon konnten 67 % (96 s) AOIs zugeordnet werden, was bedeutet, dass 33 % der Fixationszeit auf Elementen lag, die nicht durch AOIs erfasst werden. Dies ist dadurch zu erklären, dass einige Bereiche nicht von AOIs abgedeckt sind, vgl. die AOIs in Abbildungen 3.6a mit den Fixationen in Abbildung 3.9. Verglichen mit den unter Audioführung erhoben Blickdaten kamen 388 s an Blickdaten zusammen, von denen 383 s eine Blickpunktberechnung ermöglichten. Davon werden 201 s Fixationen zugeordnet, was einem Anteil von 52 % entspricht, ähnlich zur freien Betrachtung. Von diesen Fixationen konnten 83 % (167 s) AOIs zugeordnet werden.

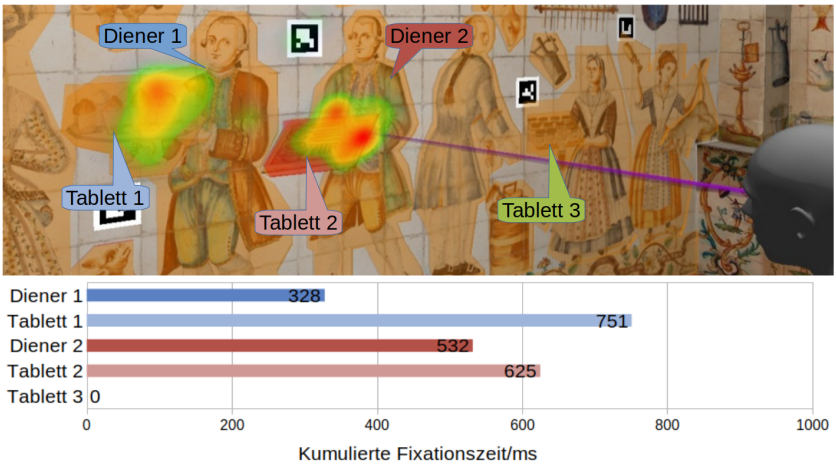
Abbildung 3.13 legt eine Unterteilung der AOIs in eine obere Hälfte der Wand sowie untere Hälfte mit den Personen nahe. Im Fall des freien Betrachtens bekamen die AOIs des oberen Teils 45 s kumulierte Fixationszeit (46 %) und der untere Bereich 52 s (54 %), was vergleichsweise ausgeglichen ist. Mit Audioführung hingegen bekam der obere Teil nur 30 s kumulierte Fixationszeit (18 %) und der untere Teil 137 s (82 %), was eindeutig die längere Betrachtung des unteren Teils belegt und auf die Audioführung zurückzuführen ist.

Die erste Fixation überhaupt fiel in beiden Fällen auf die Hausherrin, welche bei freier Betrachtung inkl. der AOIs der Rose und des Fächers, die sie in ihren zwei Händen hält, 14 s kumulierte Fixationszeit (14 %) und im Fall mit Audioführung 25 s (15 %) der gesamten Fixationszeit bekam. Hier ist folglich eine fast doppelt so lange Aufmerksamkeitsspanne mit Audioführung zu beobachten. Die Dienerin links neben der Hausherrin erhielt eine deutlich erhöhte Aufmerksamkeit durch die Audioführung. Dies spiegelt die kumulierte Fixationszeit von 4 s (4 %) ohne und 18 s (11 %) mit Audioführung wider. Die Tabletts bekamen ohne 11 s (12 %) und mit Audioführung 40 s (24 %) der Aufmerksamkeit, was ebenfalls durch den geleiteten Fokus der Beschreibungen der Audioführung zu erklären ist. Ein weiteres Beispiel des Einflusses der Audioführung

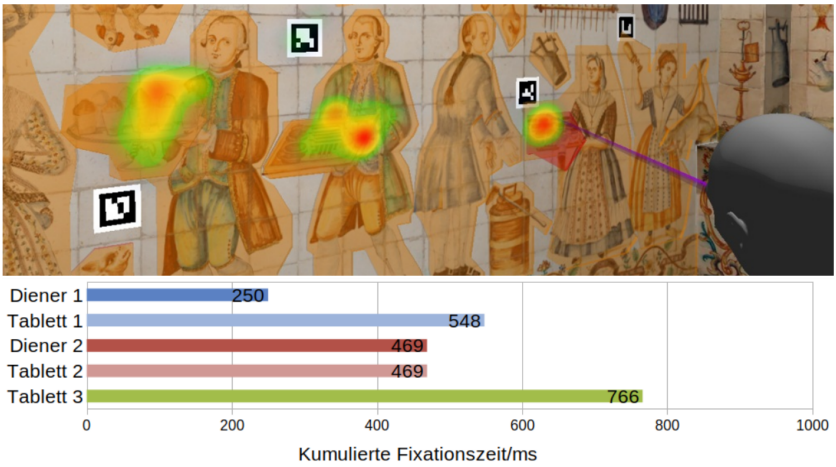
sind die Angestellte und die Katze ganz rechts im Bild, zu denen es eine ausführliche Beschreibung gab. Beide zusammen bekamen ohne Audioführung 7 s (7 %) und mit 46 s (27 %) der Aufmerksamkeit, was bzgl. der kumulierten Fixationszeit mehr als eine Versechsfachung ist. Im Fall ohne Audioführung und ohne Katze bekam diese Angestellte eine kumulierte Fixationszeit von 3 s (3 %). Durch die Audioführung erhöhte sich diese auf 32 s (19 %), also zehnmal so viel Aufmerksamkeit. Die beschriebenen Elemente oder Gruppen von Elementen, z. B. die Tabletts, wurden folglich mit Audioführung zwei- bis zehnmal so lange inspiziert wie ohne akustische Beschreibungen.

3.1.4.3 Online-Berechnung des implizit bekundeten Interesses

Die Erkennung des implizit über das Blickverhalten bekundeten Interesses unter Nutzung einer automatischen Blickanalyse von Blickpfaden kürzerer Dauer kann in interaktiven mobilen AR-Anwendungen auf verschiedene Weise genutzt werden. Wird implizit Interesse an einem Objekt bekundet und bisher keine akustische Beschreibung abgespielt, könnte über eine Audioführung eine zugehörige akustische Beschreibung abgespielt werden. Während des Abspielens kann durch eine weitere Verfolgung des Blickes festgestellt werden, ob das Gehörte visuell nachverfolgt wird, was als ein Zeichen für aufmerksames Zuhören gewertet werden kann. Dies wird am Beispiel der Szene aus Abbildung 3.14 dargestellt. Abbildung 3.16 zeigt die visuelle Aufmerksamkeit geschätzt durch die kumulierte Fixationszeit auf den betrachteten AOIs während der Audioführung. In Abbildung 3.16a ist die visuelle Aufmerksamkeit auf Tablett 2 gerichtet. Die Aufmerksamkeit auf allen Tabletts zusammen beträgt 62 % der kumulierten Fixationszeit. Eine Sekunde später, siehe Abbildung 3.16b, liegt der Fokus der Aufmerksamkeit auf Tablett 3 und die kumulierte Fixationszeit aller Tabletts beträgt 71 % der gesamten visuellen Aufmerksamkeit. Diese Berechnungen unterstützen quantitativ die Aussage, dass der Nutzer die über die Audioführung erhaltenen Informationen visuell verarbeitet und die Erklärungen den Blick durch die Szene führen. Dieses Wissen sollte ein interaktives AR-System dazu nutzen, die Erklärungen weiterzuverfolgen.



(a) Zeitpunkt t



(b) Zeitpunkt $t + 1$

Abbildung 3.16: Betrachtung der in Abbildung 3.14 dargestellten Szene bzgl. kumulierter Fixationszeiten, oben zum Zeitpunkt t und unten zum Zeitpunkt $t + 1$ s, für einen Blickpfad von 4 s Länge.

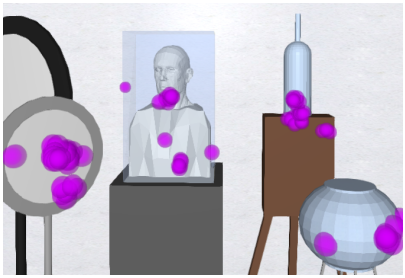
3.1.5 Vergleich zum Stand der Technik

Im obigen Abschnitt 3.1 wurde ein System für die manuelle als auch vollautomatische Analyse von Blickdaten in mobilen Anwendungen vorgestellt. Das gezeigte System ermöglichte die Berechnung von 3D-Blickpunkten und deren Analyse in Echtzeit zu einer Zeit, als kein kommerziell erhältliches Blickmessgerät inkl. mitgelieferter Software dieser Aufgabe gerecht wurde. Es wurde eine manuelle Blickanalyse anhand eines Beispiels durchgeführt und es konnte qualitativ als auch quantitativ gezeigt werden, wie eine Audioführung Einfluss auf die visuelle Aufmerksamkeit haben kann. Zudem konnte gezeigt werden, dass eine vollautomatische Blickanalyse kürzerer Blickpfade zur Detektion unbewusst bekundeten Interesses genutzt werden kann, um in einem mobilen interaktiven AR-System die Auswahl präsentierter Inhalte zu beeinflussen. Aufgrund diverser Nachteile des verwendeten älteren Blickmessgerätes bzgl. Robustheit bei der Blickmessung unter verschiedenen Umgebungsbedingungen und Nutzbarkeit für Szenarien, in denen Personen in Bewegung sind, mussten die Daten der meisten Probanden als nicht nutzbar gewertet werden. Mit aktuelleren Blickmessgeräten, auch dem Nachfolger des verwendeten Systems, haben sich mobile Blickmessgeräte als deutlich robuster erwiesen, sowohl was den Sitz auf der Nase als auch die Blickmessung angeht. Die Schwierigkeit der Posenschätzung ohne Marker bleibt allerdings bestehen.

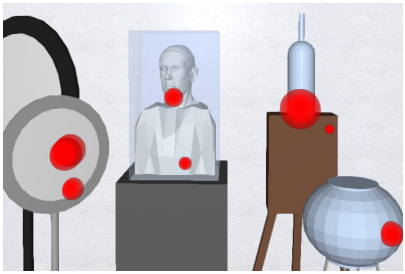
3.2 Realistische 3D-Heatmaps für die manuelle Blickanalyse

Für eine effiziente Analyse von Blickdaten ist eine informative Visualisierung notwendig. Abbildung 3.17a zeigt ca. 100 Blickpunkte, welche in Abbildung 3.17b als Fixationen dargestellt sind. Um die Bereiche größter visueller Aufmerksamkeit besser erkennen zu können, besonders bei längeren Blickpfaden, bieten sich Heatmaps an. Bei der Einfärbung wird ein Gewicht über eine Farbabbildungsfunktion auf einen Farbwert abgebildet. Abbildung 3.17c zeigt den Regenbogen-Farbverlauf und Abbildung 3.17d einen Blau-Rot-Farbverlauf. Bei beiden sind rote Bereiche diejenigen mit der höchsten Gewichtung. Bei der Luminanz-Heatmap in Abbildung 3.17e wird nicht die Farbe, sondern die Helligkeit entsprechend der Gewichtung verändert. Helle Bereiche entsprechen höheren Gewichtungen. Eine weitere interessante Heatmap-Variante ist die Veränderung der Transparenz, siehe Abbildung 3.17f. Hierbei werden Bereiche, die betrachtet wurden, durch höhere Gewichtung überhaupt erst sichtbar. Die Blau-Rot- und Transparenz-Variante der Heatmap aus Abbildung 3.13 auf Seite 74 ist in den Abbildungen 3.18 sowie 3.19 zum Vergleich längerer Blickpfade dargestellt.

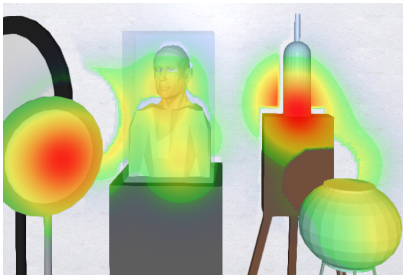
Die Person, die die Analyse der Blickdaten vornimmt, bekommt durch die Heatmap einen schnellen Eindruck davon, welche Bereiche die meiste Aufmerksamkeit auf sich gezogen haben und welche Bereiche nicht besonders bzw. gar nicht betrachtet wurden. Die Erstellung von dreidimensionalen Heatmaps aus 3D-Blickdaten ist deutlich komplizierter als für reine 2D-Blickdaten, weil eine dreidimensionale Szene dreidimensionale Oberflächen und Verdeckungen enthält. Um der Realität so nah wie möglich zu bleiben, wird im nachfolgend vorgestellten Verfahren die visuelle Schärfe des Blickes in die Welt projiziert und Verdeckungen werden dabei berücksichtigt. Zusätzlich wurde bei der Entwicklung auf eine effiziente Berechnung geachtet, um eine echtzeitfähige Visualisierung zu ermöglichen.



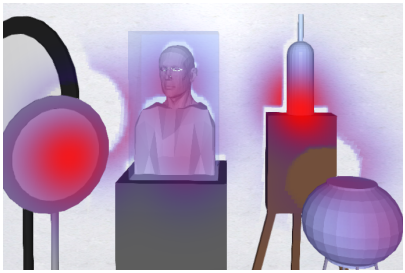
(a) Blickpunkte



(b) Fixationen



(c) Regenbogen-Heatmap



(d) Blau-Rot-Heatmap

(e) Luminanz-Heatmap

(f) Transparenz-Heatmap

Abbildung 3.17: Unterschiedliche Darstellungsmöglichkeiten für Blickpfade.

Abbildung 3.18: Abbildung 3.13 als Heatmap mit Blau-Rot-Verlauf

Abbildung 3.19: Abbildung 3.13 als Transparenz-Heatmap

Der nachfolgende Teil dieses Kapitels ist wie folgt strukturiert: Abschnitt 3.2.1 erklärt, wie die visuelle Schärfe in die Szene projiziert wird. Im nachfolgenden Abschnitt 3.2.2 wird die Berücksichtigung von Verdeckungen beschrieben. Auf die Verschnellerung der Berechnung des Verfahrens wird in den Abschnitten 3.2.3 und 3.2.4 eingegangen. Ein Vergleich zum Stand der Technik wird in Abschnitt 3.2.5 durchgeführt.

3.2.1 Projektion der visuellen Schärfe in die Szene

Die visuelle Schärfe des menschlichen Blickes ist gegeben durch den Aufbau der Netzhaut und wird typischerweise durch eine Gauß-Verteilung beschrieben, siehe Abschnitt 2.2.7, welche am Blickpunkt zentriert werden muss. Die Standardabweichung kann auf Basis des Öffnungswinkels der Fovea gewählt werden, der ca. 2° beträgt. In verwandten Arbeiten wird für 3D-Blickdaten eine dreidimensionale Gauß-Verteilung an der Stelle des jeweiligen Blickpunktes positioniert. Abbildung 3.20 veranschaulicht die Einfärbung einer Wand um einen 3D-Blickpunkt. Aus frontaler Sicht wie im oberen Bild ist die gleichmäßige kreisförmige Ausbreitung der Gewichte zu sehen. Im unteren Bild ist die Sicht auf den Blickpunkt aus Sicht der Person dargestellt. Aufgrund des Perspektivwechsels erscheint die Ausbreitung der visuellen Schärfe nicht mehr gleichmäßig in alle Richtungen verlaufend, sondern elliptisch. Realistisch wäre eine Projektion der Netzhaut in die Szene und aus Sicht der Person ergäbe sich dann eine gleichmäßige kreisförmige Ausbreitung der visuellen Schärfe. Um die Heatmap so realistisch wie möglich zu erstellen, ist die grundlegende Idee des in dieser Arbeit entwickelten *Projected Gaussians* Verfahrens, die visuelle Schärfe des menschlichen Blickes in die betrachtete Szene zu projizieren.

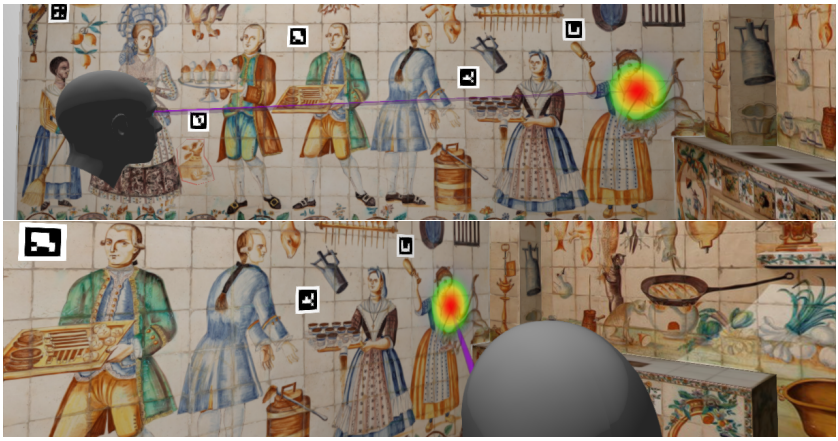


Abbildung 3.20: Gewichtung mit einer dreidimensionalen Gauß-Verteilung an Stelle des Blickpunktes. Im unteren Bild ist zu erkennen, dass dies einer Verzerrung der visuellen Schärfe des Blickes gleichkommt, da der eingefärbte Bereich aus Sicht der Person eine elliptische Form bekommt.

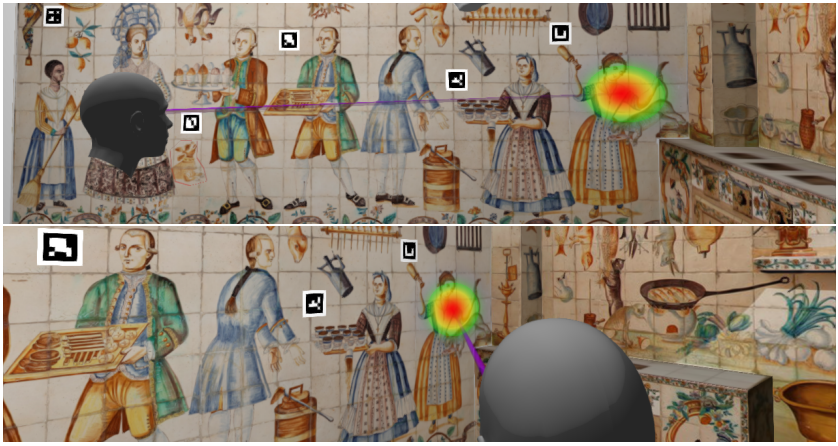


Abbildung 3.21: Gewichtung durch in die Szene projizierte Gaußverteilung. Aus Sicht der Person (unteres Bild) ist die visuelle Schärfe des Blickes nun in alle Richtungen gleich ausgedehnt. Wechselt die Perspektive des Betrachters, verzerrt sich entsprechend die Einfärbung der betrachteten Region.

Da sich die visuelle Schärfe des menschlichen Blickes nicht ändert, weil sich die Netzhaut nicht ändert, kann eine die visuelle Schärfe beschreibende Gauß-Verteilung als statische Textur erstellt und auf der Grafikkarte gespeichert werden. Die Projektion der visuellen Schärfe des Blickes in die Szene wird in Abbildung 3.22 veranschaulicht und für eine bestimmte Pose der virtuellen Kamera, für welche die Szene gezeichnet wird, wie folgt realisiert: Für jeden Bildpunkt wird derjenige 3D-Punkt berechnet, der auf dieses Pixel abgebildet wird. Dies wird während des Zeichnens der Szene für die virtuelle Kamera durchgeführt. Dieser 3D-Punkt wird anschließend über seinen Sichtstrahl in das optische System des Auges projiziert. Dieses kann als virtuelle Kamera mit der Gauß-Textur als Bildebene betrachtet werden, wobei der Knotenpunkt des Auges und der 3D-Blickpunkt die optische Achse bilden. Der Schnittpunkt des Sichtstrahls mit der Gauß-Textur ergibt die Gewichtung. Dies wird über die Transformation des 3D-Punktes, für den es die Gewichtung zu berechnen gilt, in normalisierte Kamerakoordinaten des Auges und anschließend in Texturkoordinaten realisiert. Die Gewichtung wird für jedes Pixel berechnet und es entsteht pro Blickpunkt ein Bild der Gewichtungen, nachfolgend *Gewichtungsbild* genannt. Die Akkumulation der Gewichtungsbilder von allen Blickpunkten der betrachteten Zeitspanne wird ähnlich zu Duchowski et al. [Duc12] durchgeführt, ebenso die notwendige Normalisierung des akkumulierten Gewichtungsbildes und die Einfärbung über die Farbabbildungsfunktion. Abbildung 3.21 zeigt das Ergebnis der Heatmap-Erzeugung mit dem Projected Gaussians Verfahren. Dort lässt sich im unteren Bild erkennen, dass sich die visuelle Schärfe des Blickes aus Sicht des Auges nun gleichmäßig verteilt, während sie aus anderer Perspektive (oberes Bild) verzerrt erscheint.

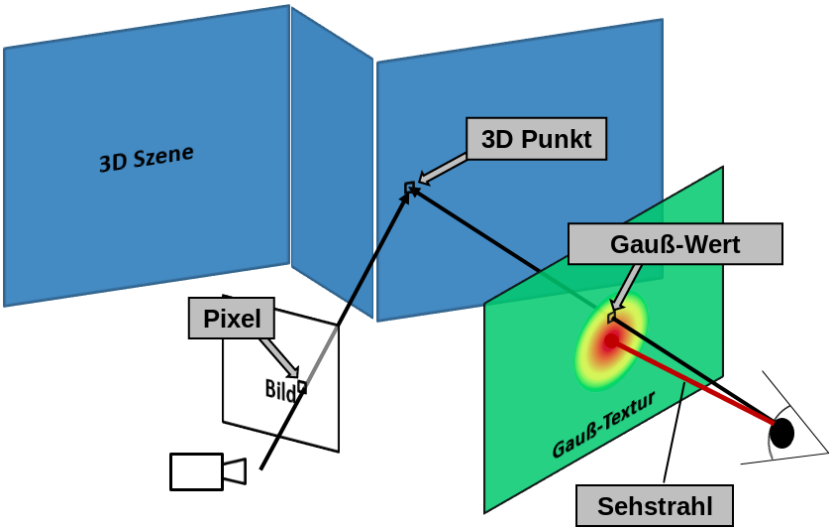


Abbildung 3.22: Darstellung der Projektion der visuellen Schärfe in die Szene

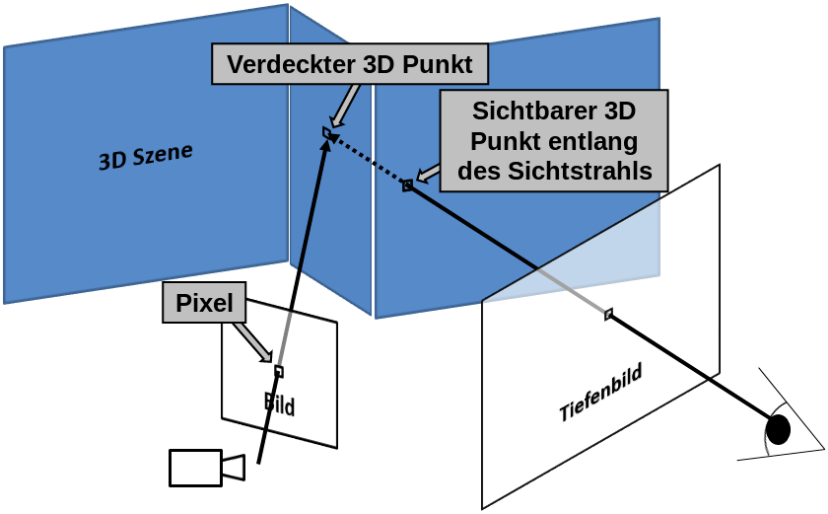


Abbildung 3.23: Darstellung der Nutzung von Tiefenkarten für die Berücksichtigung von Verdeckungen

3.2.2 Berücksichtigung von Verdeckungen

Das soeben beschriebene Verfahren hat den Nachteil, dass aus Sicht des Auges verdeckte Oberflächen aus der Perspektive des Auswerters des Blickverhaltens einsehbar sein können. Diese Bereiche können ebenfalls eingefärbt werden. Abbildung 3.24 zeigt im oberen Bild die Projektion der visuellen Schärfe in die Szene aus Sicht des Auges. Die mittlere Abbildung zeigt die Einfärbung von Bereichen, die für das Auge verdeckt sind. Um dieses Problem zu beseitigen, wird für jedes Pixel zusätzlich ein Verdeckungstest durchgeführt. Das Ergebnis ist im unteren Bild von Abbildung 3.24 zu sehen. Die verdeckten Bereiche erscheinen als Schatten der Heatmap.

Die Vorgehensweise der Verdeckungsberechnung ist in Abbildung 3.23 visualisiert und ist wie folgt in den Berechnungsprozess eingebettet: Nachdem die normalisierten Kamerakoordinaten des 3D-Punktes berechnet sind, wird aus einer zuvor für die Kamera des Auges gezeichnete Tiefenkarte die Tiefe ausgelesen. Dieser Tiefenwert stimmt mit der Entfernung des 3D-Punktes zur Kamera überein, falls er vom Auge aus sichtbar ist, ansonsten ist der Tiefenwert kleiner als die Entfernung des 3D-Punktes. Der Verdeckungstest kann also durch einen Lesezugriff auf eine weitere Textur sowie einen Vergleich von Entfernungen pro Pixel durchgeführt werden. Diese Methode wird *Shadow Mapping* genannt und geht auf Williams [Wil78] zurück.

Ein Beispiel für ein Tiefenbild ist in Abbildung 3.25a zu sehen. Da für jeden Blickpunkt ein Tiefenbild erzeugt und für eine schnelle Neuberechnung gespeichert werden muss, begrenzt der Grafikkartenspeicher die Anzahl der darstellbaren Blickpunkte. Zusätzlich beeinflusst die Auflösung des Tiefenbildes diese Anzahl. Je geringer die Auflösung gewählt wird, desto mehr Artefakte entstehen, siehe die Schattenkanten der Heatmap in Abbildung 3.25b. Bei einer Tiefenkarte mit einer Auflösung von 256×265 Pixeln und 4 Byte pro Gleitkommazahl für den Tiefenwert ergeben sich 256 KB Speicherbedarf pro Blickpunkt. Bei einem Grafikkartenspeicher von 4 GB und 30 Blickpunkten pro Sekunde ist ein Blickpfad von über neun Minuten an Blickdaten handhabbar. Zusätzlich kann der Grafikkartenspeicher auf den Hauptspeicher des

Rechners ausgeweitet werden. Des Weiteren können Methoden zur Kompression von Texturen sowie Erweiterungen für weiche Schattenkanten herangezogen werden.

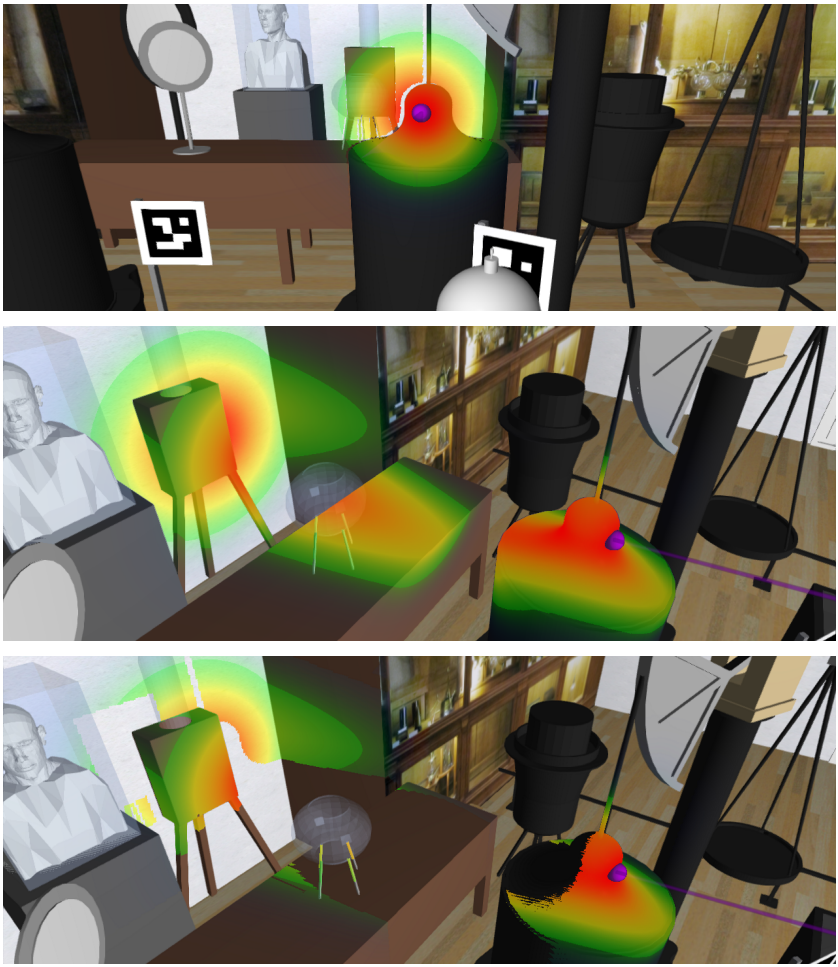
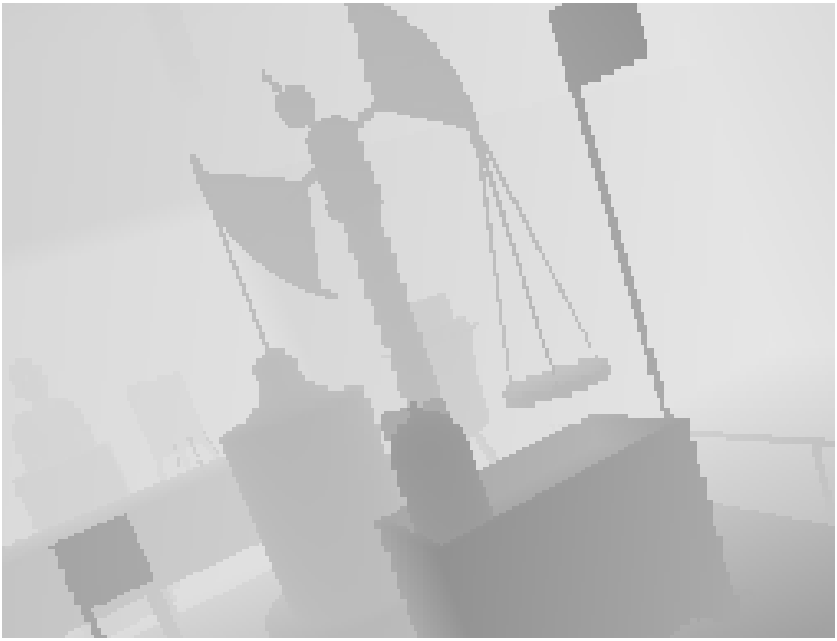
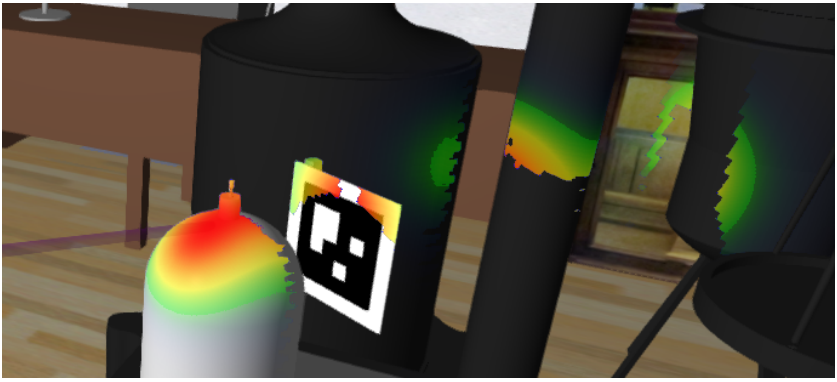


Abbildung 3.24: Beispiel für die Berücksichtigung von Verdeckungen



(a) Darstellung einer Schattenkarte bzw. Tiefenbildes



(b) Auswirkung der Auflösung der Tiefenkarte auf die Darstellung

Abbildung 3.25: Auswirkung der Auflösung von Schattenkarten auf die Visualisierung einer Heatmap

3.2.3 Begrenzung des für die Visualisierung relevanten Bereichs

Bei obigem Verfahren wird die Projektionsberechnung und der Verdeckungstest für jedes Pixel durchgeführt. Wenn man Abbildung 3.26 betrachtet und sich einen Sichtkegel vom Auge ausgehend in die Szene mit dem Sehstrahl als Rotationsachse denkt, der die Verteilung der Gewichte der visuellen Schärfe beschreibt, wird ersichtlich, dass nur ein kleiner Teil des gesamten Bildes überhaupt eine Gewichtung größer als null bekommen kann. Dieser Sichtkegel wird durch die dreidimensionale Darstellung des Sichtbereiches der virtuellen Kamera des Auges in Form eines Pyramidenstumpfes umgeben. Ein solcher Pyramidenstumpf ist in Abbildung 3.27 zu sehen. Nur rot gefärbte Pixel können eine Gewichtung durch die Projektion der visuellen Schärfe erhalten, weshalb zur Beschleunigung des Verfahrens nur diese Pixel betrachtet werden. Die Berechnung des roten Bereiches entspricht dem Zeichnen einer Pyramide, was aus Sicht des zeitlichen Aufwands vernachlässigt werden kann.

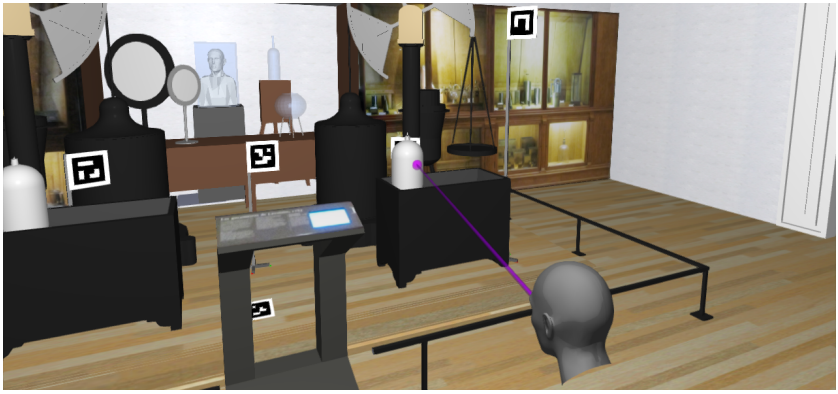


Abbildung 3.26: Visualisierung des Sehstrahls für einen Blickpunkt.

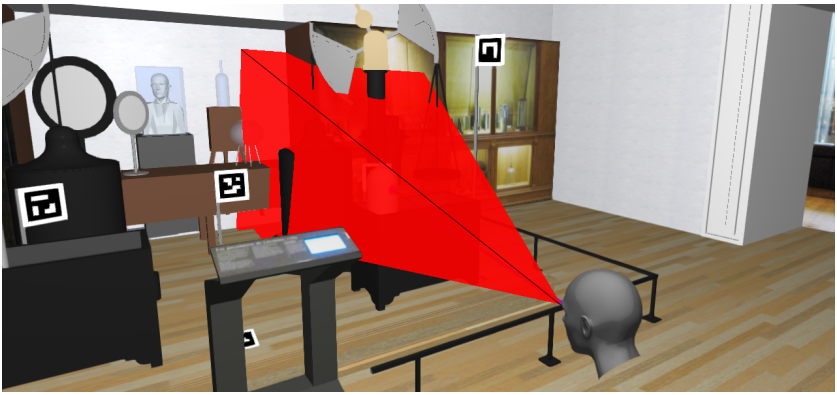


Abbildung 3.27: Visualisierung des Sichtkegels für die virtuelle Kamera des Auges als roten Pyramidenstumpf.

3.2.4 Wiederbenutzung vorheriger Berechnungen

Solange sich die Pose der virtuellen Kamera des Betrachters, die Perspektive, nicht ändert, kann das zuvor berechnete akkumulierte Gewichtungsbild weiterverwendet werden, da nur neue Blickpunkte hinzugefügt und Blickpunkte, die das betrachtete Zeitfenster verlassen, entfernt werden müssen. Die Perspektive der Augenkamera, also die Person mit dem Blickmessgerät, darf sich frei bewegen. Für die Berechnung des akkumulierten Gewichtungsbildes ist es folglich unerheblich, wie lang das betrachtete Zeitfenster ist. Bei Änderung der Perspektive muss das akkumulierte Gewichtungsbild allerdings komplett neu erstellt werden.

Eine Alternative könnte die Nutzung einer weiteren Textur für die Speicherung der Gewichte für alle Oberflächen sein. Dies würde die Wiederverwendung vorheriger Berechnungen auch bei einem Wechsel der Perspektive ermöglichen. Für den schnellen Zugriff auf die zusätzlichen Texturen könnte ein Textur-Atlas wie bei Stellmach et al. [Ste10b] verwendet werden.

3.2.5 Vergleich zum Stand der Technik

Verglichen mit dem Stand der Technik, siehe Abschnitt 2.2.7, ist das vorgestellte Verfahren das einzige, das die visuelle Schärfe in die Szene projiziert und damit am nächsten am physikalischen Prozess der Wahrnehmung bleibt. Kein zuvor vorgestelltes Verfahren berücksichtigt Verdeckungen oder ermöglicht eine von der Polygonstruktur unabhängige Einfärbung von Oberflächen. Im neuesten Verfahren zu dieser Thematik haben Pfeiffer und Memili [Pfe16] indes den obigen Ansatz zur Berücksichtigung von Verdeckungen aus der in dieser Arbeit entstandenen Publikation Maurus et al. [Mau14] übernommen und auch den dort getätigten Vorschlag zur Nutzung einer weiteren Textur zur Speicherung der Gewichtungen in die Tat umgesetzt. Im Gegensatz zum oben vorgestellten Projected-Gaussians-Verfahren nutzen Pfeiffer und Memili allerdings keine Projektion einer 2D-Gauß-Verteilung in die Szene, sondern 3D-Gauß-Verteilungen zentriert um die Blickpunkte, um Artefakten aus der Projektion entgegenzuwirken. Ein solches Artefakt ist bspw. in Abbildung 3.24 im untersten Bild zu sehen, wo der Blick von rechts auf das schwarze Objekt fällt und Objekte im Hintergrund (rechts im Bild neben der Büste) rot eingefärbt werden, obwohl diese in einer weiter entfernen Tiefe als der Fokusebene liegen. Aus dieser Sicht betrachtet sind 3D-Gauß-Verteilungen, platziert an den Blickpunkten, nicht so realistisch wie projizierte 2D-Gauß-Verteilungen, aber möglicherweise dem Verständnis des Blickverhaltens dienlicher, weil sie aus anderer Perspektive als der des Blickmessgerätes einfacher interpretiert werden können. Das Problem dieser Artefakte könnte durch eine einfache Erweiterung des Projected Gaussians Verfahren deutlich reduziert werden, indem nach der Berechnung des zum Pixel gehörenden 3D-Punktes der Abstand zum 3D-Blickpunkt zusätzlich berechnet wird. Wenn dieser Abstand einen Schwellenwert übersteigt, der linear mit der Entfernung zum Auge ansteigen muss, wird dem zugehörigen Pixel keine Gewichtung zugeteilt. Die Gewichtung würde sich, um näher an der Realität der Wahrnehmung zu

bleiben, weiterhin durch den projizierten Gauß-Wert ergeben. Diese wenigen Berechnungen würden auch die Laufzeit kaum beeinträchtigen. Aus anderer Perspektive kann durch die projektionsbasierte Einfärbung im Gegensatz zur Einfärbung mit 3D-Gauß-Verteilungen Richtungsinformation der Betrachtung übermittelt werden. Dies kann als weiterer Vorteil der projizierten 2D-Gauß-Verteilung gewertet werden.

4 Handregionsbestimmung aus der Egoperspektive auf monokularen Farbbildern

Dieser Abschnitt beschreibt die in der vorliegenden Arbeit entwickelten Verfahren zur Handregionsbestimmung auf monokularen Farbbildern aus der Egoperspektive. Zu Beginn wird in Abschnitt 4.1 die Evaluationsmethodik beschrieben. Dies beinhaltet die Beschreibung der verwendeten Datensätze für die Evaluation als auch die Methoden für die Auswertung. Abschnitt 4.2 befasst sich mit der Segmentierung der Hand durch die Erkennung von Hautfarbe bzw. der Hand mit unterschiedlichen Verfahren. Die Handpositionsbestimmung auf Basis einer Segmentierung wird in Abschnitt 4.3 behandelt. Abschnitt 4.4 zeigt eine Methode zur Fusion von Hautfarben- und Bewegungssegmentierungen zur Bestimmung des Vordergrundes. Die Bestimmung von Handregionshypothesen in Abschnitt 4.5 ist die Grundlage für zwei weitere Verfahren zur Handlokalisierung, die in den Abschnitten 4.6 und 4.7 präsentiert werden.

Abschnitt 4.8 stellt Verfahren zur simultanen Handregionsbestimmung und Posenschätzung vor. Sie sind gleichzeitig der neuste Beitrag der vorliegenden Arbeit. In der Evaluation, Abschnitt 4.9, werden die Verfahren zur Handregionsbestimmung auf dem am Fraunhofer-Institut für Optronik, Systemtechnik und Bildauswertung (IOSB) entstandenen IOSB-Hand-Tracking-Datensatz verglichen. Die Erkenntnisse aus den gesammelten Erfahrungen waren die Grundlage für die Entwicklung der Verfahren zur Handposenerkennung, die auf dem Datensatz EgoDexter mit dem Stand der Technik verglichen werden.

4.1 Evaluationsmethodik

4.1.1 Datensätze

4.1.1.1 IOSB-Hand-Tracking-Datensatz

Der IOSB-Hand-Tracking-Datensatz [Ham16b] entstand für die Erkennung von Wischgesten im Rahmen des Projektes ARtSENSE, siehe Abschnitt 1.2.1.5, und wird für die Beurteilung der Handpositionsbestimmung, siehe Abschnitt 4.1.2, genutzt. Er besteht aus 29 Sequenzen mit insgesamt 25.232 Einzelbildern, die aus der Egoperspektive an fünf verschiedenen Orten aufgenommen wurden. 24 Sequenzen entstammen einer Umgebung am IOSB, siehe Abbildungen 4.1a, 4.1b, 4.1d und 4.1e, wo im Hintergrund ein Poster zu sehen ist, das einen Angestellten abbildet, der in der Valencianischen Küche zu sehen ist, die in Abbildung 3.2 auf Seite 62 betrachtet werden kann. Diese 24 Sequenzen beinhalten zusammen 18.222 Einzelbilder. Die Hälfte davon wurde jeweils unter hellen (siehe Abbildungen 4.1a und 4.1b) als auch unter dunklen Beleuchtungsbedingungen (siehe Abbildungen 4.1d und 4.1e) aufgenommen. Die Bilder aus den Abbildungen 4.1c und 4.1f wurden im Labor von Lavoisier, siehe Abbildung 3.1, aufgenommen. Die verbleibenden drei Sequenzen, siehe Abbildungen 4.1g, 4.1h und 4.1i, wurden draußen unter Bäumen unter starkem Sonnenlichteinfall als auch in einer Wohnung und an einem holzfarbenen Schreibtisch aufgenommen. Die letzten fünf Sequenzen beinhalten zusammen 7.010 Einzelbilder. Alle Sequenzen haben eine Auflösung von 752×480 Pixeln, bis auf die Sequenzen aus den Abbildungen 4.1h und 4.1i, welche eine Auflösung von 1280×720 Pixeln aufweisen. Die Bilddaten enthalten Bewegungsunschärfe und teilweise Überbelichtung in der Szene, die draußen unter Bäumen und Sonnenlichteinfall aufgenommen wurde.

Da während der Entwicklung der Verfahren für die Handpositionsschätzung auffiel, dass es nicht trivial zu erkennen ist, ob die Hand noch im Kamerabild sichtbar ist, enthält der Datensatz zwischen den durchgeführten Gesten und Handbewegungen Abschnitte, in denen keine Hand zu sehen ist. Deshalb ist die Hand nur in 45 % der Bilder zu sehen. Von guten Verfahren für

die Handlokalisierung wird deshalb eine niedrige Falsch-Alarm-Rate erwartet. Dieser Sachverhalt wird in anderen Datensätzen wie dem nachfolgend beschriebenen EgoDexter-Datensatz nicht berücksichtigt. Dies liegt vermutlich daran, dass der Fokus bei diesem Datensatz auf der Handposenschätzung liegt, welche eine Hand im Bild voraussetzt. Wie bei EgoDexter ist auch im IOSB-Hand-Tracking-Datensatz immer nur eine Hand im Bild zu sehen.

(a) Zeigen mit der linken Hand

(b) ausgestreckte rechte Hand

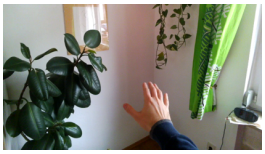
(c) Wischen von rechts nach links

(d) Greifen

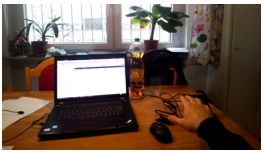
(e) Zeigen mit der rechten Hand

(f) Wischen von links nach rechts

(g) Wischen draußen unter Bäumen



(h) ganze Hand in der Raumecke



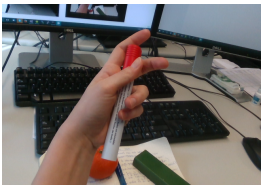
(i) rechte Hand am holzfarbenen Tisch

Abbildung 4.1: Beispielbilder des IOSB-Hand-Tracking-Datensatzes [Ham16b]. Einige Hände und Arme sind in dieser Abbildung zur besseren Sichtbarkeit rot umrandet, da man sie sonst nur schwer aufgrund des schwachen Lichtes oder der Ähnlichkeit zum Hintergrund erkennen könnte.

Die Annotationen der Grundwahrheit beinhalten für jedes Einzelbild, welches eine Hand enthält, das Zentrum der Handfläche als 2D-Position, siehe dazu

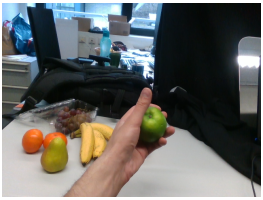
Abschnitt 4.1.2. Für die Evaluation werden 23.650 Bilder des Datensatzes genutzt. Bei 1.582 Bildern wurde manuell eine Segmentierung in Hintergrund, Hände und Arme erstellt. Sie werden für das Training zur Hautfarben- oder Handsegmentierung genutzt.

(a) Hand ausgestreckt



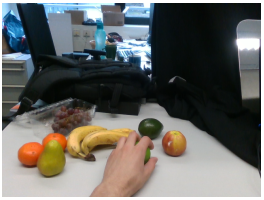
(b) Stift in Hand

(c) Gehaltener Holzblock



(d) Halten des grünen Apfels

(e) Hochhalten der Bananen



(f) Greifen des Apfels

(g) Zeitschaltuhr

(h) Bewegen von Stäbchen

(i) Papiertuch

(j) Ablegen eines Objektes

(k) Halten des Schwammes

(l) Bewegen einer Tasse

Abbildung 4.2: Beispielbilder des Datensatzes EgoDexter [Mue17]. Der Datensatz besteht aus vier Sequenzen. Jede Zeile entstammt einer Sequenz.

4.1.1.2 EgoDexter

Mueller et al. [Mue17] stellten den Datensatz *EgoDexter* vor, siehe Abbildung 4.2, der bereits in Abschnitt 2.3.4 beschrieben wurde und hier für die Beurteilung der 2D-Handposenschätzung aus der Egoperspektive, siehe Abschnitt 4.1.3, verwendet wird. Er besteht aus vier Sequenzen. Die oberste Zeile von Abbildung 4.2 zeigt Bilder der Sequenz *Desk* mit 551 Einzelbildern, die zweite Zeile zeigt die Sequenz *Fruits* mit 512 Einzelbildern, die dritte Zeile die Sequenz *Kitchen* mit 570 Einzelbildern und die unterste Zeile die Sequenz *Rotunda* mit 1.557 Einzelbildern. Zusammen kommt der Datensatz auf 3.190 Farb- und Tiefenbilder, von denen 1.485 bzgl. sichtbarer Fingerspitzen annotiert sind. Ergebnisse für die Evaluation der räumlichen Genauigkeit über den Anteil korrekt geschätzter Gelenkpositionen gibt es für die Verfahren von Zimmermann und Brox [Zim17], Mueller et al. [Mue18] sowie Iqbal et al. [Iqb18] mit welchen ein in dieser Arbeit entwickeltes Verfahren in Abschnitt 4.1.3 verglichen wird. Wie oben bereits erwähnt, beinhaltet dieser Datensatz maximal eine Hand im Bild.

4.1.2 Beurteilung der Handpositionsbestimmung

Für diesen Abschnitt wird die Handposition als das Zentrum der Handfläche definiert. Abbildung 4.3 stellt die Handposition durch einen schwarzen Kreis dar. Die Beurteilung der Handposition wird ausschließlich für den IOSB-Hand-Tracking-Datensatz durchgeführt, welcher in Abschnitt 4.1.1.1 beschrieben ist. Dieser beinhaltet für jedes Einzelbild mit sichtbarer Hand eine zugehörige Annotation der Handposition als 2D-Position. Durch die manuelle Annotation desselben Datensatzes von verschiedenen Personen konnte eine durchschnittliche Abweichung von ca. sieben Pixeln bei den resultierenden Trajektorien festgestellt werden. Eine Abweichung von sieben Pixeln bei der Positionsschätzung ist also ähnlich gut wie eine menschliche Schätzung der Handposition. In Abbildung 4.3 ist ein grüner Kreis dargestellt, welcher einen Radius mit 25 Pixeln visualisiert. Eine Schätzung der Handposition mit einer Distanz zur Grundwahrheit von weniger als 25 Pixeln

liegt folglich innerhalb des grünen Kreises und kann ebenfalls als sehr gute Schätzung betrachtet werden.

Abbildung 4.3: Vergrößerte Darstellung einer Hand mit annotierter Handposition im Zentrum der Handfläche (schwarzer Kreis). Der grüne Kreis hat hier einen Radius von 25 Pixeln. Abweichungen von weniger als 25 Pixeln liegen folglich innerhalb des grünen Kreises und können als räumlich sehr genaue Schätzung der Handposition betrachtet werden.

Für die Auswertung von Einzelbildern, in denen eine Hand sichtbar ist und in denen eine Schätzung für die Handposition vorliegt, lässt sich die Abweichung zur Grundwahrheit berechnen. Für eine bestimmte erlaubte Abweichung weist eine korrekt geschätzte Handposition eine Distanz zur Grundwahrheit kleiner oder gleich dieser Abweichung auf. Für jede erlaubte Abweichung berechnet sich so die Anzahl korrekt geschätzter Hände, welche in Relation zur Anzahl aller vorkommenden Hände die Trefferquote oder anders ausgedrückt den Anteil korrekt geschätzter Hände (2D-PCH) (engl. Percentage of Correct Hands) in Abhängigkeit dieser maximal erlaubten Abweichung ergibt. Diese Trefferquote in Abhängigkeit der erlaubten Abweichung kann als Kurve dargestellt werden, um die Handpositionsbestimmung mehrerer Verfahren zu beurteilen. Eine Handposition mit Distanz zur Grundwahrheit kleiner oder gleich der erlaubten Abweichung ist eine richtige Positivschätzung, ansonsten eine falsche Positivschätzung. Je kleiner die erlaubte Abweichung, desto mehr falsche Positivschätzung resultieren. Die Kurve wird für eine maximal erlaubte Abweichung von 50 Pixeln dargestellt, da die Handfläche

in Abbildung 4.3 einen Kreis mit einem Radius von 50 Pixeln noch einschließen würde. Die Handfläche ist hier aber auch in der größtmöglichen Ausdehnung zu sehen. Unberücksichtigt lässt die Kurve die Positivschätzungen mit Abweichungen von über 50 Pixeln und Positivschätzungen, wenn keine Hand im Bild zu sehen ist. Letztere sind auch bei einer unendlich großen erlaubten Abweichung falsche Positivschätzungen. Wird die erlaubte Abweichung auf unendlich gesetzt, dienen Maßzahlen wie die Trefferquote, auch Richtig-Positiv-Rate (TPR) genannt, die Falsch-Positiv-Rate (FPR), die Genauigkeit (PREC), das F1-Maß (F1) sowie die Korrektklassifikationsrate (ACC) als weitere Bewertungskriterien. Für eine genauere Betrachtung der Entwicklung der Trefferquote unter Einfluss der räumlichen Genauigkeit der Schätzung wird die 2D-PCH-Kurve sowie ihre Fläche unter der Kurve (AUC) herangezogen.

4.1.3 Beurteilung der Handposenschätzung

Zur Beurteilung der in dieser Arbeit entwickelten 2D-Handposenschätzung wird der EgoDexter-Datensatz verwendet. Auf ihn wurde oben in Abschnitt 4.1.1.2 genauer eingegangen. Seine Grundwahrheit beinhaltet für sichtbare Fingerspitzen die 2D-Positionen. Pro Einzelbild gibt es, da maximal eine Hand zu sehen ist, maximal fünf 2D-Positionen, die es zu finden gilt. Bestimmt ein Verfahren eine 2D-Handpose für ein Bild mit einer annotierten Grundwahrheit, gibt es zu jeder 2D-Position eine korrespondierende 2D-Schätzung, zwischen welchen eine Abweichung berechnet werden kann. Die so entstehenden maximal fünf Abweichungen pro Einzelbild können genauso wie im vorherigen Abschnitt durch den Einbezug einer maximal erlaubten Abweichung in richtige und falsche Positivschätzungen unterteilt werden. Diese Teilung erlaubt die Berechnung der Trefferquote oder des Anteils korrekt geschätzter Gelenkpositionen (engl. Percentage of Correct 2D-Keypoints, 2D-PCK). Auch hier kann eine Kurve gezeichnet werden, die dem besseren Vergleich der räumlichen Genauigkeit der Schätzungen von Verfahren dient.

4.2 Segmentierung der Hand

4.2.1 Erkennung von Hautfarbe

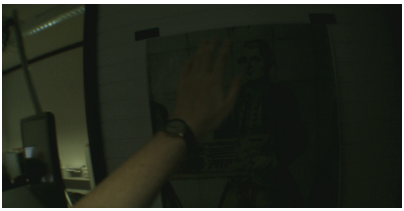
Die Erkennung von Hautfarbe ist im Bereich der Gesichtserkennung aufwendig untersucht worden. Für die pixelweise Klassifikation muss zum einen ein Farbraum gewählt werden und zum anderen eine Repräsentationsform der Hautfarbe. In anderen Arbeiten [Kak07, Phu05] konnte gezeigt werden, dass dreidimensionale gegenüber zweidimensionalen Farbräumen zu bevorzugen sind. Ob der RGB- oder der HSV-Raum zu bevorzugen ist, bleibt offen. Für die Modellierung von Hautfarbe können sowohl parametrische Modelle wie Gauß-Verteilungen oder Gauß-Mischverteilungen als auch nicht parametrische Modelle wie Histogramme genutzt werden. Mit Histogrammen wird keine Annahme über die Verteilung der Hautfarbe vorausgesetzt, während Gauß-Verteilungen eine nicht notwendigerweise vorhandene Normalverteilung implizieren. Gauß-Mischverteilungen können sich besser nicht normalverteilten Farbbereichen anpassen, aber die Anzahl der Verteilungen und deren Gewichtungsfaktoren müssen ermittelt werden. Für das Training werden hautfarbene Bereiche in Bildern annotiert und entweder das Histogramm befüllt oder eine Gauß-Verteilung bzw. Gauß-Mischverteilung berechnet. Gauß-Mischverteilungen wurden von Jones und Rehg [Jon99] für eine Datenbasis aus mehreren tausend Bildern berechnet. Es zeigte sich, dass die Verteilungen gut generalisieren können, aber in bestimmten Situationen nicht akkurat klassifizieren. Auf den in dieser Arbeit verwendeten Daten zeigten die ermittelten Gauß-Mischverteilungen von Jones und Rehg keine brauchbaren Resultate, weshalb die Wahl auf einen Bayes-Klassifikator mit einem Schwellenwert, vorgestellt in Pung et al. [Phu05], fiel. 3D-Histogramme im RGB-Raum zeigten in der vorliegenden Arbeit die besten Ergebnisse, ein Beispiel ist in Abbildung 4.4 zu sehen. Dennoch konnte auf dem IOSB-Hand-Tracking-Datensatz festgestellt werden, dass ein statisches Hautfarbenmodell nicht ausreichend ist. Eine dynamische Anpassung könnte helfen, würde aber eine perfekte Segmentierung in vorangegangenen Einzelbildern benötigen. Ansonsten würden Teile des Hintergrundes durch falsche Segmentierungen

in die Repräsentation der Hautfarbe gelangen und die Handsegmentierung würde nach wenigen Einzelbildern fehlschlagen.

Wie in Abschnitt 2.3.2 beschrieben, stellten Li und Kitani [Li13a] einen auf Random Decision Forests (RDFs) beschriebenen Klassifikator vor, der eine pixelweise Klassifikation durchführt. Dafür wird die lokale Nachbarschaft eines Pixels mitberücksichtigt und aus einer Datenbank an gelernten Umgebungsbedingungen die Menge intern genutzter RDFs bestimmt, die den vorliegenden Umgebungsbedingungen am meisten ähneln. Dieses Verfahren wird in der vorliegenden Arbeit ebenfalls für die Segmentierung von Händen verwendet. Beispiele für die Segmentierung der Bilder 4.5a, 4.6a, 4.7a und 4.8a finden sich in den Abbildungen 4.5b, 4.6b, 4.7b und 4.8b. Generell ist bei Verfahren für die Segmentierung von Hautfarbe zu beachten, dass nicht nur die Hand, sondern auch Gesichter oder Arme, wie in Abbildung 4.5b oder 4.8b, segmentiert werden. In Abbildung 4.6b sowie 4.8b ist zu erkennen, dass viele Bereiche im Hintergrund als Hautfarbe klassifiziert werden. Ein bekanntes Problem ist die Klassifikation von holzfarbenen Bereichen als Hautfarbe, siehe Abbildung 4.8b auf dem Tisch. Die Generalisierungsfähigkeit des Klassifikators ist als nicht gut zu bewerten, da die Segmentierung bei der Sequenz Rotunda aus dem EgoDexter-Datensatz, für den keine RDFs trainiert wurden, keine gute Segmentierung des Unterarms erzielt, da im Wesentlichen der Schatten des Armes auf dem Tisch erkannt wird, siehe Abbildung 4.8b. Hier wurde bei anderen Einzelbildern der Sequenz teilweise der gesamte holzfarbene Tisch als Hautfarbe erkannt.

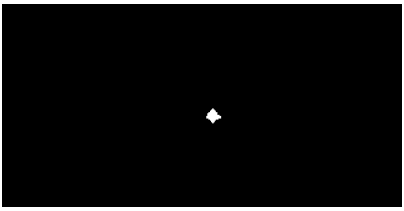
(a) Eingabebild **(b)** Hautfarbensegmentierung

Abbildung 4.4: Beispiel der Hautfarbensegmentierung mit RGB-Histogramm



(a) Eingabebild

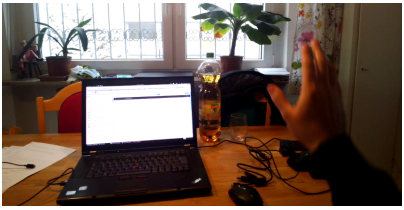
(b) RDFs



(c) Standard-HandSegNet

(d) IOSB-HandSegNet

Abbildung 4.5: Vergleich der Hautfarben- bzw. Handsegmentierung auf einer Sequenz des IOSB-Hand-Tracking-Datensatzes



(a) Eingabebild

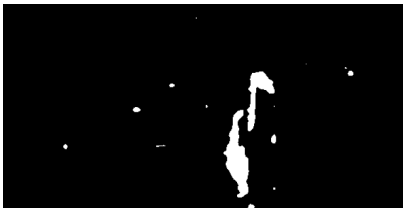
(b) RDFs

(c) Standard-HandSegNet

(d) IOSB-HandSegNet

Abbildung 4.6: Vergleich der Hautfarben- bzw. Handsegmentierung auf der *gg_go_on*-Sequenz des IOSB-Hand-Tracking-Datensatzes

(a) Eingabebild



(b) RDFs

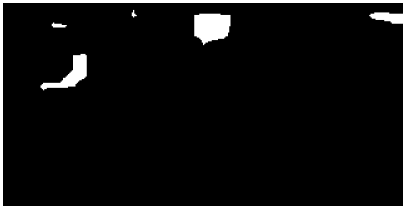
(c) Standard-HandSegNet

(d) IOSB-HandSegNet

Abbildung 4.7: Vergleich der Hautfarben- bzw. Handsegmentierung auf der *underTrees*-Sequenz des IOSB-Hand-Tracking-Datensatzes

(a) Eingabebild

(b) RDFs



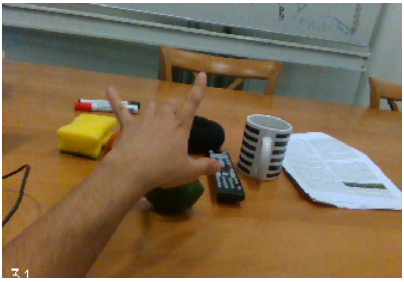
(c) Standard-HandSegNet

(d) IOSB-HandSegNet

Abbildung 4.8: Vergleich der Hautfarben- bzw. Handsegmentierung auf der *Rotunda*-Sequenz des EgoDexter-Datensatzes

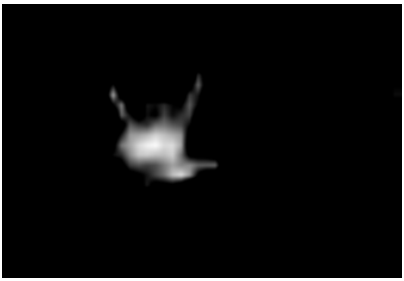
4.2.2 Segmentierung mit dem CNN HandSegNet

Zimmermann und Brox [Zim17] entwickelten das CNN HandSegNet (HSN), siehe Abschnitt 2.3.2, welches für ein Eingabebild ein Hand-Konfidenzbild und ein Hintergrund-Konfidenzbild erzeugt. Diese Ausgabebilder enthalten pro Pixel einen Wert, der die Zufriedenheit mit der Aussage „ist Handpixel“ bzw. „ist Hintergrund“ angibt. Abbildung 4.9c und 4.9b zeigen Beispiele für solche Konfidenzbilder für das Eingabebild aus Abbildung 4.9a. Zur Darstellung ist zu berücksichtigen, dass alle negativen Konfidenzen auf null gesetzt wurden, und damit schwarz erscheinen. Die Segmentierung wird wie folgt erzeugt: Jedes Pixel im Hand-Konfidenzbild, das einen Wert größer als null aufweist, wird weiß, ansonsten schwarz gefärbt. Abbildung 4.9d zeigt die resultierende Handsegmentierung.



(a) Eingabebild

(b) Hintergrund-Konfidenzbild



(c) Hand-Konfidenzbild

(d) Segmentierung

Abbildung 4.9: Beispiel der Handsegmentierung mit HandSegNet

Weil das Standard-HandSegNet keine guten Ergebnisse auf dem IOSB-Hand-Tracking-Datensatz erzielte, wurde es in der vorliegenden Arbeit durch ein zusätzliches Training verändert. Der Datensatz *EgoHands* [Bam15] sowie der in Abschnitt 4.1.1.1 bereits vorgestellte IOSB-Hand-Tracking-Datensatz wurden hierfür verwendet. Das resultierende CNN wird nachfolgend IOSB-HandSegNet genannt. Beispiele für die Handsegmentierung sind in den Abbildungen 4.5, 4.6, 4.7 und 4.8 für Einzelbilder aus unterschiedlichen Sequenzen zu sehen. Links oben ist jeweils das Eingabebild, rechts oben die Segmentierung mit RDFs nach Li und Kitani [Li13a], links unten die Handsegmentierung mit dem Standard-HandSegNet und rechts unten mit dem IOSB-HandSegNet zu sehen. Es ist bei allen vier Beispielen deutlich die Überlegenheit des IOSB-HandSegNet gegenüber dem Standard-HandSegNet als auch der RDF-basierten Segmentierung zu erkennen. Letztere erzeugt sehr viele Fehlschätzungen im Hintergrund oder deckt die Hand nicht ab, wie in Abbildung 4.8b zu sehen. Das Standard-HandSegNet findet die Hand teilweise gar nicht oder deckt sie nur schlecht ab. Mit dem IOSB-HandSegNet wird fast nur die Hand erkannt und diese gut abgedeckt. Interessant ist dies besonders bei der Rotunda-Sequenz des EgoDexter-Datensatzes, weil es auf diesen Daten nicht trainiert wurde. Das zusätzliche Training des HandSegNet, wie oben beschrieben, macht hier den Unterschied aus und steigert die Generalisierungsfähigkeit.

4.3 Handpositionsbestimmung auf Basis einer Segmentierung

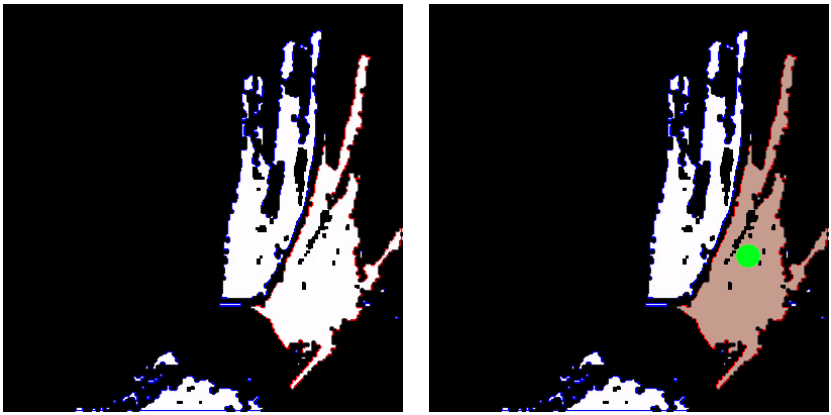
Die Bestimmung der Handposition auf Basis einer Segmentierung wird in dieser Arbeit auf zwei Weisen durchgeführt, welche in den nachfolgenden Unterabschnitten 4.3.1 und 4.3.2 behandelt werden. Sie wurden zudem in Hammer und Beyerer [Ham13b] vorgestellt.

4.3.1 Regionsbasierte Handpositionsbestimmung

Die regionsbasierte Handpositionsbestimmung führt auf einer Segmentierung zuerst eine Suche nach zusammenhängenden Regionen durch. Die größte dieser Regionen wird als Hand angenommen und der Schwerpunkt bestimmt, solange diese Region größer ist als ein gewisser Schwellenwert, 2.000 Pixel, was ungefähr dem Flächeninhalt des grünen Kreises aus Abbildung 4.3 mit 25 Pixeln Radius entspricht. Dieses Verfahren wird *Schwerpunkt-Tracking* genannt. Die Abbildungen 4.10 und 4.11 veranschaulichen das Schwerpunkt-Tracking anhand zweier Beispiele. Das zweite Beispiel zeigt die Problematik, wenn die Hand durch zwei getrennte Regionen beschrieben ist und die größte zusammenhängende Region nur einem Teil der Hand entspricht. Ergebnis ist, dass die Handposition vom Zentrum der Handfläche weggezogen wird, siehe Abbildung 4.11b.

(a) **(b)**

Abbildung 4.10: Darstellung des Schwerpunkt-Tracking-Verfahrens für eine Beispielsegmentierung (links). Rechts daneben ist die größte zusammenhängende Region (beige) und ihr Schwerpunkt (grün) markiert.



(a) (b)

Abbildung 4.11: Darstellung des Schwerpunkt-Tracking-Verfahrens für eine weitere Beispielsegmentierung (links) mit nicht optimal segmentierter Hand. Rechts daneben ist die größte zusammenhängende Region (beige) und der vom Handzentrum abweichende Schwerpunkt (grün) zu sehen.

4.3.2 Handpositionsbestimmung mit Partikelfilter

Um dem Problem von nicht perfekten Handsegmentierungen entgegenzuwirken, wird in der vorliegenden Arbeit das Partikelfilter von Isard and Blake [Isa98] genutzt. Das Partikelfilter kann als stochastisches Tracking-Verfahren genutzt werden, welches einen Status, ein Bewegungs- und ein Beobachtungsmodell benötigt. Ein einfacher Status besteht aus der aktuellen 2D-Position im Bild und das Bewegungsmodell aus einem zufällig gewählten 2D-Verschiebungsvektor. Für das Beobachtungsmodell kann bspw. die lokale Nachbarschaft der Position in der Segmentierung betrachtet werden. Je mehr Pixel in der lokalen Nachbarschaft eines Partikels als Haut klassifiziert werden, desto höher ist das Gewicht des Partikels. Seine Bewegung bestimmt die Position im nächsten Einzelbild. Das Verfolgen einer Hand mit diesem Partikeltyp wird nachfolgend *Std-Partikelfilter-Tracking* genannt und nimmt eine Hand als gefunden an, wenn die durchschnittliche Anzahl $\bar{s}_{skin}$ an segmentierten Pixeln in der lokalen quadratischen Nachbarschaft von 50×50 Pixeln der Partikel größer ist als ein Schwellenwert t_{skin} von 750. Dieser

Schwellenwert wurde fein säuberlich evaluiert und sorgt für wenige Falsch-Positiv-Schätzungen. Bei einer Reduktion auf bspw. 100 würden deutlich mehr Richtig-Positiv- aber auch Falsch-Positiv-Schätzungen erfolgen. Gute Partikel konzentrieren sich auf der segmentierten Region und bewegen sich ähnlich wie diese. Abbildung 4.12 zeigt die Konzentration von Partikeln auf der Segmentierung. Die Handposition wird pro Dimension aus dem Median aller Partikelpositionen gebildet. Deshalb sollten so viele Partikel wie möglich auf der Hand und nicht dem Arm liegen.

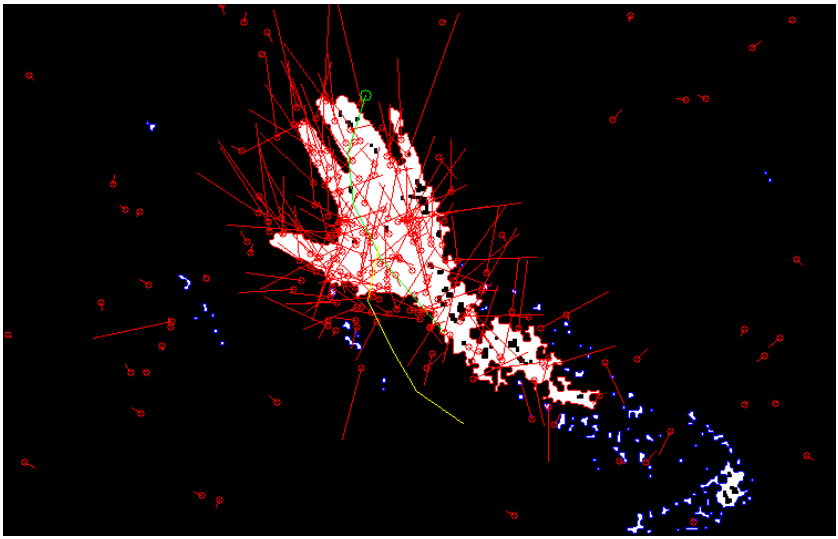
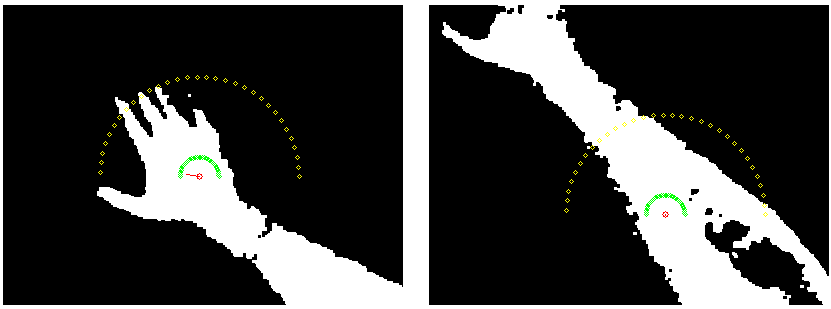


Abbildung 4.12: Darstellung der Konzentration von Partikeln auf der Handregion. Die kleinen roten Kreise stellen die Position der Partikel dar und die vom Kreis ausgehende Linien die Verschiebungsvektoren.

Um Partikel auf die Hand zu drängen, wurde das *Shape*-Partikel entwickelt, welches in Abbildung 4.13 dargestellt ist. Sein Gewicht berechnet sich aus der Summe aller weißen Pixel unter dem kleinen grünen Halbkreis verringert um die weißen Pixel unter dem größeren gelben Halbkreis. Folglich hat das Partikel in Abbildung 4.13a ein höheres Gewicht als in Abbildung 4.13b, weil das Partikel in Letzterem den Arm mit seinem gelben größeren Halbkreis schneidet. Dieses Beobachtungsmodell macht nur Sinn, wenn die Hände von unten

ins Bild kommen und die Finger am weitesten oben im Bild sind, was bei Zeigegesten und Wischgesten aus der Egoperspektive üblicherweise der Fall ist. Das Verfolgen einer Hand mit diesem Partikeltyp wird nachfolgend *Shape-Partikelfilter-Tracking* genannt und nimmt ebenfalls eine Hand als gefunden an, wenn die Anzahl an segmentierten Pixeln in der lokalen quadratischen Nachbarschaft mit einer Breite von 50 Pixeln bei allen Partikeln durchschnittlich einen Schwellenwert von 750 Pixeln übertrifft.

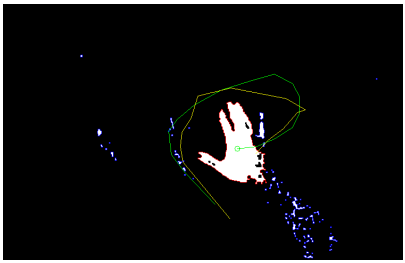


(a) Partikel mit hohem Gewicht **(b)** Partikel mit niedrigem Gewicht

Abbildung 4.13: Darstellung des Shape-Partikels für das Partikelfilter-Tracking. Das linke Bild zeigt ein Partikel mit höherem Gewicht als das Partikel im rechten Bild, bei dem der äußere Kreis den Arm schneidet.

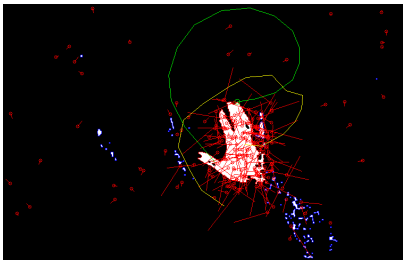
Beispiele zum Vergleich der Verfahren für die Bestimmung der Handposition werden in den Abbildungen 4.14 und 4.15 vorgestellt. Die linke Spalte stellt jeweils das Eingangsbild als auch die resultierende Handposition als grünen Punkt dar. Die rechte Spalte visualisiert die Berechnungen des jeweiligen Verfahrens. In der oberen Zeile wird das Schwerpunkt-Tracking, in der mittleren das Std-Partikelfilter-Tracking und in der unteren Zeile das Shape-Partikelfilter-Tracking mit jeweils 500 Partikeln gezeigt. Der Unterschied der Konzentration der Std-Partikel und der Shape-Partikel wird gut im direkten Vergleich von Abbildung 4.15d und 4.15f ersichtlich. Nur die Shape-Partikel konzentrieren sich auf der Handfläche.

(a) Eingabebild und Resultat Schwerpunkt-Tracking



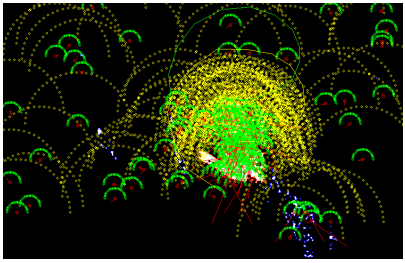
(b) Visualisierung Schwerpunkt-Tracking

(c) Eingabebild und Resultat Std-Partikelfilter-Tracking



(d) Visualisierung Std-Partikelfilter-Tracking

(e) Eingabebild und Resultat Shape-Partikelfilter-Tracking



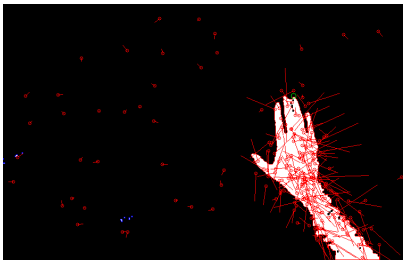
(f) Visualisierung Shape-Partikelfilter-Tracking

Abbildung 4.14: Beispiele der Handpositionsbestimmung auf einer Segmentierung, bei der hauptsächlich die Hand segmentiert ist. Die resultierenden Handpositionen der verglichenen Verfahren sind ähnlich gut mit einer Abweichung zwischen 25 und 50 Pixeln.

(a) Eingabebild und Resultat Schwerpunkt-Tracking

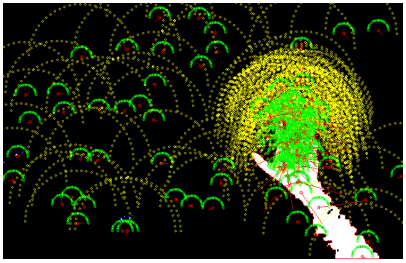
(b) Visualisierung Schwerpunkt-Tracking

(c) Eingabebild und Resultat Std-Partikelfilter-Tracking



(d) Visualisierung Std-Partikelfilter-Tracking

(e) Eingabebild und Resultat Shape-Partikelfilter-Tracking



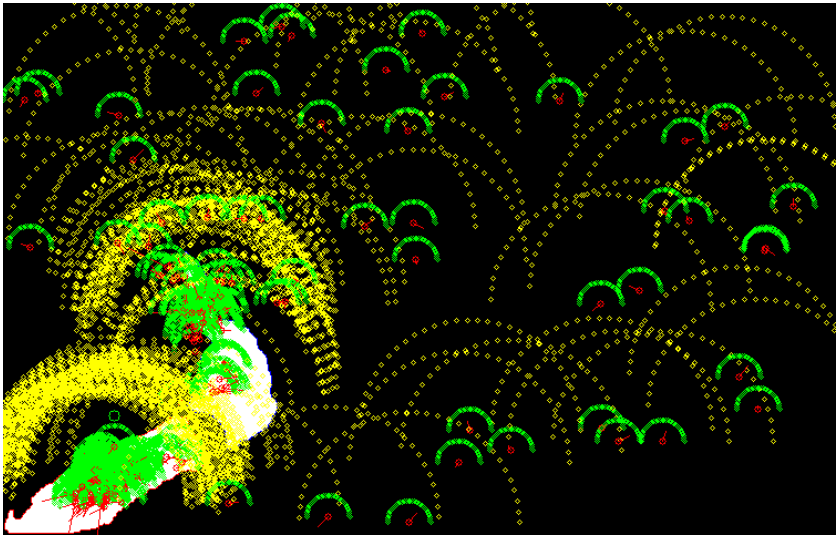
(f) Visualisierung Shape-Partikelfilter-Tracking

Abbildung 4.15: Beispiel der Handpositionsbestimmung auf einer Segmentierung, bei der zusätzlich zur Hand der Unterarm segmentiert wurde. Zu beobachten ist, dass bei den ersten beiden Verfahren die Handposition durch die zusätzliche Segmentierung des Unterarms nach unten gezogen wird. Das Shape-Partikelfilter-Tracking bleibt von diesem Problem unberührt, schätzt im vorliegenden Fall die Handposition allerdings mit ähnlich großer Abweichung nach oben in Richtung des Mittelfingers.

In Abbildung 4.16 ist zu sehen, dass es Segmentierungen gibt, die dem Shape-Partikelfilter-Tracking Probleme bereiten. Dort bewirkt die Armbanduhr eine Trennung der Segmentierung und die Partikel konzentrieren sich auf dem Unterarm, siehe Abbildung 4.16c. Die Position wird entsprechend falsch geschätzt, siehe Abbildung 4.16a.

(a) Eingabebild mit Positionsschätzung

(b) Segmentierung



(c) Konzentration der Shape-Partikel

Abbildung 4.16: Beispiel einer falschen Positionsschätzung mit dem Shape-Partikel

4.4 Handlokalisierung mit MACS

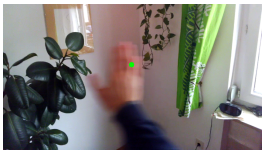
Beim Vergleich der Methoden zur Hautfarbensegmentierung aus dem Abschnitt 4.2.1 konnte festgestellt werden, dass diese Schwächen aufweisen und auf dem IOSB-Hand-Tracking-Datensatz keine guten Ergebnisse liefern, weil die Segmentierung der Hand nicht robust funktioniert und viele andere Bereiche des Hintergrundes fälschlicherweise klassifiziert werden, siehe die Hautfarbensegmentierung in Abbildung 4.17g. Aus diesem Grund wurde das in Hammer et al. [Ham16b] vorgestellte Verfahren *Motion segmentation and Appearance Change detection based Skin color detection (MACS)* entwickelt. Dieses nutzt den optischen Fluss als weiteres Hilfsmittel zur Bestimmung von Vordergrundobjekten in der Szene und führt die Segmentierung einer sich bewegenden Vordergrundregion durch, in der die Hand angenommen wird. Dieses Vordergrundsegment wird durch die Beobachtung seines Erscheinungsbildes bewertet, um sowohl robust gegenüber schlechten Flussberechnungen an Bildrändern zu sein als auch das Segment weiterhin im Auge zu behalten, wenn sich die Hand nicht bewegt, weil sich dann kein sich bewegender Vordergrund bestimmen lässt. Diese Erkennung der Veränderung des Erscheinungsbildes wird in Abschnitt 4.4.2 vorgestellt, nachdem in Abschnitt 4.4.1 die Bestimmung der sich bewegenden Vordergrundregion erklärt wird. Die Fusion dieser Bewegungssegmentierung mit der Hautfarbensegmentierung von Li und Kitani [Li13a] aus Abschnitt 4.2.1 wird in Abschnitt 4.4.3 beschrieben.

4.4.1 Segmentierung des Vordergrundes

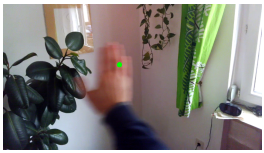
Abbildung 4.17a zeigt ein Bild aus der egozentrierten Szenenkamera zum Zeitpunkt $t-1$ und Abbildung 4.17b das nachfolgende Bild zum Zeitpunkt t. Der optische Fluss, geschätzt mit dem Verfahren von Zach et al. [Zac07], ist in Abbildung 4.17c mit der Farbdarstellung aus Baker et al. [Bak11] dargestellt. Die Farbe eines Pixels bestimmt bei dieser Farbdarstellung die Richtung des 2D-Verschiebungsvektors und die Sättigung den Betrag des Vektors. In den Bildern ist folglich eine Bewegung der Hand und des Unterarms nach links zu sehen, während sich der Hintergrund nicht bewegt.

Zur Segmentierung des Verschiebungsvektorfeldes wird die Ballungsanalyse *k-Means* genutzt. Zur Einfachheit wird jedes Ballungszentrum durch einen 2D-Vektor repräsentiert und kein affines Bewegungsmodell genutzt. Auch werden nur zwei Ballungszentren bestimmt, damit der Hintergrund und das größte Objekt im Vordergrund gefunden werden, welches in den meisten Fällen aus Hand und Unterarm besteht. Zur Bestimmung des Vorder- und Hintergrundes wird der Betrag des Differenzvektors aus den zwei Ballungszentren bestimmt. Die erste Bedingung für die Erkennung des Vordergrundes ist, dass sich dieser vom Hintergrund abhebt, der Betrag des Differenzvektors also einen gewissen Wert übersteigt. Um robust gegenüber falschen Schätzungen im Verschiebungsvektorfeld zu sein, nutzen wir einen Schwellenwert von ca. 1 % (vier Pixeln) der Bildhöhe (480 Pixel), der bei annähernd statischen Szenen ausreicht, um kein Vordergrundobjekt zu ermitteln. Zusätzlich lassen sich bei natürlichem Abstand von der Hand zur Kamera Bedingungen an die Größe der Handregion knüpfen. Diese sollte wegen teils schlechten Flussfeldsegmentierungen mindestens die Hälfte der Handfläche (ca. 5.000 Pixel) und aufgrund schneller Bewegungen maximal ein Viertel der gesamten Bildfläche (ca. 100.000 Pixel) betragen. Wenn der Betrag des Differenzvektors groß genug ist und eine Region die Bedingungen an die Größe erfüllt, wird angenommen, dass in dieser die Hand ist. Das Ergebnis der Segmentierung des Flussfeldes aus Abbildung 4.17c ist in Abbildung 4.17d zu sehen.

Bei der Nutzung des optischen Flusses ist weiter zu beachten, dass das Verschiebungsvektorfeld den vorwärts gerichteten Fluss von $t - 1$ nach t beschreibt, wenn Einzelbild t zur Verfügung steht. Die aus der Ballungsanalyse resultierende Bewegungssegmentierung liegt deshalb einen Zeitschritt zurück in der Vergangenheit und wird mit Hilfe des Flussfeldes in den aktuellen Zeitpunkt t verschoben. Ein derart bewegtes Segment ist in Abbildung 4.17e zu sehen, das aus dem Segment aus Abbildung 4.17d entstanden ist. Die Risse entstehen durch die Verschiebung, was unbeachtet bleiben kann. Das resultierende Vordergrundsegment ist in Abbildung 4.17f maskiert aus dem Farbbild dargestellt.



(a) Zeitpunkt $t - 1$

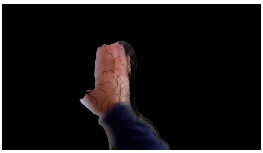


(b) Zeitpunkt t

(c) Optischer Fluss

(d) Flussfeldsegmentierung

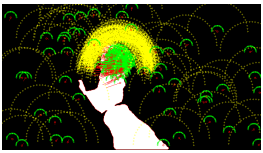
(e) verzerrte Flussfeldsegmentierung



(f) maskiertes Farbbild

(g) Hautfarbensegmentierung

(h) fusionierte Segmentierung



(i) Shape-Partikelfilter-Tracking

Abbildung 4.17: Überblick Verfahren MACS

4.4.2 Erkennung der Veränderung des Erscheinungsbildes der Handregion

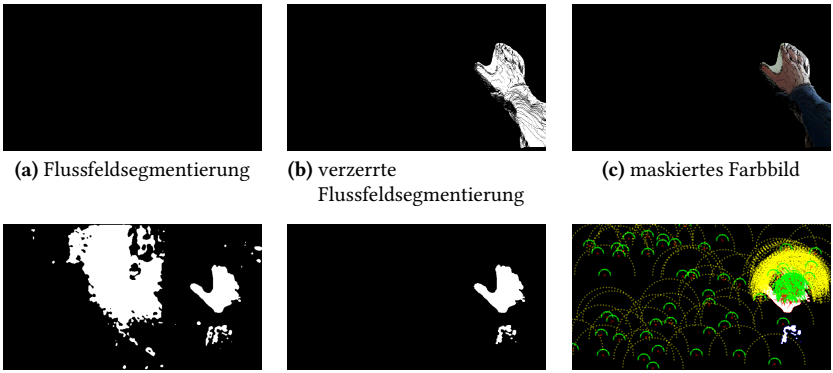
Problematisch wird die Segmentierung der Hand über das Flussfeld in drei Fällen: 1. Die Hand bewegt sich nicht oder nur sehr wenig und weist dadurch annähernd die gleiche Bewegung auf wie der Hintergrund. Bei der Ballungsanalyse entsteht dann eine Ballung, zu der fast alle Pixel im Bild gehören. 2. Die Hand verlässt das Bild und ist nur noch zu einem kleinen Teil zu sehen. Dann werden die Anforderungen an die Größe der Vordergrundregion nicht erfüllt und kein Vordergrundsegment bestimmt. 3. Das dritte Problem entsteht ebenfalls an Bildrändern, wo die Hand das Bild verlässt, weil dort die Schätzung des optischen Flusses mit dem gewählten Verfahren oft inakkurate Verschiebungsvektoren berechnet, die keine Segmentierung der Hand ermöglichen. Abbildung 4.18 zeigt eine inakkurate Flussfeldberechnung.

(a) Eingabebild **(b)** Optischer Fluss

Abbildung 4.18: Darstellung eines inakkuraten Flussfeldes, welches eine Segmentierung der Hand nicht zulässt.

Als Gegenmaßnahme wird in Fällen, in denen kein Vordergrundobjekt berechnet werden kann, das direkt davor ermittelte Vordergrundsegment mit dem aktuellen optischen Fluss verschoben, um eine Approximation des Vordergrundsegments zu erhalten. Bewegt sich die Hand nicht, wird kein optischer Fluss berechnet und das Segment bleibt dort, wo sich die Hand befindet. An Bildrändern wird dies zu einem Problem, da dort, wie oben bereits erwähnt, der geschätzte Fluss oft nicht akkurat ist. Eine Konsequenz daraus ist, dass das Vordergrundsegment an Bildrändern stehen bleibt, auch wenn die Hand sich aus dem Sichtfeld der Kamera bewegt hat und nicht mehr zu sehen ist, da das Segment durch inakkurate Verschiebungsvektoren nicht über den Bildrand aus dem Bild geschoben wird. Um solche Situationen zu erkennen, bildet MACS zu jedem Zeitpunkt von der Vordergrundregion ein Farbhistogramm, ähnlich der Erstellung eines Hautfarbenmodells wie in Abschnitt 4.2.1. Aufeinanderfolgende Farbhistogramme werden per Bhattacharyya Distanz [Bha43] verglichen. Wenn die Distanz zwischen beiden Histogrammen größer als ein Schwellenwert (hier 0,5) ist, wird davon ausgegangen, dass im Segment nicht mehr das gleiche Objekt enthalten ist und die Hand das Bild verlassen hat. Diese Methode wird Erkennung der Veränderung des Erscheinungsbildes (engl. appearance change detection) genannt und löst die oben beschriebenen drei Probleme. Abbildung 4.19 zeigt ein Beispiel für das erste Problem, bei dem die Hand stillsteht. Die aktuelle Flussfeldsegmentierung

ist in Abbildung 4.19a dargestellt und zeigt kein Vordergrundsegment. Deshalb wird das vorherige Segment mit dem optischen Fluss verschoben, siehe Abbildung 4.19b, und die Hand kann weiterhin segmentiert werden.



(a) Flussfeldsegmentierung

(b) verzerrte Flussfeldsegmentierung

(c) maskiertes Farbbild

(d) Hautfarbensegmentierung

(e) fusionierte Segmentierung

(f) Shape-Partikelfilter-Tracking

Abbildung 4.19: Handpositionsschätzung wenn die Hand sich kaum bewegt.

4.4.3 Fusion von Hautfarben- und Bewegungssegmentierung

Da zur Erkennung von Händen Hautfarbe eine wichtige Rolle spielt und die zuvor beschriebene Segmentierung von dichter Bewegungsinformation aufgrund der suboptimalen Qualität von Flussfeldern nur eine grobe Schätzung für die Handregion liefert, wird in MACS zusätzlich die Hautfarbensegmentierung von Li und Kitani [Li13a] genutzt. Für das Training des RDF-Klassifikators wurden für jede der 29 Sequenzen des IOSB-Hand-Tracking-Datensatzes fünf segmentierte Bilder der Grundwahrheit genutzt. Die Eingabebilder wurden auf eine Höhe von 300 Pixel unter Beibehaltung der Seitenverhältnisses skaliert. Über einem Raster der Länge drei wurden so an mehr als 20.000 Stellen im Bild Merkmale berechnet. Die besten Ergebnisse wurden mit einem Merkmal erzielt, das im Lab-Farbraum die Werte in einer 5×5-Nachbarschaft um ein Pixel herum konkateniert. Jedes über

das Raster betrachtete Pixel sorgt folglich für ein 75 Dimensionen umfassendes Merkmal. Für jedes Bild wird ein Entscheidungsbaum trainiert. Die meisten dieser Bäume überschreiten eine Tiefe von 25 nicht. Der Klassifikator wählt zu einem Eingabebild die fünf nächsten Entscheidungsbäume, welche den RDF für die Klassifikation bilden. Folglich stehen insgesamt 145 Entscheidungsbäume zur Auswahl. Die Ähnlichkeitsbestimmung wird über einen Histogrammvergleich realisiert. Für jedes Eingabebild werden im HSV-Farbraum drei Histogramme mit jeweils 16 Intervallen gebildet und mit den auf gleiche Weise erstellten Histogrammen der Bilder des Trainings per Bhattacharyya Distanz verglichen. Die Entscheidungsbäume der ähnlichsten Trainingsbilder berechnen jeweils ihre Klassifikationsergebnisse für ein zu klassifizierendes Pixel, welche in Form einer gewichteten Summe das Klassifikationsergebnis liefern. Nachdem dies für jedes Pixel getätigt wurde, wird die binäre Segmentierung durch die Nutzung der Methode von Otsu [Ots79a] bestimmt. Abbildung 4.17g sowie 4.19d zeigen eine Beispielsegmentierung mit obigem RDF. Beide weisen segmentierte Bereiche, die zum Hintergrund gehören, auf. Auch wenn gar keine Hand im Bild ist, findet die Binarisierung nach Otsu eine Segmentierung mit Störsegmenten. Diese Störsegmente können häufig entfernt werden, wenn nur diejenigen Segmente behalten werden, die mit dem Vordergrundsegment aus der Analyse des Flussfeldes überlappen. Die resultierenden Segmentierungen nach dieser Fusion sind in Abbildung 4.17h sowie 4.19e für zwei Beispiele dargestellt, welche keine bzw. kaum Störsegmente mehr aufweisen. Das Fusionsergebnis ist Eingabe für das Shape-Partikelfilter-Tracking aus Abschnitt 4.3.2, mit welchem die Handposition bestimmt wird.

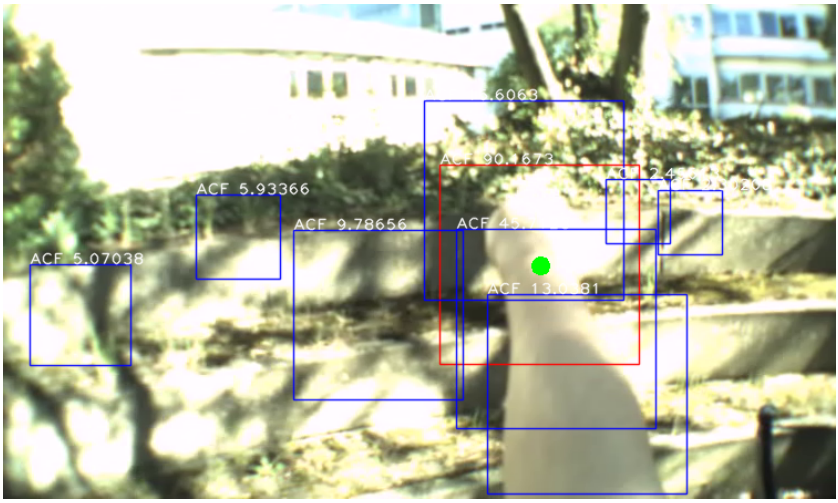
4.5 Bestimmung von Handregionshypothesen

Bei obigen Verfahren zur Handpositionsbestimmung wird angenommen, dass die Hand das größte hautfarbene Objekt ist und sich zusätzlich im Vordergrund befindet. Dies sind wichtige Hinweise, aber wünschenswert ist außerdem ein Handdetektor, der für eine bestimmte rechteckige Bildregion eine Schätzung abgibt, ob in ihr eine Hand zu sehen ist oder nicht. Nachfolgend

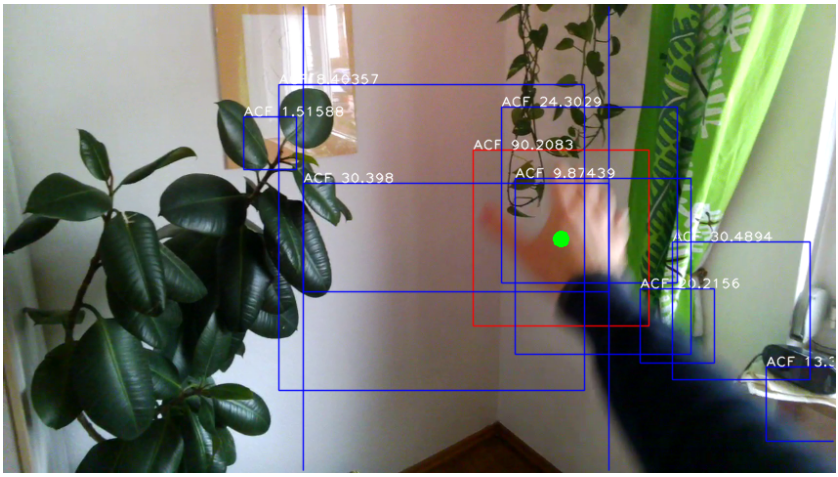
werden in den Abschnitten 4.5.1 und 4.5.2 zwei Verfahren für die Bestimmung solcher Regionen betrachtet.

4.5.1 Handregionshypothesen auf Basis aggregierter Bildkanal-Merkmale

Dollar et al. [Dol14] entwickelten das *Aggregated Channel Features* (ACF) Verfahren zur effizienten Berechnung komplexer Merkmale auf unterschiedlichen Skalierungsstufen. Die Merkmale bestehen aus den Farbwerten im LUV-Farbraum, Beträgen der Gradienten und sechs Richtungshistogrammen [Dal05]. Somit besteht jedes Merkmal aus zehn Kanälen, welche jeweils geglättet und durch eine Bündelung (engl. *Pooling*) in eine kompaktere Repräsentation umgeformt werden. Für gewöhnlich werden solche Merkmale auf jeder Ebene der Skalenpyramide berechnet und für die Detektion von Objekten genutzt. Dollar et al. umgehen diese aufwendigen Berechnungen, indem sie die Merkmale auf wenigen Ebenen exakt berechnen und die Merkmale der anderen Ebenen aus den berechneten Merkmalen durch einfache Skalierung gewinnen. Die Validität dieses Vorgehens begründet sich in einem Potenzgesetz für Statistiken in natürlichen Bildern. Für das Training in der vorliegenden Arbeit wurden quadratische Ausschnitte mit einer Länge von 50 Pixeln für eine Untermenge aus den Bildern des IOSB-Hand-Tracking-Datensatzes extrahiert und für das Training genutzt. Das ursprüngliche Verfahren von Dollar et al. [Dol14] wurde für die Erkennung von Fußgängern entwickelt. Da sich die Erscheinungsbilder von Händen und Fußgängern soweit ähneln, dass sie eine zusammenhängende größere Fläche (Oberkörper bzw. Handfläche) sowie kleinere bewegliche Bereiche aufweisen (Arme/Beine bzw. Finger), wurde für diese Arbeit die gleiche Struktur des originalen Klassifikators beibehalten, die einem Boosting-Ansatz [Fre97] mit Entscheidungsbäumen als schwachen Klassifikatoren entspricht.

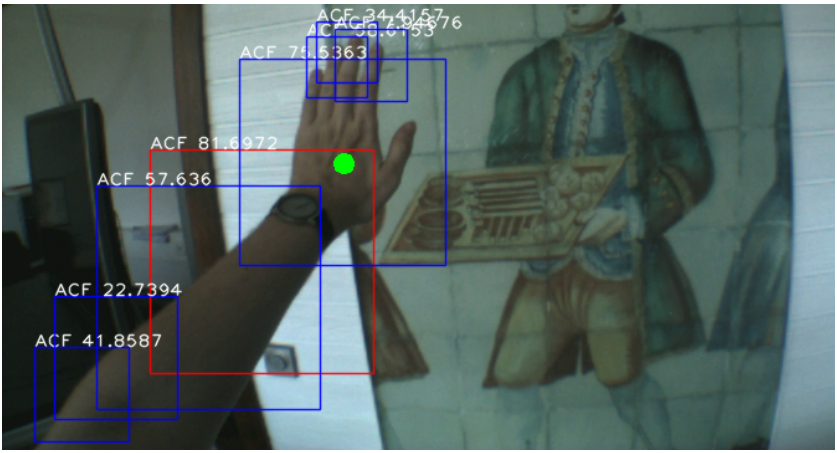


(a) Der Kandidat mit der höchsten Konfidenz entspricht in diesem Fall der besten Handregion. Andere Kandidaten mit einer Konfidenz unter 50 sind auf dem Arm, aber auch dem Hintergrund gelegen.

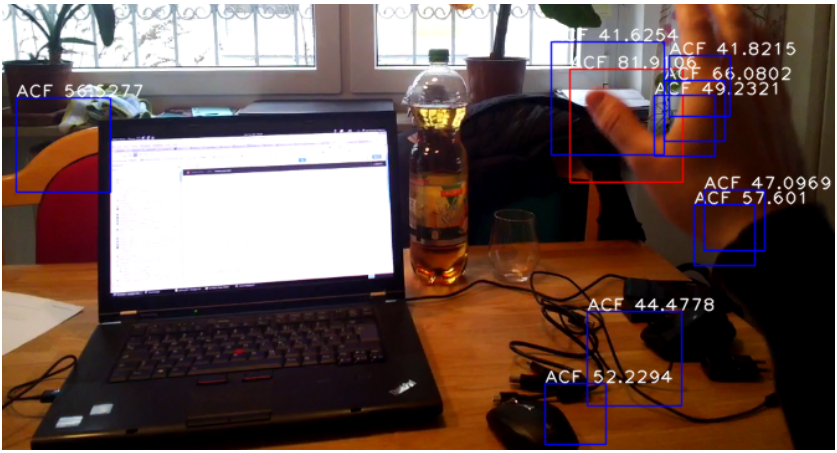


(b) Auch hier stellt die Schätzung mit der höchsten Konfidenz die beste Handregion dar.

Abbildung 4.20: Schätzungen für die Handregion mit dem ACF-Verfahren, bei denen die Schätzung mit der höchsten Konfidenz der Handregion entspricht. Dargestellt sind die zehn Schätzungen mit der höchsten Konfidenz.



(a) Der Handregionskandidat mit der höchsten Konfidenz von über 81 ist auf dem Unterarm platziert. Der zweitbeste Kandidat mit einer Konfidenz von 75 entspricht der besten Schätzung für die Handregion. Die weiteren Kandidaten sind auf dem Arm sowie den Fingern verteilt



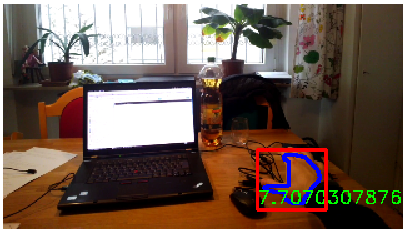
(b) Keine der zehn Handregionen mit den höchsten Konfidenzen stellt die Handregion korrekt dar. Dennoch sind einige Kandidaten mit Konfidenzen über 41 auf der Hand platziert. Der Kandidat mit der höchsten Konfidenz von 81 liegt auf dem Daumen.

Abbildung 4.21: Die hier dargestellten Schätzungen für die Handregion mit dem ACF-Verfahren zeigen, dass die Schätzung mit der höchsten Konfidenz nicht die beste Handregion sein muss.

Beispiele für Handregionsschätzungen sind in den Abbildungen 4.20 und 4.21 als blaue und rote Quadrate dargestellt. Das Verfahren schätzt Regionen, die der Hand gut entsprechen, aber auch Regionen auf dem Arm oder im Hintergrund oder für eine Hand zu kleine Regionen. In den Abbildungen sind zusätzlich die Konfidenzen der einzelnen Regionen dargestellt. Die Region mit der besten Konfidenz ist rot, die anderen sind blau gezeichnet. Es wird ersichtlich, dass die Schätzung mit der höchsten Konfidenz nicht direkt die beste Wahl ist. In Kombination mit einem Verfahren, das bereits eine Schätzung für die Handposition durchführt, ergeben sich interessante Möglichkeiten zur Bestimmung derjenigen Region, die am besten die Hand darstellt. Ein in dieser Arbeit entwickeltes Verfahren zur Handpositionsschätzung mit MACS für eine initiale Handregion und einer Verfeinerung durch aggregierte Bildkanal-Merkmale wird in Abschnitt 4.6 vorgestellt.

4.5.2 Handregionshypothesen auf Basis des CNN HandSegNet

Aufgrund der sehr guten Handsegmentierungsergebnisse des CNN HandSegNet, siehe Abschnitt 4.2.2, wird dieses Verfahren in der vorliegenden Arbeit ebenfalls für die Bestimmung von rechteckigen Handregionen genutzt. Dazu wird in der binären Segmentierung von HandSegNet nach zusammenhängenden Regionen gesucht und deren Konturen bestimmt. Abbildung 4.22 zeigt eine solche Segmentierung und Abbildung 4.22a die sich ergebende, in blau gezeichnete Kontur der einzigen gefundenen Region. Für diese wird ein einhüllendes Rechteck bestimmt und in alle Richtungen vergrößert (ca. 10 %), damit das Rechteck häufiger die gesamte Hand inklusive der Finger beinhaltet, falls diese durch die Segmentierungen nicht vom Hintergrund getrennt werden. Aus der zusammenhängenden Region innerhalb dieses Rechtecks kann aus dem von HandSegNet erstellten Konfidenzbild zusätzlich die durchschnittliche Konfidenz gebildet werden. Dieses ist in Abbildung 4.22a als grüne Zahl dargestellt. Ein in dieser Arbeit entwickeltes Verfahren zur Handpositionsbestimmung, das diese Art der Handregionsbestimmung nutzt, wird in Abschnitt 4.7 vorgestellt.



(a) Darstellung der Kontur und resultierenden rechteckigen Handregion mit Konfidenz

(b) Handsegmentierung von HandSegNet

Abbildung 4.22: Darstellung der Bestimmung einer rechteckigen Handregion aus der Handsegmentierung von HandSegNet

4.6 Handlokalisierung mit AfM

In diesem Abschnitt wird das in Hammer et al. [Ham16a] vorgestellte Verfahren Aggregated Channel Features featuring MACS (AfM) präsentiert, welches die Handpositionsbestimmung von MACS, siehe Abschnitt 4.4, als initiale Schätzung nutzt, um darauf basierend von den durch das ACF-Verfahren bestimmten Regionskandidaten denjenigen auszuwählen, der am besten die Hand widerspiegelt.

Nachfolgend sei $h_{\text{MACS}} \in \mathbb{R}^2$ die Handpositionsbestimmung von MACS. Findet MACS keine Hand im Bild, werden die Ergebnisse des ACF-Verfahrens nicht betrachtet. Das ACF-Verfahren bestimmt Regionskandidaten i zentriert an Pixel p_i mit Konfidenz s_i. Nachfolgend werden verschiedene Strategien zur Fusion von MACS und den Regionskandidaten vom ACF-Verfahren vorgestellt.

4.6.1 Simple Verfeinerung

Die simple Verfeinerung bestimmt den besten Regionskandidaten i_{best} als

$$i_{\text{best}} = \arg\max_i (d_i + v_i) \tag{4.1}$$

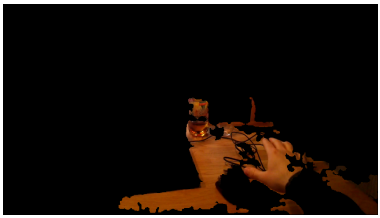
mit $i \in \{1, \ldots, n\}$ bei Betrachtung der n Regionskandidaten mit höchster Konfidenz s_i. Die Distanz-Gewichtung d_i bestraft große Abstände zu h_{MACS} durch

$$d_i := 1 - \frac{||p_i - h_{\text{MACS}}||_2}{\sum_j ||p_j - h_{\text{MACS}}||_2} \tag{4.2}$$

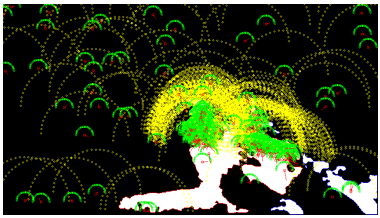
und die Konfidenz-Bewertung v_i mit

$$v_i := \frac{s_i}{\sum_j s_j}. \tag{4.3}$$

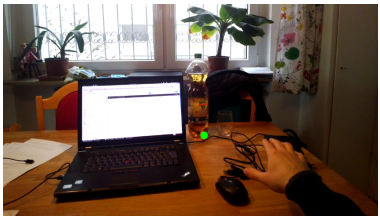
setzt die Konfidenz s_i ins Verhältnis zur Summe aller erzielten Konfidenzen. Die resultierende Handposition h_t zu einem Zeitpunkt t ist folglich $p_{i_{\text{best}}}$. In der vorliegenden Arbeit werden maximal $n = 10$ Regionskandidaten betrachtet. Abbildung 4.23 zeigt ein Beispiel für die unterschiedlichen Ergebnisse bei schlechter Hautfarbensegmentierung durch MACS und AfM.



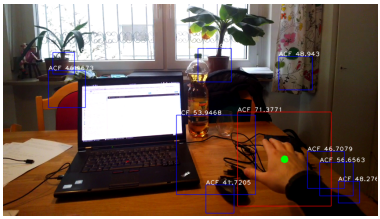
(a) Fusionierte Segmentierung von MACS



(b) Verteilung der Partikel



(c) Schätzung der Handposition von MACS (grüner Punkt)



(d) Schätzung der Handposition von AfM (grüner Punkt)

Abbildung 4.23: Beispiel für die Handpositionsbestimmung durch MACS und AfM.

4.6.2 Verfeinerung durch Propagation

Da die MACS-Schätzungen für die Handposition nicht präzise sind, bekommen häufiger nicht optimale Regionskandidaten durch die Distanzgewichtung bessere Bewertungen als der beste Regionskandidat. Um hier entgegenzuwirken, wird eine Schätzung h_g für die aktuelle Handposition aus den letzten zwei Handpositionen h_{t-1} und h_{t-2} mit

$$h_g := \begin{cases} h_{t-1} + (h_{t-1} - h_{t-2}) & \text{falls } h_{t-1} \text{ und } h_{t-2} \text{ berechnet} \\ h_{\text{MACS}} & \text{sonst} \end{cases} \tag{4.4}$$

propagiert. Die Distanz-Gewichtung $d_i{}'$ wird durch

$$d_i{}' := 1 - \frac{||p_i - h_g||_2}{\sum_j ||p_j - h_g||_2} \tag{4.5}$$

ersetzt und der Rest der Berechnung gleicht Abschnitt 4.6.1.

4.6.3 Robuste Distanz-Gewichtung

Ein Problem der oben durchgeführten Distanzbewertung ist die Normalisierung zur Summe aller auftretenden Distanzen der betrachteten Regionskandidaten. Die Distanz-Gewichtung wird folglich sowohl durch die Anzahl n als auch durch schlechte Schätzungen mit großer Distanz zu h_g beeinflusst. Deshalb wird die Distanzberechnung durch eine robuste Distanz-Gewichtung ersetzt, bei der jede einzelne Abweichung in Relation zu einer maximalen Distanz $d_{\max}$ gesetzt wird. Die robuste Distanz-Gewichtung $d_i{}''$ berechnet sich als

$$d_i{}'' := \begin{cases} 0 & \text{falls } ||p_i - h_g||_2 \geq d_{\max} \\ 1 - \frac{||p_i - h_g||_2}{d_{\max}} & \text{sonst.} \end{cases} \tag{4.6}$$

Die Distanz-Gewichtung ist dadurch unabhängig von der Anzahl der betrachteten Kandidaten als auch von der lokalen Verteilung derselben. Falls die Distanz einer Position p_i zu h_g größer als $d_{\max}$ ist, wird der Kandidat mit einer Distanz-Gewichtung von null bestraft. Bei Abweichungen von null bis $d_{\max}$ steigt die Gewichtung linear von null zu eins an. In der vorliegenden Arbeit wurden gute Erfahrungen mit 200 Pixeln für $d_{\max}$ gewonnen, was ca. der zweifachen maximal auftretenden Distanz zwischen zwei Handpositionen im IOSB-Hand-Tracking-Datensatz entspricht.

4.6.4 Medianbasierte Rückweisung von Kandidaten

Eine weitere Bestrafung von Kandidaten mit zu großen Abweichungen von der aktuell vorliegenden Handbewegung kann durch die Bestimmung des Medians h_{median} der letzten m Handpositionen durchgeführt werden. Zusätzlich wird hierbei die aktuelle Geschwindigkeit d_{last} berücksichtigt, um die Distanz-Gewichtung an die Bewegungsgeschwindigkeit anzupassen. d_{last} berechnet sich dabei als

$$d_{\text{last}} := \begin{cases} ||h_{t-1} - h_{t-2}||_2 & \text{falls } h_{t-1} \text{ und } h_{t-2} \text{ vorhanden} \\ 0 & \text{sonst.} \end{cases} \tag{4.7}$$

Kandidaten mit Abweichungen $||p_i - h_{\text{median}}||_2 > d_{\max} + d_{\text{last}}$ werden direkt verworfen. Bei $m = 1$ gilt $h_{\text{median}} = h_{t-1}$, bei $m = 3$ werden h_{t-1}, h_{t-2} und h_{t-3} betrachtet. Beide Varianten führen dazu, dass h_{median} etwas hinter der propagierten Position h_g hinterherläuft, führt aber zu höherer Robustheit, wenn die Kandidatenpositionen p_i und damit auch die älteren Schätzungen nicht perfekt sind. Besonders in diesem Fall ist die Einbindung von d_{last} essentiell, weil h_{median} sonst zu weit zurückliegen kann. Bei $m = -1$ wird $h_{\text{median}} = h_g$ gesetzt, was nur sinnvoll ist, wenn ein Verfahren genaue Schätzungen der Handposition durchführt. Beim hier vorgestellten AfM-Verfahren wird aufgrund der nicht perfekten Kandidaten $m = 3$ gewählt.

4.6.5 Anpassung der Segmentierung

Da AfM das Verfahren MACS nutzt und MACS für die Handpositionsbestimmung das Shape-Partikelfilter-Tracking, welches auf der Segmentierung der Hautfarbe aufsetzt, kann diese Segmentierung ebenfalls optimiert werden. Hierzu werden alle Pixel in der Segmentierung, die einen größeren Abstand zu h_{median} aufweisen als $1{,}5 \cdot d_{\max}$ dem Hintergrund zugewiesen. Das 1,5-fache von $d_{\max}$ entspricht ungefähr dreimal der maximal auftretenden Distanz zwischen zwei zeitlich aufeinanderfolgenden Handpositionen. So können Bereiche, die außerhalb dieses Bewegungsradius um h_{median} liegen, auch bei $m = 3$ nicht für die Handposition in Frage kommen. Durch diese Anpassung der Segmentierung wird bereits bei der Schätzung von h_{MACS} eine Verbesserung bei schlechten Segmentierungen erreicht.

4.6.6 Konfidenzbasierte Rückweisung von Kandidaten

Eine weitere einfach zu integrierende Rückweisung von Kandidaten kann über eine mindestens zu erreichende ACF-Konfidenz s_i realisiert werden. Im vorliegenden Fall hat sich hier der Schwellenwert $t_{\text{ACF}} = 20$ bewährt. Alle Kandidaten mit einer niedrigeren Konfidenz werden direkt verworfen. Ein weiterer positiver Nebeneffekt der Einführung von t_{ACF} ist, dass die Entscheidung, ob eine Region überhaupt eine Hand enthält, nicht im Partikel-Filter-Tracking durch den Vergleich von $\bar{s}_{\text{skin}}$ mit t_{skin}, siehe Abschnitt 4.3.2, getroffen werden muss. Dadurch kann t_{skin} auf 100 reduziert werden, wodurch auch bei schlechteren Hautfarbensegmentierungen potentiell mehr Hände erkannt werden können, weil sie nicht vorher bereits als „Hand nicht sichtbar“ klassifiziert werden. Durch die Betrachtung der ACF-Konfidenzen kann später im Prozess entschieden werden, ob wirklich eine Hand im Bild enthalten ist.

Abbildung 4.24 zeigt die unterschiedlichen Berechnungsschritte. Die von MACS geschätzte Position h_{MACS} ist als dunkelblauer Punkt zwischen Zeige- und Mittelfinger zu sehen. Die propagierte Handposition h_g aus Abschnitt 4.6.2 ist als pinkfarbener Punkt rechts auf der Handfläche eingezeichnet. Für

die medianbasierte Rückweisung wird die Position h_{median} genutzt, die als hellblauer Punkt unter dem Mittelfinger auf der Handfläche zu sehen ist. Die hellblauen Kreise haben einen Durchmesser von $d_{\max}$ sowie $d_{\max} + d_{\text{last}}$. Alle Kandidaten außerhalb des größeren Kreises werden durch die medianbasierte Rückweisung verworfen. Obwohl das ACF-Verfahren den Daumen mit einer Konfidenz von knapp 80 als beste Region bestimmt, wird durch AfM dasjenige Rechteck ausgewählt, welches am besten zur zentralen Handposition passt und den grünen Punkt als Mittelpunkt aufweist.

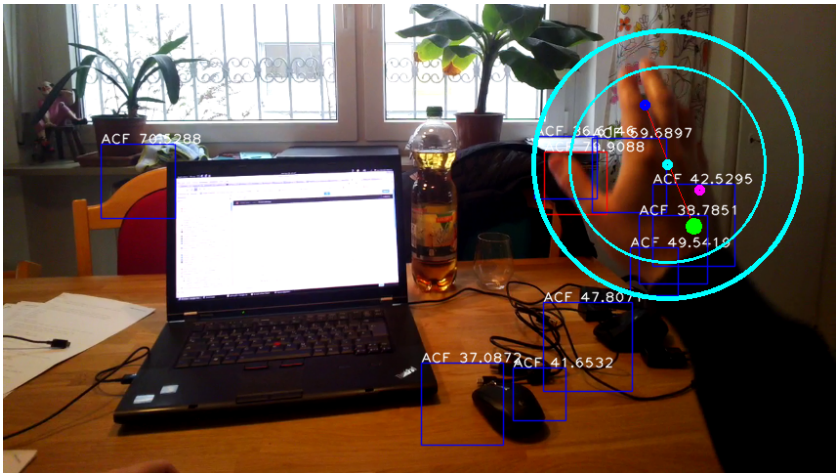


Abbildung 4.24: Beispiel für die Bestimmung mit AfM

4.7 Handlokalisierung mit HandSegNet

Das CNN HandSegNet kann sowohl für die Handsegmentierung, siehe Abschnitt 4.2.2, als auch für die Bestimmung von Handregionshypothesen, siehe Abschnitt 4.5.2, genutzt werden. Bei den bisher vorgestellten Verfahren Partikelfilter-Tracking, Abschnitt 4.3.2, MACS, Abschnitt 4.4, sowie AfM, Abschnitt 4.6, kann die bisher genutzte Handsegmentierung durch diejenige von HandSegNet ersetzt werden. Eine weitere Möglichkeit ist die Nutzung der Handregionshypothesen von HandSegNet. Nachfolgend werden verschiedene Strategien hierfür beschrieben.

4.7.1 Direkte Nutzung der Handregionshypothesen

Ein in der vorliegenden Arbeit zusätzlich betrachtetes Verfahren nutzt die Handregionshypothesen, bestimmt wie in Abschnitt 4.5.2, auf direkte Weise ohne die Zwischenschaltung eines Partikelfilters. Es zeigt sich, dass die Konfidenz der Hypothesen sehr aussagekräftig ist und Handregionshypothesen mit einer Konfidenz von über eins mit hoher Sicherheit eine Hand beinhalten. Die einfachste Variante des HSN-Trackings bestimmt die Handregion mit der höchsten Konfidenz und wählt ihr Zentrum als Handposition. Dieses Verfahren wird nachfolgend *HSN-Rect-Direct* genannt. Ein Beispiel für eine Handregionshypothese ist in Abbildung 4.25a mit rechteckiger Handregion (rot) sowie segmentierten Handpixeln innerhalb der blauen Kontur dargestellt. Die durch HSN-Rect-Direct bestimmte Handposition ist in Abbildung 4.25b eingezeichnet.

4.7.2 Trackingbasierte Nutzung der Handregionshypothesen

Da HSN-Rect-Direct keine Informationen über zuvor getätigte Schätzungen verwendet, besteht die Gefahr, dass eine einzige Schätzung mit einer höheren

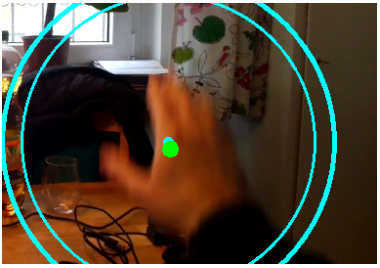
Konfidenz als der richtigen Handregion eine falsche Schätzung der Handposition erzeugt. Aus diesem Grund wird der Tracking-Mechanismus des AfM-Verfahrens genutzt. Dieses Verfahren wird nachfolgend *HSN-Rect-Tracking* genannt, das ebenfalls das Zentrum der besten Handregionshypothese als Handposition berechnet. Abbildung 4.25c zeigt im Vergleich zu Abbildung 4.25b zusätzlich die blauen Kreise der medianbasierten Rückweisung, hier allerdings zentriert an der propagierten Handposition. Die resultierende Handposition gleicht der von HSN-Rect-Direct.

4.7.3 Nutzung der Handsegmentierung innerhalb einer Region

HSN-Rect-Direct und HSN-Rect-Tracking verwenden als letztendliche Handposition das Zentrum der Handregion. Aufgrund der vorhandenen Segmentierung der Hand kann der Schwerpunkt der segmentierten Handpixel berechnet werden und als Handposition genutzt werden. Dies hat den Vorteil, dass die Finger die Handposition weniger vom Zentrum der Handfläche wegziehen. Dieses Verfahren wird nachfolgend *HSN-Seg-Tracking* genannt und Abbildung 4.25d zeigt die resultierende Handpositionsschätzung (grüner Punkt), welche näher am Zentrum der Handfläche platziert ist als die von HSN-Rect-Direct und HSN-Rect-Tracking, welche in der Abbildung zum Vergleich als dunkelblauer Punkt eingezeichnet ist.

(a) Handregionshypothese

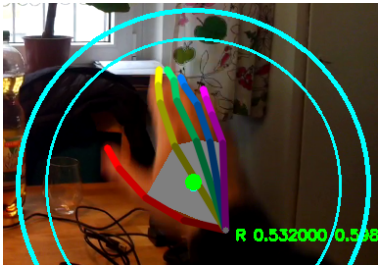
(b) HSN-Rect-Direct



(c) HSN-Rect-Tracking

(d) HSN-Seg-Tracking

(e) 2D-Handpose von OpenPose



(f) HSN-Pose-Tracking

Abbildung 4.25: Beispiele für die Bestimmung der Handposition mit verschiedenen Verfahren auf Basis von Handregionshypothesen von HandSegNet. Der grüne Punkt stellt die vom jeweiligen Verfahren geschätzte Handposition dar. In Abbildung (d) ist zum Vergleich als dunkelblauer Punkt das Ergebnis von HSN-Rect-Tracking eingezeichnet. Abbildung (e) zeigt die ermittelte Handpose von OpenPose. HSN-Pose-Tracking (siehe Abschnitt 4.8.1) wird in Abbildung (f) veranschaulicht. In Form des grauen Polygons ist die durch die Handpose angenäherte Handfläche und als grüner Punkt die daraus ermittelte Handposition dargestellt.

4.8 Simultane Handregionsbestimmung und Posenschätzung

In Abschnitt 2.3.2 wurde bereits die Software *OpenPose* [Cao18] vorgestellt, welche das Verfahren zur 2D-Handposenschätzung von Simon et al. [Sim17] beinhaltet. Abbildung 4.26 zeigt die Nummerierung der Gelenkpositionen der Hand. Für jedes Gelenk schätzt OpenPose eine Konfidenz. Es lässt sich also für die ganze Hand eine durchschnittliche Konfidenz berechnen, welche zur Bewertung der Handregionshypothese genutzt werden kann.

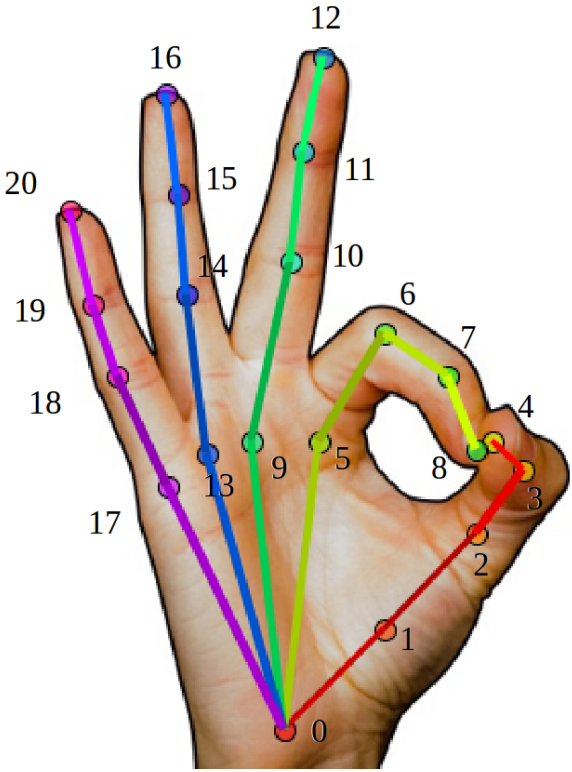


Abbildung 4.26: Nummerierung der Gelenke der Handpose in OpenPose [Sim17, Cao18]

4.8.1 Schätzung des Zentrums der Handfläche durch die Handpose

Die Handpose kann zur besseren Schätzung des Zentrums der Handfläche herangezogen werden. Für eine Handregionshypothese wird von OpenPose eine Handpose sowohl für die linke wie auch für die rechte Hand geschätzt. OpenPose benötigt eine solche Handregion und die Aussage, ob eine linke oder rechte Hand enthalten ist. Anschließend wird die Posen-Konfidenz durch Mittelung der Gelenk-Konfidenzen gebildet. Die Pose mit der besten Posenkonfidenz wird als resultierende Hand für die betrachtete Handregion genutzt. Abbildung 4.25e zeigt die Handpose zur Handregionshypothese aus Abbildung 4.25a. Die Gelenke mit den Positionen 0, 1, 2, 5, 9, 13 und 17 bilden ein Polygon, das der Handfläche am nächsten kommt. Der Schwerpunkt dieses Polygons, siehe graue Fläche auf der Hand in Abbildung 4.25f, wird vom nachfolgend *HSN-Pose-Tracking (HPT)* genannten Verfahren, das ansonsten HSN-Rect-Tracking ähnelt, als Handposition p_i angenommen. Verglichen mit HSN-Rect-Direct, HSN-Rect-Tracking und HSN-Seg-Tracking kann eine deutlich genauere Lokalisierung des Zentrums der Handfläche im dargestellten Beispiel festgestellt werden.

4.8.2 Handposenschätzung mit HandSegNet und OpenPose

Dieser Abschnitt stellt Verfahren zur 2D-Handposenschätzung vor, die auf HandSegNet sowie OpenPose basieren und beide Ansätze durch eine geschickte Wahl von Handregionen fusionieren. Das Basisverfahren ähnelt HSN-Pose-Tracking aus dem vorangegangenen Abschnitt, welches für die Handpositionsbestimmung genutzt wurde. Nachfolgend steht im Mittelpunkt die Verbesserung der Handposenschätzung. HSN-Pose-Tracking funktioniert wie folgt: Zuerst werden mit HandSegNet Handregionshypothesen inklusive ihrer Konfidenzen gebildet. Für jede Hypothese wird eine linke und rechte Handpose mit OpenPose geschätzt. Die Pose mit der besten Posen-Konfidenz wird als Pose für die Handregion bestimmt. Die Auswahl der besten Handregion aus allen betrachteten Hypothesen wird, wie bei AfM in Abschnitt

4.6 beschrieben, durchgeführt. Die Auswahl der besten Handregion i_{best} wird durch die Berechnung von Gleichung 4.1 ermittelt, welche hier für den besseren Lesefluss nochmal angegeben wird:

$$i_{\text{best}} = \arg\max_i (d_i + v_i) \tag{4.8}$$

Die Distanz-Gewichtung d_i wird über die robuste Distanz-Gewichtung aus Abschnitt 4.6.3 ermittelt,

$$d_i := \begin{cases} 0 & \text{falls } ||p_i - h_g||_2 \geq d_{\max} \\ 1 - \frac{||p_i - h_g||_2}{d_{\max}} & \text{sonst,} \end{cases} \tag{4.9}$$

wobei p_i die Zentren der rechteckigen Handregionen i sind und h_g die propagierte Handposition, siehe Abschnitt 4.6.2. Der Wert $d_{\max}$ wird auf 200 gesetzt.

Die Konfidenz-Bewertung v_i mit den Konfidenzen s_i der Regionshypothesen berechnet sich als

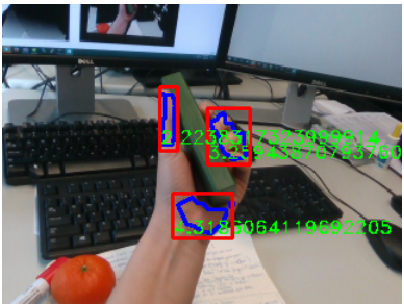
$$v_i := \frac{s_i}{\sum_j s_j}. \tag{4.10}$$

Für die medianbasierte Rückweisung wird wie bei AfM $m = 3$ gewählt. Bei HSN-Pose-Tracking ist s_i die Konfidenz der Handregionshypothesen. Da durch die Pose die Posen-Konfidenz vorhanden ist, kann diese stattdessen für s_i genutzt werden. Hier zeigt sich, dass Posen mit Konfidenzen über einem Basis-Schwellenwert $t_{\text{basic}} = 0{,}2$ recht sicher eine Hand und ihre Pose gut beschreiben. Bei Konfidenzen über einem Schwellenwert $t_{\text{assured}} = 0{,}5$ kann sehr sicher von einer guten Posenschätzung ausgegangen werden. Wird für eine Handregionshypothese durch OpenPose sowohl eine linke Handpose als auch eine rechte Handpose ermittelt, liegt meistens nur eine Posen-Konfidenz über t_{basic}, weshalb gut zwischen linken und rechten Händen unterschieden werden kann. Da in den verwendeten Datensätzen nur eine Hand vorkommt und die Seite sich somit nicht ändert, kann die Anzahl der Posenschätzungen

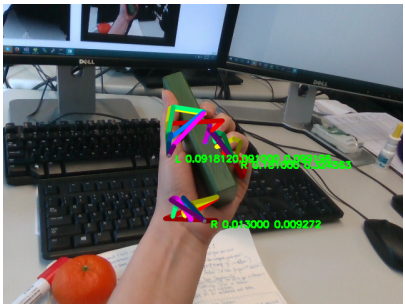
verkleinert werden, wenn nur nach einer Seite gesucht wird. Zu Anfang weiß das Verfahren nicht, welche Seite es suchen muss, und sucht daher nach beiden Seiten. Dann wird in einem Zeitfenster von n vorangegangenen Einzelbildern (im vorliegenden Fall von $n = 11$ Einzelbildern, also ca. einer drittel Sekunde) berechnet, wie viele linke Hände n_{left} und wie viele rechte Hände n_{right} mit Posen-Konfidenzen über t_{assured} gefunden wurden. Bei $n_{\text{side}} > \frac{n}{2} + 1$ mit side $\in$ {left,right} wird nur die jeweilige durch side bestimmte Hand gesucht. Ansonsten ist sich das Verfahren nicht sicher, welche Seite gerade vorzufinden ist und sucht nach beiden, bis sich das Verfahren wieder sicher ist. Dadurch wird die Anzahl der Posenschätzungen durch OpenPose fast halbiert.

4.8.2.1 Handregionshypothesen bei teilweise verdeckter Hand

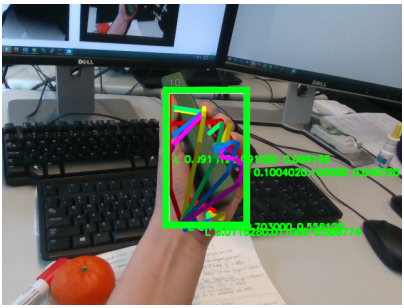
Auf dem IOSB-Hand-Tracking-Datensatz halten die Hände keine Objekte. Im EgoDexter-Datensatz werden unterschiedliche Objekte mit der Hand manipuliert. Abbildung 4.27 zeigt ein Problem von HSN-Pose-Tracking, wenn die Handsegmentierung von HandSegNet in einzelne Unterbereiche aufgeteilt wird, siehe Abbildung 4.27a. Wie im Beispiel zu sehen, entstehen anstelle einer Handregion drei Handregionshypothesen, für welche jeweils einzeln eine Pose bestimmt wird, die der wahren Handpose nicht nahekommen, siehe Abbildung 4.27b. Entsprechend kann keine richtige Pose gefunden werden. Um diesem Problem entgegenzuwirken, wird angenommen, dass die vorangegangene Posenschätzung korrekt war, und aus allen Gelenkpositionen ein einhüllendes Rechteck bestimmt, welches in Abbildung 4.27c als grünes Rechteck zu sehen ist. Dieses wird als weitere Handregionshypothese rückgeführt und verwendet. Da die Handregionen für OpenPose sowohl in der Höhe als auch Breite fast dreimal so groß gewählt werden müssen, bleiben Bewegungen berücksichtigt. Unter Nutzung dieser rückgeführten Handpose kann trotzdem eine korrekte Pose bestimmt, siehe Abbildung 4.27c, und ausgewählt werden, siehe Abbildung 4.27d. HSN-Pose-Tracking mit Rückführung der vorherigen Handpose wird nachfolgend HPT with Feedback (HPTwF) bezeichnet.



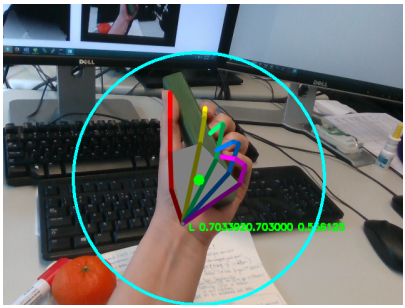
(a) Gefundene Handregionen bei Verdeckung



(b) Ermittelte Posen von HSN-Pose-Tracking



(c) Berücksichtigung der letzten gefundenen Handpose durch ihr einhüllendes Rechteck



(d) Ermittelte Pose von HPT erweitert um die Rückführung der vorherigen Pose (HPTwF)

Abbildung 4.27: Problem bei der Handsegmentierung und Handregionsbestimmung

4.8.2.2 Berücksichtigung der Ähnlichkeit von Handposen

Da aufeinanderfolgende Handposen eine gewisse Ähnlichkeit zueinander haben, kann Gleichung 4.8 um einen Ähnlichkeitsterm erweitert werden, der die vorangegangene Handpose mit einer Kandidatenpose vergleicht. Dabei seien die Gelenkpositionen g_j mit $j \in \{0, \ldots, 20\}$ die der vorangegangenen Handpose mit Handposition p und $g_{i,j}$ die Gelenkpositionen des Handposenkandidaten i mit Handposition p_i. Beide Handposen werden durch die Ähnlichkeit sim_i

verglichen, welche sich als

$$\text{sim}_i := \frac{\sum_{j=0}^{20} ||(g_j - p) - (g_{i,j} - p_i)||_2}{21} \tag{4.11}$$

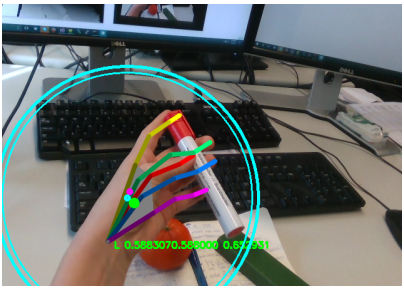
berechnet. Wie zu erkennen, werden unterschiedliche Verschiebungen herausgerechnet. Diese Ähnlichkeit wird bzgl. einer maximalen Ähnlichkeit sim_{max} normalisiert, wodurch sich der Ähnlichkeitsterm a_i ergibt als

$$a_i := \begin{cases} 0 & \text{falls } \text{sim}_i \geq \text{sim}_{\max} \\ 1 - \frac{\text{sim}_i}{\text{sim}_{\max}} & \text{sonst.} \end{cases} \tag{4.12}$$

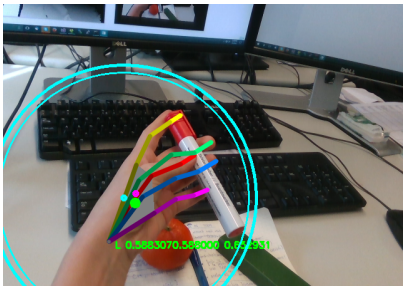
Die beste Handpose berechnet sich durch:

$$i_{\text{best}} = \arg\max_i (d_i + v_i + a_i). \tag{4.13}$$

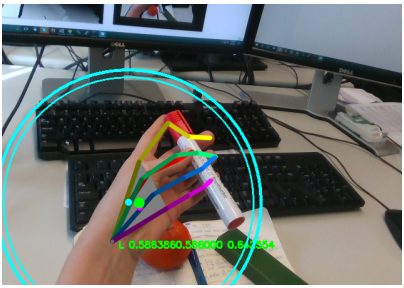
Dieser Ansatz wird nachfolgend *HPTwF-Sim* (HPTwF with Similarity) bezeichnet. Abbildung 4.28 stellt ein Beispiel dar, bei der der Ähnlichkeitsterm verhindert, dass kurzzeitig anstatt einer linken eine rechte Hand erkannt wird. Eine weitere Verbesserung, die der Ähnlichkeitsterm erzielt, ist in Abbildung 4.29 zu sehen, wo verhindert wird, dass der linke Ringfinger eine fälschliche Veränderung erfährt.



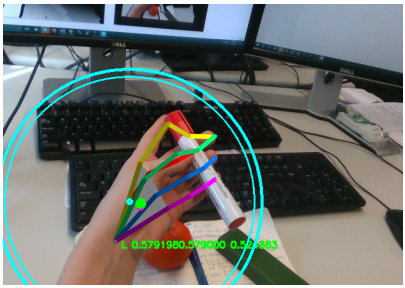
(a) HPTwF Einzelbild 64



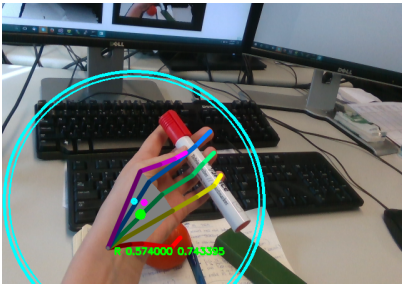
(b) HPTwF-Sim Einzelbild 64



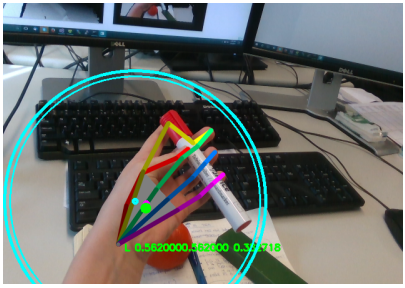
(c) HPTwF Einzelbild 65



(d) HPTwF-Sim Einzelbild 65

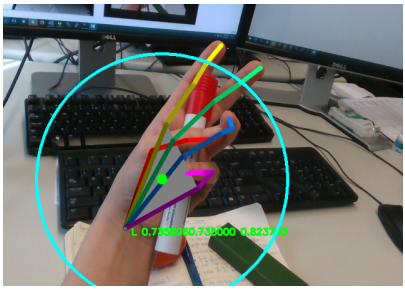


(e) HPTwF Einzelbild 66

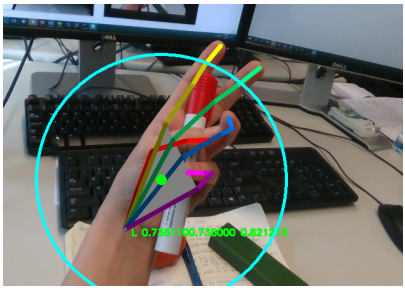


(f) HPTwF-Sim Einzelbild 66

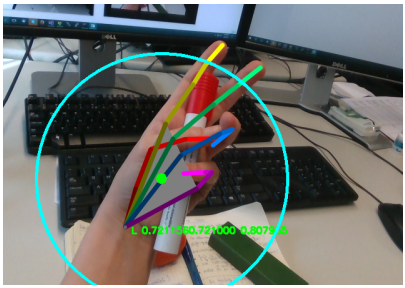
Abbildung 4.28: Darstellung des Unterschieds zwischen den Posenschätzungen von HPTwF und HPTwF-Sim auf Bildern der Sequenz *Desk* des EgoDexter-Datensatzes. Beim HPTwF-Verfahren (linke Spalte) wird in Abbildung (e) fälschlicherweise eine rechte Handpose erkannt, obwohl zuvor, siehe Abbildung (c), korrekt eine linke Hand berechnet worden ist. Beim HPTwF-Sim-Verfahren (rechte Spalte) geschieht dies nicht (siehe auch die L- und R-Bezeichnungen in grüner Schrift unter den eingezeichneten Posen).



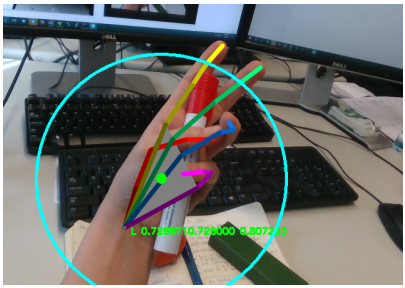
(a) HPTwF Einzelbild 148



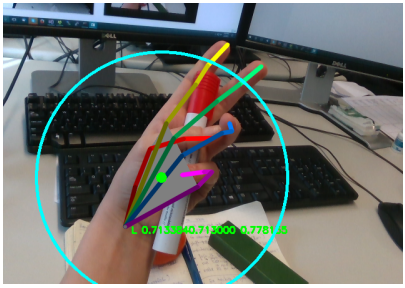
(b) HPTwF-Sim Einzelbild 148



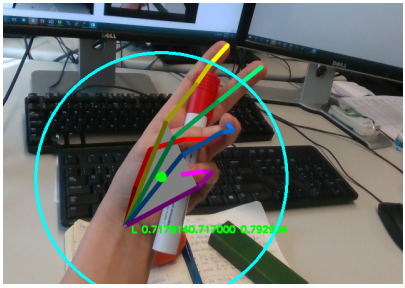
(c) HPTwF Einzelbild 149



(d) HPTwF-Sim Einzelbild 149



(e) HPTwF Einzelbild 150



(f) HPTwF-Sim Einzelbild 150

Abbildung 4.29: Darstellung des Unterschieds zwischen den Posenschätzungen von HPTwF und HPTwF-Sim auf Bildern der Sequenz *Desk* des EgoDexter-Datensatzes. Beim HPTwF-Verfahren links wird der Ringfinger (dunkelblau) in Einzelbild 149, Abbildung (c), verformt. Abbildung (d) zeigt, dass dies beim HPTwF-Sim-Verfahren nicht geschieht.

4.8.2.3 Berücksichtigung der Segmentierung

Da durch HandSegNet eine Segmentierung der Hand vorliegt und unter der Annahme, dass diese Segmentierung qualitativ hochwertig ist, sollten die Gelenkpositionen einer Handpose auf der Hand, also auf segmentierten Pixeln liegen. Abbildung 4.30 zeigt eine Handsegmentierung sowie zwei verschiedene Posenschätzungen. Die untere Pose, geschätzt von HPTwF-Supp, sollte bevorzugt werden, da bei ihr der Daumen ebenfalls auf segmentiertem Bereich liegt. Dies kann durch einen Unterstützungsterm b_i realisiert werden, der die Unterstützung durch die Segmentierung beschreibt als

$$b_i := \frac{\sum_{j=0}^{20} \text{support}(g_{i,j})}{21} \tag{4.14}$$

mit

$$\text{support}(p) = \begin{cases} 1, & \text{falls Pixel } p \text{ als Handpixel segmentiert} \\ 0, & \text{sonst.} \end{cases} \tag{4.15}$$

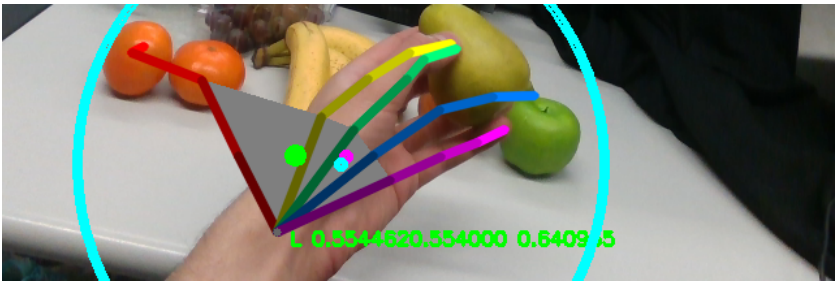
Die beste Handpose berechnet sich durch

$$i_{\text{best}} = \arg\max_i (d_i + v_i + b_i). \tag{4.16}$$

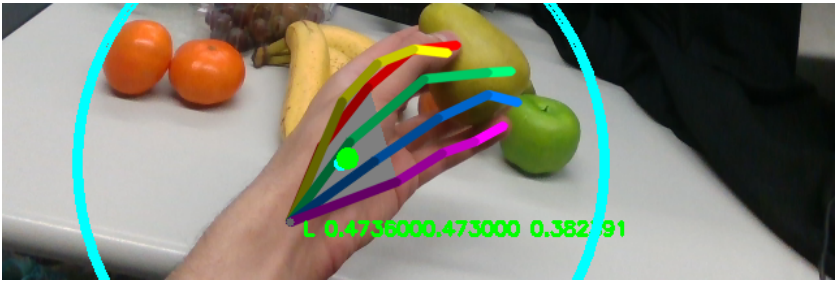
Dieser Ansatz wird nachfolgend *HPTwF-Supp* (HPTwF with Support) bezeichnet. Bei gleichzeitiger Berücksichtigung des Ähnlichkeitsterms a_i und Unterstützungsterms b_i berechnet sich die beste Handpose im nachfolgend *HPTwF-All* bezeichneten Verfahren als

$$i_{\text{best}} = \arg\max_i (d_i + v_i + a_i + b_i). \tag{4.17}$$

(a) Handsegmentierung



(b) HPTwF



(c) HPTwF-Supp

Abbildung 4.30: Darstellung des Unterschieds zwischen den Posenschätzungen von HPTwF und HPTwF-Supp auf Bildern der Sequenz *Fruits* des EgoDexter-Datensatzes. Abbildung (a) zeigt die Handsegmentierung von HandSegNet. Abbildung (b) zeigt die Schätzung des HPTwF-Verfahrens. Der Daumen (rot) ist komplett falsch geschätzt. Durch den Unterstützungsterm b_i lässt sich eine solche Schätzung verhindern, siehe Abbildung (c).

4.9 Evaluation

In diesem Abschnitt werden die zuvor vorgestellten Verfahren evaluiert. Abschnitt 4.9.1 vergleicht die Verfahren zur Handpositionsbestimmung anhand ihrer historischen Entwicklung. Die Verfahren zur Handposenschätzung werden in Abschnitt 4.9.2 evaluiert und mit dem Stand der Technik verglichen.

4.9.1 Beurteilung der Verfahren zur Handpositionsbestimmung

Die Verfahren zur Handpositionsbestimmung werden anhand ihrer historischen Entwicklung auf dem IOSB-Hand-Tracking-Datensatz, siehe Abschnitt 4.1.1.1, verglichen. Deshalb werden zuerst die älteren Verfahren in Abschnitt 4.9.1.1 gegenübergestellt. Mit der Veröffentlichung des von Zimmermann und Brox vorgestellten CNN HandSegNet [Zim17] gibt es ein weiteres Verfahren zur Handsegmentierung, das auf den IOSB-Hand-Tracking-Datensatz angepasst wurde, siehe Abschnitt 4.2.2. Dieses kann auch in den älteren Verfahren genutzt werden, weshalb in Abschnitt 4.9.1.2 die Integration von HandSegNet in die älteren Verfahren getestet wird. Die neueren in dieser Arbeit entwickelten Verfahren für die Handpositionsbestimmung aus den Abschnitten 4.7 und 4.8.1 werden in Abschnitt 4.9.1.3 auf dem IOSB-Hand-Tracking-Datensatz evaluiert.

4.9.1.1 Vergleich der älteren Ansätze

Die Verfahren Schwerpunkt-Tracking aus Abschnitt 4.3.1 sowie Std-Partikelfilter-Tracking und Shape-Partikelfilter-Tracking aus Abschnitt 4.3.2 zählen zu den älteren Verfahren, die in dieser Arbeit entwickelt wurden. Sie nutzen die Hautfarbenerkennung per statischem RGB-Histogramm und wurden in Hammer und Beyerer [Ham13b] sowie Abschnitt 4.3.2 vorgestellt. Durch die Hautfarbensegmentierung mit RDFs und einer Datenbank von Umgebungen bei Li und Kitani [Li13a] entstand ein weiteres Verfahren zur Hautfarbensegmentierung. Dieses Verfahren wird von MACS, siehe Hammer et al. [Ham16b]

und Abschnitt 4.4, genutzt und fusioniert die Hautfarbensegmentierung zusätzlich mit einer Segmentierung der Bewegung. Es folgte die Publikation von AfM in Hammer et al. [Ham16a], siehe Abschnitt 4.6, welche das MACS Ergebnis nutzt, um aus den Handregionshypothesen des ACF-Verfahrens dasjenige herauszusuchen, welches die Hand am besten beschreibt. Diese fünf Verfahren werden nachfolgend auf dem IOSB-Hand-Tracking-Datensatz gegenübergestellt.

Tabelle 4.1: Erkennungswerte der älteren Verfahren auf dem IOSB-Hand-Tracking-Datensatz

	TPR	FPR	PREC	F1	ACC
Schwerpunkt-Tracking	0,66	0,02	0,96	0,78	0,83
Std-Partikelfilter	0,61	0,01	0,99	0,75	0,81
Shape-Partikelfilter	0,65	0,03	0,95	0,77	0,82
MACS	0,88	0,01	0,99	0,93	0,94
AfM	0,88	0,01	0,99	0,93	0,94

Zuerst werden die maximal möglichen Erkennungswerte betrachtet, siehe Abschnitt 4.1.2. Diese sind in Tabelle 4.1 dargestellt. Während Schwerpunkt-Tracking und Shape-Partikelfilter Falsch-Positiv-Raten (FPR) von 2 bis 3 % erreichen, liegen Std-Partikelfilter, MACS und AfM bei unter 1 %. Deutlich erkennbar ist der Vorteil der Hautfarbensegmentierung mit dem RDF-basierten Verfahren von Li und Kitani [Li13a], welches bei MACS und AfM genutzt wird, mit einer 20 % höheren Trefferquote (TPR) im Vergleich zu den Verfahren Schwerpunkt-Tracking, Std-Partikelfilter und Shape-Partikelfilter, welche statische RGB-Histogramme für die Repräsentation der Hautfarbe nutzen. Die Genauigkeit (PREC) liegt bei Std-Partikelfilter, MACS und AfM bei ca. 99 %, während sie beim Schwerpunkt- und Shape-Partikelfilter-Tracking durch die schlechtere Falsch-Positiv-Rate bei ca. 95 % liegt. Die Trefferquote und Genauigkeit kombinierende F1-Maßzahl (F1) liegt bei MACS und AfM dank der vergleichsweise hohen Richtig-Positiv-Rate bei ca. 93 % und damit deutlich höher, als bei den drei anderen Verfahren mit maximal 78 %. Ähnliches gilt für die Korrektklassifikationsrate (ACC), welche bei MACS und AfM bei ca. 94 % liegt und bei den anderen Verfahren maximal 83 % erreicht.

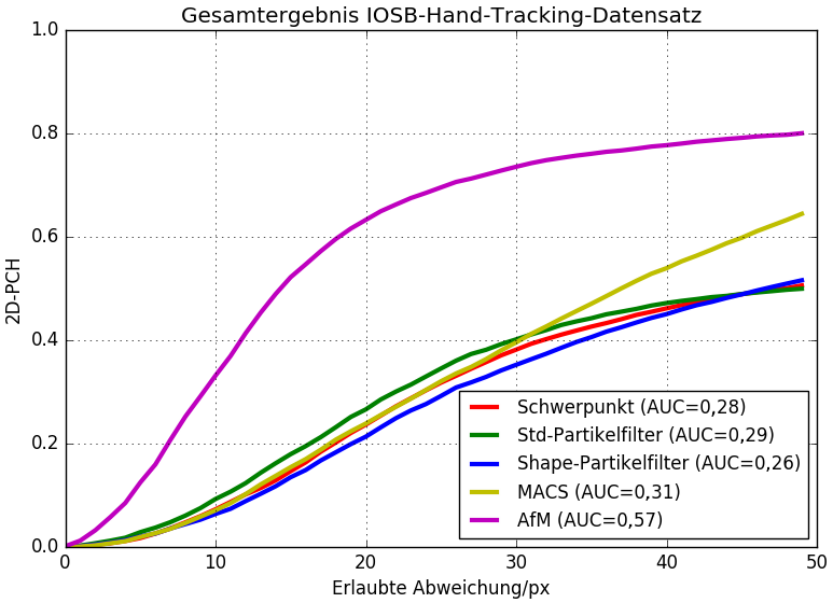


Abbildung 4.31: Darstellung des Anteils korrekt erkannter Hände (2D-PCH) in Abhängigkeit der maximal erlaubten Abweichung bei der Handpositionsschätzung für die älteren Verfahren auf dem IOSB-Hand-Tracking-Datensatz

Da diese Werte nur die maximal möglichen Erkennungswerte darstellen und dem 2D-PCH-Wert, siehe Abschnitt 4.1.2, bei unendlich großer erlaubter Abweichung entsprechen, wird nachfolgend die räumliche Genauigkeit der Handpositionsschätzung betrachtet. Diese wird in Abbildung 4.31 dargestellt. Schwerpunkt-, Std-Partikelfilter- und Shape-Partikelfilter-Tracking weisen bis zu einer erlaubten Abweichung von 50 Pixeln einen ähnlichen Verlauf auf, der bei 50 Pixeln eine Trefferquote von ca. 50 % annimmt. MACS hat bis 30 Pixel Abweichung einen ähnlichen Verlauf, die Trefferquote steigt dann aber stärker an und erreicht bei 50 Pixeln Abweichung eine Trefferquote von ca. 64 %. Bis zu den maximal erreichbaren 88 % Trefferquote liegt hier noch ein großer Bereich von Bildern, in denen richtigerweise eine Hand gefunden wurde, die Positionsschätzung aber zu ungenau ist. MACS verrichtet folglich bei der Klassifikation, ob ein Bild eine Hand enthält, einen deutlich besseren

Dienst, als bei der Handpositionsschätzung. Aus diesem Grund wurde MACS um die Handregionshypothesen des ACF-Verfahrens erweitert, was im AfM-Verfahren mündete, welches sich in der räumlichen Genauigkeit klar von den anderen Verfahren abhebt und bei einer Abweichung von 25 Pixeln bereits eine Trefferquote von ca. 70 % erreicht, welche bei 50 Pixeln auf 80 % ansteigt und bei unendlicher Abweichung 88 % erreicht.

4.9.1.2 Auswertung der Integration von HandSegNet in die älteren Verfahren

Die zuvor ausgewerteten Verfahren bedürfen einer Hautfarbensegmentierung, die durch die Handsegmentierung von HandSegNet ersetzt werden kann. Hierfür stehen zwei Varianten zur Wahl: Zum einen das originale HandSegNet von Zimmermann und Brox [Zim17], das nachfolgend durch *Std-HSN* gekennzeichnet wird, und zum anderen die in dieser Arbeit trainierte Variante *IOSB-HSN*. Für die Integration wurde das Shape-Partikelfilter-Tracking als Repräsentant für die Verfahren aus Abschnitt 4.3 sowie MACS und AfM gewählt. Tabelle 4.2 zeigt die maximal erreichbaren Erkennungswerte und Abbildung 4.32 zeigt die Trefferquote unter Variation der maximal erlaubten Abweichung. Bei allen Verfahren ist zu erkennen, dass sie unter Nutzung des CNN HandSegNet eine deutliche Verbesserung bei der maximal erreichbaren Falsch-Positiv-Rate erzielen. Diese sinkt teilweise auf weit unter 0,1 %. Auch die Genauigkeit liegt fast bei 100 %. Weitere deutliche Verbesserungen lassen sich für das Shape-Partikelfilter-Tracking unter Nutzung der Std-HSN-Handsegmentierung erkennen. So steigt das F1-Maß von 77 % auf über 85 % und die Korrektklassifikationsrate von 82 % auf 88 %. Auch die Trefferquote unter Variation der erlaubten Abweichung wird ab ca. 27 Pixeln deutlich besser und erzielt eine AUC von 28 % im Vergleich zu vorher 26 %. Wird anstelle der Std-HSN-Variante die IOSB-HSN-Variante genutzt, kann das F1-Maß auf 96 % und die Korrektklassifikationsrate auf 97 % verbessert werden. Die räumliche Genauigkeit der Handpositionsbestimmung steigt ebenfalls deutlich, siehe Abbildung 4.32, und die AUC steigt von 28 % mit Std-HSN-Variante auf 43 %. Dies zeigt deutlich die Verbesserung durch das in

dieser Arbeit durchgeführte zusätzliche Training des originalen HandSegNet-Netzes, siehe Abschnitt 4.2.2. Diese Verbesserung ist bei MACS gleichfalls zu erkennen, wenn dort das IOSB-HSN genutzt wird, da die AUC von 0,31 auf 0,41 steigt. AfM kann bei Nutzung des IOSB-HSN ähnliche Ergebnisse (AUC bei 0,56) jedoch keine Verbesserung erzielen. Dies kann dadurch erklärt werden, dass AfM durch die Verfeinerung der MACS-Hypothese mit Hilfe der ACF-Kandidaten unabhängiger von der Handsegmentierung ist.

Tabelle 4.2: Erkennungswerte der älteren Verfahren auf dem IOSB-Hand-Tracking-Datensatz mit integrierten HandSegNet-Varianten Std-HSN und IOSB-HSN

	TPR	**FPR**	**PREC**	**F1**	**ACC**
Shape-Partikelfilter	0,65	0,03	0,95	0,77	0,82
Shape-Std-HSN	0,74	≈ 0	≈ 1	0,85	0,88
Shape-IOSB-HSN	0,92	≈ 0	≈ 1	0,96	0,97
MACS	0,88	0,01	0,99	0,93	0,94
MACS-Std-HSN	0,71	≈ 0	≈ 1	0,83	0,86
MACS-IOSB-HSN	0,86	≈ 0	≈ 1	0,93	0,94
AfM	0,88	0,01	0,99	0,93	0,94
AfM-Std-HSN	0,76	≈ 0	≈ 1	0,87	0,89
AfM-IOSB-HSN	0,86	≈ 0	≈ 1	0,93	0,94
ACF-Std-HSN	0,83	≈ 0	≈ 1	0,91	0,92
ACF-IOSB-HSN	0,92	≈ 0	≈ 1	0,96	0,96

Bei MACS ergibt sich die Segmentierung aus Fusion der Hautfarben- und Bewegungssegmentierung, siehe Abschnitt 4.4.3. Ohne Hinzunahme der Bewegungssegmentierung sind die Ergebnisse deutlich schlechter, weil die genutzte Hautfarbensegmentierung fehlerhaft ist. Nutzt man allerdings AfM mit dem IOSB-HSN und deaktiviert die Nutzung der Bewegungssegmentierung, ist die Handpositionsbestimmung minimal besser als bei AfM selbst, siehe Ergebnisse des sog. ACF-IOSB-HSN-Verfahrens. Dies spricht wieder für die gute Handsegmentierung mit dem IOSB-HandSegNet.

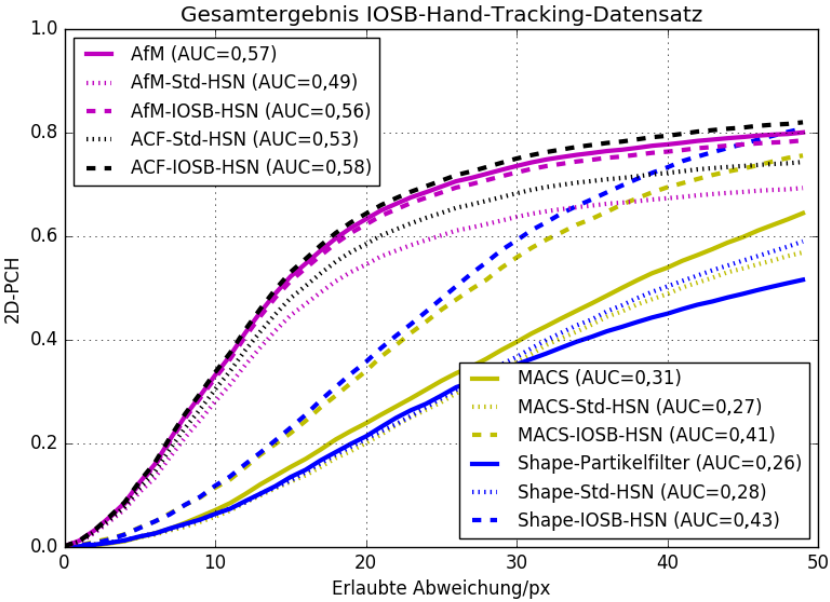


Abbildung 4.32: Darstellung des Anteils korrekt erkannter Hände (2D-PCH) in Abhängigkeit der maximal erlaubten Abweichung bei der Handpositionsschätzung für die älteren Verfahren mit integriertem HandSegNet

4.9.1.3 Evaluation der HSN-Tracking-Varianten

In diesem Abschnitt werden die neuesten der in der vorliegenden Arbeit entwickelten Verfahren, siehe Abschnitt 4.7 sowie 4.8.1, auf dem IOSB-Hand-Tracking-Datensatz evaluiert. Zum Vergleich werden die Ergebnisse von AfM gezeigt, da dieses die bisher besten Ergebnisse der zuvor entwickelten Verfahren erzielte. Tabelle 4.3 zeigt die maximal erreichbaren Erkennungswerte und Abbildung 4.33 die Trefferquote unter Variation der maximal erlaubten Abweichung. HSN-Rect-Direkt als einfachstes Verfahren erzielt bereits einen F1-Wert von 97 %, der besser ist als bei AfM mit 93 %. Auch die Korrektklassifikationsrate ist mit 97 % besser als 94 %. Werden die Handpositionen mit einbezogen, dreht sich das Bild. Die AUC liegt mit 54 % unter der von AfM

mit 57 %. Zusätzlich ist zu erkennen, dass AfM bis zu einer maximalen Abweichung von ca. 28 Pixeln deutlich präziser ist.

Die zusätzliche Bewertung der Hypothesen in HSN-Rect-Tracking erzielt keine Verbesserung zu HSN-Rect-Direct. Die identischen Werte sowohl in der Tabelle als auch bei Betrachtung der Kurve für die Entwicklung des 2D-PCH-Wertes bestätigen nochmal die hohe Güte der Handsegmentierung durch das IOSB-HandSegNet, da die zusätzliche Logik keine Verbesserung erzielt. Dies wird weiter bestätigt durch die Ergebnisse von HSN-Seg-Tracking, dessen geschätzte Handpositionen noch mehr von der pixelweisen Segmentierung des HandSegNet beeinflusst ist. Die maximal erreichbaren Erkennungswerte sind wieder identisch zu denen von HSN-Rect-Tracking. Bei Betrachtung des 2D-PCH-Wertes zeigt sich durchweg eine höhere räumliche Genauigkeit, die sich in einer AUC von 64 % im Vergleich zu 54 % widerspiegelt. Erst die Nutzung der Handpose in HSN-Pose-Tracking kann eine weitere Verbesserung der Schätzung der Handposition erzielen, welche nochmal 2 % mehr bei der AUC erreicht. Wie wichtig die Schätzung der Handpose für das korrekte Auffinden der Hand ist, zeigt sich bei der Evaluation der Handpose im nachfolgenden Abschnitt.

Tabelle 4.3: Erkennungswerte von AfM und der HSN-Tracking-Verfahren mit integriertem IOSB-HSN.

	TPR	**FPR**	**PREC**	**F1**	**ACC**
AfM	0,88	0,01	0,99	0,93	0,94
HSN-Rect-Direct	0,94	≈ 0	≈ 1	0,97	0,97
HSN-Rect-Tracking	0,94	≈ 0	≈ 1	0,97	0,97
HSN-Seg-Tracking	0,94	≈ 0	≈ 1	0,97	0,97
HSN-Pose-Tracking	0,94	≈ 0	≈ 1	0,97	0,97

Zusammenfassend kann festgestellt werden, dass das Verfahren HSN-Pose-Tracking das derzeit beste Verfahren auf dem IOSB-Hand-Tracking-Datensatz ist und diesen als erstes Verfahren zufriedenstellend beherrscht. Das in dieser Arbeit neu trainierte HandSegNet erzeugt eine robuste Segmentierung der Hand, dessen Schwerpunkt mit Hilfe der Handpose am genauesten bestimmt werden kann.

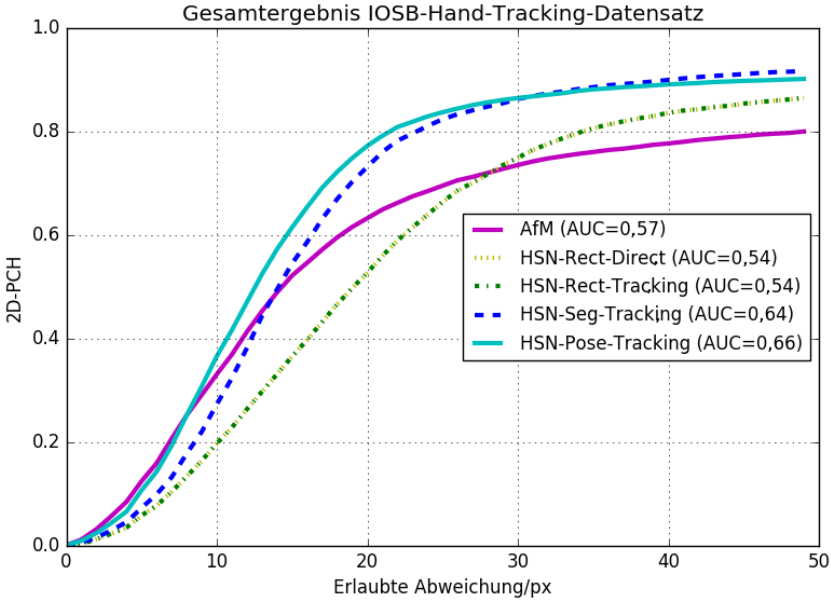


Abbildung 4.33: Darstellung des Anteils korrekt erkannter Hände (PCH) in Abhängigkeit der maximal erlaubten Abweichung bei der Handpositionsschätzung für die HSN-Tracking-Verfahren auf dem IOSB-Hand-Tracking-Datensatz

4.9.2 Beurteilung der Verfahren zur Handposenschätzung

In diesem Abschnitt werden die Ergebnisse der Handposenschätzung auf dem EgoDexter-Datensatz, siehe Abschnitt 4.1.1.2, für die Verfahren aus Abschnitt 4.8.2 präsentiert und mit Hilfe des Anteils korrekt geschätzter Gelenkpositionen (2D-PCK), siehe Abschnitt 4.1.3, verglichen. Abbildung 4.34 zeigt den 2D-PCK unter Variation der erlaubten Abweichung. Zu erkennen ist die deutliche Verbesserung durch das Rückführen der letzten Handpose bei den HPTwF-Varianten, siehe Abschnitt 4.8.2.1, da der AUC-Wert von 52 % auf 61 % steigt. Die Verbesserung durch die Berücksichtigung der Ähnlichkeit von Handposen im HPTwF-Sim-Verfahren, siehe Abschnitt 4.8.2.2, sowie durch die Berücksichtigung der Segmentierung, siehe Abschnitt 4.8.2.3, ist, wie anhand der

Kurven zu erkennen, minimal und sorgt größtenteils für die Verbesserung der Handposenschätzung bei einzelnen Bildern. Der Verbund aller einzelnen Verbesserungen im HPTwF-All-Verfahren, beschrieben am Ende von Abschnitt 4.8.2.3, erreicht mit 62 % den besten AUC-Wert.

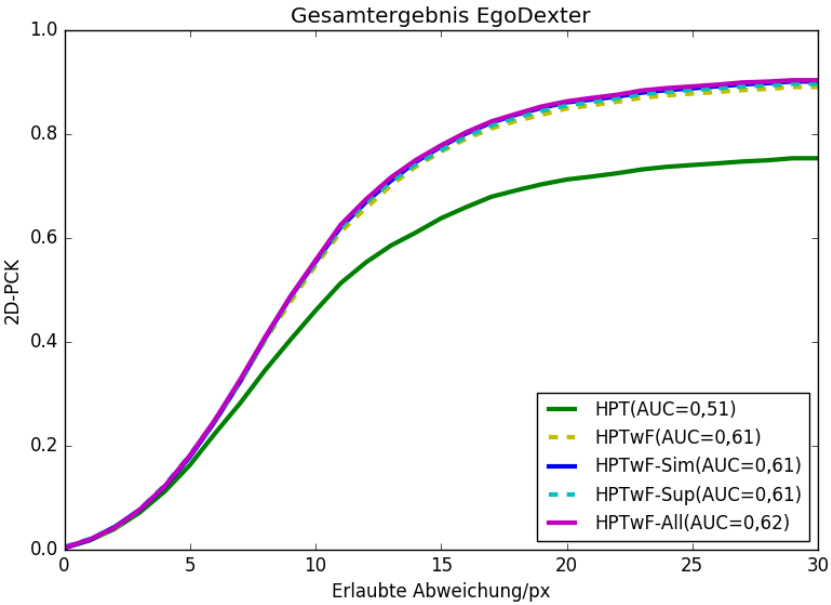


Abbildung 4.34: Darstellung des Anteils korrekt erkannter Gelenkpositionen (2D-PCK) in Abhängigkeit der maximal erlaubten Abweichung bei der Handposenschätzung für HSN-Pose-Tracking-Varianten auf dem EgoDexter-Datensatz

Die öffentliche Verfügbarkeit des EgoDexter-Datensatzes macht die oben vorgestellten Ergebnisse vergleichbar zum Stand der Technik der Handposenschätzung, welcher durch die Verfahren von Zimmermann und Brox [Zim17], Mueller et al. [Mue18] sowie Iqbal et al. [Iqb18] repräsentiert wird, siehe Abschnitt 2.3.2. Diese Verfahren führen eine 3D-Handposenschätzung durch, Ergebnisse für den 2D-PCK auf dem EgoDexter-Datensatz liegen allerdings vor [Mue18, Iqb18] und sind in Abbildung 4.35 dargestellt.

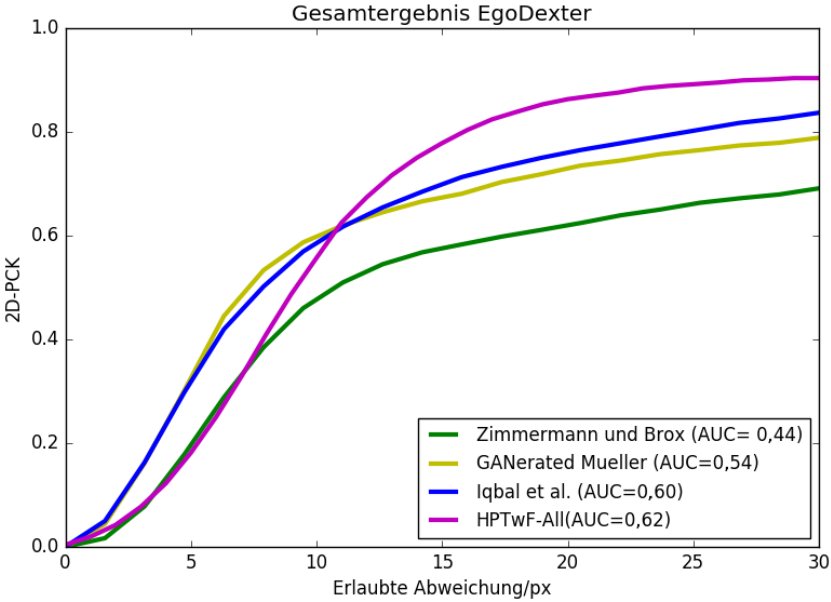


Abbildung 4.35: Darstellung des Anteils korrekt erkannter Gelenkpositionen (2D-PCK) in Abhängigkeit der maximal erlaubten Abweichung bei der Handposenschätzung für den Stand der Technik sowie HPTwF-All

Das Verfahren von Zimmermann und Brox erreicht einen AUC-Wert von 44 %. Es wird mit deutlichen 10 % vom GANerated-Mueller-Verfahren mit 54 % übertroffen, welches für jede erlaubte Abweichung besser ist. Einen besseren AUC-Wert als das Verfahren von Mueller et al. zeigt mit 60 % das Verfahren von Iqbal et al. [Iqb18]. Die höhere räumliche Genauigkeit wird dabei im Wesentlichen ab ungefähr elf Pixeln erlaubter Abweichung erzielt. Das in dieser Arbeit entwickelte HPTwF-All Verfahren, siehe Abschnitt 4.8.2.3, zeigt von null bis sieben Pixeln erlaubter Abweichung einen ähnlichen 2D-PCK wie das Verfahren von Zimmermann und Brox. Ab sieben Pixeln wird es genauer und erzielt bei einer erlaubten Abweichung von ungefähr zehn Pixeln den gleichen 2D-PCK wie die Verfahren von Mueller et al. und Iqbal et al., welche bis dahin eine ähnlich höhere räumliche Genauigkeit aufweisen als das HPTwF-All-Verfahren. Dies liegt vermutlich daran, dass die beiden zuvor

genannten Verfahren die Handpose im dreidimensionalen Raum schätzen, während OpenPose seine Schätzung im Zweidimensionalen durchführt. Ab zehn Pixeln erlaubter Abweichung zeigt das HPTwF-All-Verfahren durchweg einen deutlich besseren 2D-PCK als die anderen zum Vergleich zur Verfügung stehenden Verfahren und erzielt mit 62 % den besten AUC-Wert, der auf dem EgoDexter-Datensatz bis Februar 2019 publiziert wurde.

Dieses Ergebnis zeigt, dass eine Optimierung der Handregionsbestimmung, wie sie in der vorliegenden Arbeit mit dem HPTwF-All-Verfahren entwickelt wurde, die räumliche Genauigkeit aktuell entwickelter Handposenschätzer auf einfache Weise verbessern kann. Aktuelle Verfahren verwenden zwar teilweise vorangegangene Handpositionen weiter, hier besteht aber anscheinend ein Defizit, welches die Verfahren ungenauer werden oder gar das Verfolgen der Handregion fehlschlagen lässt.

5 Diskussion und Ausblick

5.1 Diskussion

Visuelle Augmented Reality hat das Potential, die Art und Weise, wie der Mensch mit Maschinen kommuniziert, derart zu verändern, dass ganze Klassen von Geräten wie Bildschirme als auch Smartphones überflüssig werden können. Grundvoraussetzung dafür sind angenehm zu tragende binokulare AR-Brillen mit einem großen Sichtfeld für visuelle Einblendungen mit hohem Kontrast, so dass virtuelle Elemente als Teil der realen Umgebung dargestellt werden können. Gleichzeitig bedürfen derartige AR-Systeme einer intuitiven Interaktion mit ihrem Benutzer, um akzeptiert zu werden. Blick und Handgesten bilden neben Sprache die Interaktionstechniken der Wahl, um mit virtuellen Elementen zu interagieren, denn der Blick ist ein Hinweis auf die Aufmerksamkeit und mit den Händen kommuniziert der Mensch und manipuliert Objekte auf bekannte natürliche Weise. Die vorliegende Arbeit beschäftigte sich deshalb mit der Analyse des Blickes für eine implizite unbewusste Interaktion und mit der Erfassung von Handgesten für die explizite Interaktion in mobilen Anwendungen.

Um Blickanalyse für die Interaktion in mobilen Anwendungen nutzen zu können, muss sie die Erfassung des dreidimensionalen Fokus der Aufmerksamkeit in Echtzeit ermöglichen. In dieser Arbeit wurde eines der ersten Verfahren zur vollautomatischen echtzeitfähigen Blickbewegungsanalyse in dreidimensionalen Umgebungen anhand eines Beispiels aus dem Museumskontext vorgestellt. Dafür wurde eine 3D-Blickpunktberechnung mit einem monokularen Blickmessgerät und dreidimensionalen Umgebungsmodellen sowie eine darauf aufsetzende echtzeitfähige Blickanalyse von 3D-Blickpfaden realisiert, als kein kommerziell erhältliches Blickmessgerät inkl. ausgelieferter

Software diese Aufgabe leisten konnte. In diesem Zusammenhang wurde ein Defizit an Techniken für die Darstellung dreidimensionalen Blickverhaltens als Heatmaps festgestellt und das echtzeitfähige Verfahren namens Projected Gaussians zur realistischen Visualisierung von Heatmaps in dreidimensionalen Umgebungen entwickelt. Dieses Verfahren ist das weltweit einzige, das die visuelle Schärfe des menschlichen Blickes in die Szene projiziert und damit am nächsten am physikalischen Prozess der Wahrnehmung bleibt. Kein zuvor vorgestelltes Verfahren berücksichtigte Verdeckungen oder ermöglichte eine von der Polygonstruktur unabhängige Einfärbung von Oberflächen. Projected Gaussians ist weiterhin zum Stand der Technik in diesem Bereich zu zählen, welchen es zusammen mit einem neueren Verfahren bildet, das die Methode zur Berechnung von Verdeckungen aus dem Projected Gaussians Verfahren sowie den im Rahmen der Veröffentlichung getätigten Vorschlag zur Nutzung einer weiteren Textur zur Speicherung der Gewichtungen übernommen hat. Das in dieser Arbeit entwickelte Projected Gaussians Verfahren hat folglich den Stand der Technik in diesem Bereich bis heute geprägt. An einem realen Beispiel aus dem Museumskontext konnten die entwickelten Verfahren im Bereich der Blickanalyse auf echte Blickdaten angewendet werden, um qualitativ und quantitativ zu zeigen, wie eine Audioführung Einfluss auf die visuelle Aufmerksamkeit und eine vollautomatische Blickanalyse kurzer Blickpfade zur Detektion unbewusst bekundeten Interesses genutzt werden kann, um die Aufmerksamkeit nachzuvollziehen. Gerade dieses Nachvollziehen der Aufmerksamkeit kann in zukünftigen intelligenten Systemen verwendet werden, um sinnvoll auf den Nutzer einzugehen.

Für die explizite Interaktion mit den Händen beschäftigte sich diese Arbeit mit dem ersten Schritt der Handgestenerkennung in monokularen Farbbildern, der Handregionsbestimmung, bei der die Region der Hand in einem Kamerabild ermittelt wird. Die Verfahren erzeugen hierfür eine Segmentierung der Hand. Zuerst entwickelte Verfahren nutzen hierfür Hautfarbenmodelle wie (nicht-)parametrische Hautfarbenmodelle oder Hautfarbenklassifikatoren auf Basis von Entscheidungsbäumen. Diese wurden mit einer Segmentierung des optischen Flusses kombiniert, um den Vordergrund zu bestimmen, in welchem sich eine Hand zwangsläufig befinden muss, falls sie sich sichtbar bewegt. Die Fusion von Hautfarben- und Bewegungssegmentierung

mündete im Verfahren *MACS*. Es folgte die zusätzliche Nutzung eines Objektdetektors auf Basis aggregierter Bildkanal-Merkmale, dem ACF-Detektor, der Handregionshypothesen liefert. Eine geschickte Fusion von vorangegangenen Handpositionsschätzungen mit den neuen Handregionshypothesen des ACF-Detektors verbesserte die Bestimmung der Handposition im Vergleich zu vorherigen Ansätzen deutlich und resultierte im *AfM* genannten Verfahren. Durch den Einzug von CNNs in die Welt der Bildverarbeitung wurde das CNN HandSegNet öffentlich verfügbar. Dieses CNN wurde durch ein Training auf Daten des in dieser Arbeit entwickelten IOSB-Hand-Tracking-Datensatz adaptiert. Die damit erzielten Segmentierungen der Hände erwiesen sich gegenüber den zuvor entwickelten Verfahren als derart überlegen, dass auf die Nutzung des optischen Flusses als zusätzliches Hilfsmittel verzichtet werden konnte. Durch eine Analyse der segmentierten Bereiche mit Hilfe des so entstandenen IOSB-HandSegNet und der ermittelten Konfidenzwerte entstand eine Handregionsbestimmung, die gleichsam als Hand-Klassifikator genutzt werden konnte und Handregionshypothesen berechnete, welche deutlich besser waren als die des ACF-Detektors. Die in AfM etablierte Fusion von vorangegangenen Handpositionen mit aktuell geschätzten Handregionshypothesen wurde beibehalten. Mit dem so entstandenen HSN-Seg-Tracking Verfahren konnten auf dem IOSB-Hand-Tracking-Datensatz zum ersten Mal überzeugende Ergebnisse bei der Handregionsbestimmung erzielt werden.

Mit Hilfe eines öffentlich zur Verfügung gestellten 2D-Handposenschätzers, welcher lediglich Handregionen als Eingabe benötigt, konnten zusätzlich geschätzte Handposen genutzt werden, um Handregionen zu bewerten. Der in dieser Arbeit entwickelte Einbezug des 2D-Handposenschätzers resultierte im Verfahren HSN-Pose-Tracking, welches die Ergebnisse auf dem IOSB-Hand-Tracking-Datensatz nochmal verbesserte. Um einen Vergleich zum aktuellen Stand der Technik herleiten zu können, wurde der bekannte öffentliche Datensatz EgoDexter einbezogen. Es zeigte sich, dass das IOSB-HandSegNet auch hier der Originalversion von HandSegNet überlegen ist. Nach einer Analyse der Schwächen auf diesem Datensatz wurde HSN-Pose-Tracking dahingehend erweitert, dass es die zuletzt geschätzte Handregion zusätzlich als Handregionshypothese nutzte. Weiter wurde der Fusionsmechanismus, der mit AfM eingeführt wurde, um Gewichtungsterme

erweitert, die ähnliche Handposen bevorzugen und fördern, dass die resultierende Handpose innerhalb der aktuellen Handsegmentierung liegt. Dieses HPTwF-All genannte Verfahren übertrifft den aktuellen Stand der Technik (bis inkl. Februar 2019) im Bereich Handposenerkennung auf den Sequenzen des EgoDexter-Datensatzes bei der 2D-Handposenerkennung, selbst wenn einige dieser Verfahren eine 3D-Handposenerkennung durchführen. Dieses Ergebnis zeigt, dass eine Optimierung der Handregionsbestimmung, wie sie in der vorliegenden Arbeit mit dem HPTwF-All-Verfahren entwickelt wurde, die Genauigkeit aktuell entwickelter Handposenschätzer deutlich verbessern kann.

5.2 Ausblick

Die Verschiebung des Zielmarktes der AR-Brille Google Glass auf den Bereich Unternehmenskunden zeigt, dass es bereits heute sinnvolle Anwendungen für mobile AR-Systeme gibt. Auch wenn diese Brille eher als kleiner am Kopf getragener Monitor verstanden werden muss und nicht als vollwertiges AR-System gesehen werden kann, betont dieses Gerät die Sinnhaftigkeit, Informationen näher an den Menschen zu bringen und sie somit besser verständlich zu gestalten. Neben Verbesserungen im Bereich optischer Anzeigetechnologien müssen robuste Verfahren für die Posenschätzung kopfgetragener Systeme ohne optische Referenzmarkierungen und im Optimalfall für rein passive Sensorik realisiert werden, da aktive Sensorik mit komplexerer Hardware und höherem Energieverbrauch einhergeht. Gleiches gilt für die Interaktion.

Neben dem Problem der Posenschätzung von kopfgetragenen Systemen steht der Nutzung des Blickes für die Interaktion weiterhin die Kalibrierung auf jeden Nutzer und der Blickpunktberechnung unter allgegenwärtigen Bedingungen (Sonneneinstrahlung, nicht entspiegelte Brillengläser) im Weg. Eine Lösung für den nachfolgenden Schritt der Blickanalyse, sowohl manuell als auch automatisch, wurde in dieser Arbeit für mobile Anwendungen detailliert betrachtet und eine Machbarkeit gezeigt.

Das Problem der Artefakte des Projected Gaussians Verfahrens könnte durch eine einfache Erweiterung deutlich reduziert werden, indem nach der Berechnung des zu einem Pixel gehörenden 3D-Punktes der Abstand zum 3D-Blickpunkt zusätzlich berechnet wird. Wenn dieser Abstand einen Schwellenwert übersteigt, der linear mit der Entfernung zum Auge ansteigen muss, wird dem zugehörigen Pixel keine Gewichtung zugeteilt. Die Gewichtung würde sich, um näher an der Realität der Wahrnehmung zu bleiben, weiterhin durch den projizierten Gauß-Wert ergeben. Diese wenigen Berechnungen würden auch die Laufzeit kaum beeinträchtigen.

Eine interessante Alternative für die Blickanalyse in mobilen Anwendungen, bei der auf die Schätzung der Pose des kopfgetragenen Systems relativ zur Umwelt verzichtet werden kann, ist die Nutzung einer Szenensegmentierung im Kamerabild und Verwendung eines binokularen Blickmessgerätes. Ein dann relativ zum Gerät berechneter 3D-Blickpunkt kann zurück in die Szenenkamera projiziert und durch die Segmentierung mit Segmenten verknüpft werden, sodass eine vollautomatische AOI-basierte Blickanalyse im Kamerabild möglich wird. Grundvoraussetzung hierfür ist entsprechend eine robuste Segmentierung und Erkennung von Objekten, sowie eine akkurate 3D-Blickpunktberechnung.

Bei der Handposenschätzung ist festzustellen, dass aktuelle 3D-Handposenschätzer, die die 3D-Handpose ohne Tiefeninformationen schätzen, noch nicht die Genauigkeit erreichen, die mit Tiefeninformationen erzielt werden kann. Hier besteht weiter Verbesserungsbedarf. Ein möglicher Ansatzpunkt ist die Wiederverwendung vorangegangener Handposenschätzungen, wie in dieser Arbeit zur Verbesserung der Handregionsbestimmung vorgestellt. Aktuelle Verfahren verwenden zwar teilweise vorangegangene Handposen, hier besteht aber anscheinend ein Defizit, was dazu führt, dass die Verfahren ungenauer werden oder das Verfolgen der Handregion fehlschlägt.

Literatur

[Acc19] Accuvein: AccuVein Vein Visualization-Improves IV 1-Stick Success 3.5x. 2019. url: https://www.accuvein.com/ (besucht am 11. 02. 2019) (siehe S. 6).

[Aja10] Ajanki, A. u. a.: „Contextual information access with Augmented Reality“. In: *Machine Learning for Signal Processing (MLSP), 2010 IEEE International Workshop on*. 2010, S. 95–100. doi: 10.1109/MLSP.2010.5589228 (siehe S. 34).

[Ata14] Atac, Robert; Spink, Scott; Calloway, Tom und Foxlin, Eric: „Scorpion Hybrid Optical-based Inertial Tracker (HObIT) test results“. In: *Display Technologies and Applications for Defense, Security, and Avionics VIII; and Head- and Helmet-Mounted Displays XIX*. Display Technologies and Applications for Defense, Security, and Avionics VIII; and Head- and Helmet-Mounted Displays XIX. Bd. 9086. International Society for Optics und Photonics, 13. Juni 2014, 90860U. doi: 10.1117/12.2050363 (siehe S. 18).

[Bad09] Bader, Thomas; Räpple, Rene und Beyerer, Jürgen: „Fast invariant contour-based classification of hand symbols for HCI“. In: *Computer analysis of images and patterns: 13th international conference, CAIP 2009, Münster, Germany, September 2-4, 2009 ; proceedings*. Lecture notes in computer science; 5702. Springer, Berlin [u.a.], 2009, S. 689–696 (siehe S. 41, 47).

[Bad11] Bader, Thomas: „Multimodale Interaktion in Multi-Display-Umgebungen“. Diss. 2011. 240 S. doi: 10.5445/KSP/1000024819 (siehe S. 47).

[Bad15] Badrinarayanan, Vijay; Kendall, Alex und Cipolla, Roberto: „SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation“. In: *arXiv:1511.00561 [cs]* (Nov. 2015). arXiv: 1511.00561. url: http://arxiv.org/abs/1511.00561 (besucht am 17. 01. 2019) (siehe S. 53).

[Bak11] Baker, Simon; Scharstein, Daniel; Lewis, J.P.; Roth, Stefan; Black, MichaelJ. und Szeliski, Richard: „A Database and Evaluation Methodology for Optical Flow“. English. In: *International Journal of Computer Vision* 92 (1 2011), S. 1–31. doi: 10.1007/s11263-010-0390-2 (siehe S. 117).

[Bam15] Bambach, Sven; Lee, Stefan; Crandall, David J. und Yu, Chen: „Lending A Hand: Detecting Hands and Recognizing Activities in Complex Egocentric Interactions“. In: *The IEEE International Conference on Computer Vision (ICCV)*. Dez. 2015 (siehe S. 50, 59, 109).

[Bar18] Barz, Michael; Daiber, Florian; Sonntag, Daniel und Bulling, Andreas: „Error-aware gaze-based interfaces for robust mobile gaze interaction“. In: *Proceedings of the 2018 ACM Symposium on Eye Tracking Research & Applications - ETRA '18*. Warsaw, Poland: ACM Press, 2018, S. 1–10. doi: 10.1145/3204493.3204536 (siehe S. 35).

[Bau12] Baumgarten, J.; Schuchert, T.; Voth, S.; Wartenberg, P.; Richter, B. und Vogel, U.: „Aspects of a head-mounted eye-tracker based on a bidirectional OLED microdisplay“. In: *Journal of information display* 13, No. 2 (2012), S. 67–71 (siehe S. 14).

[Bha43] Bhattacharyya, A.: „On A Measure of Divergence Between Two Statistical Populations Defined by their Probability Distributions“. In: *Bulletin of Cal. Math. Soc.* 35.1 (1943), S. 99–109 (siehe S. 120).

[Bin05] Binh, Nguyen Dang; Shuichi, Enokida und Ejima, Toshiaki: „Real-Time Hand Tracking and Gesture Recognition System“. In: *Proceedings of International Conference on Graphics, Vision and Image Processing (GVIP-05*. 2005, S. 362–368 (siehe S. 40).

[Bla17] Blascheck, T.; Kurzhals, K.; Raschke, M.; Burch, M.; Weiskopf, D. und Ertl, T.: „Visualization of Eye Tracking Data: A Taxonomy and Survey“. In: *Computer Graphics Forum* 36.8 (Dez. 2017), S. 260–284. doi: 10.1111/cgf.13079 (siehe S. 38).

[Bre01] Breiman, Leo: „Random Forests“. In: *Machine Learning* 45.1 (Okt. 2001), S. 5–32. doi: 10.1023/A:1010933404324 (siehe S. 43).

[Bus35] Buswell, G. T.: How People Look at Pictures. Chicago: University of Chicago Press, 1935 (siehe S. 32, 71).

[Cak06] Cakmakci, O. und Rolland, J.: „Head-worn displays: a review“. In: *Journal of Display Technology* 2.3 (Sep. 2006), S. 199–216. doi: 10.1109/JDT.2006.879846 (siehe S. 14).

[Cam12] Cameron, Alexander A.: „Optical waveguide technology and its application in head-mounted displays“. In: *Head- and Helmet-Mounted Displays XVII; and Display Technologies and Applications for Defense, Security, and Avionics VI*. Hrsg. von Marasco, Peter L.; II, Paul R. Havig; Desjardins, Daniel D. und Sarma, Kalluri R. Bd. 8383. International Society for Optics und Photonics. SPIE, 2012, S. 109–119. doi: 10.1117/12.923660 (siehe S. 15).

[Cam13a] Cameron, Alex: „“Head up and eyes out” advances in head mounted displays capabilities“. In: *Display Technologies and Applications for Defense, Security, and Avionics VII*. Display Technologies and Applications for Defense, Security, and Avionics VII. Bd. 8736. International Society for Optics und Photonics, 4. Juni 2013, 87360G. doi: 10.1117/12.2021135 (siehe S. 17).

[Cam13b] Campbell, Mark; Moran, Aidan und Kenny, Ian: „Characteristics of expertise in able and disabled elite golfers: the role of vision and technique“. In: *British Association of Sport and Exercise Sciences Annual Conference*. British Association of Sport und Exercise Sciences, 2013. url: http://hdl.handle.net/10344/3372 (besucht am 30. 01. 2014) (siehe S. 37).

[Cam15] Cameron, Alex: „In the blink of an eye: head mounted displays development within BAE Systems“. In: *Display Technologies and Applications for Defense, Security, and Avionics IX; and Head- and Helmet-Mounted Displays XX*. Bd. 9470. International Society for Optics und Photonics, Mai 2015, S. 94700V. doi: 10.1117/12.2181380 (siehe S. 14, 15).

[Cao17] Cao, Zhe; Simon, Tomas; Wei, Shih-En und Sheikh, Yaser: „Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields“. In: *CVPR*. 2017 (siehe S. 51).

[Cao18] Cao, Zhe; Hidalgo, Gines; Simon, Tomas; Wei, Shih-En und Sheikh, Yaser: „OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields“. In: *arXiv:1812.08008 [cs]* (Dez. 2018). arXiv: 1812.08008. url: http://arxiv.org/abs/1812.08008 (besucht am 17. 01. 2019) (siehe S. 51, 52, 136).

[Cas12] Casiez, Géry; Roussel, Nicolas und Vogel, Daniel: „1 € filter: a simple speed-based low-pass filter for noisy input in interactive systems“. In: *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems - CHI '12*. Austin, Texas, USA: ACM Press, 2012, S. 2527. doi: 10.1145/2207676.2208639 (siehe S. 53).

[Chi17] Chi, Wanli; Saarikko, Pasi und Lee, Hee Yoon: „Waveguide display with two-dimensional scanner“. U.S. Patent US20170235143A1. Aug. 2017. url: https://patents.google.com/patent/US20170235143A1/en (besucht am 26. 07. 2018) (siehe S. 2).

[Chy12] Chyou, Te-yu: „A 3D Computer Vision System in Radiotherapy Patient Setup“. In: (2012). url: https://ir.canterbury.ac.nz/handle/10092/7176 (besucht am 11. 02. 2019) (siehe S. 6).

[Cue17] Cuervo, Eduardo: „BEYOND REALITY: Head-Mounted Displays for Mobile Systems Researchers“. In: *GetMobile: Mobile Computing and Communications* 21.2 (4. Aug. 2017), S. 9–15. doi: 10.1145/3131214.3131218 (siehe S. 13, 18).

[Dal05] Dalal, N. und Triggs, B.: „Histograms of oriented gradients for human detection“. In: *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. Bd. 1. Juni 2005, 886–893 vol. 1. doi: 10.1109/CVPR.2005.177 (siehe S. 49, 123).

[Dam12] Damala, Areti; Stojanovic, Nenad; Schuchert, Tobias; Moragues, Jorge; Cabrera, Ana und Gilleade, Kiel: „Adaptive Augmented Reality for Cultural Heritage: ARtSENSE Project“. In: *Progress in Cultural Heritage Preservation*. Hrsg. von Ioannides, Marinos; Fritsch, Dieter; Leissner, Johanna; Davies, Rob; Remondino, Fabio und Caffo, Rossella. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2012, S. 746–755 (siehe S. 6, 40).

[Dam18] Damen, Dima u. a.: „Scaling Egocentric Vision: The EPIC-KITCHENS Dataset“. In: *arXiv:1804.02748 [cs]* (Apr. 2018). arXiv: 1804.02748. url: http://arxiv.org/abs/1804.02748 (besucht am 10. 01. 2019) (siehe S. 60).

[Dey15] Deyle, Travis: Valve's "Lighthouse"Tracking System May Be Big News for Robotics | Hizook. Mai 2015. url: http://www.hizook.com/blog/2015/05/17/valves-lighthouse-tracking-system-may-be-big-news-robotics (besucht am 24. 09. 2018) (siehe S. 18).

[Dib18] Dibra, Endri; Melchior, Silvan; Balkis, Ali; Wolf, Thomas; Öztireli, Cengiz und Groß, Markus: „Monocular RGB Hand Pose Inference From Unsupervised Refinable Nets“. In: *CVPR Workshops*. 2018 (siehe S. 53, 55).

[Dol14] Dollar, P.; Appel, R.; Belongie, S. und Perona, P.: „Fast Feature Pyramids for Object Detection“. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36.8 (Aug. 2014), S. 1532–1545. doi: 10.1109/TPAMI.2014.2300479 (siehe S. 123).

[Duc12] Duchowski, Andrew T.; Price, Margaux M.; Meyer, Miriah und Orero, Pilar: „Aggregate gaze visualization with real-time heatmaps“. In: *Proceedings of the Symposium on Eye Tracking*

Research and Applications. ETRA '12. Santa Barbara, California: ACM, 2012, S. 13–20. DOI: 10.1145/2168556.2168558 (siehe S. 36, 87).

[Elb18] Elbamby, Mohammed S.; Perfecto, Cristina; Bennis, Mehdi und Doppler, Klaus: „Towards Low-Latency and Ultra-Reliable Virtual Reality". In: *arXiv:1801.07587 [cs, math]* (Jan. 2018). arXiv: 1801.07587. URL: http://arxiv.org/abs/1801.07587 (besucht am 24. 09. 2018) (siehe S. 15).

[End99] Endsley, M. R.: „Situation Awareness in Aviation Systems. IN: HANDBOOK OF AVIATION HUMAN FACTORS". In: *Handbook of Aviation Human Factors, Publication of: Lawrence Erlbaum Associates, Incorporated* (1999) (siehe S. 1, 11).

[Eng10] Engelbrecht, M.; Betz, J.; Klein, C. und Rosenberg, R.: „Dem Auge auf der Spur: Eine historische und empirische Studie zur Blickbewegung beim Betrachten von Gemälden." In: *IMAGE 11 (January 2010)* (2010) (siehe S. 32, 71).

[Eng14] Engel, Jakob; Schöps, Thomas und Cremers, Daniel: „LSD-SLAM: Large-scale direct monocular SLAM". In: *Computer Vision–ECCV 2014*. Springer, 2014, S. 834–849. URL: http://link.springer.com/chapter/10.1007/978-3-319-10605-2_54 (besucht am 23. 02. 2015) (siehe S. 19).

[Eng15] Engel, Jakob; Stuckler, Jorg und Cremers, Daniel: „Large-scale direct SLAM with stereo cameras". In: *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Hamburg, Germany: IEEE, Sep. 2015, S. 1935–1942. DOI: 10.1109/IROS.2015.7353631 (siehe S. 19).

[Eng18] Engel, J.; Koltun, V. und Cremers, D.: „Direct Sparse Odometry". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40.3 (März 2018), S. 611–625. DOI: 10.1109/TPAMI.2017.2658577 (siehe S. 19).

[Erg11] Ergoneers: Dikablis - The Eye Tracking System. 2011. URL: http://real.psych.ubc.ca/images/9/9b/SW_Dikablis_Handbuch_V2.0_ENG.pdf (besucht am 12. 02. 2019) (siehe S. 25, 27).

[Erg19] Ergoneers: Mobiles Eye Tracking System für Verhaltensforschung: Dikablis Glasses 3. 2019. URL: https://www.ergoneers.com/eye-tracking/dikablis-glasses/ (besucht am 14. 02. 2019) (siehe S. 25).

[Fat11] Fathi, A.; Ren, X. und Rehg, J. M.: „Learning to recognize objects in egocentric activities". In: *CVPR 2011*. Juni 2011, S. 3281–3288. DOI: 10.1109/CVPR.2011.5995444 (siehe S. 49, 55, 58).

[Fer91] Ferrin, Frank J.: „Survey of helmet tracking technologies". In: *Large Screen Projection, Avionic, and Helmet-Mounted Displays*. Large Screen Projection, Avionic, and Helmet-Mounted Displays. Bd. 1456. International Society for Optics und Photonics, 1. Aug. 1991, S. 86–95. DOI: 10.1117/12.45422 (siehe S. 16, 18).

[Fin05] Findlay, J. M.: „Covert attention and saccadic eye movements." In: *Neurobiology of attention*. Hrsg. von Itti, L.; Rees, G. und Tsotsos, J. London ; New York: Elsevier Academic Press, März 2005, S. 114–117 (siehe S. 24).

[Fox00] Foxlin, Eric M.: „Head tracking relative to a moving vehicle or simulator platform using differential inertial sensors". In: *Helmet- and Head-Mounted Displays V*. Helmet- and Head-Mounted Displays V. Bd. 4021. International Society for Optics und Photonics, 23. Juni 2000, S. 133–145. DOI: 10.1117/12.389141 (siehe S. 18).

[Fox04] Foxlin, E.; Altshuler, Y.; Naimark, L. und Harrington, M.: „FlightTracker: a novel optical/inertial tracker for cockpit enhanced vision". In: *Third IEEE and ACM International Symposium on Mixed and Augmented Reality*. Third IEEE and ACM International Symposium on Mixed and Augmented Reality. Nov. 2004, S. 212–221. DOI: 10.1109/ISMAR.2004.32 (siehe S. 16, 18).

[Fre08] Freedman, Edward G.: „Coordination of the Eyes and Head during Visual Orienting“. In: *Experimental brain research. Experimentelle Hirnforschung. Experimentation cerebrale* 190.4 (Okt. 2008), S. 369–387. doi: 10.1007/s00221-008-1504-8 (siehe S. 4).

[Fre97] Freund, Yoav und Schapire, Robert E: „A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting“. In: *Journal of Computer and System Sciences* 55.1 (Aug. 1997), S. 119–139. doi: 10.1006/jcss.1997.1504 (siehe S. 123).

[Gar18] Garcia-Hernando, Guillermo; Yuan, Shanxin; Baek, Seungryul und Kim, Tae-Kyun: „First-Person Hand Action Benchmark with RGB-D Videos and 3D Hand Pose Annotations“. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. Salt Lake City, UT, USA: IEEE, Juni 2018. doi: 10.1109/cvpr.2018.00050 (siehe S. 56).

[Gol10] Goldberg, Joseph H und Helfman, Jonathan I: „Comparing Information Graphics: A Critical Look at Eye Tracking“. In: *Proceedings of the 2010 Workshop on BEyond time and errors: novel evaLuation methods for Information Visualization*. BELIV'10. Atlanta, GA, USA: ACM, Apr. 2010, S. 71–78 (siehe S. 29).

[Gol95] Goldberg, JosephH. und Schryver, JackC.: „Eye-gaze-contingent control of the computer interface: Methodology and example for zoom detection“. English. In: *Behavior Research Methods, Instruments, & Computers* 27 (3 1995), S. 338–350. doi: 10.3758/BF03200428 (siehe S. 66).

[Gol99] Goldberg, Joseph H und Kotval, Xerxes P: „Computer interface evaluation using eye movements: methods and constructs“. In: *International Journal of Industrial Ergonomics* 24.6 (1999), S. 631–645. doi: 10.1016/S0169-8141(98)00068-7 (siehe S. 29).

[Gom17] Gomez-Donoso, Francisco; Orts-Escolano, Sergio und Cazorla, Miguel: „Large-scale Multiview 3D Hand Pose Dataset“. In: *arXiv:1707.03742 [cs]* (Juli 2017). arXiv: 1707.03742. url: http:

//arxiv.org/abs/1707.03742 (besucht am 17. 01. 2019) (siehe S. 52, 55, 57).

[Gue06] Guestrin, Elias Daniel D. und Eizenman, Moshe: „General theory of remote gaze estimation using the pupil center and corneal reflections." In: *IEEE transactions on bio-medical engineering* 53.6 (Juni 2006), S. 1124–1133. doi: 10.1109/TBME.2005.863952 (siehe S. 27).

[Ham13a] Hammer, Jan Hendrik; Maurus, Michael und Beyerer, Jürgen: „Real-time 3D gaze analysis in mobile applications". In: *Proceedings of the 2013 Conference on Eye Tracking South Africa*. ETSA '13. Cape Town, South Africa: ACM, 2013, S. 75–78. doi: 10.1145/2509315.2509333 (siehe S. 61, 71).

[Ham13b] Hammer, JanHendrik und Beyerer, Jürgen: „Robust Hand Tracking in Realtime Using a Single Head-Mounted RGB Camera". In: *Human-Computer Interaction. Interaction Modalities and Techniques*. Hrsg. von Kurosu, Masaaki. Bd. 8007. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2013, S. 252–261. doi: 10.1007/978-3-642-39330-3_27 (siehe S. 109, 146).

[Ham16a] Hammer, J. H.; Qu, C.; Voit, M. und Beyerer, J.: „2D Hand Tracking with Motion Information, Skin Color Classification and Aggregated Channel Features". In: *Proceedings of the International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV'16: July 2016, USA)*. Juli 2016, S. 365–371. url: http://www.worldcomp-proceedings.com/proc/proc2016/IPCV16_Final_Edition/IPCV16_Papers.pdf (besucht am 07. 02. 2019) (siehe S. 127, 147).

[Ham16b] Hammer, J. H.; Voit, M. und Beyerer, J.: „Motion segmentation and appearance change detection based 2D hand tracking". In: *2016 19th International Conference on Information Fusion (FUSION)*. Juli 2016, S. 1743–1750 (siehe S. 98, 99, 117, 146).

[Han10] Hansen, D. W. und Ji, Q.: „In the Eye of the Beholder: A Survey of Models for Eyes and Gaze“. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32.3 (März 2010), S. 478–500. doi: 10.1109/TPAMI.2009.30 (siehe S. 27).

[Har05] Harris, Don und Muir, Helen C.: Contemporary Issues in Human Factors and Aviation Safety. Ashgate, Jan. 2005 (siehe S. 15).

[Har13] Harwood, Tracy; Jones, Martin und Carreras, Ashley: „Shedding light on retail environments“. In: *Proceedings of the 2013 Conference on Eye Tracking South Africa*. ETSA '13. Cape Town, South Africa: ACM, 2013, S. 2–7. doi: 10.1145/2509315.2509316 (siehe S. 37).

[Här15] Härter, Hendrik: Elektronik soll die Sicherheit im Operationssaal erhöhen. 2015. url: https://www.elektronikpraxis.vogel.de/elektronik-soll-die-sicherheit-im-operationssaal-erhoehen-a-487505/ (besucht am 14. 02. 2019) (siehe S. 6).

[Har17] Harvard Business Review Staff: „Augmented Reality in the Real World“. In: *Harvard Business Review* (Nov. 2017). url: https://hbr.org/2017/11/a-managers-guide-to-augmented-reality#augmented-reality-in-the-real-world (besucht am 11. 02. 2019) (siehe S. 4).

[He15] He, Kaiming; Zhang, Xiangyu; Ren, Shaoqing und Sun, Jian: „Deep Residual Learning for Image Recognition“. In: *arXiv:1512.03385 [cs]* (Dez. 2015). arXiv: 1512.03385. url: http://arxiv.org/abs/1512.03385 (besucht am 17. 01. 2017) (siehe S. 46, 52, 55).

[Hen12] Henry, Peter; Krainin, Michael; Herbst, Evan; Ren, Xiaofeng und Fox, Dieter: „RGB-D mapping: Using Kinect-style depth cameras for dense 3D modeling of indoor environments“. In: *The International Journal of Robotics Research* 31.5 (1. Apr. 2012), S. 647–663. doi: 10.1177/0278364911434148 (siehe S. 19).

[Hol18] Hololens: Gestures - Mixed Reality. 2018. url: https://docs.microsoft.com/en-us/windows/mixed-reality/gestures (besucht am 15. 02. 2019) (siehe S. 41).

[Iqb18] Iqbal, Umar; Molchanov, Pavlo; Breuel, Thomas; Gall, Juergen und Kautz, Jan: „Hand Pose Estimation via Latent 2.5D Heatmap Regression“. In: *Computer Vision – ECCV 2018*. Hrsg. von Ferrari, Vittorio; Hebert, Martial; Sminchisescu, Cristian und Weiss, Yair. Lecture Notes in Computer Science. Springer International Publishing, 2018, S. 125–143 (siehe S. 54, 55, 101, 154, 155).

[Isa98] Isard, Michael und Blake, Andrew: „Condensation - Conditional Density Propagation for Visual Tracking“. English. In: *International Journal of Computer Vision* 29 (1 1998), S. 5–28. doi: 10.1023/A:1008078328650 (siehe S. 111).

[Jac90] Jacob, Robert J. K.: „What you look at is what you get: eye movement-based interaction techniques“. In: *Proceedings of the SIGCHI conference on Human factors in computing systems: Empowering people*. CHI '90. Seattle, Washington, United States: ACM, 1990, S. 11–18. doi: 10.1145/97243.97246 (siehe S. 35).

[Jia14] Jia, Yangqing; Shelhamer, Evan; Donahue, Jeff; Karayev, Sergey; Long, Jonathan; Girshick, Ross; Guadarrama, Sergio und Darrell, Trevor: „Caffe: Convolutional Architecture for Fast Feature Embedding“. In: *arXiv:1408.5093 [cs]* (Juni 2014). arXiv: 1408.5093. url: http://arxiv.org/abs/1408.5093 (besucht am 15. 01. 2019) (siehe S. 50).

[Jin18] Jin, Bo und Yang, Feng: „An overview of SLAM“. In: *Global Intelligence Industry Conference (GIIC 2018)*. Global Intelligent Industry Conference 2018. Hrsg. von Lv, Yueguang. Beijing, China: SPIE, 31. Aug. 2018, S. 26. doi: 10.1117/12.2504048 (siehe S. 19).

[Jon99] Jones, Michael J. und Rehg, James M.: „Statistical Color Models with Application to Skin Detection“. In: *International Journal of Computer Vision*. 1999, S. 274–280 (siehe S. 104).

[Kak07] Kakumanu, P.; Makrogiannis, S. und Bourbakis, N.: „A survey of skin-color modeling and detection methods“. In: *Pattern Recognition* 40.3 (2007), S. 1106–1122. doi: 10.1016/j.patcog.2006.06.010 (siehe S. 48, 49, 104).

[Kan04] Kang, Hyun; Lee, Chang Woo und Jung, Keechul: „Recognition-based gesture spotting in video games“. In: *Pattern Recognition Letters* 25.15 (2004), S. 1701–1714. doi: 10.1016/j.patrec.2004.06.016 (siehe S. 48).

[Kan10] Kandemir, Melih; Saarinen, Veli-Matti und Kaski, Samuel: „Inferring object relevance from gaze in dynamic scenes“. In: *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*. ETRA '10. Austin, Texas: ACM, 2010, S. 105–108. doi: 10.1145/1743666.1743692 (siehe S. 33).

[Kas11] Kaspar, Kai und König, Peter: „Overt Attention and Context Factors: The Impact of Repeated Presentations, Image Type, and Individual Motivation“. In: *PLoS ONE* 6.7 (Juli 2011), e21719 (siehe S. 32).

[Kas14] Kassner, Moritz; Patera, William und Bulling, Andreas: „Pupil: An Open Source Platform for Pervasive Eye Tracking and Mobile Gaze-based Interaction“. In: *Adjunct Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. UbiComp '14 Adjunct. Seattle, Washington: ACM, 2014, S. 1151–1160. doi: 10.1145/2638728.2641695 (siehe S. 26).

[Kla08] Klami, Arto; Saunders, Craig; Campos, Teófilo E. de und Kaski, Samuel: „Can relevance of images be inferred from eye movements?“ In: *Proceedings of the 1st ACM international conference on Multimedia information retrieval*. MIR '08. Vancouver, British Columbia, Canada: ACM, 2008, S. 134–140. doi: 10.1145/1460096.1460120 (siehe S. 33).

[Köl04] Kölsch, M. und Turk, M.: „Fast 2D Hand Tracking with Flocks of Features and Multi-Cue Integration“. In: *Computer Vision and*

Pattern Recognition Workshop, 2004. CVPRW '04. Conference on. Juni 2004, S. 158. DOI: 10.1109/CVPR.2004.71 (siehe S. 48).

[Kom10] KOMOGORTSEV, Oleg V.; JAYARATHNA, Sampath; KOH, Do Hyong und GOWDA, Sandeep Munikrishne: „Qualitative and quantitative scoring and evaluation of the eye movement classification algorithms". In: *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications - ETRA '10*. Austin, Texas: ACM Press, 2010, S. 65. DOI: 10.1145/1743666.1743682 (siehe S. 28, 66).

[Kon15] KONSENSOP-KONSORTIUM: KonsensOP | Kontextsensitive Assistenz im aufmerksamen OP. de-DE. 2015. URL: http://konsensop.de/ (besucht am 14. 02. 2019) (siehe S. 6, 26, 40).

[Kre13] KRESS, Bernard und SHIN, Meimei: „Diffractive and holographic optics as optical combiners in head mounted displays". In: *Proceedings of the 2013 ACM conference on Pervasive and ubiquitous computing adjunct publication - UbiComp '13 Adjunct*. Zurich, Switzerland: ACM Press, 2013, S. 1479–1482. DOI: 10.1145/2494091.2499572 (siehe S. 15).

[Kre17] KRESS, Bernard C. und CUMMINGS, William J.: „Optical architecture of HoloLens mixed reality headset". In: *Digital Optical Technologies 2017*. Bd. 10335. International Society for Optics und Photonics, Juni 2017, 103350K. DOI: 10.1117/12.2270017 (siehe S. 15, 20, 21, 35, 41).

[Kur02] KURIHARA, K.; HOSHINO, S.; YAMANE, K. und NAKAMURA, Y.: „Optical motion capture system with pan-tilt camera tracking and real time data processing". In: *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)*. Bd. 2. Mai 2002, 1241–1248 vol.2. DOI: 10.1109/ROBOT.2002.1014713 (siehe S. 18).

[Lar08] Larsson, Jörgen und Blomqvist, Tommy: „The Cobra helmet mounted display system for Gripen“. In: *Head- and Helmet-Mounted Displays XIII: Design and Applications*. Bd. 6955. International Society for Optics und Photonics, 3. Apr. 2008, S. 695505. doi: 10.1117/12.778442 (siehe S. 17).

[Len18] Lengenfelder, Christian; Horne, Matthias; Hammer, Jan Hendrik; Voit, Michael und Beyerer, Jürgen: „Low-cost and Retrofittable Pose Estimation of Rigid Objects Using Infrared Markers“. In: *Procedia CIRP*. 51st CIRP Conference on Manufacturing Systems 72 (Jan. 2018), S. 839–844. doi: 10.1016/j.procir.2018.04.024 (siehe S. 18).

[Les16] Leswing, Kif: Apple is working on smart glasses. 2016. url: https://www.businessinsider.de/apple-is-working-on-smart-glasses-2016-11 (besucht am 26. 07. 2018) (siehe S. 2).

[Li13a] Li, Cheng und Kitani, Kris M.: „Pixel-level hand detection in ego-centric videos“. In: *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*. IEEE, 2013, S. 3570–3577 (siehe S. 49, 50, 55, 59, 105, 109, 117, 121, 146, 147).

[Li13b] Li, Hua; Zhang, Xin; Shi, Guangwei; Qu, Hemeng; Wu, Yanxiong und Zhang, Jianping: „Review and analysis of avionic helmet-mounted displays“. In: *Optical Engineering* 52.11 (2013), S. 110901–110901. doi: 10.1117/1.OE.52.11.110901 (siehe S. 14, 15).

[Loc08] Locher, P. und Nodine, C.: „What Does Visual Exploration of an Artwork Contribute to a Viewer's Immediate Aesthetic Reaction to It?“ In: *20th Congress of the International Association of Empirical Aesthetics*. Chicago, Illinois, USA, August 19-22, 2008 (siehe S. 32).

[Low04] Lowe, David G.: „Distinctive image features from scale-invariant keypoints“. In: *International journal of computer vision* 60.2 (2004), S. 91–110. url: http://link.springer.com/article/10.1023/B:VISI.0000029664.99615.94 (besucht am 21. 05. 2014) (siehe S. 49).

[Mag18] MagicLeapOne: Gesture - Learn | Magic Leap. 2018. URL: https://creator.magicleap.com/learn/guides/design-gesture (besucht am 15. 02. 2019) (siehe S. 41).

[Mai12] Maier, Sebastian und Hammer, Jan Hendrik: „DigET Digitaler Engineering Tisch - ein interaktives Assistenzsystem für Multi-User-Engineering“. In: *Fortschrittliche Anzeigesysteme für die Fahrzeug- und Prozessführung*. Koblenz, Deutschland, Okt. 2012 (siehe S. 41, 48).

[Man07] Manning, Sharon D. und Rash, Clarence E.: „A moveable view“. In: *AeroSafety world* 2.8 (Aug. 2007). URL: http://trid.trb.org/view.aspx?id=838241 (besucht am 18. 09. 2014) (siehe S. 15).

[Mau14] Maurus, M.; Hammer, J. H. und Beyerer, J.: „Realistic heatmap visualization for interactive analysis of 3D gaze data“. Englisch. In: *Conference on Eye Tracking Research and Applications (ETRA'14), Safety Harbor, Florida/USA, March 26-28, 2014*. ACM, New York (NY), 2014, S. 295–298. DOI: 10.1145/2578153.2578204 (siehe S. 61, 94).

[Mei14] Meier, Peter und Angermann, Frank: „Method for representing virtual information in a real environment“. U.S. Patent US8896629B2. Nov. 2014. URL: https://patents.google.com/patent/US8896629/en (besucht am 26. 07. 2018) (siehe S. 2).

[Mer14] Merced, David Gelles and Michael J. de la: Google Invests Heavily in Magic Leap's Effort to Blend Illusion and Reality. 2014. URL: https://dealbook.nytimes.com/2014/10/21/google-invests-in-magic-leap-an-augmented-reality-firm/ (besucht am 26. 07. 2018) (siehe S. 2).

[Mil95] Milgram, Paul; Takemura, Haruo; Utsumi, Akira und Kishino, Fumio: „Augmented reality: a class of displays on the reality-virtuality continuum“. In: *Telemanipulator and Telepresence Technologies*. Hrsg. von Das, Hari. Bd. 2351. International Society for Optics und Photonics. SPIE, 1995, S. 282–292. URL: https://doi.org/10.1117/12.197321 (siehe S. 19–22).

[Mis09] Mistry, Pranav und Maes, Pattie: „SixthSense: a wearable gestural interface“. In: *ACM SIGGRAPH ASIA 2009 Sketches*. SIGGRAPH ASIA '09. Yokohama, Japan: ACM, 2009, 11:1–11:1. doi: 10.1145/1667146.1667160 (siehe S. 48).

[Moo17] Moon, Gyeongsik; Chang, Ju Yong und Lee, Kyoung Mu: „V2V-PoseNet: Voxel-to-Voxel Prediction Network for Accurate 3D Hand and Human Pose Estimation from a Single Depth Map“. In: *arXiv:1711.07399 [cs]* (Nov. 2017). arXiv: 1711.07399. url: http://arxiv.org/abs/1711.07399 (besucht am 10. 01. 2019) (siehe S. 47, 55).

[Mue17] Mueller, Franziska; Mehta, Dushyant; Sotnychenko, Oleksandr; Sridhar, Srinath; Casas, Dan und Theobalt, Christian: „Real-Time Hand Tracking Under Occlusion from an Egocentric RGB-D Sensor“. In: *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*. Venice, Italy: IEEE, Okt. 2017. doi: 10.1109/iccvw.2017.82 (siehe S. 46, 53, 55, 58, 100, 101).

[Mue18] Mueller, Franziska; Bernard, Florian; Sotnychenko, Oleksandr; Mehta, Dushyant; Sridhar, Srinath; Casas, Dan und Theobalt, Christian: „GANerated Hands for Real-Time 3D Hand Tracking from Monocular RGB“. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. Salt Lake City, UT, USA: IEEE, Juni 2018. doi: 10.1109/cvpr.2018.00013 (siehe S. 53–55, 101, 154).

[Muk09] Mukawa, Hiroshi; Akutsu, Katsuyuki; Matsumura, Ikuo; Nakano, Satoshi; Yoshida, Takuji; Kuwahara, Mieko und Aiki, Kazuma: „A full-color eyewear display using planar waveguides with reflection volume holograms“. In: *Journal of the Society for Information Display* 17.3 (März 2009), S. 185–193. doi: 10.1889/JSID17.3.185 (siehe S. 15).

[Mur15] Mur-Artal, R.; Montiel, J. M. M. und Tardós, J. D.: „ORB-SLAM: A Versatile and Accurate Monocular SLAM System“. In: *IEEE Transactions on Robotics* 31.5 (Okt. 2015), S. 1147–1163. doi: 10.1109/TRO.2015.2463671 (siehe S. 19).

[Mur17] Mur-Artal, R. und Tardós, J. D.: „ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras“. In: *IEEE Transactions on Robotics* 33.5 (Okt. 2017), S. 1255–1262. doi: 10.1109/TRO.2017.2705103 (siehe S. 19).

[Nak16] Nakano, Gaku: „A Versatile Approach for Solving PnP, PnPf, and PnPfr Problems“. In: *Computer Vision – ECCV 2016*. Springer, Cham, Okt. 2016, S. 338–352. doi: 10.1007/978-3-319-46487-9_21 (siehe S. 64).

[Nod93] Nodine, C F; Locher, P J und Krupinski, E A: „The Role of Formal Art Training on Perception and Aesthetic Judgment of Art Compositions“. In: *Leonardo* 26.3 (1993), S. 219 (siehe S. 32).

[Obe17] Oberweger, Markus und Lepetit, Vincent: „DeepPrior++: Improving Fast and Accurate 3D Hand Pose Estimation“. In: *arXiv:1708.08325 [cs]* (Aug. 2017). arXiv: 1708.08325. url: http://arxiv.org/abs/1708.08325 (besucht am 16. 01. 2019) (siehe S. 55).

[Oik11] Oikonomidis, Iason; Kyriazis, Nikolaos und Argyros, Antonis: „Efficient model-based 3D tracking of hand articulations using Kinect“. In: *Procedings of the British Machine Vision Conference 2011*. Dundee: British Machine Vision Association, 2011. doi: 10.5244/c.25.101 (siehe S. 43–45, 54).

[Ols13] Olsson, Maj Isabelle; Heinrich, Mitchell Joseph; Kelly, Daniel und Lapetina, John: „Wearable device with input and output structures“. U.S. Patent US20130044042A1. Feb. 2013. url: https://patents.google.com/patent/US20130044042A1/en?oq=20130044042 (besucht am 10. 09. 2018) (siehe S. 14, 20).

[Ots79a] Otsu, N.: „A threshold selection method from gray level histograms“. In: *IEEE Trans. Systems, Man and Cybernetics* 9 (März 1979), S. 62–66 (siehe S. 122).

[Ots79b] Otsu, N.: „A Threshold Selection Method from Gray-Level Histograms“. In: *IEEE Transactions on Systems, Man, and Cybernetics* 9.1 (Jan. 1979), S. 62–66. doi: 10.1109/TSMC.1979.4310076 (siehe S. 50).

[Pan18] Panteleris, Paschalis; Oikonomidis, Iason und Argyros, Antonis: „Using a Single RGB Frame for Real Time 3D Hand Pose Estimation in the Wild“. In: *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*. Lake Tahoe, NV: IEEE, März 2018. doi: 10.1109/wacv.2018.00054 (siehe S. 52, 54, 55).

[Pfe12a] Pfeiffer, Thies: „3D Attention Volumes for usability studies in virtual reality“. In: *Proceedings of the 2012 IEEE Virtual Reality*. VR '12. Washington, DC, USA: IEEE Computer Society, 2012, S. 117–118. doi: 10.1109/VR.2012.6180910 (siehe S. 38).

[Pfe12b] Pfeiffer, Thies: „Measuring and visualizing attention in space with 3D attention volumes“. In: *Proceedings of the Symposium on Eye Tracking Research and Applications*. ETRA '12. Santa Barbara, California: ACM, 2012, S. 29–36. doi: 10.1145/2168556.2168560 (siehe S. 28).

[Pfe16] Pfeiffer, Thies und Memili, Cem: „Model-based real-time visualization of realistic three-dimensional heat maps for mobile eye tracking and eye tracking in virtual reality“. In: *Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications - ETRA '16*. Charleston, South Carolina: ACM Press, 2016, S. 95–102. doi: 10.1145/2857491.2857541 (siehe S. 36, 38, 94).

[Phu05] Phung, S.L.; Bouzerdoum A., Sr. und Chai D., Sr.: „Skin segmentation using color pixel classification: analysis and comparison“. In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 27.1 (Jan. 2005), S. 148–154. doi: 10.1109/TPAMI.2005.17 (siehe S. 48, 49, 104).

[Pir12] Pirsiavash, H. und Ramanan, D.: „Detecting activities of daily living in first-person camera views“. In: *2012 IEEE Conference on Computer Vision and Pattern Recognition*. Juni 2012, S. 2847–2854. doi: 10.1109/CVPR.2012.6248010 (siehe S. 59).

[Por17a] Porter, Michael E. und Heppelmann, James E.: „How Does Augmented Reality Work?“ In: *Harvard Business Review* (Nov. 2017). url: https://hbr.org/2017/11/a-managers-guide-

to-augmented-reality#how-does-augmented-reality-work (besucht am 11. 02. 2019) (siehe S. 5).

[Por17b] Porter, Michael E. und Heppelmann, James E.: „The Battle of the Smart Glasses". In: *Harvard Business Review* (Nov. 2017). url: https://hbr.org/2017/11/a-managers-guide-to-augmented-reality#the-battle-of-the-smart-glasses (besucht am 11. 02. 2019) (siehe S. 2).

[Por17c] Porter, Michael E. und Heppelmann, James E.: „Why Every Organization Needs an Augmented Reality Strategy". In: *Harvard Business Review* (Nov. 2017). url: https://hbr.org/2017/11/a-managers-guide-to-augmented-reality#why-every-organization-needs-an-augmented-reality-strategy (besucht am 11. 02. 2019) (siehe S. 5).

[Pri04] Prinzel, Lawrence J.: „Head-Up Displays and Attention Capture". In: 2004 (siehe S. 10, 11).

[Ran17] Ranadive, Shubhendu; Harsora, Jay; Khanvilkar, Ashmi und Sayyad, Mohasin: „A Systematic Literature Review on Virtual Reality - The Oculus Rift". In: *International Journal of Research In Science* 7 (2017), S. 9 (siehe S. 18, 21).

[Ras09a] Rash, Clarence E.; Russo, Michael B.; Letowski, Tomasz R. und Schmeisser, Elmar T.: Helmet-Mounted Displays: Sensation, Perception and Cognition Issues. U.S. Army Aeromedical Research Laboratory, 2009 (siehe S. 2, 9–12, 15).

[Ras09b] Rashid, O.; Al-Hamadi, A. und Michaelis, B.: „A framework for the integration of gesture and posture recognition using HMM and SVM". In: *IEEE International Conference on Intelligent Computing and Intelligent Systems, 2009. ICIS 2009.* Bd. 4. Nov. 2009, S. 572–577. doi: 10.1109/ICICISYS.2009.5357615 (siehe S. 40).

[Ray98] Rayner, K.: „Eye movements in reading and information processing: 20 years of research." In: *Psychological Bulletin* 124.3 (1998), S. 372–422 (siehe S. 23).

[Red17] Redmon, Joseph und Farhadi, Ali: „YOLO9000: Better, Faster, Stronger“. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Honolulu, HI: IEEE, Juli 2017, S. 6517–6525. doi: 10.1109/CVPR.2017.690 (siehe S. 52, 54, 55).

[Rei17] Reichert, Daniel: „Transparentes Fahrerhaus - Visualisierung von echtzeit Stereorekonstruktionsdaten in erweiterter und virtueller Realität“. Masterarbeit. KIT, Karlsruhe, 31. Okt. 2017 (siehe S. 14, 20).

[Ren03] Ren und Malik: „Learning a classification model for segmentation“. In: *Proceedings Ninth IEEE International Conference on Computer Vision*. Nice, France: IEEE, 2003, 10–17 vol.1. doi: 10.1109/ICCV.2003.1238308 (siehe S. 49).

[Ren09] Ren, X. und Philipose, M.: „Egocentric recognition of handled objects: Benchmark and analysis“. In: *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. Juni 2009, S. 1–8. doi: 10.1109/CVPRW.2009.5204360 (siehe S. 48, 58).

[Ren10] Ren, Xiaofeng und Gu, Chunhui: „Figure-ground segmentation improves handled object recognition in egocentric video“. In: *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. San Francisco, CA, USA: IEEE, Juni 2010, S. 3137–3144. doi: 10.1109/CVPR.2010.5540074 (siehe S. 48, 49, 55, 59).

[Ren15] Ren, Shaoqing; He, Kaiming; Girshick, Ross und Sun, Jian: „Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks“. In: *arXiv:1506.01497 [cs]* (Juni 2015). arXiv: 1506.01497. url: http://arxiv.org/abs/1506.01497 (besucht am 16. 01. 2019) (siehe S. 52, 53, 55).

[Rot04] Rother, Carsten; Kolmogorov, Vladimir und Blake, Andrew: „”GrabCut”: Interactive Foreground Extraction Using Iterated Graph Cuts“. In: *ACM SIGGRAPH 2004 Papers*. SIGGRAPH ’04. New York, NY, USA: ACM, 2004, S. 309–314. doi: 10.1145/1186562.1015720 (siehe S. 51).

[Sal00] Salvucci, Dario D. und Goldberg, Joseph H.: „Identifying fixations and saccades in eye-tracking protocols“. In: *ETRA '00: Proceedings of the 2000 symposium on Eye tracking research & applications*. Palm Beach Gardens, Florida, United States: ACM, 2000, S. 71–78 (siehe S. 28, 66).

[Šar11] Šarić, Marin: LibHand: A Library for Hand Articulation. Version 0.9. 2011. url: http://www.libhand.org/ (besucht am 10. 02. 2019) (siehe S. 44).

[Sch13] Schleipen, Miriam; Schenk, Manfred; Maier, Sebastian; Hammer, Jan-Hendrik und Peinsipp-Byma, Elisabeth: „AutomationML als Basis einer interaktiven Umgebung für das multi-User-Engineering. Digitaler Engineering-Tisch (DigET) kombiniert OPC-UA und AutomationML“. In: *SPS-Magazin* 6 (2013), S. 34–37 (siehe S. 48).

[Sec18] Secker, Daniel: „Transparentes Cockpit - Online-Visualisierung der Fahrzeugumwelt für die Assistenz bei Rangier-Manövern“. Masterarbeit. KIT, Karlsruhe, 31. Aug. 2018 (siehe S. 14, 20).

[Sha15] Sharp, Toby u. a.: „Accurate, Robust, and Flexible Real-time Hand Tracking“. In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems - CHI '15*. Seoul, Republic of Korea: ACM Press, 2015. doi: 10.1145/2702123.2702179 (siehe S. 45, 54, 56).

[Sho11] Shotton, J.; Fitzgibbon, A.; Cook, M.; Sharp, T.; Finocchio, M.; Moore, R.; Kipman, A. und Blake, A.: „Real-time human pose recognition in parts from single depth images“. In: *CVPR 2011*. Juni 2011, S. 1297–1304. doi: 10.1109/CVPR.2011.5995316 (siehe S. 43–45, 54).

[Sim05] Simons, Daniel J. und Rensink, Ronald A.: Change blindness: Past, present, and future. 2005 (siehe S. 24).

[Sim14] Simonyan, Karen und Zisserman, Andrew: „Very Deep Convolutional Networks for Large-Scale Image Recognition“. In: *arXiv:1409.1556 [cs]* (Sep. 2014). arXiv: 1409.1556. url: http://arxiv.org/abs/1409.1556 (besucht am 17. 02. 2019) (siehe S. 52).

[Sim17] Simon, Tomas; Joo, Hanbyul; Matthews, Iain und Sheikh, Yaser: „Hand Keypoint Detection in Single Images using Multiview Bootstrapping“. In: *CVPR*. 2017 (siehe S. 51, 52, 136).

[Spr10] Spruyt, V.; Ledda, A. und Geerts, S.: „Real-time multicolourspace hand segmentation“. In: *Image Processing (ICIP), 2010 17th IEEE International Conference on*. Sep. 2010, S. 3117–3120. doi: 10.1109/ICIP.2010.5653220 (siehe S. 48).

[Ste10a] Stellmach, Sophie; Nacke, Lennart und Dachselt, Raimund: „3D attentional maps: aggregated gaze visualizations in three-dimensional virtual environments“. In: *Proceedings of the International Conference on Advanced Visual Interfaces - AVI '10*. Roma, Italy: ACM Press, 2010, S. 345. doi: 10.1145/1842993.1843058 (siehe S. 37).

[Ste10b] Stellmach, Sophie; Nacke, Lennart und Dachselt, Raimund: „Advanced gaze visualizations for three-dimensional virtual environments“. In: *Proceedings of the 2010 Symposium on Eye-Tracking Research &Applications*. ETRA '10. Austin, Texas: ACM, 2010, S. 109–112. doi: 10.1145/1743666.1743693 (siehe S. 37, 93).

[Sun15] Sun, Yuechuan; Wu, Sijing und Spence, Ian: „The Commingled Division of Visual Attention“. In: *PLOS ONE* 10.6 (Juni 2015), e0130611. doi: 10.1371/journal.pone.0130611 (siehe S. 10).

[Sur19] Surgicaltheater: Medical Virtual Reality | Virtual Surgical Planning | Surgical Theater. 2019. url: https://www.surgicaltheater.net/ (besucht am 11. 02. 2019) (siehe S. 6).

[Tan14] Tang, Danhang; Chang, Hyung Jin; Tejani, Alykhan und Kim, Tae-Kyun: „Latent Regression Forest: Structured Estimation of 3D Articulated Hand Posture“. In: *2014 IEEE Conference on Computer Vision and Pattern Recognition*. Columbus, OH, USA: IEEE, Juni 2014, S. 3786–3793. doi: 10.1109/CVPR.2014.490 (siehe S. 55).

[Tan15] Tang, Danhang; Taylor, Jonathan; Kohli, Pushmeet; Keskin, Cem; Kim, Tae-Kyun und Shotton, Jamie: „Opening the Black Box: Hierarchical Sampling Optimization for Estimating Human Hand Pose". In: *2015 IEEE International Conference on Computer Vision (ICCV)*. Santiago: IEEE, Dez. 2015. doi: 10.1109/iccv.2015.380 (siehe S. 45, 46, 55).

[Tan17] Tang, Danhang; Chang, Hyung Jin; Tejani, Alykhan und Kim, Tae-Kyun: „Latent Regression Forest: Structured Estimation of 3D Hand Poses". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.7 (Juli 2017), S. 1374–1387. doi: 10.1109/tpami.2016.2599170 (siehe S. 55).

[The13] Theis, Sabine; Alexander, Thomas und Wille, Matthias: „Voruntersuchung zur Bewertung des sicheren und beanspruchungsoptimalen Einsatzes von Head-Mounted Displays". In: *Zeitschrift für Arbeitswissenschaft* 67.3 (Sep. 2013), S. 159–167. doi: 10.1007/BF03374403 (siehe S. 11).

[Tob10] Tobii Eye Tracking: An introduction to eye tracking and Tobii Eye Trackers. Jan. 2010 (siehe S. 23).

[Tom14] Tompson, Jonathan; Stein, Murphy; Lecun, Yann und Perlin, Ken: „Real-Time Continuous Pose Recovery of Human Hands Using Convolutional Networks". In: *ACM Transactions on Graphics* 33.5 (Sep. 2014), 169:1–169:10. doi: 10.1145/2629500 (siehe S. 44, 45, 51, 52, 54, 56).

[Toy12] Toyama, Takumi; Kieninger, Thomas; Shafait, Faisal und Dengel, Andreas: „Gaze guided object recognition using a head-mounted eye tracker". In: *Proceedings of the Symposium on Eye Tracking Research and Applications*. ETRA '12. Santa Barbara, California: ACM, 2012, S. 91–98. doi: 10.1145/2168556.2168570 (siehe S. 29).

[Tse99] Tseng, Paul: „Fortified-Descent Simplicial Search Method: A General Approach". In: *SIAM J. on Optimization* 10.1 (Mai 1999), S. 269–288. doi: 10.1137/S1052623495282857 (siehe S. 44).

[Wac11] Wachs, Juan Pablo; Kölsch, Mathias; Stern, Helman und Edan, Yael: „Vision-based hand-gesture applications“. In: *Commun. ACM* 54 (2 Feb. 2011), S. 60–71. doi: 10.1145/1897816.1897838 (siehe S. 48, 55).

[Wan09] Wang, Robert Y. und Popović, Jovan: „Real-time hand-tracking with a color glove“. In: *ACM Trans. Graph.* 28.3 (Juli 2009), 63:1–63:8. doi: 10.1145/1531326.1531369 (siehe S. 48).

[Wei16] Wei, Shih-En; Ramakrishna, Varun; Kanade, Takeo und Sheikh, Yaser: „Convolutional pose machines“. In: *CVPR*. 2016 (siehe S. 51, 52).

[WHI17] WHITEHOUSE, William; Foxlin, Eric; CALLOWAY, Thomas und Popoolapade, John: „Dual-mode illuminator for imaging under different lighting conditions“. US-Pat. U.S. Patent US20170026560A1. Inc, Thales Visionix. 26. Jan. 2017. url: https://patents.google.com/patent/US20170026560A1/en (besucht am 26. 09. 2018) (siehe S. 2).

[Wil78] Williams, Lance: „Casting curved shadows on curved surfaces“. In: *Proceedings of the 5th annual conference on Computer graphics and interactive techniques*. SIGGRAPH '78. New York, NY, USA: ACM, 1978, S. 270–274. doi: 10.1145/800248.807402 (siehe S. 89).

[Wu18] Wu, Xiaokun; Finnegan, Daniel; O'Neill, Eamonn und Yang, Yong-Liang: „HandMap: Robust Hand Pose Estimation via Intermediate Dense Guidance Map Supervision“. In: *Computer Vision – ECCV 2018*. Hrsg. von Ferrari, Vittorio; Hebert, Martial; Sminchisescu, Cristian und Weiss, Yair. Lecture Notes in Computer Science. Springer International Publishing, 2018, S. 246–262 (siehe S. 55).

[Yas10] Yasuda, T.; Matsumura, Y. und Ohkura, K.: „Extended pso with partial randomization for large scale multimodal problems“. In: *2010 World Automation Congress*. Sep. 2010, S. 1–6 (siehe S. 44, 45).

[Ye16] Ye, Qi; Yuan, Shanxin und Kim, Tae-Kyun: „Spatial Attention Deep Net with Partial PSO for Hierarchical Hybrid Hand Pose Estimation“. In: *Computer Vision – ECCV 2016*. Hrsg. von Leibe, Bastian; Matas, Jiri; Sebe, Nicu und Welling, Max. Lecture Notes in Computer Science. Springer International Publishing, 2016, S. 346–361 (siehe S. 46, 55, 56).

[Yeo17] Yeoh, Ivan L.; Edwin, Lionel E. und Macnamara, John Graham: „Wavelength multiplexing in waveguides“. U.S. Patent US20170329075A1. Nov. 2017. url: https://patents.google.com/patent/US20170329075A1 (besucht am 19. 09. 2018) (siehe S. 15, 35, 41).

[Yua17a] Yuan, Shanxin; Ye, Qi; Garcia-Hernando, Guillermo und Kim, Tae-Kyun: „The 2017 Hands in the Million Challenge on 3D Hand Pose Estimation“. In: *arXiv:1707.02237v1* (2017), S. 7 (siehe S. 46, 55, 57).

[Yua17b] Yuan, Shanxin; Ye, Qi; Stenger, Bjorn; Jain, Siddhant und Kim, Tae-Kyun: „BigHand2.2M Benchmark: Hand Pose Dataset and State of the Art Analysis“. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Honolulu, HI: IEEE, Juli 2017. doi: 10.1109/cvpr.2017.279 (siehe S. 56, 57).

[Yua18] Yuan, Shanxin u. a.: „Depth-Based 3D Hand Pose Estimation: From Current Achievements to Future Goals“. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. Salt Lake City, UT, USA: IEEE, Juni 2018. doi: 10.1109/cvpr.2018.00279 (siehe S. 46).

[Zac07] Zach, C.; Pock, T. und Bischof, H.: „A duality based approach for realtime TV-L1 optical flow“. In: *Proceedings of the 29th DAGM conference on Pattern recognition*. Heidelberg, Germany: Springer-Verlag, 2007, S. 214–223. url: http://dl.acm.org/citation.cfm?id=1771530.1771554 (besucht am 03. 11. 2013) (siehe S. 117).

[Zan10] Zander, Thorsten O.; Gaertner, Matti; Kothe, Christian und Vilimek, Roman: „Combining Eye Gaze Input With a Brain-Computer Interface for Touchless Human-Computer Interaction". In: *International Journal of Human-Computer Interaction* 27.1 (2010), S. 38–51 (siehe S. 35).

[Zha12] Zhang, Zhengyou: „Microsoft Kinect Sensor and Its Effect". In: *IEEE Multimedia* 19.2 (Feb. 2012), S. 4–10. doi: 10.1109/MMUL.2012.24 (siehe S. 43).

[Zha16] Zhang, Jiawei; Jiao, Jianbo; Chen, Mingliang; Qu, Liangqiong; Xu, Xiaobin und Yang, Qingxiong: „3D Hand Pose Tracking and Estimation Using Stereo Matching". In: *arXiv:1610.07214 [cs]* (Okt. 2016). arXiv: 1610.07214. url: http://arxiv.org/abs/1610.07214 (besucht am 17. 01. 2019) (siehe S. 53, 54, 56).

[Zho18] Zhou, Yidan; Lu, Jian; Du, Kuo; Lin, Xiangbo; Sun, Yi und Ma, Xiaohong: „HBE: Hand Branch Ensemble Network for Real-Time 3D Hand Pose Estimation". In: *Computer Vision – ECCV 2018*. Hrsg. von Ferrari, Vittorio; Hebert, Martial; Sminchisescu, Cristian und Weiss, Yair. Lecture Notes in Computer Science. Springer International Publishing, 2018, S. 521–536 (siehe S. 47, 55).

[Zie15] Ziegler, Peter-Michael: c't. 2015. url: https://www.heise.de/ct/ausgabe/2015-12-aktuell-Forschung-2641661.html (besucht am 14. 02. 2019) (siehe S. 6).

[Zim17] Zimmermann, Christian und Brox, Thomas: „Learning to Estimate 3D Hand Pose from Single RGB Images". In: *IEEE International Conference on Computer Vision (ICCV)*. 2017. url: https://arxiv.org/abs/1705.01389 (besucht am 25. 07. 2018) (siehe S. 52–55, 57, 101, 108, 146, 149, 154).

[Zuc54] Zuckerman, Joshua: Perimetry. Lippincott, 1954 (siehe S. 13, 23).

Eigene Publikationen

[1] Hammer, Jan Hendrik: „Entwicklung eines echtzeitfähigen 2-D-Bewegungsschätzers zur Objektdetektion“. Diplomarbeit. KIT, Karlsruhe, 2011.

[2] Maier, Sebastian und Hammer, Jan Hendrik: „DigET Digitaler Engineering Tisch - ein interaktives Assistenzsystem für Multi-User-Engineering“. In: *Fortschrittliche Anzeigesysteme für die Fahrzeug- und Prozessführung*. Koblenz, Deutschland, Okt. 2012.

[3] Hammer, Jan Hendrik; Maurus, Michael und Beyerer, Jürgen: „Real-time 3D gaze analysis in mobile applications“. In: *Proceedings of the 2013 Conference on Eye Tracking South Africa*. ETSA ’13. Cape Town, South Africa: ACM, 2013, S. 75–78. doi: 10.1145/2509315.2509333.

[4] Hammer, JanHendrik und Beyerer, Jürgen: „Robust Hand Tracking in Realtime Using a Single Head-Mounted RGB Camera“. In: *Human-Computer Interaction. Interaction Modalities and Techniques*. Hrsg. von Kurosu, Masaaki. Bd. 8007. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2013, S. 252–261. doi: 10.1007/978-3-642-39330-3_27.

[5] Schleipen, Miriam; Schenk, Manfred; Maier, Sebastian; Hammer, Jan-Hendrik und Peinsipp-Byma, Elisabeth: „AutomationML als Basis einer interaktiven Umgebung für das multi-User-Engineering. Digitaler Engineering-Tisch (DigET) kombiniert OPC-UA und AutomationML“. In: *SPS-Magazin* 6 (2013), S. 34–37.

[6] Maurus, M.; Hammer, J. H. und Beyerer, J.: „Realistic heatmap visualization for interactive analysis of 3D gaze data“. Englisch. In: *Conference on Eye Tracking Research and Applications (ETRA'14), Safety Harbor, Florida/USA, March 26-28, 2014*. ACM, New York (NY), 2014, S. 295–298. doi: 10.1145/2578153.2578204.

[7] Hammer, J. H.; Qu, C.; Voit, M. und Beyerer, J.: „2D Hand Tracking with Motion Information, Skin Color Classification and Aggregated Channel Features“. In: *Proceedings of the International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV'16: July 2016, USA)*. Juli 2016, S. 365–371. url: http://www.worldcomp-proceedings.com/proc/proc2016/IPCV16_Final_Edition/IPCV16_Papers.pdf (besucht am 07. 02. 2019).

[8] Hammer, J. H.; Voit, M. und Beyerer, J.: „Motion segmentation and appearance change detection based 2D hand tracking“. In: *2016 19th International Conference on Information Fusion (FUSION)*. Juli 2016, S. 1743–1750.

[9] Hild, Jutta; Klaus, Edmund; Hammer, Jan-Hendrik; Martin, Manuel; Voit, Michael; Peinsipp-Byma, Elisabeth und Beyerer, Jürgen: „A Pilot Study on Gaze-Based Control of a Virtual Camera Using 360°-Video Data“. In: *Engineering Psychology and Cognitive Ergonomics*. Hrsg. von Harris, Don. Cham: Springer International Publishing, 2018, S. 419–428.

[10] Lengenfelder, Christian; Horne, Matthias; Hammer, Jan Hendrik; Voit, Michael und Beyerer, Jürgen: „Low-cost and Retrofittable Pose Estimation of Rigid Objects Using Infrared Markers“. In: *Procedia CIRP*. 51st CIRP Conference on Manufacturing Systems 72 (Jan. 2018), S. 839–844. doi: 10.1016/j.procir.2018.04.024.

Betreute studentische Arbeiten

[1] Graf, Leonard: „Entwicklung eines Hand-Tracking-Verfahrens für mobile Anwendungen“. Bachelorarbeit. KIT, Karlsruhe, 11. Mai 2012.

[2] Konrad, Ulrich: „Bestimmung von Fixationen aus 3D-Blickpunkten“. Bachelorarbeit. KIT, Karlsruhe, 30. Apr. 2012.

[3] Kurz, Gerhard: „Entwicklung einer Handgestenerkennung für mobile Anwendungen“. Diplomarbeit. KIT, Karlsruhe, 13. Jan. 2012.

[4] Lengenfelder, Christian: „Entwicklung eines Segmentierungsverfahrens von 2D-Verschiebungsvektorfeldern“. Bachelorarbeit. KIT, Karlsruhe, 31. Mai 2013.

[5] Maurus, Michael: „Echtzeit-Visualisierung von 3D-Blickpfaden für die Blickanalyse“. Diplomarbeit. KIT, Karlsruhe, 31. Aug. 2013.

[6] Bauer, Andreas: „Quadratic Pseudo-Boolean Optimization for Scene Analysis using CUDA“. Masterarbeit. KIT, Karlsruhe, 30. Juni 2014.

[7] Schmidt, Karin: „Blick-Registrierung in Multi-Display-Umgebungen mit einem mobilen Blickmessgerät“. Diplomarbeit. KIT, Karlsruhe, 30. Nov. 2014.

[8] Tiefert, Daniel: „Entwicklung und Evaluierung einer robusten Hand Erkennung mit egozentrischen Videos“. Masterarbeit. Hochschule Karlsruhe Technik und Wirtschaft. Karlsruhe, 1. Dez. 2014.

[9] Horne, Matthias: „Entwicklung eines Verfahrens zur Posenschätzung von Objekten basierend auf Infrarot-Punkt-Markern“. Masterarbeit. KIT, Karlsruhe, 31. Okt. 2016.

[10] Lengenfelder, Christian: „Entwicklung eines Verfahrens zur Positionsschätzung von Infrarot-Markern zur Posenschätzung von Objekten“. Masterarbeit. KIT, Karlsruhe, 31. Okt. 2016.

[11] Reichert, Daniel: „Transparentes Fahrerhaus - Visualisierung von echtzeit Stereorekonstruktionsdaten in erweiterter und virtueller Realität“. Masterarbeit. KIT, Karlsruhe, 31. Okt. 2017.

[12] Secker, Daniel: „Transparentes Cockpit - Online-Visualisierung der Fahrzeugumwelt für die Assistenz bei Rangier-Manövern“. Masterarbeit. KIT, Karlsruhe, 31. Aug. 2018.

[13] Garcia, Carlos: „Multi-modal activity recognition from the first person perspective using gaze and hand movements“. Bachelorarbeit. KIT, Karlsruhe, 31. Jan. 2019.

Abbildungsverzeichnis

Tabellenverzeichnis

Abkürzungsverzeichnis

2D-PCH	Anteil korrekt geschätzter Hände
2D-PCK	Anteil korrekt geschätzter Gelenkpositionen
ACC	Korrektklassifikationsrate
AfM	Aggregated Channel Features featuring MACS
AOI	Area of Interest
AR	Augmented Reality
AUC	Fläche unter der Kurve
CAD	Computer-Aided Design
CNN	Convolutional Neural Network
F1	F1-Maß
FPR	Falsch-Positiv-Rate
HMD	Head-Mounted-Display
HPT	HSN-Pose-Tracking
HPTwF	HPT with Feedback

HSN	HandSegNet
HUD	Head-Up-Display
IMU	inertiale Messeinheit
IOSB	Fraunhofer-Institut für Optronik, Systemtechnik und Bildauswertung
LED	Light-Emitting-Diode
MACS	Motion segmentation and Appearance Change detection based Skin color detection
MEMS	mikroelektronische mechanische Systeme
PREC	Genauigkeit
RDF	Random Decision Forest
SLAM	Simultaneous Localization and Mapping
TPR	Richtig-Positiv-Rate
VR	Virtual Reality