Westfälische Hochschule

University of Applied Sciences
Gelsenkirchen Bocholt Recklinghausen

Masterarbeit

Titel der Arbeit // Title of Thesis

Visuelle Helfer: Ein Augmented Reality Prototyp zur Unterstützung der visuellen Wahrnehmung für die Steuerung eines Roboterarms

---

Akademischer Abschlussgrad: Grad, Fachrichtung (Abkürzung) // Degree

Master of Science (M. Sc.)

Autorenname, Geburtsort // Name, Place of Birth

Franziska Rücker, Herford

Studiengang // Course of Study

Medieninformatik

Fachbereich // Department

Informatik und Kommunikation

Erstprüferin/Erstprüfer // First Examiner

Prof. Dr. Jens Gerken

Zweitprüferin/Zweitprüfer // Second Examiner

Prof. Dr. Gregor Lux

Abgabedatum // Date of Submission

14.05.2020

Eidesstattliche Versicherung

Rücker, Franziska

Name, Vorname // Name, First Name

Ich versichere hiermit an Eides statt, dass ich die vorliegende Abschlussarbeit mit dem Titel

Visuelle Helfer: Ein Augmented Reality Prototyp zur Unterstützung der visuellen Wahrnehmung für die Steuerung eines Roboterarms

selbstständig und ohne unzulässige fremde Hilfe erbracht habe. Ich habe keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie wörtliche und sinngemäße Zitate kenntlich gemacht. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Ort, Datum, Unterschrift // Place, Date, Signature

II Kurzfassung

Körperliche Behinderungen können einen Menschen soweit einschränken, dass für sie ein autonomes und selbstbestimmtes Leben, trotz intakter mentaler und kognitiven Fähigkeiten, nicht mehr möglich ist. Daher ist für Menschen, die beispielsweise vom Hals abwärts gelähmt sind, sogenannten Tetraplegikern, jede Zurückgewinnung von Autonomie eine Steigerung der Lebensqualität. In dieser Masterarbeit wird ein Augmented Reality Prototyp entwickelt, der es Tetraplegikern oder Menschen mit einer ähnlichen körperlichen Einschränkung erlaubt, an einem Mensch-Roboter Arbeitsplatz Montageaufgaben durchzuführen und ihnen somit eine Integration ins Arbeitsleben ermöglichen kann. Der Prototyp erlaubt es den Benutzer ohne die Nutzung der Hände, einen Kuka iiwa Roboterarm mit der Microsoft HoloLens zu steuern. Dabei wird ein Fokus darauf gelegt, das Blickfeld des Benutzers mit speziellen virtuellen Visualisierungen, sogenannten visuellen Helfern, anzureichern, um die Nachteile, die durch die Bewegungseinschränkungen der Zielgruppe ausgelöst werden, auszugleichen. Diese visuellen Helfer sollen bei der Steuerung des Roboterarms unterstützen und die Bedienung des Prototyps verbessern. Eine Evaluation des Prototyps zeigte Tendenzen, dass das Konzept der visuellen Helfer den Benutzer den Roboterarm präziser steuern lässt und seine Bedienung unterstützt.

III Abstract

Physical disabilities can limit a person to such an extent that an autonomous and self-determined life is no longer possible for them, despite intact mental and cognitive abilities. Therefore, for people who are paralyzed from the neck down, so-called quadriplegics, every bit of autonomy is an increase in their quality of life. In this master thesis, an augmented reality prototype is developed that allows quadriplegics or people with similar physical disabilities to perform assembly tasks at a human-robot workstation and thus enable them to be integrated into working life. The prototype allows the user to control a Kuka iiwa robot arm with the Microsoft HoloLens without using their hands. A focus is placed on enriching the user's field of vision with special virtual visualizations, so-called visual cues, in order to compensate for the disadvantages that are triggered by the movement restrictions of the target group. These visual cues are intended to support the control of the robot arm and improve the operation of the prototype. An evaluation of the prototype showed tendencies that the concept of visual cues allows the user to control the robot arm more precisely and supports his operation.

IV Inhaltsverzeichnis

I Eidesstattliche Versicherung **II**

II Kurzfassung **III**

III Abstract **III**

IV Inhaltsverzeichnis **IV**

V Abbildungsverzeichnis **VII**

VI Abkürzungsverzeichnis **X**

1 Einführung **1**

1.1 Das MIA-Projekt 2

1.2 Aufbau der Arbeit 3

2 Mensch-Roboter-Interaktion **4**

2.1 Roboterarm 5

2.1.1 Bedienung von Roboterarmen 7

2.1.2 Zusammenarbeit von Mensch und Roboter 10

2.2 Kommunikation durch Virtual, Augmented und Mixed Reality 11

2.2.1 Das Reality-Virtuality-Kontinuum 12

2.2.2 Visualisierung über Displays und Projektionen 14

2.2.3 Visualisierung über Head-Mounted Display 18

2.3 Autonomie für Tetraplegiker 22

2.3.1 Eingabegeräte für Tetraplegiker 23

2.3.2 Tetraplegiker und Roboter 25

3 Konzeption des Prototyps **29**

3.1 Zentrale Anforderungen 29

3.2 Auswahl der Ein- und Ausgabetechnologien 32

3.2.1 Die Microsoft HoloLens 32

3.2.2 Eingabemethode über Sprache und Kopfbewegung 34

3.2.3 Der Kuka LBR iiwa 7R800 35

3.3 Anforderungen an die visuellen Helfer 37

3.4 Architekturbeschreibung 40

3.5 Vorgehen 41

4 Implementierung des Prototyps 42

4.1 Robotersteuerung 43
4.2 Darstellung der Hologramme 44
4.3 Interaktion 46
4.3.1 Interaktion über Buttons 48
4.3.2 Interaktion über Sprachbefehle 51
4.4 Visuelle Helfer 53
4.4.1 Laserpointer 54
4.4.2 Down-, Pick/Place- und Schließ-Indikator 54
4.4.3 Gripper-Bereich 57
4.4.4 Farbschema 58
4.4.5 Kraft-Feedback 61

5 Evaluation des Prototyps 62

5.1 Versionen der visuellen Helfer 62
5.1.1 No Cues 63
5.1.2 Advanced Cues 63
5.1.3 Vergleich der verschiedenen Versionen der Helfer 67
5.2 Hypothesen 69
5.3 Methodik 70
5.4 Probanden 72
5.5 Versuchsaufbau 73
5.5.1 Aufgabe 73
5.5.2 Versuchsablauf 76
5.6 Analyse 77
5.6.1 Analyse der Aufgabenzeit und Fehlerrate 78
5.6.2 Analyse der Präzision 79
5.6.3 Analyse der Anzahl der Befehle und Feinjustierzeit 81
5.6.4 Analyse des Nasa-TLX 83
5.6.5 Analyse des USE Fragebogens 84
5.6.6 Analyse der Interviews 84
5.7 Limitationen 85
5.8 Bewertung der Ergebnisse 86

6 Schluss 87

6.1 Ausblick 88

6.2 Fazit 89

A Literaturverzeichnis **90**

B Anhang **98**

B.1 Fragebogen zur Demografie und Hintergrundwissen 98

B.2 Nasa Task-Load-Index Fragebogen 99

B.3 USE Fragebogen: Nutzen, Zufriedenheit, und Bedienkomfort 100

B.4 Interviewfragen 101

B.5 Deskriptive Statistiken zu den Diagrammen 102

B.6 Inhalt des beiliegenden USB-Sticks 104

V Abbildungsverzeichnis

1 Roboterarm im Vergleich zum menschlichen Arm. 5
2 Vergleich zwischen einem (a) Schwerlast-Roboter und einem (b) kollaborativen Leichtbau-Roboter. 7
3 Das SmartPad vom Kuka iiwa. 8
4 Darstellung vom Reality-Virtuality-Kontinuum. 12
5 Monitor-basierende AR-Prototypen für Mensch-Roboter-Interaktion von (a) Fang, Ong und Nee und (b) Milgram, Zhai, Drascic und Grodski. 15
6 Screenshot vom Mobilgerät, das auf den Arbeitsbereich vom Roboter gerichtet ist und das Grid und die vom Roboter erreichbaren Flächen augmentiert (Frank, Moorhead & Kapila, 2017). 16
7 Projektions-basierte Prototypen für Mensch-Roboter-Interaktion von (a) Gao und Huang und (b) Ganesan, Rathore, Ross und Amor. 17
8 Datenbrille von Epson (a) und Ansicht aus der Datenbrille vom Augmented Reality System von Makris, Karagiannis, Koukas und Matthaiakis (b). 18
9 Verwendetes Virtual Reality Headset sowie Eingabegerät (a) und Ansicht vom Virtual Reality Headset (b) des Prototyps von Ong, Yew, Thanigaivel und Nee. 19
10 Das verwendete Augmented Reality Headset (a) und die Ansicht durch die HoloLens (b) des Prototyps von Rosen et al. 20
11 Der Allzweckroboter PR2, wie er gerade ein Handtuch aus einer Schublade holt (T. L. Chen et al., 2013). 26
12 Tragbare Steuerungsschnittstelle mit einem MARG Sensor und drei EMG Sensoren zur Steuerung eines Leichtbauroboterarms (Baldi, Spagnoletti, Dragusanu & Prattichizzo, 2017). 27
13 Ein beispielhafter Aufbau einer festen Arbeitsstation für die Montage eines Bett-Verstellbeschlags aus (a) frontaler und (a) seitlicher Ansicht. Die seitliche Ansicht macht deutlich, dass bei der frontalen Ansicht wichtige visuelle Informationen verloren gehen. 30
14 Die (a) Microsoft HoloLens, (b) Microsoft HoloLens 2 und (c) Magic Leap im Vergleich. 33

15 Der Kuka iiwa mit (a) Kennzeichnung und Benennung seiner sieben Achsen und (a) wie er eine Nullraumbewegung durchführt (KUKA AG, 2020). 35
16 Innen, außen und formschlüssiges Greifen vom Robotiq 2F-85 Gripper (Bouchard, 2014). 37
17 Schwierigkeiten bei der Positionierung des Grippers nach vorne /hinten (siehe (c) und (d)) sowie nach links/rechts (siehe (a) und (b)). 38
18 Architektur des Prototyps. 40
19 Roboter, Plattform auf der der Roboter befestigt ist und der Tisch bilden die Arbeitsstation. Am Tisch ist ein roter Marker befestigt, der die Position der Arbeitsstation markiert. Der Benutzer trägt die HoloLens und sitzt frontal vor der Arbeitsstation. Virtuelle Objekte werden am, sowie unter dem Gripper angezeigt. 42
20 Ursprung des Roboterkoodinatensytems und des HoloLens-Koordinatensytems. Der Abstand vom Marker zum Ursprung des Roboterkoodinatensytems ist in Gelb markiert. Rot, Grün, Blau entspricht der x-, y-, z-Achse. 45
21 Der Arbeitsbereich ist 59,5 cm x 29 cm x 23,2 cm (Länge x Breite x Höhe) groß. Nur innerhalb dieser Abmessungen kann sich der Roboter bewegen. 46
22 Die vier Buttons der Applikation mit gekennzeichneten Indikatoren. 48
23 Der ausgelöste Präzisionsmodus in (a) vertikaler und in (b) horizontaler Richtung. 50
24 Kennzeichnung der unterschiedlichen visuellen Helfer. 53
25 Die Animation des (b) Down-Indikators und des (a) Pick/Place-Indikators. 55
26 Der Schließ-Indikator im Vergleich zum (a) geöffneten und (b) geschlossenen Gripper. 56
27 Die *Version für die erste Studie* mit gekennzeichneten Gripper-Verlängerungen und der ersten Version der Farb-Indikatoren. . . . 57
28 Farbschema mit den gekennzeichneten virtuellen Repräsentationen. 58
29 Der Cursor kollidiert mit dem Objekt und wird an dessen Oberfläche angezeigt. 64
30 Laserpointer und Gripper-Bereich der Advanced Cues (a) ohne Verdeckung und (b) mit Verdeckung. 65
31 Kollision der Gripper-Verlängerung mit einem Objekt. 66

32 Hat der Gripper ein Objekt gegriffen, wird (a) ein dem Objekt entsprechendes Ghost-Objekt angezeigt. Zusätzlich werden (b) zukünftige Kollisionen des Grippers mit umliegenden Objekten dargestellt. 67
33 Gegenüberstellung der visuellen Helfer von Cues und Advanced Cues abhängig von der hauptsächlichen Hilfsfunktion. 68
34 Hintergrundwissen der Probanden im Bezug auf Erfahrungen mit der HoloLens und Erfahrungen mit Robotern. 73
35 Versuchsaufbau mit der festen Sitzposition der Probanden, die 160 cm vom Anfang des Arbeitsbereiches entfernt ist. 74
36 Position der Objekte aus der Perspektive des Probanden (a) bevor die Aufgabe gestartet ist und (b) nach erfolgreicher Durchführung der Aufgabe. 75
37 Aufgabenaufbau (a) vor und (b) nach der Aufgabe. Die Kennzeichnung der Objekte entspricht der Reihenfolge. 76
38 Mittelwert der Aufgabenzeit (a) über die Versuche und (b) über die drei Versionen der visuellen Helfer mit Standardabweichung. . 78
39 Mittelwert der Präzision in mm bei den drei verschiedenen Objekten über alle drei Versuche mit Standardabweichung. Die Präzision entspricht dem Abstand des Grippers vom Objekt oder zur perfekten Platzierposition, als das Objekt gegriffen oder platziert wurde. 79
40 Mittelwert der Anzahl der Befehle bei den Objekten mit Standardabweichung. 81
41 Mittelwert der Feinjustierzeit bei den Objekten aufgeteilt auf Pick und Place mit Standardabweichung. 82
42 Auswertung der sechs Kategorien des Nasa-TLX und der daraus berechneten insgesamten Arbeitsbelastung mit Standardabweichung. Je geringer der Wert desto besser schneidet jede Kategorie ab. . . . 83
43 Auswertung der vier Dimensionen des USE Fragebogens mit Standardabweichung. 84

VI Abkürzungsverzeichnis

API Application Programming Interface.

AR Augmented Reality.

AV Augmented Virtuality.

BCI Brain-Computer-Interface (Gehirn-Computer-Schnittstelle).

DOF Degrees of Freedom.

EEG Elektroenzephalografie.

EMG Elektromyografie.

FOV Field of View.

FPS Frames Per Second.

HMD Head-Mounted Display.

IDP Interpupillardistanz.

IMU Inertial Measurment Unit (Inertiale Messeinheiten).

MIA Mensch-Roboter Interaktion im Arbeitsleben bewegungseingeschränkter Personen.

MR Mixed Reality.

MRI Mensch-Roboter-Interaktion.

RE Real Environment.

UDP User Datagramm Protocol.

VE Virtual Environment.

VR Virtual Reality (Virtuelle Realität).

1 Einführung

Körperliche Behinderungen können einen Menschen soweit einschränken, dass für sie ein autonomes und selbstbestimmtes Leben nicht mehr möglich ist. Tetraplegiker sind meist vollständig vom Hals abwärts gelähmt. Durch diese schwere Behinderung sind sie in ihrem Alltag stark von einer intensiven Betreuung abhängig. Ihre körperlichen Einschränkungen erschweren ihnen das Ausleben ihrer kognitiven Fähigkeiten. Daher ist jede Art der Selbstbestimmung oder Autonomie für sie eine Steigerung der Lebensqualität. Die Integration ins Arbeitsleben spielt eine wichtige Rolle für ein unabhängiges und selbstbestimmtes Leben in der Gesellschaft. Durch den Einsatz unserer kognitiven Fähigkeiten bekommen wir ein Gefühl von Anerkennung und Zufriedenheit. Zusätzlich ermöglicht der eigenständige Verdienst des Lebensunterhalts finanzielle Unabhängigkeit.

Damit Tetraplegiker oder Menschen mit einer ähnlichen körperlichen Einschränkung ihre kognitiven Fähigkeiten in der Arbeitswelt einsetzen können, ist in dieser Masterarbeit ein Prototyp entstanden, mit dem sie einen Roboterarm steuern können. Dieser Prototyp ermöglicht ihnen das Durchführen von Montageaufgaben an einer festen Arbeitsstation. Solche Mensch-Roboter Arbeitsplätze profitieren von menschlichen Eigenschaften, die denen von Robotern oder moderner Technik überlegen sind. Diese menschlichen Eigenschaften sind unter anderem die Fähigkeiten der flexiblen, komplexen Aufgabenlösungen oder der vorausschauenden Planung von Arbeitsprozessen. Solche Arbeitsplätze bieten sich vor allem für Tetraplegiker oder Menschen mit einer ähnlichen körperlichen Einschränkung an, da die Steuerung eines Roboters ihnen die Interaktionen mit ihrer Umwelt ermöglicht.

Bei der Entwicklung dieses Prototyps stehen die besonderen Voraussetzungen, die diese Zielgruppe mit sich bringt, im Fokus. Eine wichtige Voraussetzung ist, dass sich der Prototyp ohne die Verwendung der Hände steuern lässt und für Tetraplegiker bedienbare Eingabemethoden verwendet. Zusätzlich muss berücksichtigt werden, dass sie eine fixe Haltung und somit Perspektive auf die Arbeitsstation einnehmen und nicht wie gesunde Benutzer durch ein zur Seite lehnen o. ä. die Perspektive einfach ändern können. Wenn zum Beispiel ein Objekt von einem anderem teilweise verdeckt wird, kann ein Tetraplegiker nicht einfach aufstehen, um so das Objekt vollständig zu erkennen. Um diesen Nachteil auszugleichen, wurden visuelle Helfer entwickelt, die das Blickfeld des Benutzers mit weiteren Informationen ausstatten und somit die Steuerung des Roboterarms unterstützen. Augmented Reality (AR) stellt einen vielversprechenden Ansatz dar, visuelle Helfer anzuzeigen und die Steuerung des Roboters verständlich über virtuelle Interaktionselemente

zu ermöglichen. Dabei ist ein Vorteil von AR, dass virtuelle Informationen in das Blickfeld des Benutzers projiziert werden können und dabei die reale Welt und somit auch der Roboterarm weiterhin sichtbar sind. Neben den Voraussetzungen von Tetraplegikern wurde auch berücksichtigt, dass sich Arbeitsplätze verändern können und Objekte, mit denen bei Montageaufgaben gearbeitet wird, sich in einem stetigen Wandel befinden. Daher sollen vor allem die visuellen Helfer unabhängig von Objekten funktionieren, damit der Prototyp sich dynamisch an einen sich verändernden Arbeitsplatz anpassen kann.

Zielsetzung dieser Masterarbeit, ist die Entwicklung eines Prototyps, mit dem Tetraplegiker oder Menschen mit einer ähnlichen körperlichen Einschränkung einen Roboterarm steuern können, um Montageaufgaben durchzuführen. Dabei wird ein besonderer Fokus auf visuelle Helfer gelegt, die den Benutzer bei der Bedienung unterstützen sollen. Das Konzept der visuellen Helfer und inwiefern es den Benutzer unterstützt, wurde in einer Studie mit dem Prototyp evaluiert.

1.1 Das MIA-Projekt

Diese Masterarbeit wurde im Rahmen des Forschungsprojekts Mensch-Roboter Interaktion im Arbeitsleben bewegungseingeschränkter Personen (MIA) entwickelt. Das MIA-Projekt ist ein Forschungsprojekt der Westfälischen Hochschule mit Partnerschaften von Universitäten und der Industrie (Gerken, 2020). Das Projekt erforscht Mensch-Roboter Arbeitsplätze, bei denen Mensch und Roboter kooperativ zusammenarbeiten. Solche Arbeitsplätze sind Teil der Industrie von morgen und bieten große Variantenvielfalt, sowie neue Dienstleistungen in Form eines *Workplace as a Service*. Sie besitzen großes Potenzial als Arbeitsplätze für Tetraplegiker oder Menschen mit ähnlichen Bewegungseinschränkungen. Daher werden im MIA-Projekt innovative Sensortechnologien und Interaktionsdesigns erforscht, um komplexe Robotersteuerungen ohne den Einsatz von Händen zu ermöglichen. Es sollen neue Konzepte erprobt werden, zum Beispiel Eingabemöglichkeiten mittels Inertial Measurment Unit (Inertiale Messeinheiten) (IMU), Eyetracking oder Elektrookulografie, sowie Feedback über AR Steuerungs- und Interaktionsmöglichkeiten.

1.2 Aufbau der Arbeit

Das erste einleitende Kapitel dient als Einführung in das Thema und beleuchtet die Motivation und Zielsetzung dieser Masterarbeit. Den Stand der Technik von Mensch-Roboter-Interaktion und deren Umsetzung mittels AR wird in Kapitel 2 erläutert. Zusätzlich wird auf die speziellen Voraussetzungen von Tetraplegikern eingegangen und bereits existierende Forschungen der Interaktion zwischen Tetraplegikern und Robotern vorgestellt. Im Kapitel 3 wird die Konzeption des Prototyps erläutert, dabei werden die zentralen Anforderungen und die darauf begründeten Designentscheidungen beschrieben. Der daraus entstandene Prototyp und dessen Umsetzung wurde in Kapitel 4 dargestellt. Neben der Implementierung der Interaktion werden vor allem die visuellen Helfer näher erläutert. Über die Evaluation des Prototyps wird in Kapitel 5 berichtet. Dabei werden das Vorgehen, die Analyse und die Bewertung der generierten Ergebnisse beschrieben. Das abschließende Kapitel 6 fasst die Masterarbeit sowie die Ergebnisse zusammen und ein abschließendes Fazit wird gezogen.

2 Mensch-Roboter-Interaktion

Durch die zweite industrielle Revolution in den 1960er Jahren hielten Roboter Einzug in Fabriken, um die Arbeiter von repetitiven, gefährlichen und/oder körperlich anstrengenden Arbeiten zu entlasten. Diese ersten industriellen Roboter ließen sich entweder programmieren oder über eine undirektionale Steuerung bedienen. Solch eine undirektionale Steuerung findet zum Beispiel über Joysticks, die die Achsen steuern, statt. Über die Jahre wuchsen Anzahl und Komplexität von Industrierobotern. Sie wurden immer mehr für mannigfache Aufgaben in Produktionsprozessen verwendet. Die Industrieroboter mussten flexibler und intelligenter werden, um den umfangreichen Anforderungen gerecht werden zu können. Die künstliche Intelligenz der Roboter stieg. Damit ähnelte die Kommunikation zwischen Mensch und Roboter immer weniger der Verwendung eines passiven Werkzeugs und immer mehr der Beziehung zwischen zwei Menschen (Sheridan, 1997). Der wissenschaftliche Bereich der Mensch-Roboter-Interaktion (MRI) untersucht diese Beziehung und widmet sich dem Verstehen, Designen und Evaluieren von Robotersystemen, die von oder mit Menschen genutzt werden. MRI ist eng mit den Bereichen der Mensch-Computer- und Mensch-Maschine-Interaktion verwandt, unterscheidet sich von diesen aber in den komplexen, dynamischen Bedienungs- und Steuersystemen, die autonom und in sich verändernden, realen Umgebungen arbeiten. Der Einsatz von Robotern geht über die Herstellung von Industrieerzeugnissen hinaus. Sie werden z. B. im Militär, in der Raumfahrt oder in der Medizin als kleine ferngesteuerte und bewaffnete Fahrzeug, als Erkundungsfahrzeuge auf dem Mars oder als Assistenz für die Chirurgie eingesetzt (Bogue, 2012; Hockstein, Gourin, Faust & Terris, 2007; Mies, 2010). Durch die steigende künstliche Intelligenz und Flexibilität können Roboter in immer mehr Anwendungsfeldern eingesetzt werden. Ein schnell wachsendes Anwendungsfeld ist die Entwicklung von Servicerobotern, die eng mit dem Menschen zusammenarbeiten und Unterstützung anbieten. Serviceroboter finden beispielsweise Einsatz in der Haushaltsführung, Rehabilitation oder in der Bildung (Teresa, 2014). Der Einsatz bei Benutzern, die nicht professionell oder noch nie einen Roboter verwendet haben, steigt damit. Daher ist ein Schwerpunkt von MRI die intuitive Bedienung und Interaktion von und mit Robotern über natürliche und direkte Kommunikationstechniken.

2.1 Roboterarm

Ein Roboterarm ist eine mechanische Konstruktion, die programmierbar ist und eine analoge Funktionalität zum menschlichen Arm aufweist. Ein Roboterarm kann Teil eines komplexeren Roboters sein, z. B. eines humanoiden Roboters mit zwei Roboterarmen auf einer mobilen Plattform, oder für sich alleine stehen. Die bewegliche physische Struktur des Roboterarms wird Manipulator genannt. Seine Bestandteile entsprechen Schulter, Arm, Handgelenk und Hand eines menschlichen Arms (Saha, 2008, S. 16). Die Ähnlichkeit eines typischen Roboterarms mit sechs Achsen zu einem menschlichen Arm ist in Abbildung 1 veranschaulicht.

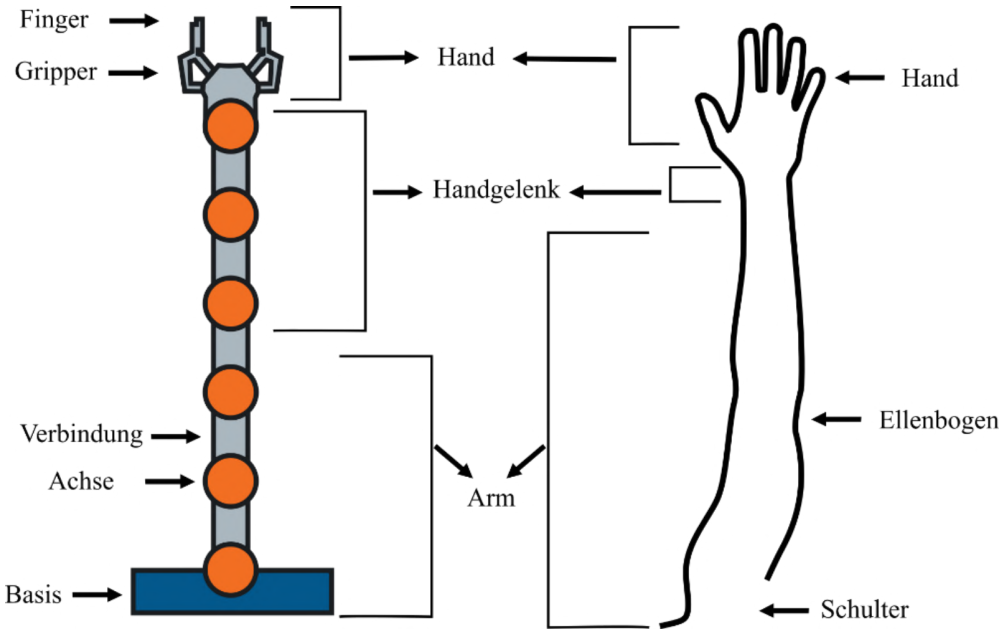


Abbildung 1: Roboterarm im Vergleich zum menschlichen Arm.

Die Schulter eines Roboterarms ist meistens an einer stationären Basis oder an einer mobilen Plattform analog zum Menschen befestigt. Insgesamt besteht der Manipulator aus mehreren Achsen und deren Verbindungen. Die Achsen sind vergleichbar mit einem Gelenk und werden auch Rotationsachsen genannt, da sie die an ihnen befestigten Verbindungen rotieren und damit die Bewegungen in einem festgelegten geometrischen Raum ermöglichen. Ein typischer Industrieroboter besteht aus sechs Achsen, die ihm sechs Degrees of Freedom (DOF) ermöglichen. Der menschliche Arm hat zum Vergleich sieben DOF. Die ersten drei Achsen bilden den Arm und bestimmen die Position im dreidimensionalen kartesischen Raum und die letzten drei orientieren, analog zum menschlichen Handgelenk, einen Endeffektor. Er ist am Ende des Manipulators befestigt und übernimmt die Aufgabe der Hand. Aufgrund unterschiedlicher spezifischer Anwendungserfordernisse wurden viele verschiedene Arten von Endeffektoren entwickelt. Ein verbreiteter Endef-

fektor ist eine vereinfachte Version der menschlichen Hand, die unterschiedliche Objekte greifen und transportieren kann. Ein sogenannter einfacher Gripper, wie er in Abbildung 1 dargestellt wird, besteht meistens aus zwei Fingern. Für komplexere Greifaufgaben gibt es Gripper mit drei, vier oder fünf Fingern. Als Endeffektor werden auch deutlich spezialisiertere Werkzeuge, wie z. B. Schweißelektroden, Gasschweißbrenner oder Pinsel, eingesetzt. Zur besseren und sichereren Interaktion mit der Umwelt werden Endeffektoren meistens mit Sensoren ausgestattet, die eine Kontrolle des Druckes oder der Greifstärke ermöglichen.

Noch vor 20 Jahren war das Feld der Robotik durch Industrieroboter geprägt. Diese sind sehr schnell, schwer und haben mit meistens sechs DOF keine große Beweglichkeit. Diese Eigenschaften machen sie für den Menschen gefährlich und müssen durch Schutzzäune abgetrennt werden. Derartige Industrieroboter sind vor allem in der Autoindustrie stark vertreten. Jedoch verlangen sowohl Industrie als auch der Dienstleistungssektor immer mehr nach engerer Zusammenarbeit zwischen Mensch und Roboter. Eine Kollaboration von Mensch und Roboterarm kann zu mehr Flexibilität in der Produktion und weiterer Entlastung von körperlich anstrengenden Aufgaben führen. Jedoch steigen bei solchen kollaborativen Robotern die Sicherheitsanforderungen. Sie müssen weniger Kollisionsmasse und abgerundete oder mit Schaumstoff umhüllte Ecken, Kante und Klemmstellen besitzen. Daher wird nicht nur der Endeffektor mit Sensorik ausgestattet, sondern der gesamte Arm benötigt Sensoren, die Berührungen und Kollisionen erkennen können. Darüber hinaus brauchen sie mehr Beweglichkeit, um nicht nur die Position im Raum zu verändern, sondern auch die der einzelnen Achsen. Ein Anwendungsfall besteht darin, dass ein Roboterarm seine Position und Orientierung mit dem Endeffektor beibehält, während er alle anderen Achsen bewegen kann. Zur Ermöglichung dieser Art der Beweglichkeit benötigt der Roboterarm mehr Achsen/DOF, als er für seine eigentliche Aufgabe beansprucht, er muss kinematisch redundant sein. Für die Positionierung und Orientierung im dreidimensionalen Raum sind sechs DOF ausreichend, während ein Roboterarm mit diesem Aufgabenraum der sieben Achsen/DOF besitzt, redundant ist und somit z. B. Kollisionen vermeiden kann. In Abbildung 2 ist der Unterschied zwischen einem Industrieroboter und einem kollaborativen Roboterarm dargestellt. Die unterschiedlichen Eigenschaften ermöglichen nicht nur eine sicherere Zusammenarbeit, sondern erlauben auch durch die Ähnlichkeit zum menschlichen Arm eine adaptive Kollaboration (Onnasch, Maier & Jürgensohn, 2016).

(a) Der KR 1000 Titan Schwerlast-Roboter besitzt eine Traglast von bis zu 1300 kg und eine Reichweite von bis zu 3,6 m (KUKA AG, 2019a).

(b) Der LBR iiwa ist ein kollaborativer Leichtbau-Roboter und besitzt eine Traglast von bis zu 14 kg und eine Reichweite von 0,8 m (KUKA AG, 2019b).

Abbildung 2: Vergleich zwischen einem (a) Schwerlast-Roboter und einem (b) kollaborativen Leichtbau-Roboter.

2.1.1 Bedienung von Roboterarmen

Manche Roboterarme können über ein Handbediengerät in Echtzeit gesteuert werden. In Abbildung 3 ist ein Handbediengerät dargestellt, das die gezielte Steuerung der einzelnen Achsen und weitere Einstellungen des Roboters ermöglicht. Diese Art der Steuerung ähnelt der eines ferngesteuerten Autos und ist schon für einfache Aufgaben relativ umständlich. In der Industrie wird jeder Roboterarme speziell für seinen Aufgabenbereich programmiert. Danach führt der Roboterarm den Arbeitsablauf autonom durch oder kann ihn mit Hilfe von Sensorinformationen in Grenzen variieren. Es existieren dafür unterschiedliche Programmiersprachen und -arten, die hauptsächlich von Experten angewendet werden können. Denn Roboterarme für eine spezifische Aufgabe zu programmieren ist schwierig zu erlernen, zeitaufwändig und teuer. Beispielsweise dauert die Programmierung einer Roboterlichtbogenschweißanlage für die Herstellung eines Großfahrzeugs mehr als acht Monate, während die Zykluszeit des Schweißprozesses selbst nur 16 Stunden beträgt. In diesem Fall nimmt die Programmierzeit in etwa das 360-fache der Ausführungszeit in Anspruch (Pan, Polden, Larkin, Van Duin & Norrish, 2010). Daher ist das Interesse aus Industrie und Gewerbe groß, neue Wege der Roboterprogrammierung oder Steuerung zu erforschen. Dem Benutzer soll es ermöglicht werden, einen Roboterarm dazu zu bringen eine Aufgabe auszuführen, ohne eine Programmiersprache beherrschen zu müssen.

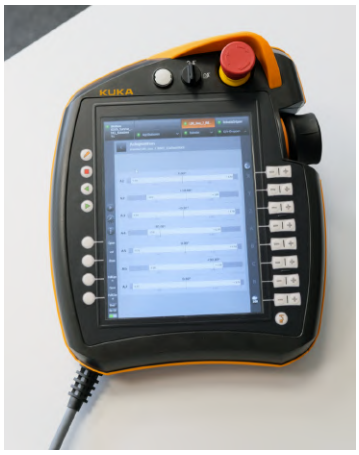


Abbildung 3: Das SmartPad vom Kuka iiwa.

Programmieren durch Demonstration ist eine häufig genutzte Methode, durch die Roboter anhand der Führung/Anleitung durch Menschen neue Fertigkeiten erlernen. Es wird unter anderem auch Lernen durch Imitation genannt und orientiert sich an der Art und Weise, wie Menschen lernen. Dabei existieren viele unterschiedliche Methoden, mit denen neue Fähigkeiten auf einen Roboter übertragen werden können. Eine davon ist das kinästhetische Lernen (kinesthetic teaching), bei dem der Roboterarm physisch bewegt wird (Akgun, Cakmak, Jiang & Thomaz, 2012; Akgun & Subramanian, 2011; Kormushev, Nenchev, Calinon & Caldwell, 2011). Dazu werden die Bremsen der Achsen gelöst, so dass sie nur noch die Schwerkraft kompensieren und eine einfache Steuerung durch den Menschen ermöglichen. Dieser kann den Roboterarm dann am Endeffektor führen und der Manipulator bewegt sich mit. Der Roboter merkt sich die angefahrenen Positionen oder den geführten Weg und kann diesen wiedergeben. Diese Methode ist einfach zu erlernen und verlangt vom Benutzer kein Vorwissen. Jedoch muss er sich nah am Roboterarm aufhalten, was nicht immer möglich ist. Außerdem können bereits kleine Änderungen an den Positionen oder dem Weg im Nachhinein sehr aufwendig werden (Pan et al., 2010). Ein anderer Ansatz ermöglicht die Steuerung eines Roboterarms mittels Gesten. Sie sind für den Menschen eine natürliche Art und Weise der Kommunikation und können daher ein intuitives Interaktionskonzept darstellen. Es existieren unterschiedliche Methoden, um Gesten vom Menschen zu erkennen und sie als Bewegung auf einen Roboterarm zu übertragen (Liu & Wang, 2018). Elektromechanische Sensoren, die in einem Handschuh (Kukliński et al., 2014) oder in einer Art Band, das am Unterarm getragen wird (Wolf, Assad, Vernacchia, Fromm & Jethani, 2013), befestigt sind, können die Position der Hand und Finger bestimmen. Dadurch kann der Benutzer durch das Bewegen der Hand den Endeffektor manipulieren und durch das Schließen der Hand ebenfalls das Schließen des Grippers veranlassen. Ein Nachteil dieses Ansatzes ist, dass solche Handschuhe oder Bänder aufgrund der notwendigen technischen Präzision sehr kostenintensiv

sind. Darüber hinaus sind Hände von Mensch zu Mensch unterschiedlich und erfordern daher häufig eine Neukalibrierung. Diese kann die Bedienung solcher Handschuhe verkomplizieren und zum Verlust der intuitiven Kommunikation führen. Der visuell basierende Ansatz bestimmt die Position von der Hand und Finger mit Hilfe von digitalen Kameras (Hasanuzzaman et al., 2004) oder Kameras mit Tiefensensoren (Bolano, Tanev, Steffen, Roennau & Dillmann, 2018), wie z. B. die Microsoft Kinect. Bildverarbeitungsalgorithmen können dann Gesten in Echtzeit erkennen und auf den Roboter übertragen. Dieser Ansatz benötigt keine tragbaren Sensoren, allerdings ist die Bewegungsfreiheit durch den Aufnahmebereich der Kameras eingeschränkt. Außerdem funktioniert dieser Ansatz am besten in einer kontrollierten Umgebung, in der das Kamerabild nicht verdeckt werden kann und die Lichteinflüsse optimal eingestellt wurden. Beim Gesten und kinästhetischen Ansatz handelt es sich um Ansätze, die wenig bis kein autonomes Verhalten des Roboterarms verlangen. Natural Language Interfaces stehen dazu im Gegensatz. Bei dieser Methode wird der Roboterarm mit verbalen Kommandos so instruiert, dass er Objekte manipulieren kann. Diese bestehen aus räumlicher Sprache, die die Objektreferenz ausdrücken und Positions- und Richtungspräpositionen verwenden (Forbes, Rao, Zettlemoyer & Cakmak, 2015; Jia et al., 2014). Zum Beispiel „hebe den kleine brauen Würfel auf“ oder „gehe zu dem roten Objekt ganz links“. Über eine Kamera und Bildverarbeitungsalgorithmen werden die Objekte und ihre Position erkannt, damit der Roboterarm den Weg zu diesem Objekt berechnen und die Anweisung autonom ausführen kann. Obwohl Sprache eine sehr natürliche Art zur Interaktion ist und einen vielversprechenden Ansatz darstellt, besitzen Natural Language Interfaces noch ungelöste Herausforderungen. Diese lassen sich auf die Diskrepanz zwischen der Art und Weise, wie Menschen und Roboter Informationen darstellen und kommunizieren, zurückführen. Dadurch wird es dem Roboter erschwert, seine Absichten effektiv auszudrücken oder die des Menschen richtig zu interpretieren. Das Lösen dieser Kommunikationsbarriere (Kaminka, 2013) ist ein wichtiger Forschungsbereich und gewinnt an Wichtigkeit, je enger Mensch und Roboter zusammenarbeiten.

2.1.2 Zusammenarbeit von Mensch und Roboter

Neben dem Forschungsbereich zur schnellen und präzisen Programmierung von Robotern, existiert ein Bereich der erforscht, wie die Fähigkeiten von Menschen und Robotern kombiniert werden können, um ein gemeinsames Ziel zu erreichen. Natural Language Interfaces bieten unter anderem diese Möglichkeit. Der Mensch kennt das übergeordnete Ziel und leitet den Roboter an, der die tatsächliche Umsetzung übernimmt. Die Kombination der jeweiligen Stärken von Mensch und Robotern ist ein vielversprechender Ansatz, um flexibel komplexe Aufgaben lösen zu können. Menschen besitzen im Gegensatz zu Robotern Attribute, wie heuristisches Wissen, Kreativität, Geschicklichkeit und gesunden Menschenverstand. Umgekehrt sind schnelle Berechnungen, mechanische Leistung und Ausdauer Vorteile von Robotern. Der technologische Fortschritt ermöglicht Robotern, physisch mit dem Menschen zusammen zu arbeiten und somit die Fähigkeit beider zu kombinieren und zu verbessern. Diese Zusammenarbeit bietet vielfältige potenzielle Anwendungen in der Industrie als Roboterkollegen und im Servicebereich als persönlicher Assistent (Chandrasekaran & Conrad, 2015).

Die Interaktion mit einem Roboter lässt sich nach Goodrich, Schultz et al. (2008) in zwei generelle Kategorien aufteilen. Die erste Kategorie ist die entfernte Interaktion, bei der Mensch und Roboter räumlich oder sogar zeitlich voneinander getrennt sind. Ein Beispiel stellt die Teleoperation eines Mars Rovers dar, welche räumlich und zeitlich von der Erde getrennt sind. Bei der zweiten Kategorie handelt es sich um die unmittelbare Interaktion, wobei sich Mensch und Roboter am selben Standort zur selben Zeit befinden. Wie es zum Beispiel bei der Kollaboration von Mensch und Roboter an einem Industrieförderband der Fall ist (Goodrich, Schultz et al., 2008). Die unmittelbare Interaktion fasst Kooperation und Kollaboration zusammen. Die Kollaboration unterscheidet sich von der Kooperation dadurch, dass Mensch und Roboter ein Ziel, das sie nur durch gemeinsames Handeln erreichen, verfolgen und der Erfolg beider voneinander abhängt (Hord, 1981). Jedoch spielt bei Kooperation und Kollaboration Sicherheit die wichtigste Rolle (Flacco & De Luca, 2013). Die Sicherheitsanforderungen unterscheiden sich von den üblichen Methoden, bei denen der Mensch durch Käfige, Verlangsamung oder Anhalten der Roboter geschützt werden. Sie sind für eine Zusammenarbeit ungeeignet und lassen keine wirkliche Interaktion zu. Für eine effektive Interaktion ist es nicht nur nötig, dass Roboter und Mensch sicher miteinander arbeiten, sondern auch kommunizieren können. So muss der Roboter in der Lage sein, Kommunikationsmechanismen, die denen der Mensch-Mensch-Interaktion ähneln, zu verstehen und zu interpretieren. Außerdem muss der Roboter Feedback über seine Absichten und Ziele mitteilen können. Fehlende

Kommunikation, kann das Verhalten des Roboters unnachvollziehbar machen und dazu führen, dass der Mensch sich nicht wohl fühlt oder Angst empfindet, wenn er in der unmittelbaren Nähe des Roboters arbeitet (Hoffman & Breazeal, 2004).

2.2 Kommunikation durch Virtual, Augmented und Mixed Reality

Ein integraler Bestandteil jeder Mensch-Roboter-Interaktion ist die Kommunikation der Partizipierenden. Mit Natural Language Interfaces und Gesten wurden Bedienungsmethoden vorgestellt, die sich zwar eines natürlichen Kommunikationskanals bedienen, wobei die Kommunikation jedoch größtenteils einseitig vom Menschen ausgeht. Dabei ist es von großer Bedeutung, dass Menschen die Aktivitäten von Robotern erkennen und lesen können, um das Verhalten des Roboters zu interpretieren (Christensen et al., 2009). Daher ist es wichtig, dass ein Roboter seine Absichten kommunizieren und Feedback geben kann. Die Techniken von Virtual, Augmented und Mixed Reality bieten vielversprechende Ansätze zur Vermittlung von Mensch-Roboter-Interaktionen. Sie ermöglichen die Kommunikation von Absichten, indem sie den physischen Interaktionsraum zwischen Mensch und Roboter als gemeinsame Fläche für virtuelle Hinweise nutzen. Während in diesem Bereich seit vielen Jahren Vorarbeiten durchgeführt werden, verbessern die jüngsten technologischen Fortschritte die Durchführbarkeit und machen Virtual, Augmented und Mixed Reality daher zu einem vielversprechenden Ansatz (Williams, Szafir, Chakraborti & Ben Amor, 2018).

Um den Fokus darauf zu legen, was Virtual, Augmented und Mixed Reality als Interaktionsdesign bieten können, ist es zunächst wichtig, eine klare Definition und Abgrenzung der Begrifflichkeiten zu beschreiben. Daher werden im Folgenden die verschiedenen Konzepte und das Reality-Virtuality-Kontinuum vorgestellt, um danach auf die Verwendungsmöglichkeiten im Kontext von Mensch-Roboter-Interaktion einzugehen.

2.2.1 Das Reality-Virtuality-Kontinuum

Virtual Reality bezeichnet die Simulation und gleichzeitige Wahrnehmung einer Welt, die in Echtzeit eine computergenerierte, interaktive und virtuelle Umgebung anzeigt. Dabei muss die reale Welt nicht nachgeahmt werden, sondern kann über ihre physikalischen Grenzen hinaus gehen. Augmented Reality schließt im Gegensatz dazu die reale Welt nicht aus, sondern erweitert sie mit computergenerierten Inhalten. Reale Welt, virtuelle Gegenstände und Informationen werden also gemischt. Milgram, Takemura, Utsumi und Kishino (1995) sehen diese beiden Techniken nicht als Antithesen, sondern vereinen sie als gegenüberstehende Enden im Reality-Virtuality-Kontinuum. Dieses Kontinuum ist in Abbildung 4 dargestellt. Am linken Ende steht die Real Environment (RE), das die reale Welt ohne Erweiterungen repräsentiert. Am gegenüberliegenden rechten Ende befindet sich die Virtual Environment (VE), in der das Sichtfeld des Benutzers komplett mit virtuellem Inhalt ersetzt wird. Alles zwischen diesen beiden Extremen fällt unter Mixed Reality (MR). Darunter befinden sich AR und Augmented Virtuality (AV). AV liegt näher an VE, da das meiste vom Sichtfeld des Benutzers durch virtuelle Inhalte ersetzt wird. Zum Beispiel wenn die Umgebung computergeneriert ist, aber manche Objekte der realen Welt hineinprojiziert werden. AR befindet sich näher an RE, dabei sieht der Benutzer die reale Umgebung, die durch virtuelle Inhalte erweitert wird. Je mehr virtuelle Inhalte zur AR hinzugefügt werden, desto weiter bewegt es sich weg von der RE hin zur Virtual Reality (Virtuelle Realität) (VR). Daraus lässt sich schließen, dass AR nicht als diskreter Punkt zwischen realer und virtueller Welt, sondern überall auf dem stufenlosen Reality-Virtuality-Kontinuum existieren kann (Milgram et al., 1995).

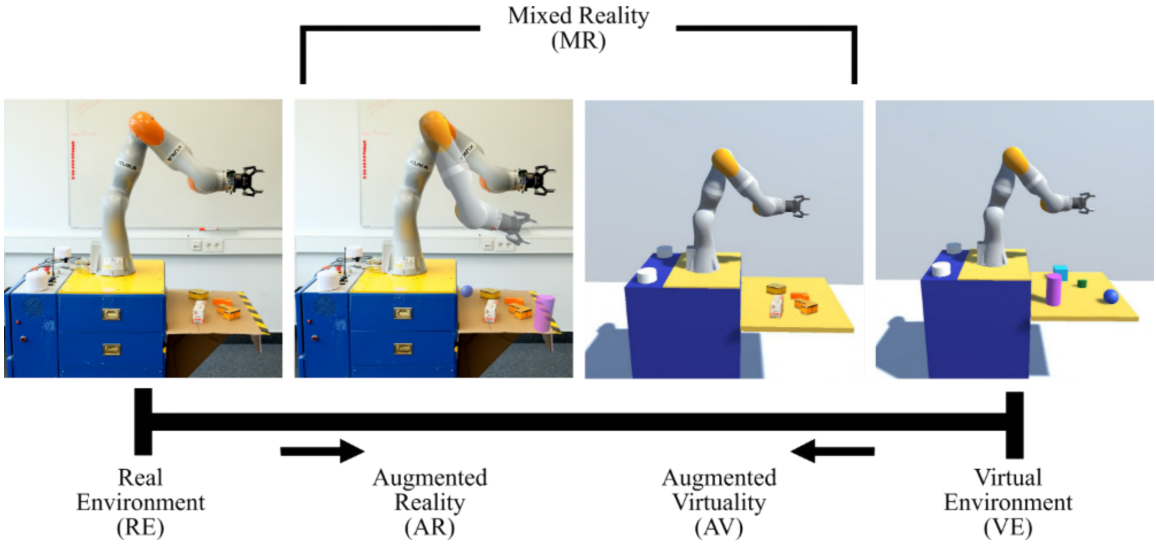


Abbildung 4: Darstellung vom Reality-Virtuality-Kontinuum.

Milgram et al. definieren Virtual, Augmented und Mixed Reality, indem sie diese Technologien in einen Kontext miteinander bringen. Sie fanden damit eine Definition, die in der Forschung breite Akzeptanz findet. Eine ebenfalls anerkannte Definition, die sich gut mit der von Milgram et al. vereinen lässt, stellt Azuma (1997) für Augmented Reality auf. Diese ist unabhängig von technologischen Geräten und besteht aus folgenden drei Eigenschaften:

„1. Combines real and virtual
2. Is interactive in real time
3. Is registered in three dimensions“ (Azuma, 1997)

Diese Punkte beschreiben die Fähigkeiten, die ein System mit sich bringen muss, um Augmented Reality zu ermöglichen. Der erste Punkt formuliert die Haupteigenschaft von Augmented Reality: Um Augmentierung hervorzurufen, muss die physische Welt mit virtuellen Elementen überlagert werden können. Dies beschränkt sich nicht nur auf das Sehen von virtuellen Elementen, sondern schließt akustische, haptische oder olfaktorische Stimuli mit ein. Damit geht seine Definition über die von Milgram et al., die sich nur mit den visuellen Komponenten beschäftigt, hinaus. Des Weiteren beschreibt er im zweiten Punkt, dass das System interaktiv in Echtzeit funktionieren muss. Das bedeutet, dass zum Beispiel der Kamerastream in Echtzeit erfasst und verarbeitet wird, um die virtuellen Elemente, ebenfalls in Echtzeit, abhängig vom Kamerabild anpassen zu können. Abschließend beschreibt der dritte und letzte Punkt, dass reale und virtuelle Objekte dreidimensional zueinander in Bezug stehen müssen. Dadurch werden die virtuellen Elemente in der realen Welt verankert und sind unabhängig vom Standpunkt des Benutzers oder seiner Kamera.

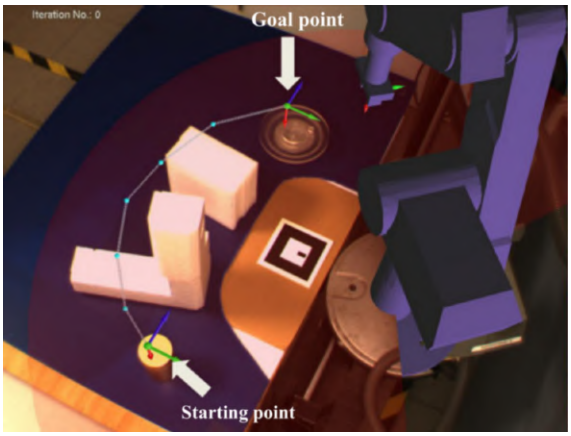
Im Gegensatz zu diesen beiden anerkannten Definitionen erstellte Microsoft seine eigene Definition von Mixed Reality und Augmented Reality. Diese lehnt sich lose an der von Milgram et al. an, unterscheidet sich aber in dem wichtigen Punkt, dass Augmented Reality kein Teil von Mixed Reality ist, sondern Mixed Reality dort anfängt, wo Augmented Reality aufhört. Nach Microsofts Definition ist Augmented Reality nur das Überlagern von physischen und virtuellen Umgebungen, aber nicht das Mischen dieser (Bray, McCulloch, Schonning & Matt, 2018). Diese konträre Ansicht führt oft zu Verwirrungen und wird daher in dieser Arbeit außen vorgelassen: Die Begriffe Augmented Reality und Mixed Reality können austauschbar verwendet werden, dabei ist Augmented Reality die genauere Beschreibung der Technologie und Mixed Reality der Überbegriff.

2.2.2 Visualisierung über Displays und Projektionen

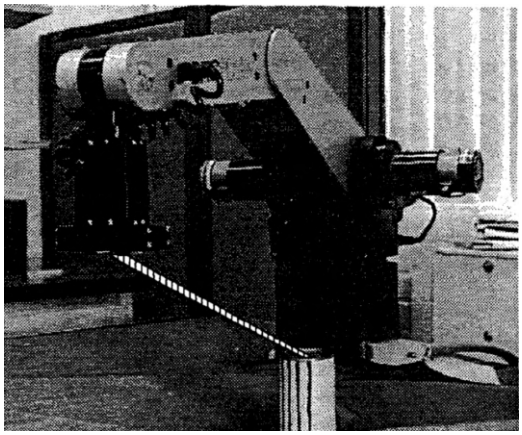
Fang, Ong und Nee (2014) stellen in ihrer Arbeit ein AR-basiertes Interface für Mensch-Roboter-Interaktion vor, das durch einen Monitor visualisiert wird. Es soll die Roboterprogrammierung verbessern, indem es die Pfadplanung vereinfacht. Der geplante Pfad wird angezeigt und die Bewegung, die der Roboterarm beim Abfahren des Pfades durchläuft, wird simuliert. Dadurch kann der Benutzer den Pfad besser nachvollziehen und ggf. Veränderungen vornehmen. Um das zu ermöglichen, ist eine Kamera mit einer festen Position und Orientierung auf den Arbeitsbereich des Roboterarms gerichtet. Mit Hilfe von Markern kann so ein virtueller Roboter über den realen Roboter augmentiert werden. Der Benutzer kann mit einem Marker bestückten Interaktionswerkzeug den Pfad im realen Arbeitsbereich zeichnen. Danach erscheint der Pfad in Form von virtuellen Linien, die über das Kamerabild augmentiert sind, auf dem Monitor. Weitere virtuelle visuelle Helfer, wie z. B. die farbliche Hervorhebung des Arbeitsbereiches oder das Anzeigen der verschiedenen Koordinatensysteme des Roboterarms, werden ebenfalls auf dem Monitor dargestellt und unterstützen den Benutzer bei der Interaktion (Fang, Ong & Nee, 2012, 2014). Diese visuellen Helfer sind in Abbildung 5a dargestellt.

Einen ähnlichen Aufbau hatten auch Milgram, Zhai, Drascic und Grodski (1993), knapp 20 Jahre früher, für ihren Mensch-Roboter-Interaktions Prototyp gewählt. Jedoch unterscheidet sich dieser in zwei wesentlichen Punkten. Erstens ist der Prototyp für die Teleoperation eines Roboterarms gedacht, daher findet keine Interaktion im Arbeitsbereich des Roboterarms statt. Zweitens wird kein einfacher Monitor verwendet, sondern ein stereoskopischer Monitor und dementsprechend auch eine stereoskopische Kamera. Dies ermöglicht die dreidimensionale Darstellung des Roboterarms und seines Arbeitsbereiches. Stereoskopische Monitore zeigen eine Dimension mehr an und dadurch auch mehr Informationen. Diese Art der Darstellung verlangt weniger mentale Verarbeitung und ist daher für den Benutzer zugänglicher. Zusätzlich zu der stereoskopischen Darstellung des Arbeitsbereiches des Roboterarms, werden auch visuelle Helfer als stereoskopische Grafiken über diese augmentiert. Unter diesen visuellen Helfern sind unter anderem ein virtueller Zeiger, der für die Beurteilung der absoluten Tiefe zuständig ist und damit die Tiefenwahrnehmung verbessert. Darüber hinaus wird ein virtuelles Maßband eingesetzt, das bei der Quantifizierung der realen Welt helfen soll. Eine virtuelle Linie, die zwischen dem Endeffektor und dem angestrebten Zielpunkt aufgespannt ist und sich mit der Position des Roboterarms fortlaufend anpasst, soll die Wahrnehmung von Bewegungen verbessern. Zusätzlich stellt dieser sogenannte virtuelle Tether, der in Abbildung 5b dargestellt ist, dreidimensionale Informationen zur Position und Orientierung des Endef-

fektors relativ zum Ziel dar (Milgram et al., 1993). Studien über diesen stereoskopischen Tether und Display zeigten, dass im Gegensatz zur monoskopischen Darstellung mehr nützliche räumliche Informationen angezeigt werden konnten. Denn die Probanden machten weniger Fehler und waren in ihrer Leistung konsistenter (Ruffo & Milgram, 1992).



(a) AR-basiertes Interface mit dem augmentierten Roboterarm, Pfadplanung und Koordinatensystemen (Fang, Ong & Nee, 2012).



(b) Konzept vom virtuellen Tether der auf dem stereoskopischen Monitor dargestellt wird (Milgram, Zhai, Drascic & Grodski, 1993).

Abbildung 5: Monitor-basierende AR-Prototypen für Mensch-Roboter-Interaktion von (a) Fang, Ong und Nee und (b) Milgram, Zhai, Drascic und Grodski.

Fang et al. (2014) und Milgram et al. (1993) verfolgen mit ihrem Prototypen den Ansatz von statischen Kameras und einen festen Arbeitsbereich. Dieser Aufbau hat den Nachteil, dass durch die statische Perspektive das Situationsbewusstsein des Benutzers negativ beeinträchtigt werden kann. Beispielsweise kann das eingeschränkte Sichtfeld der Kameras einen Schlüssellocheffekt verursachen, bei dem der Benutzer wichtige Ereignisse außerhalb des Bildschirms nicht mitbekommt (Woods, Tittle, Feil & Roesler, 2004). Zusätzlich wird die mentale Arbeitsbelastung erhöht, wenn Benutzer und Roboter, wie bei Fang et al., co-lokalisiert sind. Denn der Benutzer tendiert dazu, seine Aufmerksamkeit zwischen Monitor und der direkten Perspektive hin und her zu wechseln. Dies erhöht die mentale Arbeitsbelastung (J. Y. Chen, Haas & Barnes, 2007). Frank, Moorhead und Kapila (2017) versuchen mit ihrem Mensch-Roboter-Interaktions Prototyp diese beiden Nachteile zu umgehen, indem sie als Kamera und Display ein Mobilgerät verwenden. Dieses wird vom Benutzer auf den Arbeitsbereich des Roboters gerichtet und kann dann zusätzliche räumliche Informationen über das Kamerabild des Mobilgeräts auf diesem augmentieren. Damit wird ein Bezugsrahmen für den Roboter und Benutzer geschaffen und räumliche Informationen für Objektmanipulierung können effektiv ausgetauscht werden. Diese Informationen bestehen unter anderem aus einem virtuellen Grid oder dem

aktuellen Status des Roboters. Die neuesten technologischen Fortschritte bei mobilen Geräten, wie Smartphones oder Tablets, machen sie zu attraktiven Geräten für die Visualisierung der Mensch-Roboter-Interaktion. Ein großer Vorteil ist, dass keine Latenz durch Kodierung oder Streaming entsteht, da die Kamera des Gerätes direkt verwendet werden kann. Die Mobilität ermöglicht eine flexible Perspektive, die der des Benutzers entspricht (Frank et al., 2017). Neben diesen Vorteilen besteht der Nachteil, dass das Mobilgerät vor das Sichtfeld des Benutzers gehalten werden muss. Daher sind beide Hände des Benutzers belegt, was nach einiger Zeit als umständlich und anstrengend empfunden werden kann.

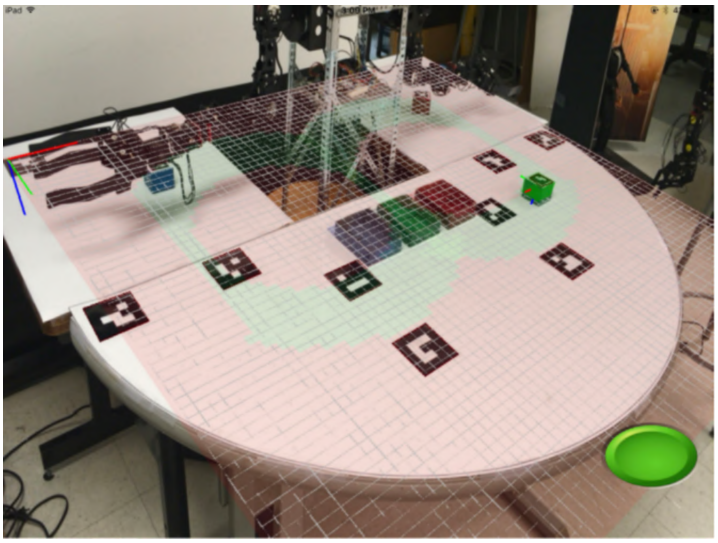
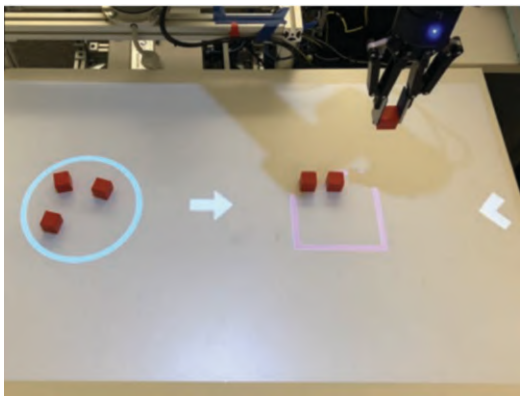


Abbildung 6: Screenshot vom Mobilgerät, das auf den Arbeitsbereich vom Roboter gerichtet ist und das Grid und die vom Roboter erreichbaren Flächen augmentiert (Frank, Moorhead & Kapila, 2017).

Ein Ansatz ohne Monitore verfolgen Gao und Huang (2019) mit ihren projektionsbasierten augmentierten Table-Top Interface für Roboterprogrammierung, kurz PATI (Projection-based Augmented Table-top Interface for robot programming). Bei diesem Prototyp sind Roboterarm und Benutzer co-lokalisiert. Sie sitzen sich gegenüber, mit einem Tisch in der Mitte und einem Projektor über dem Tisch. Das Interface wird auf den Tisch und damit auf den geteilten Arbeitsbereich von Benutzer und Roboter, projiziert. Mit Hilfe von allgemein bekannten Touchscreen Gesten kann der Benutzer mit dem Roboter Objektmanipulationen durchführen. Dabei werden zum Beispiel zu manipulierende Objekte mit einem projizierten Kreis oder die Transportrichtung mit einem Pfeil, wie in Abbildung 7a, gekennzeichnet (Gao & Huang, 2019). Da die Projektionen nur flach auf dem Tisch dargestellt werden können, ist dieser nicht viel mehr als ein Display Ersatz. Ein Prototyp, der mehr Potenzial aus dem Ansatz der projektions-basierten Augmentierung holt, stellen Ganesan, Rathore, Ross und Amor (2018) vor. Dieser soll durch das Projizieren von

visuellen Helfern die Mensch-Roboter-Kollaboration verbessern. Sowohl die physische Arbeitsumgebung als auch Objekte werden als Projektionsoberfläche benutzt, um aufgabenbezogene Anweisungen und Absichten des Roboters in Form von visuellen Helfern zu kommunizieren. Mit Hilfe visueller Objekterkennung und -verfolgung werden Objekte getrackt und damit die Visualisierung, wie in Abbildung 7b, auf ihnen ermöglicht. Zusätzlich können dadurch Just-in-Time Instruktionen für den Benutzer gegeben werden. Eine Studie mit diesem Prototyp zeigte, dass mit den visuellen Helfern kollaborative Konstruktionsarbeiten nicht nur effizienter und effektiver durchgeführt werden konnten, sondern die Probanden die Aufgaben angenehmer wahrnahmen und sich mit dem Roboter als ein Team empfanden (Ganesan et al., 2018). Bei beiden projektions-basierten Ansätzen ist der Projektor statisch montiert. Dies hat den Nachteil, dass die Projektion vom Menschen oder Roboter leicht verdeckt werden kann. Dadurch können wichtige Hinweise, trotz der achtsamen Positionierung von Benutzer und Roboter, überlagert werden. Ein Beispiel einer Verdeckung, die der Roboterarm verursacht, ist in Abbildung 7a zu erkennen.



(a) Prototyp PATI mit Verdeckung durch den Roboterarm (Gao & Huang, 2019).

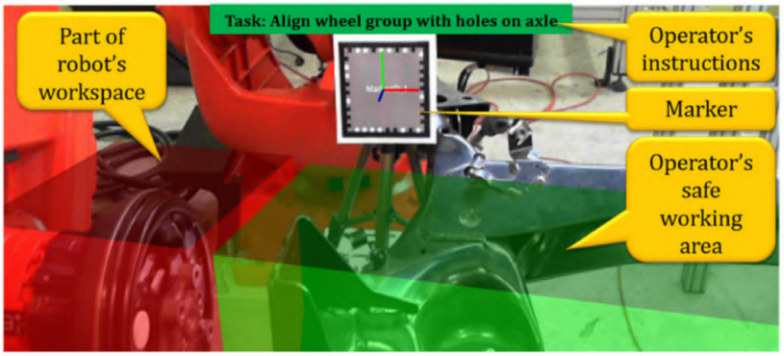
(b) Projektions-basierte Mensch-Roboter Kollaboration mit visuellen Helfern (Ganesan, Rathore, Ross & Amor, 2018).

Abbildung 7: Projektions-basierte Prototypen für Mensch-Roboter-Interaktion von (a) Gao und Huang und (b) Ganesan, Rathore, Ross und Amor.

2.2.3 Visualisierung über Head-Mounted Display

Head-Mounted Displays sind Anzeigegeräte, die wie eine Brille oder ein Visier am Kopf getragen werden und eine kleine Anzeigeoptik meist vor beiden Augen besitzen. Sie lassen sich in drei Kategorien unterteilen: Datenbrillen oder auch Smartglasses genannt, Augmented Reality Headsets/ Mixed-Reality Headsets und Virtual Reality Headsets. Alle drei Kategorien wurden im Zusammenhang mit Mensch-Roboter-Interaktion erforscht und bieten vielversprechende Ansätze.

(a) Datenbrille von Epson (Epson, 2019).

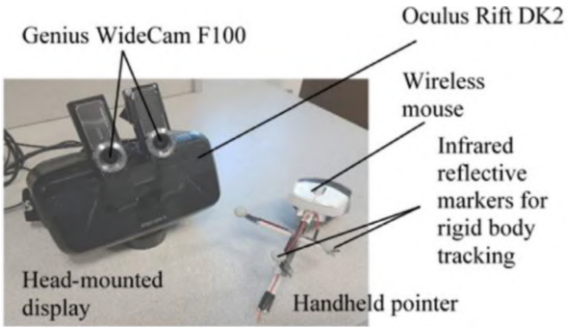


(b) Ansicht aus der Datenbrille vom sicheren (grün) und unsicheren (rot) Arbeitsbereich vom Roboter zur Unterstützung von Mensch-Roboter Kollaboration (Makris, Karagiannis, Koukas & Matthaiakis, 2016).

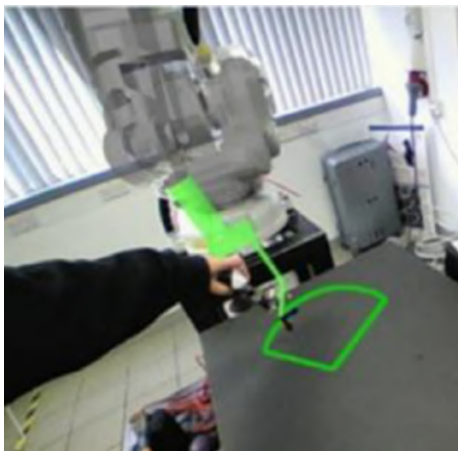
Abbildung 8: Datenbrille von Epson (a) und Ansicht aus der Datenbrille vom Augmented Reality System von Makris, Karagiannis, Koukas und Matthaiakis (b).

Datenbrillen sind eine Art von Brillen, die Informationen neben oder zudem, was der Benutzer sieht, hinzufügt. Dabei werden die Informationen nur monoskopisch, als zweidimensionale Bilder ohne Tiefeneindruck, dargestellt. Das bedeutet, dass die dargestellten Informationen als halbtransparente Fenster entweder direkt vor dem Sichtfeld des Benutzers oder in einer Ecke visualisiert werden. Ein bekanntes Beispiel für Datenbrillen sind Google Glass oder die Epson Moverio, die in Abbildung 8a zu sehen ist. Makris, Karagiannis, Koukas und Matthaiakis (2016) benutzten eine Datenbrille von Epson zusammen mit einer Android Smartwatch zur Unterstützung der Mensch-Roboter-Kollaboration. Das entwickelte Augmented Reality System zeigt dem Benutzer produktions- und prozessbezogene sowie sicherheitsrelevante Informationen, um das Sicherheitsgefühl und die Akzeptanz von Benutzern beim Arbeiten in der Nähe von großen Industrierobotern zu erhöhen. Dabei dient die Smartwatch als Eingabegerät, worüber der Benutzer mit dem Augmented Reality System interagiert. In Abbildung 8b ist zu sehen, wie die Datenbrille Informationen über den sicheren und unsicheren Arbeitsbereich des Roboters darstellt.

Darüber hinaus werden Texte eingeblendet, die Montageschritte erklären oder Produktinformationen anzeigen. Eine Fallstudie zeigte, dass das entwickelte Augmented Reality System zu einer verbesserten Integration des Benutzers in den Montageprozess führt und die Akzeptanz der Zusammenarbeit mit großen Industrierobotern ohne Schutzzäune erhöht (Makris et al., 2016). Die Limitationen des Systems liegen vor allem in der benutzten Technologie der Datenbrille. Obwohl die Abbildung 8b überzeugend wirkt, muss dabei beachtet werden, dass der Sichtbereich und die Auflösung des Bildes von der Datenbrille begrenzt sind und die Darstellung monoskopisch ist. Dies schränkt das Potenzial der Visualisierungen stark ein.



(a) Virtual Reality Headset erweitert durch zwei Weitwinkelkameras für die Darstellung und daneben der Zeiger als Eingabegerät (Ong, Yew, Thanigaivel & Nee, 2020).



(b) Anzeige des Pfades und farbliche Andeutung der Erreichbarkeit und Manipulierbarkeit vom Roboterarm (Ong, Yew, Thanigaivel & Nee, 2020).

Abbildung 9: Verwendetes Virtual Reality Headset sowie Eingabegerät (a) und Ansicht vom Virtual Reality Headset (b) des Prototyps von Ong, Yew, Thanigaivel und Nee.

Im genauen Gegensatz zur Datenbrille steht das Virtual Reality Headset. Dieses schirmt die Sicht des Benutzers von der realen Welt ab und projiziert die virtuelle Reality für den Benutzer als separate Bilder in jedes Auge. Dadurch entsteht eine stereoskopische Ansicht. Sie sind mit Stereoton und Bewegungssensoren, die die Rotation und Translation des Kopfes messen, ausgestattet. Dies erlaubt dem Benutzer, sich in der virtuellen Welt umzuschauen und zu bewegen. Außerdem besitzen sie ein deutlich größeres Sichtfeld, als Datenbrillen. Ong, Yew, Thanigaivel und Nee (2020) benutzten das Virtual Reality Headset von Oculus und einen in der Hand gehaltenen Zeiger, um die Roboterprogrammierung für Schweißarbeiten und Pick und Place Aufgaben zu vereinfachen und somit auch unerfahrenen Benutzern den Umgang damit zu ermöglichen. Sie verwenden ein Vir-

tual Reality Headset, das mit zwei Weitwinkelkameras, wie in Abbildung 9a dargestellt, erweitert wurde. Diese sind im ungefähren Augenabstand auf dem Virtual Reality Headset befestigt und ermöglichen dadurch eine stereoskopische Aufnahme der Umgebung, die vom Virtual Reality Headset wiedergeben wird. Virtuelle visuelle Helfer, wie ein virtueller Roboter, können die Anzeige des gewählten Pfades, sowie die Erreichbarkeit und Manipulierbarkeit des Roboterarms, über die Aufnahme stereoskopisch augmentieren. In Abbildung 9b ist der virtuelle Roboter, der über den realen Roboter augmentiert wird und dem Zeiger folgt, dargestellt. Außerdem wird der gezeichnete Pfad in grün visualisiert. Die Farbe des Endeffektors kennzeichnet die Manipulierbarkeit des Roboters in dieser Position. Der Benutzer kann sich frei im Arbeitsbereich des Roboterarms bewegen und mit dem Zeiger im realen Raum dreidimensional Punkte zum Planen eines Pfades definieren. Der virtuelle Roboter bewegt sich entlang des definierten Pfades und der Benutzer kann beobachten, ob der Roboter mit physischen Hindernissen in der Roboterzelle kollidieren würde. Das Planen und Simulieren in der tatsächlichen physischen Umgebung bieten daher große Vorteile gegenüber einer rein virtuellen Umgebung. Eine Studie mit dem Prototyp verifiziert, dass das System die Programmierung von Roboteraufgaben erheblich beschleunigt und sich der Bedarf an Anwenderkenntnissen verringert (Ong et al., 2020).

(a) Die Microsoft HoloLens (Microsoft, 2019a).

(b) Ansicht aus der HoloLens auf den simulierten zukünftigen Pfad des Roboterarms (Rosen et al., 2019).

Abbildung 10: Das verwendete Augmented Reality Headset (a) und die Ansicht durch die HoloLens (b) des Prototyps von Rosen et al.

Augmented Reality Headsets funktionieren ähnlich wie Datenbrillen, da der Benutzer durch sie hindurchsehen kann und virtuelle Objekte über die reale Sicht augmentiert werden. Der größte Unterschied zur Datenbrille ist die hochauflösende, stereoskopische sowie dreidimensionale Darstellung der Augmentierungen und das Sammeln von Umgebungsinformationen. Die Umgebung wird permanent durch integrierte Sensoren getrackt, was die Interaktion von virtuellen Objekten mit reellen Objekten ermöglicht. Zum

Beispiel kann dadurch ein virtuelles Objekt auf dem realen Tisch platziert werden und der Benutzer kann um den Tisch herumgehen und das virtuelle Objekt von allen Seiten betrachten. Die Größe des Sichtfelds von Augmented Reality Headsets ist größer, als die von Datenbrillen, aber noch deutlich kleiner als die von Virtual Reality Headsets. In Abbildung 10a ist ein bekannter Vertreter der Augmented Reality Headsets, die Microsoft HoloLens, dargestellt. Sie verbindet ein hochauflösendes stereoskopisches Display mit integrierten Sensoren, Lautsprechern und Rechnereinheit in einem kompakten Headset. Neben den Augmented Reality Eigenschaften verfügt die HoloLens über Gesten und Spracherkennung (Microsoft, 2019a). Krupke et al. (2018) haben ein Mixed Reality-Mensch-Roboter-Kollaborationssystem mit der HoloLens entwickelt, bei dem ein Mensch auf natürliche Weise mit einem co-lokalisierten Roboter Pick und Place Aufgaben ausführen kann. Dieses soll sowohl für Experten im Feld der Robotik, als auch für Anfänger ermöglicht werden. Die HoloLens wurde ausgewählt, da sie die Sicht auf die Realität nicht verdeckt und somit die Sicherheit, Akzeptanz und Vorhersehbarkeit bei der Mensch-Roboter-Interaktion verbessert. Sie soll die zukünftige Position visualisieren und somit dem Benutzer Feedback über die nächste Aktion geben. Damit der Benutzer mit dem Roboter interagieren kann, testen Krupke et al. mit ihrem System verschiedene Eingabemethoden. Die Erste ist das kopf-basierte Anvisieren, dabei bewegt der Benutzer mit Kopfbewegungen einen virtuellen Cursor. Wenn dieser sich mit dem gewünschten Objekt überschneidet, bestätigt er mit einem Sprachbefehl diese Position. Bei der zweiten Eingabemethode versucht der Benutzer das gewünschte Objekt mit dem Finger zu überlagern und bestätigt dann ebenfalls mit einem Sprachbefehl die Position. Eine Studie mit den unterschiedlichen Eingabemethoden ergab, dass die kopf-basierte Methode präziser ist und weniger als physische, zeitliche und mentale Anstrengung wahrgenommen wurde, als die finger-basierte Methode (Krupke et al., 2018). Rosen et al. (2019) haben einen ähnlichen Prototyp entwickelt, welcher sich mehr auf die Visualisierung und ihre Wirksamkeit konzentriert, als auf die Eingabemethode. Der Prototyp soll die Kollaboration mit einem co-lokalisierten Roboterarm verbessern und erlaubt dem Benutzer die Steuerung und Kommunikation mit ihm. Rosen et al. verwenden die HoloLens, um die beabsichtige, zukünftige Roboterbewegung über die reale Sicht des Benutzers auf den Roboter und seine Umgebung zu simulieren. Dies ist in Abbildung 10b dargestellt. Außerdem kann die Endposition des Endeffekors mit Hilfe von Handgesten manipuliert werden. Die Wirksamkeit dieses Systems gegenüber einer zweidimensionalen Visualisierung über einen Monitor und gegenüber keiner Visualisierung wurde mit einer Studie ermittelt. Bei dieser bestimmten 32 Teilnehmer, ob der Pfad vom Roboterarm, der mit der HoloLens, einem Monitor oder gar nicht visualisiert wurde, mit auf einen Tisch angeordneten Blöcken kollidiert oder nicht. Die Ergebnisse zeigen, dass der Benutzer mit der HoloLens die

Aufgabe schneller und genauer bestimmen kann, wohin sich der Roboter bewegen wird, als mit den anderen Methoden. Im Vergleich zur nächst besseren Methode hat sich die Genauigkeit um 15 Prozent gesteigert und die Zeit um 38 Prozent verkürzt (Rosen et al., 2019).

Chakraborti, Sreedharan, Kulkarni und Kambhampati (2017) kombinieren die HoloLens mit Elektroenzephalografie (EEG) als weiteres Eingabegerät in einen Prototyp für die Mensch-Roboter-Interaktion. Sie untersuchen damit eine alternative Kommunikationsform im besonderen Hinblick auf Sicherheit, Produktivität und flüssige Teamarbeit. Mit dem EEG kann der mentale Zustand des Menschen im Arbeitsbereich gemessen und an den mobilen Roboter weitergeben werden. Dadurch kann der Roboter auf den geistigen Zustand des Menschen reagieren. Zum Beispiel kann durch das EEG die hohe mentale Belastung des Benutzers als Reaktion auf ein kritisches Ereignis erkannt werden und der Roboter kann mit einem Stillstand reagieren. Dies ist ein weiterer Kommunikationskanal vom Menschen zum Roboter. Damit umgekehrt der Roboter auch mit den Menschen kommunizieren kann, werden visuelle Helfer eingesetzt. Diese zeigen unter anderem den aktuellen Arbeitsbereich des Roboters an. Wenn der Roboter sich aber nicht im Blickfeld des Benutzers befindet, erscheint ein Pfeil, der in Richtung Roboter deutet. Eine erste Studie weist darauf hin, dass dieser Ansatz erfolgversprechend ist. Weiter Studien dazu sollen in Zukunft folgen (Chakraborti et al., 2017).

2.3 Autonomie für Tetraplegiker

Tetraplegie ist eine durch Krankheit oder Verletzung verursachte Lähmung, die zum teilweisen oder vollständigen Funktionsverlust aller vier Gliedmaßen und des Rumpfes führt. Der Funktionsverlust umfasst sowohl die sensorischen als auch die motorischen Fähigkeiten. Das bedeutet, dass Betroffene sowohl die Empfindungen als auch die Kontrolle über ihre Gliedmaßen verlieren. Durch diese schwere Behinderung ist die Bewältigung des Alltags nicht ohne intensive Betreuung durch Pflegepersonal möglich. Jede Art der Zurückgewinnung von Autonomie bedeutet daher eine Steigerung der Lebensqualität. Dies gelingt durch Selbstbestimmung im Alltag, wie z. B. die Steuerung eines elektrischen Rollstuhls, bis hin zur Teilhabe am Arbeitsleben. Arbeit bildet die wesentliche Voraussetzung für ein selbstbestimmtes und gleichberechtigtes Leben in der Gesellschaft und hat daher einen besonders hohen Stellenwert nicht nur für Tetraplegiker, sondern generell für Menschen mit Behinderungen (Mensch eV, 2015).

2.3.1 Eingabegeräte für Tetraplegiker

Der Überbegriff Tetraplegie fasst unterschiedliche Ausprägungen von Rückenmarksverletzungen der Halswirbel und damit verschiedene Grade des Funktionsverlustes zusammen. Das Krankheitsbild ist individuell und abhängig von Ursache und Ausprägung der Rückenmarksverletzung. Zum Beispiel verlieren manche Tetraplegiker die Beweglichkeit ihrer Arme und Hände, jedoch bleibt ihnen die Funktionalität mancher Finger erhalten. Andere Tetraplegiker hingegen haben nicht nur den kompletten Funktionsverlust aller Gliedmaßen erlitten, sondern sind auch in ihrer Kopfbewegung beschränkt (Sabharwal, 2020). Durch diese Individualität entstanden unterschiedliche Eingabegeräte, die auf den Tetraplegiker personalisiert sind, um seine Möglichkeiten bestmöglich auszuschöpfen. Aufgrund ihrer schnellen Integration und einfachen Bedienung sind Joysticks und Gamepads der Standard unter den Eingabetechnologien. Sie müssen jedoch mit Händen, Armen und/ oder Fingern bedient werden und sind daher für Tetraplegiker meistens nicht verwendbar. Daher wurden viele verschiedene Eingabegeräte entwickelt, die sich an die individuellen Fähigkeiten von Tetraplegikern anpassen. Um die Vielfalt von Eingabegeräten zu verdeutlichen, werden im Folgenden Eingabegeräte zur Steuerung von elektrischen Rollstühlen kurz vorgestellt.

- Kopfbewegungen können als Eingabegerät verwendet werden. Koordinierte Bewegung des Kopfes geben die Richtung und Geschwindigkeit des Rollstuhls an. Dabei kann die Kopfbewegung durch unterschiedliche Technologien wahrgenommen werden. Durch eine Kopfstütze, die mit drei Näherungssensoren ausgestattet ist oder durch eine Kamera, die durch Gesichtserkennung den Kopf und dessen Ausrichtung wahrnimmt. Eine dritte Möglichkeit ist eine IMU, die am Kopf des Benutzers befestigt ist. Diese Kombination aus Beschleunigungssensor, Gyroskop und Magnetometer misst die Bewegung des Kopfes und kann dadurch seine Position und Orientierung bestimmen (Dymond & Potter, 1996; Mazo, 2001).

- Sprachbefehle werden mit einem Mikrofon aufgenommen und durch Spracherkennungssoftware ausgewertet. Die ausgewerteten Befehle, wie z. B. “Vorwärts“, „Stopp“ oder „Drehen“, ermöglichen die Steuerung eines Rollstuhles. Dabei sind Sprachbefehle ein einfaches Konzept, welches schnell zu erlernen ist. Jedoch ist die erfolgreiche Erkennungsrate bei Sprachbefehlen von großer Bedeutung. Laute Hintergrundgeräusche oder Akzente können diese Erkennungsrate verringern (Nishimori, Saitoh & Konishi, 2007).

- Bei der Elektrookulografie werden Elektroden um die Augen platziert. Diese messen die Position der Augen relativ zum Kopf. Dadurch können Augenbewegungen

gemessen und als Befehle, die zur Steuerung eines Rollstuhls dienen, interpretiert werden (Mazo, 2001).

- Eyetracking stellt einen weiteren Ansatz dar, um die Augen als Eingabegerät zu nutzen. Im Gegensatz zu Elektrookulografie wird hier ermittelt, was der Benutzter gerade mit den Augen fokussiert. Dafür werden Kameras, welche die Augen aufnehmen, an einer Brille montiert. Mit Hilfe von Bilderkennungsverfahren werden die Pupillen und ihre Bewegungen erfasst und der Fokus des Benutzers in fast Echtzeit ermittelt. Dies ermöglicht die Bedienung eines Computer-Interfaces, welches einen elektrischen Rollstuhl steuert. Indem der Benutzer z. B. Richtungspfeile in diesem Interface für eine bestimmt Zeit fokussiert oder durch bestimmte Augenbewegungen, die als Befehle dienen (Corno, Farinetti & Signorile, 2002).

- Brain-Computer-Interface (Gehirn-Computer-Schnittstelle) (BCI) ermöglicht eine Verbindung zwischen Gehirn und Computer ohne die Nutzung des peripheren Nervensystems, wie z. B. die Extremitäten. Dazu werden die elektrischen Aktivitäten des Gehirns mit invasiven oder nichtinvasiven Elektroden gemessen. Die invasiven Elektroden werden operativ unter die Kopfhaut implantiert. Sie liefern dadurch genauere Messergebnisse, als die nichtinvasiven Elektroden. Diese werden auf die Kopfhaut platziert und messen mittels EEG die Gehirnaktivität. Sie benötigen keine Operation und haben damit auch keine Nebenwirkungen. Außerdem reichen ihre Signale aus, um sie mittels Mustererkennung in Steuersignale umzuwandeln. Das System benötigt anfangs eine Kalibrierung, bei der der Benutzer die verschiedenen Befehle, wie z. B. "fahre gerade aus", denkt und das entstehende Signal dem Befehl zugeordnet wird. Ist die Kalibrierung abgeschlossen, kann der Benutzer den Rollstuhl durch die erlernten Befehle steuern (Jayabhavani, Raajan & Rubini, 2013; Wolpaw & Wolpaw, 2012, S. 5ff).

- Breath-Expulsion Sensoren (Atem-Ausstoß Sensoren) können sowohl die Stärke als auch die Richtung der Atmung erfassen, wodurch der Rollstuhl in seiner Geschwindigkeit und Fahrtrichtung fast in Echtzeit gesteuert werden kann (Mazo, 2001).

2.3.2 Tetraplegiker und Roboter

Der Fortschritt in der Entwicklung von assistiven Technologien, wie z. B. die elektrischen Rollstühle und ihre unterschiedlichen Eingabegeräte, haben einen großen Einfluss auf die Lebensqualität von Tetraplegikern. Trotzdem sind diese Menschen im Alltag sehr eingeschränkt und können alltägliche Aufgaben, wie z. B. das Öffnen von Türen oder Eingießen und Trinken eines Glases Wassers, nicht ohne Hilfe einer Pflegeperson bewältigen. Roboter bieten durch ihre vielfältige Einsatzweise und die Fähigkeit ihre Umwelt zu manipulieren, einen der vielversprechendsten Möglichkeiten den Mobilitätsverlust von Tetraplegikern auszugleichen. Diese Zurückgewinnung von Autonomie durch Roboter kann mit unterschiedlichen Ansätzen erreicht werden. Leichtbauroboterarme können auf elektrische Rollstühle montiert werden und dadurch bei alltäglichen Aufgaben helfen. Bei festen Arbeitsstationen, bei denen ein Roboterarm z. B. neben einem Tisch montiert ist, können spezielle oder komplexe Aufgabenabläufe unterstützt werden und somit auch zur Integration in das Arbeitsleben dienen. Ein anderer Ansatz verfolgt den Einsatz von mobilen Robotern, die sich autonom fortbewegen und Aufgaben erledigen können. Dadurch sollen sie Pflegepersonal entlasten oder teilweise ersetzen.

T. L. Chen et al. (2013) entwickelten solch einen mobilen Roboterassistenten, der Menschen mit starken körperlichen Einschränkungen die Selbstpflege, Bewältigung mancher Alltagsaufgaben, sowie soziale Interaktion ermöglichen soll. Dafür wurde der Allzweckroboter PR2 verwendet, welcher sich mobil fortbewegen und seine Umwelt mit Hilfe von zwei Roboterarmen manipulieren kann. In Abbildung 11 ist der PR2 abgebildet, wie er ein Handtuch aus einer Schublade holt. Der Roboter wird über ein zweidimensionales Interface mit Hilfe von Kopfbewegungen gesteuert. Dabei setzt das entwickelte System zur Steuerung des Roboters auf geteilte Autonomie. Das bedeutet, dass der Roboter autonom Aufgaben ausführen kann und der Benutzer nur bei Problemen einschreitet. Beispielsweise kann der Benutzer den Roboter jeder Zeit stoppen und dadurch verhindern, dass der Roboter etwas beschädigt, wenn er z. B. an einer Tischdecke hängen bleibt. Das System erlaubt einem Tetraplegiker, sich mit dem Roboter zu kratzen und zu rasieren oder ein Objekt in seinem Haus zu finden (T. L. Chen et al., 2013).

Autonome Systeme und künstliche Intelligenz werden nicht nur bei assistiven mobilen Robotern eingesetzt, sondern finden häufig im Zusammenhang mit anderen Robotern ihren Einsatz. Denn sie sind nicht nur in der Lage die Bedienung eines Roboters zugänglicher zu machen, sondern können auch die Komplexität von Aufgaben reduzieren. Ein allgemeiner Nachteil für die Bedienung von Robotern ist, dass sie mehr DOF als ihre

Steuerschnittstelle besitzen. Beispielsweise weißt ein normaler Joystick vier DOF auf, was die Steuerung eines Roboters mit sieben DOF verkompliziert. Dies führt zu einer schwierigeren Bedienbarkeit, welche gerade für Benutzer mit körperlichen Beeinträchtigungen ein Problem darstellt.

Abbildung 11: Der Allzweckroboter PR2, wie er gerade ein Handtuch aus einer Schublade holt (T. L. Chen et al., 2013).

D.-J. Kim et al. (2011) führten eine Studie durch, die die Steuerung eines Roboterarms mit einem manuellen und einem autonomen System verglich. Diese Studie wurde mit zehn Personen, die eine traumatische Rückenmarksverletzung erlitten und dadurch Einschränkungen der oberen Extremitäten besitzen, durchgeführt. Über drei Wochen lernten die Testpersonen mit dem Roboterarm Gegenstände aus einem Regal anzureichen. Während dieser Zeit steuerten sie den Roboter entweder manuell oder autonom. Bei der manuellen Steuerung wurde die kartesische Position des Endeffektors in Echtzeit gesteuert. Bei der autonomen Steuerung wählte die Testperson über einen Bildschirm das Objekt aus und das System führte die Bewegung und das Anreichen autonom durch. Nach drei Wochen waren beide Gruppen in der Lage, den Roboter zu steuern und sich die Objekte erfolgreich und in einer angemessenen Geschwindigkeit anzureichen. Dabei wurde festgestellt, dass die Benutzer des autonomen Modus die Aufgabe mit einem erheblich geringeren Aufwand ausführen konnten. Trotz des erhöhten Aufwands wies die Analyse eine höhere mittlere Zufriedenheit der Benutzer im manuellen Modus auf. Während nicht behinderte Personen die Autonomie des Roboters präferieren, ist das Verhältnis von behinderten Personen zum Roboter ein Anderes. Denn für diese Benutzergruppe ist der Roboter nicht nur ein Mittel, um sich Objekte anzureichen, sondern ein grundlegendes Werkzeug, das ihnen die Interaktion mit ihrer Umgebung erst ermöglicht und mit dem sie ihre kogniti-

ven Fähigkeiten ausleben können. Darüber hinaus kann ein autonomer Roboter schnell zu einer weiteren externen Hilfskraft werden, die die Arbeit für die behinderte Person übernimmt. Daher führt die geringere Abhängigkeit von einer menschlichen Pflegekraft nicht unbedingt zu einer entsprechend höheren Zufriedenheit, wenn die Interaktion mit dem Roboter nur sporadisch und aufsichtsbedingt ist (D.-J. Kim et al., 2011).

Daher besteht die Herausforderung eine Interaktionsmöglichkeit mit einem Roboter zu finden, welche eine zugängliche Steuerung zulässt, bei der der Nutzer allerdings möglichst viel Kontrolle beibehält. Forscher verfolgen daher den Ansatz, unterschiedliche Eingabemöglichkeiten zu kombinieren, um somit den Unterschied in der Dimensionalität des Eingabegerätes und Roboter auszugleichen.

Baldi, Spagnoletti, Dragusanu und Prattichizzo (2017) kombinieren in ihren Prototyp Kopfbewegung mit Elektromyografie (EMG) Signalen, um einen Leichtbauroboterarm, der auf einem Rollstuhl montiert ist, zu steuern. Dabei verwenden sie einen MARG Sensor (Magnetic, Angular Rate und Gravity), der die Kopfbewegung misst und EMG Sensoren, um die Muskelkontraktionen der Stirn zu erkennen. Diese Muskelkontraktionen ersetzen den Einsatz von Buttons oder Hebel und ermöglichen dem Benutzer innerhalb verschiedener Modi, wie Translation oder Rotation des Roboters, zu wechseln. Durch die Kopfbewegung wird die Position des Roboters gesteuert. Die beiden Sensoren können kompakt in einer Kappe, wie in Abbildung 12 dargestellt, angebracht werden. Eine LED gibt dem Benutzer Feedback über den aktuell ausgewählten Modus. Diese Steuerungsschnittstelle ist tragbar, unabhängig von stationärer Hardware, sowie einsetzbar in unstrukturierten Umgebungen mit unterschiedlichen Lichtverhältnissen. Eine Studie zeigte, dass Personen mit körperlichen Behinderungen den Prototyp bedienen können und eine Leistung, die mit der einer Joystick-basierten Steuerung vergleichbar ist, erreichen (Baldi et al., 2017).

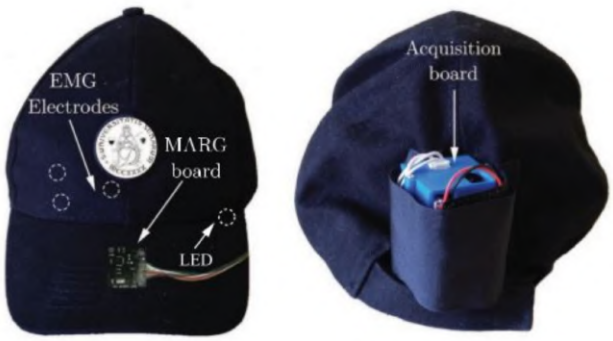


Abbildung 12: Tragbare Steuerungsschnittstelle mit einem MARG Sensor und drei EMG Sensoren zur Steuerung eines Leichtbauroboterarms (Baldi, Spagnoletti, Dragusanu & Prattichizzo, 2017).

Zeng et al. (2017) verfolgen einen Ansatz, der sich nicht nur auf die Kombination von Eingabegeräten konzentriert, sondern auch darauf, wie der Benutzer durch Feedback unterstützt werden kann. Sie kombinieren ein nicht invasives BCI mit Eyetracking, um einen Roboterarm an einer festen Arbeitsstation zu steuern. Dabei wird der Roboterarm über ein Interface auf einem Monitor bedient. Durch das Eyetracking wird die Position des Cursors gesteuert, dabei dient das BCI als Trigger-Befehl oder Mausklick. Eine Kamera, die den Arbeitsbereich des Roboterarms aufnimmt, ermöglicht das Anzeigen von AR-Feedback. Dieses besteht aus visuellen Helfern, wie der Anzeige der aktuellen Höhe des Grippers oder wie weit der Gripper geöffnet ist. Das System erlaubt dem Benutzter Pick und Place Aufgaben auszuführen und es zeigt, dass der Benutzter von den Informationen des AR-Feedbacks profitiert. Aus diesem Grund verbesserte sich die Effizienz und verringerte sich die kognitive Belastung (Zeng et al., 2017).

3 Konzeption des Prototyps

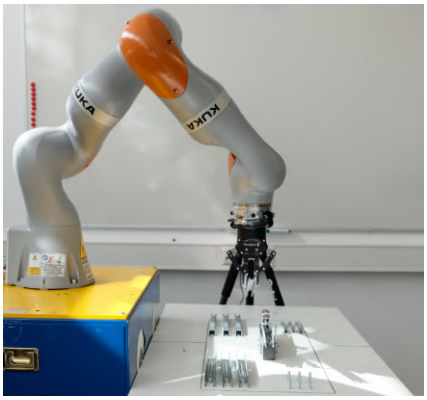
Im Rahmen dieser Masterarbeit wurde ein Prototyp entwickelt, der es Tetraplegikern oder körperlich behinderten Personen mit einem ähnlichen Krankheitsbild erlauben soll, Montageaufgaben zu meistern. Die physische Durchführung der Montageaufgaben soll ein Roboterarm umsetzen, der vom Tetraplegiker gesteuert wird. Der Fokus des Prototyps liegt in der bestmöglichen Unterstützung des Benutzers bei der Steuerung des Roboterarms. Im Folgenden wird die Konzeption des Prototyps erläutert und die wichtigsten Entscheidungen begründet.

3.1 Zentrale Anforderungen

Montageaufgaben besitzen spezifische Abläufe, die stark von dem zu produzierenden Teil abhängig sind. Um den Prototyp variabel zu halten und nicht speziell für nur eine einzige Montageaufgabe zu entwickeln, wurde der Montageprozess auf Pick und Place Aufgaben abstrahiert. Denn dies ist der Kern jeder Montageaufgaben: Ein Teil wird von einer Stelle, z. B. einem Förderband oder einer Kiste, genommen und passend zu einem anderen Teil platziert. Tetraplegiker oder Personen mit einem ähnlichen Krankheitsbild sind körperlich nicht in der Lage, Pick und Place Aufgaben auszuführen. Daher brauchen sie ein Werkzeug, das Befehle empfangen kann und die physische Umsetzung übernimmt. Ein Roboterarm, mit seiner Ähnlichkeit zum menschlichen Arm und vielfältigen Einsetzbarkeit, ist dafür sehr gut geeignet. Um weitere Variablen auszuschließen und sich auf die Ausführung von Pick und Place Aufgaben zu konzentrieren, ist der Roboterarm Teil einer festen Arbeitsstation und nicht mobil. Der Prototyp soll Tetraplegikern ermöglichen, einen Roboterarm an einer festen Arbeitsstation zu steuern, um Pick und Place Aufgaben ausführen zu können. In Abbildung 13 ist beispielhaft eine feste Arbeitsstation mit einem Roboterarm für die Montage eines Bett-Verstellbeschlags, dargestellt. Bei der Entwicklung des Prototyps, der den Roboterarm für solch eine Arbeitsstation steuert, müssen die speziellen Anforderungen, die Tetraplegiker mit sich bringen, berücksichtigt werden. So muss der Prototyp ohne die Verwendung von Händen steuerbar sein und Eingabegeräte verwenden, die von Tetraplegikern bedient werden können. Darüber hinaus wird der Benutzer eine statische Arbeitsposition einnehmen und dadurch eine feste Perspektive auf die Arbeitsstation haben. Solche eine feste Perspektive ist in Abbildung 13a zu sehen. Durch die frontale Ansicht auf die Arbeitsstation kann nicht erkannt werden, ob sich das Teil im Gripper direkt über den anderen Teilen befindet oder nicht. Erst in Abbildung 13b wird ersichtlich, dass sich der Gripper zu weit hinten befinden. Ein gesunder Benutzer könnte durch Aufstehen oder Neigen des Körpers seine Perspektive ändern, um so einen

besseren Eindruck zu bekommen, wie das zu platzierende Teil positioniert ist. Um dies zu kompensieren und die Bedienung zu vereinfachen, muss der Benutzer bei der Steuerung unterstützt werden. Die visuelle Wahrnehmung ist unser dominierender Sinneskanal, daher haben visuelle Helfer das größte Potenzial den Benutzer zu unterstützen (Posner, Nissen & Klein, 1976). Wie in Kapitel 2.2.2 und 2.2.3 aufgezeigt, stellt Augmented Reality einen vielversprechenden Ansatz dar, um bei der Steuerung von Robotern zu unterstützen und visuelle Helfer anzuzeigen. Denn AR ermöglicht die Realität mit virtuellen Helfern zu erweitern und so einen nahtlosen Übergang zwischen der realen Umgebung und virtuellen Objekten zu schaffen. So können virtuelle visuelle Helfer im Arbeitsbereich oder am Roboter direkt räumlich dargestellt werden. Im Gegensatz zu VR hat der Benutzer dabei immer den realen Roboter im Blickfeld, was die Sicherheit und das Vertrauen bei der Interaktion mit einem räumlich nahen Roboter erhöht.

(a) Frontale Ansicht einer beispielhaften Arbeitsstation für die Montage eines Bett-Verstellbeschlags.



(b) Seitliche Ansicht einer beispielhaften Arbeitsstation für die Montage eines Bett-Verstellbeschlags.

Abbildung 13: Ein beispielhafter Aufbau einer festen Arbeitsstation für die Montage eines Bett-Verstellbeschlags aus (a) frontaler und (a) seitlicher Ansicht. Die seitliche Ansicht macht deutlich, dass bei der frontalen Ansicht wichtige visuelle Informationen verloren gehen.

Virtuelle Helfer benötigen Informationen über das, was sie unterstützen sollen. Zum Beispiel kann die Position des Grippers in Verhältnis zum Benutzer nur dann verdeutlicht werden, wenn die Position des Grippers und des Benutzers bekannt sind. Je mehr Informationen über die Umgebung und dem Roboter vorhanden sind, desto mehr und bessere Helfer können angezeigt werden. Bei einer festen Arbeitsstation sind bestimmte Informationen, wie z. B. die Position des Roboters, statisch und können im Vorfeld gesammelt werden. Dynamische Informationen, wie die sich immer wieder verändernde Position des Grippers, Position und Blickrichtung des Benutzers oder die Position und Orientierung von Objekten im Arbeitsbereich des Roboters, müssen hingegen ste-

tig gesammelt werden. Über die Roboter Application Programming Interface (API) kann auf viele Informationen vom Roboter, wie z. B. die aktuelle Position des Endeffektors, zugriffen werden. Einige AR-Geräte sind in der Lage, Informationen über den aktuellen Zustand des Benutzers, wie z. B. die Blickrichtung oder Position, zu erkennen. Eine Herausforderung hingegen ist das Sammeln von Informationen über die handzuhabenden Objekte, denn diese Objekte müssen nicht nur klassifiziert, sondern auch ihre Position und Orientierung im Raum erkannt werden, um angemessene Hilfestellungen geben zu können. Solche 6D-Objekterkennungsmodule, die zuverlässig und schnell arbeiten, sind wichtige Komponenten moderner Bildverarbeitungssysteme für Anwendungen, wie mobile Robotermanipulation und Augmented Reality. Daher ist die Forschung in diesem Themenfeld sehr aktiv und jüngste Ergebnisse vielversprechend (Hinterstoisser, Lepetit, Rajkumar & Konolige, 2016; Hodan et al., 2017; Kehl, Manhardt, Tombari, Ilic & Navab, 2017). Jedoch ist eine flexible, robuste, schnelle und allgemeine Lösung noch nicht verfügbar. Unterschiedliche Ansätze haben verschiedene Vor- und Nachteile: Der Ansatz der Feature Extraction (Mustererkennung) ist ein schneller und robuster Ansatz, setzt aber voraus, dass die Objekte asymmetrisch geformt sind und eine strukturierte Oberfläche besitzen (K. Kim, Lepetit & Woo, 2012; Lepetit, Pilet & Fua, 2004; Lowe, 1999). Diese Eigenschaften treffen auf viele Objekte nicht zu. Vor allem industriell hergestellte Objekte, wie sie bei Montageprozessen üblich sind, besitzen meist eine symmetrische Form und texturlose Oberflächen. Template-basierte Lösungen hingegen haben keinen Anforderungen an das Erscheinungsbild der Objekte, jedoch reagieren diese empfindlich auf Objektverdeckung, eine unstrukturierte Umgebung und Lichtveränderungen (Brachmann, Michel, Krull, Ying Yang, Gumhold et al., 2016; Hinterstoisser et al., 2012; Tekin, Sinha & Fua, 2018). Der Ansatz basiert auf Deep Learning, was bedeutet, dass ein neuronales Netz auf die zu erkennenden Objekte trainiert werden muss. Dafür benötigt das Netz Daten, mit denen es trainiert werden kann. Diese Daten bestehen aus sehr vielen Bildern vom Objekt, auf denen die 6D Pose gekennzeichnet ist. Das Erstellen eines solchen Datensatzes ist mit großem Aufwand, Expertenwissen und einem komplexen Aufbau verbunden (Hodan et al., 2017). Aufgrund dieser Nachteile wurde eine Methode entwickelt, die mit synthetischen Bildern trainiert (Hinterstoisser et al., 2011; Kehl et al., 2017; Sundermeyer, Marton, Durner, Brucker & Triebel, 2018). Dabei wird aus einem 3D-Modell, welches das zu lernende Objekt repräsentiert, die benötigten Bilder gerendert und viel Zeit beim Erstellen des Datensatzes gespart. Jedoch bleibt eine lange Trainingszeit, die mehrere Tage in Anspruch nehmen kann und in der Praxis noch nicht zielorientiert ist. Insgesamt ist 6D-Objektposenerkennung in seiner Erkennungsrate noch nicht perfekt und durch seine lange Trainingszeit unflexibel im Hinblick auf das Integrieren von neuen Objekten. Andererseits könnten durch die Objekterkennung sehr effektive virtuelle Helfer generiert

werden. Zum Beispiel könnte angezeigt werden, ob der Gripper sich über einem Objekt befindet. Jedoch könnte durch ein fehlerhaftes Erkennen eines Objektes Misstrauen beim Benutzer entstehen und der hilfreiche Effekt entfallen. Darüber hinaus würde es, bedingt durch die lange Trainingszeit, mehrere Tage dauern, bis ein neues Objekt in den Montageprozess integriert werden kann. Dies würde die Arbeitsstation insgesamt sehr unflexibel machen.

Im Rahmen des MIA-Projektes wurde daher entschieden, zwei Ansätze zu verfolgen: Visuelle Helfer basierend auf Objekterkennung und visuelle Helfer, die ohne Objekterkennung funktionieren. Diese Masterarbeit konzentriert sich dabei auf den Ansatz ohne Objekterkennung. Der Ansatz mit Objekterkennung wird von Tim Dierks in seiner Masterarbeit „Visual Cues: Integration of object pose recognition with an augmented reality system as means to support visual perception in human-robot control“ bearbeitet.

3.2 Auswahl der Ein- und Ausgabetechnologien

Der Prototyp, der in dieser Masterarbeit entstanden ist, wurde für die Microsoft HoloLens und dem Kuka iiwa Roboterarm entwickelt. Für das Bedienungskonzept, welches ohne Hände funktioniert, wurden Kopfbewegung und Sprache als Eingabemethoden gewählt. Die Vor- und Nachteile dieser Technologien werden im Folgenden aufgezeigt.

3.2.1 Die Microsoft HoloLens

AR kann über unterschiedliche Geräte wiedergegeben werden, darunter sind Monitore, Projektoren, Smartphones oder Tablets und Head-Mounted Display (HMD). Für diese Masterarbeit wurde sich für die Microsoft HoloLens, welche in Abbildung 14a dargestellt ist, entschieden. Die HoloLens ist ein HMD und projiziert räumlich hochauflösende 3D-Inhalte direkt in das Sichtfeld des Benutzers. Im Gegensatz zu einer Darstellung auf Monitoren oder Smartphones/Tablets muss der Benutzer daher seine Aufmerksamkeit nicht zwischen einer Darstellung auf einem Display und der Realität aufteilen. Außerdem kann die HoloLens ohne den Einsatz von Händen bedient werden. Smartphone oder Tablet hingegen müssen vom Benutzer in der Hand gehalten werden. Die HoloLens ermöglicht im Gegensatz zu AR über Projektoren eine stereoskopische Darstellung der Augmentierung und sammelt mit Hilfe von mehreren Sensoren Informationen über die Umgebung. Dadurch müssen keine weiteren externen Sensoren, wie z. B. ein Mikrophon oder Kamera, gekauft und aufgestellt werden.

Mit Hilfe von vier Kameras und einer Tiefenkamera erkennt die HoloLens die Umgebung und kann sie räumlich wiedergeben. Dieses Spatial Mapping ermöglicht, die Bestimmung der Position des Benutzers im Raum. Zusätzlich verfügt die HoloLens über eine IMU, mit der die Kopfbewegung und damit die Blickrichtung des Benutzers erfasst werden. Vier Mikrophone ermöglichen eine Sprachsteuerung und zwei Lautsprecher einen räumlichen Klang. Dadurch lässt sich die HoloLens neben der angebotenen Gestensteuerung auch ohne den Einsatz der Hände, nur über Sprache und Kopfbewegungen, steuern. Nachteile der HoloLens sind das mit 579 Gramm hohe Gewicht, das den Tragekomfort negativ beeinflusst und das mit 30° x 17° relativ kleine Sichtfeld (Field of View (FOV)). Das kleine Sichtfeld hat den Nachteil, dass virtuelle Objekte oft abgeschnitten oder gar nicht mehr sichtbar sind. Daher können virtuelle Objekte nur aus einer gewissen Distanz in ihrer Vollständigkeit betrachtet werden. Die HoloLens ist nicht das einzige AR HMD auf dem Markt: Ein Gerät, das vergleichbare Eigenschaften besitzt, ist die Magic Leap. Sie bietet eine ähnliche Sensorik, wie die HoloLens, besitzt aber mit 40° x 30° ein größeres FOV. Zusätzlich ist sie mit 316 Gramm deutlich leichter. Im Gegensatz zur HoloLens ist die Recheneinheit nicht direkt in der Brille verbaut, sondern wird vom Benutzer z. B. in der Hosentasche getragen und über ein Kabel mit dem HMD verbunden (Magic Leap, 2020). Ende 2019 veröffentlichte Microsoft die HoloLens 2, welche in Abbildung 14b dargestellt ist. Auf dieser Abbildung ist die bessere Gewichtsverteilung und der damit einhergehende Tragekomfort zu erkennen. Außerdem besitzt sie mit einem FOV von 43° x 29° ein deutlich größeres Sichtfeld als die HoloLens 1 und unterscheidet sich damit wenig von der Meagic Leap. Zusätzlich verfügt die HoloLens 2 über Eyetracking und eine verbesserte Rechenleistung (Cooley & Moore, 2019; Heaney, 2019). Damit ist die HoloLens 2 der Stand der Technik für Augmented Reality Head-Mounted Displays. Da weder die HoloLens 2 noch die Magic Leap zum Zeitpunkt dieser Masterarbeit in der EU erhältlich waren, wurde auf die HoloLens der ersten Generation zurückgegriffen.

(a) Die Microsoft HoloLens Generation 1 (Kowalewski, 2016).

(b) Die Microsoft HoloLens 2 (Microsoft, 2020c).

(c) Die Magic Leap 1 (Magic Leap, 2020).

Abbildung 14: Die (a) Microsoft HoloLens, (b) Microsoft HoloLens 2 und (c) Magic Leap im Vergleich.

3.2.2 Eingabemethode über Sprache und Kopfbewegung

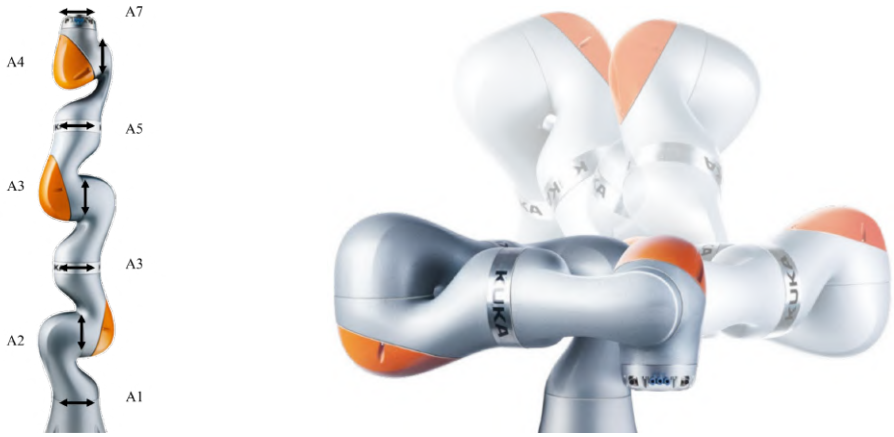
Als Eingabemethoden für den Prototyp wurden Sprache und Kopfbewegungen verwendet. Ein großer Vorteil besteht darin, dass die HoloLens neben der Gestensteuerung auch Sprache und Kopfbewegungen als etablierte Eingabemethoden anbietet. So kann der Benutzer den Prototyp ähnlich wie den Rest der HoloLens bedienen und muss keine neuen Eingabemethoden erlernen. Mit dem Kopf steuert der Benutzer einen Cursor, der sich immer in der Mitte seines Sichtfeldes befindet und mit virtuellen Objekten kollidiert. Fixiert der Benutzer mit dem Cursor z. B. einen Button und sagt den Sprachbefehl „Select“ kann er diesen auslösen. Darüber hinaus existieren auch Sprachbefehle, die kein Anvisieren von Objekten benötigt, wie z. B. die Befehle „go to start“ oder „Hey Cortana“. Da Sprache und Kopfbewegung Standardeingabemethoden der HoloLens sind, werden auch keine weiteren Sensoren, wie z. B. externe Mikrophone oder Kameras, benötigt. Dies spart Kosten und der Benutzer wird nicht durch weitere Sensoren, die an ihm befestigt werden, eingeschränkt. In der HoloLens 2 ist ein Eyetrackingsystem direkt verbaut. und für die HoloLens gibt es eine Erweiterung namens PupilLabs, bei der sich ein Eyetrackingsystem direkt an der Brille befestigen lässt (Pupil Labs GmbH, 2020). Jedoch stellte sich PupilLabs in vorangegangener Forschung als zu unpräzise für eine zuverlässige Eingabemethode heraus. Dies liegt unter anderem daran, dass sich eine vorangegangene Kalibrierung immer wieder aufgrund der beweglichen Teile der HoloLens und der Bewegung des Benutzers verstellt. Zusätzlich zur technischen Problematik hinsichtlich der Präzision ist anzumerken, dass Eyetracking als Eingabemethode für den Benutzer schnell anstrengend werden kann. Denn es zwingt ihn dazu Objekte, wie z. B. einen Button, für eine bestimmte Zeit mit dem Auge zu fixieren. Der Mensch bewegt seine Augen unbewusst. Das Ausführen von bewussten Augenbewegungen kann schnell unnatürlich und damit anstrengend werden. Qian und Teather und Hansen, Rajanna, MacKenzie und Bækgaard zeigen mit jeweils einer Studie auf, dass Kopfbewegung als Eingabemethode über Eyetracking von Probanden bevorzugt wird (Hansen et al., 2018; Qian & Teather, 2017).

Im Gegensatz zu den Eingabemethoden über EMG- oder EEG-Signalen benötigen weder Sprache noch Kopfbewegungen eine Kalibrierung oder Trainingsphase. Zusätzlich müssen bei der Bedienung über EMG- und EEG-Signalen Sensoren mit direktem Hautkontakt am Benutzer befestigt werden. Insgesamt macht dies die Eingabemethode über EMG- und EEG-Signalen umständlicher und zeitaufwändiger, als die über Sprache und Kopfbewegung. Jedoch haben auch Sprache und Kopfbewegung ihre Nachteile. Bei Spracherkennung ist der wichtigste Faktor die Erkennungsrate, mit der die gesprochenen Wörter erkannt werden. Diese nimmt stark ab, je lauter die Umgebungsgeräusche sind. Daher

eignet sich Spracherkennung eigentlich nicht für eine laute Umgebung. Jedoch kann mit einem gerichteten Mikrophon diesem Effekt entgegengewirkt werden (Mazo, 2001). Bei der Verwendung von Kopfbewegung als Eingabemethode kann es zur Erschöpfung des Nackens kommen. Daher ist bei der Umsetzung darauf zu achten, dass die Kopfbewegungen klein gehalten werden und der Benutzer nicht z. B. erst den Kopf stark rechts und direkt danach stark links bewegen muss.

3.2.3 Der Kuka LBR iiwa 7R800

Für dieses Projekt stand der Roboterarm Kuka LBR iiwa 7R800 (kurz Kuka iiwa) zur Verfügung. LBR steht für Leichtbauroboter und iiwa für intelligent industrial work assistant (intelligenter Industriearbeitsassistent). Der Kuka iiwa ist der Erste in Serie gefertigte sensitive Roboter, der für Mensch-Roboter-Kollaboration geeignet ist (KUKA AG, 2020). Er findet sowohl in der Forschung als auch in der Industrie für Fertigungsprozesse Einsatz.



(a) Kennzeichnung und Benennung der sieben Achsen vom Kuka iiwa.

(b) Nullraumbewegung des Kuka iiwas.

Abbildung 15: Der Kuka iiwa mit (a) Kennzeichnung und Benennung seiner sieben Achsen und (a) wie er eine Nullraumbewegung durchführt (KUKA AG, 2020).

Ähnlich zum menschlichen Arm besitzt der Kuka iiwa sieben Achsen. Dadurch ist er in der Lage, alle seine Achsen zu bewegen und dabei trotzdem die Position und Orientierung des Endeffektors beizubehalten. Eine solche Nullraumbewegung wird in Abbildung 15b verdeutlicht und ermöglicht eine hohe Flexibilität, da er dadurch Störobjekte einfach umfahren kann. In jeder seiner sieben Achsen besitzt der Kuka iiwa mehrere Sensoren, die die Temperatur, den Bewegungsbereich der Achse und das Gelenkmoment überwachen. Der Temperatursensor überwacht den thermischen Grenzwert der Elektronik und schaltet

den Kuka iiwa bei einem eventuellen Überschreiten sofort ab. Damit der Bewegungsbereich der einzelnen Achsen nicht überschritten wird, überwacht ein Achsenbereichssensor den Winkel der Achse. Der Bewegungsbereich und die Maximalgeschwindigkeit der einzelnen Achsen unterscheiden sich voneinander und sind in Tabelle 1 aufgelistet. Durch den eingeschränkten Bewegungsbereich in Kombination mit der abgerundeten Form besitzt der Kuka iiwa keine Klemm- oder Scherstellen. Die spezielle Form des Kuka iiwas kommt ohne Kanten oder eckige Stellen aus und ist in Abbildung 15a zu erkennen. Ein Gelenkmomentsensor in jeder Achse befähigt den Kuka iiwa zu einer Kontakterkennung und einer programmierbaren Nachgiebigkeit. Diese Gelenkmomentsensoren reagieren schon auf geringe Kräfte von außen und können bei einem unerwarteten Kontakt den Kuka iiwa sofort anhalten. Durch diesen integrierten Kollisionsschutz benötigt der Kuka iiwa keine weitere Sicherheitstechnik, wie z. B. einen Schutzzaun.

Achse	**Bewegungsbereich**	**Maximalgeschwindigkeit**
A1	$\pm 170°$	98°/s
A2	$\pm 120°$	98°/s
A3	$\pm 170°$	100°/s
A4	$\pm 120°$	130°/s
A5	$\pm 170°$	140°/s
A6	$\pm 120°$	180°/s
A7	$\pm 175°$	180°/s

Tabelle 1: Der Bewegungsbereich und die Maximalgeschwindigkeit jeder Achse des Kuka iiwa.

Bei einem Gewicht von 23,9 kg hat der Kuka iiwa eine Traglast von maximal 7 kg und eine Reichweite von 800 mm. Durch sein geringes Gewicht und die Sensoren ist er flexibel einsetzbar. Er benötigt keine weiteren Sicherheitsmaßnahmen und passt sich durch seine Wendigkeit an die Umgebung, wie z. B. beengte Räume, an. Mit einer Wiederholungsgenauigkeit von $\pm 0,1$mm ist der Kuka iiwa auch für komplexe Montageaufgaben geeignet (KUKA AG, 2020). Insgesamt eignet sich der Kuka iiwa durch seine Sicherheit, Flexibilität und seine Genauigkeit sehr gut für die Umsetzung des Prototyps. Damit der Kuka iiwa auch Montageaufgaben durchführen kann, wurde als Endeffektor der Robotiq 2F-85 Gripper montiert. Dieser verfügt über zwei Finger mit einer Spannweite von

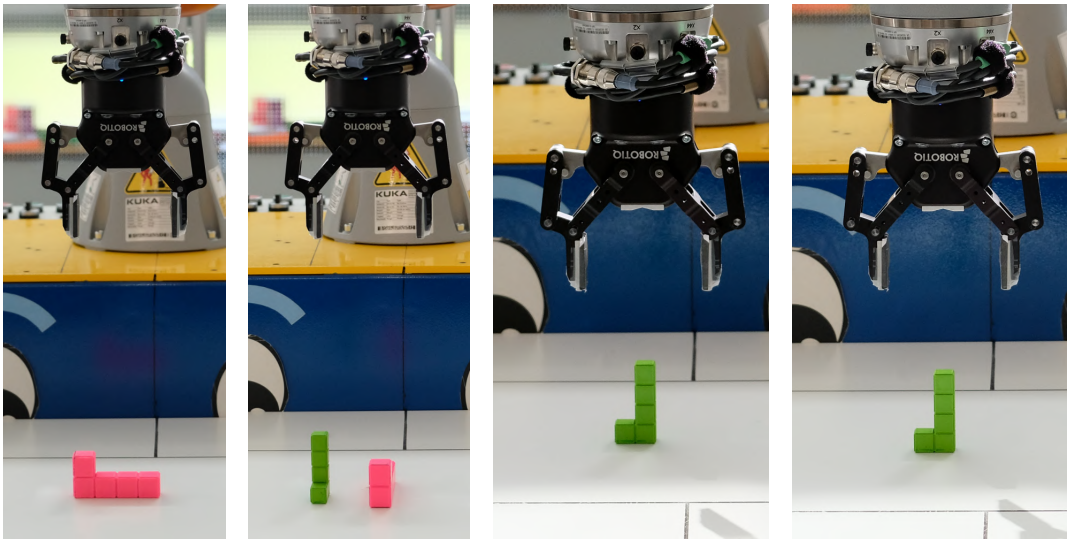
85 mm. Seine spezielle Bauweise ermöglicht ein paralleles innen und außen, sowie ein formschlüssiges Greifen. Diese unterschiedlichen Griffarten sind in Abbildung 16 dargestellt und ermöglichen ein sicheres Greifen auch von komplexen Objekten. Durch eine integrierte Griff- und Maßvalidierung ist der Gripper in der Lage, automatisch genügend Druck auszuüben, um ein Objekt zu greifen und festzuhalten. Darüber hinaus kann der Gripper Rückmeldung über seine interne Position, wie z. B. über seine momentane Öffnungsweite, geben (Robotiq, 2020).

Abbildung 16: Innen, außen und formschlüssiges Greifen vom Robotiq 2F-85 Gripper (Bouchard, 2014).

3.3 Anforderungen an die visuellen Helfer

Die Steuerung des Kuka iiwas ohne Visualisierungen basiert darauf, dass der Benutzer nur zwei visuelle Anhaltspunkte besitzt: Den Gripper und das Objekt, mit dem er interagieren möchte. Wenn der Benutzer den Roboter steuert, um beispielsweise ein Objekt zu greifen, so nähert er sich dem Objekt mit dem Gripper langsam an, um das Verhältnis, in dem der Gripper zum Objekt positioniert ist, besser wahrzunehmen. Dieses Annähern besteht aus mehreren Wiederholungen von horizontaler Positionierung des Gripper mit darauffolgender vertikaler Positionierung. Denn je näher sich Gripper und Objekt sind, desto präziser kann der Gripper positioniert werden. Das langsame Annähern an ein Objekt ist schon bei einfachen Aufgaben mühselig und zeitaufwändig. Daher besteht eine Anforderung an die visuellen Helfer, dass der Gripper mit Objekten in ein Verhältnis gebracht werden kann, ohne dass sich dieser erst annähern muss. Die Schwierigkeit bei der Positionierung des Grippers ohne Annäherung ist in den Abbildungen in 17 veranschaulicht. Dabei lässt sich die Problematik in zwei Kategorien aufteilen: Die Abbildungen 17a und 17b beleuchten die Schwierigkeit beim Positionieren des Grippers nach links oder rechts. Vor allem in Abbildung 17a stellt sich die Frage, ob der Greifbereich des Grippers überhaupt groß genug ist, um das Objekt in dieser Ausrichtung greifen zu können und wenn ja, ob dieser auch mittig über dem Objekt platziert ist. In Abbildung 17b wird

gezeigt, dass bei der Positionierung des Grippers auch Objekte mitberücksichtigt werden müssen, die nicht gegriffen werden sollen, sich aber in unmittelbarer Nähe befinden. Um Gripper und Objekt in ein Verhältnis zu bringen, würden in diesem Fall Linien von den Gripperfingern und der Mitte des Grippers zum Objekt helfen. Diese würden auch die Dimension des Grippers verdeutlichen, und ob dieser mit umliegenden Objekten kollidieren könnte. In Abbildung 17c und 17d sind die Schwierigkeiten bei der Positionierung des Grippers nach vorne und hinten dargestellt. Zwischen den Abbildungen 17c und 17d ist kaum ein Unterschied zu erkennen, obwohl in Abbildung 17d der Gripper so weit hinter dem Objekt ist, dass er es bei einem senkrechten Herunterfahren nicht greifen könnte. An diesen Abbildungen wird deutlich, dass vor allem die Tiefenwahrnehmung aus dieser frontalen Perspektive stark eingeschränkt ist. Virtuelle Linien oder ein virtuelle Repräsentation des Grippers würden in diesem Fall nicht helfen, da die Augmentierung über dem realen Objekt dargestellt wird. Wenn eine virtuelle Repräsentation des Grippers hinter dem Objekt platziert ist, wäre dies aus frontal Ansicht nicht zu sehen, da der Gripper das Objekt verdeckt und nicht anderes herum.



(a) Kollidiert mit pinkem Objekt.

(b) Kollidiert mit grünem Objekt.

(c) Über dem Objekt.

(d) Hinter dem Objekt

Abbildung 17: Schwierigkeiten bei der Positionierung des Grippers nach vorne/hinten (siehe (c) und (d)) sowie nach links/rechts (siehe (a) und (b)).

Ein erster Ansatz zur Kompensation der eingeschränkten Tiefenwahrnehmung, war die Integration einer Seitenansicht auf die Arbeitsstation. Die Seitenansicht würde von einer Kamera aufgenommen, die ihr Bild in Echtzeit an die HoloLens sendet. Auf der HoloLens wäre das Bild der Kamera auf einer Fläche, die neben der Arbeitsstation platziert

ist, zu sehen. Dieser Ansatz wäre technische umsetzbar, besitzt jedoch mehrere Nachteile. Da die HoloLens ein kleines Sichtfeld besitzt, wäre es nicht möglich, das Bild der Kamera und gleichzeitig die Arbeitsstation zu sehen. Außerdem kann es nicht irgendwo innerhalb der Arbeitsstation platziert werden, da es die Objekte und/oder den Roboterarm überdecken würde. Daher müsste das Kamerabild außerhalb der Arbeitsstation platziert werden. Das hätte zur Folge, dass der Benutzer ständig seine Aufmerksamkeit zwischen Kamerabild und Roboter hin und her wechseln müsste. Zusätzlich müsste er sich in eine andere Perspektive hineindenken. Dadurch hätte die Verwendung von AR keinen Vorteil mehr, es könnte genauso gut ein Monitor verwendet werden, der das Kamerabild wiedergibt. Der Monitor wäre dafür sogar geeigneter, da die HoloLens durch ihren kleinen Sichtbereich und die geringe Auflösung eingeschränkt ist. Das Kamerabild müsste auf der HoloLens mindestens 2 m vom Benutzer entfernt sein und wäre mit der Auflösung der HoloLens von 720p nicht hochaufgelöst (Microsoft, 2019b). Dies würde es schwierig machen, feine Positionsunterschiede des Grippers zu erkennen. Aufgrund dieser Nachteile wurde diese Idee verworfen und erörtert, wie das Blickfeld des Benutzers auf den Roboterarm mit visuellen Informationen angereichert werden kann, ohne die Aufmerksamkeit vom Roboter und der Realität abzulenken.

Ein zweiter Ansatz zur Verbesserung der Tiefenwahrnehmung, bestand in der Idee eines virtuellen Gitters, welches über den gesamten Arbeitsbereich augmentiert wird. Auf diesem Gitter könnte die aktuelle Position des Grippers durch z. B. Farben hervorgehoben werden, um somit die Position des Grippers im Vergleich zu Objekten zu verdeutlichen. Dieser Ansatz wurde nach der prototypischen Umsetzung direkt verworfen, da das virtuelle Gitter alles im Arbeitsbereich überlagerte, obwohl es halbtransparent und grobmaschig war. Objekte waren unter dem Gitter kaum zu erkennen, da die Augmentierung des Gitters die Realität zu stark überlagerte. Dies brachte die Erkenntnis, dass kein virtuelles Element über den gesamten Arbeitsbereich augmentiert werden kann. Die virtuellen Objekte müssen sich unauffällig in das Gesamtbild einfügen, um die Realität nicht zu sehr zu überdecken.

Insgesamt lassen sich daher folgende Anforderungen für die visuellen Helfer zusammenfassen: Sie müssen den Benutzer helfen, die Position des Grippers in ein Verhältnis mit Objekten zu bringen und dabei vor allem bei der Tiefenwahrnehmung zu unterstützen. Außerdem sollte die Dimension des Grippers hervorgehoben werden, um ein Verhältnis auch zu umliegenden Objekten zu ermöglichen. Dabei sollen die visuellen Helfer nicht die Aufmerksamkeit vom Roboterarm und der Realität ablenken. Sie sollen sich unauffällig in das Gesamtbild einfügen und die Realität nicht zu stark überlagern.

3.4 Architekturbeschreibung

Der Prototyp besteht aus zwei grundlegenden Komponenten: Dem User Interface auf der HoloLens und der Robotersteuerung des Kuka iiwa. In Abbildung 18 sind diese Komponenten und ihr Zusammenspiel dargestellt. Sie sind über ein User Datagramm Protocol (UDP) miteinander verbunden und tauschen Informationen, sowie Befehle miteinander aus. Das User Interface auf der HoloLens wurde in C# programmiert und erlaubt dem Benutzer die Steuerung des Roboters. Dazu kann der Benutzer die Position und Orientierung des Endeffektors bestimmen und mittels UDP an die Robotersteuerung senden. Analog wird der Befehl zum Schließen und Öffnen des Grippers übermittelt. Die Robotersteuerung ist eine in Java geschriebene Anwendung auf dem Kuka iiwa und steuert sowohl den Roboterarm als auch den Gripper. Dazu empfängt sie Befehle der HoloLens und setzt sie um. Darüber hinaus gibt die Robotersteuerung Rückmeldung über den aktuellen Zustand des Roboters oder des Grippers. Dadurch erhält das User Interface Informationen darüber, ob sich der Roboterarm im Moment bewegt oder eine Bewegung abgeschlossen hat. Zusätzlich können Informationen über die Öffnungsweite des Grippers oder das Gelenkmoment ausgetauscht werden. Über die Öffnungsweite des Grippers kann erfasst werden, ob der Gripper etwas gegriffen hat und über das Gelenkmoment kann festgestellt werden, ob und wie viel Kraft auf den Endeffektor ausgeübt wird. Die Zustandsinformationen der Robotersteuerung werden vom User Interface interpretiert und für den Benutzer visuell aufbereitet.

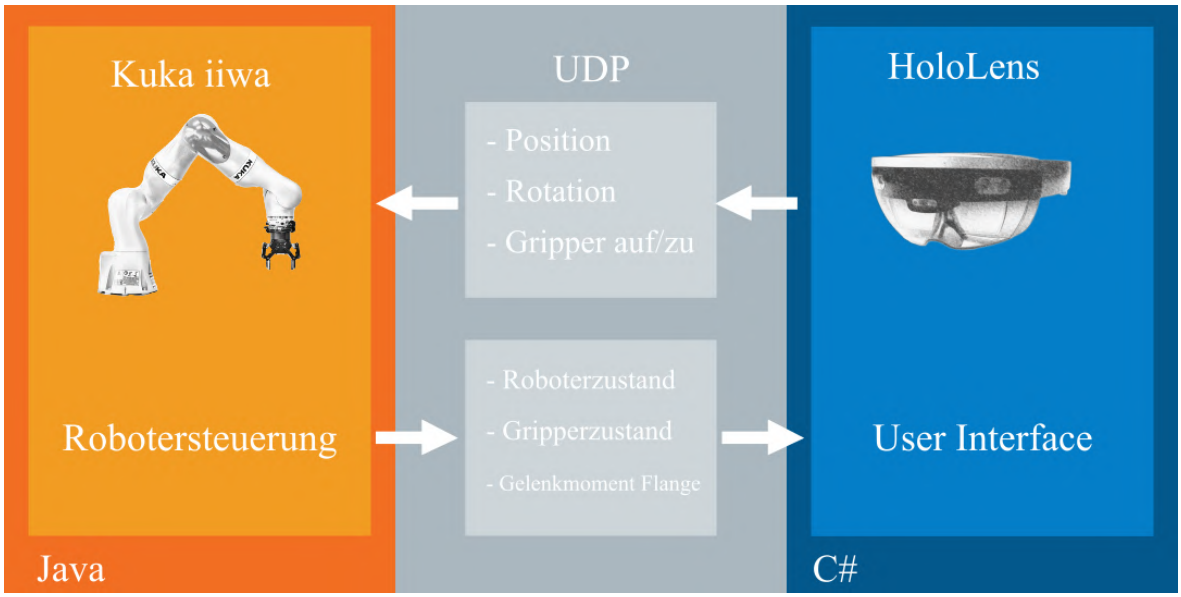


Abbildung 18: Architektur des Prototyps.

3.5 Vorgehen

Der im Rahmen dieser Masterarbeit entstandene Prototyp wurde in einem iterativen Prozess entwickelt. Dabei wurden Ideen und Konzepte prototypisch umgesetzt und von Mitarbeitern innerhalb des MIA-Projektes getestet. Durch das erhaltene Feedback wurden Ideen und Konzepte ausgearbeitet, verändert oder verworfen. Zwischenstände des Prototyps konnten mit weiteren Kundigen aus dem Bereich der Mensch-Roboter-Interaktion getestet werden. Durch dieses Feedback entstand eine frühe erste Version des Prototyps, der in einer ersten Studie getestet werden konnte. Im Folgenden wird diese Version des Prototyps als *Version für die erste Studie* referiert. Die Studie fand im Rahmen des MIA-Projektes statt und umfasste 27 Probanden. Aufgrund des frühen Status des Prototyps, wurden in dieser Studie nur Pick-Aufgaben getestet. Dabei konnten die Probanden die wichtigsten visuellen Helfer testen. Der Down-, Pick/Place- und Schließ-Indikator (s. Kapitel 4.4.2), sowie das Kraft-Feedback (s. Kapitel 4.4.5) waren zu diesem Zeitpunkt noch nicht entwickelt. Zu dem Laserpointer (s. Kapitel 4.4.1), dem Gripper-Bereich (s. Kapitel 4.4.3) und dem Farbschema (s. Kapitel 4.4.4) konnten die Probanden unter anderem in einem Interview ihre Meinung mitteilen. Basierend auf diesem Feedback wurden Veränderungen und Anpassungen am Prototyp vorgenommen, die vor allem die visuellen Helfer betrafen. Da die Studie im Rahmen des MIA-Projektes stattfand und nicht im Rahmen dieser Masterarbeit, wird hier nicht auf Details zur Durchführung oder Auswertung eingegangen. Jedoch hatte das Feedback der Probanden maßgeblichen Einfluss auf die Weiterentwicklung des Prototyps. Daher werden in Kapitel 4.4.3 und 4.4.4 die Änderungen, die auf dieser Studie basieren, erklärt und begründet.

Da diese Masterarbeit im Rahmen des MIA-Projektes stattfand, wurde die Konzeption in enger Zusammenarbeit mit Stephanie Arévalo, einer angestellten Doktorandin im MIA-Projekt, und Tim Dierks durchgeführt. Da Tim Dierks die Interaktion mit dem Roboter (siehe Kapitel 4.3) ebenfalls für seine Masterarbeit benötigte, entstand die Implementation der Interaktion in Zusammenarbeit mit Tim Dierks. Die Planung und Durchführung der Studie in Kapitel 5 wurde zusammen mit Stephanie Arévalo und Tim Dierks umgesetzt.

4 Implementierung des Prototyps

Für den im Rahmen dieser Masterarbeit entstanden Prototyp wurde eine Robotersteuerung für den Kuka iiwa und ein User Interface für die Microsoft HoloLens entwickelt. Die Robotersteuerung wurde mit der Programmierumgebung Kuka Sunrise Workbench Version 1.16.1.9 in Java entwickelt. Die Sunrise Workbench ist eine von Kuka entwickelte Programmierumgebung, die dazu ausgelegt ist, Anwendungen für Kuka Roboter zu programmieren. Das User Interface wurde in der Spiel-Engine Unity 2018.2.18f1 in C# umgesetzt. Unity im Zusammenspiel mit dem von Microsoft entwickelten HoloToolkit Plugin ist der von Microsoft empfohlene Weg, um schnell und effizient Applikationen für die HoloLens zu entwickeln (Microsoft, 2020a). Für diesen Prototyp wurde das HoloToolkit 2017.2.1.4 verwendet (Microsoft, 2020d).

Abbildung 19: Roboter, Plattform auf der der Roboter befestigt ist und der Tisch bilden die Arbeitsstation. Am Tisch ist ein roter Marker befestigt, der die Position der Arbeitsstation markiert. Der Benutzer trägt die HoloLens und sitzt frontal vor der Arbeitsstation. Virtuelle Objekte werden am, sowie unter dem Gripper angezeigt.

In Abbildung 19 ist der Aufbau der Arbeitsstation zu erkennen. Der Roboterarm ist vor einem Tisch aufgebaut, auf welchem die Montageaufgaben innerhalb des Arbeitsbereiches stattfinden. Der Marker, der am Tisch befestigt ist, markiert die Position der gesamten Arbeitsstation. In Kapitel 4.2 wird die Funktion des Markers genauer erläutert. Der Benutzer sitzt frontal vor dem Tisch und steuert den Roboter mit der HoloLens über die entwickelte Applikation. Wie in Abbildung 19 zu erkennen ist, befinden sich virtuelle

Objekte direkt am oder unter dem Gripper. Diese dienen entweder zur direkten Steuerung des Roboters oder sind visuelle Helfer, die bei der Steuerung unterstützen sollen. In Zusammenarbeit mit Tim Dierks ist ein Video (siehe https://www.youtube.com/watch?v=UAVUTd8l_EE oder Anhang B.6) entstanden, welches den Prototyp und seine Funktionen in Aktion zeigt. Dieses Video gibt einen ersten Eindruck über die Umsetzung der Interaktion und der visuellen Helfer. Im Folgenden wird die Umsetzung und Implementierung der einzelnen Bestandteile des Prototyps genauer erläutert.

4.1 Robotersteuerung

Die programmierte Robotersteuerung ist eine Anwendung, die auf der Recheneinheit des Kuka iiaws, dem Sunrise Cabinet, ausgeführt wird. Ebenfalls auf dem Sunrise Cabinet läuft ein Hintergrundprozess, der den Gripper steuert (ELZo3, 2019). Die Robotersteuerung kontrolliert nicht nur den Roboterarm, sondern kommuniziert auch mit diesem Hintergrundprozess, um den Gripper zu steuern und auszulesen. Über eine UDP-Verbindung empfängt die Robotersteuerung Nachrichten, die entweder den Zustand des Roboterarms abfragen oder Bewegungsbefehle für den Roboterarm oder Gripper geben. Zum Beispiel enthält ein empfangener Bewegungsbefehl die Koordinaten der Zielposition, die der Gripper einnehmen soll. Diese Zielposition wird angefahren, indem die interne inverse Kinematik die Achsenpositionen so berechnet, dass der Gripper sich linear von der Startposition zur Zielposition bewegt. Ist die Zielposition erreicht, sendet die Robotersteuerung eine Bestätigung an den Absender des initialen Bewegungsbefehls zurück. Ähnlich wie diese diskrete Bewegung wird eine kontinuierliche Bewegung ausgeführt. Bei einem kontinuierlichen Bewegungsbefehl werden konstant Zielpositionen gesendet, die der Roboterarm als interpolierte Bewegungen anfährt. Neben einer Zielposition, die der Gripper anfahren soll, kann auch eine Rotation um die Z-Achse der letzten Achse (A7) des Roboterarms ausgeführt werden. Beim Befehl zum Öffnen oder Schließen, müssen keine weiteren Daten mit gesendet werden. Denn der Gripper schließt und öffnet sich bis zum Maximum oder stoppt automatisch, wenn er eine bestimmte Kraft ausübt. Ist das Schließen oder Öffnen des Grippers abgeschlossen, wird eine Bestätigung an den Absender mit der prozentualen Öffnungsweite zurückgesendet. Aus dieser Öffnungsweite lassen sich Rückschlüsse ziehen, ob der Gripper etwas gegriffen hat und wie breit dieses Objekt ist. Neben diesen Bewegungsbefehlen, gibt es einen Befehl, der die externe Kraft bzw. das Drehmoment, die auf den Gripper in X-, Y-, oder Z-Richtung einwirken, zurückgibt.

Während die Anwendung läuft, ist immer eine Sicherheitstabelle aktiv. In dieser lassen sich Sicherheitsrichtlinien definieren, die immer eingehalten werden. Unter anderem ist darin definiert, dass der Roboterarm sich mit einer maximalen Geschwindigkeit von 800 mm/s bewegen darf und dass jede Achse ein externes Drehmoment von 20 Nm nicht überschreiten darf. Dies bewirkt einen sofortigen Stillstand des Roboterarms bei einer zu schnellen Bewegung oder Kollision. Das Drehmoment von 20 Nm ist relativ gering gewählt, so können bei einer Kollision mit einem Menschen keine gefährlichen Verletzungen entstehen. Werden die Werte in der Sicherheitstabelle überschritten, kommt es zu einem automatischen Sicherheitsabbruch und die Robotersteuerung wird unterbrochen. Dies hat zur Folge, dass der Roboter sich nicht mehr bewegt, bis die Robotersteuerung manuell neugestartet wurde.

4.2 Darstellung der Hologramme

Die virtuellen Objekte, die die HoloLens rendert und stereoskopisch darstellt, werden Hologramme genannt. Sie können vom Benutzer in dessen realen Umgebung platziert werden. Dadurch entsteht ein sogenannter spatial Anker (räumlicher Anker), der die Position des Hologramms im Raum fixiert. Damit behält das Hologramm seine Position, auch wenn der Benutzer sich bewegt oder das Hologramm nicht mehr im Sichtfeld ist. Die HoloLens ermöglicht dies, indem sie ihre Umgebung stetig mit ihren Sensoren einscannt und ihre Position im realen Raum berechnet. Es ist wichtig, dass dieses Tracking und die damit verbundene Positionierung des Hologramms möglichst genau ist. Für das menschliche Auge ist es sehr einfach, Abweichungen von virtuellen Objekten im Vergleich zu realen Objekten von nur wenigen Millimetern zu erkennen (Billinghurst, Clark, Lee et al., 2015). Vassallo, Rankin, Chen und Peters haben die Stabilität der Hologramme untersucht und einen mittleren Verschiebungsfehler von 5.83 $\pm$0.51 mm gefunden (Vassallo et al., 2017). Dies reicht für die Illusion, dass virtuelle Inhalte in der realen Welt fixiert sind, aus. Für den Prototyp ist diese Eigenschaft ausschlaggebend, da die Hologramme direkt am oder unter dem Gripper dargestellt werden sollen. Sie sollen dem Benutzer Orientierung und Hilfestellung geben. Daher ist es wichtig, dass die Hologramme korrekt und stabil über den gesamten Zeitraum der Bedienung angezeigt werden. Die HoloLens ist dazu grundlegend in der Lage, jedoch muss bei der Implementierung der Applikation auf wichtige Punkte geachtet werden. Ein wichtiger Punkt für die Hologrammstabilität besteht darin, dass die Applikation mit stabilen 60 Frames Per Second (FPS) ausgeführt wird. Bei der Programmierung wurde daher sehr auf die Performance geachtet. Darüber hinaus dürfen die Hologramme nicht zu nah oder zu weit vom Nutzer entfernt platziert werden. Die optimale Entfernung liegt zwischen 1,25 und 5 m (Microsoft, 2020b). Ein

weiterer wichtiger Punkt ist der Augenabstand, auch Interpupillardistanz (IDP) genannt. Dieser muss für jeden Benutzer, noch bevor die Applikation geöffnet wird, bestimmt und eingestellt werden. Dafür stellt Microsoft eine Applikation zur Verfügung. Wenn der IDP nicht auf den Benutzer angepasst wird, können Hologramme instabil erscheinen oder eine falsche Entfernung suggerieren.

Abbildung 20: Ursprung des Roboterkoodinatensytems und des HoloLens-Koordinatensytems. Der Abstand vom Marker zum Ursprung des Roboterkoodinatensytems ist in Gelb markiert. Rot, Grün, Blau entspricht der x-, y-, z-Achse.

Damit Hologramme am Gripper dargestellt werden können, benötigt die HoloLens die Position des Grippers. Die Robotersteuerung kennt diese Position, jedoch nur im Koordinatensystem des Roboters. Damit die Position in das Koordinatensystem der HoloLens umgerechnet werden kann, wird die Position des Koordinatenursprungs des Roboters benötigt. Im Gegensatz zum Roboter wird der Ursprung des Koordinatensystems der HoloLens dynamisch und abhängig von der Position und Orientierung während des Startvorgangs der HoloLens festgelegt. Daher muss direkt nach dem Start der Applikation des Prototyps eine kurze Kalibrierungsphase, bei der die Position der Arbeitsstation erfasst wird, stattfinden. Die Position der Arbeitsstation reicht für die Berechnung des Ursprungs des Roboterkoordinatensystems aus, da sie statisch ist und zuvor der Abstand vom Marker zum Ursprung ausgemessen wurde. In Abbildung 20 sind dieser Abstand in Gelb, sowie die Koordinatensysteme des Roboters und der HoloLens zu erkennen. Während dieser Kalibrierungsphase wird ein 2D-Marker mit der integrierten Kamera der HoloLens erfasst. Mit Hilfe der kamerabasierten Markererkennung von Vuforia (PTC, 2020) werden die Position und Orientierung des Markers im Raum ermittelt. Um die Präzision zu stei-

gern, dauert diese Kalibrierungsphase fünf Sekunden. Während dieser Zeitdauer werden Position und Orientierung des Markers für jeden Frame gespeichert. Nach den 5 Sekunden werden die mittlere Position und Rotation aus diesen Werten berechnet. Erst aus diesen gemittelten Werten wird die Position der Arbeitsstation bestimmt. Dadurch werden Erkennungsfehler von Vuforia, die z. B. entstehen, wenn der Benutzer seinen Kopf schnell bewegt, ausgeglichen. Um die Präzision zusätzlich zu erhöhen, wird das Vorwissen, dass die Arbeitsstation waagerecht steht, verwendet. Daher muss nur die Rotation um die y-Achse bestimmt werden. Dies reduziert die Freiheitsgrade der Markererkennung von 6 DOF auf 4 DOF und damit den Einfluss von Fehlern bei der Erkennung. Diese Maßnahmen erlauben eine präzise Erkennung der Position der Arbeitsstation und die Berechnung des Ursprungs des Roboters. Dadurch kann die Position des Grippers vom Koordinatensystem des Roboters in das Koordinatensystem der HoloLens umgerechnet werden. Dabei ist zu beachten, dass das Koordinatensystem des Roboters rechtshändig und das der HoloLens linkshändig ist. Daher müssen die Koordinaten entsprechend transformiert werden.

4.3 Interaktion

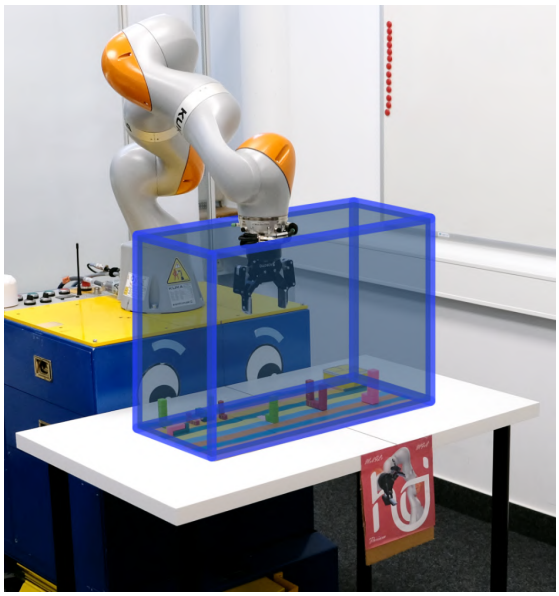


Abbildung 21: Der Arbeitsbereich ist 59,5 cm x 29 cm x 23,2 cm (Länge x Breite x Höhe) groß. Nur innerhalb dieser Abmessungen kann sich der Roboter bewegen.

In Abbildung 21 ist der 29 cm x 59,5 cm große und 23,2 cm hohe Arbeitsbereich zu erkennen. Nur in diesen Abmessungen kann sich der Roboter, während die Applikation des Prototyps läuft, bewegen. Der Arbeitsbereich wurde so gewählt, dass der Roboter nicht in

den Tisch fahren kann oder sich zu weit vom ihm entfernt. Darüber hinaus kann er sich linear von jedem Punkt zu jedem Punkt bewegen. Bewegungen außerhalb des Arbeitsbereiches würden eventuell zu einem weiten Ausschlag der Achsen führen und damit die Bewegung des Roboters für den Benutzer nicht mehr vorhersehbar machen. Durch diese Vorhersehbarkeit werden Visualisierungen, die anzeigen, wie der Roboter sich zu einer Position bewegen wird, überflüssig. Denn auch ohne Visualisierungen macht der Roboter keine unerwarteten Bewegungen, die den Benutzer erschrecken oder Gegenstände umkippen könnten.

Der Prototyp wird über Kopfbewegungen, die einen Cursor steuern, und Sprachbefehle gesteuert. Die Position des Cursors ist dreidimensional und wird immer in der Mitte des Sichtfeldes des Benutzers angezeigt. Die Tiefe ergibt sich aus der Kollision mit virtuellen Objekten. Diese virtuellen Objekte sind die Tischplatte und die virtuellen Buttons. Dadurch kann der Benutzer den Cursor über die reale Tischplatte bewegen und Punkte auf ihr anvisieren. Für die Kollision des Cursors wird nicht das Spatial Mapping der HoloLens verwendet, da es zu ungenau ist und z. B. die Tischplatte nicht als waagerechte Fläche sondern uneben anzeigen würde. Da durch die Kalibrierungsphase am Anfang die Position der Arbeitsstation und damit die der Tischplatte bekannt ist, wird eine virtuelle und durchsichtige Tischplatte an derselben Position wie die Reale platziert. Eine Interaktion mit weiteren Teilen der Arbeitsstation ist nicht nötig, daher werden dafür auch keine entsprechenden virtuellen Objekte gesetzt. Hologramme überlagern reale Objekte, wenn kein virtuelles Pendant dafür an derselben Stelle existiert. Da dieser Prototyp keine Objekterkennung enthält, wurde bei der Entwicklung besonderes darauf geachtet, dass die Objekte nicht so stark überlagert werden, dass der Benutzer sie nicht mehr erkennen kann oder als störend empfindet.

Die Steuerung des Roboters lässt sich in zwei Arten aufteilen: Die Buttons bieten dem Benutzer eine kontinuierliche manuelle Steuerung an. Die Sprachbefehle hingegen sollen dem Benutzer Arbeit abnehmen, indem sie eine teilautomatisierte Steuerung erlauben. Sie bieten dem Benutzer in einem Befehl ganze Bewegungsabläufe oder Zusammenfassungen von Button-Funktionen an. Der selbstbestimmte Wechsel zwischen manueller Steuerung und Teilautomatisierung ermöglicht ein flexibles und veränderbares System, welches sich an unterschiedliche Montageaufgaben anpassen kann. Ziel bei der Umsetzung der Interaktion war es, sie möglichst einfach und leicht verständlich zu halten, dem Benutzer aber trotzdem nicht in der Steuerung des Roboters einzuschränken.

4.3.1 Interaktion über Buttons

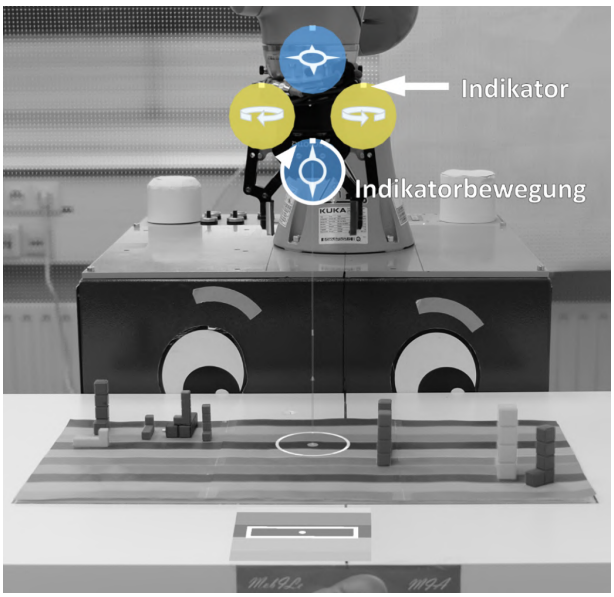


Abbildung 22: Die vier Buttons der Applikation mit gekennzeichneten Indikatoren.

In Abbildung 22 sind die insgesamt vier Buttons der Applikation und ihre Position am Gripper dargestellt. Sie folgen der Bewegung des Grippers, jedoch verschiebt sich ihre vertikale Position am Gripper, wenn sich dieser dem Tisch nähert. Je näher der Gripper an der Tischplatte ist, desto weiter verschiebt sich die Position der Buttons nach oben. Dadurch muss der Benutzer den Kopf nicht zu stark nach oben bewegen, wenn der Gripper von der Tischplatte weiter entfernt ist und er den obersten Button verwenden möchte. Gleichzeitig verdecken die Buttons keine Objekte, wenn der Gripper sich nahe an der Tischplatte befindet. Die Buttons werden ausgelöst indem der Benutzer mit dem Cursor einen Button anvisiert und darauf für eine bestimmte Zeit verweilt. Befindet sich der Cursor über den Buttons, so verändert der Button leicht seine Farbe und ein Indikator zeigt die verbleibende Zeit bis zur Auslösung an. Dieser Indikator kreist um den äußeren Rand des Buttons. Wenn er eine Umrundung zurückgelegt hat, wird der Button ausgelöst. Die Indikatoren und ihre Animation werden in Abbildung 22 dargestellt.

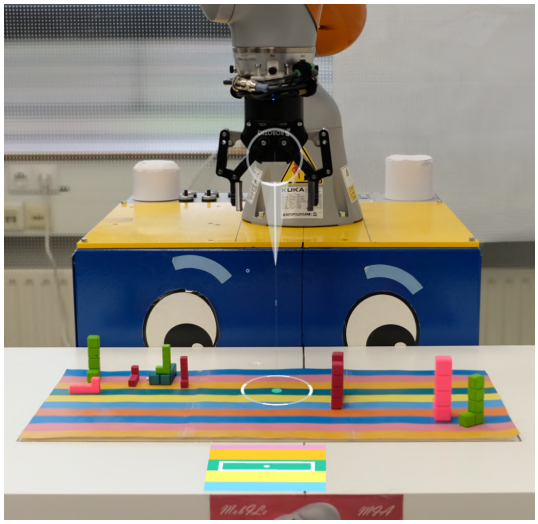
Die Buttons lassen sich, wie die Farben schon andeuten, in zwei Kategorien unterteilen: Die gelben Buttons rotieren den Gripper und die Blauen lösen eine kontinuierliche manuelle Steuerung des Roboters aus. Die Rotationsbuttons rotieren die letzte Achse (A7) entweder nach links (linker Button) oder rechts (rechter Button) um die z-Achse des Roboters. Dabei entspricht ein Auslösen des Buttons einer Rotation um 10°. Der Benutzer kann den Button mehrmals hintereinander auslösen. Die Einschränkung auf 10° hat den

Vorteil, dass sie die Rotation vereinfacht und beschleunigt. Ein sehr präzises Rotieren z. B. von nur 1° ist gerade beim Greifen nicht nötigt, da der Gripper das Objekt passend schiebt. Auch das Platzieren von Objekten mit einer ungeraden Gradzahl ist bei Montageaufgaben eher ungewöhnlich.

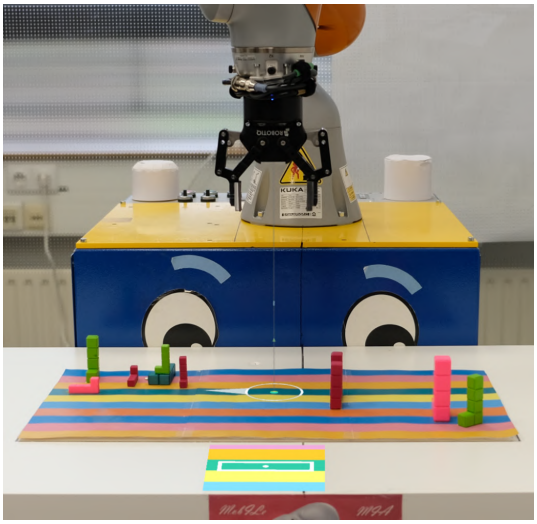
Die manuelle kontinuierliche Steuerung, die von den blauen Buttons ausgelöst wird, ermöglicht dem Benutzer ein präzises Justieren der Position des Grippers. Der untere blaue Button löst diesen sogenannten Präzisionsmodus in vertikaler Richtung aus. Dieser bewegt den Gripper auf der z-Achse des Roboters, also nach oben oder nach unten. Der Präzisionsmodus in horizontaler Richtung wird mit dem oberen blauen Button ausgelöst. Dieser bewegt den Gripper nach links/rechts und vorne/hinten, folglich auf der x- und y-Achse des Roboters. Während der Präzisionsmodus aktiviert ist, sollen keine anderen Buttons aktiviert werden. Sie würden den Benutzer von der Steuerung des Roboters ablenken und gegebenenfalls ungewollte Bewegungen auslösen. Daher werden alle Buttons ausgeblendet, solange der Präzisionsmodus aktiv ist. Zusätzlich signalisiert dies dem Benutzer, dass ein Präzisionsmodus aktiv ist. Als weiteres Feedback wird ein Ton beim Starten und Beenden abgespielt. Nachdem die Buttons beim Auslösen des vertikalen Präzisionsmodus verschwunden sind, erscheint an der Stelle des unteren blauen Buttons ein Kreis. Wie in Abbildung 23a zu erkennen, befindet sich an dem Kreis ein Pfeil der dem Cursor nur in vertikaler Richtung folgt. Folglich kann der Pfeil nur nach oben oder unten zeigen. Der Gripper bewegt sich kontinuierlich in die Richtung des Pfeils. Je weiter weg der Cursor sich vom Kreis befindet, desto länger wird der Pfeil und desto schneller bewegt sich der Gripper. Wenn der Cursor sich im Kreis befindet, verschwindet der Pfeil und der Gripper bewegt sich nicht mehr. Wenn z. B. der Benutzer mit dem Cursor ein Objekt anvisiert, so nähert sich der Gripper erst schnell und dann immer langsamer dem Objekt an. Der Kreis markiert die neutrale Position. Verweilt der Cursor für einige Zeit im Kreis, wird der Präzisionsmodus beendet. Diese Zeit wird, wie bei den Buttons, mit Hilfe eines Indikators angezeigt. Zusätzlich kann der Modus über den Sprachbefehl *Cancel* beendet werden.

Der Präzisionsmodus in horizontaler Richtung funktioniert ähnlich. Der größte Unterschied besteht darin, dass der Benutzer den Gripper nicht nach oben oder unten lenkt, sondern nach links/rechts oder oben/unten. Daher findet die Bewegung des Cursors, im Gegensatz zum vertikalen Präzisionsmodus, nicht in der Luft, sondern auf der Tischplatte statt. Deshalb erscheint kein neuer Kreis, sondern der Blick des Benutzers wird mit einer kleinen Animation auf den schon vorhandenen Kreis unter dem Gripper gelenkt. Dieser Kreis ist erst rot und es erscheint kein Pfeil. Somit findet noch keine Bewegung

des Grippers statt. Der Benutzer muss erst mit dem Cursor den roten Kreis berühren, um den horizontalen Präzisionsmodus zu starten. Diese zweite Bestätigung dient dazu, ungewollte Bewegungen des Grippers zu verhindern. Denn im Gegensatz zum vertikalen Präzisionsmodus befindet sich der Cursor nach dem Auslösen des Buttons nicht in einer neutralen Position, also in einer Position, in welcher der Gripper sich nicht bewegt. Die zweite Bestätigung leitet den Benutzer dazu, den Cursor erst in die neutrale Position zu bewegen und verhindert damit ungewollte Bewegungen des Grippers. Nach der zweiten Bestätigung erscheint ein Pfeil am Kreis, der dem Cursor in vier Richtungen folgt. Diese Richtungen werden durch die Rotation des Grippers bestimmt. Ist der Gripper beispielsweise, wie in Abbildung 23b parallel zum Tisch gedreht, bewegt er sich auch parallel zum Tisch. Wenn er jedoch um 40° gedreht ist, bewegt er sich diagonal zum Tisch. Die Einschränkung auf nur vier Richtungen vereinfacht die Steuerung erheblich. Bei einer Bewegung die in jede horizontale Richtung gehen würde, ist es schwer den Cursor richtig zu Positionieren. Denn die Konzentration des Benutzers liegt auf dem Gripper und seiner kollisionsfreien Steuerung und nicht auf dem Cursor. Sich parallel auf beides zu konzentrieren, wäre für den Benutzer deutlich schwieriger und erhöht die Wahrscheinlichkeit von Kollisionen. Bis auf die oben genannten Unterschiede funktioniert die Steuerung wie die des vertikalen Präzisionsmodus.



(a) Ausgelöster Präzisionsmodus in vertikaler Richtung.



(b) Ausgelöster Präzisionsmodus in horizontaler Richtung.

Abbildung 23: Der ausgelöste Präzisionsmodus in (a) vertikaler und in (b) horizontaler Richtung.

4.3.2 Interaktion über Sprachbefehle

Die HoloLens erkennt Sprachbefehle mit einer kleinen Verzögerung, die je nach Umgebungslautstärke und Aussprache des Benutzers leicht variiert. Daher wird nach dem Erkennen eines Sprachbefehl ein Ton abgespielt, der die Registrierung bestätigt. Der wichtigste Sprachbefehl des Prototyps ist der *Move*-Befehl. Er bewegt den Gripper auf der x- und y-Achse des Roboters zur Position des Cursors. Dies erlaubt dem Benutzer den Gripper schnell über eine längere Distanz im Arbeitsbereich zu bewegen. Der Benutzer kann einen beliebigen Punkt im Arbeitsbereich anvisieren und den Sprachbefehle *Move* benutzen. Nach dessen Registrierung hat der Benutzer eine Sekunde Zeit, seine anvisierte Position zu korrigieren. Als Feedback dieser Nachjustierungszeit verändert der Cursor seine Form und wird kleiner. Durch die Mundbewegung beim Aussprechen eines Sprachbefehls bewegt sich unweigerlich auch der Kopf. Dadurch kann sich die vorher anvisierte Position leicht verändern. Die Nachjustierungszeit gibt dem Benutzer die Möglichkeit den Fehler, den der Sprachbefehl ggf. ausgelöst hat, zu korrigieren. Nach Ablauf dieser Zeit bewegt sich der Gripper, falls er sich noch nicht auf der maximalen Höhe des Arbeitsbereichs befindet, senkrecht dorthin. Erst danach bewegt er sich linear in der xy-Ebene des Roboters zu der anvisierten Position. Die Ausführung des *Move*-Befehls findet nur in der maximalen Höhe des Arbeitsbereichs statt, um Kollisionen zu verhindern. Die maximale Höhe des Arbeitsbereichs wird daher im Folgenden als Move-Höhe bezeichnet. Der Benutzer muss nicht einschätzen, wie sich der Gripper bewegen wird und sich nicht darum sorgen, dass er etwas umstößt. Denn auf Move-Höhe kann der Gripper mit keinem Gegenstand kollidieren. Der Sprachbefehl *Down* bewegt den Gripper auf der z-Achse des Roboters und ermöglicht damit eine schnelle Bewegung Richtung Tisch. Dabei bewegt sich der Gripper immer genau auf die Hälfte des Abstandes zwischen Ende der Gripperfinger und Tisch. Beim mehrmaligen Verwenden des Sprachbefehls nähert sich der Gripper dem Tisch an, kann mit diesem aber nicht kollidieren. Für eine schnellere Rotation des Grippers gibt es den Sprachbefehl *Turn*. Dieser rotiert den Gripper auf oder um 90°. Das bedeutet, wenn der Gripper um 60° rotiert ist, rotiert ihn der *Turn* Sprachbefehl auf 90°. Wird *Turn* erneut gesagt, so rotiert er sich auf 0° und eine weitere Wiederholung würde ihn wieder auf 90° rotieren. Der Bewegungsbereich der letzten Achse des Roboters ist auf 175° beschränkt, daher ist eine Rotation auf 180° nicht möglich. Daher pendelt die Rotation beim Wiederholen des *Turn* Sprachbefehls zwischen 0° und 90°. Die Sprachbefehle *Move*, *Down* und *Turn* sind Abkürzungen oder Zusammenfassungen der Buttonfunktionen. Daher eignen sich diese Sprachbefehle für eine ungefähre Positionierung des Grippers und die Buttons ergänzen sie um eine Feinjustierung.

Die Sprachbefehle *Pick* und *Place* sind für das Greifen und Loslassen von Objekten zuständig. Bei beiden fährt der Gripper, wenn er sich auf Move-Höhe befindet, auf Tischhöhe herunter. Danach öffnet oder schließt sich der Gripper und fährt senkrecht auf Move-Höhe zurück. Wenn der Gripper nicht auf der Move-Höhe ist, fährt er nicht bis auf Tischhöhe, sondern öffnet oder schließt sich und fährt dann senkrecht wieder auf Move-Höhe. Dies erlaubt Objekte auf andere Objekte oder Erhöhungen zu stellen oder von diesen zu greifen. Gleichzeitig beschleunigt es den Prozess, wenn etwas von der Tischhöhe gegriffen oder platziert werden muss. Nachdem der *Pick* oder *Place* Sprachbefehl ausgeführt wurde, verschwinden die Buttons. Dies verhindert, dass der Benutzer beim Verfolgen der Bewegung des Grippers ungewollt einen Button aktiviert.

4.4 Visuelle Helfer

Bei der Steuerung des Roboters sitzt der Benutzer frontal vor der Arbeitsstation und nimmt eine feste Perspektive ein. Diese schränkt den Benutzer ein und erschwert vor allem die Tiefenwahrnehmung. Es fällt schwer, aus dieser Perspektive die Position des Grippers im Verhältnis zum Tisch richtig einzuschätzen. Zusätzlich ist es schwierig, die Dimensionen des Grippers zu beurteilen und ob dieser mit umliegenden Objekten kollidieren könnte. Die visuellen Helfer sollen bei diesen Aspekten den Benutzer unterstützen und ihn somit bei der Bedienung des Prototyps entlasten. Außerdem sollen sie ihm dabei helfen, Fehler zu vermeiden und somit Montageprozesse erfolgreich umzusetzen. Gleichzeitig ergänzen die visuellen Helfer das Interaktionskonzept und geben dem Benutzer wichtige Hilfestellung. Vor allem soll er dabei unterstützt werden, *Pick* und *Place* Sprachbefehle möglichst auf Move-Höhe ausführen zu können, um so Montageaufgaben schneller und einfacher durchzuführen. In Abbildung 24 sind die Standardansicht auf den Prototyp und die gekennzeichneten visuellen Helfer zu sehen. Im Folgenden wird jeder visuelle Helfer und die Art und Weise, wie er den Benutzer unterstützen soll, erläutert.

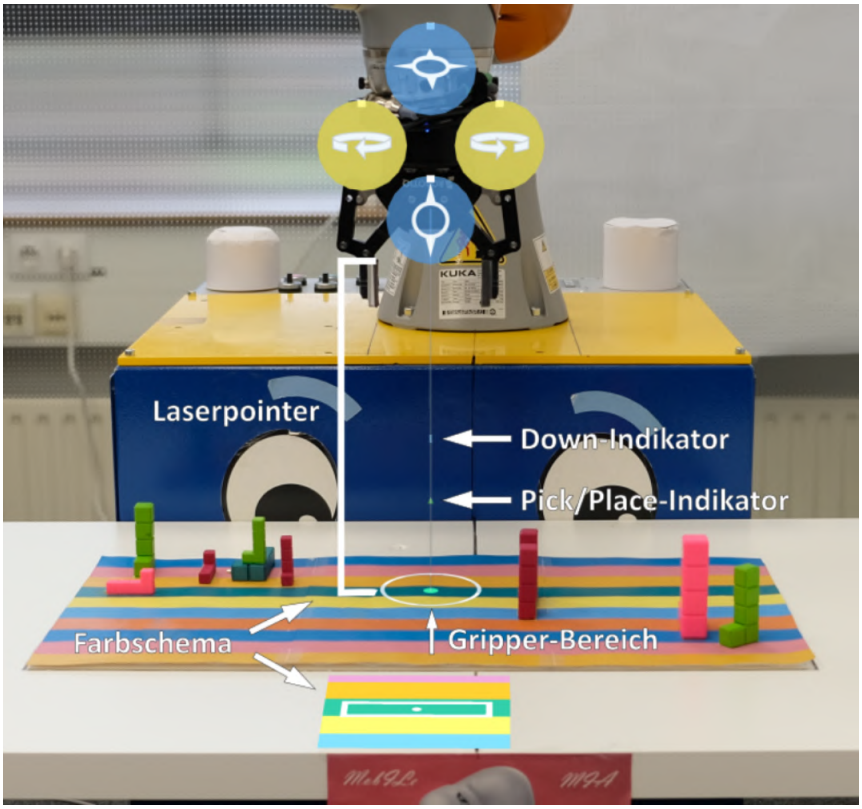


Abbildung 24: Kennzeichnung der unterschiedlichen visuellen Helfer.

4.4.1 Laserpointer

Der Laserpointer ist eine halbtransparente weiße Linie, die immer in der Mitte des Grippers angezeigt wird. Der Ursprung der Linie entspringt der Oberkante des Greifbereichs und verläuft bis auf den Tisch. Die Linie verfolgt die Position des Grippers und ihre Länge passt sich dynamisch dem Abstand des Grippers vom Tisch an. Er hat den Namen *Laserpointer* erhalten, weil sich am Ende der Linie ein kleiner Kreis befindet, der sich der Farbe des Farbschemas anpasst (siehe Kapitel 4.4.4). Der Laserpointer dient als visuelle Verlängerung des Grippers auf dem Tisch und hilft die Position des Grippers im Verhältnis zum Tisch einzuschätzen. Dadurch lässt sich der Gripper einfacher mittig über Objekte positionieren. Insgesamt unterstützt er den Benutzer bei der Positionierung des Grippers auf der y-Achse des Roboters, um Bewegungen nach links/rechts im Verhältnis zum Tisch richtig einschätzen zu können. Dabei wurde der Laserpointer möglichst dünn und unauffällig entworfen, damit er Objekte nicht zu stark überlagert und somit nicht stört.

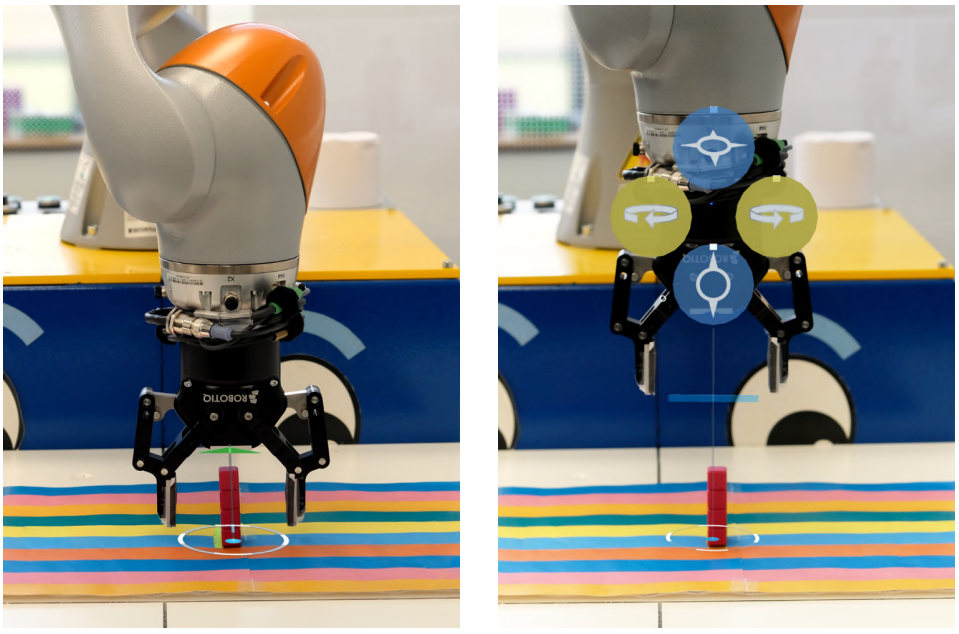
4.4.2 Down-, Pick/Place- und Schließ-Indikator

Es existieren drei unterschiedliche Indikatoren, die auf dem Laserpointer in verschiedenen Höhen angezeigt werden. Es sind immer zwei von den drei Indikatoren auf einmal sichtbar. Der Down-Indikator wird immer angezeigt und die Sichtbarkeit des Pick/Place- und Schließ-Indikators hängt von der aktuellen Höhe des Grippers ab. Ist dieser auf Move-Höhe, so wird der Pick/Place-Indikator angezeigt und befindet sich der Gripper niedriger, wird der Schließ-Indikator dargestellt.

Hat der Benutzer den Gripper zu einem Objekt passend auf Move-Höhe positioniert und möchte dieses greifen, so hilft ihm der Pick/Place-Indikator dabei, die Höhe vom Objekt richtig einzuschätzen und somit Kollisionen zu vermeiden. Denn der Pick/Place-Indikator markiert die maximale Höhe, die ein Objekt besitzen darf, damit es mit dem *Pick* Sprachbefehl automatisch gegriffen werden kann. Bei einem höheren Objekt würde der Gripper, bevor er Tischhöhe erreicht, mit dem Objekt kollidieren. Außerdem erinnert der Pick/Place-Indikator daran, dass der Gripper bis auf Tischhöhe fährt, wenn sich der Gripper auf Move-Höhe befindet und er den *Pick* oder *Place* Sprachbefehl verwendet. Der Pick/Place-Indikator ist grün und besitzt eine dreieckige Form. Wenn der *Pick* oder *Place* Sprachbefehl verwendet wird, startet eine Animation, die den Pick/Place-Indikator ausfahren und zu einem längeren Dreieck werden lässt. Dies ist in Abbildung 25a verdeutlicht. Die Animation soll den Benutzer auf die Funktion des Pick/Place-Indikators aufmerksam machen und seine Position im Vergleich zum Gripper verdeutlichen. Die

Animation endet kurz nachdem der Gripper das Objekt gegriffen oder platziert hat.

Erkennt der Benutzer durch den Pick/Place-Indikator, dass ein Objekt zu groß ist und die maximale Höhe für den *Pick* Sprachbefehl überschreitet, so muss er den Gripper manuell vertikal zum Objekt steuern. Verwendet er dafür den *Down* Sprachbefehl, so hilft ihm der Down-Indikator beim Einschätzen der zukünftigen Position des Grippers und verhindert damit Fehleinschätzungen sowie Fehler. Der *Down* Sprachbefehl lässt den Gripper auf die halbe Höhe des Abstandes zwischen Gripperfinger und Tisch fahren. Der Down-Indikator markiert diese Position mit einem blau markierten Abschnitt auf dem Laserpointer. Die Position des Down-Indikators kennzeichnet die Position, bei der die Spitzen der Gripperfinger enden würden. Wie auch der Pick/Place-Indikator wurde der Down-Indikator möglichst unauffällig gestaltet, um den Benutzer während der Steuerung nicht zu stören. Der Benutzer kann sich, kurz bevor er den *Down* Sprachbefehl ausführen möchte, auf den Down-Indikator konzentrieren, um abzuschätzen, ob er den Sprachbefehl verwenden will oder nicht. Wird der *Down* Sprachbefehl verwendet, startet analog zum Pick/Place-Indikator, eine Animation, die in Abbildung 25b veranschaulicht ist. Die Animation endet kurz nachdem der Gripper die Zielposition des *Down* Sprachbefehl erreicht hat. Danach wird der Down-Indikator wieder normal auf halbem Abstand zwischen Gripperfinger und Tisch angezeigt.

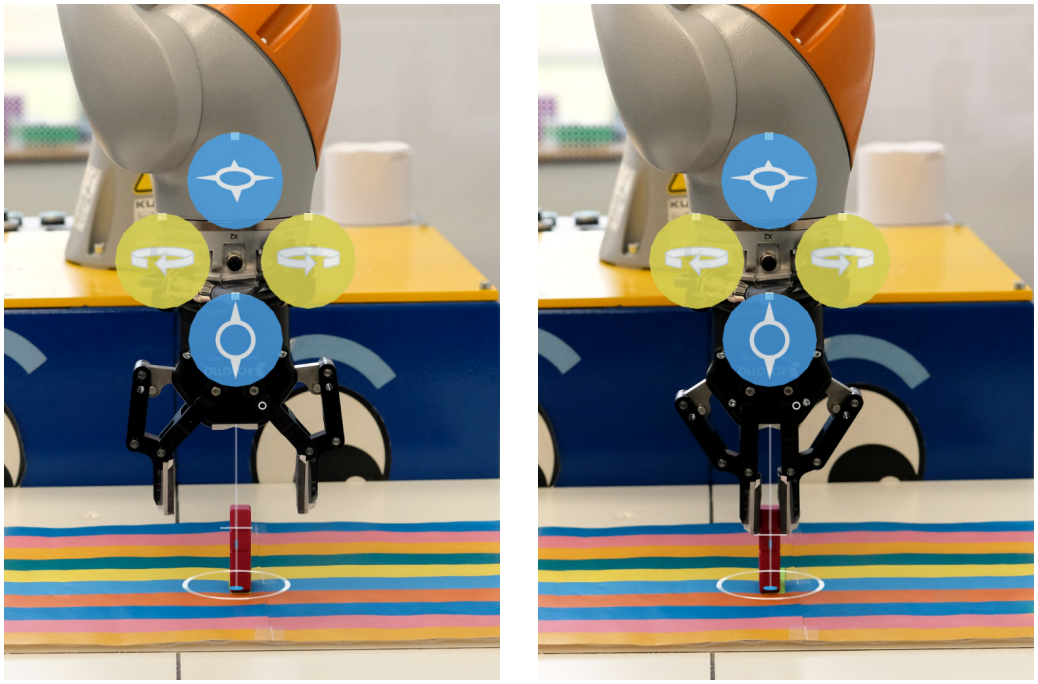


(a) Positionierung und Animation des Pick/Place-Indikators.

(b) Positionierung und Animation des Down-Indikators.

Abbildung 25: Die Animation des (b) Down-Indikators und des (a) Pick/Place-Indikators.

Sobald der Benutzer den *Down* Sprachbefehl verwendet hat und der Gripper sich nicht mehr auf Move-Höhe befindet, verschwindet der Pick/Place-Indikator und der Schließ-Indikator wird angezeigt. Der Schließ-Indikator hilft dem Benutzer bei der Positionierung des Grippers auf der z-Achse des Roboters, um ein Objekt zu greifen. Durch ihn muss der Benutzer nicht selbst einschätzen, ob das Objekt zu hoch oder zu tief gegriffen wird. Denn die Gripperfinger schließen sich nicht waagerecht, sondern schlagen nach unten aus. Das beutet, dass der Gripper wenige Zentimeter weiter unten greift, als die Ausgangsstellung des Grippers vermuten lässt. Der Schließ-Indikator zeigt die Höhe der Spitzen der Gripperfinger nach dem Schließen des Grippers an. Der Höhenunterschied zwischen dem geöffneten und geschlossen Gripper im Vergleich zum Schließ-Indikator sind in Abbildung 26a und 26b dargestellt. Auch der Schließ-Indikator wurde möglichst unauffällig gestaltet, um Objekte nicht zu überlagern und dem Benutzer beim Ansteuern von Objekten nicht zu stören.



(a) Die Position des Schließ-indikators im Vergleich zum offenen Gripper.

(b) Die Position des Schließ-indikators im Vergleich zum geschlossenen Gripper.

Abbildung 26: Der Schließ-Indikator im Vergleich zum (a) geöffneten und (b) geschlossenen Gripper.

4.4.3 Gripper-Bereich

Der Gripper-Bereich soll die Dimension, sowie die Position des Grippers mit dem Tisch in ein Verhältnis bringen. Damit unterstützt er den Benutzer beim Positionieren des Grippers und beim Einschätzen der Größe von Objekten. Dies hilft Fehler zu vermeiden, da z. B. Objekte, die länger als der Abstand zwischen den Gripperfingern sind, nur von der kürzeren Seite gegriffen werden können. Zusätzlich werden durch das Anzeigen der Dimension des Grippers auf dem Tisch eventuelle Kollisionen vermieden. Sollte sich ein weiteres Objekt innerhalb des Gripper-Bereichs befinden, könnte der Gripper mit diesem beim *Pick* Sprachbefehl kollidieren. Nur wenn im Gripper-Bereich kein weiteres Objekt existiert, kann der Benutzer sicher sein, nicht mit einem anderen Objekt zu kollidieren.

Abbildung 27: Die *Version für die erste Studie* mit gekennzeichneten Gripper-Verlängerungen und der ersten Version der Farb-Indikatoren.

Ein erster Ansatz zur Darstellung des Gripper-Bereiches wurde mit Hilfe von Gripper-Verlängerungen umgesetzt. Die Gripper-Verlängerungen waren Linien, die die Breite der Gripperfinger besaßen und von den Gripperfingern bis auf den Tisch verliefen. Sie wurden in der *Version für die erste Studie* verwendet und sind in der Abbildung 27 gekennzeichnet. Bei der ersten Studie stellte sich heraus, dass die Dimension des Grippers für die

Benutzer durch die Gripper-Verlängerungen nicht ersichtlich wurde. Obwohl die Gripper-Verlängerungen die Breite der Gripperfinger besaßen, waren sie aus der Perspektive des Benutzers schwer abzuschätzen. Die Mehrheit der Probanden hatte dafür den Kreis verwendet, der beim horizontalen Präzisionsmodus auf dem Tisch angezeigt wird. Sie haben den horizontalen Präzisionsmodus oft nur aktiviert, um den Kreis zu sehen und nicht, um den Gripper zu bewegen. Manche Probanden empfanden den Kreis sogar als den hilfreichsten visuellen Helfer. Daher wurde die Anzeige des Gripper-Bereiches mit dem Kreis, der sich immer unter dem Gripper auf Tischhöhe befindet, erweitert. Der Durchmesser des Kreises entspricht dem Abstand der Gripperfinger. Der Gripper-Bereich wird als halbtransparente Umrandung eines Kreises dargestellt, da er damit keine Objekte überlagert und den Benutzer beim Positionieren des Grippers über einem Objekt nicht stört. Wäre der Gripper-Bereich z. B. mit einem farbig gefüllten Rechteck dargestellt, wäre ein darunterliegendes Objekt nur noch schwer zu erkennen.

Der Kreis in Kombination mit den Gripper-Verlängerungen hat den Sichtbereich des Benutzers mit zu vielen virtuellen Elementen verdeckt. Da der Kreis die Funktion der Gripper-Verlängerungen ersetzt und erweitert, wurde auf die Gripper-Verlängerungen verzichtet.

4.4.4 Farbschema

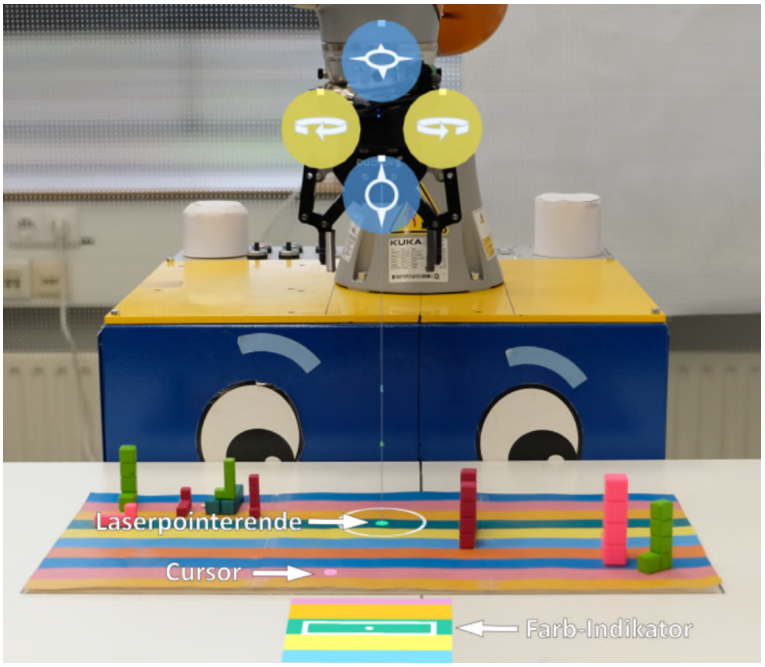


Abbildung 28: Farbschema mit den gekennzeichneten virtuellen Repräsentationen.

Das Farbschema besteht aus einem ausgedruckten Blatt mit farbigen Streifen, das im kompletten Arbeitsbereich des Roboters auf dem Tisch ausliegt und den virtuellen Repräsentationen. Sie bestehen aus dem Farb-Indikator, Cursor und dem kleinen Kreis am Ende des Laserpointers (Laserpointerende) und sind in Abbildung 28 hervorgehoben. Die Kombination aus realem und virtuellem Farbschema soll dem Benutzer bei der Tiefenwahrnehmung helfen, indem die Tiefenposition des Gripper mittels virtueller Farben verdeutlicht wird. Dabei bieten die drei virtuellen Repräsentationen des Farbschemas unterschiedliche Hilfestellungen. Der Cursor und das Laserpointerende verändern ihre Farbe, je nachdem über welcher realen Farbe sie sich befinden. Die Farbveränderung beim Cursor hilft dem Benutzer beim Anvisieren einer Position und damit bei der Auswahl der zukünftigen Position. Ohne Farbveränderung des Cursors ist es schwer, seine Tiefenposition richtig einzuschätzen. Damit die aktuelle Tiefenposition des Grippers für den Benutzer ersichtlich wird, gibt es den Laserpointer und den Farb-Indikator. Der Farb-Indikator befindet sich außerhalb des Arbeitsbereiches und folgt der Position des Grippers auf der y-Achse des Roboters. Damit ist der Farb-Indikator immer in der Nähe des Blickfeldes des Benutzers. Er zeigt einen Ausschnitt des Farbschemas aus dem Bereich, in dem sich der Gripper aktuell befindet. Die Position und Rotation des Grippers auf dem Farbausschnitt wird mit Hilfe eines Rechtecks mit einem Punkt in der Mitte dargestellt. Das Rechteck besitzt dieselbe Breite, wie die Gripperfinger und die Länge entspricht dem Abstand zwischen den Gripperfingern. Dadurch kann der Benutzer die Position des Grippers im Verhältnis zu den anderen Farben wahrnehmen. Dies ermöglicht z. B. den Gripper an den Rand oder genau zwischen zwei Farbstreifen zu navigieren. Da das Sichtfeld der HoloLens sehr eingeschränkt ist, muss der Benutzer den Blick vom Gripper abwenden, um den Farb-Indikator sehen zu können. Das Laserpointerende bietet dem Benutzer eine Rückmeldung über die aktuelle Tiefenposition des Grippers, ohne dass dieser seinen Blick vom Gripper abwenden muss, weil das Laserpointerende die Farbe annimmt, die sich direkt unter der Mitte des Grippers befindet.

Das Farbschema umfasst zehn Farbstreifen mit sieben verschiedenen Farben. Die Farbstreifen sind 3 cm breit und so lang wie der Arbeitsbereich (59,5 cm). Die Breite der Farbstreifen wurde in Abhängigkeit der 2,2 cm breiten Gripperfinger festgelegt. Die Farbstreifen wurden etwas breiter gestaltet, da auch die Ränder der Farbstreifen dem Benutzer Anhaltspunkte über die Position eines Objektes geben. Dadurch kann der Benutzer den Gripper entweder mittig oder an den Rand eines Farbstreifens navigieren. Die Reihenfolge der Farben ist fest und wiederholt sich. Dadurch ist das Farbschema beliebig erweiterbar und kann an unterschiedlich große Arbeitsbereiche angepasst werden. Bei der Auswahl der Farben wurde eine speziell an Farbenblinde angepasste Farbskala verwendet

(Wong, 2011). Dadurch sind die Farben auch für Menschen, die unter einer Farbblindheit leiden, gut zu unterscheiden.

Wie in Abbildung 27 zu erkennen, war das Farbschema des Prototyps in der *Version für die erste Studie* anders gestaltet. Es bestand nicht aus Farbstreifen, sondern aus einem Farbgradienten, der gleichmäßige Farbübergänge zwischen fünf Farben enthielt. Kleine ovale Farb-Indikatoren befanden sich unter den Gripper-Verlängerungen und wurden direkt über dem Farbschema angezeigt. Nur der Farbwechsel des Cursors und des Laserpointerendes sind gleich geblieben. Die Idee hinter dem Farbgradienten bestand darin, dass das Farbschema durch die Farbübergänge mehr Informationen enthält. Der Benutzer hätte mehr Anhaltspunkte gehabt. Jedoch zeigte sich in der ersten Studie, dass die Probanden die feinen Unterschiede im Farbverlauf nur schlecht wahrnahmen. Dies lag unter anderem daran, dass sich die virtuellen Farben in ihrer Erscheinung von den realen Farben unterscheiden. Die Darstellung von Hologrammen funktioniert über eine additive Anzeige. Das bedeutet, dass Hologramme durch das Hinzufügen von Licht in der realen Welt erzeugt werden. Dies lässt manche Farben in den Hologrammen anders erscheinen (Microsoft, 2018). Aufgrund dieser Erkenntnis, wurden die Farbstreifen ausgewählt, da die harten Grenzen der aneinander liegenden Farben deutlich bessere Anhaltspunkte bieten. Eine weitere Veränderung wurde mit den Farb-Indikatoren vorgenommen. Sie wurden als Ovale unter der Gripper-Verlängerung angezeigt. Die ovalen Farb-Indikatoren waren etwas breiter, als die Gripper-Verlängerungen und sollten die Position des Grippers in ein Verhältnis mit den anderen Farben bringen. Jedoch hatte diese Eigenschaft die Probanden verwirrt, da es für sie nicht ersichtlich war, über welcher Farbe sich der Gripper genau befand. Außerdem waren die Farb-Indikatoren direkt über dem Farbschema schlecht sichtbar und haben daher mehr gestört als geholfen. Daher wurde ein Farb-Indikator außerhalb des realen Farbschemas gewählt. Dort kann ein größerer Ausschnitt des Farbschemas angezeigt werden, ohne den Sichtbereich des Benutzers zu überladen. Jedoch hat dies auch den Nachteil, dass die Aufmerksamkeit des Benutzers vom Roboterarm weggelenkt wird. Jedoch ist dies ein Kompromiss der gemacht werden musste, um den Benutzer nicht zu verwirren oder den Sichtbereich zu überladen.

4.4.5 Kraft-Feedback

Das Kraft-Feedback visualisiert die Kraft, die auf die letzte Achse des Roboters ausgeübt wird und vermeidet dadurch die im Folgenden beschriebenen Fehler. Wenn der Benutzer z. B. ein Objekt auf ein anderes Objekt platzieren will, navigiert er den Gripper mit dem vertikalen Präzisionsmodus herunter, bis das gegriffene Objekt das Andere berührt und er den Gripper mit dem Sprachbefehl *Place* öffnen kann. Aufgrund der festen Distanz und Position des Benutzers ist es nicht leicht zu erkennen, wann sich die Objekte berühren. Solange der Benutzer nicht erkennt, dass sich die Objekte berühren, wird er versuchen, den Gripper weiter herunter zu navigieren. Dabei wird immer mehr Kraft auf das Objekt und den Gripper ausgeübt. Erzeugt diese Kraft ein Drehmoment von 20 Nm in der letzten Achse des Roboters, stoppt die Sicherheitstabelle den Roboter sofort mit der Folge, dass die Robotersteuerung manuell neugestartet werden muss. Dies soll durch das Kraft-Feedback verhindert werden. Die Visualisierung des Kraft-Feedbacks wird über einen Farbverlauf am Pfeil des vertikalen Präzisionsmodus verwirklicht. Dabei verändert sich die Farbe des Pfeils von weiß nach rot, je mehr Kraft ausgeübt wird. Überschreitet die Kraft ein Drehmoment von 10 Nm wird der vertikale Präzisionsmodus automatisch beendet und der Gripper fährt 1 mm hoch. Dieses letzte Hochfahren bringt den Gripper automatisch in die optimale Position zum Platzieren des Objektes. Die Sicherheitstabelle ist mit 20 Nm relativ gering eingestellt, dadurch wird der Grenzwert im vertikalen Präzisionsmodus sehr schnell erreicht. Damit der Benutzer mehr zeitlichen Spielraum hat, um auf das Kraft-Feedback reagieren zu können, wurde unter das Farbschema eine dünne Schaumstoffisolierung angebracht. Sie federt die Kraft des Grippers ab. Dadurch wird die Grenze der Sicherheitstabelle nur noch ausgelöst, wenn der vertikale Präzisionsmodus mit sehr schneller Geschwindigkeit durchgeführt wird.

5 Evaluation des Prototyps

Im Rahmen dieser Masterarbeit wurde eine Studie durchgeführt, um herauszufinden ob und wie gut die entwickelten visuellen Helfer des Prototyps den Benutzer bei der Bedienung des Roboters unterstützen. Dazu sollten Probanden den Prototyp verwenden, um Pick und Place Aufgaben durchzuführen. Inwieweit die visuellen Helfer ausschlaggebend für Unterstützung sind, sollte getestet werden, indem der Prototyp mit zwei anderen Versionen der visuellen Helfer verglichen wurde. Im Folgenden werden die Studie und das Vorgehen beschrieben. Im Anschluss werden die Analyse der erhobenen Daten und die daraus gezogenen Erkenntnisse erläutert.

5.1 Versionen der visuellen Helfer

Die drei verschiedenen Versionen der visuellen Helfer sind in Tabelle 2 aufgelistet und wurden wie folgt benannt: *Cues* (Helfer) ist die Version der Helfer, die in Kapitel 4.4 beschrieben ist und ohne Objekterkennung funktioniert. *Advanced Cues* (erweiterte Helfer) beinhaltet die Version der visuellen Helfer, die mit Objekterkennung funktionieren und in Kapitel 5.1.2 beschrieben ist. Die Version des Prototyps ohne jegliche Helfer wird *No Cues* (keine Helfer) genannt und wird in Kapitel 5.1.1 erläutert. Im Folgenden wird beschrieben, inwieweit sich *Advanced Cues* und *No Cues* von den in dieser Masterarbeit bereits beschriebenen *Cues* unterscheiden.

Name	**Übersetzung**	**Beschreibung**	**Kapitel**
No Cues	Keine Helfer	Keine visuellen Helfer	5.1.1
Cues	Helfer	Visuelle Helfer ohne Objekterkennung	4.4
Advanced Cues	Erweiterte Helfer	Visuelle Helfer mit Objekterkennung	5.1.2

Tabelle 2: Die verschiedenen Versionen des Prototyps und seiner visuellen Helfer, die gegeneinander getestet wurden.

5.1.1 No Cues

Diese Version des Prototyps besitzt keinerlei visuelle Helfer und fungiert als Ausgangszustand, mit dem die anderen beiden Versionen verglichen werden können. Die virtuellen Objekte beschränken sich auf den Cursor und die vier Buttons für die Interaktion. Bei beiden Präzisionsmodi fehlen der Kreis und der Pfeil, die als Orientierungshilfe verwendet werden könnten. Gerade beim horizontalen Präzisionsmodus ist der auf dem Tisch darstellte Kreis eine Hilfe, um die Position des Gripper im Verhältnis zum Tisch festzustellen. Da Kreis und Pfeil nicht anzeigt werden, fehlt der Kreis für die zweite Bestätigung des horizontalen Präzisionsmodus. Damit die Interaktion dadurch nicht benachteiligt wird, fungiert ein Sprachbefehl als zweite Bestätigung. Für das Verwenden des horizontalen Präzisionsmodus muss der Benutzer nach der Aktivierung des Buttons mit dem Cursor den Tisch anvisieren und mit den Sprachbefehl *Slow* die zweite Bestätigung aktivieren. Erst nachdem der Sprachbefehl erkannt wurde, bewegt sich der Gripper. Der Rest der Interaktion funktioniert wie bei Cues, nur ohne visuelle Helfer.

5.1.2 Advanced Cues

Die Advanced Cues wurden von Tim Dierks im Rahmen seiner Masterarbeit „Visual Cues: Integration of object pose recognition with an augmented reality system as means to support visual perception in human-robot control“ entwickelt. Diese Version der visuellen Helfer basiert auf Objekterkennung. Im Folgenden werden nur die visuellen Helfer der Advanced Cues und wie sie unterstützen zusammenfassend erläutert. Weitere Details zur Objekterkennung und den Advanced Cues können in der Masterarbeit von Tim Dierks nachgelesen werden. Der größte Unterschied zwischen den Advanced Cues und den Cues besteht in dem autonomen Anfahren der optimalen Höhe beim Greifen oder Platzieren von Objekten. Dadurch sind der Sprachbefehl *Down* und der vertikale Präzisionsmodus überflüssig. Das System erkennt alle Objekte mit ihrer Position und Orientierung innerhalb des Arbeitsbereiches und berechnet dadurch die optimale Höhe für das Greifen oder Platzieren. Daher muss der Benutzer den Gripper nicht nach oben oder unten navigieren, sondern nur die Sprachbefehle *Pick* oder *Place* verwenden. Dies nimmt dem Benutzer Arbeit beim Steuern des Roboters ab und verhindert somit potenzielle Fehler.

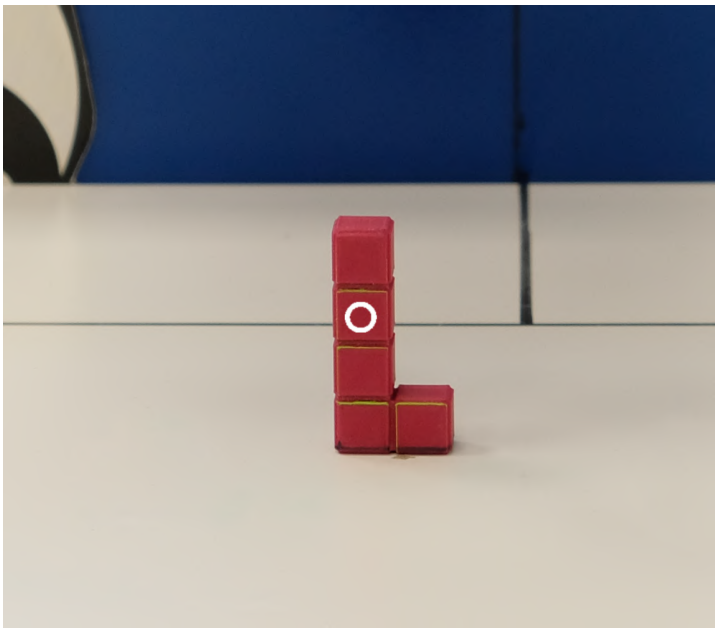
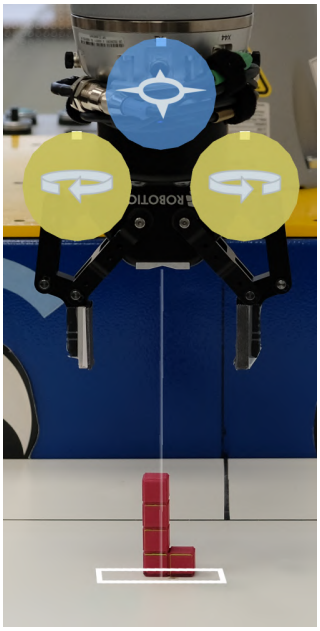


Abbildung 29: Der Cursor kollidiert mit dem Objekt und wird an dessen Oberfläche angezeigt.

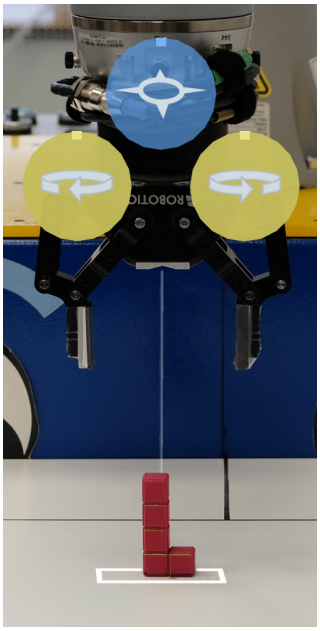
Darüber hinaus kann der Cursor mit den Objekten kollidieren und wird dadurch auch auf oder an einem Objekt angezeigt. In Abbildung 29 ist zu erkennen, wie der Cursor mit einem Objekt kollidiert und auf dessen Oberfläche angezeigt wird. Das Kollidieren des Cursors mit Objekten macht das Anvisieren von Objekten leichter. Neben diesen grundsätzlichen Unterschieden in der Interaktion, sind auch die visuellen Helfer verschieden. Die Unterschiede werden im Folgenden erläutert.

Verdeckung

Die visuellen Helfer der Advanced Cues bestehen aus einem Laserpointer und einer Anzeige für den Gripper-Bereich, die vergleichbar sind mit denen der Cues. Der größte Unterschied besteht darin, dass der Laserpointer und der Gripper-Bereich bei Advanced Cues durch die realen Objekte verdeckt werden können. Die Verdeckung gibt dem Benutzer Informationen darüber, ob der Laserpointer oder Gripper-Bereich mit einem Objekt kollidieren kann. Ist beispielsweise der Laserpointer teilweise verdeckt, befindet sich der Gripper direkt über dem Objekt. Der Vergleich von Laserpointer und Gripper-Bereich mit und ohne Verdeckung ist in den Abbildungen 30 verdeutlicht. Die Verdeckung verhindert, dass die visuellen Helfer die realen Objekte überlagern. Daher konnte der Gripper-Bereich im Vergleich zu den Cues anderes gestaltet werden. Er wird als rechteckiger Rahmen angezeigt, der die Breite und den Abstand der Gripperfinger hat. Er repräsentiert den aktuellen Zustand des Grippers, ohne dabei Objekte zu verdecken.



(a) Laserpointer und Gripper-Bereich ohne Objekterkennung und damit ohne Verdeckung.



(b) Laserpointer und Gripper-Bereich mit Verdeckung.

Abbildung 30: Laserpointer und Gripper-Bereich der Advanced Cues (a) ohne Verdeckung und (b) mit Verdeckung.

Gripper-Verlängerung

Die Gripper-Verlängerungen sind Linien, die von den Gripperfingern auf den Tisch verlaufen. Sie werden nur angezeigt, wenn sich unter dem jeweiligen Gripperfinger ein Objekt befindet. Dies warnt den Benutzer vor einer Kollision, die beim Verwenden des Sprachbefehls *Pick* erfolgen würde. Solch ein Szenario wird in Abbildung 31 verdeutlicht. Die Gripper-Verlängerungen werden nur dann angezeigt, wenn sie den Benutzer vor einer Kollision warnen und verringern dadurch die visuellen Reize im Sichtfeld des Benutzers. Sie erweitern die Funktionalität des Gripper-Bereichs und Laserpointers, indem sie den Benutzer explizit auf einen potenziellen Fehler aufmerksam machen.

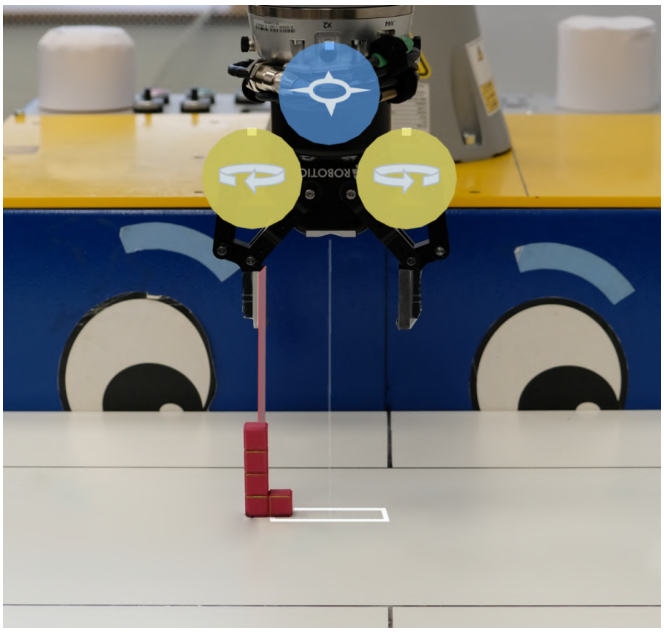
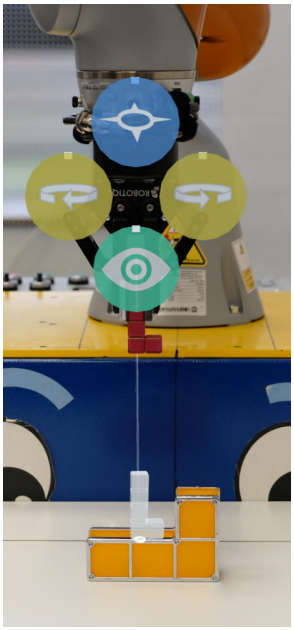


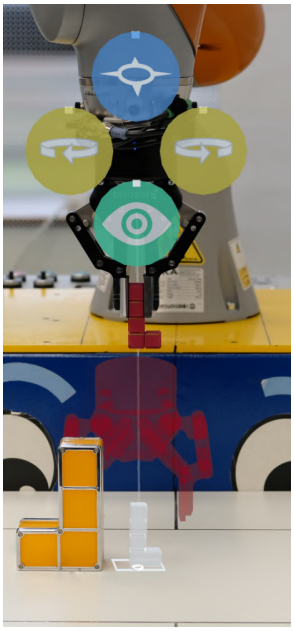
Abbildung 31: Kollision der Gripper-Verlängerung mit einem Objekt.

Ghost-Objekt

Das Ghost-Objekt ist ein visueller Helfer, der speziell das Platzieren von Objekten unterstützt. Sobald ein Objekt gegriffen wurde, erscheint ein virtuelles Ebenbild dieses Objektes, welches Ghost-Objekt genannt wird. Es erscheint an der Position des Cursors und verfolgt diesen. Der Benutzer muss sich das Objekt nicht selbst an der Zielposition vorstellen, sondern kann mit Hilfe des Ghost-Objektes sehen, wie es dorthin passt. In Abbildung 32a ist das Ghost-Objekt und ein grüner Button mit einem Auge zu sehen. Dieser Button erscheint nur, wenn ein Objekt gegriffen wurde. Wird er ausgelöst, verfolgt das Ghost-Objekt nicht mehr den Cursor, sondern wird unter dem Gripper angezeigt. Dies entspricht der zukünftigen Position des Objektes nachdem der Sprachbefehl *Place* verwenden wurde. Dadurch wird der Benutzer bei der Feinjustierung des Objektes unterstützt. Ein virtueller roter Gripper, der sogenannte Ghost-Gripper, warnt den Benutzer vor Kollisionen, die beim Platzieren mit umliegenden Objekten ausgelöst werden könnten. Er ist in Abbildung 32b dargestellt.



(a) Anzeige des Ghost-Objekts, welches dem Cursor folgt.



(b) Anzeige einer evetuellen Kollision des Grippers mit umliegenden Objekten.

Abbildung 32: Hat der Gripper ein Objekt gegriffen, wird (a) ein dem Objekt entsprechendes Ghost-Objekt angezeigt. Zusätzlich werden (b) zukünftige Kollisionen des Grippers mit umliegenden Objekten dargestellt.

5.1.3 Vergleich der verschiedenen Versionen der Helfer

In Abbildung 33 werden die visuellen Helfer von Cues und Advanced Cues in ihrer hauptsächlichen Hilfsfunktion verglichen. Die Hilfsfunktionen sind zwischen Pick und Place aufgeteilt, da Advanced Cues spezielle visuelle Helfer jeweils für Pick und Place besitzt. Bei Cues wird kein Unterschied zwischen Pick oder Place gezogen, daher wiederholen sich dort die visuellen Helfer. Bei Advanced Cues wurde das autonome Verhalten während des Pick oder Place Ablaufs als visueller Helfer gezählt, da das automatische Finden der optimalen Höhe zum Greifen oder Platzieren einen visuellen Helfer, der beim vertikalen Positionieren hilft, obsolet macht. Innerhalb der Tabelle ist darauf zu achten, dass beim Gripper-Bereich bei Cues und Advanced Cues die jeweiligen unterschiedlich dargestellten Gripper-Bereiche gemeint sind. Das heißt, dass bei Cues auf den Kreis und bei Advanced Cues auf das Rechteck referiert wird.

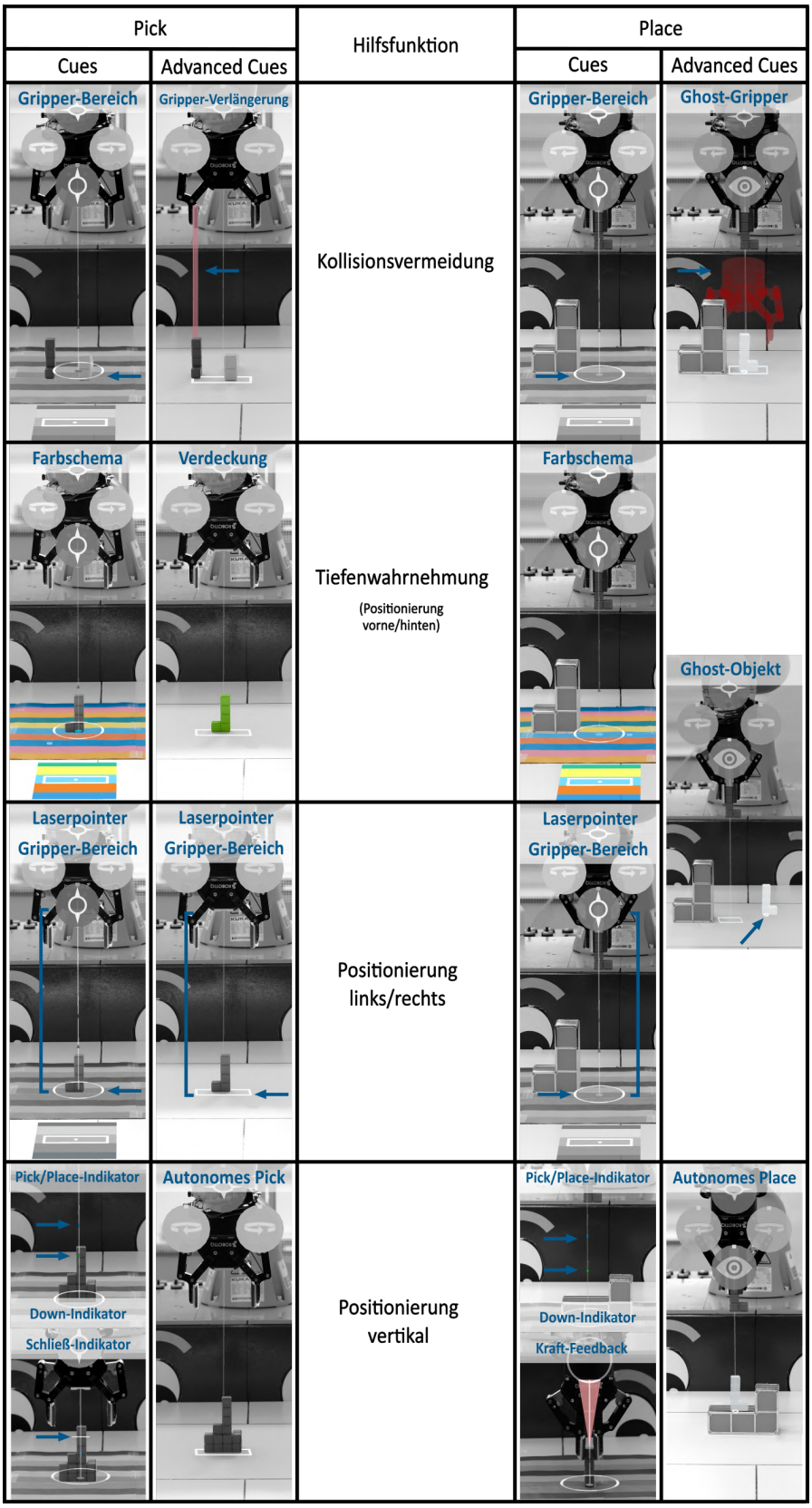


Abbildung 33: Gegenüberstellung der visuellen Helfer von Cues und Advanced Cues abhängig von der hauptsächlichen Hilfsfunktion.

5.2 Hypothesen

Die Studie untersuchte die zentrale Fragestellung, in welchem Maß die Cues im Vergleich zu No Cues und Advanced Cues bei der Durchführung einer herausfordernden Pick und Place Aufgabe abschneiden. Dabei dient No Cues als Ausgangswert, mit dem Cues und Advanced Cues verglichen werden können. Die Cues wurden unter der Annahme entwickelt, dass sie gegenüber No Cues den Benutzer mehr unterstützen und entlasten. Dabei wird davon ausgegangen, dass Advanced Cues dem Benutzer am meisten unterstützt, da das autonome Verhalten bei Pick und Place Aufgaben ihm Arbeit abnimmt und damit Fehler verhindert. Der Vergleich von Cues mit No Cues und Advanced Cues widmet sich der Fragestellung, ob die Cues besser abschneiden, als No Cues und inwieweit die Leistung von Cues mit der von Advanced Cues zu vergleichen ist. Diese lässt sich in folgende Hypothesen zusammenfassen. Die Maße für die Hypothesen sind im nächsten Kapitel erläutert.

1. Probanden schneiden bei der Durchführung einer anspruchsvollen Pick und Place Aufgabe im Hinblick auf Fehlerrate, Durchführungszeit und Präzision mit Cues besser ab als No Cues und schlechter als Advanced Cues.

2. Die Usability des Prototyps mit Cues ist besser als No Cues und schlechter als Advanced Cues.

3. Die mentale Arbeitsbelastung mit Cues ist niedriger als No Cues und höher als Advanced Cues.

5.3 Methodik

Die Studie wurde mit Menschen ohne körperliche Behinderungen durchgeführt, da das grundlegende Konzept der visuellen Helfer getestet werden sollte. Eine Studie mit Tetraplegikern, der eigentlichen Zielgruppe des Prototyps, bringt mehrere Schwierigkeiten mit sich. Einerseits ist es sehr schwierig eine genügend große Anzahl von Probanden zu finden. Andererseits ist Tetraplegie eine sehr individuelle Erkrankung mit unterschiedlichem Schweregrad und damit verbundenen verschieden starken Einschränkungen. Diese Individualität würde zu großen Varianzen in den Daten führen, welche sich nicht auf die unterschiedlichen Prototypen zurückführen lassen und etwaige Unterschiede zwischen diesen überdecken könnte. Bei der Durchführung der Studie wurde dennoch darauf geachtet, dass die Probanden bei der Bedienung des Prototyps eine feste sitzende Position einnehmen und diese auch beibehalten, vergleichbar zu einem Tetraplegiker in seinem Rollstuhl.

Die Studie wurde als ein *Between-Subject-Design* konzipiert. Das bedeutet, dass jeder Proband nur eine Version der visuellen Helfer, also No Cues, Cues oder Advanced Cues, getestet hat. Dadurch muss ein Proband nicht alle drei Versionen der Cues ausprobieren und die Dauer des Experimentes ist deutlich kürzer. Außerdem könnte das Verwenden der unterschiedlichen Versionen der visuellen Helfer direkt hintereinander die Probanden verwirren und zu ungewollten Übertragungseffekten führen, wie z. B. das Übertragen von bestimmten Strategien. Zusätzlich können die Probanden die Versionen nicht untereinander vergleichen und bewerten sie daher unabhängiger. Gleichzeitig ist dies auch ein Nachteil, da die Probanden nicht ihre subjektive Meinung zu den Unterschieden der Versionen mitteilen können. Ein weiterer Nachteil dieser Methode besteht darin, dass dafür relativ viele Probanden benötigt werden.

Die unabhängige Variable dieser Studie entspricht den verschiedenen Versionen der visuellen Helfer. Mit jeder Version der Helfer wurde die gleiche anspruchsvolle Pick und Place Aufgabe, die in Kapitel 5.5.1 erläutert ist, in dreimaliger Wiederholung durchgeführt. Dabei wurden objektive und subjektive Daten gesammelt. Die objektiven Daten wurden während der Durchführung der Aufgabe gemessen und sind im Folgenden aufgelistet:

- Aufgabenzeit: Die Zeit, die der Proband für die Durchführung der Aufgabe gebraucht hat.
- Fehlerrate: Wie viele Fehler hat der Proband während der Durchführung der Auf-

gabe gemacht. Als Fehler gelten das ungewollte Berühren oder Umstoßen eines Objektes, ein fehlgeschlagener Greifversuch oder das Auslösen eines Sicherheitsabbruches.

- Präzision: Der Abstand von der Position des Grippers auf der y- und x-Achse des Roboters zur Position des Objektes beim Greifen oder der optimalen Platzierposition beim Platzieren.

- Anzahl der Befehle: Wie viele Sprachbefehle und Buttons wurden aktiviert, um ein Objekt zu greifen oder zu platzieren. Dabei wurde die wiederholte Aktivierung des Rotationsbuttons als ein Befehl zusammengefasst.

- Feinjustierzeit: Beschreibt die Zeit, die der Proband, vom *Move* bis zum *Pick* oder *Place* Sprachbefehl benötigte. Diese Zeit beschreibt, wie lange der Proband zum Feinjustieren gebraucht hat.

Die abhängigen Variablen sollen die Leistung der Probanden bei der Durchführung der Aufgabe festhalten. Die Präzision, Anzahl der Befehle und Feinjustierzeit wurden von der Applikation aufgenommen und nach dem Experiment ausgelesen. Aufgabenzeit und Fehlerrate wurden während des Experimentes vom Leiter des Experimentes gemessen. Die restlichen Daten wurden während der Durchführung notiert. Zusätzlich zu diesen objektiven Daten wurden auch subjektive Daten in Form von Fragebögen erhoben, die der Proband nach Durchführung der Aufgabe ausgefüllt hat. Der Nasa Task-Load-Index, kurz Nasa-TLX, wurde verwendet, um die Arbeitsbelastung bei der Durchführung der Aufgabe mit den unterschiedlichen visuellen Helfern zu messen. Dieser Index wird auf einer Skala von 0 bis 100 berechnet und setzt sich aus sechs Kategorien zusammen: geistige Arbeitsbelastung, körperliche Arbeitsbelastung, zeitliche Arbeitsbelastung, Leistung, Anstrengung und Frustration. Jede dieser Kategorien wird auf einer Skala von 0 bis 100 bewertet. Der Nasa-TLX Fragebogen, welcher in der Studie verwendet wurde, ist im Anhang B.2 zu finden. In der Raw-Variante des Nasa-TLX werden die Kategorien einheitlich gewichtet und zur Berechnung der durchschnittlichen Gesamtarbeitsbelastung verwendet (Hart, 2006).

Der Usefulness, Satisfaction und Ease of Use Fragebogen (USE Fragebogen) untersucht die Usability des Prototyps unter Verwendung der verschiedenen visuellen Helfer. Er untersucht vier Dimensionen der Usability: Nützlichkeit, Bedienfreundlichkeit, Lernfreundlichkeit und Zufriedenheit. Der USE Fragebogen wurde als Sieben-Punkte-Likert-Bewertungsskalar konstruiert, von 1 (stimme gar nicht zu) bis 7 (stimme voll zu). Die Probanden bearbeiteten eine verkürzte Fassung mit insgesamt 22, statt 30 Fragen, die in

die vier Dimensionen aufgeteilt sind (Lund, 2001). Der verkürzte USE Fragebogen wurde fälschlicherweise aus einem vorherigen Experiment wiederverwendet und ist im Anhang B.3 zu finden.

5.4 Probanden

Es war geplant, dass insgesamt 36 Probanden an der Studie teilnehmen und somit jede Version der visuellen Helfer von 12 Probanden getestet wird. Jedoch wurde aufgrund des Ausbruchs von COVID-19 und den bundesweiten Sicherheitsmaßnahmen die Studie nach 23 Probanden abgebrochen. Da eine Fortführung der Versuche mit der noch fehlenden Anzahl an Probanden zeitlich nicht abzuschätzen war, wurde die Auswertung mit den schon vorhandenen Daten durchgeführt. Dies führte zu einer ungleichen Anzahl von Probanden zwischen den Versionen, bei der jeweils sieben Probanden No Cues und Advanced Cues, aber neun Probanden Cues testeten. Die Probanden waren entweder Studenten oder Mitarbeiter der Westfälischen Hochschule Gelsenkirchen und haben freiwillig am Experiment teilgenommen. Die Aufteilung zwischen den Geschlechtern, war mit insgesamt 4 weiblichen und 19 männlichen Probanden nicht gleich verteilt. Dies hatte zur Folge, dass keine weiblichen Probanden bei No Cues, nur eine Advanced Cues und drei Cues getestet wurden. Die Alterspanne der Probanden umfasst 19 bis 55 Jahre und beträgt im Mittel 29 Jahre, dies entspricht auch dem durchschnittlichen Alter innerhalb der verschiedenen Versionen der visuellen Helfer, No Cues (M = 29,57, SD = 11,44), Cues (M = 28,89, SD = 4,73) und Advanced Cues (M = 29,71, SD = 11,91). Mit Hilfe des ersten Fragebogens, wurde das Hintergrundwissen der Probanden ermittelt. In den Diagrammen in Abbildung 34 ist zu sehen, wie sich dieses Hintergrundwissen zwischen den verschiedenen Versionen der Cues aufteilt. Insgesamt hatten 19 von 23 der Probanden schon Erfahrungen mit VR oder AR gesammelt. Dies teilt sich unter den verschiedenen Versionen der Cues vergleichbar auf (No Cues: 5 von 7, Cues: 8 von 9, Advanced Cues: 6 von 7). Direkte Erfahrung mit der HoloLens besaßen insgesamt 12 der 23 Probanden und wie in Abbildung 34 zu erkennen ist, verteilten sich diese ungleich zwischen den Versionen der visuellen Helfer. 6 der 7 Probanden, die Advanced Cues testeten, hatten schon Vorerfahrung mit der HoloLens, während nur 4 von 9 bei Cues und 2 von 7 bei No Cues Erfahrung vorweisen konnten. Insgesamt 14 der 23 Probanden hatten schon einmal mit einem Roboter gearbeitet oder ihn programmiert. Dies teilt sich jeweils mit 5 von 7 Probanden gleich zwischen No Cues und Advanced Cues auf, im Gegensatz zu 4 von 9 bei Cues.

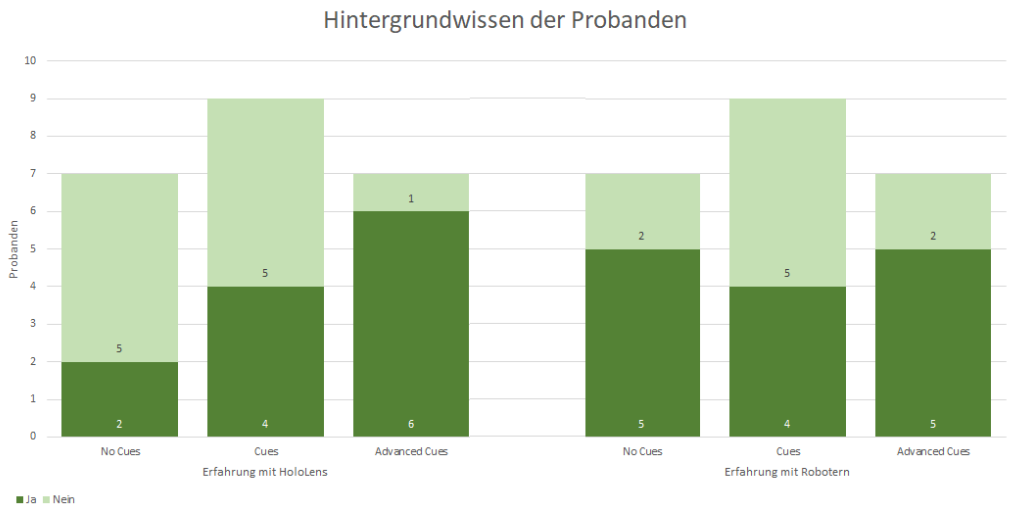


Abbildung 34: Hintergrundwissen der Probanden im Bezug auf Erfahrungen mit der HoloLens und Erfahrungen mit Robotern.

5.5 Versuchsaufbau

Alle Probanden, die an der Studie teilnahmen, absolvierten denselben Versuchsablauf und dieselbe anspruchsvolle Pick und Place Aufgabe. Dabei wurde darauf geachtet, dass alle Probanden die Studie unter vergleichbaren Bedingungen durchführten und sich nur die Version der visuellen Helfer unterschied. Um das zu gewährleisten, wurde bei den Advanced Cues die Position der Objekte nicht mit einem Objekterkennungsverfahren erkannt, sondern im Voraus in den Prototyp implementiert. Aufgrund von inkonsistentem Tracking der Objekte zum Zeitpunkt der Studie, konnte es beim Erkennen der Objekte zu unterschiedlichen Fehlern kommen. Durch den festen Aufbau und Ablauf der Aufgabe dieser Studie konnten die Positionen der Objekte vorher bestimmt und in den Prototyp der Advanced Cues integriert werden. Dies erlaubt eine sehr gute Simulation der Objekterkennung und die Probanden konnten die Advanced Cues unter gleichen Bedingungen testen.

5.5.1 Aufgabe

Die Aufgabe wurde so gewählt, dass sie eine möglichst realistische Situation in kontrollierter Form nachstellt. Da die visuellen Helfer vor allem bei einem fehlenden Perspektivwechsel unterstützen sollen, konzentrierte sich die Aufgabe darauf, Probleme einer festen Perspektive hervorzurufen. Ein solches Problem stellt sich zum Beispiel dar, wenn ein Objekt hinter einem Anderem platziert werden soll und es daher teilweise verdeckt wird. Die Aufgabe besteht aus insgesamt drei Objekten, die gegriffen und platziert wer-

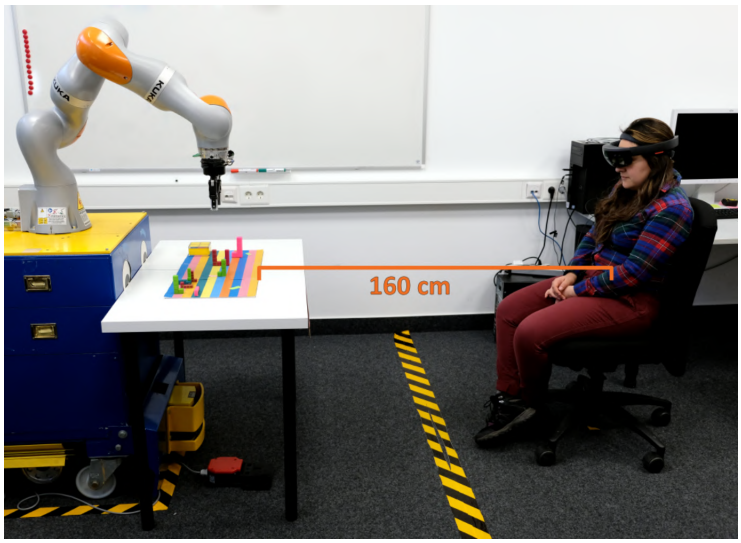
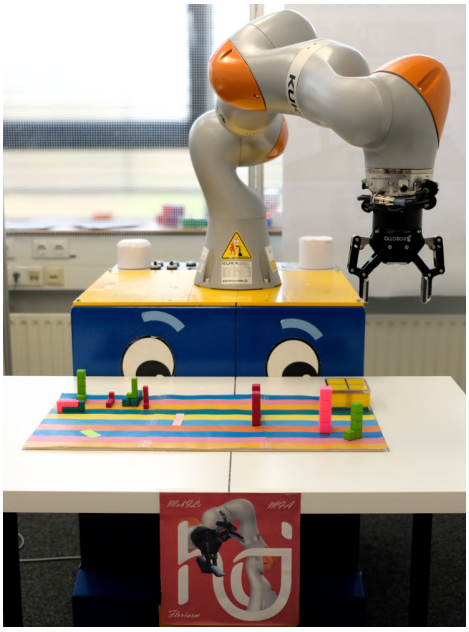
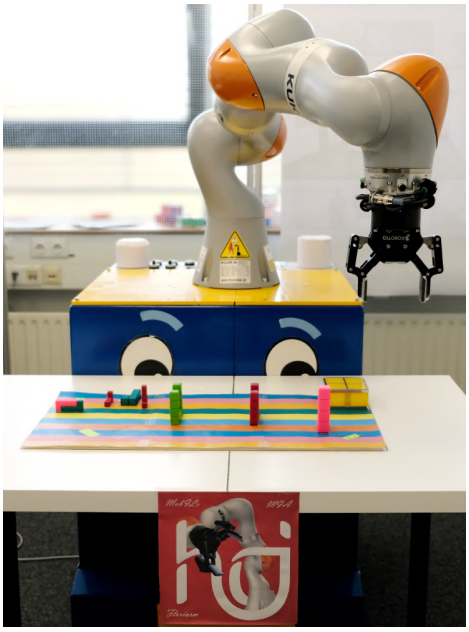


Abbildung 35: Versuchsaufbau mit der festen Sitzposition der Probanden, die 160 cm vom Anfang des Arbeitsbereiches entfernt ist.

den müssen. Der Prototyp sollte nicht an einem speziellen Montageprozess, sondern an einem möglichst breiten Umfang von Pick und Place Aufgaben getestet werden. Daher wurde die Aufgabe möglichst allgemein und die Objekte abstrakt gehalten. Die Objekte sind L-förmig mit unterschiedlichen Farben und drei verschiedenen Größen. Der Aufbau der Aufgabe ist in Abbildung 36 aus der Perspektive der Probanden dargestellt. Nur grüne Objekte sollen vom Probanden gegriffen und platziert werden. Roboter und Gripper dürfen die anderen Objekte nicht berühren. Der Proband befand sich während der gesamten Durchführung auf einer festen Sitzposition, die sich frontal vor dem Roboter befindet. In Abbildung 35 ist die feste Sitzposition, die 160 cm vom Anfang des Arbeitsbereiches und 185 cm vom Roboter entfernt ist, dargestellt. Dies entspricht einer realistischen Sitzposition für einen Montagearbeitsplatz, da sie sich gerade außerhalb der Reichweite des Roboters befindet. Um den Ablauf der Aufgabe besser nachvollziehen zu können, ist in Abbildung 37 die Positionen der Objekte 37a vor, sowie 37b nach der Aufgabendurchführung dargestellt. Die Nummern auf den Objekten zeigen die Reihenfolgen an, in denen mit den Objekte Interagiert wird. Das erste Objekt ist leicht rotiert und in seiner Nähe steht ein großes pinkes Objekt. Die Probanden müssen beim Greifen des ersten Objektes aufpassen, dass sie nicht mit dem Gripper das pinke Objekt berühren oder umstoßen. Nachdem das Objekt gegriffen wurde, soll es auf der vorderen pinken Markierung platziert werden. Das zweite Objekt befindet sich umringt von kleinen Objekten, die ebenfalls eine Kollisionsgefahr beim Greifen darstellen. Außerdem verdecken die beiden kleineren Objekte, die vor dem Zielobjekt platziert sind, die untere Hälfte des zweiten Objekts und stören dadurch beim Einschätzten seiner Position. Nachdem das zweite Objekt gegriffen wurde, soll dieses auf die zweite pinke Markierung, direkt hinter dem ersten Objekt, platziert



(a) Aufgabenaufbau aus der Perspektive des Probanden.



(b) Position der Objekte nach erfolgreichen Durchführen der Aufgabe aus der Perspektive des Probanden.

Abbildung 36: Position der Objekte aus der Perspektive des Probanden (a) bevor die Aufgabe gestartet ist und (b) nach erfolgreicher Durchführung der Aufgabe.

werden. Dies ist schwierig, da das erste Objekt das Zweite beim Positionieren teilweise verdeckt. Das dritte und letzte Objekt steht erhöht auf einem anderen Objekt. Durch die Erhöhung müssen die Probanden den Gripper präzise vertikal Positionieren, während sie auf Objekte, die sich in unmittelbarer Nähe befinden, achten müssen. Ist das dritte Objekt gegriffen, soll es zwischen die beiden roten Objekte platziert werden. Dies erfordert, dass der Gripper richtig rotiert ist, so dass er beim Öffnen nicht die roten Objekte umwirft. Der Abstand zwischen den beiden roten Objekten lässt wenig Spielraum für Fehlpositionierungen. Zusätzlich ist das vertikale Positionieren erschwert, da die Höhe des gegriffenen Objekts durch das vordere rote Objekt verdeckt wird. Das Greifen der drei Objekte ist in etwa gleich schwer gestaltet, jedoch nimmt die Schwierigkeit beim Platzieren der Objekte zu.

(a) Position der Objekte vor dem Start der Aufgabe.

(b) Position der Objekte nach erfolgreicher Durchführung der Aufgabe

Abbildung 37: Aufgabenaufbau (a) vor und (b) nach der Aufgabe. Die Kennzeichnung der Objekte entspricht der Reihenfolge.

5.5.2 Versuchsablauf

Die Studie besaß einen festen Ablauf, der im Folgenden aufgelistet ist:

1. Begrüßung
2. Vorstellung des Roboters
3. Ausfüllen eines Fragebogens zu Demografie und Hintergrundwissen (siehe Anhang B.1)
4. Einführung in den Prototyp mittels Videoausschnitte und mündlichen Erklärungen
5. Übungsphase mit dem Prototyp
6. Demonstration der Aufgabe
7. Durchführung der Aufgabe, insgesamt drei Mal hintereinander
8. Ausfüllen der Fragebögen
9. Abschließendes Interview (siehe Anhang B.4)

Den Probanden wurden, nach einer kurzen Begrüßung, der Roboter und seine Komponenten kurz erläutert. Dadurch sollten Begrifflichkeiten, wie Gripper oder Gripperfinger, erklärt werden, da sie im weiteren Verlauf häufiger verwendet werden. Außerdem wurde den Probanden erläutert, dass der Roboter sie nicht verletzen und sie weder den Roboter noch etwas im Arbeitsbereich beschädigen können. Eine Einführung bestehend aus Videoausschnitten und einer mündlichen Erläuterung, hat die Probanden mit dem Prototyp

und den jeweiligen visuellen Helfer vertraut gemacht. Die Videoausschnitte wurden direkt von der HoloLens aufgenommen und zeigen, wie die Interaktion und die visuellen Helfer funktionieren. Mit unterstützenden mündlichen Erläuterungen wurden alle Aspekte der Interaktion und der visuellen Helfer erklärt. Damit die Probanden die Sprachbefehle nicht auswendig lernen mussten, wurde ihnen ein Ausdruck mit den Sprachbefehlen übergeben. Diesen konnten sie während der Durchführung der Aufgabe nutzen. Nach der Einführung konnten die Probanden in einer Übungsphase das Gesehene festigen. Ziel dieser Übungsphase war es, dass die Probanden jede Funktionalität des Prototyps ausprobieren. Damit sie nicht an der eigentlichen Aufgabe üben konnten, waren während der Übungsphase nur zwei Objekte im Arbeitsbereich aufgestellt. Damit sollten die Probanden trainieren, ein Objekt zu greifen, zu rotieren und es auf eine Markierung zu stellen. Außerdem sollten sie ein rotiertes Objekt greifen und auf ein anderes Objekt stellen. Nach der erfolgreichen Durchführung, sowie der mindestens einmaligen Verwendung von jedem Button und Sprachbefehl, konnten sich die Probanden entscheiden, ob sie noch etwas ausprobieren oder mit der eigentlichen Aufgabe starten möchten. Nach Beenden der Übungsphase konnten die Probanden die HoloLens absetzen, während ihnen die Aufgabe erklärt wurde. Die Pause von der HoloLens sollte die Probanden entlasten und das Tragen der HoloLens nicht zu unangenehm werden lassen. Die Aufgabe wurde den Probanden erläutert und gleichzeitig visualisiert, indem mit den Händen der Gripper und wie sie mit ihm die Objekte greifen und platzieren müssen, nachgeahmt wurde. Danach hatten die Probanden 30 Sekunden Zeit, über die Aufgabe und der besten Umsetzung nachzudenken. Im Weiteren führten sie die Aufgabe drei Mal hintereinander durch. Anschließend füllten sie die Fragebögen aus und es wurde ein kurzes Interview gehalten. Am Ende wurde den Probanden eine Aufwandsentschädigung in Höhe von 7€gezahlt.

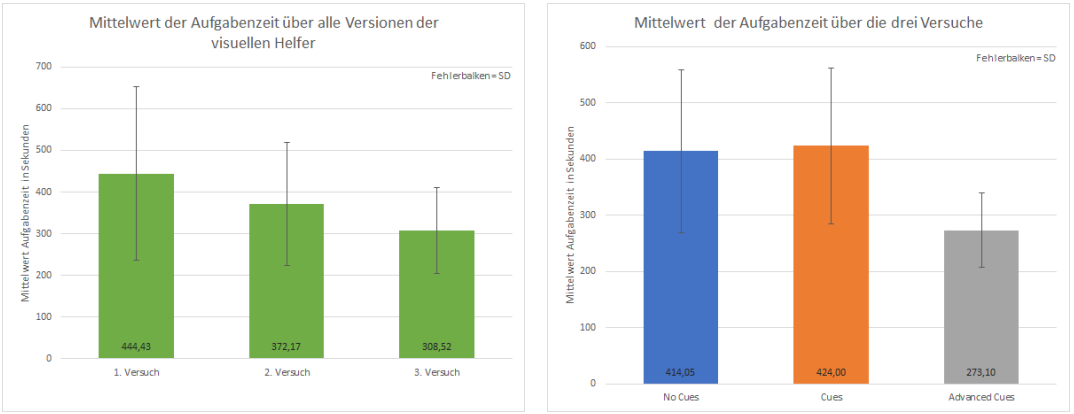
5.6 Analyse

Die Probanden hatten drei Versuche die Aufgabe durchzuführen. Die Aufgabe lässt sich in Teilaufgaben unterteilen, die den drei verschiedenen Objekten entsprechen. Jedes Objekt musste gegriffen und platziert werden, daher lassen sich die Objekte in die Unteraufgaben Pick und Place unterteilen. Diese Variablen werden im weiteren Verlauf wie folgt benannt:

- Versuche = 1. Versuch, 2. Versuch, 3.Versuch
- Objekte = 1. Objekt, 2. Objekt, 3. Objekt
- Unteraufgaben = Pick, Place

Die abhängigen Variablen Aufgabenzeit und Fehlerrate wurden jeweils pro Versuch gemessen. Daher wurden diese beiden Variablen mit einer ANOVA mit Messwiederholung über die Versuche analysiert. Die anderen abhängigen Variablen Präzision, Anzahl der Befehle und Feinjustierzeit wurden jeweils für das Platzieren und Greifen von allen drei Objekten über alle drei Versuche gemessen und daher mit einer ANOVA mit Messwiederholung über die drei Versuche, die drei Objekte und über Pick und Place untersucht (Girden, 1992). Neben der ANOVA wurden Bonferroni korrigierte Post-hoc-Tests verwendet, um den Mittelwert paarweise zu vergleichen (Bland & Altman, 1995). Für die nicht parametrischen Untersuchungen der beiden Fragebögen wurde ein Kruskal-Wallis-Test verwendet (Kruskal & Wallis, 1952). Bei der Analyse lag das Signifikanzniveau (Irrtumswahrscheinlichkeit) bei $\alpha = 0,005$.

5.6.1 Analyse der Aufgabenzeit und Fehlerrate



(a) Mittelwert der Aufgabenzeit der drei Versuche über die verschiedenen Versionen der visuellen Helfer.

(b) Mittelwert der Aufgabenzeit der verschiedenen Versionen der visuellen Helfer über alle drei Versuche.

Abbildung 38: Mittelwert der Aufgabenzeit (a) über die Versuche und (b) über die drei Versionen der visuellen Helfer mit Standardabweichung.

Die Probanden führten die Aufgabe drei Mal hintereinander durch und wurden pro Aufgabendurchlauf schneller. Eine ANOVA mit Messwiederholung der drei Versuche über alle Versionen der visuellen Helfer zeigte einen signifikanten Unterschied zwischen den Versuchen F(2,40) = 9,776; p < 0,001. Die Probanden lernten bei jedem Versuch etwas dazu und wurden dadurch schneller. Der Lerneffekt bei der Aufgabenzeit ist in Abbildung 38a über alle Versionen der visuellen Helfer verdeutlicht. Die ANOVA mit Messwiederholung über die Versuche zeigt keine statistisch signifikante Interaktion zwischen den Versuchen und den verschiedenen Versionen der visuellen Helfer, F(4,40) = 0,577; p =

0,681. Daher ist von einem vergleichbaren Lerneffekt aller Versionen der visuellen Helfer auszugehen und deshalb wird im Weiteren über alle Versuche analysiert. In Abbildung 38b ist der Mittelwert der Aufgabenzeit der verschiedenen Versionen der visuellen Helfer über die drei Versuche mit Standardabweichung zu erkennen. Probanden, die No Cues und Cues verwendeten, haben im Mittel in etwa 7 Minuten und die, die Advanced Cues verwendeten 4,5 Minuten für eine Durchführung der Aufgabe benötigt: Advanced Cues (M = 237,1 s; SD = 66,23), No Cues (M = 414,05 s; SD = 145,04) und Cues (M = 424 s; SD = 138,64). Jedoch zeigen Bonferroni korrigierte Post-hoc-Tests mit paarweisen Mittelwertvergleichen keinen signifikanten Unterschied zwischen den Versionen der visuellen Helfer auf. Auch bei der Fehlerrate ergaben Bonferroni korrigierte Post-hoc-Tests mit paarweisen Mittelwertvergleichen keinen signifikanten Unterschied zwischen den Versionen der visuellen Helfer. Die Fehlerrate lag bei No Cues und Cues im Mittel in etwa bei zwei Fehlern und bei Advanced Cues bei etwa einem Fehler: Advanced Cues (M = 1,1; SD = 0,74), No Cues (M = 2,1; SD = 0,66) und Cues (M = 2,26; SD = 0,83). Damit lässt sich ein Trend erkennen, bei dem Probanden, die Advanced Cues verwendeten, schneller die Aufgabe lösten und dabei weniger Fehler machten, als Probanden mit No Cues oder Cues.

5.6.2 Analyse der Präzision

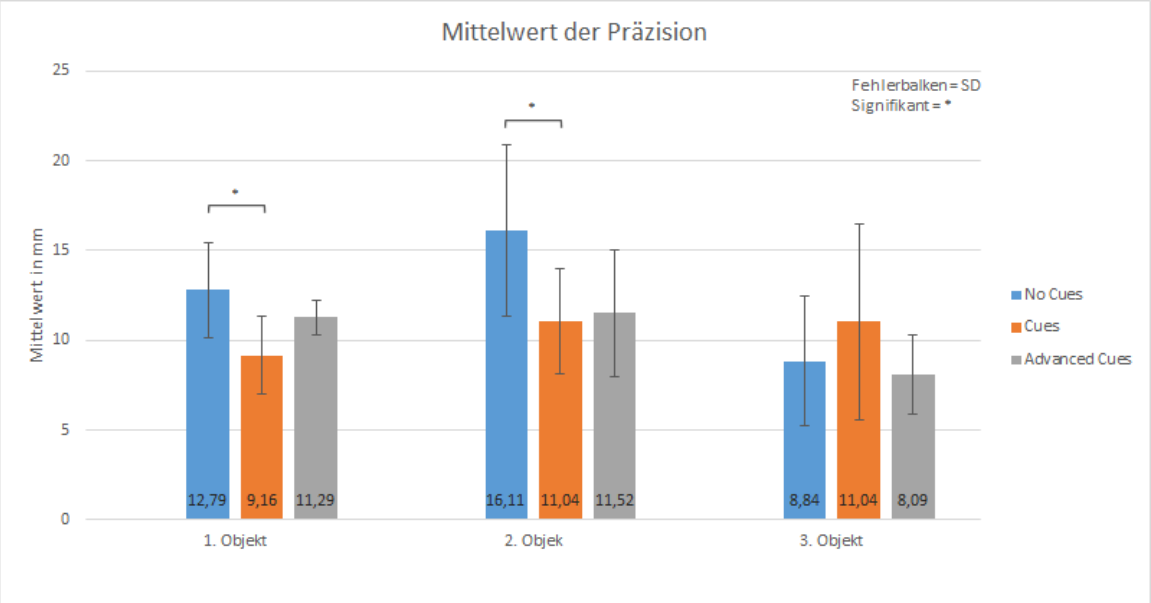


Abbildung 39: Mittelwert der Präzision in mm bei den drei verschiedenen Objekten über alle drei Versuche mit Standardabweichung. Die Präzision entspricht dem Abstand des Grippers vom Objekt oder zur perfekten Platzierposition, als das Objekt gegriffen oder platziert wurde.

Im Diagramm in Abbildung 39 ist der Mittelwert der Präzision über die drei Objekte mit Standardabweichung dargestellt. Die ANOVA mit Messerwiederholungen über die Messergebnisse der Präzision ergab eine statistisch signifikante Interaktion zwischen den Objekten und den Versionen der visuellen Helfern, $F(4, 40) = 3{,}27$; $p = 0{,}021$. Die Versionen der visuellen Helfer unterscheiden sich beim ersten Objekt ($p = 0{,}008$) und beim zweiten Objekt ($p = 0{,}032$) signifikant. Gemäß eines anschließenden Bonferroni Post-Hoc-Tests mit paarweisen Mittelwertvergleichen unterschied sich Cues von No Cues signifikant zum ersten Objekt ($t = -3{,}63$; $p = 0.007$) und zum zweiten Objekt ($t = -5{,}07$; $p = 0{,}043$). Damit waren Probanden, die Cues verwendet haben, präziser beim Greifen und Platzieren des ersten und zweiten Objektes, aber nicht beim dritten Objekt. Das dritte Objekt unterscheidet sich von den ersten Beiden stärker, da es von einer erhöhten Position gegriffen und zwischen zwei anderen Objekten platziert werden musste. Dadurch hatten die Probanden gerade beim Platzieren durch die beiden anderen Objekte mehr visuelle Anhaltspunkte. Der gesamte Mittelwert über die Versuche und über die Objekte spiegelt wieder, dass Probanden mit Advanced Cues ($M = 10{,}3$; $SD = 1{,}5$) und Cues ($M = 10{,}41$; $SD = 2{,}09$) vergleichbar abschnitten und sie im Mittel präziser waren als Probanden, die No Cues ($M = 12{,}58$; $SD = 2{,}92$) verwendeten.

5.6.3 Analyse der Anzahl der Befehle und Feinjustierzeit

Eine ANOVA mit Messwiederholung über die Versuche, die Objekte sowie Pick und Place wurde auch für die Variablen Anzahl der Befehle und Feinjustierzeit durchgeführt. Es ergab sich kein statistisch signifikante Interaktion zwischen Objekten und Versionen der visuellen Helfer, Anzahl der Befehle $F(4, 40) = 2{,}37$; $p = 0{,}068$ und Feinjustierzeit $F(4, 40) = 2{,}18$; $p = 0{,}89$. Jedoch wurden aufgrund des Aufbaus des Experiments und den Unterschieden zwischen den Objekten auch diese beiden Variablen über die Objekte analysiert.

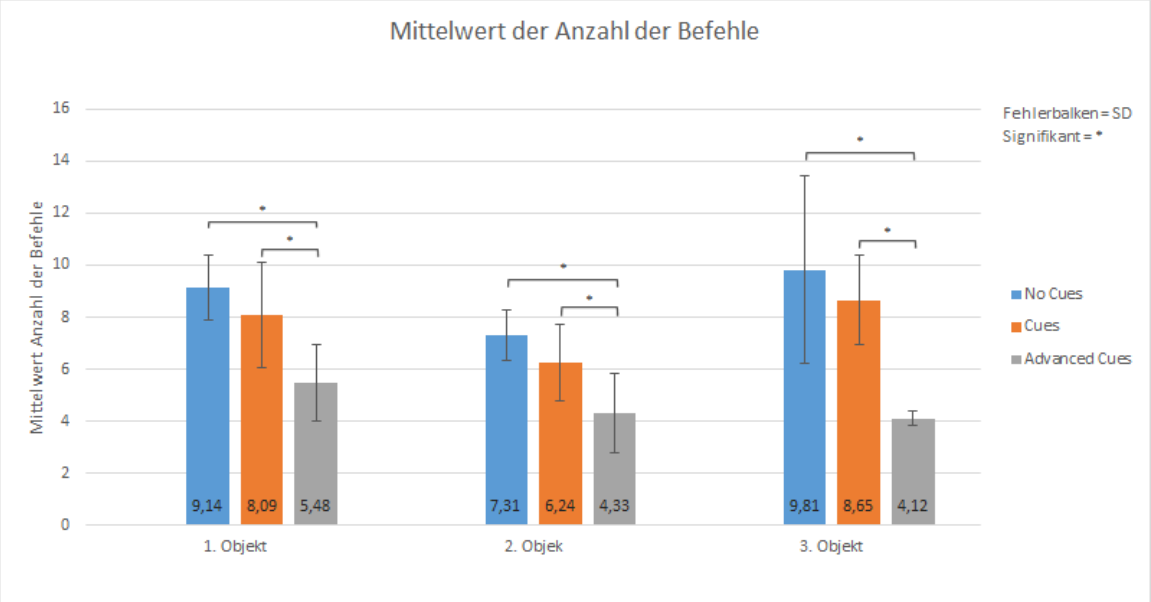


Abbildung 40: Mittelwert der Anzahl der Befehle bei den Objekten mit Standardabweichung.

Die mittlere Anzahl der Befehle bei den Objekten mit Standardabweichung ist in Abbildung 40 dargestellt. Die Versionen der visuellen Helfer unterscheiden sich signifikant beim ersten Objekt ($p = 0{,}002$), zweiten Objekt ($p = 0{,}002$) und beim dritten Objekt ($p < 0{,}001$). Anschließende Bonferroni Post-Hoc-Tests mit paarweisen Mittelwertvergleichen zeigten, dass sich beim ersten Objekt Advanced Cues zu No Cues ($t = -3{,}67$; $p = 0{,}002$) und zu Cues ($t = -2{,}62$; $p = 0{,}016$), beim zweiten Objekt Advanced Cues zu No Cues ($t = -2{,}98$; $p = 0{,}002$) und Cues ($t = -1{,}91$; $p = 0{,}033$), sowie beim dritten Objekt Advanced Cues zu No Cues ($t = -1{,}16$; $p < 0{,}001$) und Cues ($t = -4{,}53$; $p = 0.002$) signifikant unterschieden. Auch der gesamt Mittelwert über die drei Versuche und über die drei Objekte zeigt, dass die Anzahl der Befehle bei Advanced Cues ($M = 4{,}64$; $SD = 0{,}65$) geringer als bei Cues ($M = 7{,}66$; $SD = 1{,}51$) oder No Cues ($M = 8{,}75$; $SD = 1{,}71$) ist. Dies lässt sich dadurch erklären, dass bei Advanced Cues keine vertikale Positionierung vorgenommen

werden muss. Zum Beispiel ist es beim dritten Objekt, welches auf einem anderen Objekt steht und platziert werden muss, bei der Verwendung von Cues und No Cues unumgänglich, Befehle für die vertikale Positionierung zu verwenden. Darüber hinaus ist ein Trend zu erkennen, dass die Probanden mit Cues weniger Befehle nutzen mussten als mit No Cues.

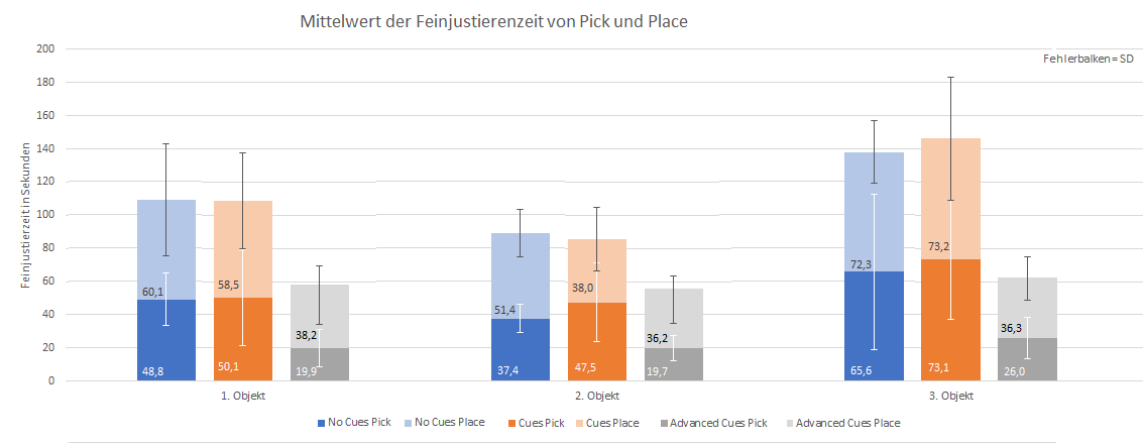


Abbildung 41: Mittelwert der Feinjustierzeit bei den Objekten aufgeteilt auf Pick und Place mit Standardabweichung.

In Kapitel 5.6.1 ist die Zeit, die Probanden pro Versuch gebraucht haben, beschrieben und diese Zeit wird mit der Feinjustierzeit genauer aufgeschlüsselt. In Abbildung 41 ist der Mittelwert der Feinjustierzeit bei den Objekten aufgeteilt auf Pick und Place mit Standardabweichung zu sehen. Die Versionen der visuellen Helfer unterscheiden sich nur beim dritten Objekt ($p = 0{,}013$) signifikant. Bonferroni Post-Hoc-Tests mit paarweisen Mittelwertvergleichen zeigen, dass sich beim dritten Objekt Advanced Cues zu No Cues ($t = -37{,}8$; $p = 0{,}05$) und Cues ($t = -42$; $p = 0{,}018$) signifikant unterscheidet. Der gesamte Mittelwert über die Versuche, über die Objekte und über Pick und Place folgt dem Trend der Analyse der Aufgabenzeit, da Advanced Cues ($M = 29{,}37$; $SD = 9{,}3$) sich von No Cues ($M = 55{,}94$; $SD = 19{,}91$) und Cues ($M = 56{,}74$; $SD = 22{,}34$) stark unterscheidet, jedoch Cues und No Cues sich ähneln.

5.6.4 Analyse des Nasa-TLX

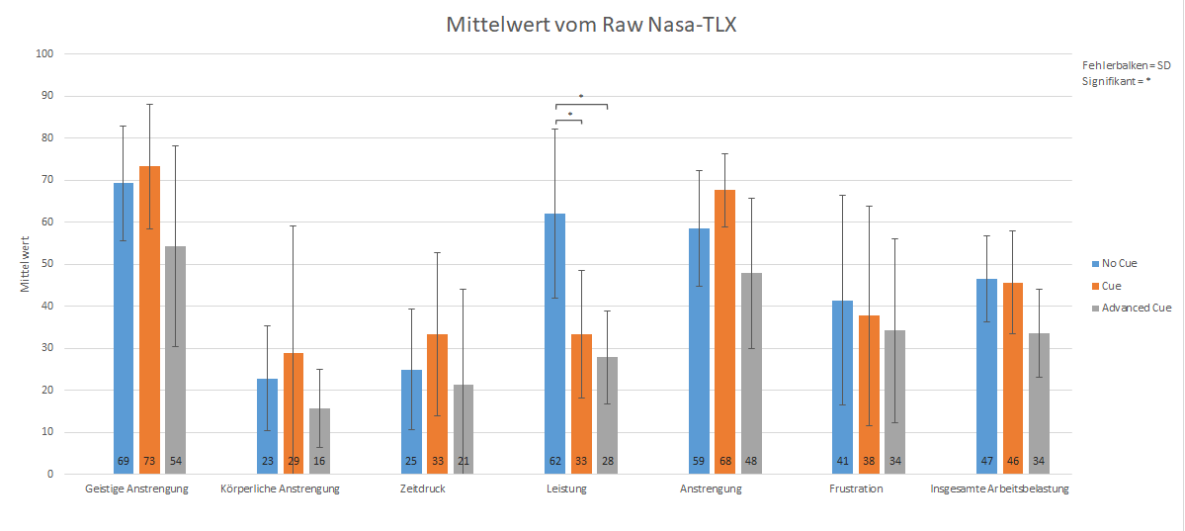


Abbildung 42: Auswertung der sechs Kategorien des Nasa-TLX und der daraus berechneten insgesamten Arbeitsbelastung mit Standardabweichung. Je geringer der Wert desto besser schneidet jede Kategorie ab.

In Abbildung 42 ist die Auswertung des Nasa-TLX über die sechs Kategorien und der daraus berechneten gesamten Arbeitsbelastung dargestellt. Ein Kruskal-Wallis-Test ergab, dass es nur in der Kategorie Leistung einen signifikanten Unterschied zwischen den Versionen der visuellen Helfer gab ($p = 0{,}008$). Anschließende Bonferroni korrigierte Post-hoc-Tests mit paarweisen Mittelwertvergleichen zeigen, dass sich nur Cues zu No Cues ($t = 2{,}55$; $p = 0{,}011$) und Advanced Cues zu No Cues ($t = 2{,}84$; $p = 0{,}005$) signifikant unterscheiden. Damit haben die Probanden, die No Cues verwendet haben, ihre Leistung schlechter eingeschätzt, als die, die Cues oder Advanced Cues verwendeten. Allgemein zeigt die Auswertung des Nasa-TLX, dass der Prototyp mit allen drei Versionen der Helfer geistig beanspruchend ist. Dies lässt sich unter anderem damit erklären, dass die Probanden den Prototyp das erste Mal verwendeten und gleich eine anspruchsvolle Aufgabe bearbeiten mussten. Jedoch liegt die gesamte Arbeitsbelastung im Mittelfeld des Spektrums, Advanced Cues ($M = 33{,}57$; $SD = 10{,}54$), Cues ($M = 45{,}72$; $SD = 12{,}22$) und No Cues ($M = 46{,}55$; $SD = 10{,}26$). Des Weiteren ist ein Trend zu erkennen, dass Advanced Cues in jeder Kategorie besser abschneidet, als Cues oder No Cues. Dies war zu erwarten, da Studien über Mensch-Roboter-Interaktion, die den Nasa-TLX verwenden, zeigen, dass die subjektiven Bewertungen der Arbeitslast mit zunehmender Autonomie abnimmt (Steinfeld et al., 2006).

5.6.5 Analyse des USE Fragebogens

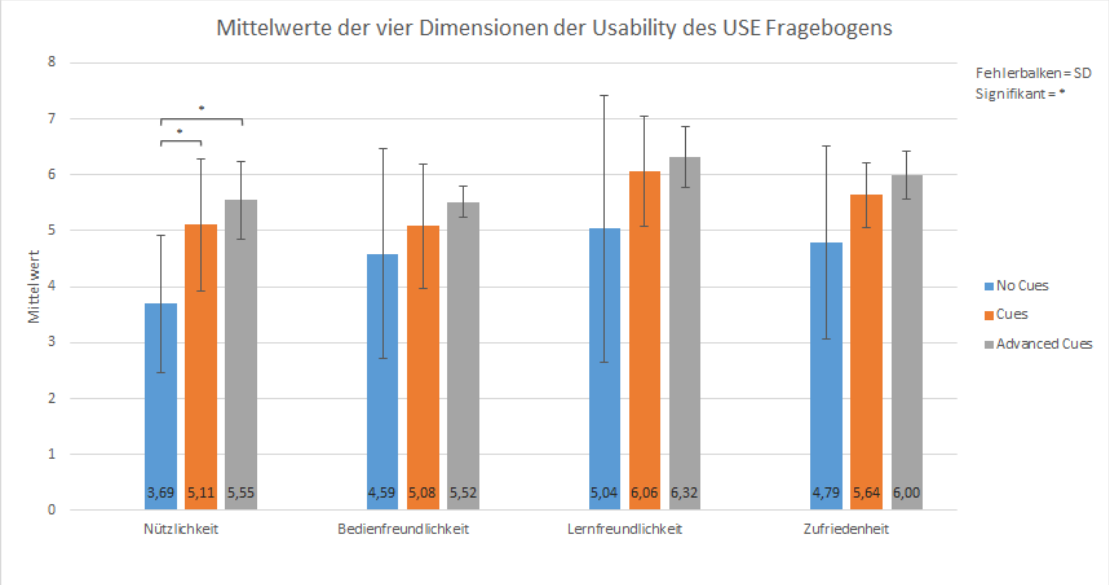


Abbildung 43: Auswertung der vier Dimensionen des USE Fragebogens mit Standardabweichung.

Die Mittelwerte der vier Dimensionen der Usability des USE Fragebogens sind in Abbildung 43 zu erkennen. Der Kruskal-Wallis-Test zeigte nur in der Dimension Nützlichkeit eine Signifikanz (p = 0,023). Die Bonferroni Post-hoc-Tests mit paarweisen Mittelwertvergleichen zeigen, dass sich No Cues und Cues (t = -2,13; p = 0,033) und No Cues und Advanced Cues (t = -2,61; p = 0,009) signifikant unterscheiden. Damit empfanden die Probanden die No Cues verwendeten den Prototyp insgesamt nicht so nützlich wie die, die Cues oder Advanced Cues benutzten. In den anderen Dimensionen Bedienfreundlichkeit, Lernfreundlichkeit und Zufriedenheit zeichnet sich der Trend ab, dass No Cues schlechter abschneidet, als Cues und Advanced Cues. Dies weist darauf hin, dass die visuellen Helfer die Usability im Bezug auf die vier Dimensionen verbessern.

5.6.6 Analyse der Interviews

Die Analyse der Interviews ergab unter anderem, dass unabhängig von den visuellen Helfern 17 der 23 Probanden die gleiche Strategie verfolgten. Diese bestand daraus, erst den Gripper zu rotieren, ihn dann zum Objekt zu navigieren und anschließend feinzujustieren. Dies deckt sich mit den Beobachtungen, die während des Experimentes bei den Probanden gemacht wurden. Außerdem gaben 17 Probanden, unabhängig von den visuellen Helfern, an, dass sie einen Drang verspürten, sich nach links oder rechts zu bewegen, um ihre Perspektive zu ändern. Nur 4 Probanden gaben an, dass sie erst diesen Drang verspürten,

dieser aber, je länger sie mit dem Roboter arbeiteten, abnahm. Auf die Frage, was den Probanden am meisten bei der Vervollständigung der Aufgabe half, waren die Antworten zwischen den Versionen der visuellen Helfer unterschiedlich. Bei No Cues antworteten 4 der 7 Probanden, dass ihnen nichts geholfen hat. Bei den Advanced Cues halfen 5 der 7 Probanden das Ghost-Objekt beim Platzieren und der Gripper-Bereich beim Greifen am meisten. Bei Cues gaben alle Probanden an, dass ihnen das Farbschema am besten geholfen hat. Dabei erwähnten einige, dass gerade die Tiefenwahrnehmung ein Problem für sie darstellte und das Farbschema sie dabei unterstützte. Es half ihnen gleichermaßen beim Platzieren und Greifen von Objekten. Viele Probanden merkten an, dass das Sichtfeld der HoloLens klein ist und sie dadurch nicht alle visuellen Helfer und Interaktionselemente gleichzeitig in ihrem Sichtfeld hatten. Zum Beispiel beschrieb ein Proband der Cues verwendete, dass er den Farb-Indikator, den er hilfreich fand, manchmal vergessen hatte, da er außerhalb seines Sichtfelds lag.

5.7 Limitationen

Alle Probanden hatten in etwa die gleiche Zeit für die Einführungs- und Probephase. Diese Einarbeitungszeit könnte die erhobenen Daten beeinflusst haben. Dadurch, dass Cues mehr visuelle Helfer als No Cues und Advanced Cues besitzt, hatten die Probanden mehr zu lernen und mussten sich an mehr erinnern. Die Aufnahmefähigkeit und Lernzeit können zwischen Probanden variieren. Zusätzlich hatten mehr Probanden, die No Cues und Advanced Cues verwendeten, schon einmal Erfahrungen mit Robotern sammeln können. Bei den Beobachtungen während des Experimentes wurde ein überdurchschnittlich souveräner Umgang mit dem Roboter vor allem bei den Probanden festgestellt, die No Cues verwendeten. Dadurch könnten die erhobenen Daten bei No Cues deutlich positiver ausgefallen sein. Daher bräuchte es mehr Teilnehmer für die Studie, um die in Kapitel 5.2 vorgestellten Hypothesen eindeutig zu verifizieren. Dies würde auch die hohe Standardabweichung, die vor allem bei den Fragebögen sehr hoch lag, senken. Zusätzlich könnten dadurch Ausreißer besser identifiziert werden, denn bei Cues und No Cues gab es jeweils ein Proband, die sehr lange für die Durchführung gebraucht haben. Diese würden sich vermutlich mit mehr Probanden als Ausreißer ausschließen lassen können.

5.8 Bewertung der Ergebnisse

Alle Probanden konnten mit Hilfe des Prototyps und den verschiedenen Versionen der visuellen Helfer den Roboter bedienen und die Aufgabe bearbeiten. Dabei hat die Mehrheit der Probanden mindestens einmal die Aufgabe für alle drei Objekte erfolgreich durchgeführt und das ungeachtet der kurzen Einführungs- und Probephase. Dass die Probanden den Roboter bedienen und eine anspruchsvolle Aufgabe lösen konnten, unabhängig von der Version der visuellen Helfer, spricht grundsätzlich für das Design des Prototyps. Es lassen sich Trends erkennen, dass Cues in Hinblick auf Präzision, Anzahl der Befehle und Usability besser abschneidet als No Cues, aber die beiden sich bei benötigter Zeit und Arbeitsbelastung ähneln. Die Probanden benötigten mit Cues genauso viel Zeit, wie mit No Cues, jedoch lässt der Unterschied in der Anzahl an Befehlen den Rückschluss zu, dass sie ihr Ziel mit weniger Aufwand erreichten. Hervorzuheben ist, dass Cues und Advanced Cues vergleichbare Ergebnisse bei der Präzision erzielten. Weitere Untersuchungen könnten zeigen, dass Probanden mit Cues genauso präzise arbeiten, wie Probanden mit Advanced Cues. In den anderen untersuchten Variablen schneidet Advanced Cues besser ab, als Cues und No Cues. Diese Ergebnisse waren zu erwarten, da die autonome Unterstützung den Probanden Arbeit abnimmt und sie somit entlastet. Jedoch ist dabei zu beachten, dass Advanced Cues nur simuliert und damit unter optimalen Voraussetzungen getestet wurde. Wie sich eventuelle Fehler und Ungenauigkeiten in der Objekterkennung auf die Benutzung der Advanced Cues auswirken, muss noch getestet werden.

6 Schluss

Mit dem im Rahmen dieser Masterarbeit entstandenen Augmented Reality Prototyp kann ein Roboter gesteuert werden, um mit ihm Montageaufgaben durchzuführen. Der Prototyp wurde für Tetraplegigker oder Menschen mit einer ähnlichen körperlichen Behinderung und für ihre speziellen Anforderungen entwickelt, um ihnen das Arbeiten an Montageplätzen zu ermöglichen. Mit dem entwickelten Prototyp steuert der Benutzer einen Roboterarm an einer festen Arbeitsstation mit der Microsoft HoloLens. Diese Arbeitsstation besteht aus dem Kuka iiwa Roboterarm, der innerhalb eines festgelegten Arbeitsbereiches bewegt werden kann. Eine Anforderung von Tetraplegigkern besteht in der Bedienung des Roboterarms ohne Hände. Daher wurden als Eingabemethoden Sprachbefehle und Kopfbewegungen gewählt. Die Microsoft HoloLens kombiniert diese Eingabemodalitäten mit der Darstellung von Augmented Reality in einem Gerät. Die Steuerung lässt eine feine manuelle Steuerung, sowie eine schnelle teilautomatisierte Steuerung durch den Benutzer zu. Damit wird eine flexible Interaktion geboten, welche sich an unterschiedlichste Montageaufgaben anpassen kann und dem Benutzer größtmögliche Kontrolle bietet. Tetraplegigker nehmen eine feste Perspektive auf die Arbeitsstation ein und können nicht durch z. B. zur Seite lehnen ihre Perspektive ändern. Daher wurden visuelle Helfer entwickelt, die das Blickfeld mit zusätzlichen Informationen anreichern. Die visuellen Helfer bestehen aus Informationen, die von der HoloLens und/oder dem Roboter stammen, und werden als visuelle Hinweise für den Benutzer in dessen Blickfeld augmentiert. Sie unterstützen den Benutzer bei der Tiefenwahrnehmung und bei der Einschätzung der Position des Grippers, um die Bedienung zu vereinfachen, Fehler zu vermeiden und den Benutzer zu helfen, den Roboter präziser zu steuern. Die visuellen Helfer haben keine festen Voraussetzungen an die Objekte oder der Arbeitsumgebung und passen sich daher flexibel an unterschiedlichste Aufgaben und Montageprozesse an.

Eine Studie evaluierte das Konzept der visuellen Helfer. Dabei wurden die in dieser Masterarbeit vorgestellten visuellen Helfer mit einer Version des Prototyps ohne jegliche visuelle Helfer als auch mit einer Version der visuellen Helfer, die auf Objekterkennung basieren, verglichen. Die auf Objekterkennung basierenden Helfer besitzen mehr Informationen, wie die Position und Orientierung von Objekten im Arbeitsbereich. Dadurch können sie gezieltere Unterstützung, sowie teilweise autonome Bewegungsabläufe anbieten. Insgesamt 23 Probanden benutzten jeweils nur eine Version der visuellen Helfer, um eine anspruchsvolle Pick und Place Aufgabe zu lösen. Die Studie zeigte Tendenzen auf, dass Benutzer mit visuellen Helfern präziser arbeiteten und weniger Befehle benötigten, als ohne Helfer. Außerdem lässt sich ein Trend erkennen, dass visuelle Helfer die Usa-

bility des Prototyps steigern. Jedoch unterscheidet sich die Version mit visuellen Helfern nicht in benötigter Zeit und Arbeitsbelastung von der Version ohne visuelle Helfer. Die auf Objekterkennung basierenden visuellen Helfer schneiden wie zu erwarten besser in den oben genannten Kategorien ab. Jedoch erreichen die visuellen Helfer bezogen auf die Präzision ein ähnliches Level, wie die auf Objekternennung basierenden visuellen Helfer.

6.1 Ausblick

Da die in dieser Masterarbeit durchgeführte Studie abgebrochen werden musste, wurde das Konzept der visuellen Helfer mit zu wenigen Teilnehmern getestet, um aussagekräftige Ergebnisse erzielen zu können. Daher wird empfohlen, die Studie weiterzuführen, um die bisher festgestellten Tendenzen zu validieren. Eine anschließende Studie mit Tetraplegikern oder Menschen mit einer ähnlichen körperlichen Behinderung könnte weiterführende Erkenntnisse, speziell über die Anforderungen und Ansprüche dieser Zielgruppe hervorbringen. Für eine solche Studie war der zeitliche Rahmen dieser Masterarbeit zu knapp, da sie deutlich längere Einführungen und ggf. Anpassungen des Prototyps an die individuellen Einschränkungen der Probanden mit sich ziehen würde. Für eine solche Studie könnte der Prototyp durch ein Hauptmenü, welches individuelle Einstellungen zulässt, ergänzt werden. Über diese Einstellungen könnten z. B. die Geschwindigkeit, in der die Buttons ausgelöst werden oder ähnliches verändert werden, um sich besser und schneller an individuelle Einschränkungen von Tetraplegikern anpassen zu können. Eine zusätzliche Erweiterung des Prototyps könnte über die Implementierung der Rotation des Grippers um die x- und y-Achse des Roboters stattfinden. Um den Prototyp in der Bedienung möglichst einfach zu halten, wurde nur die Drehung des Grippers um die z-Achse des Roboters umgesetzt. Die Erweiterung der Rotation würde die Anzahl der Interaktionselemente erhöhen, was eine Verlängerung der Einarbeitungszeit in den Prototyp nach sich ziehen würde. Der Benutzer könnte aber mehr Freiheitsgrade nutzen und den Roboter so universeller einsetzen.

Da die HoloLens 2 für diese Masterarbeit nicht zur Verfügung stand, wurde der Prototyp für die Microsoft HoloLens der ersten Generation entwickelt. Diese hat die Nachteile, dass sie relativ schwer ist und ein kleines Sichtfeld besitzt. Durch ihr Gewicht und dessen suboptimale Verteilung wird das Tragen der HoloLens schon nach 15 Minuten als unangenehm empfunden. Die HoloLens 2 verspricht eine bessere Gewichtsverteilung und ein größeres Sichtfeld. Daher wäre eine Portierung des Prototyps auf die HoloLens 2 vielversprechend. Aber auch neue aufkommende Innovationen im Bereich VR oder AR könnten eine geeignetere Hardware hervorbringen.

Eine weitere denkbare Erweiterung könnte darin bestehen, die in dieser Masterarbeit vorgestellten visuellen Helfer mit den visuellen Helfern, die auf Objekterkennung basieren, zu kombinieren. Dadurch könnten die Vorteile von beiden Ansätzen kombiniert werden. Das autonome Verhalten der auf Objekterkennung basierenden visuellen Helfer könnte vom Benutzer überwacht werden und bei fehlerhafter Objekterkennung könnte er die Aufgabe mit den visuellen Helfern, die nicht auf Objekterkennung basieren, eigenständig fortführen. Dadurch könnten Montageprozesse flexibel getestet werden und mit Objekterkennung ggf. zu einem späteren Zeitpunkt erweitert werden.

6.2 Fazit

Der in dieser Masterarbeit entwickelte Augmented Reality Prototyp erlaubt die Steuerung eines Roboterarms zur Durchführung unterschiedlichster Montageaufgaben. Dabei ist die Steuerung, welche nur über Sprachbefehle und Kopfbewegungen funktioniert, so einfach gehalten worden, dass Laien schon nach einer kurzen Einführung anspruchsvolle Pick und Place Aufgaben durchführen konnten. Die entwickelten visuellen Helfer unterstützten den Benutzer und zeigen großes Potenzial, die Präzision zu steigern und die Bedienung zu vereinfachen. Zwar konnten diese Eigenschaften in der durchgeführten Studie noch nicht ausreichend verifiziert werden, jedoch ist eine Tendenz zu erkennen. Die visuellen Helfer haben keine speziellen Voraussetzungen im Bezug auf Objekte oder ihrer Umgebung, daher können sie sich flexibel an unterschiedliche Montageaufgaben und ihre verschiedenen Teile anpassen. Dies erhöht die Praktikabilität der visuellen Helfer und ihr Einsatzpotenzial. Die in dieser Masterarbeit entstandene Schnittstelle zum Kuka iiwa erlaubt nicht nur dem Prototyp die Steuerung des Roboterarms, sondern kann ohne Änderungen auch von anderen Anwendungen als Schnittstelle zum Kuka iiwa benutzt werden. Eventuell folgende Studien und Folgeprojekte können einfach auf dem bestehenden Prototyp und seinem Quellcode aufgebauen. Insgesamt konnten die gesetzten Ziele dieser Masterarbeit erfolgreich umgesetzt werden.

A Literaturverzeichnis

Akgun, B., Cakmak, M., Jiang, K. & Thomaz, A. L. (2012). Keyframe-based learning from demonstration. *International Journal of Social Robotics*, *4*(4), 343–355.

Akgun, B. & Subramanian, K. (2011). Robot learning from demonstration: kinesthetic teaching vs. teleoperation. *Unpublished manuscript*.

Azuma, R. T. (1997). A survey of augmented reality. *Presence: Teleoperators & Virtual Environments*, *6*(4), 355–385.

Baldi, T. L., Spagnoletti, G., Dragusanu, M. & Prattichizzo, D. (2017). Design of a wearable interface for lightweight robotic arm for people with mobility impairments. In *2017 International Conference on Rehabilitation Robotics (ICORR)* (S. 1567–1573). IEEE.

Billinghurst, M., Clark, A., Lee, G. et al. (2015). A survey of augmented reality. *Foundations and Trends® in Human–Computer Interaction*, *8*(2-3), 73–272.

Bland, J. M. & Altman, D. G. (1995). Multiple significance tests: the Bonferroni method. *Bmj*, *310*(6973), 170.

Bogue, R. (2012). Robots for space exploration. *Industrial Robot: An International Journal*, *39*(4), 323–328.

Bolano, G., Tanev, A., Steffen, L., Roennau, A. & Dillmann, R. (2018). Towards a Vision-Based Concept for Gesture Control of a Robot Providing Visual Feedback. In *2018 IEEE International Conference on Robotics and Biomimetics (ROBIO)* (S. 386–392). IEEE.

Bouchard, S. (2014). Robot End Effector: Definition and Examples. Verfügbar 18. Februar 2020 unter https://blog.robotiq.com/bid/53266/Robot-End-Effector-Definition-and-Examples

Brachmann, E., Michel, F., Krull, A., Ying Yang, M., Gumhold, S. et al. (2016). Uncertainty-driven 6d pose estimation of objects and scenes from a single rgb image. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (S. 3364–3372).

Bray, B., McCulloch, J., Schonning, N. & Matt, Z. (2018). What is mixed reality? Verfügbar 14. Oktober 2019 unter https://docs.microsoft.com/en-us/windows/mixed-reality/mixed-reality

Chakraborti, T., Sreedharan, S., Kulkarni, A. & Kambhampati, S. (2017). Alternative modes of interaction in proximal human-in-the-loop operation of robots. *arXiv preprint arXiv:1703.08930*.

Chandrasekaran, B. & Conrad, J. M. (2015). Human-robot collaboration: A survey. In *SoutheastCon 2015* (S. 1–8). IEEE.

Chen, J. Y., Haas, E. C. & Barnes, M. J. (2007). Human performance issues and user interface design for teleoperated robots. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, *37*(6), 1231–1245.

Chen, T. L., Ciocarlie, M., Cousins, S., Grice, P. M., Hawkins, K., Hsiao, K., ... Leeper, A. E. et al. (2013). Robots for humanity: using assistive robotics to empower people with disabilities. *IEEE Robotics & Automation Magazine*, *20*(1), 30–39.

Christensen, H. I., Batzinger, T., Bekris, K., Bohringer, K., Bordogna, J., Bradski, G., ... Eastman, R. et al. (2009). A roadmap for us robotics: from internet to robotics. *Computing Community Consortium*, *44*.

Cooley, S. & Moore, G. (2019). HoloLens-Hardware (1. Generation). Verfügbar 5. Januar 2020 unter https://docs.microsoft.com/de-de/hololens/hololens1-hardware

Corno, F., Farinetti, L. & Signorile, I. (2002). A cost-effective solution for eye-gaze assistive technology. In *Proceedings. IEEE International Conference on Multimedia and Expo* (Bd. 2, S. 433–436). IEEE.

Dymond, E. & Potter, R. (1996). Controlling assistive technology with head movements-a review. *Clinical rehabilitation*, *10*(2), 93–103.

ELZo3, Z. (2019). Robotiq Gripper 2F85 EtherCat. Verfügbar 26. Februar 2020 unter https://github.com/ELZo3/RobotiqGripper-2F85-EtherCat-Control

Epson. (2019). Epson Moverio BT-200. Verfügbar 14. November 2019 unter https://www.epson.de/products/see-through-mobile-viewer/moverio-bt-200#manuals

Fang, H., Ong, S. & Nee, A. (2012). Interactive robot trajectory planning and simulation using augmented reality. *Robotics and Computer-Integrated Manufacturing*, *28*(2), 227–237.

Fang, H., Ong, S. & Nee, A. (2014). Novel AR-based interface for human-robot interaction and visualization. *Advances in Manufacturing*, *2*(4), 275–288.

Flacco, F. & De Luca, A. (2013). Safe physical human-robot collaboration. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems* (S. 2072–2072). IEEE.

Forbes, M., Rao, R. P., Zettlemoyer, L. & Cakmak, M. (2015). Robot programming by demonstration with situated spatial language understanding. In *2015 IEEE International Conference on Robotics and Automation (ICRA)* (S. 2014–2020). IEEE.

Frank, J. A., Moorhead, M. & Kapila, V. (2017). Mobile mixed-reality interfaces that enhance human–robot interaction in shared spaces. *Frontiers in Robotics and AI*, *4*, 20.

Ganesan, R. K., Rathore, Y. K., Ross, H. M. & Amor, H. B. (2018). Better teaming through visual cues: how projecting imagery in a workspace can improve human-robot collaboration. *IEEE Robotics & Automation Magazine*, *25*(2), 59–71.

Gao, Y. & Huang, C.-M. (2019). PATI: a projection-based augmented table-top interface for robot programming. In *Proceedings of the 24th International Conference on Intelligent User Interfaces* (S. 345–355). ACM.

Gerken, J. (2020). MIA - Human-Robot Interaction at the Workplace. Verfügbar 22. April 2020 unter https://hci.w-hs.de/research/projects/mia/

Girden, E. R. (1992). *ANOVA: Repeated measures*. Sage.

Goodrich, M. A., Schultz, A. C. et al. (2008). Human–robot interaction: a survey. *Foundations and Trends® in Human–Computer Interaction*, *1*(3), 203–275.

Hansen, J. P., Rajanna, V., MacKenzie, I. S. & Bækgaard, P. (2018). A Fitts' law study of click and dwell interaction by gaze, head and mouse with a head-mounted display. In *Proceedings of the Workshop on Communication by Gaze Interaction* (S. 1–5).

Hart, S. G. (2006). NASA-task load index (NASA-TLX); 20 years later. In *Proceedings of the human factors and ergonomics society annual meeting* (Bd. 50, S. 904–908). Sage Publications Sage CA: Los Angeles, CA.

Hasanuzzaman, M., Ampornaramveth, V., Zhang, T., Bhuiyan, M., Shirai, Y. & Ueno, H. (2004). Real-time vision-based gesture recognition for human robot interaction. In *2004 IEEE International Conference on Robotics and Biomimetics* (S. 413–418). IEEE.

Heaney, D. (2019). HoloLens 2's Field of View Revealed. Verfügbar 5. Januar 2020 unter https://uploadvr.com/hololens-2-field-of-view/

Hinterstoisser, S., Cagniart, C., Ilic, S., Sturm, P., Navab, N., Fua, P. & Lepetit, V. (2011). Gradient response maps for real-time detection of textureless objects. *IEEE transactions on pattern analysis and machine intelligence*, *34*(5), 876–888.

Hinterstoisser, S., Lepetit, V., Ilic, S., Holzer, S., Bradski, G., Konolige, K. & Navab, N. (2012). Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In *Asian conference on computer vision* (S. 548–562). Springer.

Hinterstoisser, S., Lepetit, V., Rajkumar, N. & Konolige, K. (2016). Going further with point pair features. In *European conference on computer vision* (S. 834–848). Springer.

Hockstein, N. G., Gourin, C., Faust, R. & Terris, D. J. (2007). A history of robots: from science fiction to surgical robotics. *Journal of robotic surgery*, *1*(2), 113–118.

Hodan, T., Haluza, P., Obdržálek, Š., Matas, J., Lourakis, M. & Zabulis, X. (2017). T-LESS: An RGB-D dataset for 6D pose estimation of texture-less objects. In *2017*

IEEE Winter Conference on Applications of Computer Vision (WACV) (S. 880–888). IEEE.

Hoffman, G. & Breazeal, C. (2004). Collaboration in Human-Robot Teams. In *AIAA 1st Intelligent Systems Technical Conference* (S. 6434).

Hord, S. M. (1981). Working Together: Cooperation or Collaboration?

Jayabhavani, G., Raajan, N. & Rubini, R. (2013). Brain mobile interfacing (BMI) system embedded with wheelchair. In *2013 IEEE Conference on Information & Communication Technologies* (S. 1129–1133). IEEE.

Jia, Y., Xi, N., Chai, J. Y., Cheng, Y., Fang, R. & She, L. (2014). Perceptive feedback for natural language control of robotic operations. In *2014 IEEE International Conference on Robotics and Automation (ICRA)* (S. 6673–6678). IEEE.

Kaminka, G. A. (2013). Curing robot autism: A challenge. In *Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems* (S. 801–804). International Foundation for Autonomous Agents und Multiagent Systems.

Kehl, W., Manhardt, F., Tombari, F., Ilic, S. & Navab, N. (2017). Ssd-6d: Making rgb-based 3d detection and 6d pose estimation great again. In *Proceedings of the IEEE International Conference on Computer Vision* (S. 1521–1529).

Kim, D.-J., Hazlett-Knudsen, R., Culver-Godfrey, H., Rucks, G., Cunningham, T., Portee, D., … Behal, A. (2011). How autonomy impacts performance and satisfaction: Results from a study with spinal cord injured subjects using an assistive robot. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, *42*(1), 2–14.

Kim, K., Lepetit, V. & Woo, W. (2012). Real-time interactive modeling and scalable multiple object tracking for AR. *Computers & Graphics*, *36*(8), 945–954.

Kormushev, P., Nenchev, D. N., Calinon, S. & Caldwell, D. G. (2011). Upper-body kinesthetic teaching of a free-standing humanoid robot. In *2011 IEEE International Conference on Robotics and Automation* (S. 3970–3975). IEEE.

Kowalewski, K. (2016). Aufgesetzte HoloLens. Verfügbar 13. April 2020 unter https://de.wikipedia.org/wiki/Microsoft_HoloLens#/media/Datei:MicrosoftHoloLensOnHead.JPG

Krupke, D., Steinicke, F., Lubos, P., Jonetzko, Y., Görner, M. & Zhang, J. (2018). Comparison of Multimodal Heading and Pointing Gestures for Co-Located Mixed Reality Human-Robot Interaction. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (S. 1–9). IEEE.

Kruskal, W. H. & Wallis, W. A. (1952). Use of ranks in one-criterion variance analysis. *Journal of the American statistical Association*, *47*(260), 583–621.

KUKA AG. (2019a). KR 1000 titan. Verfügbar 6. September 2019 unter https://www.kuka.com/de-de/produkte-leistungen/robotersysteme/industrieroboter/kr-1000-titan

KUKA AG. (2019b). LBR iiwa. Verfügbar 6. September 2019 unter https://www.kuka.com/de-de/presse/mediathek

KUKA AG. (2020). LBR iiwa. Verfügbar 12. Februar 2020 unter https://www.kuka.com/de-de/produkte-leistungen/robotersysteme/industrieroboter/lbr-iiwa

Kukliński, K., Fischer, K., Marhenke, I., Kirstein, F., Maria, V., Sølvason, D., … Savarimuthu, T. R. (2014). Teleoperation for learning by demonstration: Data glove versus object manipulation for intuitive robot control. In *2014 6th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)* (S. 346–351). IEEE.

Lepetit, V., Pilet, J. & Fua, P. (2004). Point matching as a classification problem for fast and robust object pose estimation. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.* (Bd. 2, S. II–II). IEEE.

Liu, H. & Wang, L. (2018). Gesture recognition for human-robot collaboration: A review. *International Journal of Industrial Ergonomics*, *68*, 355–367.

Lowe, D. G. (1999). Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision* (Bd. 2, S. 1150–1157). Ieee.

Lund, A. M. (2001). Measuring usability with the use questionnaire12. *Usability interface*, *8*(2), 3–6.

Magic Leap. (2020). Magic Leap 1. Verfügbar 5. Januar 2020 unter https://www.magicleap.com/magic-leap-1

Makris, S., Karagiannis, P., Koukas, S. & Matthaiakis, A.-S. (2016). Augmented reality system for operator support in human–robot collaborative assembly. *CIRP Annals*, *65*(1), 61–64.

Mazo, M. (2001). An integral system for assisted mobility [automated wheelchair]. *IEEE Robotics & Automation Magazine*, *8*(1), 46–56.

Mensch eV, A. (2015). Inklusionsbarometer Arbeit, ein Instrument zur Messung von Fortschritten bei der Inklusion von Menschen mit Behinderung auf dem deutschen Arbeitsmarkt. 3. Jahrgang (2015).

Microsoft. (2018). Farbe, Licht und Material. Verfügbar 28. März 2020 unter https://docs.microsoft.com/de-de/windows/mixed-reality/color-light-and-materials

Microsoft. (2019a). HoloLens. Verfügbar 18. November 2019 unter https://www.microsoft.com/de-de/hololens

Microsoft. (2019b). Rendering. Verfügbar 28. April 2020 unter https://docs.microsoft.com/de-de/windows/mixed-reality/rendering

Microsoft. (2020a). Erlernen von Tools und Architektur. Verfügbar 26. Februar 2020 unter https://docs.microsoft.com/de-de/windows/mixed-reality/development#unity

Microsoft. (2020b). Hologram stability. Verfügbar 16. März 2020 unter https://docs.microsoft.com/de-de/windows/mixed-reality/hologram-stability

Microsoft. (2020c). HoloLens 2 - Get to know the new features and technical specs. Verfügbar 5. Januar 2020 unter https://www.microsoft.com/en-us/hololens/hardware

Microsoft. (2020d). MixedRealityToolkit-Unity. Verfügbar 25. Februar 2020 unter https://github.com/Microsoft/MixedRealityToolkit-Unity/releases/tag/2017.2.1.4

Mies, G. (2010). Military robots of the present and the future. *Technology*, *9*(1), 125–137.

Milgram, P., Takemura, H., Utsumi, A. & Kishino, F. (1995). Augmented reality: A class of displays on the reality-virtuality continuum. In *Telemanipulator and telepresence technologies* (Bd. 2351, S. 282–292). International Society for Optics und Photonics.

Milgram, P., Zhai, S., Drascic, D. & Grodski, J. (1993). Applications of augmented reality for human-robot communication. In *Proceedings of 1993 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'93)* (Bd. 3, S. 1467–1472). IEEE.

Nishimori, M., Saitoh, T. & Konishi, R. (2007). Voice controlled intelligent wheelchair. In *SICE Annual Conference 2007* (S. 336–340). IEEE.

Ong, S., Yew, A., Thanigaivel, N. & Nee, A. (2020). Augmented reality-assisted robot programming system for industrial applications. *Robotics and Computer-Integrated Manufacturing*, *61*, 101820.

Onnasch, L., Maier, X. & Jürgensohn, T. (2016). Mensch-Roboter-Interaktion-Eine Taxonomie für alle Anwendungsfälle.

Pan, Z., Polden, J., Larkin, N., Van Duin, S. & Norrish, J. (2010). Recent progress on programming methods for industrial robots. In *ISR 2010 (41st International Symposium on Robotics) and ROBOTIK 2010 (6th German Conference on Robotics)* (S. 1–8). VDE.

Posner, M. I., Nissen, M. J. & Klein, R. M. (1976). Visual dominance: an information-processing account of its origins and significance. *Psychological review*, *83*(2), 157.

PTC. (2020). Vuforia Engine 8.6. Verfügbar 16. März 2020 unter https://developer.vuforia.com/downloads/sdk

Pupil Labs GmbH. (2020). Add awareness to your VR/AR experience. Verfügbar 10. Februar 2020 unter https://pupil-labs.com/products/vr-ar/

Qian, Y. Y. & Teather, R. J. (2017). The eyes don't have it: an empirical comparison of head-based and eye-based selection in virtual reality. In *Proceedings of the 5th Symposium on Spatial User Interaction* (S. 91–98). ACM.

Robotiq. (2020). 2-Finger-Greifer 2F-85 und 2F-140. Verfügbar 17. Februar 2020 unter https://robotiq.com/de/produkte/adaptiver-2-finger-robotergreifer-2f85-140

Rosen, E., Whitney, D., Phillips, E., Chien, G., Tompkin, J., Konidaris, G. & Tellex, S. (2019). Communicating and controlling robot arm motion intent through mixed-reality head-mounted displays. *The International Journal of Robotics Research*, 0278364919842925.

Ruffo, K. & Milgram, P. (1992). Effect of stereographic+ stereovideo'tether'enhancement for a peg-in-hole task. In *[Proceedings] 1992 IEEE International Conference on Systems, Man, and Cybernetics* (S. 1425–1430). IEEE.

Sabharwal, S. (2020). Spinal Cord Injury (Cervical). In *Essentials of Physical Medicine and Rehabilitation (Fourth Edition)* (Fourth Edition, S. 902–915). Content Repository Only!

Saha, S. K. (2008). *Introduction to Robotics*. Tata McGraw-Hill Education.

Sheridan, T. B. (1997). Eight ultimate challenges of human-robot communication. In *Proceedings 6th IEEE International Workshop on Robot and Human Communication. RO-MAN'97 SENDAI* (S. 9–14). IEEE.

Steinfeld, A., Fong, T., Kaber, D., Lewis, M., Scholtz, J., Schultz, A. & Goodrich, M. (2006). Common metrics for human-robot interaction. In *Proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction* (S. 33–40).

Sundermeyer, M., Marton, Z.-C., Durner, M., Brucker, M. & Triebel, R. (2018). Implicit 3d orientation learning for 6d object detection from rgb images. In *Proceedings of the European Conference on Computer Vision (ECCV)* (S. 699–715).

Tekin, B., Sinha, S. N. & Fua, P. (2018). Real-time seamless single shot 6d object pose prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (S. 292–301).

Teresa, Z. (2014). History of service robots. In *Robotics: Concepts, Methodologies, Tools, and Applications* (S. 1–14). IGI Global.

Vassallo, R., Rankin, A., Chen, E. C. & Peters, T. M. (2017). Hologram stability evaluation for Microsoft HoloLens. In *Medical Imaging 2017: Image Perception, Observer Performance, and Technology Assessment* (Bd. 10136, S. 1013614). International Society for Optics und Photonics.

Williams, T., Szafir, D., Chakraborti, T. & Ben Amor, H. (2018). Virtual, augmented, and mixed reality for human-robot interaction. In *Companion of the 2018 ACM/IEEE International Conference on Human-Robot Interaction* (S. 403–404). ACM.

Wolf, M. T., Assad, C., Vernacchia, M. T., Fromm, J. & Jethani, H. L. (2013). Gesture-based robot control with variable autonomy from the JPL BioSleeve. In *2013 IEEE International Conference on Robotics and Automation* (S. 1160–1165). IEEE.

Wolpaw, J. & Wolpaw, E. W. (2012). *Brain-computer interfaces: principles and practice*. OUP USA.

Wong, B. (2011). Points of view: Color blindness. Nature Publishing Group.

Woods, D. D., Tittle, J., Feil, M. & Roesler, A. (2004). Envisioning human-robot coordination in future operations. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, *34*(2), 210–218.

Zeng, H., Wang, Y., Wu, C., Song, A., Liu, J., Ji, P., ... Wen, P. (2017). Closed-loop hybrid gaze brain-machine interface based robotic arm control with augmented reality feedback. *Frontiers in neurorobotics*, *11*, 60.

B Anhang

B.1 Fragebogen zur Demografie und Hintergrundwissen

Alter:

Geschlecht: ☐ männlich ☐ weiblich

Beruf:

Benutzen Sie eine Sehhilfe?

☐ Keine ☐ Brille ☐ Kontaktlinsen

Sind Sie Farbenblindheit?

☐ Ja ☐ Nein

Wenn sie mit Ja geantwortet haben, bitte geben sie die Art von Farbenblindheit an:

Haben Sie jemals die Microsoft HoloLens benutzt?

☐ Ja ☐ Nein

Haben Sie jemals mit einem Roboter gearbeitet oder programmiert?

☐ Ja ☐ Nein

Wenn Sie einen Roboter programmiert haben, bitte geben Sie ihre Kompetenz an:

☐ Anfänger ☐ Fortgeschritten ☐ Experte

Haben Sie jemals Augmented oder Virtual Reality erlebt?

☐ Ja ☐ Nein

Haben Sie Erfahrungen mit einer Augmented oder Virtual Reality Programmiersoftware gearbeitet?

☐ Ja ☐ Nein

Wenn sie mit Ja geantwortet haben, bitte geben sie ihre Kompetenz an:

☐ Anfänger ☐ Fortgeschritten ☐ Experte

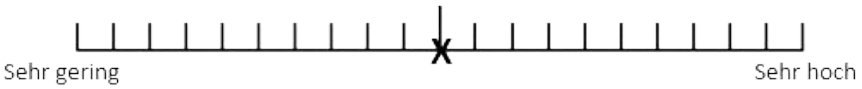
Haben Sie Erfahrungen mit 3D Modelling Software (z. B. Blender, AutoCAD, Cinema4D, ...)?

☐ Ja ☐ Nein

B.2 Nasa Task-Load-Index Fragebogen

Markieren Sie dazu auf den folgenden Skalen, in welchem Maße Sie sich in den sechs genannten Dimensionen von der Aufgabe beansprucht oder gefordert gesehen haben:

Beispiel:



Geistige Anstrengung — Wie viel geistige Anstrengung war bei der Bearbeitung der Aufgaben erforderlich?

Körperliche Anstrengung — Wie viel körperliche Anstrengung war erforderlich (z. B. zum Auswählen, Drehen, Bewegen, Greifen)?

Zeitdruck — Wie viel Zeitdruck empfanden Sie bei der Bearbeitung der Aufgaben?

Leistung — Wie erfolgreich haben Sie Ihrer Meinung nach die Aufgaben bearbeitet?

Anstrengung — Wie sehr mussten Sie sich anstrengen, um die Aufgaben erfolgreich zu bearbeiten?

Frustration — Wie unsicher, entmutigt, irritiert, gestresst und verärgert waren Sie bei der Bearbeitung der Aufgabe?

B.3 USE Fragebogen: Nutzen, Zufriedenheit, und Bedienkomfort

Bitte bewerte deine Zustimmung zu den folgenden Aussagen.

Aussagen	Stimme gar nicht zu						Stimme voll zu
Nutzen							
1. Es hilft mit effektiv zu sein	1	2	3	4	5	6	7
2. Es hilft mit produktiv zu sein	1	2	3	4	5	6	7
3. Es erleichtert das Durchführen von Aufgaben	1	2	3	4	5	6	7
4. Es spart Zeit	1	2	3	4	5	6	7
5. Es entspricht meinen Anforderungen	1	2	3	4	5	6	7
6. Es macht alles, was ich erwarte	1	2	3	4	5	6	7
Bedienkomfort							
7. Es ist einfach zu benutzen	1	2	3	4	5	6	7
8. Es ist einfach gehalten	1	2	3	4	5	6	7
9. Es ist benutzerfreundlich	1	2	3	4	5	6	7
10. Es benötigt die geringste Zahl an Arbeitsschritten um das Ziel zu erreichen	1	2	3	4	5	6	7
11. Es ist flexibel	1	2	3	4	5	6	7
12. Die Benutzung gelingt mühelos	1	2	3	4	5	6	7
13. Ich kann es ohne schriftliche Anleitung benutzen	1	2	3	4	5	6	7
14. Die Benutzung ist jedes Mal erfolgreich	1	2	3	4	5	6	7
Lernprozess							
15. Ich habe schnell gelernt es zu benutzen	1	2	3	4	5	6	7
16. Ich kann mir leicht merken, wie man es benutzt	1	2	3	4	5	6	7
17. Die Benutzung ist einfach erlernbar	1	2	3	4	5	6	7
18. Meine Fähigkeit es zu benutzen, hat sich schnell gesteigert.	1	2	3	4	5	6	7
Zufriedenheit							
19. Es funktioniert so wie ich möchte	1	2	3	4	5	6	7
20. Die Benutzung fühlt sich gut an	1	2	3	4	5	6	7
21. Ich bin zufrieden mit dem Produkt	1	2	3	4	5	6	7
22. Die Benutzung macht Spaß	1	2	3	4	5	6	7

B.4 Interviewfragen

1. Erzählen Sie uns, was für eine Strategie Sie bei der Durchführung der Aufgaben angewandt haben.

2. Was hat Ihnen am meisten bei der Vervollständigung der Aufgabe geholfen?

3. Was hat Ihnen beim Greifen eines Objektes am meisten geholfen und was beim Platzieren?

4. Verspürten Sie den Drang sich nach links oder rechts zu bewegen, um ihre Perspektive zu ändern?

5. Waren sie zu irgendeinem Zeitpunkt verwirrt? Wie? Wann? Erklären Sie ihre Antwort.

6. Können Sie sich noch weitere Elemente vorstellen, die ihnen bei der Durchführung der Aufgabe geholfen hätte?

B.5 Deskriptive Statistiken zu den Diagrammen

Aufgabenzeit über alle Versionen der visuellen Helfer in Sekunden				Aufgabenzeit über alle Versuche in Sekunden			
	1. Versuch	2. Versuch	3. Versuch		No Cues	Cues	Advanced Cues
N	23	23	23	N	7	9	7
Mittelwert	444,43	372,17	308,52	Mittelwert	414,05	424,00	273,10
Median	367	348	300	Median	340,67	423,33	260,00
Std.-Abweichung	208,728	147,838	103,127	Std.-Abweichung	145,04	138,64	66,24
Spannweite	790	548	398	Spannweite	405,33	476,67	189,67
Minimum	209	190	150	Minimum	315,00	241,33	197,00
Maximum	999	738	548	Maximum	720,33	718,00	386,67

Präzision bei den Objekte über alle Versuche in Millimeter									
	No Cues			Cues			Advanced Cues		
	1. Objekt	2. Objekt	3. Objekt	1. Objekt	2. Objekt	3. Objekt	1. Objekt	2. Objekt	3. Objekt
N	7	7	7	9	9	9	7	7	7
Mittelwert	12,79	16,11	8,84	9,16	11,04	11,04	11,29	11,52	8,09
Median	11,99	17,31	6,95	8,41	10,23	9,76	11,99	11,20	7,59
Std.-Abweichung	2,68	4,79	3,63	2,17	2,94	5,47	0,98	3,52	2,20
Spannweite	7,89	11,96	10,27	6,41	8,30	14,66	2,18	10,76	5,97
Minimum	10,63	9,44	5,40	7,14	7,63	5,27	10,03	6,93	5,14
Maximum	18,52	21,40	15,68	13,55	15,93	19,93	12,21	17,68	11,10

Anzahl der Befehle über alle Versuche									
	No Cues			Cues			Advanced Cues		
	1. Objekt	2. Objekt	3. Objekt	1. Objekt	2. Objekt	3. Objekt	1. Objekt	2. Objekt	3. Objekt
N	7	7	7	9	9	9	7	7	7
Mittelwert	9,14	7,31	9,81	8,09	6,24	8,65	5,48	4,33	4,12
Median	8,83	7,50	8,17	8,83	6,33	9,33	5,00	3,67	4,00
Std.-Abweichung	1,25	0,94	3,60	2,02	1,47	1,71	1,49	1,52	0,28
Spannweite	3,83	3,00	10,00	5,83	4,50	5,17	3,50	4,50	0,83
Minimum	7,50	5,50	6,83	4,83	3,67	5,83	4,17	3,00	3,83
Maximum	11,33	8,50	16,83	10,67	8,17	11,00	7,67	7,50	4,67

Feinjustierzeit über alle Versuche in Sekunden

Cues		1. Objekt		1. Objekt		1. Objekt	
		Pick	Place	Pick	Place	Pick	Place
No Cues	N	7	7	7	7	7	7
	Mittelwert	48,83	60,09	37,41	51,41	65,62	72,26
	Median	50,48	50,80	40,06	55,89	41,13	69,00
	Std.-Abweichung	15,80	33,96	8,45	14,27	46,71	19,09
	Spannweite	52,41	95,72	25,45	42,13	133,60	49,91
	Minimum	22,67	32,19	24,89	26,68	21,21	50,76
	Maximum	75,08	127,91	50,34	68,80	154,81	100,67
Cues	N	9	9	9	9	9	9
	Mittelwert	50,13	58,52	47,52	37,99	73,11	73,18
	Median	44,73	47,69	43,41	36,38	71,49	61,02
	Std.-Abweichung	28,78	28,97	23,64	19,03	35,75	37,05
	Spannweite	84,31	75,97	72,75	51,29	111,36	125,21
	Minimum	13,81	19,83	14,27	18,90	14,79	37,22
	Maximum	98,12	95,80	87,02	70,19	126,15	162,44
Advanced Cues	N	7	7	7	7	7	7
	Mittelwert	19,86	38,24	19,69	36,16	25,99	36,30
	Median	18,14	25,12	17,34	26,86	29,91	33,72
	Std.-Abweichung	11,29	24,17	7,64	21,33	12,55	13,61
	Spannweite	33,10	59,93	21,36	56,96	34,76	38,65
	Minimum	4,68	17,15	12,04	14,18	8,72	12,82
	Maximum	37,78	77,08	33,40	71,15	43,48	51,46

USE Fragebogen

Visuelle Helfer		Nützlichkeit	Bedienfreund-lichkeit	Lernfreund-lichkeit	Zufriedenheit
No Cues	N	7	7	7	7
	Mittelwert	3,69	4,59	5,04	4,79
	Median	4,17	5,00	5,75	5,25
	Std.-Abweichung	1,23	1,88	2,39	1,72
	Spannweite	3,50	5,00	5,75	4,25
	Minimum	2,00	1,75	1,25	2,25
	Maximum	5,50	6,75	7,00	6,50
Cues	N	9	9	9	9
	Mittelwert	5,11	5,08	6,06	5,64
	Median	5,17	5,63	6,50	5,50
	Std.-Abweichung	1,18	1,11	0,98	0,59
	Spannweite	4,00	3,38	2,50	2,00
	Minimum	3,00	2,88	4,50	4,50
	Maximum	7,00	6,25	7,00	6,50
Advanced Cues	N	7	7	7	7
	Mittelwert	5,55	5,52	6,32	6,00
	Median	5,33	5,50	6,25	6,00
	Std.-Abweichung	0,70	0,28	0,55	0,43
	Spannweite	2,00	0,88	1,50	1,00
	Minimum	5,00	5,13	5,50	5,50
	Maximum	7,00	6,00	7,00	6,50

Raw Nasa-TLX

Visuelle Helfer		Geistige An-strengung	Körperliche Anstrengung	Zeitdruck	Leistung	Anstrengung	Frustration	Insgesamte Ar-beitsbelastung
No Cue	N	7	7	7	7	7	7	7
	Mittelwert	69,29	22,86	25,00	62,14	58,57	41,43	46,55
	Median	75,00	20,00	20,00	65,00	60,00	40,00	43,33
	Std.-Abweichung	13,67	12,54	14,43	20,18	13,76	24,95	10,26
	Spannweite	30,00	35,00	40,00	60,00	35,00	80,00	31,67
	Minimum	50,00	5,00	15,00	25,00	40,00	10,00	32,50
	Maximum	80,00	40,00	55,00	85,00	75,00	90,00	64,17
Cue	N	9	9	9	9	9	9	9
	Mittelwert	73,33	28,89	33,33	33,33	67,67	37,78	45,72
	Median	80,00	15,00	25,00	35,00	69,00	40,00	39,17
	Std.-Abweichung	14,79	30,29	19,36	15,21	8,67	26,11	12,22
	Spannweite	50,00	90,00	55,00	40,00	25,00	75,00	30,83
	Minimum	40,00	0,00	15,00	15,00	55,00	0,00	32,50
	Maximum	90,00	90,00	70,00	55,00	80,00	75,00	63,33
Advanced Cue	N	7	7	7	7	7	7	7
	Mittelwert	54,29	15,71	21,43	27,86	47,86	34,29	33,57
	Median	60,00	15,00	10,00	25,00	50,00	25,00	35,83
	Std.-Abweichung	23,88	9,32	22,68	11,13	17,99	21,88	10,54
	Spannweite	60,00	20,00	60,00	30,00	45,00	65,00	27,50
	Minimum	20,00	5,00	0,00	20,00	25,00	10,00	17,50
	Maximum	80,00	25,00	60,00	50,00	70,00	75,00	45,00

B.6 Inhalt des beiliegenden USB-Sticks

Ordner	**Beschreibung**
Bilder	Für diese Masterarbeit angefertigten Bilder in voller Auflösung
Masterarbeit	Die Masterarbeit als .pdf
ProgrammierCode	Der Programmiercode der Applikation für die HoloLens, sowie die Schnittstelle zum Kuka iiwa
Studie	Alle Rohdaten und SPSS-Dateien von der Studie in Kapitel 5
Video	Video vom entstandenen Prototyp