

Augmented Reality als intuitive Benutzungsschnittstelle für das Roboterprogrammieren

Carolin Horn, Christoph-Philipp Schreiber

Das Programmieren der Bewegungsbahnen von Robotern erfordert Fachwissen und ist ein zeitintensiver und aufwendiger Prozess. Dieser Beitrag beschäftigt sich mit dem Einsatz von Augmented Reality (AR) in Form eines AR Head Mounted Display (HMD) als intuitives Schnittstelle (engl. Interface) für die Roboterprogrammierung. Zunächst wird ein Überblick über aktuelle und relevante Forschung im Bereich AR Anwendungen in der Robotik gegeben. Aktuelle Forschungsarbeit auf dem Gebiet widmet sich vorrangig der technischen Umsetzung einzelner Funktionalitäten. In diesem Beitrag aus der Praxis sollen die technischen Möglichkeiten den Problematiken potenzieller Anwender:innen angepasst werden. Der Fokus liegt damit auf dem Mehrwert für spezifische Nutzergruppen und der einfachen und intuitiven Bedienung des AR Interfaces selbst. Zunächst wird, einem nutzerzentrierten Entwicklungsprozess folgend, erhoben, welchen Herausforderungen Expert:innen und Laien bei der Roboterprogrammierung begegnen. Auf dieser Basis werden Anforderungen abgeleitet und ein erlebbarer Prototyp entwickelt und gestaltet, der eine weitere Untersuchungen ermöglicht. Ein geplantes Untersuchungskonzept hinsichtlich Aspekten der User Experience (UX) wird im Ausblick beleuchtet.

Keywords: Augmented Reality, Robotik, Interface-Design, User Experience, Usability

Einleitung

Sanneman et al. (2020, S. 65 f.) beschreiben als aktuelle Herausforderungen der Roboterautomatisierung lange und teure Integrationsarbeiten gepaart mit einem generellen Mangel an Expert:innen der Integration und Programmierung. Eine fehlende Standardisierung der Robotik verhindere Modularität und den einfachen Austausch von Programmen und Systemen. Vor allem für kleine und mittlere Unternehmen (KMU), die geringe Stückzahlen produzieren, sei die mangelnde Flexibilität der Systeme problematisch. Grund dafür sind u. a. Benutzungsschnittstellen (engl. Interface), die Informatikwissen voraussetzen. Bei der online Programmierung werden Industrieroboter Zeile für Zeile programmiert, indem Roboterpositionen mit einem Programmierhandgerät (engl. Teach Pendant) nacheinander angefahren werden (Shepherd et al.: 2019, S.13). Diese in der Praxis vorherrschende Methode erfordert Erfahrung und Zeit, da

räumliche Orientierungen umständlich über ein zweidimensionales Interface eingestellt werden müssen (Heimann & Guhl: 2020, S.696). Die offline Programmierung, die nicht am Roboter selbst, sondern in spezieller Simulationssoftware vorgenommen wird, ist oft herstellereigen und mit hohen Investitionskosten verbunden (Heimann & Guhl: 2020, S.697). Angetrieben durch die angeführten Bedarfe, konzentriert sich die Entwicklungsarbeit bei der *Wandelbots GmbH* auf die einfache Programmierung von Robotern. Mit der Vision, die Robotik einer breiteren Nutzergruppe zu erschließen, entwickelt sie intuitive Produkte zur Roboterprogrammierung. Dazu gehört ein stiftähnliches Eingabegerät, der *TracePen*, mit dem sich *Trajektorien* (in diesem Kontext synonym verwendet mit Roboterbahnen oder Pfaden) mithilfe von Positionstracking aufnehmen lassen. In Kombination mit einer iPad App ist das Editieren von Trajektorien und das Aufbauen einfacher Programmabläufe für Laien möglich. Eine große Hürde für die Anwender:innen besteht darin, dass trotz App und TracePen schwer zu erkennen ist, wie sich der Roboter im Raum bewegen wird und wo die programmierten Punkte im Raum liegen, ohne sie mit dem Roboter anzufahren.

Eine Technologie, die die Roboterprogrammierung in Bezug auf die User Experience (UX) verbessern kann, ist Augmented Reality (AR). Mit mobiler AR Technologie, z. B. auf Tablets, lassen sich die Roboterbahnen räumlich anzeigen und bearbeiten, wie das Start-Up *Augmentus* (Augmentus, 2021) anhand einer iPad App demonstriert. Allerdings besteht bei solchen Ansätzen weiterhin das Problem, dass räumliche Informationen über zweidimensionale Interfaces eingegeben werden müssen. Deshalb verspricht der Gebrauch von AR Head Mounted Displays (AR HMDs) ein höheres Maß an Intuitivität, da zusätzlich zur räumlichen Darstellung auch natürliche Interaktion wie z. B. die Gestenmanipulation möglich ist.

Augmented Reality in der Robotik

Forschungsarbeiten der letzten vier Jahre zeigen innovative Ansätze auf, wie AR in der Robotik und v. a. in der Roboterprogrammierung integriert werden kann (z. B. Krupke et al.:2018, Guhl et al.: 2018 oder Quintero et al.: 2018).

Die Arbeiten beziehen sich meist auf die produktive Interaktion zwischen Robotern und Nutzer:innen mit geteiltem Arbeitsraum oder auf die Interaktion zwischen Programmierer:innen und Robotern zur Pfadprogrammierung und Prüfung. Für beide Szenarien werden über AR meist visuelle Hinweise über bevorstehende Roboterbewegungen umgesetzt, wie z. B. bei der robotergestützten Montage (Krupke et al.: 2018, S.1). Guhl et al. (2018, S.1569) und Shepherd et al. (2019, S.15) beschreiben Programmiermethoden, die die Vorteile einer CAD Offline Simulation von Roboterpfaden in

Virtual Reality (VR) und der Online Anpassung inklusive Kollisionserkennung mit AR verbinden. Quintero et al. (2018, S.1840) verwenden eine *Microsoft HoloLens* AR-Brille, um Pfadpunkte mittels Handinteraktion (Greifen, Ziehen und Drehen) in ihrer Position und Rotation zu verändern. In der Studie wird gezeigt, dass das AR System grundsätzlich und vor allem für komplexe Trajektorien schneller zu bedienen ist als ein zum Vergleich herangezogenes Gamepad für die Robotersteuerung.

Zusammenfassend kann festgestellt werden, dass vor allem einzelne AR Funktionalitäten funktional umgesetzt und teils mit Nutzerstudien hinsichtlich Usability und Leistungsaspekten wie z. B. die Bedienzeit erforscht werden. Keine der gesichteten Studien befasst sich jedoch gleichzeitig mit der Erforschung von Bedarfen und Zielgruppen und möglichen Lösungen auf technologischer Ebene. Auch die Erhebung subjektiv wahrgenommen Qualitäten über die Pragmatik hinaus ist im Kontext AR in der Roboterprogrammierung wenig beforscht.

Dieser Beitrag aus der Praxis beschreibt eine mögliche Anwendung von AR im Kontext der Roboterprogrammierung, die mithilfe eines szenariobasierten und nutzerzentrierten Ansatzes entwickelt wurde. Die zugehörige Forschungsfrage lautet:

Welche gestalterischen Anforderungen muss ein AR HMD Interface für Visualisierung, Interaktion und Navigation innerhalb des Roboterprogrammierens erfüllen, um für spezifische Nutzer:innen eine hohe pragmatische und hedonische Qualität zu erreichen?

Dabei bezieht sich die pragmatische Qualität auf die Usability (deutsch: Gebrauchstauglichkeit) während die hedonische Qualität auf der subjektiv wahrgenommen Qualität des Produkts beruht (Hassenzahl et al.: 2008, S.78).

User Research – Experteninterviews

Praxisnahe Informationen zu den Profilen der Beschäftigten in der Roboterprogrammierung und ihren Herausforderungen lassen sich schwer aus wissenschaftlichen Arbeiten extrahieren. Daher wurden explorative Experteninterviews durchgeführt. Als Fachleute wurden Roboterprogrammierer:innen ausgesucht, weil sie durch die projektbasierte Arbeit einen guten Überblick über die Robotikbranche aufweisen. Da die Interviews am Anfang des Entwicklungsprozesses des AR HMD Interfaces durchgeführt wurden, ist das Ziel mit den Interviews eine grobe Ersteinschätzung von potenziellen Nutzer:innen und den jeweiligen Herausforderungen zu erhalten. Daher wurde auf eine große Anzahl von Interviews verzichtet. Wert wurde dagegen auf die Diversität

bezüglich der Erfahrung der Befragten gelegt (Tab. 1). So haben die Probanden Erfahrung mit unterschiedlichen Robotertypen und Applikationen wie Lackieren oder Schweißen und sind bzw. waren in verschiedenen Branchen (z. B. Systemintegrator, produzierender Endkunde) der Robotik beschäftigt.

Tabelle 1: Soziodemographische Hintergründe der befragten Experten

	Interview 1	Interview 2	Interview 3	Interview 4
Beruf	Roboterprogrammierer / Robotikingenieur	Automatisierungsingenieur im Service	Entwickler SPS und Robotik	Robotikingenieur
Roboter	KUKA iiwa, ABB IRC5	UR, ABB, KUKA, Reis	KUKA, KUKA iiwa	ABB, Fanuc, Yaskawa
Branche	Systemintegrator hauptsächlich für Automobilbranche und Zulieferer	Produzierender Endkunde mit den Applikationen	Systemintegrator für verschiedene Branchen	verschiedenen Systemintegratoren mit unterschiedlichen Anwendungen
Applikation	Schweißen, Kleben, Stopfen setzen, Handling	Lackieren, Kleben, Handling	Fräsrobotik, Holzbearbeitung, Handling, Kleben, Nageln	Lackieren, Handling, Schweißen, Stopfen setzen
Erfahrung	1 Jahr	15 Jahre	4,5 Jahre	11 Jahre

Bei den Herausforderungen wurde in der Befragung zwischen den zeitaufwändigen/schwerwiegenden Problematiken für Expert:innen und den spezifischen Herausforderungen der Roboterprogrammierung für Laien unterschieden. Bei Letzteren wurden die Probanden gefragt, an welche Problematiken sie sich erinnern können, die zwar am Anfang schwierig waren, mit der Zeit und der Berufserfahrung aber in Gewohnheit übergingen. Um die Laienperspektive zu unterstützen, wurde auch ein Experte mit geringer Berufserfahrung (1 Jahr) befragt. In der Zusammenfassung der Ergebnisse werden jeweils die fünf meistgenannten Problematiken aufgeführt.

Zusammenfassung der Ergebnisse

Auch wenn die Übergänge fließend sind, können zwei größere Nutzungsgruppen unterschieden werden: Die Fachleute der Roboterprogrammierung und die Laien der Roboterprogrammierung, die jedoch Expertise in bestimmten Applikationen aufweisen.

Expert:innen der Roboterprogrammierung. Alle Befragten berichten von einer großen Diversität der Beschäftigten im Feld der Roboterprogrammierung. Der Großteil der Expert:innen der Roboterprogrammierer habe keinen akademischen Abschluss, sondern

eine Berufsausbildung z. B. im Bereich Elektronik, Mechatronik, Anlagentechnik oder Instandhaltung, wobei auch Quereinsteigende nicht selten seien. Meist seien sie vor allem in der Benutzungsschnittstelle eines spezifischen Herstellers sehr gut ausgebildet. Darüber hinaus seien oft nur grundlegende Kenntnisse in den Programmiermethoden und der Steuerung weiterer Robotertypen vorhanden. Je komplexer und optimierter die Anlagen, desto eher würden auf bestimmte Anwendungen spezialisierte Roboterprogrammierer eingesetzt und desto höher seien die Integrationszeiten, z. B. im Karosserierohbau in der Automobilbranche.

Applikationsexpert:innen. Da immer mehr Roboter in KMUs eingesetzt würden, stiege der Anteil von Laien der Roboterprogrammierung im Bereich der Robotik. In diesem Kontext als Applikationsexperten bezeichnet, ist diese Gruppe über ihre Expertise in einer bestimmten Applikation charakterisiert. Mögliche Applikationen könnten industrieller Natur – wie Kenntnisse über spezielle Automatisierungsprozesse – oder auch handwerklicher Natur, z. B. das Lackieren, sein. Applikationsexperten in KMU oder im Handwerk weisen oft noch wenig Erfahrung in der Robotik auf. Typische Ausbildungen seien neben handwerklichen Ausbildungsberufen auch Automatisierungstechnik, Mechatroniker oder IT-Systemtechniker. Das Bedienen der Roboter von Nutzer:innen aus dem Bereich Maschinen- oder Anlagenführung komme dagegen sehr selten vor und gehe meist nicht über ein An- und Ausschalten von Roboterprogrammen hinaus. Voraussetzung für eine weitere Verbreitung der Robotik sei dementsprechend die einfache Benutzung der Schnittstellen, sodass Anpassungen durch Laien der Robotik durchgeführt werden könnten.

Herausforderungen für Experten (Anzahl der Nennungen):

1. Jeder Roboterhersteller wie z. B. KUKA oder Fanuc hat ein eigenes Bedienkonzept. Zeitintensiv ist das Suchen nach Funktionen und das Erarbeiten von Behelfslösungen in unbekanntem Systemen. (III)
2. Crashes beim Testen von Robotertrajektorien sind problematisch, da finanzielle und personelle Schäden groß sein können. Ursachen dafür sind z.B. Zeitmangel, enge Arbeitsräume und die hohe notwendige Konzentration. (III)
3. Die Optimierung der Robotertrajektorien auf wenige Punkte und Einhaltung der Taktzeit ist zeitaufwändig, da alle Punkte nacheinander angefahren und korrigiert werden müssen. (II)
4. Strapaziöse und enge Umgebung macht die Arbeit anstrengend und gefährlich, z.B. bei der Hohlraumkonservierung. Die Ursache sind z.B. klebrige Mittel. (II)

5. Wenn Roboterprogrammierer:innen auf eine Baustelle kommen, ist es für sie oft schwer, sich einen Überblick über die komplexen Kommunikationswege zwischen Roboter und Steuerung zu verschaffen Ursache sind z. B. fehlende Standards. (II)

Herausforderungen für Laien (Anzahl der Nennungen):

1. Oft fehlt das Verständnis für Koordinatensysteme. Schwierig ist es zu antizipieren, in welche Richtung der Roboter bei den gespeicherten Koordinatensystemen fährt. (III)
2. Es ist am Anfang kompliziert, die verschiedenen Bewegungsarten des Roboters nachzuvollziehen. Laien kennen die Vor- und Nachteile noch nicht. (II)
3. Das Gefühl für das Steuern des Roboters mit dem Teach Pendant und für die Grenzen der Achsen und Singularitäten kommt erst mit der Erfahrung. (II)
4. Am Anfang haben einige Bediener:innen Angst, Schäden zu verursachen und großen Respekt vor dem Roboter. (II)
5. Es ist für Laien schwierig, eine Roboterprogrammiersprache zu erlernen, wenn sie keine Programmiererfahrung haben. (II)

Auswertung hinsichtlich AR Anwendungsfälle

Betrachtet man die erhobenen Herausforderungen (2), (3) und (4) für Expert:innen, so lassen sich folgende mögliche Anwendungsfälle für AR ableiten. Crashes ließen sich mithilfe einer Kollisionsprüfung durch eine virtuelle Simulation der Robotertrajektorien reduzieren. Die Optimierung der Robotertrajektorie könnte man beschleunigen, indem nicht alle Punkte einzeln angefahren werden müssen, sondern der virtuelle Pfad optimiert wird. AR kann auch eine Lösung sein, um möglichst wenig Zeit in strapaziösen, engen oder gefährlichen Umgebungen zu verbringen. Roboterbahnen können per *Remote Teaching* angelernt werden, sodass die Bediener:innen unabhängig vom Standort des realen Roboters sind. Der zur Verfügung stehende Arbeitsraum kann per AR zunächst im realen Raum vermessen werden und ist anschließend virtuell, unterstützend beim Remote Teaching, vorhanden. Das Remote Teaching hat weiterhin den Vorteil, dass Ausfallzeiten von Produktionslinien reduziert werden können. Weiterhin können das Anlagenlayout, Arbeitsräume und Werkstückpositionierungen mit diesem Verfahren ohne kostspielige Montageaufwänden prototypisch getestet werden.

Für Laien bietet sich AR als Lösungsmöglichkeit für die Herausforderungen (1), (3) und (4) an. Das Verständnis für Koordinatensysteme kann durch das räumliche Anzeigen direkt am Roboter oder im Raum erhöht werden. Die Steuerung des Roboters und das

Einstellen von Konfigurationen lässt sich in AR unter Zuhilfenahme einer inversen Kinematik des virtuellen Roboters über die Manipulation des virtuellen Roboters lösen. Dies entspricht in etwa dem handgeführten Online Programmieren, das bei kollaborierenden Robotern meist möglich ist. Dabei entfällt das Erlernen der Steuerung via Teach Pendant. Mit einem AR Trainingsinterface kann auch die Angst im Umgang mit Robotern reduziert werden, da kritische Abläufe vorab in einer Trainingssimulation erlernt werden können.

Zusammengefasst ergibt die Analyse der Herausforderungen zwei Anwendungsfälle, die sich durch das Vorhandensein eines realen Roboters unterscheiden (Abbildung 1).

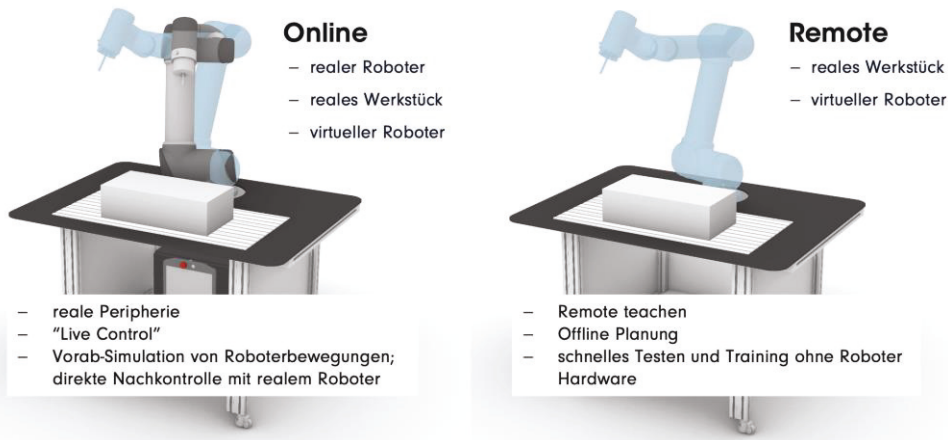


Abbildung 1: Anwendungsfälle Online und Remote Teaching

Szenario

Die gesammelten Informationen deuten darauf hin, dass Expert:innen in Teilbereichen – wie z. B. der Kollisionsprüfung – einen großen Mehrwert durch ein AR Interface erhalten können. Allerdings stellt diese Nutzergruppe sehr hohe Anforderungen an den Funktionsumfang der Bedienung und benötigen meist sehr spezifische Funktionen. Für Laien der Roboterprogrammierung kann ein AR Interface dagegen einen sehr schnellen Einstieg in die Roboterprogrammierung bieten. Als Zielgruppe werden aus den genannten Gründen deshalb Applikationsexpert:innen in KMU adressiert, die Kleinstserien automatisieren möchten, die bisher manuell bearbeitet wurden.

Als Applikationen werden Oberflächenbeschichtungen wie Lackieren, Sandstrahlen oder Pulverbeschichten gewählt, da sie sich als manuelle Aufgaben repetitiv und durch das Tragen von Schutzausrüstung als physisch anstrengend gestalten. Da die Losgrößen klein sind, müssen Pfade oft angepasst oder neu erstellt werden, ohne dass große Änderungen im Programmablauf selbst getätigt werden oder Endeffektoren getauscht werden müssen. Außerdem wird angenommen, dass die Anforderungen an die Genauigkeit bei diesen Applikationen geringer sind als bei bahnbasierten Anwendungen wie dem Schweißen. Aufgrund dieser Bedingungen wird das in Abbildung 2 dargestellte Szenario vorgestellt, das Remote und Online Teaching verbindet. Zunächst wird die Robotertrajektorie am realen Werkstück mithilfe der Gestenmanipulation der verwendeten AR HMD Technologie definiert. Die Arbeitsraumbeschränkung wird mit virtuellen Objekten eingegeben. Anhand der definierten programmierten Trajektorie kann die Position des Werkstückes relativ zum Roboter und die Bahn in Schritt 2 und 3 optimiert und korrigiert werden. Im letzten Schritt wird die Trajektorie an den realen Roboter übergeben, indem virtueller und realer Roboter mithilfe von Referenz-Trackern überlagert werden.

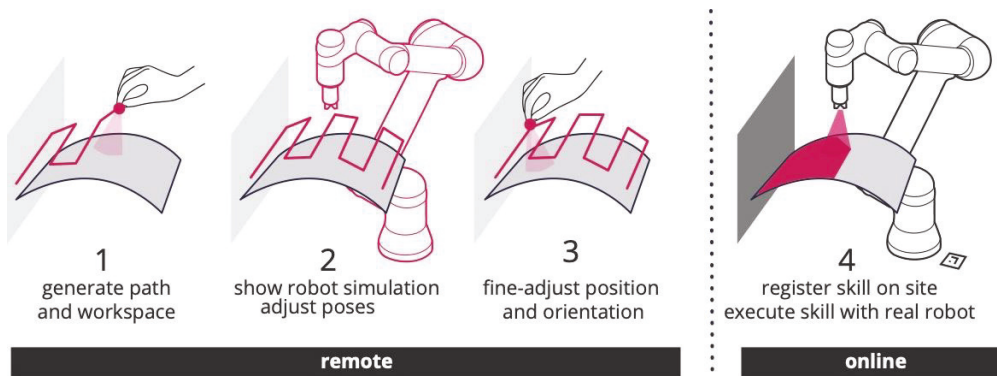


Abbildung 2: Arbeitsablauf des Szenarios

Umsetzung

Technisches Setup

Als AR HMD wird die *Microsoft HoloLens 2* Brille (abgekürzt: HoloLens) verwendet, die digitale Hologramme raumfest über die reale Welt legen kann. Dies wird mit einer Trägheitsmesseinheit, vier Kameras und einem Infrarot Tiefensensor erreicht. Für die

Entwicklung wird die Entwicklungsumgebung *Unity3D* in Verbindung mit dem *MixedReality Toolkit* (MRTK) von Microsoft genutzt. Das *Vuforia* Framework dient dem Markertracking.

User Interface

Wie in Abbildung 3 gezeigt, wird ein holografisches Hauptmenü (Abbildung 3) auf der Basis des `HandMenu_Large_AutoWorldLock_On_HandDrop` des MRTK entwickelt. Schauen die Nutzer:innen auf die offene Handfläche, so erscheint das Menü unabhängig vom Standpunkt der Brille im Raum neben der Hand. Es kann mit einer Greifgeste an eine beliebige Position im Raum verschoben und verdreht werden. In der Robotik ist das vorteilhaft gegenüber den üblichen schweren und kabelgebundenen Teach Pendants, da von überall und auch in engeren Räumen mühelos gearbeitet werden kann. Um das Blickfeld der HoloLens nicht zu stark einzuschränken, besitzt das Menü ein schmales horizontales Format, sodass es am Rand des Sichtfelds platziert werden kann. In dem hier gezeigten Prototyp können die Nutzer zwischen fünf Funktionalitäten auswählen: *Training*, *Workspace*, *Positioning*, *Skills* und *Setup*, die die Funktionalitäten des Szenarios abbilden.



Abbildung 3: holografisches Handmenü (Bildaufnahme HoloLens)

Insgesamt ergibt sich durch die gewählte Gestaltung der Navigation eine flache Hierarchie des Interfaces von maximal drei Ebenen. Das soll dazu führen, dass alle Funktionalitäten schnell zu erreichen sind und die Zielgruppe der Laien der Robotik nicht den Überblick verliert. Durch eine möglichst einfache und reduzierte Darstellung der Funktionen selbst (Icons und Texte) und ergänzenden Erklärungstexten wurde die Zielgruppe in der Gestaltung berücksichtigt. Zusätzlich wurde berücksichtigt, dass die kog-

nitive Belastung durch die Sicht durch die Brille potenziell erhöht ist. Eine Gegenmaßnahme der Gestaltung beruht auf möglichst großen Interface-Elementen und der Möglichkeit, das Interface in einem angenehmen Abstand zu positionieren.

Funktionalitäten

Der Prototyp für die AR HMD Programmierumgebung für das definierte Szenario umfasst folgende Funktionalitäten:

- Bearbeiten von Pfadpunkten mithilfe von Greif- und Ziehgesten
- Virtuelle Simulation des Roboters und der Roboterbewegung (s. Abbildung 4)
- Handgeführtes Einstellen von Roboterposen
- Anzeigen von Koordinatensystemen
- Anzeigen und Einstellen des Arbeitsraums per Handinteraktion mit Begrenzungsobjekten
- Registrierung der virtuellen Szene im Raum anhand von Referenzmarkern
- Trainingsszenario für Applikationen im Bereich Lackieren

Zusammenfassung und Ausblick

Betrachtet man die gestellte Forschungsfrage, so konnte mit dem entwickelten Prototyp eine erste Antwort gegeben werden. Das Ergebnis bildet ein AR Interface zur Roboterprogrammierung für das Anwendungsszenario „Oberflächenbeschichten“ mit kleinen Stückzahlen. Über dieses Szenario hinaus sind weitere Einsatzfälle, wie z. B. Handling denkbar. Offen bleibt derzeit, wie sich Eingabegeräte mit erhöhtem Funktionsumfang (z. B. Wandelbots iPad App) in den AR Prozess integrieren lassen. Beispielsweise ist denkbar, dass Experten einmalig für das Setup von Roboter, Programmabläufen, Endeffektor und TCPs verantwortlich sind und die AR Schnittstelle dann von den jeweiligen Applikationsexperten genutzt wird.

Studie und Methodik

Um die mit dem Prototyp getroffenen Annahmen zu prüfen, wird ein Usability Test konzipiert. Der Test soll Erkenntnisse zu folgenden Schwerpunktthemen generieren:

- *Handtracking und Sichtfeld.* Kann die Interaktionsmethodik der HoloLens innerhalb des Tests zufriedenstellend erlernt werden?
- *Interaktionselemente.* Sind die gewählten Interaktionselemente intuitiv bedienbar?
- *Qualität.* Wie zufriedenstellend ist die Usability und wie erleben die Nutzer den Prototypen auf einer hedonischen Ebene?

Dafür werden Beobachtung der Nutzer:innen und das *laute Denken* (engl. Think Aloud) sowie ein vorbereiteter Leitfaden für die Erhebung qualitativen Feedbacks verwendet. Die Eignungsbeurteilung der Testpersonen erfolgt mithilfe eines Fragebogens. Auswahlkriterien für die Proband:innen sind Vorerfahrung in technischen Berufen, erste Erfahrungen in der Robotik sowie handwerkliche Vorerfahrung. Explizit ausgeschlossen werden Expert:innen der Robotik. Um eine Ersteinschätzung zur pragmatischen und hedonischen Qualität (Stimulation, Identität) sowie zur Attraktivität zu erhalten, wird nach dem Usability Test zusätzlich eine quantitative Erhebung mit einem *Attrak-Diff*-Fragebogen durchgeführt (Hassenzahl et al.: 2008, S.78).

Erwartet werden Zweifel gegenüber dem (sofortigen) Einsatz der HoloLens in der Robotik aufgrund der wahrscheinlich ungenügenden genauen Registrierung im Raum durch Referenzmarker. Weitere bremsende Faktoren könnte das kleine Sichtfeld und die Stabilität des Handtrackings sein. Dagegen wird eine erhöhte hedonische Qualität des Prototyps erwartet, da die Technologie für viele Nutzer:innen neuartig ist und dadurch ein positives Nutzungserleben hervorrufen könnte.

Literaturverzeichnis

Augmentus (2021): Augmentus. <https://augmentus.tech>, abgerufen am 10.03.2021.

Guhl, J., Nikoleizig, S., Heimann, O., Hügler, J. & Krüger, J. (2019): Combining the Advantages of On- and Offline Industrial Robot Programming. In: *2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, 1567–1570.

Hassenzahl, M., Koller, F. & Burmester, M., (2008): Der User Experience (UX) auf der Spur: Zum Einsatz von www.attrak-diff.de. In: Brau, H., Diefenbach, S., Hassenzahl, M., Koller, F., Peissner, M. & Röse, K. (Hrsg.): *Tagungsband UP08*, 78–82, Stuttgart:Fraunhofer Verlag.

Heimann, O., & Guhl, J. (2020). Industrial Robot Programming Methods: A Scoping Review. In: *25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, 1, 696–703.

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)*, 1–9.

Quintero, C. P., Li, S., Pan, M. K., Chan, W. P., Loos, H. F. M. V. der, & Croft, E. (2018). Robot Programming Through Augmented Trajectories in Augmented Reality. In: *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 1838–1844.

Sanneman, L., Fourie, C. & Shah, J. A. (2020): The State of Industrial Robotics: Emerging Technologies, Challenges, and Key Research Directions. In: *arXiv preprint*, 2010.14537.

Shepherd, D. C., Kraft, N. A. & Francis, P. (2019). Visualizing the “Hidden” Variables in Robot Programs. In: *2019 IEEE/ACM 2nd International Workshop on Robotics Software Engineering (RoSE)*, 13–16.

Kontakt

Dipl.-Ing. Carolin Horn
Dipl.-Ing. Christoph-Philipp Schreiber
Wandelbots GmbH
Tiergartenstr. 38
01219 Dresden
www.wandelbots.com