

DLR-IB-MO-HF-2021-115

**Visualisierung lokalisierter
Ultraschalldaten am inspizierten
Prüfteil in AR mit Hilfe der
HoloLens 2**

Masterarbeit

Turgut, Lezgin



DLR

**Deutsches Zentrum
für Luft- und Raumfahrt**

Visualisierung lokalisierter Ultraschalldaten am inspizierten Prüfteil in AR mit Hilfe der HoloLens 2

Master-Thesis
zur Erlangung des akademischen Grades
Master of Science (M. Sc.)

Lezgin Turgut

2077569



Hochschule für Angewandte Wissenschaften Hamburg
Fakultät Design, Medien und Information
Department Medientechnik
Studiengang Digital Reality



Deutsches Zentrum für Luft- und Raumfahrttechnik e.V.
Institut für Instandhaltung und Modifikation
Fachbereich für Wartungs- und Reparaturtechnologien

Erstprüfer: Prof. Dr. Roland Greule

Zweitprüfer: M. Sc. Andreas Wilken

Hamburg, den 01.05.2021

Danksagung

An dieser Stelle möchte ich mich bei meinen Prüfern Herrn Prof. Dr. Roland Greule von der Hochschule für Angewandte Wissenschaften Hamburg (HAW) und Herrn M.Sc. Andreas Wilken vom Deutschen Zentrum für Luft- und Raumfahrt (DLR e.V.) für ihre intensive Betreuung und Unterstützung während der Zeit zur Erstellung meiner Masterthesis bedanken. Gleichzeitig haben mir beide Prüfer darüber hinaus die Teilnahme an interessanten Veranstaltungen und spannenden Projekten ermöglicht.

Insbesondere bedanke ich mich beim DLR e.V. und dessen Institut für Instandhaltung und Modifikation (MO) und dessen Abteilung für Wartung- und Instandhaltungstechnologien (WRT) am Zentrum für angewandte Luftfahrtforschung (ZAL) Finkenwerder in Hamburg für die Möglichkeit der Themenbearbeitung. Besonderen Dank geht dabei an die gesamte Abteilungsleitung, der Abteilungsleiterin Frau Rodeck und meinem tatkräftigen Kollegium neben Herrn Wilken mit Herrn Heilemann, Herrn Raddatz, Frau Schmied-Kowarzik und Frau Reisewitz.

Mir wurden unschätzbare Erfahrungen und Einblicke in die Luft- und Raumfahrt, sowie eine umfangreiche Beratung und Hilfe bei der Einarbeitung in die Theorien und Praxis gegeben. Ebenso wurden mir trotz der Covid-19-Pandemie Regelungen die Labore, Werkzeuge, Einrichtungen, Programme und Rechenanlagen so umfangreich und sicher wie möglich zur Verfügung gestellt.

Daneben war die gegebene Unterstützung vom „Forschungs- und Transferzentrum Digital Reality“ eine weitere Quelle, die ich dankend annahm.

Gesondert bedanke ich mich bei meinen Mitmenschen im näheren sozialen Umfeld, die mir besonders bei der Entstehungsphase dieser Arbeit Geborgenheit gaben und sich auch nach den vielen Stunden im Labor, Büro und Homeoffice für mich gesorgt haben und/ oder um mich besorgt waren.

Themenstellung zur Masterarbeit

für
Herrn cand. M.Sc. Lezgin Turgut
Matrikel- Nr.: 2077569

Zeitraum der Bearbeitung

6 Monate

Thema:

Visualisierung lokalisierter Ultraschalldaten am inspizierten Prüfteil in AR mit Hilfe der HoloLens 2

Übersicht

In der Luft- und Raumfahrt wurden immer komplexere Materialien und Verbundteile aufgrund der Verbesserungen in der Herstellungstechnologie verwendet. Einen Teil der gestiegenen Zuverlässigkeit dieser Teile bietet dabei der Fortschritt der zerstörungsfreien Prüfung mittels Ultraschall. Neben der Einführung automatisierter Verfahren haben sich auch die Ideen der Darstellungsmethoden von 2D auf 3D weiterentwickelt. Solche Darstellungen erleichtern die geometrische Zuordnung und vereinfachen das Verständnis der Messergebnisse. Um diese Messergebnisse noch verständlicher zu Gestalten macht eine Übertragung auf die digital „Erweiterte Realität“ (AR) Sinn, da so eine Überlappung zwischen analogem Bauteil und den digitalen Messdaten ermöglicht wird. Problematisch wird jedoch die Art, wie eine exakte Überlappung erzielt werden kann. Hier stellen die Möglichkeiten des Trackings sowie die Art der Informationsverarbeitung und Weitergabe zwischen den nötigen Komponenten eine entscheidende Rolle dar.

Aufgabenstellung

Für eine zukünftige Luftfahrzeuginstandhaltung wird am DLR Institut für Instandhaltung und Modifikation ein System entwickelt, welches die zerstörungsfreie Prüfung mittels Ultraschalls digitalisieren soll, um diese später an einen möglichen digitalen Zwilling anbinden zu können. In einem ersten Schritt soll die Position eines Ultraschallkopfes mit einem markerbasierten Trackingverfahren verfolgt werden, um so die aufgenommenen Ultraschalldaten mit Ortsinformationen zu verknüpfen. Nach erfolgter Abstraktion der aufgenommenen Ultraschalldaten soll eine prüfteilbezogene Visualisierung mit Hilfe der Datenbrille HoloLens 2 erfolgen. Ein späteres Wiederabrufen von gespeicherten Informationen und die korrekte Anzeige am Prüfteil soll ermöglicht werden.

Arbeitsschritte

- Einarbeitung in die Grundlagen zerstörungsfreier Prüfungsmethoden mittels Ultraschalls
- Einarbeitung in Trackingsysteme für xR-Technologien
- Einarbeitung in Kommunikationswege zwischen verschiedenen Komponenteneinheiten
- Erläuterung und Analyse von Visualisierungsideen der Anwendung in AR mit der HoloLens
- Integration der HoloLens 2 in ein externes Trackingsystem
- Realisierung einer Datenübertragung an die HoloLens 2
- Planung, Entwicklung und Implementierung der ausgewählten Visualisierung durch entsprechende Tools in einer Testumgebung
- Analyse und Bewertung der Implementierung
- Schriftlicher Bericht

Die Masterarbeit wird am Deutschen Zentrum für Luft- und Raumfahrt e.V. (DLR), Institut für Instandhaltung und Modifikation am ZAL in Hamburg Finkenwerder durchgeführt und seitens des DLR von Herrn Andreas Wilken betreut.

Für Planung, Berechnung, Fertigung, Untersuchungen und Dokumentation können Labore, Werkzeuge, Einrichtungen, Programme und Rechenanlagen des DLR benutzt werden. Die Unterstützung durch das Institut umfasst die Beratung sowie Hilfe bei der Einarbeitung in die Theorie und Praxis. Die Ergebnisse sind entsprechend dem Fortschritt der Arbeit, mindestens aber einmal wöchentlich mit dem Betreuer abzusprechen.

M.Sc. Andreas Wilken

(Betreuer DLR)

Abstract

Ultrasonic inspection data is visualized in the form of C and D images to perform simple or complex inspection tasks on probes. These visualization methods are an effective means of identifying and evaluating deficiencies inside the material. However, it is often difficult to interpret the results of the tested structure, so it is important to illustrate the data in a more intuitive way. One approach is to display the ultrasound data in “Augmented Reality” (AR). AR is a computer technology that supports human perception by merging real images and artificial, computer-generated data. With regard to ultrasonic testing, AR technology can be used to merge the ultrasonic test data (e.g. in the form of a C or D image) with the visual image of the structure being tested. In this way, the ultrasound image is visually linked to the tested component and appears less abstract and more understandable than the pure image. In addition, it is easier to evaluate the data and differentiate between actual defects and structure related characteristics. Future AR applications are not limited to ultrasonic testing, but also offers advantages for other non-destructive testing methods. However, the present work focuses on creating an impression of the potential of AR technology for ultrasonic testing and providing a solution for several different forms of data representation, possible with one data set.

Zusammenfassung

Ultraschallprüfdaten werden in Form von C- und D-Bildern visualisiert, um einfache oder komplexe Inspektionsaufgaben von Prüfteilen zu vollziehen. Dabei sind diese Darstellungsmethoden ein Mittel zur Identifizierung und Bewertung von Mängeln des vorliegenden Materials. Allerdings ist es keine simple Aufgabe die Ultraschallergebnisse mit der getesteten Struktur in Beziehung zu setzen. Daher ist es erstrebenswert, erweiterte Darstellungsmethoden zu entwickeln. Ein Ansatz ist die Darstellung der Ultraschalldaten in einer „Augmented Reality“-Umgebung (AR). AR ist eine computergestützte Technik, die die menschliche Wahrnehmung durch das Zusammenführen von realen Bildern und künstlichen, computergenerierten Daten unterstützt. In Bezug auf Ultraschallprüfungen kann die AR-Technologie verwendet werden, um die Ultraschalltestdaten (z. B. in Form eines C- oder D-Bilds) mit dem visuellen Bild der realen getesteten Struktur zusammenzuführen. Auf diese Weise wird das Ultraschallbild mit der getesteten Komponente verknüpft und die Daten lassen sich somit direkt am Prüfteil wiederfinden, als bei bisherigen Darstellungsmethoden. Darüber hinaus erleichtert das neue deckungsgleiche Bild die Auswertung der Daten und die Unterscheidung zwischen tatsächlichen Mängeln und strukturbezogenen Materialeigenschaften. Die vorliegende Arbeit fokussiert sich darauf einen Eindruck vom Potenzial der AR-Technologie für Ultraschallprüfung zu schaffen und einen Lösungsansatz zu liefern, bei dem mit nur einem Datensatz mehrere verschiedene Darstellungsformen möglich sind.

Inhaltsverzeichnis

Danksagung	II
Themenstellung zur Masterarbeit	III
Abstract	V
Nomenklatur.....	IX
1. Einleitung.....	12
1.1 Motivation	13
1.2 Geplante Vorgehensweise	13
1.3 Vorangestellter Systemaufbau.....	14
2. Grundlagen	16
2.1 Zerstörungsfreie Prüfungen mittels Ultraschall.....	16
2.2 Augmented Reality mit Microsofts HoloLens 2	19
2.3 Optische Trackingsysteme für xR-Technologien.....	21
2.4 Stand der Technik.....	26
3. Hardwareseitige Systembeschreibung.....	29
3.1 Planungsprozess	29
3.2 Erweiterter Systemaufbau	33
4. Softwareseitige Systembeschreibung	35
4.1 Übersicht	35
4.2 LabVIEW-Applikation.....	37
4.3 ROS-Applikation.....	39
4.4 HoloLens 2 Applikation (NDT-US_HL2).....	41
4.4.1 Bereitstellung und Definition der Entwicklungsumgebung	41
4.4.2 Applikation-Funktionen	42
5. Visualisierung von Ultraschalldaten für ein HMD	50
6. Integration der HoloLens 2 in ein externes, markerbasiertes, Outside-In Trackingsystem...56	56
7. Versuchsprotokoll	58
8. Fazit und Ausblick.....	64
Abbildungsverzeichnis	66
Tabellenverzeichnis.....	68

Quellenverzeichnis	69
9. Anhang	73
9.1 Berechnungen.....	74
9.2 Ergebnisse aus dem Versuchsprotokoll.....	75
9.3 Datenträger CD	76
9.4 Eigenständigkeitserklärung	77

Nomenklatur

Lateinische Formelzeichen

c	[m/s]	Schallgeschwindigkeit
d	[m]	Distanz / Tiefe
f	[Hz]	Frequenz
t	[s]	Laufzeit

Griechische Formelzeichen

λ	[m]	Wellenlänge
-----------	-----	-------------

Tiefgestellte Indizes

Avr.	Durchschnitt/ durchschnittlich
D	Tiefen-Wert
Dif.	Differenz
max	Maiximal
min	Minimal
P	Position
S	Samples
V	Amplitude-Wert
x	x-Richtung
y	y-Richtung
z	z-Richtung

Abkürzungen

A-Bild	Echobild am Ultraschallgerät
AR	Augmented-Reality
ARM	Advanced-RISC-Machines
ASIC	Application-Specific-Integrated -Circuit
B-Bild	Längsschnitt auf der Prüfspur durch das Werkstück
C-Bild	Projektion des Fehlers auf die Werkstückoberfläche
CFK	Kohlenstofffaserverstärkter Kunststoff
DLR	Deutsche Zentrum für Luft- und Raumfahrt
DoF	Degrees-of-Freedom
GPS	Global Positioning System
HF	Hochfrequent, nicht gleichgerichtet
HL2	HoloLens 2
HMD	Head-Mounted-Display
HPU	Holographic- Processing- Unit
HSV	Farbraum definiert über Hue (Farbwert) Saturation (Sättigung) Value (Helligkeit)
IMU	Inertial-Measurement-Unit
IR	Infrarot
KI	Künstliche Intelligenz
KNN	Künstliches neuronales Netz
KS	Koordinatensystem
L.	Links / linke Seite
M.	Mitte / mittig
MIT	Massachusetts-Institute-of-Technology
MP	Megapixel
MRTK	Mixed-Reality-Tool-Kit
NDT	Non-Destructive-Testing
O.	Oben / oberhalb
QR	Quick-Response Code
R.	Rechts / rechte Seite
RE	Rückwandecho
RGB	Additiver Farbraum mit Rot, Grün und Blau als Grundfarben
RJ	Registered-Jack
ROS	Robotic-Operation-System

SAFT	Synthetic-Aperture-Focusing-Technique
SE	Sende-Empfang
SI	Sendeimpuls
SLAM	Simultaneous-Localization-And-Mapping
TCP	Transmission-Control-Protocol. Ein verbindungsorientiertes Netzwerkprotokoll.
ToF	Time-of-Flight
U.	Unten / unterhalb
UDP	User-Datagram-Protocol. Ein verbindungsloses Netzwerkprotokoll
US	Ultraschall / Ultrasonic
USB	Universal-Serial-Bus
UT	Ultrasonic-Testing
UWP	Universal-Windows-Platform
VR	Virtual-Reality / Virtuelle Realität
WDP	Windows-Device-Portal
xR	Extended-Reality. Ein Sammelbegriff für VR, AR, MR.
ZAL	Zentrum für Angewandte Luftfahrtforschung
ZfP	Zerstörungsfreien Prüftechniken

1. Einleitung

Um eine dokumentierbare Gewährleistung von Funktion und Qualität eines physischen Produkts zu erhalten wird oftmals eine Materialprüfung während der Herstellung, am Endprodukt oder zur Wartung sowie Instandhaltung durchgeführt [Hof19, S.1]. Dabei werden bildgebende, zerstörungsfreie Prüfverfahren eingesetzt, um Beschädigungen zu erkennen und einen Ausfall des Produkts abzuwenden. *„Die Prüfdaten werden, gerade bei der Untersuchung größerer Bereiche, nach genormten Methoden rechnergestützt, graphisch aufbereitet und über einen Monitor angezeigt“* [Hol17, S.2]. Eine direkte Zuordnung der Daten ist dadurch in vielen Fällen jedoch erschwert. Dies ist darin zu begründen, dass die verwendeten Darstellungsverfahren ein hohes Abstraktions- und Interpretationsvermögen der prüfenden Person erfordern. Darüber hinaus sind die konventionellen Methoden nicht in der Lage, einen räumlichen Bezug zwischen den Prüfdaten und der geprüften Struktur zu vermitteln. Insbesondere bei Schäden, die nach Außen keine sichtbaren Merkmale zeigen, sind die Erkennung und die Unterscheidung zwischen tatsächlichen Schäden und Bauteil bedingten Eigenschaften daher eine anspruchsvolle Aufgabe, was sich allgemein auch in der Dauer der Ausbildung des Personals [Deu19, S.164] und beispielsweise speziell in der Qualifikationsnorm DIN EN 4179 für Luft- und Raumfahrt widerspiegelt.

Generell wird eine zerstörungsfreie Prüfung (ZfP) mit dem Ultraschallverfahren empfohlen, da dieses Verfahren im Gegensatz zu anderen Methoden eine äquivalente Qualität zur Erkennung von Innen- sowie Außenfehlern in schallleitfähigen Werkstücken aufweist [Deu97, S338 -340].

Im Rahmen des *„Applikationszentrum MRO 4.0“* arbeitet das Deutsche Zentrum für Luft- und Raumfahrt e.V. (DLR) im Zentrum für Angewandte Luftfahrtforschung (ZAL) an neuen Technologien zur Realisierung zukünftiger Digitalisierungs- und Automatisierungsprozesse in der Instandhaltung. Ein Aufgabenteil ist dabei die *„Visualisierung lokalisierter Ultraschalldaten am inspizierten Prüfteil in AR mit Hilfe der HoloLens 2“*, um einen möglichen Ansatz für die Lösung der beiden Problematiken (Lokalisierung und Interpretation der Daten am Prüfteil) zu liefern.

Im Verlauf dieser Arbeit wird daher aufgeführt, welche Hardware- und Softwarekomponenten für den Ansatz verwendet werden und wie diese miteinander agieren. Zusätzlich werden eigene visuelle Konzepte für Ultraschalldaten in einem Versuch erläutert und bewertet.

1.1 Motivation

Wenn es gelingt, eine Anwendung für die HoloLens 2 zu entwerfen, die es im Vergleich zu herkömmlichen Darstellungsformen von Ultraschallmessungen schafft, die Ergebnisse grafisch und lokalisierbar auf das Prüfteil deckungsgleich zu projizieren, ist somit ein räumlicher Bezug zwischen den Prüfergebnissen und der geprüften Struktur gegeben, was die Stärke der Abstraktion der Daten verringert. Sollte es zudem realisiert werden neue Darstellungsformen zu schaffen, die ein Verständnis der gemessenen Daten erleichtern, kann damit schneller unterschieden werden, ob tatsächliche Schäden vorliegen, oder Bauteil bedingte Eigenschaften angezeigt werden. Es ließe sich zudem protokollieren, auf welche Art und Weise der Prüfkopf bewegt wurde, was für eine Bewertung der Effizienz oder zu Schulungszwecken eingesetzt werden kann, um neuen Personal die Methodik der Ultraschallprüfung anwendungsnah und verständlicher zu vermitteln.

1.2 Geplante Vorgehensweise

Beginnend soll ein grundlegendes Verständnis der behandelten Thematiken erfolgen. Dabei werden als Ausgangspunkt, in Abschnitt 1.3 die bisherigen Ergebnisse in Form eines vorangestellten Systemaufbaus erläutert. Anschließend werden in Kapitel 2 die essentiellen Inhalte zu zerstörungsfreien Prüfungsmethoden mittels Ultraschall erklärt und auf die konventionellen Darstellungsmethoden und deren Interpretation eingegangen. Daneben werden die wichtigsten Kausalitäten in Bezug auf Augmented Reality und im speziellen der HoloLens 2 (HL2) definiert. Parallel werden verschiedene Strategien zur Positionserfassung erläutert und dadurch insgesamt ein umfängliches Wissen für die relevanten Thematiken bereitgestellt, die für ein Verständnis des nachfolgenden Systems notwendig sind. In Kapitel 3 wird der Entwicklungsprozess von der Planungsphase bis zur hardwareseitigen Umsetzung thematisiert, wobei detailliert auf die Definition von verschiedenen Lösungsmöglichkeiten und dessen Analyse eingegangen wird. Nach der Aufdeckung der einzelnen Hardwarekomponenten erfolgt in Kapitel 4 die softwareseitige Systembeschreibung, in der die Datenwege und Datenumwandlungen von einer verwendeten Softwarestruktur zur Anderen beschrieben werden. Zudem wird auf den Funktionsumfang der HL2 eingegangen. Ergänzend erfolgen in den Kapiteln 5 und 6 eine Dokumentation über den Versuch der Umsetzung einer alternativen Tracking-Methode sowie die Darstellungsmöglichkeiten von Ultraschallwerten im dreidimensionalen Raum. In Kapitel 7 wird der Einsatz der entwickelten AR-Anwendung in einem Laborversuch geplant, durchgeführt und analysiert.

Zusammenfassend und abschließend endet diese Arbeit nach Kapitel 8 mit der Erschließung eines Fazits und dem Ausblick auf mögliche erweiterte Themenfelder, die zukünftig behandelt werden können.

1.3 Vorangestellter Systemaufbau

Das Thema der „Visualisierung lokalisierter Ultraschalldaten am inspizierten Prüfteil“ wird fokussiert beim DLR im ZAL behandelt. Hierzu entstanden Konzepte, Hardwarelösungen und Umsetzungen, die für diese akademische Arbeit ergänzt wurden, um eine „Visualisierung lokalisierter Ultraschalldaten am inspizierten Prüfteil in AR mit Hilfe der HoloLens 2“ zu realisieren. Dies stellt ein Teilprodukt einer Reihe von weiteren Projekten dar, die am Institut für Instandhaltung und Modifikation für die Virtualisierung und Digitalisierung von Wartungs- und Reparaturtechnologien erforscht und entwickelt werden. Der Grundaufbau des vorangestellten Systemstands dieses Projektes besteht, wie in Abb. 1 zu sehen, aus sieben Hardwarekomponenten.

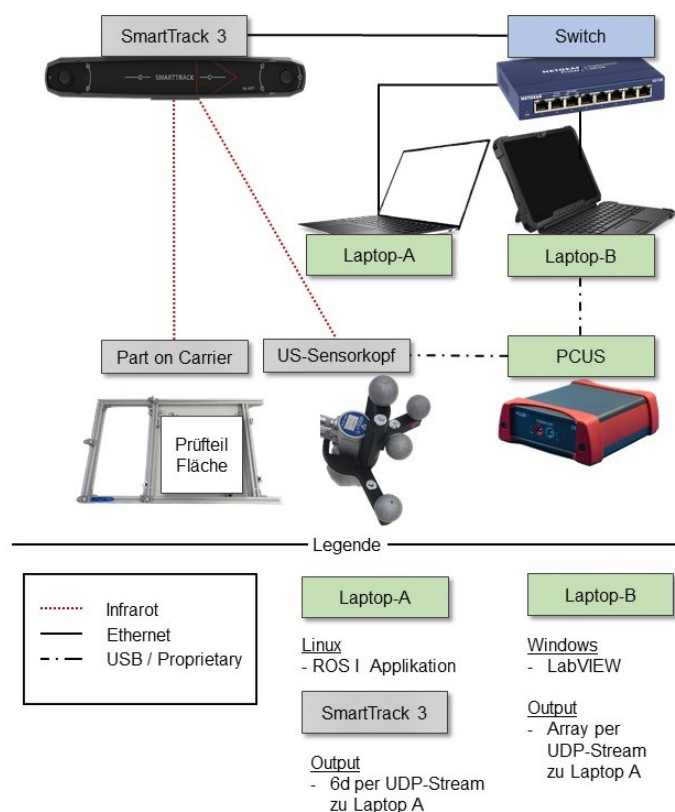


Abb. 1: Schematische Darstellung des vorangestellten Systemaufbaus mit Realbildern ohne einheitlichen Maßstab

Der *Part-on-Carrier* ist ein Metall-Steck-Gestell, an dem retroreflektierende Marker befestigt sind. Dieses Gestell wird je nach Anwendungsfall entweder an das zu untersuchende Prüfteil befestigt oder das Prüfteil auf die Prüffläche gelegt. Es dient als Referenzpunkt, an dem ein digitales Koordinatensystem aufgespannt wird, auf das sich das gesamte virtuelle System bezieht.

Der *Ultraschall (US) – Sensorkopf* (verwendet wird der Einkanal-Sende-Empfang-Prüfkopf namens SE 10/6 PB 4 C), ist eine Komponente zum Senden und Empfangen von Ultraschallwellen. Hier sind ebenfalls retroreflektierende Marker mit Hilfe eines eigens entworfenen und mit einem 3D-Drucker erstellten Marker-Gestells befestigt.

Der Sensorkopf wird händisch geführt und liefert in Verbindung mit dem Hardwarebauteil *PCUS Pro*

Single vom Fraunhofer-Institut und dessen Signalumwandlung digitalisierte Ultraschalldaten. Die beiden Komponenten werden mit einem SEKG2 53887-Kabel verbunden. Der PCUS wiederum wird mittels USB2 Anschluss an das *Laptop-B* angeschlossen. Dieses Laptop wird über ein Windows 10 Betriebssystem ausgeführt, da der *PCUS* und dessen Treiber sowie die benötigte Software LabVIEW hierfür kompatibel sind. Mit Hilfe der Systementwicklungssoftware LabVIEW von National Instruments Corp. wird der *PCUS* in Verbindung mit dem Sensor eingestellt sowie die Sensorrohdaten gefiltert und für den digitalen Transport vorbereitet. Der *Laptop-B* ist mit einem RJ45 Ethernetkabel über einen *Switch* mit *Laptop-A* verbunden, worüber die Ultraschalldaten gesendet werden.

Laptop-A wird über eine Linux-Distribution betrieben, um das Framework ROS I (Robotic Operation System I) stabil nutzen zu können. ROS dient in diesem Projekt zur Transformation der aufgenommenen Positionsdaten und der anschließenden Zusammenführung dieser Informationen mit den dazugehörigen Ultraschalldaten.

Die Positionsdaten werden unter Einbeziehung einer Infrarot-Stereokamera (*SmartTrack 3*) ermittelt, die mit einem RJ45 Kabel über den *Switch* mit *Laptop-A* verbunden ist. Der *SmartTrack 3* sendet Infrarotlichtstrahlen aus und kann dessen Reflektionen durch die Marker-Bäume vom *Part-on-Carrier*- und *US-Sensorkopf* empfangen und daraus die Position beider Objekte bestimmen. Aus diesen Daten (6d-Array) werden in *Laptop-A* mit ROS ein digitales Referenz-Koordinatensystem zum Prüfteil aufgespannt und die Messpositionen mit Werten vom *US-Sensorkopf* verknüpft.

Eine in der Programmiersprache Python3 entwickelte Anwendung kann bereits diese Daten grafisch in 2D auf dem Laptop darstellen. Dabei werden sinnbildlich auf einem digitalen weißen Blatt die ortsbezogenen Messungen je nach Position und Messwert farblich eingezeichnet.

2. Grundlagen

In diesem Abschnitt werden Grundlagen über die zerstörungsfreie Prüfung mittels Ultraschall und dessen Darstellungsformen aufgeführt. Zudem erfolgt eine Erläuterung über den generellen Einsatz von Augmented Reality und dessen Abgleich von Realität und Virtualität. Anschließend werden die heutigen Möglichkeiten zur Visualisierung von Daten, bezogen aus zerstörungsfreien Prüfungen in Kombination mit „Extended Reality“ (xR) veranschaulicht, um ein Gesamtverständnis der behandelten Inhalte sicherzustellen.

2.1 Zerstörungsfreie Prüfungen mittels Ultraschall

Die Ultraschallprüfung (Abkürzung UT für Ultrasonic Testing) gehört zu der Familie der zerstörungsfreien Prüftechniken (kurz ZfP oder auch NDT für Non-Destructive-Testing), die auf dem Prinzip der Ausbreitung und Reflexion von Ultraschallwellen in dem getesteten Objekt oder Material (Prüfteil) beruht. Als zerstörungsfreie Prüfmethode dient die Ultraschallprüfung dazu Fehler und Defekte zu erkennen, ohne das Prüfteil zu beschädigen und einen Ausfall eines Teils, einer Komponente oder eines Systems, z. B. die Tragelemente eines Flugzeugs, zu verhindern. Dadurch lassen sich Bauteile ggf. auch im eingebauten Zustand prüfen, ohne diese demontieren zu müssen, sofern dies überhaupt noch ohne Zerstörung möglich ist [Deu97, S.4].

Schall ist, vereinfacht dargestellt, die mechanische Bewegung von gasförmiger, flüssiger oder fester Materie. Atome und Moleküle schwingen periodisch um ihre Ruhelage und übertragen ihre Bewegung wiederum auf benachbarte Atome und Moleküle. Die Schwingung breitet sich dabei mit einer materialspezifischen Schallgeschwindigkeit (c) aus. Dabei lässt sich feststellen, dass sich Schwingungszustände in festen Abständen wiederholen. Dieser Abstand heißt Wellenlänge (λ). Die Anzahl der Schwingungen pro Sekunde bezeichnet man als Frequenz f , die in Hertz (Hz) angegeben wird. Wellenlänge und Frequenz hängen über die Schallgeschwindigkeit wie folgt voneinander ab:

$$c = f \cdot \lambda \quad 1$$

Für die Schallgeschwindigkeit wird meistens die Ausbreitung der Longitudinalwelle, auch Längswelle genannt, angegeben, welche in Ausbreitungsrichtung schwingt. Ihr Gegenstück ist die Transversalwelle, deren Amplitude senkrecht zur Ausbreitungsrichtung steht. Der Abstand bzw. Tiefenwert d einer reflektierenden Wand oder eines Fehlers kann aus der materialspezifischen Schallgeschwindigkeit c und der Laufzeit t wie folgt bestimmt werden [Deu97, S.4]:

$$d = \frac{c \cdot t}{2} \quad 2$$

Die häufigste Form der Ultraschallprüfung basiert auf der Impuls-Echo-Technik (Abb. 7). Hier werden Schallwellen mit Frequenzen zwischen 0,2 MHz und 100 MHz [Deu97, S.5] in das zu prüfende Bauteil

durch einen Sender-Empfänger (SE) Prüfkopf induziert (siehe Abb. 2). Ein Kopplungsmittel aus Gel, Öl oder Wasser stellt den gleichmäßigen Transfer der Ultraschallwellen in das Prüfteil sicher und dient als zusätzlicher Reibungsschutz. Die Pulsdauer beträgt dabei einige Mikrosekunden. Diese Schallimpulse breiten sich im Prüfteil mit charakteristischer, materialabhängiger Schallgeschwindigkeit aus. Orte im Bauteil, an denen sich die Ausbreitungsgeschwindigkeit der Ultraschallimpulse aufgrund einer Absorption/ Transmission oder Reflexion ändert, werden Grenzfläche genannt. Die Reflexion nennt sich „Echo“. Echos treten insbesondere bei Ungängen, wie Poren, Hohlräumen oder Rissen auf, welche sich Fehlerechos nennen. Im Gegensatz dazu treten auch Reflexionen an der „Rückwand“ des Testmaterials auf, die entsprechend als Rückwandechos (RE) betitelt werden.

Die reflektierten Schallimpulse werden vom SE-Prüfkopf registriert. Aus der verstrichenen Zeit zwischen dem Senden eines Schallimpulses (Sendeimpuls, SI) und dem Empfangen eines Echos kann der Ort bzw. die Tiefe des Echopunkts und damit die Position der Reflexionsquelle ermittelt werden. Ein separates Diagnosegerät zeigt diese Ergebnisse nach internen Berechnungen z.B. in Form eines Signals mit einer Amplitude (A-Bild) an, die die Intensität der Reflexion und die Entfernung darstellt [Deu97, S.55-62] (siehe Abb. 3 & 4) Dies ist nur ein Beispiel zur bildgebenden Ultraschallprüfung.

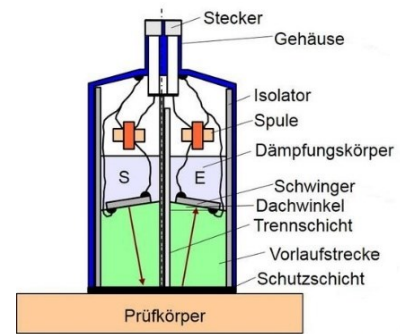


Abb. 2: Aufbau eines SE-Prüfkopfes [PoU]

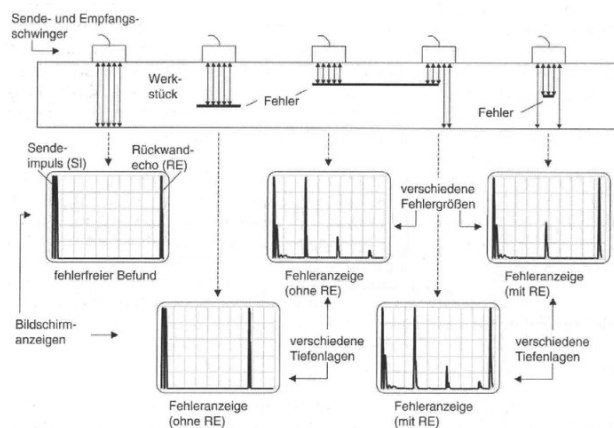


Abb. 3 (L): Beispielhafte Darstellung eines Reflektogramms (A-Bild) mit 5 verschiedenen Testfällen. [Chr15, S.9]

Abb. 4 (R): Beispiel eines handgeführten SE-Prüfkopfes mit einem Echograph mit A-Bild. [KDP21, S.1]

Bildgebende Ultraschallprüfung

In der Gesamtbetrachtung umfasst die bildgebende Ultraschallprüfung prinzipiell alle Darstellungsweisen (vorzugsweise die gemessenen Sensor-Amplituden) in Abhängigkeit von der Zeit und/ oder dem Ort. Sie reichen von der einfachen HF-Bild-Darstellung mittels eines simplen Oszillographen, über das A-Bild, B-Bild, C/D-Bild bis hin zum 3D und 4D-Bild. Dabei unterscheiden sich die Bilder je nach

Darstellung in ihrem Aussagegehalt (Abb. 5 & 6). Das A-Bild wird dabei seit den Anfängen der Ultraschallprüfung eingesetzt, welche je nach Quelle zwischen den Jahren 1928 und 1937 datiert werden [Deu97, S.1]. Erst Jahrzehnte später wurde das B-Bild, welches einen Querschnitt liefert und der Flächen-Scan (C/D-Bild) technisch nutzbar. Der Scan ist dabei eigentlich eine Rasterung, weil die A-Bilder je linien- oder flächenartig als Bild zusammengesetzt werden [Deu97, S.62]. Damit erhält jedes A-Bild einen Bildpunkt als Farb- oder Grauwert. Im C/D-Bild werden diese Bildpunkte zusammengesetzt und die Farbwerte relativ zu der Skalierung der verwendeten Farbpalette eingestuft (Abb. 6).

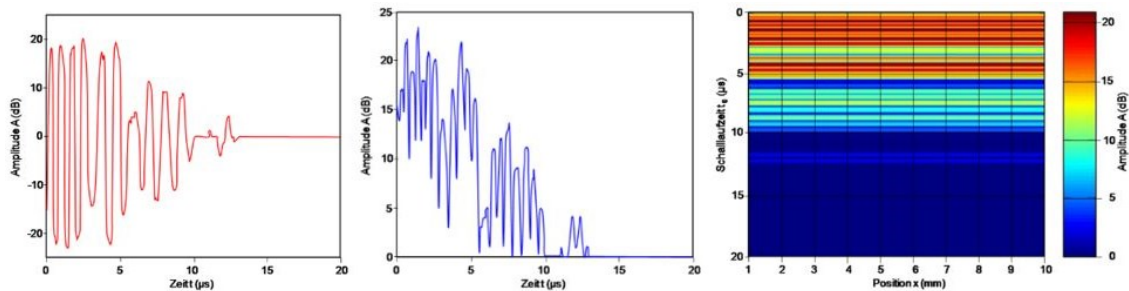


Abb. 5: Darstellungsmethoden. Logarithmisches HF-Bild (L.), A-Bild (M.) mit den absoluten Amplitudenwerten und B-Bild (R.), was pro Einheit in x- Richtung jeweils ein A-Bild darstellt, welches um 90° nach rechts gedreht wurde [PolB]

Für diese Arbeit haben sich die Eigenschaften des C/D-Bilds am geeignetsten herausgestellt, da sie eine Art „Draufsicht“ [Deu97, S.62] auf das Prüfteil ermöglichen und gleichzeitig ein 2D-Bild generiert wird, welches außerhalb der Anwendung weiterverwendet werden kann. Gleichzeitig ist der benötigte Datensatz geringer, als bei 3D- und 4D- Methoden. Der Unterschied zwischen einem C-Bild und einem D-Bild liegt in der Einbeziehung der verwendeten Daten. Ein C- Bild wird aus den Amplitudenwerten generiert und beschreibt neben der oberflächlichen Lage die Art eines Fehlers, während ein D-Bild aus den verortbaren Tiefenwerten (Laufzeit zwischen den Echos) des Signals berechnet wird (Formel 2).

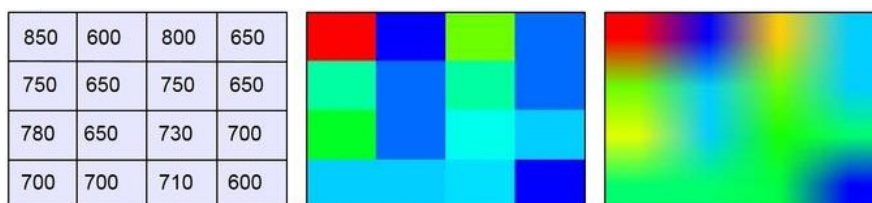


Abb. 6: C-Bild Darstellung. Beispiel einer Matrix aus Amplitudensummen der A-Bilder (L.), Matrix in einer Farbcodierung (M.) und die geglättete Darstellung der Farbmatrix (R.) [PolC]

Durchführung einer zerstörungsfreien Ultraschallprüfung

Eine Ultraschallprüfung kann über verschiedene Wege und Werkzeuge in verschiedensten Ausführungen vollzogen werden. Neben dem einfachen Einzelkopf, der nur für das Senden oder Empfangen von Ultraschallwellen zuständig ist (siehe Abb. 7), gibt es ebenso die bereits erwähnten SE-Prüfköpfe. Diese werden in einer einfachen Variante (Single SR), oder als Array angefertigt, in der mehrere Prüfköpfe verbaut sind und somit andere Aufnahmewinkel und Auswertungen möglich sind (siehe Abb. 8).

Prüfköpfe können per Hand, mechanisch, automatisch via Roboter oder stationär (demnach gar nicht) geführt werden. Das Messsignal wird je nach Darstellungsmedium auf einem externen Prüfgerät in einfacher Form dargestellt und von der prüfenden Person direkt während der Messung interpretiert. Ein Protokoll wird meist handschriftlich erstellt [Deu97, S.203] oder die erkannten Mängel werden auf dem Prüfteil selbst festgehalten. Hierbei erhält die prüfende Person keine weitere technische Unterstützung. „Eine stetige Interpretation der Messwerte, Kontrolle der vollständigen Abdeckung des Prüfbereichs und die Erstellung des Prüfprotokolls verlangen dabei eine hohe Expertise des Prüfers“ [Fra21].

Aus vorgegebenen und daraus empfohlenen Gründen wird in dieser Arbeit das Impuls-Echo-Verfahren mit einer Direktankopplung eines händisch geführten, einfachen SE-Prüfkopfes verwendet.

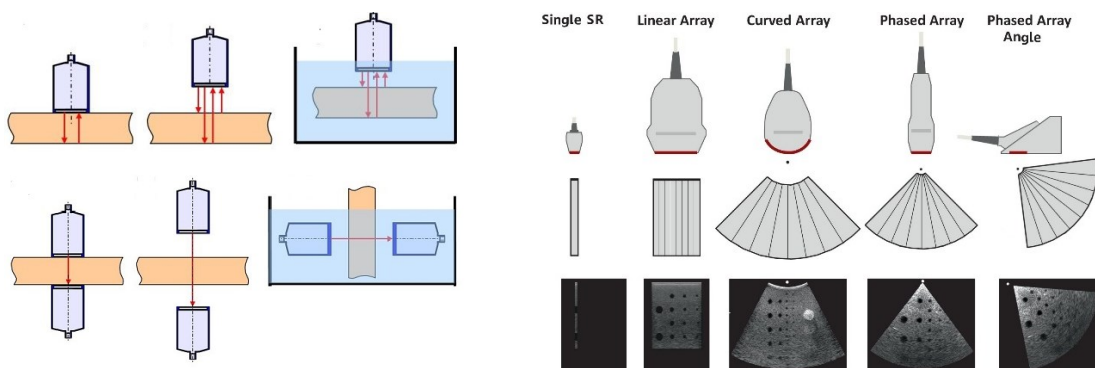


Abb. 7 (L.): Schematische Darstellung Impuls-Echo-Verfahren (O.) und Durchschallverfahren (U.). Jeweils bei Direktankopplung (L.), mittels Luftultraschall (M.) und Tauchbad-Technik (R) [PolIE].

Abb. 8 (R.): Variationen von Ultraschall-Prüfköpfen und deren bildgebende Eigenschaft in Anlehnung an [JKV18, S.290]

Einsatzgebiete

Zusammengefasst werden Ultraschallprüfungen an Metallen und Legierungen durchgeführt, die jedoch auch an Beton, Holz und Verbundwerkstoffen angewendet werden können. Das Verfahren wird in vielen Branchen eingesetzt, einschließlich der Metallurgie, Fertigung, Bauwesen, Luft- und Raumfahrt, Automobilindustrie und anderen Transportsektoren. Zudem zeigen diese Bereiche ein zunehmendes Interesse an dem Einsatz von Augmented Reality in zerstörungsfreien Prüfverfahren, was im Abschnitt „Stand der Technik“ näher erläutert wird.

2.2 Augmented Reality mit Microsofts HoloLens 2

Augmented Reality (kurz AR, zu Deutsch sinngemäß; Digital erweiterte Realität) ist die dynamische Verbindung von realen und virtuellen Bildern in Echtzeit [Dör13, S. 245]. Im Gegensatz zur virtuellen Realität (VR), in der die benutzenden Personen vollständig eine künstliche Welt wahrnehmen, überlagert Augmented Reality zusätzliche digitale Informationen zur realen Welt. So werden z.B. computer-generierte Grafiken, wie 3D- Modelle/ Grafiken, Pfeile, Texte und weitere Hervorhebungen in das Sicht-

feld der benutzenden Personen eingeblendet, um zusätzliche Informationen über die Umgebung zu erhalten. Dies soll z.B. helfen Aufgaben effizienter zu bearbeiten [Dör13, S. 241].

Aktuell sind mehrere AR-Hardware- und Software-Systeme auf dem Markt erhältlich. Grundsätzlich kann man diese zwischen Handgeräten (engl. Handheld) wie dem. Apple™iPad mit ARKit und tragfähigen Geräten (engl. Wearables), wie die Datenbrille Microsoft™HoloLens 2 mit MRTK, unterscheiden. Ein Vorteil von Datenbrillen gegenüber Handgeräten ist, dass sie am Kopf getragen werden. Diese Art von Wearables werden auch als Head-Mounted Displays (HMD) bezeichnet. Mit Hilfe von HMD's bleiben beide Hände weiterhin frei zum Agieren und Informationen stehen flexibel im Fokus, ohne dabei das Sichtfeld zwangsweise verändern zu müssen. Somit gibt es diesbezüglich keine Einschränkung beim Tragen [Sch19, S.1].

Die HoloLens 2 (HL2) (siehe Abb. 9) ist dabei die zweite Generation von Microsofts Head-Mounted Augmented Reality Geräten, bestehend aus einem visier-basierten durchsichtigem Display mit Komponenten, die im Kopfbügel verbaut wurden. Sie verfügt über vier Kameras für sichtbares Licht zur Verfolgung der Kopfausrichtung, zwei Infrarotkameras für die Augenverfolgung, einem *Time-of-Flight* (ToF) -Tiefensensor, einer *inertialen Messeinheit* (IMU) bestehend aus Beschleunigungsmesser, Gyroskop und Magnetometer, einer 8MP-Standbild- / Videokamera (1080p30) sowie einem 5-Kanal-Mikrofonarray und zwei verbauten 3D-Lautsprechern [Mic20].

Die HL2 ist ein eigenständiges, eingebettetes Windows 10 *Ein-Chip-System* mit ARM - Architektur. Das implementierte Bluetooth (5.0) und integrierte WLAN (802.11ac) sorgen dafür, dass die Brille als *Client* für Daten fungiert, die sich entweder auf dem Gerät selbst, in Microsofts Azure-Cloud oder allgemein im Internet befinden [Mic20].

Die verbaute Holographic- Processing- Unit (HPU) ist eine eigene *anwendungsspezifische integrierte Schaltung* (ASIC). Das Design kombiniert die verschiedenen Hardware Module für die Signalverarbeitung mit einem *künstlichen neuralem Netz* (KNN), um die rechenintensive Aufgabe schnell und richtig zu bewältigen, ortsgebundenen 3D Inhalte möglichst positionsgenau darzustellen, während sich der Kopf der tragenden Person bewegt [Mic17].



Abb. 9: HoloLens 2 Explosionsdarstellung [Mic20].

Anwendungen können für die HoloLens in Form von *Universal-Windows-Platform* – Apps (UWP) entwickelt werden. Die gängige Praxis dabei ist mittlerweile die Einbeziehung einer grafischen Laufzeitumgebung (engl. Game-Engine), wie Unity™, da sich so ein vielschichtiger Funktionsumfang, wie die Inhaltspipeline und die Scripting-Infrastruktur an einem Ort befindet [Sch19, S.1]. Mit Hilfe von Microsofts *Mixed Reality Tool Kit* (MRTK) lassen sich in Unity Anwendungen für die Brille und dessen Sensoren schreiben. Das Prinzip des *freizügigen Lizenzmodells* (MIT License) von Unity und dem MRTK sorgt zudem für eine große und offene Entwickler-Gemeinschaft, die im Austausch steht und die ihre Erfahrungen und Lösungsansätze in Internetforen teilt. So können z.B. unter anderem allgemein für AR und VR stetig neue oder zusätzliche Trackinglösungen entwickelt werden, die in Unity mithilfe von Plugins in die Anwendung implementiert und optimiert werden können.

2.3 Optische Trackingsysteme für xR-Technologien

„Tracking“ (zu Deutsch Verfolgung) wird im xR-Kontext als „Messung von Positions- und Rotationsinformationen (d.H. Orientierung) von Objekten oder Personen [...]“ bezeichnet [ART19, S18]. Allgemein ist die Bestimmung der Position und Orientierung der benutzenden Person oder eines Objekts innerhalb der Umgebung eine wichtige Funktion in VR und AR, die es dem System ermöglicht, das Sichtfeld und die Perspektive zu erkennen, sodass die virtuellen Elemente entsprechend dargestellt werden können. Für eine vollständige Bewegungsverfolgung muss das Trackingsystem die Bewegung in den sechs Freiheitsgraden (6DoF) messen können. Diese bestehen aus den Translationen der drei senkrechten Achsen (x, y, z), die als Links- / Rechts- (Schwankung), Aufwärts- / Abwärts- (Hub), Vorwärts- / Rückwärts- (Schwall) und Drehung der drei senkrechten Achsen, die als Nicken (Querachse), Gieren (Normalachse) und Rollen (Längsachse) bezeichnet werden [Dör13, S. 99] (siehe Abb. 10).

Ein Trackingsystem besteht aus der Kombination von Hardware und Software, welche die Koordinatensysteme beider Welten (real und virtuell) erfasst und mathematisch zusammenführt, damit diese visuell überlagert werden können. Es gibt verschiedene Methoden zur Positionsverfolgung, dessen Auswahl von verschiedenen Faktoren abhängen [Bog14], wie z. B.:

- Genauigkeit des Trackings
- Benötigte Messwiederholfrequenz
- Größe des Verfolgungsbereichs
- Art der Umgebung (Raum- oder Außenbereich)
- Stromverbrauch
- Benötigte/verfügbare Rechenleistung
- Kosten
- Starre oder sich bewegende Objekte
- Gespeicherte Objekte oder Objekte, die sich zur Programmlaufzeit flexibel ändern können

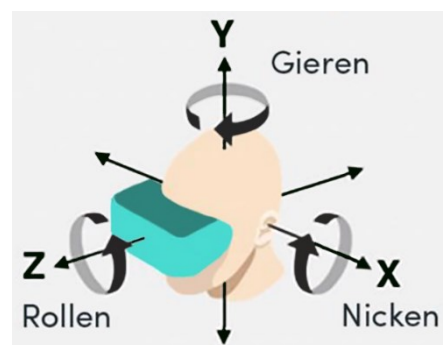


Abb. 10: Die 6 Freiheitsgrade der 3 senkrechten Achsen

Die für diese Arbeit relevanten Trackingsysteme verwenden aufgrund ihrer Genauigkeit und Zuverlässigkeit im Arbeitsbereich die Methode des optischen/ visuellen Trackings. Die Gemeinsamkeit der Umsetzungen dieser Methode besteht in der Verwendung von Kameras zum Sammeln von Positionsinformationen [Bog14].

Die eigene Kopfhaltung kann verfolgt werden, indem in einem Videostream Umgebungsmerkmale erkannt und analysiert werden. Typischerweise wird für AR, wie der HoloLens, und einige VR Systeme (z.B. der Oculus Quest™) eine oder mehrere Kameras auf dem am Kopf montierten Display angebracht. Eine interne oder externe Computereinheit berechnet die Pose dieser Kamera(s) bzw. des HMD in Bezug auf bekannte Funktionen in der Umgebung, wie dem Boden, Wände und sonstige Flächen und Kanten. Dies ist nur eine von vielen optischen Tracking-Varianten, die folgend präsentiert werden.

Markerbasiertes - Tracking

Diese Art des optischen Trackings verwendet ein bestimmtes 3D-Punkt-Gebilde oder 2D/ 3D-Muster von Markierungen, die auf einem Objekt platziert werden und von einer Kamera erfasst werden. Zur besseren Unterscheidung werden in dieser Arbeit die 3D-Punkt-Gebilde als „*Punktgebilde*“ und die 2D/ 3D-Muster als „*Muster*“ benannt und erläutert. Marker können zwischen den Varianten *passiv* und *aktiv* unterschieden werden (mehr dazu im nächsten Abschnitt). Auch können die Verfahren darin unterschieden werden, ob Kameras von „außen“ auf das Objekt gerichtet sind (Outside-In), oder ob die Kameras mit dem zu überwachenden Objekt verbunden sind und die Umgebung aufnehmen (Inside-Out) [Dör13, S. 104 - 105] (siehe Abb. 11).

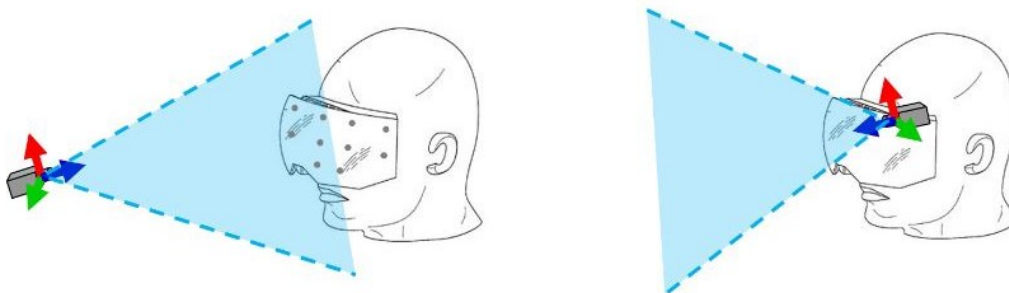


Abb. 11: Schema des Outside-In- Tracking (L.) und Inside-Out Tracking (R.) [Pag16, S.2].

Die Wahl einer markerbasierten Lösung hängt von verschiedenen Variablen ab, z. B. dem Abstand zwischen den Markierungen und der Kamera, dem Oberflächentyp des Objekts, der visuellen Umgebung (Farbvarianz, Lichtverhältnisse) und der erforderliche Ausrichtung der Kameras.

Markerbasiertes Tracking mit „Punktgebilden“

Eine oder mehrere Kameras verfolgen diese Gebilde z.B. in Form von Marker-Bäumen (siehe Abb. 12). Mittels Abgleichs des erkannten Musters mit einem im System hinterlegten Musters berechnet ein Algorithmus die Position und Ausrichtung des verfolgten Objekts. Die Anzahl, Position und Anordnung der Marker müssen dabei so ausgewählt werden, dass dem System so viele Referenz-Informationen wie möglich geliefert werden, damit den Algorithmen keine Daten zur Berechnung fehlen. Eine Art von passiven Markern reflektieren Infrarotlicht (IR) zur Lichtquelle (Abb. 12 links). In diesem Fall liefert die Kamera das IR-Signal, das von den Markierungen zur Erkennung reflektiert wird. Eine andere Möglichkeit sind markante, jeweils einfarbige Markierungen, die von einer Videokamera erfasst werden (Abb. 12 rechts).

Aktive Marker sind als Beispiel IR-Lichter, die von den Kameras erkannt werden. Das VR-HMD von Facebook Oculus Rift verwendet ein solches Outside-In-Tracking. Die aktuellere Oculus Quest Reihe hingegen setzt dieses aktive Verfahren als Inside-Out-Lösung ein, um z.B. die zu bedienenden Controller zu Orten.

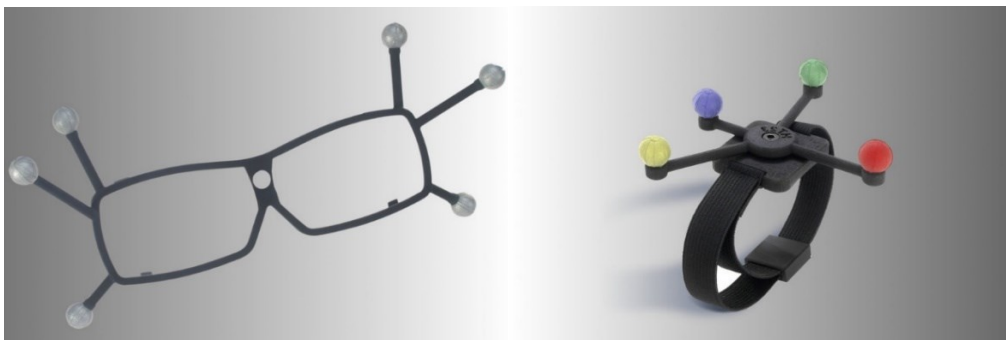


Abb. 12: Beispiel für xR-Tracking „Punktgebilde“ in Form von passiven Markerbäumen [ART20].

Markerbasiertes Tracking mit „Mustern“

Visuelle Marker, die in Form von Mustern angebracht sind, werden auch beim optischen Tracking verwendet. Die Kamera erkennt die im System hinterlegten Muster und dessen Positionen, die zu einer Bestimmung der Ortslage und Ausrichtung des Objekts führen. Die Muster müssen dabei einmalig und klar zu erkennen sein (Abb. 13).

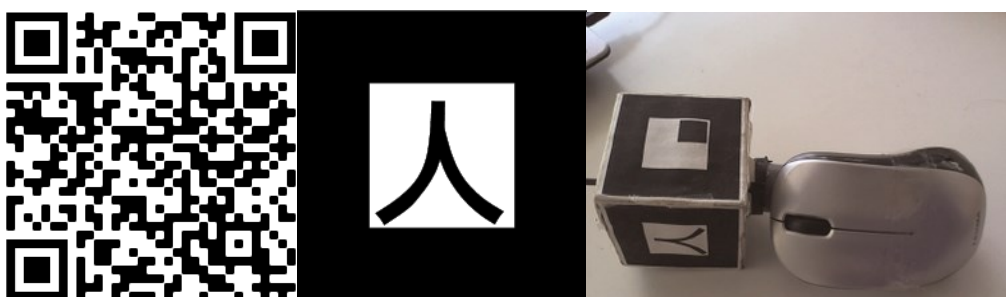


Abb. 13: Passive Marker als „Muster“. Ein QR-Code (L.), Grafik (M.) & Muster als 3D Würfel (R.) [Pai14, S.7]

Markerloses Tracking

Objekte können auch ohne Markierungen/ Marker verfolgt werden. Die Technologie kombiniert dabei Software, mit Kameras aufgenommene Videografiken und eventuelle Sensordaten, wie aus einem Gyroskop, Beschleunigungsmesser, Tiefensensoren und/oder Ortungsdiensten (z.B. GPS) eines Geräts.

Unter Einbeziehung von Verfahren, wie SLAM (engl. Simultaneous Localization And Mapping technology), zu Deutsch, *simultane Lokalisierungs- und Kartierungstechnologie*, verbesserte sich die Genauigkeit der markerlosen Bildanalyse. Ursprünglich aus der Robotik stammend, scannt bei dem Verfahren das Gerät die Umgebung und erstellt virtuelle Karten. Dieses Konzept wird nicht nur in der Robotik, sondern mittlerweile auch für Bediengeräte, wie der HoloLens 2, eingesetzt. Hier wird die Kartographie bzw. räumliche Abbildung als „*Spatial Mapping*“ bezeichnet (siehe Abb. 14).

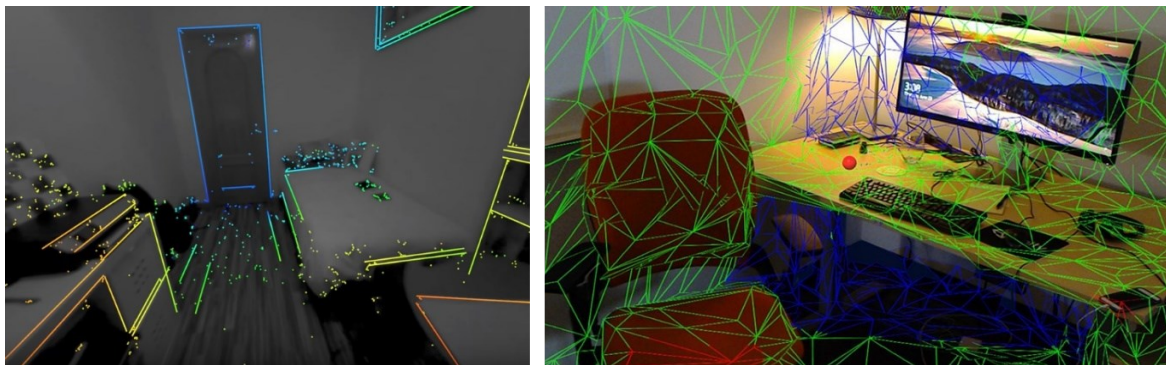


Abb. 14: SLAM Visualisierung mit der OpenCV library (L.) [Alm19]. Spatial Mapping der HoloLens (R.) [Uni21].

Für zu beobachtende Objekte, die nicht mit Sensoren ausgestattet sind, werden Position und Ausrichtung allein durch das optische Registrieren von Merkmalen, wie Ecken, Kanten, Formen oder Farbübergänge, ohne vorherige Kenntnis der Umgebung berechnet [Dör13, S.107]. Allgemein fällt dieser Themenbereich unter den Begriff „*Computer Vision*“. Erweiternd geht es hier nicht nur um das Positionstracking, sondern auch um das (Wieder-) Erkennen von einzelnen Objekten oder ganzen Umgebungen (siehe Abb. 15 als Beispiel).

Je nach Komplexität kann unter Einbeziehung von künstlicher Intelligenz (KI) und externen Rechenzentren ein System aufgesetzt werden, welches reale Objekte erkennt und dessen virtuell hinterlegten 3D-Modelle in Echtzeit vereint darstellt. Da die Kosten für die Computer- und Kameralistung stetig sinken und immer leistungsstärker werden, kann das optische Tracking ein zunehmender Standard zur Positionserfassung werden. Für diese Arbeit wurde jedoch die markerbasierte Variante unter Verwendung folgender Stereokamera verwendet, um stabile, zuverlässige und präzise Daten zu erhalten.



Abb. 15: Markerlose Objekterkennung mittels des VisionLib SDK's [Vis18].

SmartTrack 3

Das Tracking-System SmartTrack 3 von der Firma Advanced Realtime Tracking GmbH & Co. KG (ART) ist ein eigenständiges, optisches Infrarot-System, welches für den Einsatz in begrenzten Räumen (ca. 2 m³) ausgelegt ist. Zwei Kameras senden synchronisierte IR-Blitze aus, die von retroreflektierenden Punktgebilden (Marker-Bäumen) reflektiert werden. Basierend auf der empfangenen IR-Strahlung wird ein Graustufenbild erstellt. Im nächsten Schritt werden diese Bilder von den Kameras vorverarbeitet, um die Markierungspositionen (2DOF) auf dem Kamerasensor unter Verwendung von Mustererkennungsalgorithmen zu berechnen. Diese 2DOF-Daten werden dann an eine Steuereinheit gesendet, um 3DOF- oder 6DOF-Informationen zu generieren. Für die 6DOF-Verfolgung ist daher ein eindeutiger Marker-Baum, bestehend aus mehreren unabhängigen Punkten essentiell. Diese Punktgebilde müssen starr sein, um diese anhand ihrer relativen Markerpositionen erkennen zu können. Abb. 16 zeigt das Konzept der optischen Infrarotverfolgung. Die Grundlage für die Berechnung der Positionsverfolgung ist, dass sich das Sichtfeld beider Kameras überlappt. Die Wege der optischen Strahlen von den Kameras zu den Markern wird berechnet, um die Strahlschnittpunkte in dreidimensionale Koordinaten übertragen zu können. Diese Schnittpunkte sind die Positionen der Markierungen. Auf dieser Grundlage können 6DOF-Daten (6d-Array) berechnet und schließlich die Position und Ausrichtung des Ziels und damit des zu verfolgenden Objekts oder der zu verfolgenden Person ermittelt werden. Die Daten werden daraufhin an den Computer gesendet, der diese Informationen zur weiteren Verarbeitung benötigt [ART19, S.17].

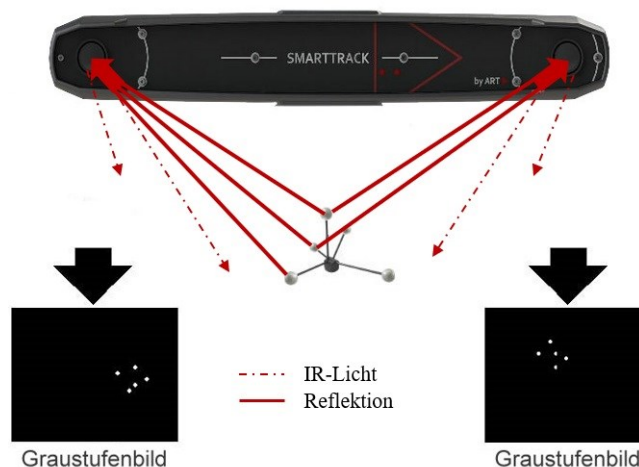


Abb. 16: Das Konzept des optischen Trackings mit Hilfe einer Stereo-IR-Kamera

Dies ist eine Möglichkeit zur Positionserfassung, die für dieses Projekt verwendet wurde. Weitere Methoden von anderen Projekten, die als Inspiration oder zum Vergleich analysiert wurden, werden im folgenden Abschnitt zusätzlich erwähnt.

2.4 Stand der Technik

Das Thema der „Visualisierung lokalisierter Ultraschalldaten am inspizierten Prüfteil“ ist parallel mit den bereits erwähnten bildgebenden Methoden im stetigen Wandel und somit keine Neuheit. Die Idee einer generellen Einbeziehung von xR-Technologien für eine vereinfachte und verständliche Darstellung von lokalisierten Ultraschalldaten ist zwar mittlerweile auch keine Neuheit mehr, jedoch wird diesbezüglich immer noch an generellen Möglichkeiten geforscht und ein etablierter Standard ist noch nicht zu erkennen, da im xR-Kontext allgemein ebenfalls noch keine langjährigen Hardware und Software-Standards vorhanden sind.

Im Jahr 2015 patentierte Olympus den Prozess einer verbesserten Methode der Inspektion mit AR HMD's, welche einen vordefinierten Weg vorgeben, an dem der Prüfkopf entlanggeführt wurde. Zwar konnte eine Abweichung vom vorgegebenen Pfad festgestellt werden, allerdings fehlte z.B. die Funktion einer Speicherung der Daten [Lan15].

Viele der recherchierten und relevanten Möglichkeiten der Thematik lassen sich jedoch mit dem Patent [Hol17] (Abb. 17) aus dem Jahr 2017 grundlegend beschreiben:

„Darstellungseinrichtung (1) zur Sichtbarmachung von Unregelmäßigkeiten (2) in Faserverbundwerkstoffen, aufweisend mindestens ein Bauteil (3), mindestens eine Messeinrichtung (4), mindestens eine Recheneinheit (5), mindestens eine Ausgabeeinrichtung (6), dadurch gekennzeichnet, dass durch die Ausgabeeinrichtung eine Überlagerung eines Bildes des Bauteils (3) mit Daten der Recheneinheit (5) wiedergebbar ist.“

Die Darstellungseinrichtung (1) wird beispielsweise mit einem Ultraschallgerät (4), einem Computer (5) und einer „Brille“ für die Ausgabe in AR (6a) aufgebaut. Die nutzende Person erhält dabei ein von dem Gerät (6a) erfasstes Kamerabild des Bauteils (3) in die Brille. Der Computer (5) überlagert dieses Bild mit den aufbereiteten Daten des Ultraschallgeräts. Die Referenz zwischen Bauteil (3) und Messergebnissen (2) erfolgt hierbei über optische Marker [Hol17, S.2].

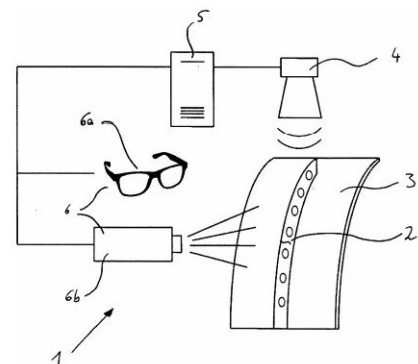


Abb. 17: Patentdarstellung [Hol17, S.1]

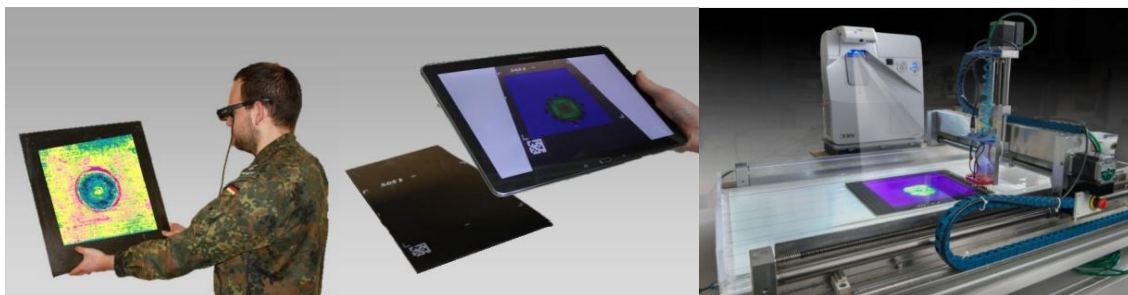


Abb. 18: Konzeptionelle Visualisierung von C-Bildern einer CFK Struktur in verschiedenen AR-Umgebungen zur Vorarbeit des Patents „Hol17“ [Mey15, S.].

Wie [Dep18] auf Abb. 19 zeigt können z.B. aber auch Metalteile geprüft werden. In dem konkreten Beispiel wird jedoch das VR-HMD Vive Pro und dessen externe IR-Kameras (Basisstation/ Lighthouse) sowie der Vive-Tracker verwendet, um das HMD und den US-Sensor zu verfolgen. Durch den Einsatz einer zusätzlichen Stereokamera (ZEDmini von Stereolabs Inc.), die am HMD befestigt wurde, wird die Vive Pro in ein AR fähiges Gerät umfunktioniert. Die Dimension der Prüffläche wird durch ein vorgeschaltetes Setup definiert, bei dem der Ultraschallsensor, an dem der Vive-Tracker befestigt wurde, an feste Positionen am Prüfteil geführt und diese Koordinaten in der Anwendung gespeichert werden. Mit dem Patent [Lan15] und dem Projekt des Papers [Dep18] lässt sich jedoch nur der zurückgelegte oder noch zu beschreitende Weg anzeigen. Die eigentlichen Messergebnisse werden nicht auf den Geräten angezeigt. Einen Ansatz für die Darstellung von Messdaten eines Prüfteils zeigt das Paper [Sch_2] Abb. 20 (U.): (L.). Hier wird ein Beispiel aus dem Bauwesen gezeigt, bei dem vorausgenommene Messdaten nachträglich in ein 3D-Modell umgewandelt wurden, welches mit Hilfe der HoloLens 1 und eines QR-Markers deckungsgleich in ein Prüfteil projiziert wurden. Jedoch ist das verwendete Verfahren der Messdaten-Umwandlung in ein 3D-Modell mit Hilfe der sogenannten SAFT-Rekonstruktion (Synthetic Aperture Focusing Technique) rechenintensiv und bisher nicht in Echtzeit möglich, da zugleich ein gewisser Datensatz zur Ausführung vornherein vorliegen muss. Einen anderen Ansatz für eine 3D-Visualisierung verfolgte [Las14] (Abb. 20 U. R.) bereits 2014, bei dem jedoch damals die Darstellung nicht auf einem xR-Gerät erfolgte, sondern an der externen Recheneinheit mit klassischem Desktopbildschirm.

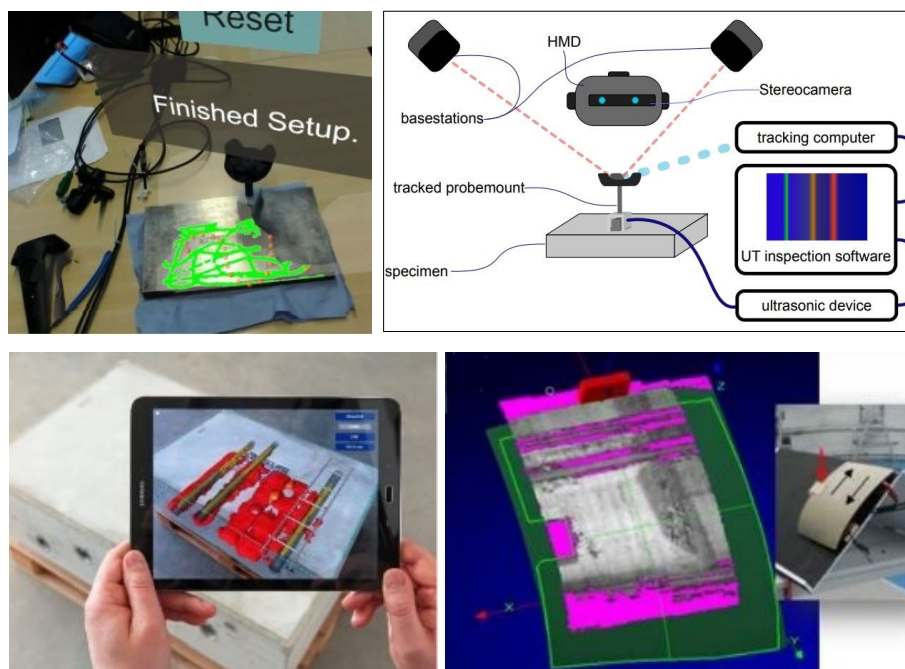


Abb. 19 (O.): Setup der AR-Applikation (L.) und Schematischer Aufbau (R.) [Dep18, S.4, S.5].

Abb. 20 (U.): (L.) AR Anwendung von [Sch_2, S.5]. (R.) Visualisierung von [Las14, S.6]

Das Fraunhofer-Institut benennt Anforderungen an ihr Produkt „3D-SmartInspect“, die nach Produktblatt [Fr221] und Produktpräsentation folgend lauten:

- Eine digitale Erfassung des manuellen Prüfprozesses über Kameras.
- Die Verfolgung der Prüfkopfposition.
- Eine automatische Auswertung der Messsignale sowie die automatische Erstellung eines digitalen Prüfprotokolls.

Als mögliche Darstellungsmedien werden neben einem Notebook auch AR-fähige Geräte, wie die HoloLens 1, genannt.

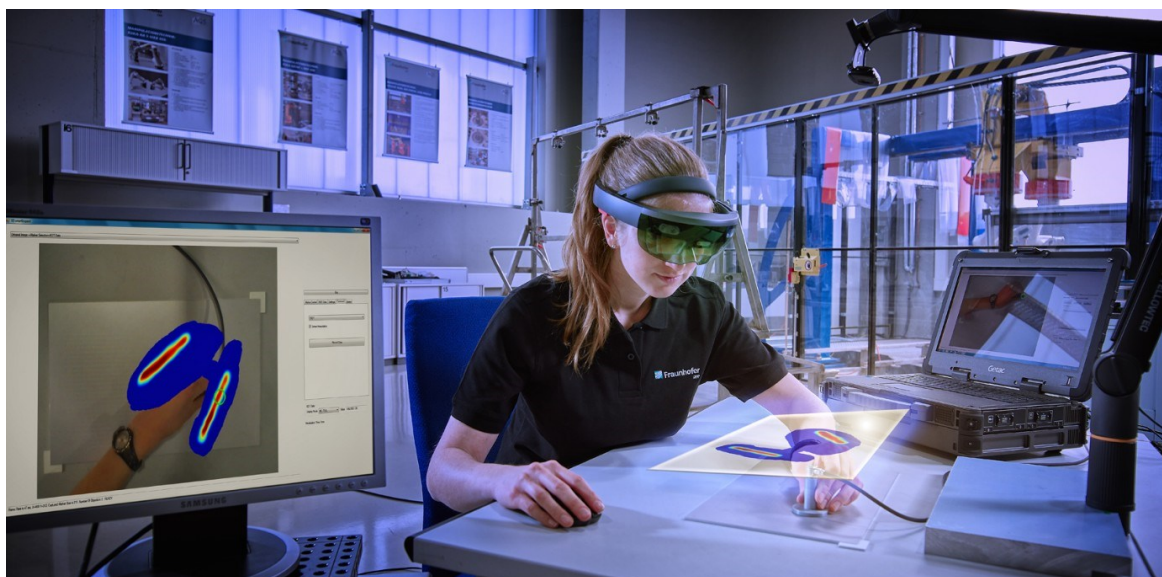


Abb. 21: Systemaufbau 3D-SmartInspect [Fra21]

3. Hardwareseitige Systembeschreibung

Da nun alle benötigten Grundlagen behandelt wurden wird in diesem Abschnitt auf den Planungsprozess der in diesem Projekt eingesetzten Hardware und Software zur Erweiterung des Systemstands anhand einer morphologischen Analyse erläutert. Darauffolgend werden die aus der Analyse entstandenen Ergänzungen zu dem vorangestelltem Hardwareaufbau aufgeführt.

3.1 Planungsprozess

Die Art der morphologischen Analyse wird auf Fritz Zwicky und dem Jahr 1967 zurückgeführt und *„ist ein analytisches Verfahren zur Strukturierung eines Problems mit dem Ziel, systematisch und unvoreingenommen Lösungskandidaten zu generieren.“* [Zec19, S. 2]. Es wird allgemein ein dreistufiger Prozess beschrieben, bei dem zu Beginn eine Ausgangsanforderung definiert und daraufhin in einzelne Teile (Merkmale) gegliedert wird. Im zweiten Schritt werden für jedes Merkmal unabhängige und isolierte Lösungsvorschläge (Ausprägungen) entwickelt. Das Resultat hieraus wird in einem *„morphologischen Kasten“* protokolliert. Im Letzen Schritt werden die einzelnen Ausprägungen zu den Merkmalen jeweils miteinander kombiniert, um daraufhin mehrere Lösungswege/ Strategien für die Ausgangsanforderungen aufzeigen zu können. Mit Blick auf den Abschnitt 1.1 ist die Ausgangsanforderung für dieses Projekt wie folgt formuliert:

„Eine mobile, zerstörungsfreie Werkstoffprüfung soll an einem Bauteil durchgeführt werden, bei der die erhaltenen Daten visuell aufbereitet werden und diese für die nutzende Person mobil aufrufbar sind (vorzugsweise in AR/ HL2). Die (visuellen) Daten sollen zusätzlich Orts-/ Bauteilbezogen gespeichert/ geladen werden können.“

Im zweiten Schritt wird die Ausgangsanforderung in insgesamt elf Merkmale unterteilt, die zusammennaddiert 51 Ausprägungen aufweisen und im morphologischen Kasten protokolliert sind. Hieraus werden unter heuristischer Schlussfolgerung drei Lösungswege (siehe Abb. 22) definiert. Lösungsweg „A“ beschreibt dabei eine Strategie, bei der das AR-Gerät die geringste Rechenlast trägt, da alle Daten zur Visualisierung über externe Recheneinheiten bereitgestellt werden und das Tracking von Ultraschallsensor, Prüfteil und der benutzenden Person ebenfalls über externe Einheiten erfolgt. Lösungsweg „B“ hingegen stellt eine Strategie dar, bei der möglichst alle Daten und Berechnungen zur Darstellung über das AR-Gerät verarbeitet werden. Das Tracking erfolgt zudem ausschließlich über das AR-Gerät. Weg „C“ stellt eine Strategie dar, bei der kein AR-Gerät involviert wird. Dabei ist vorweg zu nehmen, dass der Lösungsweg „A“ zwar als Ideal präferiert wurde aufgrund der COVID-19 Pandemie und der einhergehenden Regulierungen sowie Restriktionen jedoch ein vierter Lösungsweg definiert werden musste, der sich zwischen Strategie „A“ und „B“ einreicht. Er wird „B²“ betitelt und ist Ziel dieser Arbeit.

Morphologischer Kasten	Abteilung MO-WRT		
	Blatt 2		

Eine mobile, zerstörungsfreie Werkstoffprüfung soll an einem Bauteil durchgeführt werden, bei der die erhaltenen Daten visuell aufbereitet werden und diese für die nutzende Person mobil aufrufbar sind (vorzugsweise in AR/ HL2). Die (visuellen) Daten sollen zusätzlich Orts-/ Bauteilbezogen gespeichert/ geladen werden können."

Merkmal	1	2	3	4	5	6	7
Ausprägung Lösungsmöglichkeiten (bei allen Lösungen wiederholt auftauchende Merkmale)	B B² A C						
1 Welche zerstörungsfreie Prüfverfahren können in Betracht gezogen werden?	Ultraschallprüfung	Wirbelstromprüfung (Dicke, nur Metalle)	Durchstrahlprüfung	Akustische Resonanzanalyse	Mikrowellenprüfung	Thermografie	
2 Auf welche Art wird der benötigte Prüfsensor geführt?	Manuell, händisch	Automatisiert, x,y,z Linearachsen	Automatisiert, Roboterarm	Keine, statischer Prüfsensor	Manuell, 3D Messarm		
3 Rohdatenverarbeitung statt?	Kombinierte Lösung	Vollständig auf Computer/ Laptop	Vollständig auf dem Prüfsystem				
4 Zu welchem Koordinatensystem werden die NDT-Daten referenziert?	Prüfteil-eigenes Koordinatensystem	Koordinatensystem eines-externen Markers					
5 Mit welcher Technologie erfolgt die Positionserfassung des US-Sensorkopfes?	(AR-Gerät-Intern) Optisch- Marker	(AR-Gerät-Extern) Optisch- Marker	Kein Tracking	(AR-Gerät-Intern) Optisch- Markerless	(AR-Gerät-Extern) IMU	(AR-Gerät-Extern) Indoor Positioning Systems (IPS)	
6 Mit welcher Technologie erfolgt die Positionserfassung des Bauteils?	(AR-Gerät-Intern) Optisch- Marker	(AR-Gerät-Extern) Optisch- Marker	Kein Tracking	(AR-Gerät-Intern) Optisch- Markerless	(AR-Gerät-Extern) IMU	(AR-Gerät-Extern) Indoor Positioning Systems (IPS)	
7 Wie erfolgt die Positionierung des Darstellungs-Geräts zum Prüfteil?	(AR-Gerät-Intern) Optisch- Marker	(AR-Gerät-Extern) Optisch- Marker	Kein Tracking	(AR-Gerät-Intern) Optisch- Markerless	(AR-Gerät-Extern) IMU	(AR-Gerät-Extern) Indoor Positioning Systems (IPS)	
8 Wo findet die Datenaufbereitung zur Visualisierung statt?	Kombinierte Lösung	Vollständig auf Computer/ Laptop	Vollständig auf AR-Gerät				
9 Über welches Gerät wird die visuelle Darstellung ermöglicht?	AR Microsoft HoloLens	Non-AR Windows PC,Laptop	AR Windows PC/Tablet/Phone	AR Android Tablet/Phone	AR Apple Tablet / Phone	Apple Laptop	Unix System
10 Welche Methode der Datenübertragung zum Darstellungsgerät wird genutzt?	Geräte Extern Wi-Fi	Geräte Intern Nur Leiterbahnen	Geräte Extern USB	Geräte Extern Bluetooth	Geräte Extern LAN		
11 Welche Art von Speichermedium wird verwendet?	Geräte Internes Speichermedium	Geräte Externes Netzwerk Speicher					



Abb.22: Morphologischer Kasten zur Planung der eingesetzten Hardwarestrategie mit zusätzlicher Farbkennzeichnung der vier erarbeiteten Lösungswege.

Ergänzende Aspekte zu den aufgeführten Merkmalen

Bevor die einzelnen Strategien weiter erläutert werden sind deren Gemeinsamkeiten zu nennen. Die elf Merkmale in Abb. 22 sind als Interrogativsätze formuliert und lauten wie folgt (siehe Tab. 3.1):

<i>Merkm. Nr.</i>	<i>Interrogativsatz</i>
1	Welche zerstörungsfreie Prüfverfahren können in Betracht gezogen werden?
2	Auf welche Art wird der benötigte Prüfsensor geführt?
3	Wo findet die Rohdatenverarbeitung statt?
4	Zu welchem Koordinatensystem werden die NDT-Daten referenziert?
5	Mit welcher Technologie erfolgt die Positionserfassung des US-Sensorkopfes?
6	Mit welcher Technologie erfolgt die Positionserfassung des Bauteils?
7	Wie erfolgt die Positionierung des Darstellungs-Geräts zum Prüfteil?
8	Wo findet die Datenaufbereitung zur Visualisierung statt?
9	Über welches Gerät wird die visuelle Darstellung ermöglicht?
10	Welche Methode der Datenübertragung zum Darstellungsgerät wird genutzt?
11	Welche Art von Speichermedium wird verwendet?

Tab. 3.1: Die elf Interrogativsätze dieser Thesis zur Erfüllung der Ausgangsforderung

Die möglichen Antworten zu den Merkmalen sind in den jeweiligen Ausprägungen zu finden.

Es sei erwähnt, dass eine „unabhängige und isolierte“ Betrachtung der Ausprägungen zu den Merkmalen „1“ bis „4“ nicht möglich ist, da für dieses Projekt bereits zu Beginn an definierte Hardware- und Softwarekomponenten vorlagen. Zur Übersicht werden Zahlenkombination, zusammengesetzt aus $\{\text{Merkm. Ausprägung}\}$, bei den Lösungswegen angegeben, die für ein besseres Verständnis in Abb. 22: Morphologischer Kasten eingesehen werden können.

Es wird das Ultraschallverfahren für die ZfP verwendet {1.1} und der entsprechende Prüfkopf *SE 10/6 PB 4 C* dabei händisch geführt {2.1} Über den US-Receiver *PCUS Pro Single* erhält der angeschlossene Windows Laptop mit der Systementwicklungssoftware *LabVIEW* die gefilterten „Rohdaten“ vom Prüfkopf, die hier zudem weiterverarbeitet werden. Dies stellt ein kombiniertes Prüfsystem aus US-Prüfkopf, PCUS und Laptop dar {3.1}. Pfad {4.1} setzt ein Markerless-Tracking voraus, welches für diese Thesis aufgrund des steigenden Arbeitsumfangs nicht weiter betrachtet wird. Stattdessen kann auf die bestehende externe IR-Kamera *SmartTrack 3* zurückgegriffen werden, weshalb das Tracking über externe optische Marker erfolgt {4.2}. Für die Merkmale „5“ bis „7“ wurde ursprünglich jeweils eine einheitliche Trackinglösung für die drei Komponenten (US-Prüfkopf, Part-on-Carrier und AR-Device) angedacht, um eine „Daten- und Koordinaten-Homogenität“ zu gewährleisten, damit das Darstellungsgerät mit einheitlichen Koordinaten arbeitet ohne zusätzliche Transformationen berechnen zu müssen. Als AR-Device ist zudem die HoloLens 2 vorgeschrieben {9.1}

Heuristische Einschätzung zur Strategie „A“

Für den Lösungsweg „A“ werden Marker-Bäume verwendet, um die Positionen des US-Prüfkopfs, des Part-on-Carriers und der HL2 festzustellen ({4.2}, {5.2}, {6.2} und {7.2}). Dadurch wird vermutet, dass mit dieser Kombination die genauesten und stabilsten Positionsdaten der drei Komponenten verarbeitet werden. Die Marker-Bäume stellen jedoch eine zusätzliche Komponente dar, die Beschädigt werden können. Die Positionsdaten im Zusammenhang mit den Messdaten des US-Sensors bilden zudem einen großen Datenstrom, der zeitkritisch von einem Computer erfasst, gefiltert, verarbeitet {8.2} und an die HL2 {9.1} via Wi-Fi {10.2} gesendet und dort weiterverarbeitet werden muss, was zu Latenzen und Datenverlust führen kann.

Die Daten sollen von einem Netzwerkspeicher aus geräteextern gespeichert und geladen werden {11.2}. Hierfür müssen entsprechende Zugriffsrechte vorliegen, was bei betriebsinternen Netzen mit mehrschichtigen Sicherheitsbestimmungen einen Mehraufwand bedeutet.

Heuristische Einschätzung zur Strategie „B“

Für den Lösungsweg „B“ erfolgt das Tracking ausschließlich intern über die HoloLens 2 und wird mit Image-Targets {4.2} optisch erzielt, die jeweils am US-Sensor {5.1} und Part-on-Carrier {6.1} befestigt sind. Diese müssen jedoch ebenfalls auf eine Weise angebracht werden, sodass diese bei einer Messdurchführung die prüfende Person nicht behindern und das Prüfteil verdecken. Die Stabilität und Genauigkeit des Trackings ist ausschließlich Arbeit der HL2 {7.1} und somit eventuell nicht permanent gegeben. Hierzu wurden jedoch keine wissenschaftlichen Auswertungen gefunden und eine eigene Belegung ist für diese Behauptung aufgrund des steigenden Arbeitsumfangs vorerst nicht möglich. Der einfallende Datenstrom beschränkt sich im Gegensatz zu Lösung „A“ auf die Messdaten des Ultraschallsensors {8.1}, die ebenfalls über Wi-Fi {10.1} vom Messcomputer an die HoloLens 2 {9.1} zur Darstellung übertragen werden. Die protokollierten Messergebnisse werden intern auf das AR-Gerät gespeichert {11.1} und insgesamt wird eine größere Leistungsanforderung an das AR-Gerät erwartet.

Heuristische Einschätzung zur Strategie „C“

Lösungsweg „C“ stellt einen erweiternden Zusatz dar und kann nicht als einzelner Lösungsansatz betrachtet werden, da eine Darstellung bauteilentfernt an einen Desktop / Laptop stattfindet {9.2}, dessen Position nicht festgestellt und verfolgt wird {7.4}. Der Messcomputer selbst wird zum Darstellungsgerät, weshalb keine externe Kommunikation für die Darstellung erfolgt {10.2} Diese Strategie ähnelt dennoch Lösung „A“ in 8 Ausprägungen.

Definition und Auswahl der Strategie „B²“

Da mit der Umsetzung der Strategie „B“ ein hoher Leistungsaufwand an die HL2 gestellt wird (Positionsverfolgung von zwei Objekten sowie die Verarbeitung und Darstellung der US-Sensordaten) wurde der Lösungsweg „A“ mit seinem erwartetem stabilen und genauen Tracking anfänglich bevorzugt und angestrebt. Mit den verschärften Covid-19 Pandemieregulierungen musste jedoch ab dem Monat Dezember im Jahr 2020 eine Alternative definiert werden, da die Räumlichkeiten und nötige Hardware nicht mehr flexibel zur Verfügung stand und daneben überwiegend mit protokollierten Daten anstelle von Echtzeitaufnahmen gearbeitet werden musste. Die Ausprägungen zu den Merkmalen „6“, „7“, „8“ und „11“ wurden dabei aus der Strategie „B“ übernommen, bzw. werden bei Merkmal „6“ und „7“ hybride Umsetzungen angestrebt. Dies bedeutet konkret folgende Strategie:

Die Zuordnung der Koordinatensysteme erfolgt weiterhin über externe Marker {4.2} in Form von Marker-Bäumen, angebracht an US-Sensor {5.2}, Part-on-Carrier {6.2} und HL2 {7.2}, die über die IR-Kamera verfolgt werden.

Zusätzlich ist der Einsatz eines QR-Codes, der am Part-on-Carrier {6.1} befestigt ist, möglich, der alternativ über die HL2 verfolgt wird {7.1}. Die US-Mess- sowie Koordinatendaten werden soweit vorbereitet und über Wi-Fi versendet {10.1}, sodass diese von der HL2 weiterverarbeitet werden können {8.1}. Die protokollierten Daten werden dabei auf das interne Speichermedium der HL2 gesichert und daraus geladen {11.1}.

Mit der Bestrebung zur Umsetzung dieser Lösungsstrategie ergibt sich ein erweiterter Systemaufbau.

3.2 Erweiterter Systemaufbau

Der vorangestellte Systemaufbau aus Abschnitt 1.3 Abb. 1 wurde im Sinne der Strategie „B²“ für diese Arbeit um drei Hardwarekomponenten ergänzt (siehe Abb. 23). Die *HoloLens 2* ist zu nennen, an der die in *Laptop-A* zusammengefassten US- und Positionsdaten weitergeleitet werden. Damit dies geschehen kann wurde der Ethernet-Switch durch einen *Wireless-Router* ersetzt und ein Netzwerkanschluss aus HL2, Laptop-A und Laptop-B definiert.

Da zur Hauptphase des Projektes im Homeoffice mit protokollierten Datenaufnahmen und ohne Einsatz der IR-Kamera gearbeitet wurde, konnte die HL2 erst im Nachgang mit einem Marker-Baum ausgestattet werden. Stattdessen wurde während des Zeitraums ein QR-Code generiert, um einen Bezugspunkt zwischen der HL2-Applikation und den Daten herzustellen. Später wurde auch für den QR-Code eine entsprechende Halterung am *Part-on-Carrier* konzipiert.

Somit sind zwei verschiedene Ansätze entstanden, um eine Deckungsgleichheit zwischen den realen Komponenten und deren virtuellen Repräsentanten umzusetzen:

- 1. Bei dem Einsatz der Variante mit einem QR-Marker wird das HL2-interne Tracking verwendet, um die Visualisierung relativ zum Marker zu positionieren. Gleichzeitig werden der Part-on-Carrier und der US-Sensor durch das externe Tracking erfasst, um die benötigten Positionsdaten für die Visualisierung zu erhalten, die auf das QR-System referenziert werden (mehr dazu in Abschnitt 4.4.2).
- 2. Unter Verwendung der Variante, die ausschließlich über das externe Tracking betrieben wird, erfolgt eine Referenz der Visualisierung über den Part-on-Carrier (mehr dazu in Abschnitt 4.3).

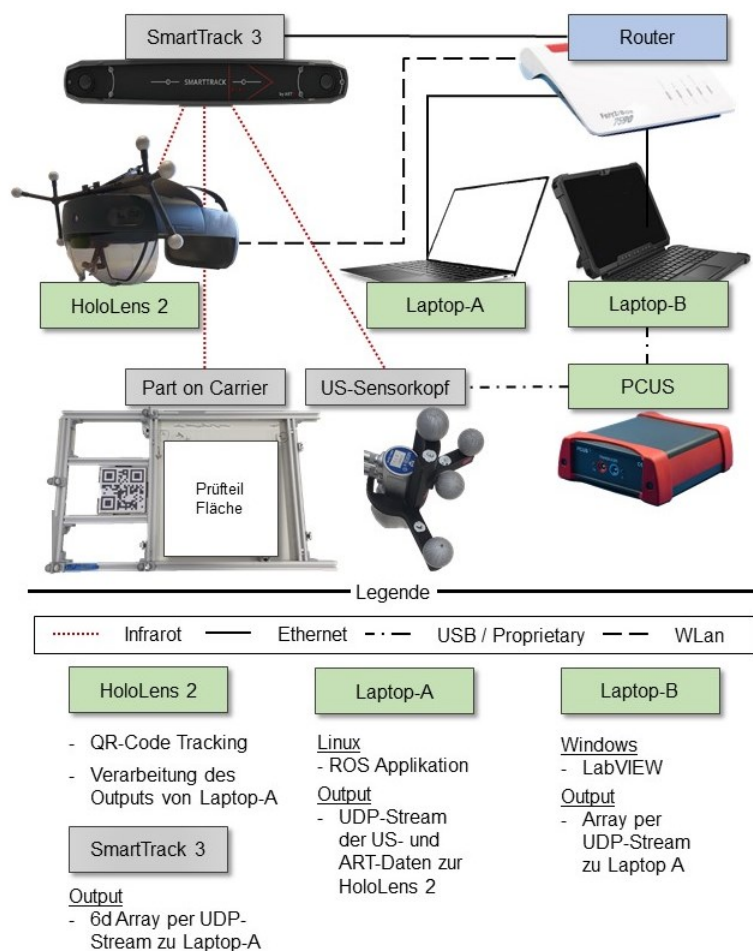


Abb. 23: Schematische Darstellung des erweiterten Systemaufbaus mit Realbildern ohne einheitlichen Maßstab

4. Softwareseitige Systembeschreibung

Für ein besseres Verständnis der einzelnen Hardwarekomponenten und dessen Funktionsweisen, erfolgt in diesem Abschnitt die softwareseitige Systembeschreibung. Dabei wird das komplette System mit Fokus auf die drei Hauptsoftwarebestandteile *LabVIEW*, *ROS I* und die *HL2 Applikation* beschrieben. Da der Kern dieser Arbeit die *HL2 Applikation* ist und die anderen beiden Komponenten aus dem vorangestellten Systemaufbau entstanden und an dessen Funktionen angeknüpft wurde, werden die *LabVIEW* und *ROS I* Anwendung jeweils nur im Rahmen eines nötigen Verständnisses erläutert, während bei der *HL2 Applikation* zusätzlich auf die benötigte Entwicklungsumgebung und die einzelnen Schritte der Benutzung eingegangen wird.

4.1 Übersicht

In Abb. 24 sind die Verknüpfungen der Grundfunktionen der Softwareelemente schematisch hinterlegt. Der US-Sensorkopf ist in diesem Aufbau ein passives Element, welches keine eigene Softwarestruktur besitzt. Mit Hilfe des Signalwandlers *PCUS* wird der US-Sensorkopf mit Strom versorgt, wodurch dessen verbaute Sende- und Empfangseinheiten funktionieren können. Zudem wandelt der *PCUS* die empfangenen analogen Impulse vom US-Sensor in digitale Signale um, die über den USB-Anschluss an *Laptop-B* gesendet werden. Die entsprechenden *Treiber* sorgen hier dafür, dass die digitalen *PCUS*-Signale vom Windows-System verstanden werden und andererseits die vorgenommenen Einstellungen auf das *PCUS* angewendet werden. Die Systementwicklungsumgebung *LabVIEW* greift dabei auf die *PCUS* Treiber zu und erhält die digitalisierten Ultraschalldaten und verarbeitet diese intern je nach eingestellten Datenfiltern und Verstärkern weiter. Nähere Informationen hierzu sind im Abschnitt 4.2 zu finden. Ein in der Programmiersprache Python3 geschriebenes UDP-Skript erhält die gefilterten Daten von *LabVIEW*, verpackt diese in ein Daten-Array und sendet diese als Bitstream über den Router zu *Laptop-A*.

Zur gleichen Zeit nimmt die IR-Kamera *SmartTrack 3* die reflektierten Strahlen von den drei Einheiten *HoloLens 2*, *Part-on-Carrier* und *US-Sensorkopf* auf und sendet die Informationen über den Router zu *Laptop-A*. Hier werden diese Daten von der Software *DTrack* und dessen *Treiber* aufbereitet. Näheres hierzu wurde bereits im Optische Trackingsysteme für xR-Technologien erläutert. Andererseits kann die IR-Kamera auch hierrüber eingestellt und das Tracking-Umfeld sowie die zu verfolgenden Objekte definiert werden.

Mit Hilfe des Open-Source Frameworks *ROS* und in Python 3 verfasste Skripte wird auf Grundlage der Positionsdaten vom US-Sensorkopf sowie vom *Part-on-Carrier* durch die IR-Kamera ein Koordinatensystem definiert. Dessen Ursprung wird verfolgend auf den *Part-on-Carrier* gesetzt, um eine annähernde, digitale Referenz zum realen Prüfteil zu schaffen. Es kann sich hierbei jedoch nur um eine imaginäre Referenz handeln, da das zu untersuchende Prüfteil jederzeit ausgetauscht werden könnte. Nur der *Carrier*

ist dem System bekannt und nicht das eigentliche Prüfteil. Weitere Informationen hierzu befinden sich im Kapitel „ROS-Applikation“. In ROS erhält der US-Sensorkopf seine umgerechneten Koordinaten. Gleichzeitig werden die Ultraschalldaten aufgegriffen und auf die nötigsten Daten, dem Header, reduziert. Die neuen Positionsdaten vom Ultraschallsensor und die Ultraschallmessdaten aus dem Header werden zusammen verknüpft und in ein Daten-Array verpackt. Die beiden Quellen sind dabei zwar nicht explizit aufeinander synchronisiert, die jeweilige Messwiederholfrequenz beider Systeme sorgt jedoch für eine Zugehörigkeit der Daten. Das verknüpfte Daten-Array kann je nach Nutzung vor dem Versenden in eine separate Text-Datei für den späteren Wiederaufruf protokolliert werden. Daraus entstanden z.B. die Datenaufnahmen, die während der Homeoffice-Regelungen genutzt wurden. Ein entsprechendes UDP-Skript wandelt das Daten-Array in einen Bitstream um und sendet dieses an die *HoloLens 2*. In der HL2 Applikation mit dem Projektnamen *NDT-US_HL2*, die für diese Arbeit programmiert wurde, kann zwischen zwei Tracking-Varianten entschieden werden:

- 1. Wird die interne Tracking-Variante ausgewählt, schaltet sich die Frontkamera der HL2 ein und analysiert den Videostream intern auf den hinterlegten QR-Code, der an dem Part-on-Carrier befestigt ist. Wird dieser gefunden und wurde in der Anwendung der Ursprungspunkt zum ROS-Koordinatensystem definiert, können die versendeten ROS-Daten homogen verarbeitet werden.
- 2. Wird die externe Tracking-Variante gewählt, verarbeitet die HL2 Applikation direkt die Daten vom SmartTrack, ignoriert die Koordinaten von ROS und verarbeitet allein den Ultraschalldaten-Header. Dieser Variante ist zum bisherigen Zeitpunkt jedoch noch nicht fehlerfrei nutzbar, weshalb die auftretenden Problematiken in Abschnitt 6 detaillierter erläutert werden.

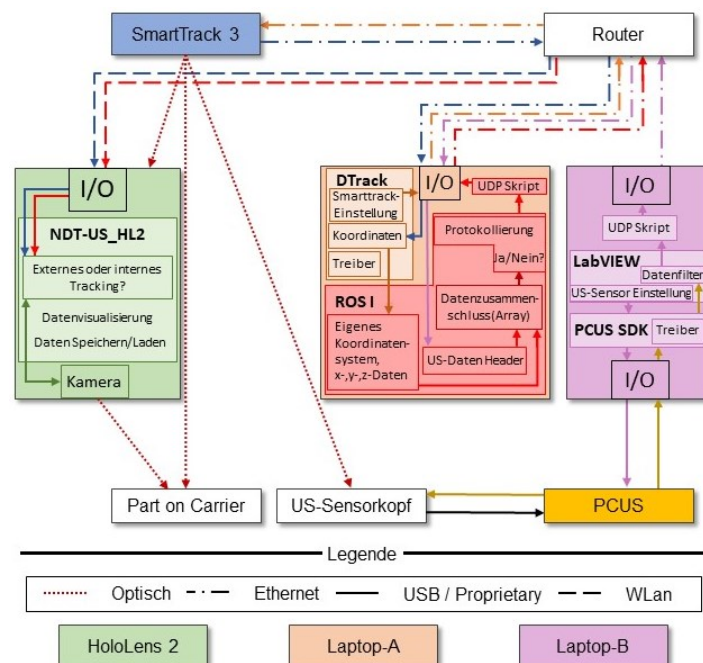


Abb. 24: Schematische Softwarestruktur im Gesamtkontext des Projekts

4.2 LabVIEW-Applikation

Mit Hilfe des grafischen Programmiersystem LabVIEW wurde beim DLR eine Applikation entwickelt, mit deren Hilfe Ultraschallprüfgeräte eingestellt und deren Rohdaten am Computer dargestellt und verarbeitet werden können. Dies ist die „erste Anlaufstelle“, an der die Ultraschalldaten für die Weitergabe an Laptop-A definiert und gefiltert werden.

Im rot-markierten Kasten in Abb. 25 sind die Einstellungen für das Ultraschallgerät PCUS zu sehen, die später ebenfalls über die HL2-Applikation angezeigt werden. Hier wird eine Vorlaufzeit, gemessen in Samples (S), definiert (Recording Delay, hier 970 S), die bereits visuell in Abb. 2 unter dem Begriff Vorlaufstrecke zu sehen war und angibt, ab wann eine Analyse der Messwerte erfolgen soll. Zudem ist die Sample-Rate in Hz einzusehen, die im späteren Verlauf für die Berechnung des Tiefenwerts benötigt wird. Im lila Kasten ist bereits das A-Bild zu sehen. Hierrüber werden die Rohdaten (schwarz) und der zu analysierende Bereich (man spricht auch von „Blende“ hier in der Farbe Blau) grafisch dargestellt. Im rosa Kasten werden der Time-of-Flight-Wert (ToF) in Samples und der maximale Amplitudenwert (max. V) in Volt aus dem blauen Bereich angezeigt.

Aus diesen Daten erstellt das UDP-Skript von Laptop-B jeweils ein Daten-Array und versendet dieses als Bitstream, welches beispielhaft wie Folgt aussieht:

```
# 1, 2, 3, 4, 5, 6, 7, ...
[[7.0, 970.0, 1e-08, 500.0, 2.0, 181.0, 0.1384],[...]]
```

4.2.1

Das Array besteht aus 2 Unterpaketen (Paket rot und Paket gelb). Im roten Paket befindet sich der US-Daten-Header. Dieser setzt sich aus folgenden sieben Werten zusammen:

- 1. Anzahl an Werten im Header Paket (HD)
- 2. Sensor-Delay (SD)
- 3. Zeitinkrement als Kehrwert der Sample-Rate (SR), also die Dauer eines Samples in Sekunden
- 4. Messlänge eines Sets in der Einheit Samples (DS)
- 5. Anzahl an Sets (SA)
- 6. ToF
- 7. max. V (V)

Im gelben Paket sind die Rohdaten enthalten. Die Anzahl der hier hinterlegten Rohdaten (Amplitudenwerte) setzt sich aus der Messlänge eines Sets multipliziert mit der Anzahl an Sets zusammen (im angegebenen Beispiel sind es insgesamt 1000 Werte). Dies sind die Daten, die zur Darstellung eines A-Bilds verwendet werden.

Dieses Array wird an die entsprechende ROS-Applikation, die auf Laptop-A operiert, übersendet.

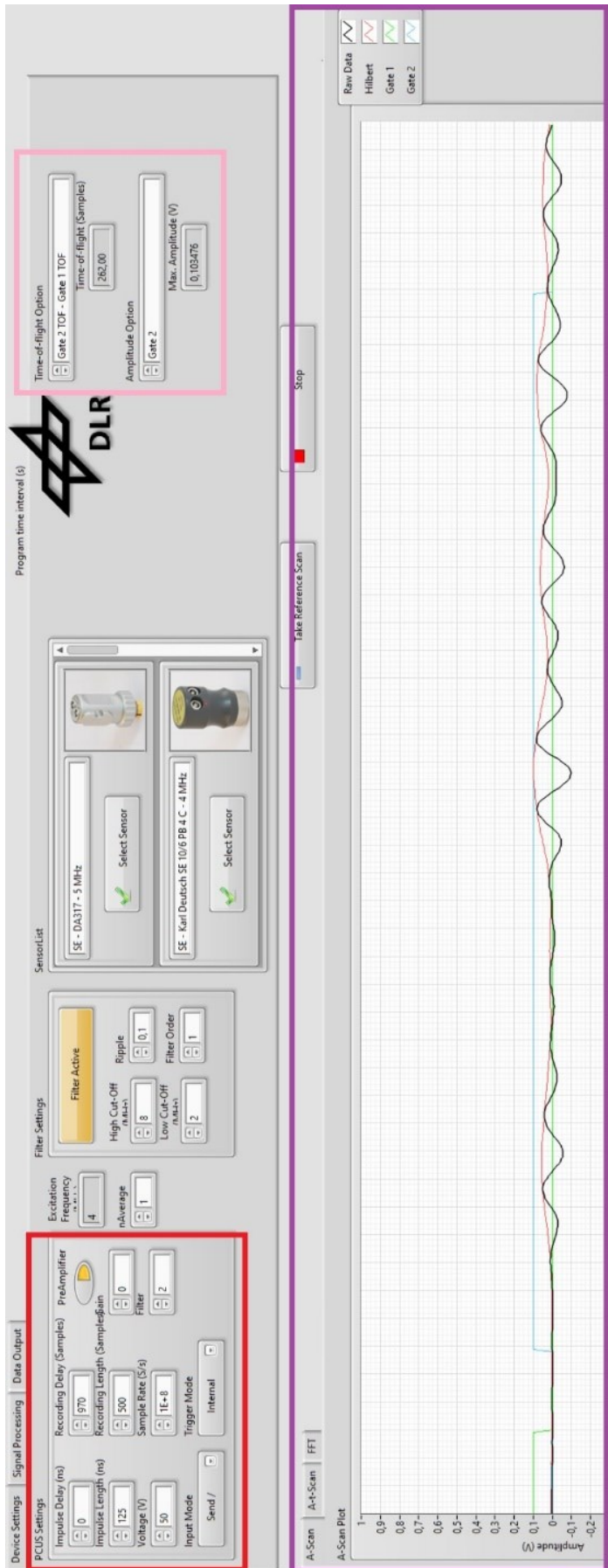


Abb. 25: LabVIEW Anwendung für das Projekt. Roter Kasten: PCUS-Einstellungen, Lila Kasten: A-Bild mit (Schwarz: Rohdaten und Blau: Auswertungsbereich), Rosa Kasten: ToF und maximale Amplitude aus dem A-Bild.

4.3 ROS-Applikation

ROS ist ein Framework zur Erstellung von Software und wird in diesem Projekt verwendet, um zwei wesentliche Aufgaben zu erfüllen:

1. Die Erstellung einer virtuellen Relation in Form eines Koordinatensystems $(KS)_{Edge}$ zwischen den physischen Objekten des Part-on-Carriers und des Prüfobjekts.
2. Die Verbindung von US-Sensordaten und dessen Position in dem Koordinatensystem $(KS)_{Edge}$ zur einheitlichen, digitalen Bestimmung der Messposition des US-Sensors am Prüfobjekt.

Erstellung eines Referenz-Koordinatensystems

Da das Koordinatensystem (KS) des Part-on-Carriers $(KS)_{Carrier}$ nicht mit der eigentlichen Position des Bauteils verknüpft ist, muss zuerst die Lage des Bauteils relativ zum Part-on-Carrier kalibriert werden. Dies geschieht unter jeweiliger Verwendung der optischen Marker des Carriers und des US-Sensors. Beide Objekte haben ihre eigenen Koordinatensysteme, die intern auf den Ursprung des Koordinatensystems der Kamera $(KS)_{Cam}$ berechnet werden.

Mit Hilfe des in ROS implementierten Kalibrierungsprozesses wird das Referenz-Koordinatensystem $(KS)_{Edge}$ definiert. Dazu werden die vier äußeren Eckpunkte der Prüffläche von Position „1“ bis Position „4“ der Reihe nach mit dem US-Sensor abgetastet und aus diesen das Koordinatensystem $(KS)_{Edge}$ gebildet.

Zur letztendlichen Positionsbestimmung des Ultraschallsensors im Referenz-Koordinatensystem wird das $(KS)_{US}$ verwendet und mit Hilfe der Transformationsmatrix (Gelb) auf das Referenz-Koordinatensystem von ROS $(KS)_{Edge}$ transformiert. $(KS)_{Edge}$ spannt dabei wie in Abb. 26 zu sehen ist auf.

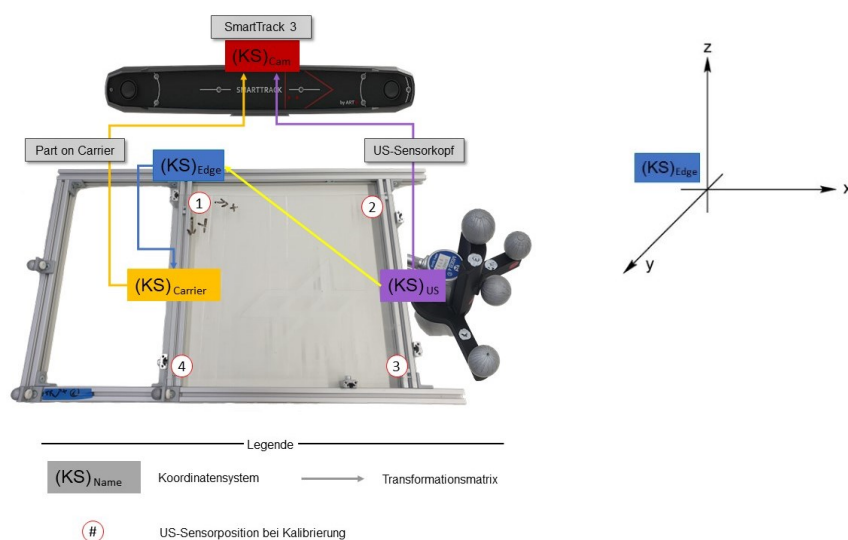


Abb. 26: Übersicht der Koordinatensysteme zur Ermittlung der Position vom Ultraschallsensor zum Referenz-Koordinatensystem $(KS)_{Edge}$.

Verbindung von Sensordaten und Koordinaten sowie Übersendung an die HL2

Das in Python3 geschriebene Skript greift das Ultraschalldaten-Array von der LabVIEW-Applikation ab und verschachtelt es wiederum weiter mit den neuen Koordinatendaten des Ultraschallsensors, welches danach an die HoloLens 2-Applikation als Bitstream über das entsprechende UDP-Skript versendet wird. Vor der Versendung der Informationen wird noch der Tiefenwert in der Einheit Meter errechnet und das Ergebnis hinzugefügt. Die Berechnung erfolgt grundsätzlich dabei wie bereits in Formel 2 erwähnt. Diese Formel muss jedoch mit den entsprechenden Werten aus LabVIEW bereichert werden, wodurch sich mit Rückblick auf die Informationen 4.2.1 folgende Formel ergibt:

$$d = \frac{c \cdot ((SD + ToF) \cdot SR)}{2} \quad 4.3.1$$

Der Faktor t wird aus der Summe von dem Sensor-Delay (SD) und Time-of-Flight (ToF) und dessen Multiplikation mit dem Zeitinkrement (SR) berechnet. Der Wert c wird auf Grundlage von Formel_4 im Anhang bestimmt.

Das neu verschachtelte Daten-Array sieht beispielhaft wie Folgt aus:

```
# x , y , z , d , 1 , 2 , 3 , 4 , 5 , 6 , 7 , ...
[[0.043, 0.105, 0.003], 0.013, [7.0, 970.0, 1e-08, 500.0, 2.0, 181.0, 0.1384], [...]]
```

4.3.2

Zu Beginn stehen die Koordinaten der drei Raum-Achsen (hier in Grün und mit gerundeten Werten angegeben) und an zweiter Stelle der berechnete Tiefenwert des Ultraschallsignals (hier in Braun und ebenfalls mit gerundeten Werten). Darauf folgt der bereits bekannte Informations-Teil aus 4.2.1. in Rot und Gelb.

Aufgrund einer Leistungsüberlastung des Prozessors und des Arbeitsspeichers der HL2 durch den UDP-Stream mussten die US-Sensor-Rohdaten am Laptop-A (gelbes Paket) gekürzt werden, wodurch lediglich mit den Positionsdaten, Tiefenwerten und den US-Sensor-Header-Daten in der HL2-Applikation gearbeitet wird. Eine Darstellung von A- oder B-Bildern in der HL2 ist daher in der Anwendung aktuell nicht möglich.

Unabhängig davon kann dieses Daten-Array vor der Versendung zudem für die spätere Wiederverwendung in ROS als JSON-Datenformat protokolliert werden. Hierdurch ist zudem eine nachträgliche Visualisierung auf der HL2 ebenfalls möglich und die Führung des Ultraschallsensors kann dadurch protokolliert und für das Prüfpersonal beispielsweise bei einer Schulung bewertet werden.

4.4 HoloLens 2 Applikation (NDT-US_HL2)

Zur Erläuterung der HoloLens 2 Applikation mit dem Projektnamen NDT-US_HL2 wird beginnend auf die verwendete Entwicklungsumgebung eingegangen. Anschließend werden die wesentlichen Grundfunktionen erklärt und deren Anwendung aufgeführt.

4.4.1 Bereitstellung und Definition der Entwicklungsumgebung

Mit Beginn des Jahres 2021 und den darin veröffentlichten Unity- und MRTK- Versionen haben sich wieder neue Vorgehensweisen entwickelt, wodurch die hier genannte Kombination aus Software und Plugins zwar für dieses Projekt weiterhin zur Entwicklung eingesetzt werden kann, diese jedoch für zukünftige Projekte mit neueren Softwareversionen nicht mehr ohne Einschränkungen zu empfehlen sind.

Auf dem Windows 10 – Entwicklungsgerät ist die Unity-Version 2019.4.12f1 installiert, als Programmierungsschnittstelle (API) wird Microsofts Visual Studio in der Version 16.9 verwendet. In Visual Studio selbst werden die Erweiterungen *UWP Essentials* und *Windows Mixed Reality App Template*, eingesetzt, um die nötigen Klassen-Bibliotheken und Tools zur Entwicklung einer UWP-App bereitzustellen.

In Unity wird hauptsächlich das interne Plugin *Windows Mixed Reality 4.2.3* verwendet, welches die native Entwicklung in Unity für WindowsMR- (Windows MixedReality-) Geräte unterstützt. Daneben wird das externe Plugin MRTK 2.5.1 von Microsoft eingesetzt, um vordefinierte Inhalte, Strukturen und Logik, wie z.B. UI-Elemente (User-Interface, zu Deutsch Bedienoberflächen) nutzen zu können. Zusätzlich wird das Plugin NuGet 2.0.1 eingesetzt, um unterstützend auf spezielle .Net-Framework-Pakete zugreifen zu können, die zur Benutzung von UWP-C#-Code in diesem Projekt notwendig sind. Die Funktion des „Holographic Remoting“ im Unity-Betrieb sowie der QR-Scanner greifen als Beispiel hierauf zurück. Auf diese und weitere Funktionen wird nachfolgend detaillierter eingegangen.

4.4.2 Applikation-Funktionen

Im folgenden Abschnitt werden alle relevanten Hauptfunktionen der HL2-Applikation erläutert und auf die fundamentale Programmierstruktur und Logik eingegangen. Der Aufbau des Kapitels folgt dabei dem Stil einer „Schritt-für-Schritt“-Anleitung, um die Zusammenhänge möglichst simpel nachvollziehen zu können. Eine zusammenfassende Veranschaulichung der wichtigsten Elemente und Funktionen ist bereits in Abb. 27 zu sehen, die jedoch erst durch die danach folgenden Inhalte erläutert wird.

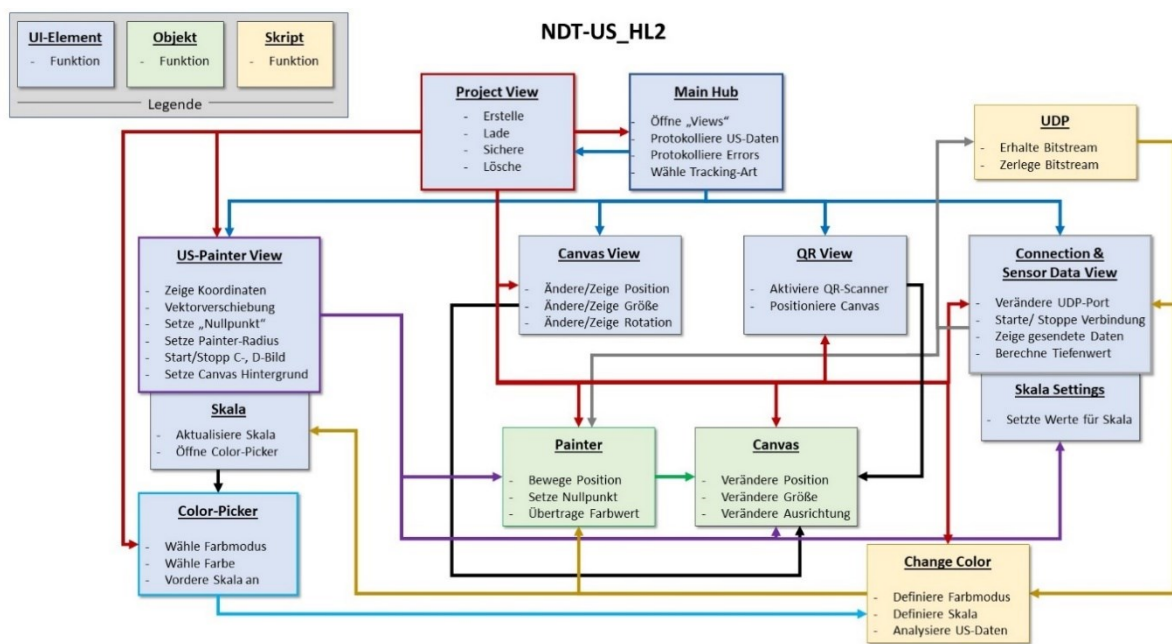


Abb. 27: Schematische Darstellung der HL2-Applikation.

Ein Projekt neu erstellen oder ein bestehendes Projekt speichern / laden

Zu Beginn der Anwendung ist das Hauptbedienfeld (Main-Hub) und die „Project-View“ geöffnet. Das Main-Hub dient aus klassischer Anwendungssicht als Hauptmenü, über das tiefgehend auf alle Funktionen zugegriffen werden kann. Diese werden jedoch erst nach der Erstellung oder des Ladens eines Projekts umfangreich nutzbar. Die Projekt-View verwaltet dabei über 200 Variablen und hat Zugriff auf alle wichtigen Skripte und Objekte, von denen die verschiedenen UI-Fenster ihre Inhalte beziehen. Die einzelnen Fenster (Views) können dabei beliebig aufgerufen und im Arbeitsbereich positioniert werden. Es wurde eine flache System-Hierarchie gewählt, um auf diese Weise möglichst viele Informationen auf einem Blick einsehen zu können, was besonders während der Entwicklungsphase von Nutzen ist, um Zeit zu sparen. Zwar haben die „US-Painter View“ und „Connection & Sensor Data View“ eine Unterkategorie, diese kann aber zusätzlich zu dem höhergestellten Fenster „aufgeklappt“ werden, wodurch das bestehende Fenster erweitert wird (Abb. 30). Nach der Erstellung oder des Ladens eines Projekts können verschiedene Einstellungen vorgenommen werden, die über die vier Hauptfenster erfolgen.

Die vier Hauptfenster und deren Funktionen

Die vier Hauptfenster sind die „QR View“, sofern die entsprechende Tracking-Variante im Main-Hub ausgewählt wurde, „US-Painter View“ mit „Skala“ und „Color Picker“, „Canvas View“ und die „Connection & Sensor Data View“ mit „Skala Settings“.

Zu Beginn eines neuen Projekts mit „unbekannter“ Umgebung wird das in ROS definierte Koordinatensystem mit dem HL2-Koordinatensystem abgeglichen. Wird im Main-Hub das Tracking mit zusätzlichem QR-Code ausgewählt, kann die „QR-View“ aufgerufen, und der Scanvorgang gestartet werden. Hierbei schaltet sich die Frontkamera der HoloLens 2 ein und sucht nach dem hinterlegten QR-Code. Wird dieser im Videostream der Kamera erkannt, werden dessen Daten im Fenster angezeigt (siehe Abb. 29). Alle drei Sekunden lang wird der Vorgang automatisch wiederholt, um eine Veränderung der realen QR-Marker Position zu registrieren. Das QR-eigene Koordinatensystem ist dabei nach internen Transformationen wie folgt aufgespannt (Abb. 28 (R)):

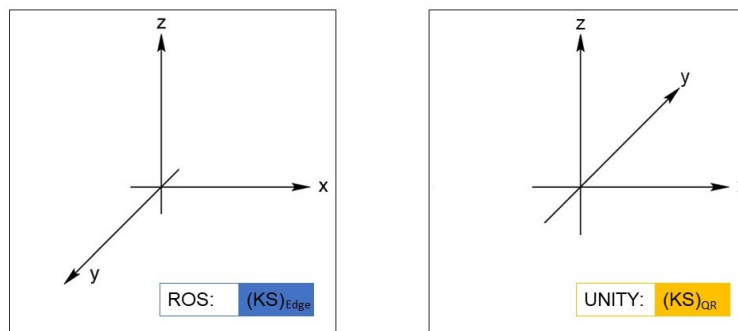


Abb. 28: Ausrichtung der Koordinatensysteme von ROS (L.) und dem QR-Marker in Unity (R.)

Das Koordinatensystem aus ROS hat die Eigenschaft, dass die y-Achse positiv „nach hinten“ zeigt, dadurch ist diese Achse im Gegensatz zum QR-Code in Unity invertiert (siehe Abb. 28).

Diese Eigenschaft wird in der „US-Painter View“ hinterlegt (Abb. 30).



Abb. 29: QR-View und QR-Code

Hier erfolgt zudem der weitere Abgleich, zwischen den Koordinaten von ROS und der HL2-Anwendung. Der reale Ultraschallprüfsensor wird dabei händisch auf den Ursprung des ROS ($KS)_{Edge}$ geführt. Danach wird der digitale Sensor aktiviert und mit Hilfe einer virtuellen Steuereinheit händisch auf die Position des realen Sensors geführt und diese Position gespeichert, die nun als Referenzpunkt gilt (siehe Abb. 31). Da hierzu kein automatisierter Vorgang implementiert wurde kann es sich bei dieser Methode nur um eine Annäherung beider Systeme handeln, dessen Präzision im Kapitel 7 analysiert wird.

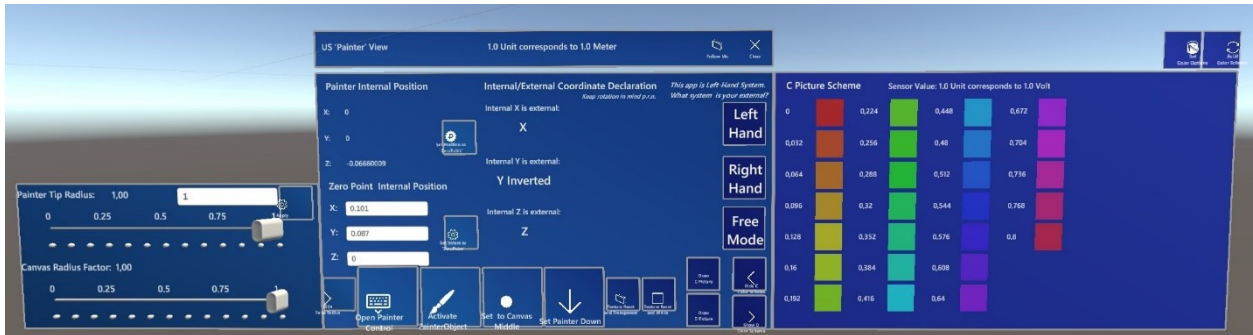


Abb. 30: „US-Painter View“ mit erweiterter Darstellung (Skala (L.) und Painter-Radius (R.))

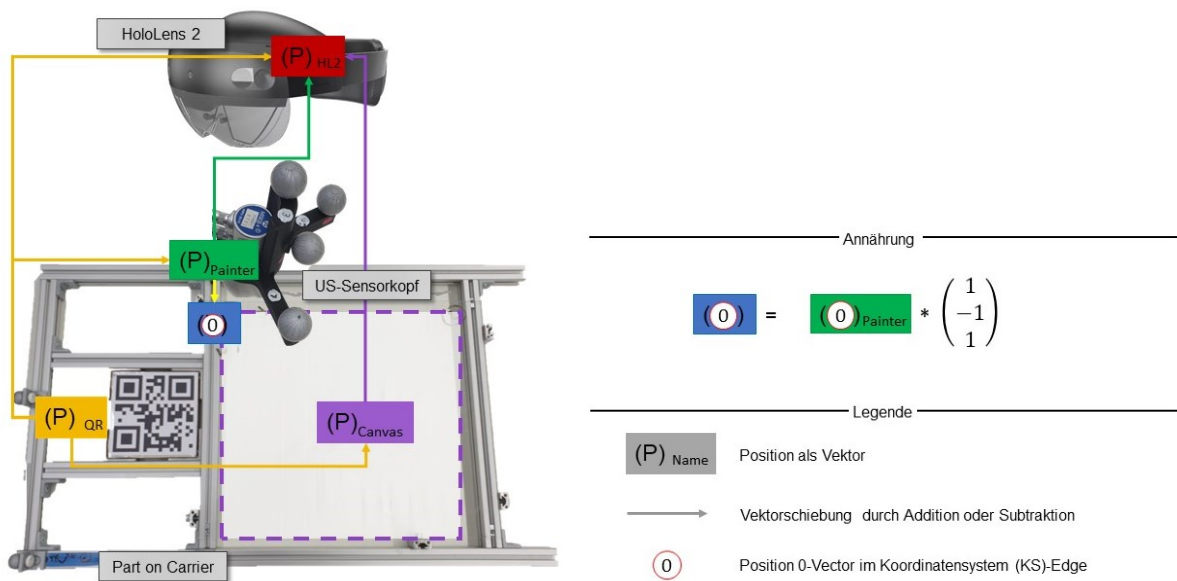


Abb. 31: Schematische Darstellung einer Annäherung der Koordinatensysteme von ROS und des QR-Markers

Durch das Scannen des QR-Codes ist bereits eine vordefinierte virtuelle Prüffläche zu sehen. Dabei befindet sich der Mittelpunkt dieser virtuellen Prüffläche auf dem Ursprungspunkt des QR-Codes. In der „Canvas View“ können die Größe, Position und Ausrichtung dieser Prüffläche angepasst werden. Dabei nähert sich eine virtuelle Einheit einem realen Meter an [Uni20], was zudem nach einer händischen Gegenmessung des virtuellen „Canvas“-Objekts und eines realen Objekts mit „gleichen“ Maßen für dieses Projekt ebenfalls angenommen wird.

Mit Hilfe der „*Connection & Sensor Data View*“ wird die Erlaubnis und der Port zum Datenerhalt an das UDP-Skript übermittelt. Dieses wandelt außerdem den eintreffenden Bitstream in einen String-Datenformat um und zerlegt die darin enthaltenen Inhalte (bekannt aus 4.3.2, hier jedoch ohne die Rohdaten) in einzelne Variablen, die in diesem Fenster einzusehen sind. Hier befindet sich zudem die Analyse des aktuellen US-Sensor-Wertes und dessen farbliche Repräsentation. Wie diese Analyse und Farbzuzuweisung eingestellt und erfolgt wird nun erläutert.

Im erweiterten Bereich der „*Connection & Sensor Data View*“ kann der Werte-Bereich der Farb-Skala definiert werden, der zur Analyse der US-Sensordaten im „*Change Color*“-Skript verwendet wird (siehe Abb. 32).

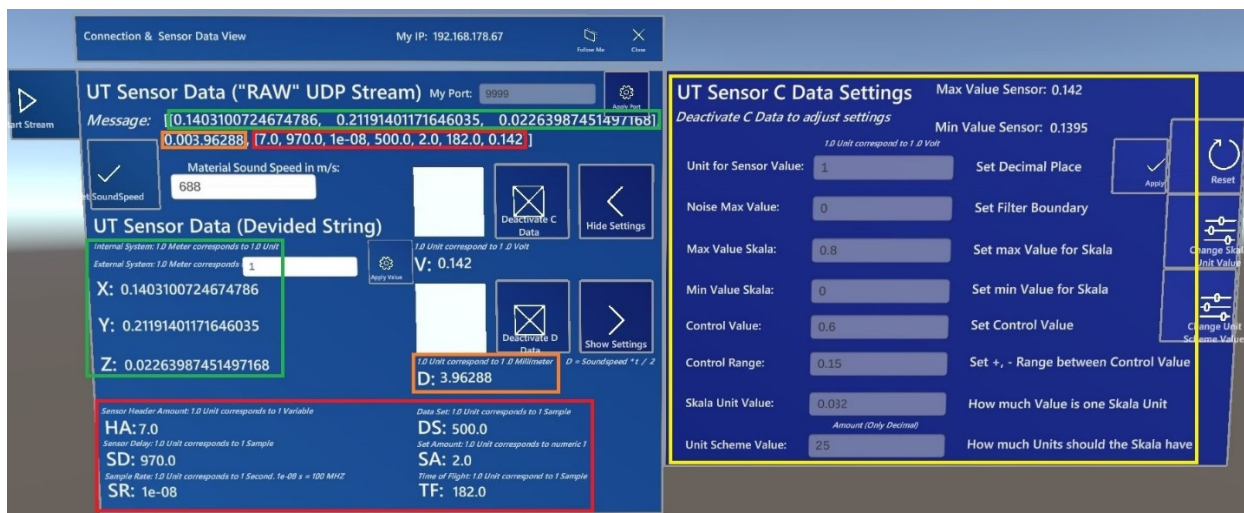


Abb. 32: ROS-Daten-Array in der „*Connection & Sensor Data View*“ (L.) (Grün: Koordinaten, Braun: Tiefenwert, Rot: Ultraschalldaten-Header) und erweiterter Darstellung. (Gelb).

Hier muss bei der Eingabe eine Einschätzung erfolgen, in welchem Werte-Bereich die Ultraschalldaten aus LabVIEW liegen und in welchem Bereich dadurch eine farbliche Zuordnung erfolgen soll („*Min Value Skala*“ bis „*Max Value Skala*“). Eine Einheit entspricht dabei 1.0 Volt. Das tatsächliche Maximum und Minimum wird an der Stelle „*Max Value Sensor*“ und „*Min Value Sensor*“ angezeigt und bei einer Werteänderung aktualisiert. Daneben kann zusätzlich ein Kontrollwert („*Control Value*“) mit entsprechender Bandbreite („*Control Range*“) definiert werden. Diese Werte können z.B. dafür benutzt werden, um dem Analyseprozess im Skript („*Change Color*“) zu verständigen, dass alle Sensorwerte in diesem Bereich ein unbeschädigtes Material bedeuten.

Über die Variablen „*Skala Unit Value*“ und „*Unit Scheme Value*“, wird die Anzahl an Farbeinstufungen bzw. die Einheit einer Farbrepräsentation festgelegt und eingesehen.

In Abb. 32 wird die Skala zur Darstellung eines C-Bilds eingestellt und ein Messbereich von 0.0 V bis 0.8 V festgelegt. Dabei werden 25 Farbeinstufungen verwendet, die jeweils eine Einheit von 0.032 V repräsentieren. Der Kontrollbereich liegt zwischen 0.45 V und 0.75 V (0.6 +/- 0.15).

Separat müssen für die Aufnahme eines D-Bilds dieselben Einstellungen in der „*Connection & Sensor Data View*“ vollzogen werden. Anstelle von Amplitudenwerten in der Einheit Volt müssen jedoch entsprechend Tiefenwerte in der Einheit Millimeter angegeben werden.

Da die Werte nun festgelegt sind (diese aber zu jeder Zeit auch angepasst werden können) werden aus diesen Eingaben die Skala gebildet, die in der erweiterten Ansicht im „*US-Painter View*“ zu sehen ist (bereits auf Abb. 30 abgebildet). Von hier aus kann das Fenster „*Color Picker*“ aufgerufen werden, um die Farbvergabe oder den gesamten Darstellungsmodus zu ändern.

Es sind zwei grundsätzliche Modi zur Farbvergabe möglich: Der „*Full Range Mode*“, bei welchem die komplette Farbskala im HSV-Farbraum verwendet wird und der „*Contrast Mode*“, bei dem aus vordefinierten Farben im RGB-Farbraum zurückgegriffen wird und dessen Verläufe entsprechend dynamisch berechnet werden (siehe Abb. 33).

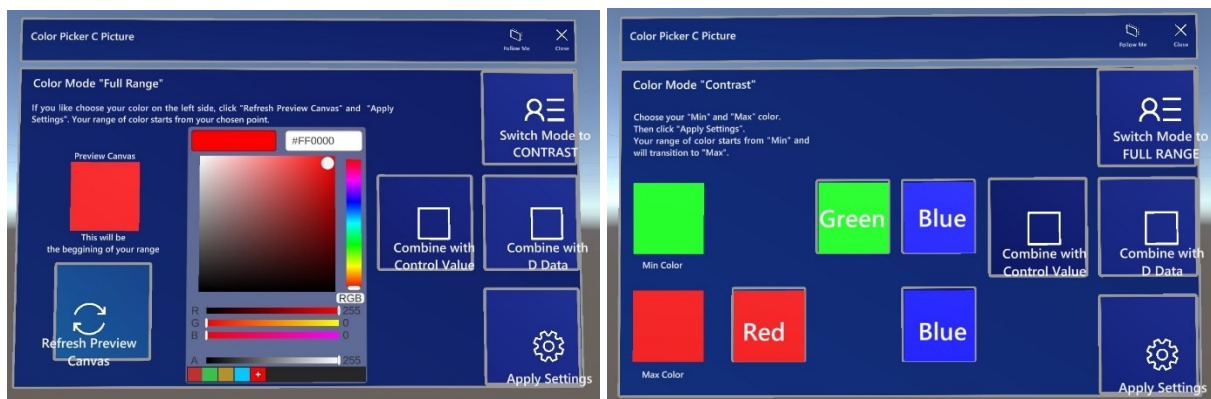


Abb. 33: UI-Darstellung „*Color Picker*“ im *Full-Range-Mode* (L.) und *Contrast-Mode* (R.)

Grundsätzlich wird bei der Sensordatenanalyse beim C-Bild der Amplitudenwert und beim D-Bild der Tiefenwert farblich kategorisiert. Durch die Hinzunahme weiterer Kategorisierungsfaktoren (Kontrollbereich und/ oder „zusätzlicher Wert“, welches beim C-Bild der Tiefenwert und beim D-Bild der Amplitudenwert ist) ändert sich die Darstellungsform. Wird der Darstellungsmodus nur um den Wert „zusätzlicher Wert“ erweitert, bleibt die bisherige Skala bestehen und in der Analyse wird die entsprechende Grundfarbe als Hauptwert verwendet. Der „zusätzliche Wert“ wird entsprechend seiner C- oder D-Skala durch einen Grauwert dargestellt, der mit der Grundfarbe verbunden wird (eine beispielhafte Darstellung erfolgt in Kapitel 7).

Bei der alleinigen Hinzunahme des Kontrollwerts ändert sich die Farbskala drastisch und wird auf drei wesentliche Farbwerte reduziert, unabhängig davon, wie viele Einheiten vorab in den „Skala-Settings“ definiert wurden. Zudem repräsentieren die Farben hierbei keinen Sensorwert mehr, sondern einen von drei Zuständen.

Diese Zustände lauten: „Über dem Kontrollbereich“, „Im Kontrollbereich“ und „Unter dem Kontrollbereich“.

Entsprechend wird der Sensorwert dahingehend analysiert, ob dieser „über“, „im“ oder „unter“ dem Kontrollbereich liegt (siehe Abb. 34).



Abb. 34: UI zur farblichen Bestimmung des Kontroll-Zustandes im *Full Range Mode* (L.) und im *Contrast Mode* (R.)

Soll bei der Analyse sowohl der Kontrollbereich, als auch der „zusätzliche Wert“ berücksichtigt werden, müssen vier Zustände farblich repräsentiert werden, da nun der Kontrollbereich des Amplitudenwerts und des Tiefenwerts verwendet wird (Abb. 35). Die vier zustände lauten:

- „Amplitude & Depth in Range“: Beide Werte liegen in ihren jeweiligen Kontrollbereichen
- „Depth not in Range“: Nur der Tiefenwert liegt nicht in seinem Kontrollbereich
- „Amplitude not in Range“: Nur der Amplitudenwert liegt nicht in seinem Kontrollbereich
- „Amplitude & Depth not in Range“: Beide Werte liegen nicht in ihren Kontrollbereichen

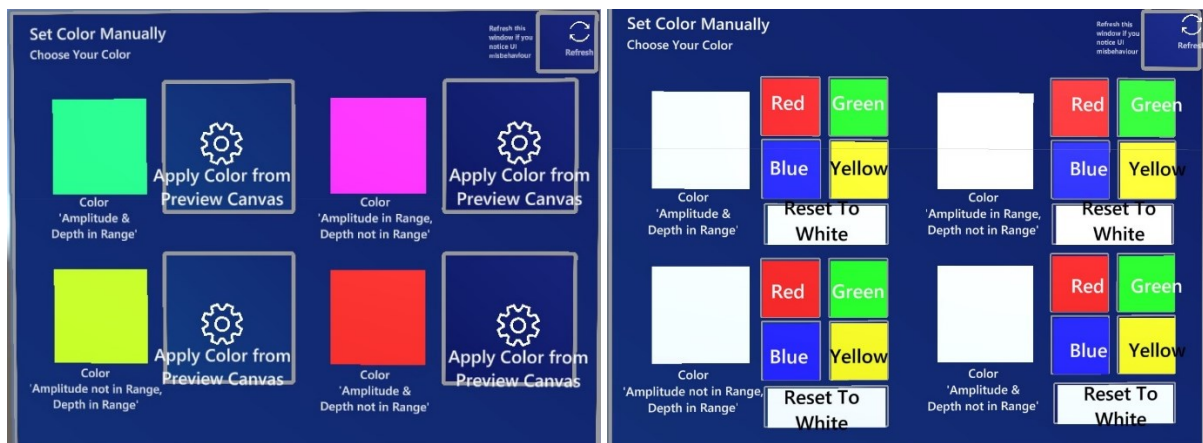


Abb. 35: UI zur farblichen Bestimmung der Kontroll-Zustände im *Full Range Mode* (L.) und im *Contrast Mode* (R.)

Nun sind alle nötigen Einstellungen in der HL2-Anwendung getätigt und die ZiP kann durchgeführt werden. Es stehen daneben weitere Funktionalitäten zur Auswahl.

Weitere Funktionalitäten

Ein wesentlicher Bestandteil zur Programmierung der genannten Funktionalitäten ist das „Task-based Asynchronous Pattern“ (TAP) zu Deutsch das „aufgabenbasierte asynchrone Programmiermodell, welches mit dem Framework .Net 4.5 umgesetzt wurde. Dieses Modell wird z.B. für das UDP-Skript zur Verarbeitung des Bitstreams verwendet sowie bei allen Methoden eingesetzt, die eine Datei auf dem Gerät erstellen, Zugriff auf diese haben und sie mit Informationen beschreiben. Allgemein dient das Modell zur Umsetzung von „Multi-Threads“, also zur Regelung mehrerer Aufgaben, die parallel ablaufen und entweder ein gemeinsames Ziel verfolgen oder mehrere verschiedene Ziele anstreben [Mic_2].

Bei der Verarbeitung des Bitstreams in der HL2 Applikation müssen z.B. Funktionen, die auf diese Informationen angewiesen sind, aufgrund von TAP nicht erst warten, bis der komplette Bitstream ausgelesen wurde, um danach weitere Methodenaufrufe zu tätigen, sondern können bereits vorweg agieren und die Informationen verarbeiten, die sie bereits vorliegen haben. Selbige Strategie wird zudem auch beim Speichern von Einstellungen, der Erstellung der Bilddatei und der Protokollierung von Systemnachrichten oder der analysierten Sensordatenwerte verwendet.

Zusätzlich mussten alle diese Methoden in doppelter Ausführung programmiert werden, sodass sie einerseits in der Entwicklungsumgebung in Unity aber andererseits auch in der Software-Umgebung der HoloLens 2 funktionieren.

Daneben entstanden eigene Implementationen von virtualisierten Eingabemethoden. Zum einem wurde eine numerische Tastatur zur Eingabe von Zahlen entwickelt, da an entsprechenden Stellen eine andere Eingabe unnötig und gar zu Fehlern führen könnte. Zum anderem wurde ein Bedienelement zur virtuellen Steuerung und Kalibrierung des US-Prüfkopf-Platzhalters entworfen (Abb. 36).

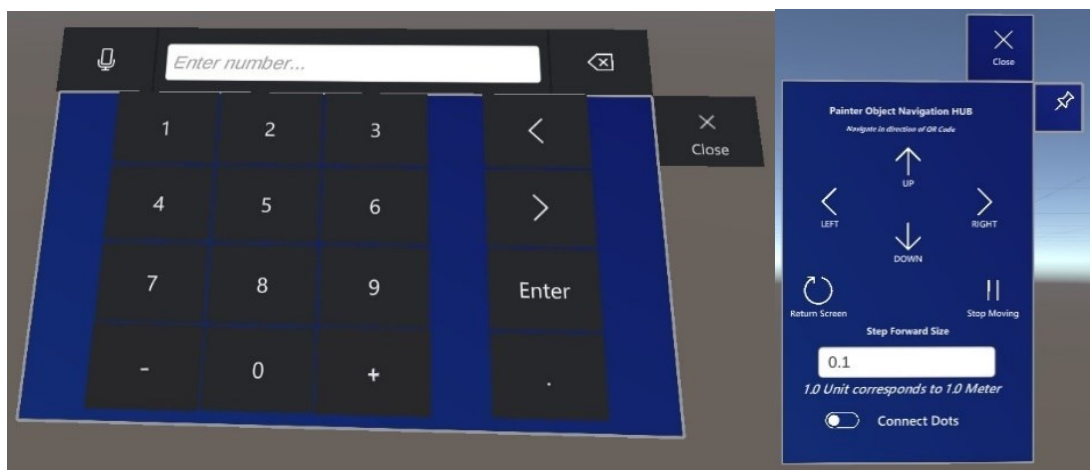


Abb. 36: Virtualisierte Eingabemethoden. Numerische Tastatur (L.) und US-Controller (R.)

Aufgrund der in Abschnitt 4.4.1 aufgebauten Entwicklungsumgebung und der Funktionalität des programmierten Codes, der sowohl in Unity als auch in der HL2 lauffähig ist, wird es ermöglicht das sogenannte „Holographic Remoting“ zu verwenden [Mic_3]. Hierdurch ist die Anwendung in der Lage, auf einem Computer betrieben zu werden, um dessen Rechenleistung zu nutzen. Eine Darstellung erfolgt jedoch auf der HoloLens 2 und deren Eingabemethoden werden verwendet. So kann aus der Entwicklungsperspektive gesehen das Programm schneller getestet werden und Fehleranalysen direkter erfolgen. Gleichzeitig ist es aus der Nutzungsperspektive betrachtet möglich, rechenintensive Programme auf der HoloLens 2 darzustellen, die diese bei direktem Betrieb auf dem Gerät überfordern würde. Hierfür muss die primäre leistungsfähige Recheneinheit eine stabile Verbindung (über WLAN oder USB) zur HL2 aufbauen können.

Eine weitere Funktion ist die Einsicht der Applikationsdaten, die auf der HL2 gespeichert werden. Die wichtigsten Daten wären für diese Applikation die erstellten C- oder D- Bilder sowie eventuelle Protokoll- oder Einstellungsdateien. Auf diese Daten kann mit Hilfe des Windows-Device-Portals (WDP) zugegriffen werden. Dabei handelt es sich um einen geräteinternen Webserver, der die Applikationsdaten zur Ansicht oder zum Download bereitstellt [Mic_4].

5. Visualisierung von Ultraschalldaten für ein HMD

Wie bereits im Grundlagenteil 2.1 erwähnt wurde befasst sich die bildgebende Ultraschallprüfung seither mit der Frage, wie die empfangenen Messdaten eines US-Sensors für die prüfenden Personen aufbereitet und dargestellt werden können. Mit der zunehmenden Verbreitung von xR-Technologien stellt sich hier erweiternd die Frage, ob die bisherigen Visualisierungsmethoden auch auf Darstellungsgeräten, wie einem Head-Mounted-Display, ihre Relevanz behalten und wie die Vorteile von HMD's genutzt werden könnten, um auf ihnen zukünftige Visualisierungsmethoden bereitzustellen.

Weshalb eine Nutzbarkeit der bisherigen Visualisierungsmethoden auf HMD's jedoch vorerst von Vorteil ist zeigt [Deu97, S.338]. Die Kosten einer Schulung werden bei einer Skala von 1 (niedrig) bis 10 (hoch) auf den Wert 10 eingestuft. Umfangreiche und kostenintensive Neuschulungen von bestehendem Personal zur Interpretierung der angepassten Darstellungen blieben daher so vorerst erspart und trotzdem könnten die Vorteile von HMD'S, wie der handfreien Mobilität und einer direkten Zuordnung der Messwerte während einer Prüfung, genutzt werden. Jedoch sind die primär einbezogenen Faktoren zur Bewertung einer geeigneten Darstellungsmethode folgende:

- Benötigte Rechenleistung zur Visualisierung
- Deutliche Interpretation der Darstellung
- Dynamische Anpassung der Visualisierung an die Prüfteilstruktur
- Benötigte Zusatzleistung zur Verwendbarkeit der Visualisierungsdaten auf externen Programmen

Die drei grundsätzlichen Darstellungsmethoden in der 3D-Computergrafik sind unter der Verwendung von Voxel-, Point-Cloud (Punktwolken) oder Polygon-Verfahren zu erzielen, die jeweils ihre eigenen oder allgemein-angepassten Algorithmen besitzen, um aus numerischen Daten geometrische Formen als Repräsentation zu erzeugen [Sco09, S.1-2] (siehe Abb. 37).

Als Voxel werden die einzelnen Kreuzungspunkte in einem dreidimensionalen Gitter bezeichnet, wodurch ihnen die äquivalente Bedeutung wie der eines Pixels in einem 2D-Bild gegeben wird. Ihre Position wird dabei nicht explizit ermittelt, sondern implizit aus der Position zu anderen Voxel hergeleitet. Im Gegensatz dazu werden bei Punkten oder Polygonen die Positionen der (Eck-) Koordinaten gespeichert.

Voxel sind im Vergleich zu Polygonen rechenintensiver. Das liegt einerseits daran, dass die konventionelle Computergrafik auf eine Polygonberechnung optimiert ist und andererseits an ihrer Grundstruktur, die den gesamten grafischen Raum füllt [Sco09, S.1-2]. Jedoch besitzen Punktwolken und Polygone dadurch grundsätzlich den Nachteil, dass deren resultierenden 3D- Grafiken ein „leeres“ oder homogengefülltes inneres besitzen, während bei Voxel die Darstellung eines divergenten Inneres innewohnt. Der

Vorteil bei Punktwolken und Polygonen liegt in der unkomplizierteren Darstellung von komplexen, runden Objekten, die bei Voxel nur durch eine erhöhte Anzahl der verwendeten Voxel erreicht wird, wodurch sie sich wieder mit den Eigenschaften von Pixeln ähneln. Es ergeben sich daraus zwei grundsätzliche Anwendungsunterschiede und Handlungsempfehlungen [Hug13, S. 349]:

- Zur Generierung von komplexen, flexiblen Oberflächen-Strukturen sind Punktwolken oder Polygonzüge gut geeignet.
- Zur Darstellung von detaillierten Innenaufnahmen eignet sich speziell der Einsatz von Voxel.

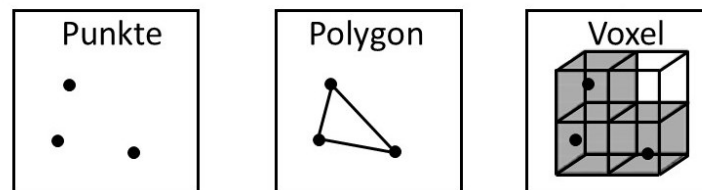


Abb. 37: Zusammenhang von Punkten (L.), Polygonen (M.) und Voxel (R.)

Ergebend aus den genannten Eigenschaften und Vorteilen von Polygonen werden nachfolgend Darstellungsstrategien fokussiert für diese Verfahren erläutert. Jedoch zeigt Abb. 37 bereits, dass eine klare Trennung der Verfahren nicht zwingend notwendig ist und sich diese durchaus auch kombinieren lassen, was jedoch den Rahmen dieser Arbeit übersteigt.

Unterstützend zur Argumentation der gewählten Methode zeigen [Ros19] und [Liv13], dass eine alleinige Zuordnung eines 3D-Holograms im Raum zudem für Fehleinschätzungen bezüglich der Tiefenlage dieses Objekts im Raum erzeugen kann. Daher wäre bei einer 3D-Darstellung von Materialschäden fokussiert auf Grundlage von Voxel aber auch von Polygonen, die zusätzliche Implementation von visuellen Hilfestellungen ratsam, um die menschliche Wahrnehmung dahingehend zu unterstützen durch geschlossene Objekte „hindurchsehen“ zu können und den virtuellen Raum stärker mit dem realen Raum zu verknüpfen. Dies kann z.B. durch die vereinzelte Verdeutlichung von virtuellen Böden, Flächen und / oder Kanten geschehen, sowie Licht- und Schatten-Verhältnisse genutzt werden, wie es z.B. Schematisch bei [Liv13, S.15-23] getan wird.

Eine Ausarbeitung und Studiendurchführung zur fokussierten Lösung dieser Problemstellung wäre dabei Aufgabe einer weiterführenden Behandlung dieser Thematik an zukünftiger Stelle.

Visualisierungskonzepte

Polygone können mit Hilfe von Algorithmen bestimmte Netzstrukturen annehmen, die zusammenfassend in drei verschiedene Arten unterteilt werden. Diese basieren auf Grundlage einer rechteckigen, dreieckigen oder zufälligen Basenstruktur (siehe Abb. 38).

Diese Strukturen werden genutzt, um sich der grafischen Form eines Objekts so genau wie möglich anzunähern. Dies kann sowohl in 3D als auch in 2D erfolgen (Abb. 39).

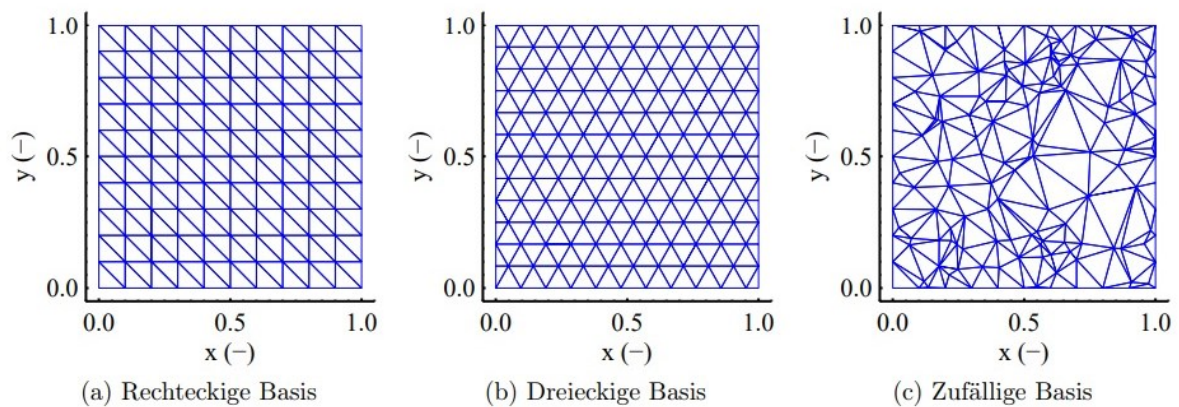


Abb. 38: Polygonnetze mit unterschiedlicher, globaler Struktur [Rad16, S.150].

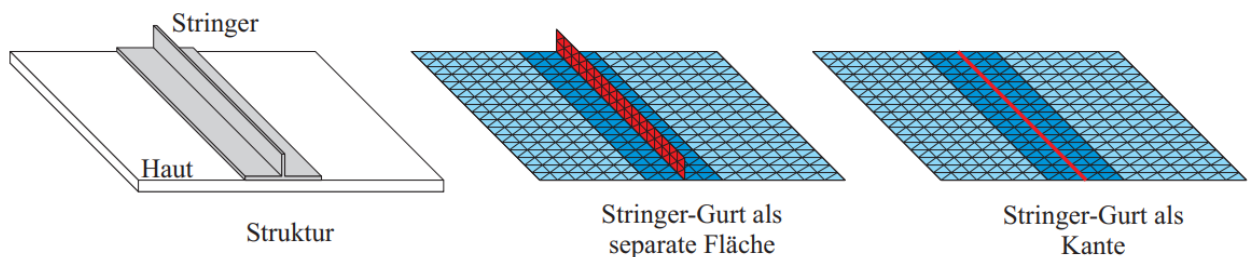


Abb. 39: Vernetzungsbeispiele einer stringerverstärkten Haut [Rad16, S.145].

Besonders sinnvoll ist dieser Ansatz, wenn eine dynamische Generierung einer virtuellen Struktur zur Laufzeit während einer US-Messung erfolgen und auf jeden Punkt und jeder Flächeneinheit zugegriffen werden soll, um diese manipulieren zu können. So lässt sich z.B. jeder Punkt als US-Messpunkt beschreiben, der die zur Position relevanten Ultraschallinformationen enthält und die anliegende Fläche entsprechend der eingestellten Farbskala einfärbt. Ein Tiefenwert ließe sich zudem zusätzlich durch Extrusion und Intrusion der Fläche im 3D-Kontext darstellen, die dadurch aus dem Körper heraus- oder hineinragt (Abb. 40). Je größer jedoch die geprüfte Fläche wird, umso rechenintensiver wird die Darstellung. Da zudem Genauigkeiten im Millimeter Bereich angestrebt werden ist von vornerein eine feine Rasterung mit mehreren Polygonen notwendig, wobei eine feine Tiefenveränderungen zudem visuell nicht wahrgenommen werden können.

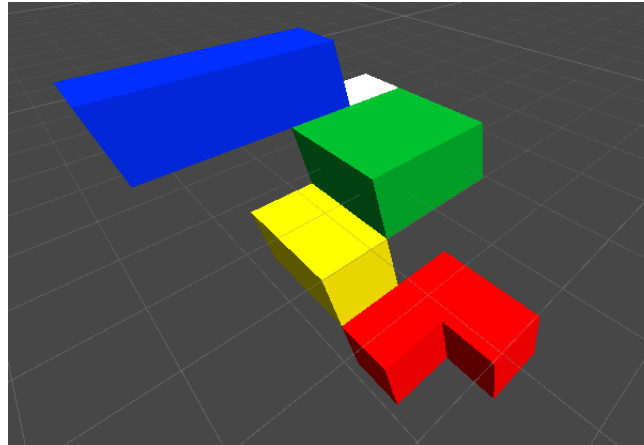


Abb. 40: Visualisierungskonzept eines rechteckigen Polygonnetzes mit US-Sensor-Farb- und Tiefenwerten

Ein zweiter Ansatz liegt nicht im Fokus der Erstellung einer Oberflächenstruktur des Prüfkörpers, sondern alleinig auf der des zurückgelegten Weges, wodurch eine Analyse und Neu-Kalkulierung zu den Nebenknoten entfällt, die für die Generierung einer geschlossenen Oberflächenstruktur nötig ist. Allerdings erhält das entstehende 3D-Objekt dadurch eine komplizierte Form, die im Nachhinein schwerer zu interpretieren ist (Abb. 41).

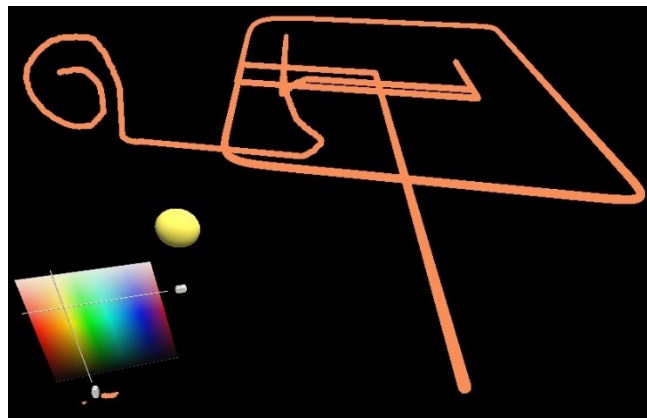


Abb. 41: Visualisierungskonzept zur Generierung eines 3D-US-Messpfades in AR

Auswahl der Visualisierung und Implementierung

Mit Erhalt der Erkenntnisse aus den beiden Ansätzen (Visualisierung einer Oberflächenstruktur oder eines Pfades) wurde eine Simplifizierung der ersten Variante entwickelt. Zum einen ist die Variante einer 2D-Fläche implementiert, die eine weiße 2D-Textur im Datenformat „JPEG“ enthält, die alternativ aber auch Transparent gesetzt werden kann.

Der virtuelle US-Sensor sendet einen mathematischen Strahl (Raycast) aus, der bei Überschneidung mit der Fläche und Erhalt eines US-Messwertes die Farbe entsprechend der Skala kalkuliert und an der überschneidenden Stelle auf die Textur setzt. Der Vorteil liegt in der geringen Rechenleistung, die jedoch zu einem Verlust des Einsatzes an dynamischen und komplexen Oberflächenstrukturen führt. Abhilfe schafft

hier die Möglichkeit zur Einbindung eines vordefinierten 3D- Modells. An Stelle der 2D – Fläche kann ein beliebiges Objekt verwendet werden, welches aktuell jedoch „händisch“ in der Entwicklungsumgebung in die Anwendung geladen werden muss und im Idealfall der zu untersuchenden Struktur ähnelt bzw. diesen als digitalen Zwilling repräsentiert. Zudem muss dem Objekt eine spezielle Textur vorliegen, die UV-Map genannt wird. Mit Hilfe der „Map“ findet eine Vereinigung von 2D-Bild und 3D-Körper statt, da die UV-Map die Textur in bestimmte Bereiche einteilt. UV-Maps folgen keiner einheitlichen Einteilung und können für jedes Objekt frei definiert werden. Als Beispiel einer UV-Map dient Abb. 42. Hier ist eine Textur für einen sechsseitigen Würfel zu sehen und die UV-Bereiche zur Veranschaulichung rot markiert.

Ein Vorteil bei dieser Visualisierungsmethode ist, dass dasselbe Prinzip wie bei der 2D-Fläche verwendet wird. Jedoch ist eine UV-Map alleinstehend als Bild sehr abstrakt, da die Bereiche und deren Orientierung nicht intuitiv sind und bei steigender Komplexität des verwendeten 3D-Modells der Abstraktionsgrad äquivalent ansteigt. So kann am Ende einer Prüfung in AR mit einem 3D-Modell auch nur eine Weitergabe des Modells in Verbindung mit der Textur für eine externe Weiterverarbeitung sinnvoll sein.

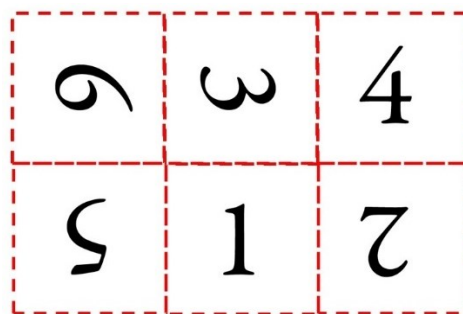


Abb. 42: UV-Map eines sechsseitigen Würfels mit eingezeichneten Bereichsabtrennungen

Dennoch zeichnen sich beide Varianten durch eine geringe Rechenlast aus, da mit dem Einsatz simpelster grafischer Primitive schnelle grafische Berechnungen vollzogen werden. Beide Lösungen zeigen im virtuellem keine zusätzliche Abstraktion, die 3D Variante kann hier sogar überwiegen, wenn exakte Modelle verwendet werden, jedoch entsteht hier eine Abstraktion bei alleiniger Betrachtung der Bilddatei.

Dynamische Anpassungen, wie die Einbindung von Verformungen oder Kerben sind in diesem Fall durch die Kopplung zusätzlicher Lösungen wie beispielsweise mit [Fli21] möglich, die über eine zukünftige Implementation erfolgen müssten.

Beide Visualisierungsmethoden entfernen sich im virtuellen Kontext dabei nicht von den konventionellen Darstellungsmethoden, da ein C/D-Bild als Textur verwendet bzw. während einer Prüfung erstellt wird und dieses weiterhin für das NDT-Personal zu verstehen ist.

Eine Methode, die zu Beginn der Recherche nach Visualisierungsmöglichkeiten als Inspiration galt, da sie Voxel- und Polygontechniken miteinander vereint, und somit beide genannten Eigenschaften (Darstellung von komplexer Oberflächen- und Volumenstruktur) verbindet, ist das verwendete Verfahren von [Pau18]. Hier werden aus Computer- oder Magnetresonanztomographen gewonnene Bildermengen zu einem Voxel-Modell geformt und danach mit Hilfe des „Marching Cube“-Algorithmus an eine Polygondarstellung angenähert.

Jedoch ist dieser Ansatz für dieses Projekt mittlerweile weniger relevant geworden, da zwar eine Umsetzung und Optimierung in AR auf die HoloLens 2 erzielt, allerdings keine Möglichkeit zur Echtzeitberechnung nach Systemstart entwickelt werden konnte und zudem ein großer Datensatz verwendet werden muss, um detaillierte Darstellungen zu erhalten (Abb. 43).

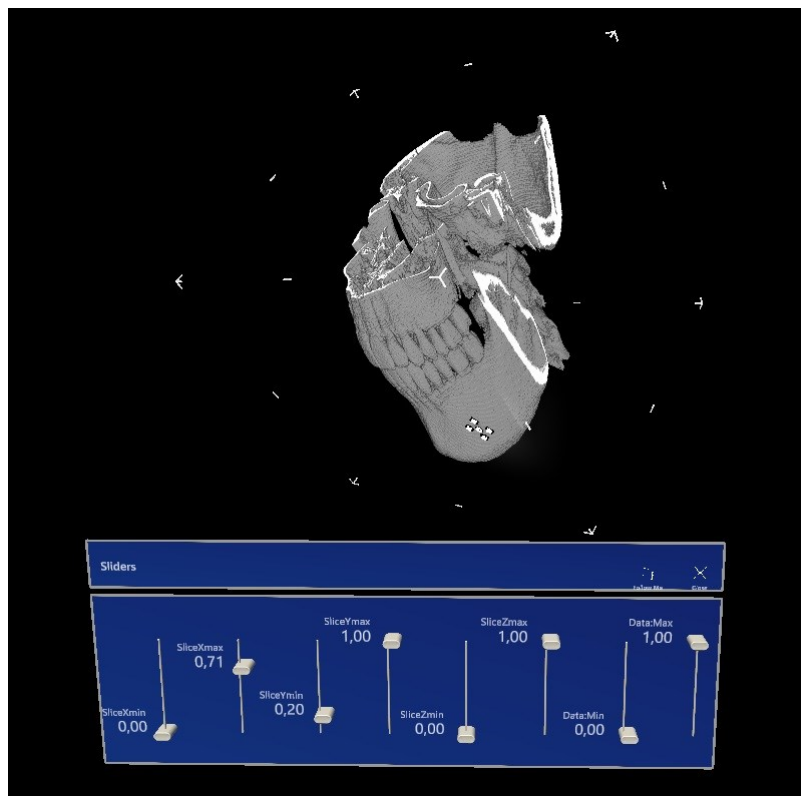


Abb. 43: Ansatz zur Fusionierung von Voxel- und Polygon-Algorithmen nach [Pau18] in einer eigenen HL2 Anwendung.

Die Nutzung von Verfahren zur Bearbeitung von Punktwolken wurde in diesem Abschnitt nicht weiter erläutert, da zur Darstellung einer geschlossenen Oberflächenstruktur hierdurch mehr Punkte entstehen und benötigt werden, als bei der in der Anwendung implementierten Variante.

Daher ist die Generierung einer Punktwolke für diesen Systemstand zwar nicht mehr relevant, besitzt jedoch für zukünftiger Implementierungen z.B. von Markerless-Tracking Varianten jedoch durchaus ihre Wichtigkeit.

6. Integration der HoloLens 2 in ein externes, markerbasiertes, Outside-In Trackingsystem

Es ist zu erwähnen, dass die Funktion einer deckungsgleichen Darstellung der virtualisierten US-Sensordaten am inspizierten Prüfteil in AR bis zum jetzigen Projektstand der Applikation nur über den kombinierten Einsatz des externen Systems (SmartTrack) mit dem internen System (HoloLens 2) umgesetzt ist. Dabei werden der Ultraschallsensor und Part-on-Carrier von der externen Kamera erfasst und durch das entsprechende ROS-Skript in ein Referenz-Koordinatensystem transformiert. Ein QR-Marker dient der HL2 als Bezugspunkt für den Part-on-Carrier. Von diesem Punkt aus nähert sich das System der HL2 dem ROS-Koordinatensystem an, um die gesendeten Positionsdaten des US-Sensors aus ROS deckungsgleich verarbeiten zu können. Hierbei ist jedoch vorab eine händische und dadurch weniger präzise Führung des Sensorkopfes zur Kalibrierung nötig, was bereits in Abschnitt 4.4.2. und Abb. 31 erläutert wurde. Es stellt sich die Frage, wie ein alleiniges, externes Tracking ohne QR-Marker umzusetzen ist, bei dem das Tracking-Externe Darstellungsgerät (in diesem Projekt die HL2) ebenfalls erfasst wird und sich daraus die darzustellende Perspektive der anderen erfassten Objekte automatisiert ergibt, ohne eine händische Kalibrierung durchführen zu müssen. Hierzu sind grundsätzlich 2 Anforderungen zu erfüllen:

1. Zuerst muss sichergestellt werden, dass die HL2 von der SmartTrack Kamera erfasst wird.
2. Anschließend müssen die Daten in einer bestimmten Form berechnet werden, sodass die Applikation den virtuellen Inhalt mit dessen korrekt transformierten Koordinaten projizieren kann.

Die Erfassung der HL2 wird, wie bei den bisherigen beiden Objekten, mit Hilfe eines retroreflektierenden Marker-Baums erzielt, der zum Zeitpunkt dieses Projektstands mit einer provisorischen Halterung an der HL2 befestigt ist und zukünftig mit einer speziellen Fassung erfolgen muss.

Der zusätzliche Marker-Baum wurde in der entsprechenden Software (DTrack) auf Laptop-A hinterlegt. Mit Hilfe des ART-Plugins für Unity ist zudem eine direkte Schnittstelle zwischen den transformierten Informationen aus DTrack und der HL2-Applikation gegeben. Jedoch sind an dieser Stelle in der Entwicklung der Anwendung noch selbstständige Eingriffe in die Programmierung nötig, da die reale Position der benutzenden Person (und somit die der HL2) bei Applikationsstart den digitalen Ursprung des Koordinatensystems der Anwendung bildet ($(KS)_{Unity}$ siehe Abb. 44) und diese sich somit dynamisch im realen Raum befindet und nicht immer dieselbige ist.

Es sei denn, es wird eine feste Position im realen Raum definiert an der die Anwendung gestartet werden muss. Hier bietet sich der Ursprungspunkt des SmartTrack Systems an $(KS)_{camZ}$ sofern dieser im Raum und nicht an der Kamera definiert wird und im tragenden Zustand der HL2 erreichbar ist.

Für eine dynamische Lösung der Problematik wurde ein Ansatz entwickelt, der beim Start der Anwendung die reale Position der HL2 über das SmartTrack-System erfasst und diese in der Anwendung über die Ausgabe des ART-Unity-Plugins speichert. Die Differenz zwischen dem Anwendungs-Ursprung

$(KS)_{Unity}$ und den HL2-Koordinaten $(KS)_{HL2}$, transformiert über das ART-Plugin in Unity, bildet die Transformationsmatrix ${}^{HL2}T_{Unity}$ für die Positionen der über das SmartTrack verfolgten Objekte ($(KS)_{Carrier}$ und $(KS)_{US-Target}$) zu ihren Repräsentanten in der HL2-Applikation ($(KS)_{Canvas}$ und $(KS)_{Painter}$).

Aufgrund des bisherigen Ergebnis wird vermutet, dass ${}^{HL2}T_{Unity}$ fehlerhaft ist, da sich jeweils ein reales Objekt und dessen digitaler Repräsentant nur in einem Punkt in der Anwendung gleichen und von diesem aus anschließend in unterschiedliche Richtungen wandern. Dieses Verhalten geschieht nicht willkürlich, da sich bei linearer Führung des realen Objekts sich die Richtung des digitalen Objekts ebenfalls linear verändert, jedoch ausschließlich in eine andere Richtung. Es müssen nun Antworten auf folgende zwei Fragen ermittelt werden:

1. Um welchen Winkel ändern sich die Positionen?
2. Warum ändern sich die Winkel um den ermittelten Wert?

Die beiden Fragen konnten jedoch weder bis zum Zeitpunkt der Versuchsdurchführung noch bis zur Publikation dieser Arbeit ermittelt werden und benötigen daher einer zukünftigen Beantwortung.

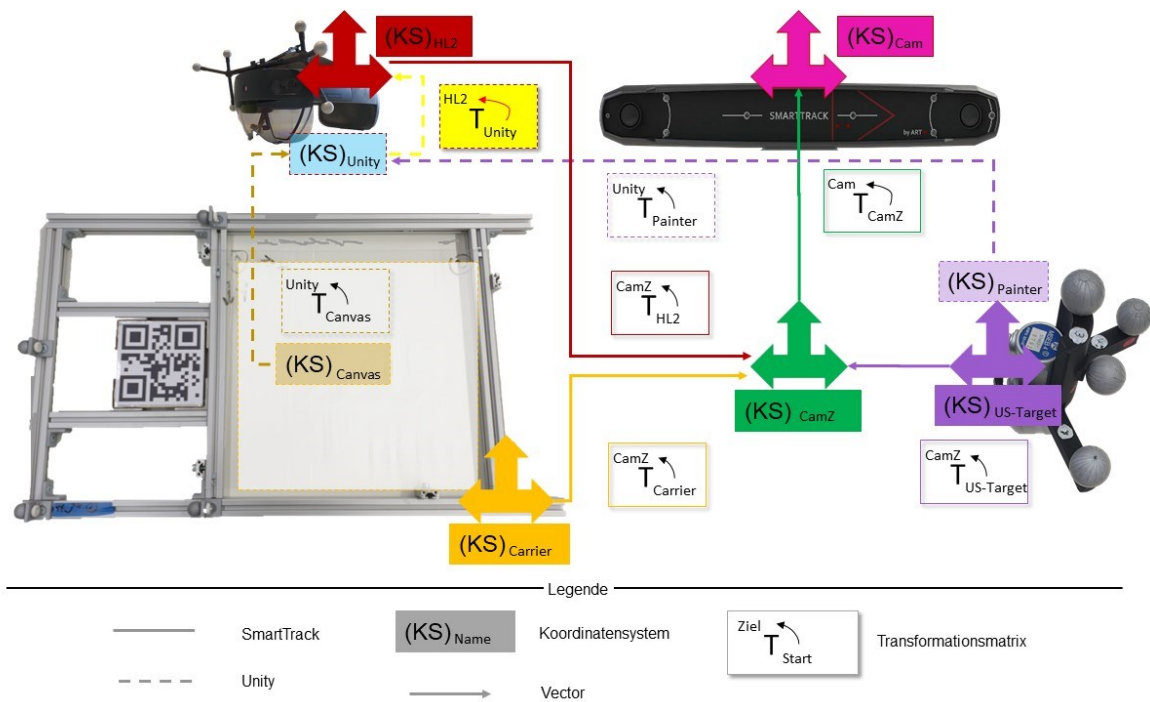


Abb. 44: Schematische Übersicht der KS in DTrack und Unity. Eine geeignete Transformation (Gelb) muss noch ermittelt werden, damit die Koordinaten aus dem Smarttrack-System in der Unity-Applikation richtig verortet werden.

7. Versuchsprotokoll

Auf Grundlage der bisher erläuterten Kausalitäten werden nachfolgend zwei Versuchsreihen entworfen und durchgeführt, um die entwickelten „Visualisierungen lokalisierter Ultraschalldaten am inspizierten Prüfteil in AR mit Hilfe der HoloLens 2“ zu analysieren. Hierbei wird zuerst die Präzision des Trackings in der Anwendung untersucht, bevor das Tracking in Verbindung mit den Messergebnissen analysiert wird.

Analyse der Präzision des Systems

Bei der ersten Untersuchung wird die generelle Präzision des Systems analysiert. Hierbei werden vier definierte Strecken von unterschiedlicher Länge (1 cm, 3 cm, 5 cm und 10 cm) mit dem realen US-Sensorkopf per Hand geführt und in der virtuellen HL2-Applikation (Systemaufbau wie in Abschnitt 3.2 beschrieben) aufgezeichnet (siehe Abb. 45). Gleichzeitig werden die Start- und Endkoordinaten (P1 u. P2) des virtuellen Sensorkopfs mit Hilfe des Holographic-Remoting aus Unity heraus protokolliert. Die entstehende Zeichnung während der Messung wird als JPEG-Datei abgespeichert und mit der Software „Inkscape“ und dessen Messwerkzeug kontrolliert, welches die Bild-Pixel in Millimeter umrechnet. Folgende Aspekte sind bei den auftretenden Ergebnissen zu berücksichtigen:

- Die reale Messtrecke wurde per Hand ausgemessen und definiert, dabei wurde der Durchmesser vom Sensorkopf (1,5 cm) berücksichtigt.
- Die Kalibrierung vom HL2-System zum SmartTrack-System ist händisch erfolgt.
- Zu dem Algorithmus zur Umrechnung von Pixeln in eine Längeneinheit durch die Software Inkscape sind keine wissenschaftlichen Analysen vorhanden. Ebenso müssen die Messpunkte in der Software händisch eingetragen werden.
- Für eine quantifizierte Bestätigung der Ergebnisse sind weitere Versuchsreihen nötig.

Wie Tab. 7 zeigt sind mit Mess-Differenzen von durchschnittlich 0.46 cm (≈ 5 mm) zwischen den realen Bedingungen und dem HL2-System zu rechnen. Durch eine Analyse mit Inkscape können Differenzen in der Auswertung von durchschnittlich 0.76 cm (≈ 8 mm) auftreten.

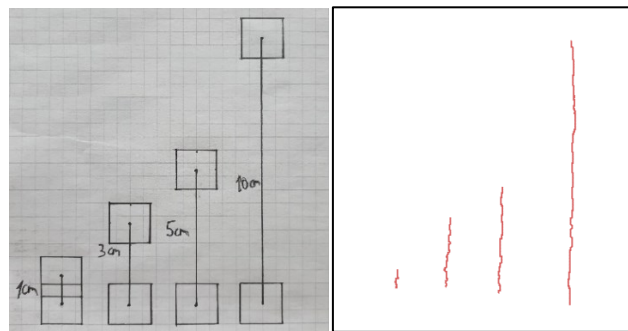


Abb. 45: Analyse Nr.1. Reale Strecken (L.) und digitale Aufnahme (R.) über den US-Sensor.

M.Strecke #	Strecke Real (cm)	Strecke HL2_App (cm)	Differenz Real-HL2	Strecke Inkscape (cm)	Differenz Real-Inkscape
1	1	0,79	0,21	0,72	0,28
2	3	2,66	0,34	2,41	0,59
3	5	4,27	0,73	3,69	1,31
4	10	9,43	0,57	9,17	0,83
Avr.-Dif:			0,46		0,76

M.-Strecke #	P1: X1 (Start)	P1: Y1 (Start)	P2: X2 (Ende)	P2: Y2 (Ende)	Distanz d (cm)
1	21,01	-15,48	21,08	-14,70	0,79
2	23,18	-15,49	23,37	-12,84	2,66
3	25,48	-15,71	25,66	-11,44	4,27
4	28,58	-16,11	28,63	-6,68	9,43

Tab. 7: Messergebnisse der vier Strecken

$$\text{Formel: } d := \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

3

Analyse der Visualisierungen

Bei der zweiten Versuchsreihe wird eine simple PMMA-Struktur untersucht, an dessen Rückseite eine 2 mm tiefe Fräsung des DLR-Logos erfolgte. Die Struktur ist 30 cm breit, 20 cm lang und 4.85 mm tief/dick. Für einen Abgleich zwischen der digitalen Aufnahme und dem realen Prüfteil wird das reale Bauteil in 22 Strecken unterteilt und händisch ausgemessen (siehe Abb. 46). Diese Strecken werden jeweils nach Beendigung der Aufnahme in der jeweiligen digitalen Textur manuell gesucht und mit der Software Inkscape kontrolliert. Bei den Ergebnissen (Tab.9.2.1) kann es sich ebenfalls nur um eine Annäherung handeln, da ähnliche Bedingungen wie im ersten Versuch vorherrschen:

- Die realen Messtrecken werden per Hand ausgemessen.
- Die Kalibrierung von HL2-System zum SmartTrack-System erfolgt händisch.
- Zu dem Algorithmus zur Umrechnung von Pixeln in eine Längeneinheit durch die Software Inkscape sind keine wissenschaftlichen Analysen vorhanden. Ebenso müssen die Messpunkte in der Software händisch eingetragen werden.
- Für eine quantifizierte Bestätigung der Ergebnisse sind weitere Versuchsreihen nötig.

Dabei ist zu erwähnen, dass nur eine physische Messung erfolgt, die digital protokolliert wird. Diese Daten werden anschließend zur Darstellung von fünf verschiedenen Visualisierungen verwendet. Somit haben alle Visualisierungen dieselbe Daten-Grundlage. Sie unterscheiden sich demnach lediglich von ihrer programmierten Logik zur Daten-Analyse und Bedeutungen der Farben.

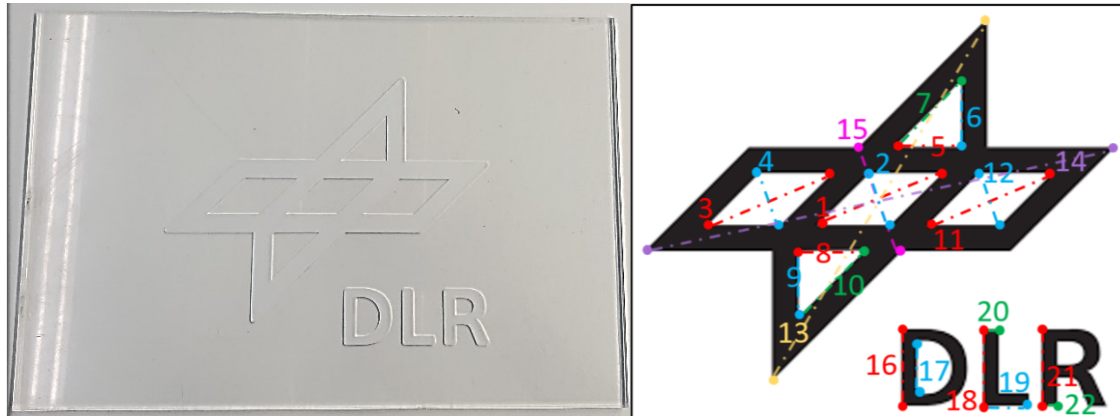
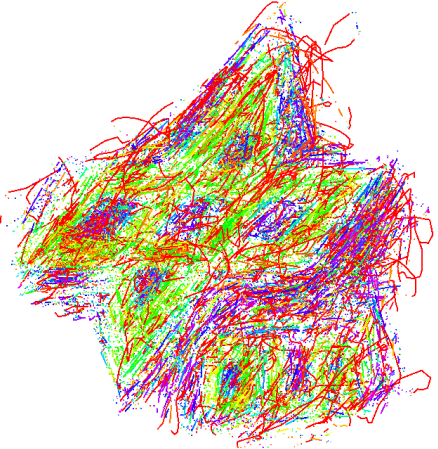







Abb. 46: Analyse Nr. 2. Reale PMMA-Struktur (L.) und die definierten Messtrecken (R.)

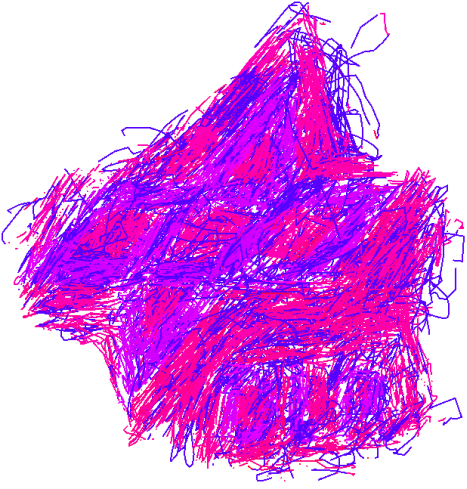

Insgesamt fallen die durchschnittlichen Differenzen von 6.2 mm bis 7.6 mm in den Bereich einer Ungenauigkeit innerhalb des Systems und dessen Überprüfung, die bereits im ersten Versuch festgestellt wurde (zwischen 5 mm bis 8 mm). Jedoch ergeben sich Abweichungen von bis zu 3 cm an Strecke 14, dessen Ursache in diesem Versuchsablauf nicht identifiziert werden konnte und durch weitere Aufnahmen überprüft werden müssen. Die Strecken 13 und 15 weisen keine derartige Abweichung auf, weshalb eine Streckung in der Breite der Visualisierungen an Strecke 14 stattfindet.



Nachfolgend werden die einzelnen Visualisierungen zusätzlich aufgezeigt und analysiert:

Tab.7.1: C-Bild	Amplituden-Werte (C) (Einheit Volt)
	 0 0.3 0.5 0.8
	Min C: 0 V Max C: 0.87 V
	Min C Skala: 0 V Max C Skala: 0.8 V
Die Fräsung des DLR-Symbols ist trotz des Mess-Rauschens (rot), die durch fehlendes Kopplungsmittel während der Aufnahme erzeugt werden, mit C-Werten im Bereich um 0.3 V zu sehen. Die PMMA-Struktur scheint unbeschadet Werte um 0.5 V aufzuzeigen. Das Bild weist die niedrigsten Mess-Differenzen von durchschnittlich 6.2 mm auf.	

Tab.7.2: C-Bild mit D-Daten	C-Werte (Volt) Tiefen-Werte (D) (Einheit Millimeter)
	 0 0.3 0.5 0.8 0 5
	Min C: 0 V Max C: 0.87 V
	Min C Skala: 0 V Max C Skala: 0.8 V
	Min D: 0 mm Max D: 4.85 mm
Min D Skala: 0 mm Max D Skala: 5 mm	
<p>Die Fräsung des DLR-Symbols ist zwar zu erkennen, jedoch ist das Rauschen verstärkt, da vermutlich die Farbanalyse zum Zeichnen der Werte zu lange benötigt. Dies würde eine Erklärung sein, weshalb die größte durchschnittliche Differenz von ≈ 8 mm zum realen Bauteil in dieser Visualisierung entsteht. Die Grauwerte, die die D-Daten repräsentieren, sorgen zudem für keinen erkennbaren Mehrwert.</p>	

Tab.7.3: C-Bild mit Kontrollbereich	US-Messwerte Zustände
	 „Im Bereich“ „Über dem Bereich“ „Unter dem Bereich“
	Min C: 0 V Max C: 0.87 V
	Min C Skala: 0 V Max C Skala: 0.8 V
	Control Value: 0.6 V Control Range: 0.15 V
<p>Die Fräsung des DLR-Symbols ist zwar zu erkennen, jedoch ist der Kontrollbereich (0.45 V – 0.75 V) zu niedrig eingestellt und das Signal-Rauschen wird „unter dem Bereich“ eingestuft, weshalb verortbare Differenzen von durchschnittlich 7 mm auftreten und weniger Fläche vom unbeschädigten Material eingezeichnet sind. Der „DLR“-Schriftzug ist daher auch nicht deutlich zu erkennen. Dies kann vermutlich durch eine Erhöhung der „Control Range“ auf 0.2 V behoben werden.</p>	

Tab.7.4: D-Bild	Tiefen-Werte (D) (Einheit Millimeter)
	 0 3.3 4.85 5
	Min D: 0 mm Max D: 4.85 mm Min D Skala: 0 mm Max D Skala: 5 mm
	<p>Die Fräsung des DLR-Symbols ist auch beim D-Bild zu erkennen. Die Dicke der PMMA-Struktur wird mit 4.85 mm (und „Mess-Fehlern“ von bis zu 3.3 mm in blau) angezeigt. Die Fräsung wird mit Tiefen-Werten von 3.6 mm bis 3.3 mm angegeben, was eine Ungenauigkeit von ≈ 1 mm entspricht, da die reale Fehlstelle bei maximal 2.85 mm von der Oberfläche entfernt liegt. Ursachen für diese Abweichung können in der Kalibrierung des US-Sensors liegen.</p> <p>Die Darstellungsungenauigkeit wird, wie beim C-Bild, auf durchschnittlich ≈ 6 mm angegeben.</p>

Tab.7.5: D-Bild mit Kontrollbereich	US-Messwerte Zustände
	 „Im Bereich“ „Über dem Bereich“ „Unter dem Bereich“
	Min D: 0 mm Max D: 4.85 mm Min D Skala: 0 mm Max D Skala: 5 mm Control Value: 4.5 mm Control Range: 0.35 mm
	<p>Die Fräsung des DLR-Symbols ist mit Hilfe des entsprechenden Kontrollbereichs (4.15 mm- 4.85 mm) klar zu erkennen. Daneben werden die als „Messfehler“ eingestuft Striche in Lila eingezeichnet. Der Wert der durchschnittlicher Abweichung zur realen Position liegt bei 6,3 mm ..</p>

Schlussfolgerungen

Insgesamt lässt sich auf Grundlage der entstehenden Visualisierungen zu diesem Projektstand bereits eine Analyse des Prüfteils mit primitiver Struktur, welche aus homogenem Material besteht, führen. Die Tiefenwerte werden dabei bis auf 1 mm genau angegeben, was sich aus den vorgegebenen Daten ermitteln lässt. Für die Genauigkeit der Amplituden-Werte müsste eine gesonderte Überprüfung z.B. in Form einer stationären Prüfanlage oder mit Hilfe eines Linear-Array-Rollenprüfkopfes erfolgen, die in diesem Rahmen des Projekts bisher nicht möglich war.

Die „reinen“ C- bzw. D-Bilder zeigen eine Darstellungsgenauigkeit in der Position von durchschnittlich 6 mm. Mit der jeweiligen Einbeziehung eines Kontroll-Bereichs erhöht sich dieser Wert auf 7 mm. Die Analyse eines „zusätzlichen Werts“ sorgt für eine Ungenauigkeit von 8 mm. Zwar liegen alle Differenzen in dem Bereich der Ungenauigkeit des Systems und dessen Auswertung (5mm – 8mm), jedoch wird bei der gruppierten Betrachtung der Ergebnisse („Rein“, „Kontrollbereich“, „Zusätzlicher Wert“), der Eindruck erweckt, dass der jeweilige Anstieg um 1 mm mit dem steigenden Aufwand der Analyse und dessen Programmierung zusammenhängt. Dazu müssten aber zusätzliche Versuchsreihen stattfinden, um signifikante Daten zu erhalten. Unabhängig davon müsste, bei einer Bestätigung der quantifizierten Messungenauigkeit des Systems, diese auf ein Minimum von 1 mm reduziert werden. Dies kann ggf. durch einen automatisierten Kalibrierungsprozess sowie einem programmierten Prüfverfahren in Unity erfolgen.

Die Einbeziehung eines zusätzlichen Wertes für die Visualisierung, die in Graustufen ausgedrückt wird und zu dem Hauptwert hinzuaddiert wird, hat keinen erkennbaren Mehrwert. Im Gegenteil, durch die zusätzliche Kalkulation entsteht eine Verzögerung im Analyse-Prozess, weshalb es für das System schneller und einfacher ist, einen Messfehler zu zeichnen als den korrekten Farbwert zeitkritisch zu aktualisieren. Diese Darstellungsform kann demnach entfernt werden.

Eine Reduzierung der Werte sowie eine klare Benennung des Zustands, wie es bei den Bildern mit einem Kontrollbereich erfolgt, kann dazu führen das vorliegende Material schneller zu „durchblicken“. Voraussetzung hierzu sind jedoch exakt eingestellte Werte in der Skala, die ein vorheriges Wissen über das vorliegende Prüfteil erfordern.



Abb. 47: Durchführung des Laborversuchs Nr.2.

8. Fazit und Ausblick

Fazit

Es ist die Vorarbeit für ein System erfolgt, welches die zerstörungsfreie Prüfung mittels Ultraschalls digitalisiert und die Einbindung eines digitalen Zwillings in Form eines 3D-Modells ermöglicht. Die Position eines Ultraschallkopfes wird mit einem markerbasierten Tracking-Verfahren verfolgt, um so die aufgenommenen Ultraschalldaten mit Ortsinformationen zu verknüpfen. Gleichzeitig ist eine prüfteilbezogene Visualisierung mit Hilfe der Datenbrille HoloLens 2 entstanden, die ein zusätzliches Wiederabrufen von gespeicherten Informationen und die deckungsgleiche Anzeige am Prüfteil ermöglicht, um die Messergebnisse zuverlässiger interpretieren zu können. Hierfür musste vorab eine Einarbeitung in die Grundlagen zerstörungsfreier Prüfungsmethoden mittels Ultraschalls und eine intensive Betrachtung der Tracking-Systeme für xR-Technologien erfolgen. Zusätzlich musste ein Verständnis für die bereits bestehende Hardware- und Softwarestruktur geschaffen werden, wodurch anschließend eine Einarbeitung in die Kommunikationswege zwischen den verschiedenen Komponenteneinheiten erfolgen konnte. Erweitert entstanden neue Visualisierungsideen, die bewertet und daraufhin für den Einsatz in der entstandenen HL2-Anwendung geplant, entwickelt und in einer Testumgebung implementiert sowie anschließend analysiert wurden. Zudem entstanden Ansätze einer vollständigen Integration der HoloLens 2 in ein externes, markerbasiertes, Outside-In Trackingsystems.

Es wurde erkannt, dass, solange eine HL2-Anwendung auf die internen Tracking-Mechanismen und Algorithmen der HL2 angewiesen ist, um den virtuellen Inhalt relativ zur eigenen Position im realen Raum darzustellen, eine Schnittstelle entwickelt werden muss, um externe Tracking-Varianten zusätzlich mit einbeziehen zu können. Dies ist bei dem Einsatz von externen, markerbasierten Tracking-Methoden ebenso wichtig wie für die Implementation von zukünftigen Trackingmethoden, wie beispielsweise bei einem markerlosen Tracking. Dabei ist das markerlose Tracking hilfreich, um einen direkten Bezug zum Prüfteil herstellen zu können ohne ein Referenz-Objekt, wie den entwickelten Part-on-Carrier, verwenden zu müssen.

Ein besseres Nutzungsverhalten der Anwendung ließe sich mit Anpassungen erzielen, die sich auf das Absichern einer verwendeten Farbskala auf der entstandenen Bilddatei, ein dynamisches importieren von virtuellen Zwillingen in Form von 3D-Modellen, der nachträglichen Einsicht der Sensorwerte an einer bestimmten Mess-Stelle am Bauteil sowie der dynamischen Anpassung von Skalen-Werten auf das überprüfte Bauteil beziehen. Zudem müsste der Einsatz anderer Prüfköpfe, wie z.B. einem „Phased Array“ getestet und dessen auftretende Datenströme analysiert werden, um die Anwendung ggf. darauf anzupassen.

Daneben befindet sich die Anwendung durch ihre Menüführung und manuellen Prozesse zur Kalibrierung noch in der Entwicklungsphase

Ausblick

Während der Bearbeitung dieses Projekts haben sich Themenfelder erschlossen, die eine weiterführende Erforschung erfordern.

Zum einen ist die Reduzierung der Ultraschall-Rohdaten zu nennen, weshalb die präsentierte HL2-Anwendung kein A- und B-Bild generieren kann. Hier müsste eine Möglichkeit entwickelt werden, wie diese Daten dennoch zum Gerät gelangen, ohne das Leistungsverhalten der Anwendung zu beeinträchtigen und eine Darstellung dieser Methoden dennoch zu ermöglichen.

Gleichzeitig ist die dynamische Generierung einer 3D-Darstellung der Oberflächenstruktur des Prüfteils für zukünftige Anwendungen interessant, um den realen Sachverhalt des Prüfteils detailgetreuer im digitalen Kontext für das Prüfpersonal anzubieten. Ob eine detaillierte Innenaufnahme wie bei CT- oder MRT-Bildern durch Voxel nötig ist, müsste analysiert werden und wie diese Bilder entsprechend auch mit dem Ultraschallverfahren zur Laufzeit der Anwendung umgesetzt werden könnten. Hinzu müsste der virtuelle „Blick“ in das Prüfteil durch die stärkere Einbeziehung der realen Struktur und digitalen, „gestalterischen“ Darstellungen zur Unterstützung der menschliche Wahrnehmung für das Tiefenempfinden einbezogen werden, um die US-Messergebnisse im Bauteil erkennbar aufzuzeigen. Die Bereitstellung der Anwendung ist zudem ein essentielles Thema, welches betrachtet werden muss. Mit dem Auftreten von „WebAR“ [Sar19] bietet sich die Möglichkeit die Rechenlast von der HL2 zu nehmen, die bei der Positionskalkulierung und der Analyse der US-Sensordaten entsteht und sie auf einen Rechenserver zu verlagern. Dadurch kann die Anwendung gleichzeitig global abrufbar gemacht werden, anstelle, dass die Anwendung lokal auf einem Gerät betrieben werden muss. Dies sorgt zudem für den Gedanken einer Einbeziehung von „Cross-Plattform“- und „Multi-User“- Gedanken, bei denen verschiedenste (xR)-Bedienegeräte verwendet werden können, um denselben virtuellen Inhalt abzurufen und in einer gemeinsamen Arbeitsumgebung zu agieren. Eine Person kann dadurch beispielsweise ein Tablet verwenden und die Messergebnisse betrachten, während eine andere Person den US-Sensor führt und gleichzeitig die HL2 verwendet.

Als letzten Punkt ist die Einbeziehung einer KI zu nennen, die unterstützend zum Prüfpersonal während einer Prüfung das Bauteil interpretiert und Aussagen über die Verortung und Art einer Fehlstelle trifft und die Unterscheidung zwischen tatsächlichen Mängeln und strukturbezogenen Materialeigenschaften vorab Fällen kann [Vir21].

Abbildungsverzeichnis

Abb. 1: Schematische Darstellung des vorangestellten Systemaufbaus mit Realbildern ohne einheitlichen Maßstab.....	14
Abb. 2: Aufbau eines SE-Prüfkopfes [PolU]	17
Abb. 3 (L): Beispielhafte Darstellung eines Reflektogramms (A-Bild) mit 5 verschiedenen Testfällen. [Chr15, S.9].....	17
Abb. 4 (R): Beispiel eines handgeführten SE-Prüfkopfes mit einem Echograph mit A-Bild. [KDP21, S.1]	17
Abb. 5: Darstellungsmethoden. Logarithmisches HF-Bild (L.), A-Bild (M.) mit den absoluten Amplitudenwerten und B-Bild (R.), was pro Einheit in x- Richtung jeweils ein A-Bild darstellt, welches um 90° nach rechts gedreht wurde [PolB]	18
Abb. 6: C-Bild Darstellung. Beispiel einer Matrix aus Amplitudensummen der A-Bilder (L.), Matrix in einer Farbcodierung (M.) und die geglättete Darstellung der Farbmatrix (R.) [PolC]	18
Abb. 7 (L.): Schematische Darstellung Impuls-Echo-Verfahren (O.) und Durchschallverfahren (U.). Jeweils bei Direktankopplung (L.), mittels Luftultraschall (M.) und Tauchbad-Technik (R) [PolIE].	19
Abb. 8 (R.): Variationen von Ultraschall-Prüfköpfen und deren bildgebende Eigenschaft in Anlehnung an [JKV18, S.290]	19
Abb. 9: HoloLens 2 Explosionsdarstellung [Mic20].....	20
Abb. 10: Die 6 Freiheitsgrade der 3 senkrechten Achsen	21
Abb. 11: Schema des Outside-In- Tracking (L.) und Inside-Out Tracking (R.) [Pag16, S.2].....	22
Abb. 12: Beispiel für xR-Tracking „Punktgebilde“ in Form von passiven Markerbäumen [ART20]..	23
Abb. 13: Passive Marker als „Muster“. Ein QR-Code (L.), Grafik (M.) & Muster als 3D Würfel (R.) [Pai14, S.7]	23
Abb. 14: SLAM Visualisierung mit der OpenCV library (L.) [Alm19]. Spatial Mapping der HoloLens (R.) [Uni21].	24
Abb. 15: Markerlose Objekterkennung mittels des VisionLib SDK's [Vis18].	24
Abb. 16: Das Konzept des optischen Trackings mit Hilfe einer Stereo-IR-Kamera.....	25
Abb. 17: Patentdarstellung [Hol17, S.1]	26
Abb. 18: Konzeptionelle Visualisierung von C-Bildern einer CFK Struktur in verschiedenen AR-Umgebungen zur Vorarbeit des Patents „Hol17“ [Mey15, S.]	26
Abb. 19 (O.): Setup der AR-Applikation (L.) und Schematischer Aufbau (R.) [Dep18, S.4, S.5].	27
Abb. 20 (U.): (L.) AR Anwendung von [Sch_2, S.5]. (R.) Visualisierung von [Las14, S.6]	27
Abb. 21: Systemaufbau 3D-SmartInspect [Fra21].....	28
Abb. 22: Morphologischer Kasten zur Planung der eingesetzten Hardwarestrategie mit zusätzlicher Farbkennzeichnung der vier erarbeiteten Lösungswege.....	30
Abb. 23: Schematische Darstellung des erweiterten Systemaufbaus mit Realbildern ohne einheitlichen Maßstab.....	34
Abb. 24: Schematische Softwarestruktur im Gesamtkontext des Projekts	36
Abb. 25: LabVIEW Anwendung für das Projekt. Roter Kasten: PCUS-Einstellungen, Lila Kasten: A-Bild mit (Schwarz: Rohdaten und Blau: Auswertungsbereich), Rosa Kasten: ToF und maximale Amplitude aus dem A-Bild.	38
Abb. 26: Übersicht der Koordinatensysteme zur Ermittlung der Position vom Ultraschallsensor zum Referenz-Koordinatensystem (KS)Edge.	39
Abb. 27: Schematische Darstellung der HL2-Applikation.....	42

Abb. 28: Ausrichtung der Koordinatensysteme von ROS (L.) und dem QR-Marker in Unity (R.).....	43
Abb. 29: QR-View und QR-Code	43
Abb. 30: „US-Painter View“ mit erweiterter Darstellung (Skala (L.) und Painter-Radius (R.))	44
Abb. 31: Schematische Darstellung einer Annäherung der Koordinatensysteme von ROS und des QR-Markers	44
Abb. 32: ROS-Daten-Array in der „Connection & Sensor Data View" (L.) (Grün: Koordinaten, Braun: Tiefenwert, Rot: Ultraschalldaten-Header) und erweiterter Darstellung. (Gelb).....	45
Abb. 33: UI-Darstellung „Color Picker" im <i>Full-Range-Mode</i> (L.) und <i>Contrast-Mode</i> (R.).....	46
Abb. 34: UI zur farblichen Bestimmung des Kontroll-Zustandes im <i>Full Range Mode</i> (L.) und im <i>Contrast Mode</i> (R.)	47
Abb. 35: UI zur farblichen Bestimmung der Kontroll-Zustände im <i>Full Range Mode</i> (L.) und im <i>Contrast Mode</i> (R.)	47
Abb. 36: Virtualisierte Eingabemethoden. Numerische Tastatur (L.) und US-Controller (R.).....	48
Abb. 37: Zusammenhang von Punkten (L.), Polygonen (M.) und Voxel (R.)	51
Abb. 38: Polygonnetze mit unterschiedlicher, globaler Struktur [Rad16, S.150].	52
Abb. 39: Vernetzungsbeispiele einer stringerversteiften Haut [Rad16, S.145].....	52
Abb. 40: Visualisierungskonzept eines rechteckigen Polygonnetzes mit US-Sensor-Farb- und Tiefenwerten	53
Abb. 41: Visualisierungskonzept zur Generierung eines 3D-US-Messpfads in AR.....	53
Abb. 42: UV-Map eines sechsseitigen Würfels mit eingezeichneten Bereichsabtrennungen	54
Abb. 43: Ansatz zur Fusionierung von Voxel- und Polygonsalgorithmen nach [Paul8] in einer eigenen HL2 Anwendung.....	55
Abb. 44: Schematische Übersicht der KS in DTrack und Unity. Eine geeignete Transformation (Gelb) muss nochermittelt werden, damit die Koordinaten aus dem Smarttrack-System in der Unity-Applikation richtig verortet werden.	57
Abb. 45: Analyse Nr.1. Reale Strecken (L.) und digitale Aufnahme (R.) über den US-Sensor.	58
Abb. 46: Analyse Nr. 2. Reale PMMA-Struktur (L.) und die definierten Messtrecken (R.)	60
Abb. 47: Durchführung des Laborversuchs Nr.2.	63

Tabellenverzeichnis

Tab. 3.1: Die elf Interrogativsätze dieser Thesis zur Erfüllung der Ausgangsforderung	31
Tab. 7.0: Messergebnisse Strecken.....	59
Tab. 7.1: C-Bild.....	61
Tab. 7.2: C-Bild mit D-Daten.....	62
Tab. 7.3: C-Bild mit Kontrollbereich.....	62
Tab. 7.4: D-Bild.....	63
Tab. 7.5: D-Bild mit Kontrollbereich.....	64
Tab. 9.2.1: Messergebnisse DLR-Logo.....	75

Quellenverzeichnis

- [Alm19] Almubarak, Joody: S.L.A.M and Tracking. Vom 30.04.2019. Letzter Aufruf am 04.12.2020 von: <https://medium.com/@joodyalmubarak/s-l-a-m-and-tracking-technology-6d1eab3d8a78>
- [ART19] Advanced Realtime Tracking GmbH: *DTrack3 User's Guide - ARTtrack , TRACKPACK, SMARTTRACK & DTrack*. Version 3.0.1. 2019 A.R.T. GmbH
- [ART20] Advanced Realtime Tracking GmbH: *Targets*. Letzter Aufruf am 04.12.2020 von: <https://ar-tracking.com/products/markers-targets/targets/>
- [Bog14] Boger, Yuval: *Overview of Positional Tracking Technologies for Virtual Reality*. Gastbeitrag. Road to VR, 2014
- [Chr15] Christian-Albrechts-Universität zu Kiel: *Grundpraktikum Versuchsreihe: Materialwissenschaft Ultraschall B40*. Vom 27.03.2015. Letzter Aufruf am 04.12.2020 von: <https://www.tf.uni-kiel.de/servicezentrum/neutral/praktika/anleitungen/b401.pdf>
- [Deu97] Deutsch, Volker: *Ultraschallprüfung: Grundlagen und industrielle Anwendungen*. Springer (Berlin, Heidelberg), 1997
- [Dep18] Deppe, Robert; et al.: *Augmented reality for supporting manual non-destructive ultrasonic testing of metal pipes and plates*, Conference: Workshop Proceedings / Tagungsband: Virtuelle und Erweiterte Realität – 15. Workshop der GI-Fachgruppe VR/AR, Düsseldorf, Germany, 2018
- [Fli21] Flick, Jasper: *Mesh Deformation*; Letzter Aufruf am 12.04.2021 von: <https://catlikecoding.com/unity/tutorials/mesh-deformation/>
- [Fra21] Homepage – Fraunhofer-Institut; Aufgerufen am 24.03.2021 von: <https://www.materials.fraunhofer.de/de/Geschaeftsfelder/Mobilitaet/3d-smartinspect---intelligence-in-inspection-and-quality-control.html>
- [Fr221] Homepage – Fraunhoferinstitut; Letzter Aufruf am 24.03.2021 von: <https://www.izfp.fraunhofer.de/content/dam/izfp/de/documents/aktuelles/2017/3D-SmartInspect-de.pdf>
- [Hof19] Hoffmann, Daniel; et al.: *Vergleich zerstörungsfreier Prüfmethode zur Ermittlung von charakteristischen Defekten in faserverstärkten Kunststoffen*. (DGZfP 2019) Deutsche Gesellschaft für Zerstörungsfreie Prüfung Jahrestagung 2019, Friedrichshafen, Germany
- [Hol17] Holtmannspötter, Jens, Dr.; et al.: *Aktenzeichen 10 2016 003 543.9*. Deutsches Patent- und Markenamt. <https://register.dpma.de/DPMAREgister/pat/PatSchrifteneinsicht?docId=DE102016003543A1&page=1&dpi=150&lang=de>, 2017
- [Hug13] Hughes, John F.; et al.: *Computer Graphics - Principles and Practice Third Edition*. Addison-Wesley (Boston, Massachusetts, USA), 2013

- [JKV18] Jenderka, Klaus-Vitold; Delorme, Stefan: *Diagnostischer Ultraschall*. In: Schlegel W., Karger C., Jäkel O. (eds) *Medizinische Physik*. Springer Spektrum, Berlin, Heidelberg, 2018. https://doi.org/10.1007/978-3-662-54801-1_10
- [KDP21] Karl Deutsch Prüf- und Messgerätebau GmbH + Co KG: *Echograph 1095 Digitales Ultraschallprüfgerät*. Letzter Aufruf am: 12.03.2021 von: https://www.karldeutsch.de/produkt/kd_echograph_1095/
- [Lan15] Langlois, Pierre; et al.: *A system and method of non-destructive inspection with a visual scanning guide*. Aktenzeichen US10156548B2. European Patent Office. <https://worldwide.espacenet.com/patent/search?q=pn%3DUS10156548B2, 2015>
- [Las14] Lasagni, Fernando; et al.: *C-Scan Ultrasonic Generation using Wireless Encoder based on Passive Markers*. 6th International Symposium on NDT in Aerospace, 12-14th November 2014, Madrid, Spain
- [Liv13] Livingston, Mark A.; et al.: *Pursuit of "X-Ray Vision" for Augmented Reality*. In: Huang W., Alem L., Livingston M. (eds) *Human Factors in Augmented Reality Environments*. 2013. Springer, New York, NY. https://doi.org/10.1007/978-1-4614-4205-9_4
- [Mey15] Meyer, Jean; et al.: *Visualisation Of Ultrasonic Testing Data Using Augmented Reality*. 7th International Symposium on NDT in Aerospace, 2015, Bremen, Deutschland.
- [Mic17] Microsoft: *Second version of HoloLens HPU will incorporate AI coprocessor for implementing DNNs*. Vom 23.07.2017. Letzter Aufruf am 18.11.2020 von: <https://www.microsoft.com/en-us/research/blog/second-version-hololens-hpu-will-incorporate-ai-coprocessor-implementing-dnns/>
- [Mic20] Microsoft: *Hololens 2 hardware*. Vom 20.10.2020. Letzter Aufruf am 18.11.2020 von: <https://docs.microsoft.com/en-us/hololens/hololens2-hardware>
- [Mic_2] Microsoft: *Aufgabenbasiertes asynchrones Programmiermodell*. Vom 19.08.2020. Letzter Aufruf am 10.04.2021 von: <https://docs.microsoft.com/de-de/dotnet/csharp/programming-guide/concepts/async/task-asynchronous-programming-model>
- [Mic_3] Microsoft: *Schreiben einer Holographic Remoting-Remote-App mithilfe der holographic-space-API*. Vom 01.12.2020. Letzter Aufruf am 10.04.2021 von: <https://docs.microsoft.com/de-de/windows/mixed-reality/develop/platform-capabilities-and-apis/holographic-remoting-create-remote-wmr>
- [Mic_4] Microsoft: *Übersicht über das Windows Device Portal*. Vom 08.01.2021. Letzter Aufruf am 10.04.2021 von: <https://docs.microsoft.com/de-de/windows/uwp/debug-test-perf/device-portal>
- [Pag16] Pagani, Alain; et al.: *Sensors for location-based Augmented Reality - the example of Galileo and EGNOS*, International Society of Photogrammetry and Remote Sensing, ISPRS 2016, 2016, Prag, Tschechische Republik.
- [Pai14] Pai, Yun Suen; et al.: *Augmented reality-based programming, planning and simulation of a robotic work cell*, Institution of Mechanical Engineers, IMechE, 2014.

- [Pau18] Paulo, Soraia F; et al.: *Volume and Surface Rendering of 3D Medical Datasets in Unity* ®, Project: IT-MEDEX - Interactive Tablets for Collaborative Scenarios Related to 3D Medical Image Exploration, 2018
- [PolB] Polymer Service GmbH Merseburg: *B-Bild-Technik*. Letzter Aufruf am 09.03.2021 von: <https://wiki.polymerservice-merseburg.de/index.php/B-Bild-Technik>
- [PolC] Polymer Service GmbH Merseburg: *C-Bild-Technik*. Letzter Aufruf am 09.03.2021 von: <https://wiki.polymerservice-merseburg.de/index.php/C-Bild-Technik>
- [PolU] Polymer Service GmbH Merseburg: *Ultraschall-Sender (S)-Empfänger (E)-Prüfköpfe*. Letzter Aufruf am 09.03.2021 von: [https://wiki.polymerservice-merseburg.de/index.php/Ultraschall-Sende\(S\)-Empf%C3%A4nger\(E\)-Pr%C3%BCfk%C3%B6pfe](https://wiki.polymerservice-merseburg.de/index.php/Ultraschall-Sende(S)-Empf%C3%A4nger(E)-Pr%C3%BCfk%C3%B6pfe)
- [PolIE] Polymer Service GmbH Merseburg: *Ultraschall-Impuls-Echo-Technik*. Letzter Aufruf am 09.03.2021 von: <https://wiki.polymerservice-merseburg.de/index.php/Ultraschall-Impuls-Echo-Technik>
- [Rad16] Raddatz, Florian; *Lokalisierung der Interaktionsorte von Lambwellen in komplexen Faserverbundstrukturen*. Dissertation, Fakultät für Maschinenbau der Technischen Universität Carolo-Wilhelmina, 2016, Braunschweig, Deutschland
- [Ros19] Rosales, Carlos S.; et al.: *Distance Judgments to On- and Off-Ground Objects in Augmented Reality*. 2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR), Osaka, Japan, 2019, pp. 237-243, doi: 10.1109/VR.2019.8798095.
- [Sar19] Saracco, Roberto: *Are you ready for WebAR?* Vom 20.07.2019. Letzter Aufruf am 17.04.2021. Von: <https://cmt.ee.ieee.org/futuredirections/2019/07/20/are-you-ready-for-webar/>
- [Sch19] Schönheits, Manfred: *Augmented Reality Based Visual Force Feedback For Physical Human-Robot Interaction*. International Conference on Robotics and Automation 2019, ICRA 2019, 2019, Montreal, Kanada
- [Sch_2] Schickert, Martin; et al.: *Augmented-Reality-Visualisierung von Abbildungsergebnissen an Betonbauteilen – Implementierung und Ausblick*. 4. Fachseminar des FA MTHz, Mikrowellen- und Terahertz-Prüftechnik in der Praxis, (DGZfP MTHz 2019), 2019, Fürth, Deutschland
- [Sco09] Scott, Gebhardt T.; et al.: *Polygons, Point-Clouds, and Voxels, a Comparison of High-Fidelity Terrain Representations*. Simulation Interoperability Workshop and Special Workshop on Reuse of Environmental Data for Simulation—Processes, Standards, and Lessons Learned, 2009.
- [Uni20] Homepage- Unity Documentation: *Scale and units*. Letzter Aufruf am 14.04.2021 von: <https://docs.unity3d.com/2019.3/Documentation/Manual/BestPracticeMakingBelievableVisuals1.html>
- [Uni21] Homepage- Unity Documentation: *HoloLens Spatial Mapping*. Letzter Aufruf am 01.04.2021 von: <https://docs.unity3d.com/es/2019.2/Manual/SpatialMapping.html>

- [Vir21] Virkkunen, Iikka; et al.: *Augmented Ultrasonic Data for Machine Learning*. Journal of Nondestructive Evaluation volume 40, Article number: 4. 02.01.2021
- [Vis18] VisionLib: #LavaVirtual. Vom 04.04.2018. Letzter Aufruf am 01.03.2021 von <https://twitter.com/visionlib/status/981484435345375232>
- [Win16] Windows Mixed Reality Developer Forum: *Physical origin of HL coordinate system?* Vom xx.03.2016. Letzter Aufruf am 13.04.2021 von: <https://forums.hololens.com/discussion/878/physical-origin-of-hl-coordinate-system>
- [Zec19] Zec, Marin: *Gestaltung von anwenderfreundlichen Gruppenunterstützungssystemen und Kollaborationsprozessen für die Problemstrukturierung und Ideengenerierung auf Grundlage der Morphologischen Analyse*, Dissertation an der Technischen Universität München, 2019, München, Deutschland

9. Anhang

9.1 Berechnungen

Bestimmung der materialspezifischen Schallgeschwindigkeit

Die Schallgeschwindigkeit der im Abschnitt 6 verwendeten PMMA- (Plexiglas-) Platte wurde auf Grundlage der in der Wissenschaft angegebenen Werte gewählt. Hier wird allgemein eine Geschwindigkeit der Longitudinalwellen zwischen 2670 bis 2750 m/s für das Material angegeben [Deu97, S.156].

Mit Bezug auf Formel 4.3.2 wurde an der Stelle mit dem größten ToF-Wert folgendes Ergebnis berechnet:
 $0,01881015 = 2670 \cdot (970 + 439) \cdot 1e-08 / 2$

Dies entspricht einen Tiefenwert d von 18,8 mm. Da die getestete Platte jedoch nur 4,85 mm dick ist und c die einzige verstellbare Konstante außerhalb der Sensorwerte ist, wurde eine Umstellung zur Berechnung von c durchgeführt mit der Annahme eines Maximums von 0,00485 m an der Stelle mit dem größten ToF-Wert. Daraus ergibt sich folgende Umrechnung:

$$c = (d / ((SD + ToF) \cdot SR) \cdot 2$$

$$c = (0,00485\text{m} / (979 + 439) \cdot 1e-08) \cdot 2$$

$$c = (0,00485\text{m} / 0.00001409\text{ s}) \cdot 2$$

4

$$c = 688.43151171\text{ m/s}$$

$$c \approx 688\text{ m/s}$$

Eine material-spezifische Schallgeschwindigkeit von 688 m/s wird in der Literatur nur annäherungsweise zu dem Material „Kork“ mit Longitudinal-Werten zwischen 500 bis 535 m/s bestimmt [Deu97, S.155].

Somit könnte die verwendete PMMA-Platte Materialeigenschaften ähnlich zu Kork besitzen.

Eine fehlerbehaftete Einstellung des Ultraschallprüfsystems ist jedoch nicht auszuschließen, was allerdings für diesen Systemstand keine gewichtete Rolle darstellt, da die Werte letztendlich nur ein Mittel zum Testen der AR-Anwendung und dessen Visualisierung sind und hier keinen weiteren Zweck erfüllen.

9.2 Ergebnisse aus dem Versuchsprotokoll

Tab. 9.2.1: Messergebnisse DLR-Logo

M.-Strecke #	Real in mm	Normal Picture in mm		Control Picture in mm		C with D Data in mm		C/D Control Picture in mm		Normal Picture in mm		Control Picture in mm	
		C-Bild N	Differenz	C-Bild Con	Differenz	C-Bild & D	Differenz	C/D Con	Differenz	D-Bild N	Differenz	D-Bild Con	Differenz
1	49	30,3	18,7	30,3	18,7	32,1	18,7	29,4	19,6	35,3	13,7	34,2	14,9
2	20	16,2	3,8	14,8	5,2	16,7	3,3	15,4	4,6	19,9	0,1	19,5	0,5
3	49	29,0	20,0	28,3	20,7	29,4	19,6	29,4	19,6	36,8	12,2	37,4	11,6
4	20	17,0	3,0	15,2	4,8	15,3	4,7	14,8	5,2	19,2	0,8	18,1	1,9
5	25	14,3	10,8	12,8	12,2	10,7	14,3	11,5	13,5	13,7	11,3	15,5	9,5
6	24	14,6	9,4	14,4	9,6	16,3	7,7	18,2	5,8	18,6	5,4	17,6	6,4
7	34	19,2	14,8	18,8	15,3	19,4	14,6	20,5	13,5	23,1	10,9	22,6	11,4
8	24	11,5	12,5	12,1	11,9	10,3	13,7	12,2	11,8	15,6	8,4	15,9	8,1
9	24	17,8	6,2	15,3	8,7	15,1	8,9	14,2	9,8	20,8	3,2	18,9	5,1
10	34	19,6	14,4	19,7	14,3	19,5	14,5	20,3	13,7	23,8	10,2	23,5	10,5
11	49	29,8	19,2	31,2	17,8	32,0	17,0	29,7	19,3	35,8	13,2	34,7	14,4
12	20	17,3	2,8	17,0	3,0	17,0	3,0	16,2	3,8	18,7	1,3	18,3	1,7
13	146	157,5	-11,5	151,6	-5,6	148,7	-2,7	158,6	-12,6	135,3	10,7	135,7	10,3
14	167	145,3	21,7	144,8	22,2	143,8	23,2	145,9	21,1	136,3	30,7	142,1	24,9
15	41	45,7	-4,7	45,0	-4,0	45,5	-4,5	43,4	-2,4	41,7	-0,7	41,2	-0,2
16	28	30,7	-2,7	28,8	-0,8	25,5	2,5	26,4	1,6	27,6	0,4	26,2	1,8
17	20	13,8	6,3	16,4	3,7	13,7	6,3	12,3	7,7	16,2	3,8	16,2	3,8
18	28	28	-0,2	29,6	-1,6	25,3	2,7	30,0	-2,0	26,4	1,6	28,6	-0,6
19	16	17,4	-1,4	15,8	0,2	17,1	-1,1	13,3	2,7	14,9	1,1	13,0	3,0
20	6	7,6	-1,6	7,9	-1,9	6,7	-0,7	6,6	-0,6	5,2	0,8	6,2	-0,2
21	28	31,2	-3,2	27,3	0,7	26,6	1,4	28,6	-0,6	26,3	1,7	27,2	0,8
22	6	7,1	-1,1	6,2	-0,2	4,3	1,7	7,0	-1,0	5,4	0,6	6,1	-0,1
Avr-Differenz		6,2		7,0		7,6		7,0		6,4		6,3	
Avr-Differenz		C-Bild N		C-Bild Con		C-Bild & D		C/D Con		D-Bild N		D-Bild Con	
-0,775		0,2		0,2		0,7		0,6		0,1		0,1	
0,3		21,7		22,2		23,2		21,1		30,7		24,9	
Modes		Normal		Control									

9.3 Datenträger CD

9.4 Eigenständigkeitserklärung

Ich versichere, die vorliegende Arbeit selbstständig ohne fremde Hilfe verfasst und keine anderen Quellen und Hilfsmittel als die angegebenen benutzt zu haben. Die aus anderen Werken wörtlich entnommenen Stellen oder dem Sinn nachentlehnten Passagen sind durch Quellenangaben eindeutig kenntlich gemacht.

(Ort, Datum)

(Unterschrift)