



Master Thesis

im Rahmen des
Universitätslehrganges „Geographical Information Science & Systems“
(UNIGIS MSc) am Interfakultären Fachbereich für Geoinformatik (Z_GIS)
der Paris Lodron-Universität Salzburg

zum Thema

„Mixed Reality zur Visualisierung unterirdischer Leitungsnetze“

vorgelegt von

Nadja Ohse

105144, UNIGIS MSc Lehrgang 2018

Betreuer/in:
Prof. Dr. Josef Strobl

Zur Erlangung des Grades
„Master of Science (Geographical Information Science & Systems) – MSc (GIS)“

Tübingen, 24. August 2020

Danksagung

An dieser Stelle möchte ich mich bei allen bedanken, die mich bei der Erstellung dieser Masterarbeit unterstützt und motiviert haben.

Zunächst möchte ich mich bei Prof. Dr. Strobl für die Betreuung und Begutachtung meiner Masterarbeit sowie für die konstruktiven Anmerkungen und die Kritik bedanken. Ein weiterer Dank geht an das iDEAS:lab des Z_GIS der Universität Salzburg für das Bereitstellen der Microsoft HoloLens, es hat mir das Verfassen der vorliegenden Arbeit und das Bearbeiten des Themas ermöglicht.

Ein besonderer Dank gilt der Firma BARAL Geohaus-Consulting AG für die Unterstützung während meines gesamten Masterstudiums und der Möglichkeit der Erstellung meiner Masterarbeit in diesem Unternehmen. Außerdem bin ich sehr dankbar über die Bereitstellung der für die Anfertigung der Arbeit notwendigen Hardware und Software durch die Firma BARAL Geohaus-Consulting AG, ohne die eine Beantwortung der These nicht möglich gewesen wäre.

Ebenfalls möchte ich mich bei der FairNetz GmbH bedanken, die mir die Versorgungsnetzdaten des Untersuchungsgebietes in Reutlingen zur Verfügung gestellt hat. Außerdem möchte ich der Stadt Reutlingen für das Bereitstellen eines Geländemodells des Untersuchungsgebietes danken. Ohne diese Daten wäre eine praktische Untersuchung nicht möglich gewesen.

Ein weiterer großer Dank gebührt dem UNIGIS-Team für die tatkräftige Unterstützung während des Studiums und der wertvollen Anregungen bei der Erstellung der Masterthesis.

Eidesstattliche Erklärung

Hiermit erkläre ich, dass ich die vorliegende Master Thesis selbständig verfasst habe.

Ich habe keine anderen als die angegebenen Quellen und Hilfsmittel benutzt. Alle Aussagen dieser Arbeit, die wörtlich oder sinngemäß aus anderen Werken übernommenen wurden, sind unter Angabe der Quelle kenntlich gemacht.

Diese Arbeit ist noch nicht veröffentlicht und war weder vollständig noch in wesentlichen Teilen Gegenstand eines anderen Prüfungsverfahrens.

Montag, 24. August 2020

Unterschrift: _____

Zusammenfassung

Für die Funktionsfähigkeit der Gesellschaft, die Gesundheit der Bevölkerung sowie eine intakte wirtschaftliche Tätigkeit ist der störungsfreie Betrieb der Infrastruktur für Wasser, Gas, Strom, Fernwärme und Telekommunikation unerlässlich. Um den störungsfreien Betrieb zu gewährleisten, wird der Verlauf der unterirdisch liegenden Leitungen dokumentiert und für den Einsatz im Außendienst in zweidimensionalen Plänen dargestellt. Da diese zweidimensionalen Pläne von der Realität abstrahiert sind, müssen sie vor Ort anwendungsspezifisch und sachkundig interpretiert werden. Diese mentale Transformation erfordert ein hohes Verständnis über den Raum und die Generalisierung auf der Karte und kann daher nur durch ausgebildetes und erfahrenes Fachpersonal erfolgen. Das zusätzliche Risiko der Fehlinterpretation kann zu Ausfällen und erheblichen Schäden führen. Entwicklungen im Bereich der Visualisierungstechnologien in Form von Mixed Reality haben das Potential, diese Risiken durch die Verknüpfung von Realität und Virtualität in einer physischen Umgebung zu reduzieren.

Die vorliegende Arbeit knüpft an diese Problematik an und beantwortet die Leitfrage, ob die bildgebende Technologie der Mixed Reality für die Visualisierung von Leitungsnetzen angewendet werden kann und der Nachweis in Form eines Prototyps möglich ist. Dadurch soll eine Vereinfachung bei der vor-Ort-Erkennung von Leitungsnetzen erreicht werden, um Kosten zu senken und die Sicherheit zu erhöhen.

Zur Beantwortung der These waren mehrere Schritte notwendig. Nach der Grundlagenermittlung über rechtliche Rahmenbedingungen, unterschiedliche potenzielle Visualisierungstechnologien und satellitengestützte Ortung erfolgte ein Literaturüberblick zum Stand der Forschung bezüglich der Nutzbarkeit von Mixed Reality zur Visualisierung unterirdischer Leitungsnetze. Ein anschließender Vergleich der Technologien führte zu der Auswahl der ersten Generation der Microsoft HoloLens für die Entwicklung des Prototyps. Diese Vorarbeit bildete die Grundlage für das Pflichtenheft, auf dessen Basis die Entwicklung des Prototyps erfolgte. Dabei wurden zweidimensionale Ausgangsdaten eines Leitungsnetzes sowie ein digitales Geländemodell genutzt. Die Ausgangsdaten wurden mithilfe der Softwareprogramme ArcGIS Pro, FME und CityEngine in ein 3D-Modell transformiert. Die anschließende Entwicklung des Prototyps unter Anbindung eines externen GPS-Gerätes erfolgte mithilfe der Entwicklungsumgebungen Unity und Visual Studio.

Abschließende Testfälle prüften und bewerteten den Prototyp anhand des Pflichtenheftes. Das erste Testszenario wurde dabei ohne Anbindung des GPS-Gerätes mit fest definierten Standortwerten des Betrachters durchgeführt, um die grundsätzliche Funktionalität der Applikation zu überprüfen. Im Ergebnis konnte festgestellt werden, dass die Anforderungen im Pflichtenheft bis auf eine Einschränkung erfüllt werden konnten. Die Einschränkung lag darin, dass die Blickrichtungs- und Positionsunabhängigkeit beim Starten der Anwendung durch die Standortfestlegung des Betrachters nicht möglich war. Das zweite Testszenario beinhaltete die Anbindung des GPS-Empfängers. Durch erhebliche Ausfälle bei der Positionsbestimmung wurden die Ergebnisse des zweiten Testszenarios als nicht-repräsentativ eingestuft.

Die Ergebnisse haben gezeigt, dass die bildgebende Technologie der Mixed Reality für die Visualisierung von Leitungsnetzen angewendet werden kann und der Nachweis in Form eines Prototyps möglich ist. Dieser Prototyp konnte die geforderten Funktionen erfüllen. Lediglich bei der praktischen GPS-Anbindung ist im Rahmen dieser Arbeit keine finale Aussage abzuleiten.

Abstract

The functional capability and economies of modern societies as well as public health highly depend on an infrastructure that secures undisturbed supply of gas, water, electricity, district heating and telecommunication. In order to ensure this objective, underground pipelines are documented and visualized in two-dimensional plans for on-site application. These 2D-plans represent an abstraction of reality and require an individual interpretation by respective experts. This mental transformation is based on a high level of individual competence regarding surroundings and the way data is generalized on the map. As a result, only qualified and proficient experts can perform this task. Additionally, there is a high risk of misinterpretation which can cause failures, damages and injuries. Recent developments in the area of visualization in the form of mixed reality offer the potential to reduce these risks by combining reality and virtuality in a given physical environment.

The present master thesis addresses this issue by answering the key question, if the imaging technology of mixed reality can be applied to visualize pipelines and if the development of a prototype of an application can be used as a proof. The purpose of this task is to simplify pipeline identification and interpretation on-site in order to reduce costs and increase security.

Several steps had to be followed to answer the thesis. The first step was to collect the relevant basics of legal conditions, visualization technologies and satellite-based positioning. The second step was a literature review about the state of research regarding the usability of mixed reality for underground visualization. The subsequent comparison of the available technologies resulted in the decision to use the first generation of the Microsoft HoloLens for the development of the prototype. Based on this prework, the specification for the prototype was developed and used as guideline for the development. A two-dimensional pipeline network and a digital terrain model provided the underlying source data. The 3D-Model was generated based on the source data with the help of the software programs ArcGIS Pro, FME and CityEngine. The subsequent development of the prototype with connection to a GPS-device took place by using the development environments Unity and Visual Studio.

Finally, test scenarios verified and evaluated the prototype based on the previously defined specification. The first test scenario was conducted without the use of the GPS-device with a fixed, predefined position of the user. The objective was to verify the basic functionality of the application. The results of the first test scenario proved that the requirements of the specification could be met with only one exception. This exception is the inability to start the application from a randomly selected position and with a freely chosen direction of the visual field. The second test scenario included the connection to the GPS-device. The test cases in this scenario showed substantial failures in determining the position, hence, the results of this scenario are were classified as non-representative.

The conclusion is that the imaging technology of mixed reality can be applied for pipeline visualization and the proof with a prototype is possible. This prototype fulfilled the required functions, only with respect to the usage of the GPS-device a final conclusion within the framework of this study is not possible.

Inhaltsverzeichnis

Danksagung.....	I
Eidesstattliche Erklärung.....	II
Zusammenfassung.....	III
Abstract.....	IV
Inhaltsverzeichnis.....	V
Abbildungsverzeichnis.....	VII
Tabellenverzeichnis.....	IX
Abkürzungsverzeichnis.....	X
1 Einleitung.....	1
1.1 Problemstellung und Relevanz des Themas.....	1
1.2 Zielsetzung und Forschungsfragen.....	3
1.3 Struktur der Arbeit.....	4
2 Grundlagen.....	5
2.1 Rechtliche Rahmenbedingungen.....	5
2.1.1 Allgemein.....	5
2.1.2 Strom.....	6
2.1.3 Gas und Wasser.....	7
2.2 Visualisierungstechnologien.....	10
2.3 Satellitengestützte Ortung und GPS.....	14
3 Aktueller Forschungsstand.....	16
4 Technologien im Vergleich.....	22
4.1 Anforderungen.....	22
4.2 Vergleich.....	24
5 Methodik.....	28
5.1 Vorgehensweise.....	28
5.2 Untersuchungsgebiet.....	30
5.3 Hardware und Software.....	33
5.3.1 Endgeräte.....	33
5.3.2 Software zur Datenaufbereitung.....	36
5.3.3 Entwicklungsumgebungen.....	36
5.4 Pflichtenheft.....	40
5.5 Datenaufbereitung.....	42
5.6 Entwicklung.....	45

5.7	Ergebnisse	58
6	Diskussion.....	65
7	Ausblick	67
	Literatur	68
	Anhang	73
	Anhang 1: Muster einer manuellen Aufnahmeskizze der Sparte Strom	73
	Anhang 2: Muster einer digitalen Aufnahmeskizze der Sparte Strom	74
	Anhang 3: Muster eines Bestandsplans Strom in Mehrstrichdarstellung	75
	Anhang 4: Muster eines Übersichtsplans Strom	76
	Anhang 5: Mindestanforderungen für die grafische Ausprägung der Pläne der Sparte Strom	77
	Anhang 6: Muster einer manuellen Aufnahmeskizze für Gas und Wasser	79
	Anhang 7: Muster einer digitalen Aufnahmeskizze für Gas und Wasser	80
	Anhang 8: Inhalte der Aufnahmeskizze für Gas und Wasser	81
	Anhang 9: Muster eines Bestandsplanes für Gas und Wasser.....	82
	Anhang 10: Muster eines Übersichtsplanes für Gas und Wasser	84
	Anhang 11: grafische Vorschriften für Gas und Wasser	86
	Anhang 12: Flussdiagramm über die Datenaufbereitung	88
	Anhang 13: Flussdiagramm über die Entwicklung	89
	Anhang 14: Gesamter Scriptcode	90
	Anhang 15: Flussdiagramm zur Funktionsweise des Skriptes zur Positionierung des Objektes	93
	Anhang 16: Rechtliche Hinweise	94

Abbildungsverzeichnis

Abbildung 1: Ziel der Arbeit.....	2
Abbildung 2: Virtuality Continuum (VC)	10
Abbildung 3: Optische Durchsicht Abbildung 4: Kameradurchsicht.....	11
Abbildung 5: Unterschiede zwischen VR, AR und MR	13
Abbildung 6: Vorgehensweise der Methodik	29
Abbildung 7: Ermittlung des Untersuchungsgebietes	30
Abbildung 8: Untersuchungsgebiet	31
Abbildung 9: Abgrenzung des Untersuchungsgebietes	31
Abbildung 10: Karte des Untersuchungsgebietes mit Leitungen	32
Abbildung 11: Die HoloLens im Einsatz	34
Abbildung 12: Die HoloLens und ihre Komponenten	34
Abbildung 13: Das Blumax GPS-710	35
Abbildung 14: Der NMEA-Satz GGA.....	35
Abbildung 15: Aufbau der Oberfläche des Programms Unity.....	38
Abbildung 16: Oberfläche von Visual Studio	39
Abbildung 17: Einstellungen des Transformers 3DForcer	43
Abbildung 18: Einstellungen des Transformers Bufferer	43
Abbildung 19: Gesamter FME-Workbench	44
Abbildung 20: Ausgewählter Strang der Sparte Wasser.....	45
Abbildung 21: In Unity importierter Wasserstrang mit seinen Einstellungen in der Inspector View....	46
Abbildung 22: Einstellungen der Main Camera	47
Abbildung 23: Code-Ausschnitt: Definition der Variablen im Unity-Editor.....	48
Abbildung 24: Code-Ausschnitt: Bluetooth-Anbindung und Datenempfang im Unity-Editor.....	49
Abbildung 25: Code-Ausschnitt: Assemblies für Windows Runtime.....	50
Abbildung 26: Code-Ausschnitt: Bluetooth-Anbindung und Datenempfang für Windows Runtime Unterstützung	50
Abbildung 27: Formel zur Berechnung der Distanz	51
Abbildung 28: Code-Ausschnitt: Bestimmung der Position des Objektes.....	52
Abbildung 29: Code-Ausschnitt: Positionierung des Objektes für den Unity-Editor.....	53
Abbildung 30: Positionierung des Objektes für Windows Runtime Unterstützung	54
Abbildung 31: Build-Einstellungen im Unity-Editor	54
Abbildung 32: Einstellungen in der AppManifest.xml	55
Abbildung 33: Debugging-Einstellungen in Visual Studio	55
Abbildung 34: Angepasstes Skript für erstes Testszenario	56
Abbildung 35: Startposition für das Testen der App.....	58
Abbildung 36: Screenshots aus der App: geringe Entfernung zum Betrachter.....	59
Abbildung 37: Screenshots aus der App: größere Entfernung zum Betrachter	59
Abbildung 38: Screenshot aus der App: größere Entfernung zur Startposition.....	60
Abbildung 39: Manuelle Aufnahmeskizze Strom.....	73
Abbildung 40: Digitale Aufnahmeskizze Strom.....	74
Abbildung 41: Bestandsplan Strom in Mehrstrichdarstellung	75
Abbildung 42: Übersichtsplan Strom 1:5000	76
Abbildung 43: Manuelle Aufnahmeskizze Gas und Wasser	79
Abbildung 44: Digitale Aufnahmeskizze Gas und Wasser.....	80
Abbildung 45: Bestandsplan Gas, 1:500	82

Abbildung 46: Bestandsplan Wasser, 1:500	83
Abbildung 47: Übersichtsplan Gas, 1:5000.....	84
Abbildung 48: Übersichtsplan Wasser, 1:5000.....	85
Abbildung 49: Flussdiagramm zur Datenaufbereitung	88
Abbildung 50: Flussdiagramm zur Entwicklung	89
Abbildung 51: Flussdiagramm zur Funktionsweise des Skriptes zur Positionierung des Objektes	93

Tabellenverzeichnis

Tabelle 1: Anforderungen und technische Anforderungen an die Technologie	24
Tabelle 2: Erfüllung der technischen Anforderungen nach Visualisierungstechnologie	27
Tabelle 3: Pflichtenheft	40
Tabelle 4: Ergebnisbewertung basierend auf dem Pflichtenheft.....	61
Tabelle 5: Bewertung der Ergebnisse von Testszenario 2	63
Tabelle 6: Bestandteile und Inhalte der Netzdokumentation Strom, Teil 1.....	77
Tabelle 7: Bestandteile und Inhalte der Netzdokumentation Strom, Teil 2.....	77
Tabelle 8: Symbol- und Signaturkatalog Strom	78
Tabelle 9: Inhalte einer Aufnahmeskizze Gas und Wasser	81
Tabelle 10: Gemeinsame grafische Vorschriften für Gas und Wasser	86
Tabelle 11: Zeichenvorschrift Gas	87
Tabelle 12: Zeichenvorschrift Wasser.....	87

Abkürzungsverzeichnis

ACAAR	Authoring for Context-Aware Augmented Reality
AR	Augmented Reality
ARGIS	Augmented Reality geografisches Informationssystem
ARIOT	Augmented Reality Internet Of Things
ARPV	Augmented Reality Pipeline Visualizer
AV	Augmented Virtuality
CAD	Computer Aided Design
COM-Port	Communication Port
DAE	Digital Asset Exchange
DGM	Digitales Geländemodell
DHDN	Deutsches Hauptdreiecksnetz
DIN	Deutsches Institut für Normung
DVGW	Deutscher Verein des Gas- und Wasserfaches e.V.
DWG	Drawing
DXF	Drawing Exchange Format
EGNOS	Geostationary Navigation Overlay Service
ETRS89	European Terrestrial Reference System 1989
FBX	Filmbox
FME	Feature Manipulation Engine
GB	Gigabyte
GDB	Geodatabase
GGA	Global Positioning System Fix Data, Time, Position and fix related data for a GPS receiver
GIS	Geografisches Informationssystem
GK	Gauß-Krüger

GLL	Geographic Position - Latitude/Longitude
GLONASS	Globalnaya Navigatsionnaya Sputnikovaya Sistema
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GSA	GPS DOP and active satellites
GSV	Satellites in View
HPU	Holographic Processing Unit
IoT	Internet of Things
IoTDI	Internet-of-Things Design and Implementation
LGLN	Landesamt für Geoinformation und Landesvermessung Niedersachsen
MR	Mixed Reality
NMEA	National Marine Electronics Association
OBJ	3D Object Format
RMC	Recommended Minimum Navigation Information
SAPOS	Satellitenpositionierungsdienst
SBAS	Satellite-based Augmentation Systems
SDCM	System for Differential Corrections and Monitoring
TIF	Tagged Image File Format
TPM	TextPad Macro Format
UTM	Universal Transverse Mercator
UWP	Universal Windows Platform
VDE	Verband der Elektrotechnik, Elektronik und Informationstechnik e.V.
VR	Virtual Reality
VTG	Track made good and Ground speed
WAAS	Wide Area Augmentation System
3DS	3D Studio Max Graphics Format

1 Einleitung

1.1 Problemstellung und Relevanz des Themas

„Notwendige Voraussetzungen für die Funktionsfähigkeit des menschlichen Gemeinwesens und städtebaulicher Weiterentwicklungen sind leistungsfähige Leitungsnetze, die eine sichere Versorgung mit Wasser, Gas, Fernwärme, Elektrizität und Telekommunikation sowie eine umweltfreundliche und sichere Entsorgung, d.h. Sammlung und Ableitung des Abwassers, gewährleisten“ (STEIN, 2003).

Mit diesem Zitat fasst Prof. Dr.-Ing. D. Stein die Bedeutung eines störungsfreien Betriebes von Ver- und Entsorgungsinfrastrukturen zusammen. Eine Störung der Infrastruktur für Wasser, Gas, Strom, Fernwärme und Telekommunikation kann innerhalb kürzester Zeit zu massiven Einschränkungen führen. Diese Einschränkungen haben erhebliche Auswirkungen auf die Funktionsfähigkeit der Gesellschaft, die Gesundheit der Bevölkerung sowie die Intaktheit der wirtschaftlichen Tätigkeit.

Zur Sicherstellung der Errichtung und des störungsfreien Betriebes der entsprechenden Leitungsnetze haben sich im Laufe von Generationen umfangreiche Regeln, Anordnungen und Gesetze entwickelt. Ein Großteil dieses Regelwerks bezieht sich auf die Erstellung, Aktualisierung und Nutzung der entsprechenden Dokumentation. Die Art der Dokumentation basiert heute weitgehend auf digitalisierten Daten, die über entsprechende Softwareprogramme zur Bearbeitung und Nutzung bedarfsgerecht visualisiert werden. Im Vergleich zu den früheren, analogen Plänen (DEUTSCHES INSTITUT FÜR NORMUNG, 1975) findet heute eine Trennung zwischen Datenspeicherung und -bearbeitung auf der einen Seite sowie der Visualisierung auf der anderen Seite statt. Die Datenspeicherung erfolgt mithilfe von Datenbanksystemen. Die unterschiedlichen Nutzer, zum Beispiel Planer, Vermesser, Projektierer, etc., greifen auf diese Datenbank zu und bereiten die jeweiligen Daten mithilfe von entsprechenden Softwareprogrammen visuell auf. Für den Einsatz im Außendienst erfolgt die Aufbereitung in Form eines zweidimensionalen Planes unter Berücksichtigung der entsprechenden Normen und Vorschriften. Als Medium kommen sowohl Papier als auch elektronische Geräte wie Tablets zum Einsatz. Die zweidimensionalen Pläne sind daher von der Realität abstrahiert und müssen vor Ort anwendungsspezifisch und sachkundig interpretiert werden. Dies erfolgt über eine mentale Transformation. Sie erfordert ein hohes Verständnis über den Raum und die Generalisierung auf der Karte (SCHALL ET AL., 2009). Die mentale Transformation kann aus diesem Grund nur durch entsprechend ausgebildetes Fachpersonal erfolgen. Zusätzlich ist ein nicht zu unterschätzendes Risiko der Fehlinterpretation vorhanden, das zu teilweise erheblichen Ausfällen und Schäden führen kann.

Die jüngste Entwicklung im Bereich der Visualisierungstechnologien hat das Potential, diese Risiken zu vermeiden, indem Realität und Virtualität in einer physischen Umgebung miteinander verknüpft werden (HUANG ET AL., 2016). Diese Verknüpfung erfolgt, indem die reale Umgebung mithilfe eines Darstellungsmediums durch virtuell dargestellte Objekte ergänzt wird. Die Daten der virtuellen Objekte werden digital vorliegenden Liegenschaftsdaten entnommen und in ein Bild der realen Umgebung projiziert, siehe Abbildung 1.

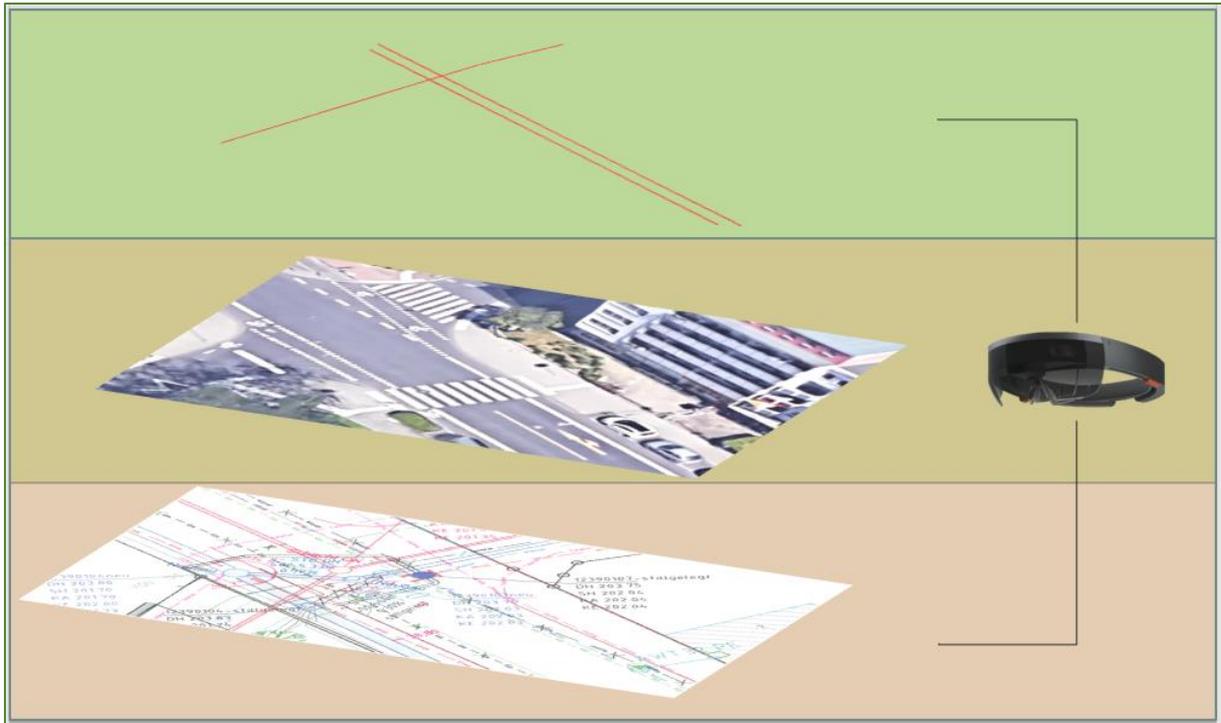


Abbildung 1: Ziel der Arbeit

Quelle: eigene Darstellung nach WEIß INGENIEURE, GOOGLE INC, CLEANPNG.COM (2018)

1.2 Zielsetzung und Forschungsfragen

Aus der oben geschilderten Problematik ergibt sich die Leitfrage, ob die bildgebende Technologie der Mixed Reality für die Visualisierung von Leitungsnetzen angewendet werden kann und ob der Nachweis in Form eines Prototyps möglich ist. Der Zweck dieser Anwendung besteht darin, eine deutliche Vereinfachung bei der vor-Ort-Erkennung von Leitungsnetzen zu erreichen, um Kosten zu senken und die Sicherheit zu erhöhen. Daraus leiten sich die folgenden operativen Teilfragen ab:

1. Welche Grundlagen sind für die Visualisierung von unterirdischen Leitungsnetzen und den Einsatz von Mixed Reality zu berücksichtigen?
2. Ist bei dem derzeitigen Forschungsstand ein Nutzen der Mixed Reality bei der Darstellung von Leitungsnetzen erkennbar?
3. Welche der Technologien für Mixed Reality eignen sich für die Darstellung von unterirdischen Leitungsnetzen?
4. Kann ein Prototyp erstellt werden, der einen unterirdischen Leitungsstrang unter Anwendung von Mixed Reality ausreichend visualisiert?
5. Sind die Testergebnisse der Anwendung aussagekräftig, um die Leitthese zu beantworten?

Die erste Forschungsfrage soll die allgemeinen Anforderungen klären, die für einen Einsatz von Mixed Reality zur Darstellung von Leitungsnetzen zu berücksichtigen sind. Diese Anforderungen bilden die Grundlage zur Auswahl einschlägiger Technologien sowie zur Spezifikation des Pflichtenheftes. Dabei werden die gegebenenfalls einzusetzenden Technologien näher beschrieben. Die zweite Forschungsfrage ermittelt den Stand der Forschung zum Einsatz von Mixed Reality. Hierbei wird geprüft, inwieweit vorliegende Erkenntnisse Rückschlüsse auf die Nutzbarkeit von Mixed Reality zur Darstellung von Leitungsnetzen zulassen. Dies untersucht, ob Gründe vorliegen, welche die Nutzung von Mixed Reality grundsätzlich ausschließen. Die dritte Forschungsfrage hat zum Ziel, die für den Prototyp auszuwählenden Technologiekomponenten zu vergleichen und auszuwählen. Die vierte Forschungsfrage beinhaltet die Abgrenzung des Untersuchungsgebietes und beschreibt die Vorgehensweise der Entwicklung. Dazu gehört neben der Erstellung des Pflichtenheftes die Aufbereitung der Daten, die Durchführung der Entwicklungsarbeit sowie die Definition und Durchführung der Testszenarien. Im letzten Teil dieser Arbeit fasst die fünfte Forschungsfrage die Testergebnisse im Vergleich zu den Spezifikationen des Pflichtenheftes zusammen und ermittelt, inwieweit die erzielten Ergebnisse die These bestätigen oder widerlegen können.

1.3 Struktur der Arbeit

Die vorliegende Arbeit gliedert sich in sieben Kapitel. Das erste Kapitel besteht aus der Definition der Problemstellung und dessen Relevanz, der Formulierung von Zielsetzung und Forschungsfragen sowie der Darstellung der Struktur dieser Arbeit.

Im zweiten Kapitel sind die Grundlagen zu ermitteln, dazu gehören die rechtlichen Rahmenbedingungen, die derzeit verfügbaren Mixed Reality Technologien und die satellitengestützte Positionierung. Zunächst werden die einschlägigen rechtlichen Regeln auf den Einsatz hinsichtlich der Visualisierung und Dokumentation von Leitungsnetzen herausgearbeitet. Dabei handelt es sich um die anerkannten Regeln der Technik durch den Verband der Elektrotechnik, Elektronik, Informationstechnik e.V. (VDE) und den Deutschen Verein des Gas- und Wasserfaches e.V. (DVGW), deren Einhaltung gesetzlich vorgeschrieben ist. Im Anschluss werden die derzeit verfügbaren Mixed Reality Technologien sowie die Grundlagen der satellitengestützten Positionierung dargestellt.

Das dritte Kapitel beinhaltet die Untersuchung wissenschaftlicher Veröffentlichungen zu Mixed Reality Anwendungen in diversen Forschungsfeldern. Die Ergebnisse der Untersuchung sollen zu Erkenntnissen der Nutzbarkeit von Mixed Reality zur Visualisierung unterirdischer Leitungsnetze führen.

In Kapitel vier werden die wesentlichen Mixed Reality Technologien miteinander verglichen. Für diesen Vergleich wird ein Anforderungsprofil erstellt, dessen Inhalte sich aus den Ergebnissen der vorangegangenen Kapitel ableiten. Auf dieser Basis wird die Technologie zur Entwicklung des Prototyps ausgewählt.

Kapitel fünf dokumentiert die Methodik und Durchführung zur Entwicklung des Prototyps. Dazu wird zunächst das Untersuchungsgebiet klar definiert sowie die für die Entwicklung und Anwendung benötigten Hardware- und Softwarekomponenten ermittelt. Im Anschluss erfolgt die Erstellung des Pflichtenheftes zur Dokumentation der Anforderungen an den Prototyp. Hier werden Rahmenbedingungen, technische Spezifikationen, Kernfunktionen, Qualitätsanforderungen, Anforderungen an die Benutzeroberfläche und die Bedienbarkeit sowie Spezifikationen der Testszenarien beschrieben. Die einzelnen Entwicklungsschritte bestehen aus Datenaufbereitung und -übertragung, Entwicklung der Anwendung sowie der Installation des Prototyps auf dem Endgerät. Zur Überprüfung der These werden Testszenarien definiert und deren Resultate mit den im Pflichtenheft spezifizierten Anforderungen verglichen.

In Kapitel sechs werden die Vorgehensweise und die Ergebnisse diskutiert, sodass die These bestätigt oder widerlegt werden kann. Die vorliegende Arbeit schließt in Kapitel sieben mit einem Ausblick der vorliegenden Arbeit unter dem Gesichtspunkt möglicher und notwendiger weiterer Forschungen ab.

2 Grundlagen

2.1 Rechtliche Rahmenbedingungen

2.1.1 Allgemein

Die Notwendigkeit über das Vorhandensein einer qualitativen Dokumentation der technischen Daten sowie der räumlichen Lage eines Versorgungsnetzes ist in gesetzlichen Vorgaben und Richtlinien verankert. Zwar existieren keine unmittelbaren gesetzlichen Vorgaben (SCHEU, 2015), jedoch steht im Energiewirtschaftsgesetz durch §49, Absatz 1, dass Energieanlagen so eingerichtet und betrieben werden müssen, „dass die technische Sicherheit gewährleistet ist. Dabei sind vorbehaltlich sonstiger Rechtsvorschriften die allgemein anerkannten Regeln der Technik zu beachten“ (BUNDESAMT FÜR JUSTIZ, 2015). In §49 Absatz 2 wird beschrieben, von welchen technischen Regeln hier die Rede ist. Für Elektrizität sind das die technischen Regeln des Verbandes der Elektrotechnik, Elektronik, Informationstechnik e.V. (VDE) und für Gas die technischen Regeln der Deutschen Vereinigung des Gas- und Wasserfaches (DVGW) (BUNDESAMT FÜR JUSTIZ, 2015).

Im vorletzten Absatz des §49 des Energiewirtschaftsgesetzes wird zudem festgelegt, dass Betreiber von Energieanlagen auf Verlangen der nach Landesrecht zuständigen Behörde Auskünfte über technische und wirtschaftliche Verhältnisse geben müssen (BUNDESAMT FÜR JUSTIZ, 2015). Diese Auskunftspflicht wird ohne eine adäquate Netzdokumentation vernachlässigt. Außerdem wird Absatz 2 des §49 (BUNDESAMT FÜR JUSTIZ, 2015) ohne eine den Regeln der Technik entsprechende Dokumentation verletzt, denn die technische Sicherheit kann dann nicht gewährleistet werden. Um Unfälle und Personenschäden zu vermeiden, benötigen Bauunternehmen das Wissen über die Lage und die technischen Eckdaten der Netzbestandteile.

Damit die gesetzlichen Vorgaben eingehalten werden können, gibt es sowohl durch den VDE als auch durch den DVGW Regelwerke zur Netzdokumentation, welche als anerkannte Regeln der Technik von den Versorgungsunternehmen eingehalten werden müssen. Dabei wird als Netzdokumentation ein „konsistentes Führen von Daten in geografischen Informationssystemen und die grafische Präsentation von Leitungsnetzen, einschließlich der erforderlichen Datenerfassung“ definiert (DVGW, 2010; VDE, 2010). Die Netzdokumentation ist nach den anerkannten Regeln der Technik also dreigeteilt und beinhaltet die Datenerfassung, die Datenspeicherung und die entsprechende Visualisierung.

Die Grundlage der Netzdokumentation bilden hierbei für beide Vereine die Geobasisdaten, welche aus amtlichen Liegenschaftskarten, eigenen vermessungstechnisch erstellten Geobasisdaten bzw. geeigneten Daten externer Quellen abzuleiten sind (DVGW, 2010; VDE, 2010). Ihnen muss ein einheitliches geodätisches Bezugssystem bzw. Koordinatensystem zu Grunde liegen, das heißt bei Übernahme von Daten aus anderen Bezugssystemen müssen diese transformiert werden (VDE, 2010; DVGW, 2010). Die fachspezifischen Regelungen der Rohrnetze und der Elektrizitätsnetze unterscheiden sich zum Teil, daher werden die Regeln in den nachfolgenden Kapiteln für die Sparte Strom und für die Sparten Gas und Wasser im Detail beschrieben.

2.1.2 Strom

Ausgangspunkt war im Bereich der Elektrizität die DIN Norm DIN 2425-7, die aufgrund der technologischen Entwicklung im Jahre 2005 außer Kraft gesetzt und durch die VDE-Anwendungsregel VDE-AR-N 4201 ersetzt wurde (VDE, 2010). Diese bildet „die Grundlage für das Erstellen und Fortführen einer digitalen Netzdokumentation“ und beschreibt die dabei zu berücksichtigenden Mindestanforderungen und Standards (VDE, 2010). Außerdem ist dort festgelegt, dass die Netzdokumentation „für Aufgaben der Unternehmen zum Betrieb elektrischer Netze der allgemeinen Versorgung [...] sowie für Anforderungen von Dritten“ zu führen ist (VDE, 2010).

Nach der Anwendungsregel werden Bauobjekte zunächst vor Ort eingemessen und aufgenommen. Dabei werden vor Ort von der für die Einmessung verantwortlichen Person Aufnahmeskizzen erstellt oder die Daten werden in einer Vermessungsdatei gespeichert (VDE, 2010). Die Anhänge 1 und 2 zeigen Muster einer manuellen Aufnahmeskizze und einer elektronischen Aufnahmeskizze.

Anschließend erfolgt im Büro die Datenerfassung, bei der die vor Ort aufgenommenen Daten in das System überführt und gespeichert werden. Dabei muss das Leitungsnetz in der Netzdokumentation aktuell und vollständig nachgewiesen werden. Die Erfassung erfolgt in der Regel auf Grundlage der Aufnahmeskizzen mithilfe eines Geoinformationssystems mit entsprechenden Zusatzfunktionen. Dabei werden die Objekte gezeichnet und miteinander verknüpft, sowie Sachdaten hinzugefügt. Die gezeichneten Objekte mit ihren Attributen werden dann in der entsprechenden Datenbank abgespeichert. Zur Gewährleistung einer qualitativ hochwertigen und konsistenten Netzdokumentation sind für die Erfassung und Pflege der Daten Erfassungsanweisungen zu führen. Durch diese wird sichergestellt, dass die Netzdaten vollständig, lesbar, richtig, aktuell erfasst und verwaltet werden. Ist dies aufgrund älterer Daten oder aus anderen Gründen nicht der Fall, müssen die „Anlagenteile und Leitungspunkte, deren Dokumentation nicht der erforderlichen Genauigkeit entspricht oder Leitungsbereiche mit unsicher dokumentierten Verlauf“ (VDE, 2010) gekennzeichnet werden, um die Qualitätssicherungsmaßnahmen einhalten zu können (VDE, 2010).

Die Datenerfassung erfolgt dabei auf Grundlage von Datenmodellen zu den Bestandteilen des Leitungsnetzes. Nach VDE sind sie „entsprechend den Anforderungen der verschiedenen Spannungsebenen und Funktionsebenen sowie der Nutzergruppen zur Analyse, Präsentation und grafischen Wiedergabe der Daten sowie in Abhängigkeit der Funktion der Leitung auszulegen. Für die Realisierung der in diesem Regelwerk beschriebenen Planwerke und Anforderungen sind die geografischen Beschreibungen und die Netztopologien zu führen“ (VDE, 2010). Somit erfolgt die Datenerfassung und -zeichnung im GIS nach einem bestimmten in der Datenbank festgelegten Schema.

Es werden die Daten zu Leitungen, zu Anlagen sowie zu Einbauteilen wie Muffen oder Masten geführt. Dabei sind nach VDE bei Leitungen die Attribute zu Typ, Spannungsebene bzw. Funktionsebene, Leitungsart, Verlegedatum, Länge, Betriebsstatus, Eigentümer, bei Anlagen die Anlagenart und die Anlagenbezeichnung und bei Einbauteilen Typ, Einbaudatum, Betriebsstatus und Eigentümer zu führen. Zur Erhaltung und Aktualisierung dieser Attribute müssen gegebenenfalls Datenbestände aus anderen IT-Systemen über Zuordnungsschlüssel gekoppelt werden (VDE, 2010).

Für die grafische Wiedergabe der eingemessenen und erfassten Bestands- und Lagenachweise des Leitungsnetzes gibt es sogenannte Netzpläne, die nach VDE entsprechend der verschiedenen Anforderungen bei z.B. Planung, Bau, Betrieb, Asset-Management und Informationsbereitstellung an

Dritte zu erstellen, zu führen und bedarfsorientiert auszugeben sind. Dabei wird zwischen Bestandsplänen, Übersichtsplänen und Sonderplänen unterschieden (VDE, 2010).

Bestandspläne sind die detaillierten Visualisierungen des Datenbestandes in einem größeren Maßstab (1:250, 1:500, 1:1000) und enthalten dabei alle erforderlichen Informationen zu den Bestandteilen des Leitungsnetzes. Somit bilden sie auf Grundlage der Basiskarte maßstäblich die Lage des Leitungsnetzes ab. Teilweise muss dabei bei vielen Netzobjekten auf engem Raum auch vom Maßstab abgewichen werden, dann muss ein vergrößerter Detailplan des entsprechenden Ausschnittes beigelegt sein. Zusätzlich enthalten die Bestandspläne Maßangaben auf Grundlage der Einmessung, um die Objekte vor Ort in ihrer Lage rekonstruieren zu können. Die Maßangaben beziehen sich dabei auf die Leitungsachse, beziehungsweise auf die Achse oder die Ränder des Leitungsbündels. Da die Leitungen im Strom teilweise untereinander in einer Trasse liegen, was im 2D-Plan nicht abgebildet werden kann, muss eine entsprechende und konsistente Darstellungsform der Leitungen im 2D Plan gewählt werden, bei der alle vorhandenen Leitungen entsprechend abgebildet oder gekennzeichnet sind, sodass die tatsächliche Lage aller Leitungen wieder rekonstruiert werden kann. Sie können dabei entweder einzeln als eine Mehrstrichdarstellung gezeichnet werden oder es kann lediglich die Trassenachse als Einstrichdarstellung gezeichnet werden (VDE, 2010). Bei den für die vorliegende Arbeit verwendeten Daten handelt es sich um eine Mehrstrichdarstellung, bei der die erste Leitung in der Trasse lagerichtig gezeichnet ist und die übrigen Leitungen aufgrund der Verdrängung verzeichnet dargestellt werden. Über Querschnitte wird dann im Plan aufgezeigt, wie die Leitungen tatsächlich in der Trasse liegen und welche Rohre belegt sind. Anhang 3 zeigt ein Beispiel eines solchen Bestandsplanes in Mehrstrichdarstellung. Ergänzend dazu werden die Schutz- und Lehrrohre, in denen sich die Leitungen befinden oder in die noch Leitungen verlegt werden können, sowie die Trassen, in denen sich die Rohre und die Leitungen befinden, dargestellt. In welcher Tiefe sich Leitungen erster Ebene, Leitungen zweiter Ebene, usw. befinden, ist durch Vorschriften über die Verlegetiefe festgelegt (VDE, 2010).

Übersichtspläne dienen der Planung von Leitungen und sind für den Betrieb der Netze erforderlich. Sie enthalten eine generalisierte Darstellung des Leitungsnetzes im Maßstab 1:2500 oder 1:5000 und kleiner, teilweise auch im Maßstab 1:25000 oder 1:50000. Dabei sind die Trassen der Netze in generalisierter Form mit den erforderlichen Netzinformationen darzustellen. Anhang 4 zeigt ein Beispiel eines Übersichtsplanes (VDE, 2010).

Sonderpläne sind weitere Pläne, die bei Bedarf auf der Grundlage der Datenbestände zur Netzdokumentation angefertigt werden. Dies sind zum Beispiel Schemapläne, Schaltpläne oder Rechtserwerbspläne (VDE, 2010).

Für die grafische Ausprägung der Pläne hat der VDE einzuhaltende Mindestanforderungen definiert, sie sind in Anhang 5 dargestellt. Dabei kann die Ausprägung über Farben, Strichstärken, Signaturen und textliche Kennzeichnungen weiter differenziert werden. Gibt es zu den Mindestanforderungen des VDE Ergänzungen oder Abweichungen, müssen diese erläutert werden. Außerdem muss die Ausrichtung der Pläne eindeutig erkennbar sein und die Beschriftung muss abhängig vom Maßstab lesbar, einheitlich und eindeutig dem betreffenden Objekt zuzuordnen sein (VDE, 2010).

2.1.3 Gas und Wasser

Auch im Bereich der Rohrleitungen für Gas und Wasser war eine DIN-Norm der Ausgangspunkt der Regel. Die DIN Normen DIN 2425-1 und DIN 2425-3 für die analoge Erstellung und Fortführung von Rohrnetzplänen der öffentlichen Gas- und Wasserversorgung sind zwar nicht außer Kraft gesetzt,

entsprechen aber aufgrund der technologischen Entwicklung und der Nutzung digitaler Daten nicht mehr dem Stand der Technik (DVGW, 2010). Daher wurde das Arbeitsblatt DVGW GW 120 als technische Regel erstellt, welches die Grundlage für das Erstellen und Fortführen einer digitalen Netzdokumentation bildet (DVGW, 2010).

Das Arbeitsblatt „gilt für das Einmessen von Leitungen und Anlagen der Gas- und Wasserverteilung sowie der Rohrfernleitungen, die Neuerstellung und Fortführung einer digitalen Netzdokumentation beim Einsatz eines GIS und die Bereitstellung und Präsentation dieser Daten“ (DVGW, 2010). Dabei werden ähnliche Anforderungen wie bereits im vorigen Kapitel bei der Anwendungsregel zur Netzdokumentation des VDE beschrieben dargelegt. Wie auch im Strom müssen die Objekte für die Rohrnetzdokumentation zunächst vor Ort eingemessen werden und in Aufnahmeskizzen dokumentiert werden (DVGW, 2010). Die Anforderungen hierfür decken sich mit denen der Regel des VDE. Anhänge 6 und 7 zeigen Muster für eine analoge Aufnahmeskizze und eine digitale Aufnahmeskizze für ein Rohrnetz, Anhang 8 zeigt die Inhalte einer Aufnahmeskizze.

Des Weiteren ist im Arbeitsblatt des DVGW festgelegt, dass anschließend eine Datenerfassung in ein System erfolgt, um die Daten dauerhaft in der Datenbank zu speichern (DVGW, 2010). Auch hier wird vorgeschrieben, dass das Leitungsnetz aktuell und vollständig nachzuweisen ist und für die Qualität und Konsistenz der Daten eine Erfassungsanweisung zu führen ist, deren Anforderungen sich mit den Anforderungen des VDE für die Sparte Strom decken (DVGW, 2010). Wie auch im Strom erfolgt die Datenerfassung auf Grundlage von in der Datenbank festgelegten Datenmodellen, damit die Daten konsistent nach einem bestimmten Schema im GIS gezeichnet und gespeichert werden (DVGW, 2010). Dabei sind die Datenmodelle nach DVGW „entsprechend den Anforderungen der Nutzergruppen zur Analyse, Präsentation und graphischen Wiedergabe der Daten sowie in Abhängigkeit der Funktion der Leitung auszulegen. Für die Realisierung der in diesem Regelwerk beschriebenen Planwerke und der zugehörigen Anforderungen sind die geografische Beschreibung und die Netztopologie zu führen“ (DVGW, 2010). Der DVGW legt außerdem fest, dass für Leitungen, Einbauteile und Anlagen bestimmte Informationen zu erfassen und zu pflegen sind (DVGW, 2010). Bei Leitungen handelt es sich hierbei um Material, Dimension, Design Pressure, Leitungsart, Rohrumhüllung, Rohrrinnenschutz, Verlegedatum, Länge, Betriebsstatus und Eigentümer (DVGW, 2010). Bei Einbauteilen müssen die Informationen zu Typ, Einbaudatum, Betriebsstatus und Eigentümer geführt werden und bei Anlagen muss die Anlagenart, die Anlagenbezeichnung der Eigentümer in Attributen gespeichert werden (DVGW, 2010). Eventuell müssen dabei Datenbestände aus anderen IT-Systemen über geeignete Zuordnungsschlüssel gekoppelt werden, um die entsprechend notwendigen Informationen zu erhalten und aktuell fortzuführen (DVGW, 2010).

Wie auch im Strom werden als Bestandteil der Netzdokumentation Netzpläne zur Visualisierung erstellt. Dabei handelt es sich im Gas und Wasser um Rohrnetzpläne, die der graphischen Wiedergabe der eingemessenen und im GIS erfassten Bestands- und Lagenachweise des Leitungsnetzes dienen (DVGW, 2010). Sie müssen im GIS unter Berücksichtigung der Anforderungen der Planung, dem Bau, dem Betrieb und dem Asset-Management, sowie für die Informationsbereitstellung an Dritte geführt werden (DVGW, 2010). Es wird zwischen Bestandsplänen, Übersichtsplänen, Bauplänen, Rechtserwerbplänen und Sonderplänen unterschieden, die im Nachfolgenden näher beschrieben werden.

Bestandspläne sind, wie im Strom auch, detaillierte Visualisierungen des Datenbestandes zur Netzdokumentation im großen Maßstab (DVGW, 2010). Die Anforderungen hierbei decken sich mit denen des VDE für die Sparte Strom, mit Ausnahme der Darstellung der Leitungen. Es gibt in der Regel immer nur eine Rohrnetzleitung der jeweiligen Sparte im Untergrund, sodass diese nicht übereinanderliegen. Somit kommt es im 2D Plan nicht wie im Strom zu einem Konflikt in der

Darstellung und die Rohre können alle lagerichtig gezeichnet werden. Daher fehlt im Arbeitsblatt des DVGW eine entsprechende Festlegung, wie die Rohre dargestellt werden. Anhang 9 zeigt ein Muster eines Bestandsplanes.

Übersichtspläne sind wie im Strom zu führen und dienen ebenfalls der Übersicht des Netzes mit einer generalisierten Geometrie des Leitungsnetzes (DVGW, 2010). Daher liegen die Übersichtspläne in kleinerem Maßstab vor (DVGW, 2010). Sowohl Übersichtspläne als auch Bestandspläne werden für Verteilungsnetze und für Rohrfernleitungen erstellt. Anhang 10 zeigt ein Muster eines Übersichtsplanes.

Baupläne und Rechtserwerbpläne werden hingegen nur für Rohrfernleitungen erstellt. Ein Bauplan enthält die Detailkonstruktion der Leitung und die Festlegung des geplanten Leitungsverlaufs für den Leitungsbau und ist somit Grundlage für die Absteckung (DVGW, 2010). Ein Rechtserwerbsplan enthält den Verlauf der Leitung und den Leitungsschutzstreifen sowie weitere Angaben, die für den Erwerb von Leitungsrechten erforderlich sind (DVGW, 2010). Er enthält die Hoheits-, Verwaltungs- und Katastergrenzen, die Katasterbezeichnungen und Eigentümerangaben, basierend auf der amtlichen Liegenschaftskarte (DVGW, 2010).

Neben diesen verpflichtenden Plänen gibt es noch Sonderpläne, die bei Bedarf auf Grundlage der Datenbestände zur Netzdokumentation angefertigt werden. Dies sind zum Beispiel Schemapläne, Schieber- und Hydrantenpläne, Druckzonenpläne oder Befliegungspläne (DVGW, 2010).

Für die graphische Ausprägung hat auch der DVGW Mindestanforderungen festgelegt, welche in Anhang 11 dargestellt sind. Dabei können zusätzlich Design Pressure und Maximum Operating Pressure der jeweiligen Leitungen über Farben, Strichstärken, Signaturen oder textliche Kennzeichnungen differenziert werden (DVGW, 2010). Ferner müssen Ergänzungen und Abweichungen zu diesen Regeln entsprechend gekennzeichnet werden, die Ausrichtung der Pläne muss eindeutig erkennbar sein und die Beschriftung muss entsprechend des Maßstabes lesbar, einheitlich und eindeutig dem betreffenden Objekt zugeordnet werden können (DVGW, 2010).

2.2 Visualisierungstechnologien

Zur Erstellung künstlicher Umgebungen, in denen die Nutzer mit virtuellen Objekten interagieren können, haben sich zur Kombination von Realität und Virtualität verschiedene Lösungsansätze entwickelt, die auf unterschiedlichen Technologiekonzepten basieren (BAE ET AL., 2020). In den nachfolgenden Ausführungen werden die unter dem Begriff „Mixed Reality“ (MR) zusammengefassten Technologien beschrieben und gegeneinander abgegrenzt.

Die Anfänge von Mixed Reality gehen auf die Arbeit von Ivan E. Sutherland in den 1960er Jahren zurück. Inhalt seines Forschungsprojektes ist ein am Kopf angebrachtes Display, das dem Nutzer dreidimensionale Informationen in seiner Umgebung bereitstellen sollte (SUTHERLAND, 1968). Durch die Kombination von geeigneten, zweidimensionalen Bildern wurde eine Illusion eines dreidimensionalen Objektes auf dem Display des Nutzers erzeugt (SUTHERLAND, 1968). Dabei wird die Arbeitsweise des menschlichen Gehirns genutzt. Es ist in der Lage, zwei zweidimensionale und versetzte Bilder, die jeweils durch ein Auge wahrgenommen werden, zu einem räumlichen Eindruck zu verschmelzen und so eine dreidimensionale Tiefenwahrnehmung zu erzeugen (SCHNEIDER, 2013). Das Ziel des Forschungsprojektes lag darin, eine realitätsnahe Illusion zu erzeugen, die auf der einen Seite dreidimensional ist und auf der anderen Seite die Perspektive analog der Kopfbewegung anpasst (SUTHERLAND, 1968). Diese Anpassung der Perspektive sollte realitätsnah erfolgen, das heißt, das dargestellte, virtuelle Objekt ändert sich so, wie ein Betrachter das gleiche Objekt real wahrnehmen würde (SUTHERLAND, 1968). Dem Forscherteam ist es gelungen, die oben beschriebene Darstellung zu erzeugen und seine Wirkung auf den Menschen zu testen. Im Anschluss daran haben sich unterschiedliche Ansätze für Visualisierungsmethoden zur Kombination der realen Welt und des virtuellen Raumes entwickelt.

Zur Klassifizierung und Abgrenzung der unterschiedlichen Methoden entwickelten Paul Milgram und Fumio Kishino das „virtuality continuum“ (siehe Abbildung 2). Das „virtuality continuum“ ist ein zweidimensionales Modell, welches die beiden Extreme der Umgebungsvariablen gegenüberstellt (MILGRAM UND KISHINO, 1994). Auf der linken Seite des zweidimensionalen Kontinuums befindet sich der Parameter „reale Umgebung“, während sich auf der anderen Seite der Parameter „virtuelle Umgebung“ befindet (MILGRAM UND KISHINO, 1994). Das Konzept dahinter bezieht sich auf die Frage, wie groß die jeweiligen Anteile bei der Vermischung von virtueller Umgebung und realer Umgebung sind, wobei reale Umgebungen als das eine Extrem und virtuelle Umgebungen als das andere Extrem angesehen werden (MILGRAM UND KISHINO, 1994). Auf der linken Seite des Kontinuums befinden sich Anwendungen, die bei der Darstellung der Umgebungen ausschließlich reelle Objekte nutzen (MILGRAM UND KISHINO, 1994). Dem gegenüber finden sich auf der rechten Seite Anwendungen, die ausschließlich virtuelle Umgebungen zeigen (MILGRAM UND KISHINO, 1994). Den Bereich zwischen den beiden Extremen haben MILGRAM UND KISHINO (1994) als „Mixed Reality“ definiert. Darunter fallen alle Anwendungen, bei denen reale Objekte und virtuelle Objekte gemeinsam in einem Display dargestellt werden (MILGRAM UND KISHINO, 1994).

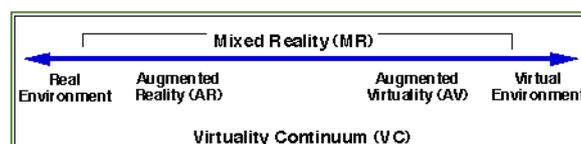


Abbildung 2: Virtuality Continuum (VC)

Quelle: MILGRAM UND KISHINO (1994)

Gemäß diesem Kontinuum werden Technologien auf der linken Seite „Augmented Reality“ (AR) genannt. Die konzeptionelle Basis von AR besteht darin, ein virtuell erzeugtes Objekt, zum Beispiel eine unterirdische Wasserleitung, in ein reales Umfeld zu projizieren. Dabei bildet das reale Umfeld die Grundlage, welche durch virtuelle Objekte ergänzt wird (AZUMA, 1997). Im Idealfall wirkt es für den Nutzer, als würden die virtuellen und die realen Objekte im selben Raum koexistieren (AZUMA, 1997).

Eine der ersten Forschungsarbeiten, die AR definierte, war die Arbeit von Ronald Azuma. In seiner Arbeit legte er drei Voraussetzungen fest, die gegeben sein müssen, damit ein AR System vorliegt:

1. Die Realität und Virtualität werden kombiniert
2. Das System ist in Echtzeit interaktiv
3. Das System enthält drei Dimensionen

Der Vorteil dieser Technologie ist nach AZUMA (1997), dass dem Nutzer ermöglicht wird, mit der realen Welt zu interagieren, während virtuelle Objekte zusätzliche Informationen bereitstellen können, die der Nutzer mit seinen eigenen Sinnen nicht erkennen kann. Er wird dadurch dabei unterstützt, unterschiedliche realweltliche Aufgaben leichter zu bearbeiten. Seit den 90er Jahren befindet sich diese Technologie in der Entwicklung und es hat sich ein eigenes Forschungsfeld etabliert. Die heute verfügbaren Produkte bestehen in der Regel aus einem darstellenden Display, einem Rechenprozessor, einer Softwarelösung und gegebenenfalls ergänzenden Modulen.

Die Darstellungstechnologien, die dabei zum Einsatz kommen, werden nach AZUMA ET AL. (2001) in drei Kategorien unterteilt: Am Kopf getragene Displays (head-mounted), Handgeräte (handheld) und projizierende Displays (projective). Head-mounted Displays werden am Kopf getragen, sodass die Bilder direkt vor den Augen erzeugt werden. Die Umsetzung kann auf zwei unterschiedliche Arten erfolgen. Die Methode der optischen Durchsicht besteht darin, die reale Welt mithilfe einer durchsichtigen Projektionsfläche zu erkennen und gleichzeitig virtuelle Objekte auf dieser Projektionsfläche als Overlay darzustellen (Abbildung 3). Im Gegensatz dazu erfasst die Methode der Video-Durchsicht die Realität über eine am Kopf getragene Kamera und fügt die virtuellen Objekte in Echtzeit ein (Abbildung 4). Die Wiedergabe erfolgt über ein blickdichtes Display (AZUMA ET AL., 2001).

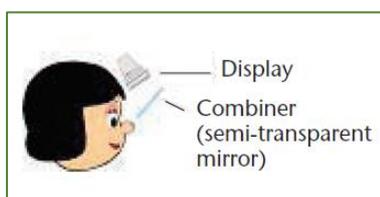


Abbildung 3: Optische Durchsicht
Quelle: AZUMA ET AL. (2001)

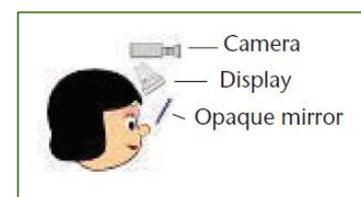


Abbildung 4: Kameradurchsicht
Quelle: AZUMA ET AL. (2001)

Die zweite Kategorie beinhaltet die Handgeräte (handheld). Dabei werden in der Hand getragene Displays mit einer angefügten Kamera genutzt, um die Objekte über eine Video-Durchsicht darzustellen (AZUMA ET AL., 2001). Beispiele für diese Kategorie sind Smartphones oder Tablets. Projizierende Displays stellen die dritte Kategorie dar. Ihre Technologie besteht darin, die virtuelle Information direkt auf physische Objekte zu projizieren (AZUMA ET AL., 2001). Hierbei wird lediglich das virtuelle Objekt darstellungstechnisch erzeugt, während die Projektionsfläche das relevante reale Objekt ist.

Im Gegensatz zu der oben beschriebenen Technologie der Augmented Reality, beschreibt die Technologie der „Augmented Virtuality“ (AV) einen anderen Ansatz, der sich auf der rechten Seite des „virtuality continuum“ nach MILGRAM UND KISHINO (1994) befindet. Die Vorgehensweise besteht darin, eine virtuelle Umgebung zu erzeugen und reale Objekte in diese Umgebung hinein zu projizieren (MILGRAM UND COLQUHOUN, 2001). Somit wird die Technologie der Augmented Reality umgekehrt und ein virtueller Raum wird durch reale Objekte überlagert (BAE ET AL., 2020). Diese Methode wird beispielsweise für Video-Konferenz-Systeme eingesetzt, um die realen Personen in einer virtuellen Umgebung zu zeigen (REGENBRECHT ET AL., 2003). Sie ist bereits in Konferenzsystem-Lösungen wie Microsoft® Teams oder Zoom Meetings integriert. Weitere Anwendungsfelder dieser Technologie sind zum Beispiel die Bildung (TERNIER ET AL., 2012), die Navigation von Drohnen (SEDLMAJER ET AL., 2019) sowie die kartenbasierte Navigation (TERNIER ET AL., 2012). Anwendungen wie Google StreetView ermöglichen das Einsehen realer Bilder auf einer virtuellen Karte (TERNIER ET AL., 2012).

Am rechten Rand des Kontinuums, in der totalen virtuellen Umgebung befindet sich das Konzept der „Virtual Reality“ (VR). Diese Technologie beschreibt die Darstellung einer reinen virtuellen Realität, bei der für den Nutzer eine künstliche, dreidimensionale Realität geschaffen wird, in die er eintauchen kann (LATTA UND OBERG, 1994).

Aus den oben ausgeführten Betrachtungen lässt sich ableiten, dass im Rahmen dieser Arbeit nur eine Visualisierungstechnologie herangezogen werden kann, durch die eine Darstellung der Realität erfolgt und virtuelle Objekte in diese Darstellung einbezogen werden können. Demzufolge fokussiert sich das weitere Vorgehen ausschließlich auf die Visualisierungstechnologie Augmented Reality. Ein Schwachpunkt dabei ist die einseitige Informationsverarbeitung, denn die Objekte werden oft lediglich in das Kamerabild projiziert. Dabei werden die virtuellen Objekte mit dem Kamerabild überlagert und können nicht mit der Realität im Kamerabild interagieren. Als im Jahr 2016 der Begriff „Mixed Reality“ durch Microsoft neu geprägt wurde, wurden Virtual Reality und Augmented Reality durch die Microsoft HoloLens miteinander kombiniert (TREPPER ET AL., 2017). Diverse Sensoren und ein in einer Brille integrierter Computer mit Kamera ermöglichen die Interaktion von in das Sichtfeld projizierten, virtuellen Objekten, sogenannten Hologrammen, mit Objekten der Realität (TREPPER ET AL., 2017). Ein Hologramm ist eine für den Menschen als dreidimensional wahrnehmbare Abbildung beziehungsweise Projektion von Objekten im Raum (GHULOUM, 2010). Um eine solche Projektion zu ermöglichen, wird zunächst eine Lichtquelle auf eine Oberfläche projiziert und gestreut (GHULOUM, 2010). Eine weitere Lichtquelle strahlt das darzustellende Objekt aus (GHULOUM, 2010). Wenn die Strahlen beider Lichtquellen aufeinander treffen, entstehen Interferenzen, die eine Erstellung von Hologrammen ermöglichen (GHULOUM, 2010). Bei diesem Verfahren entstehen durch Interaktion zwischen den Lichtquellen Beugungen des Lichts, durch die eine dreidimensionale Illustration möglich wird (GHULOUM, 2010). Die HoloLens kann diese virtuellen Hologramme in das Sichtfeld projizieren, die Beziehungen der realweltlichen Objekte mit den virtuellen Objekten erfassen und in Interaktion miteinander darstellen (WANG ET AL., 2018). Über eine Art Head-Up Display werden die Hologramme in einem definierten Bereich realitätsgetreu das Sichtfeld des Nutzers projiziert und er kann mit den Hologrammen interagieren (TREPPER ET AL., 2017). Die reale Umgebung wird durch eine einfache Durchsicht dargestellt, da die HoloLens über durchsichtige, holographische Linsen verfügt (MICROSOFT, 2020b). Abbildung 5 verdeutlicht das Konzept der Mixed Reality nach Microsoft, welches im weiteren Verlauf der Arbeit unter dem Begriff „Mixed Reality“ zusammengefasst wird.



Abbildung 5: Unterschiede zwischen VR, AR und MR
Quelle: BAREL (2017)

2.3 Satellitengestützte Ortung und GPS

Zur Ortung der Position eines Betrachters sind unterschiedliche Verfahren möglich, zum Beispiel GNSS, SAPOS®, optisch-photogrammetrische Verortung, etc. Im Rahmen dieser Arbeit wird auf die satellitengestützte Ortung eingegangen, da die wesentlichen Forschungsfragen nicht darin bestehen, die bestmögliche Ortungsvariante zu identifizieren, sondern eine Applikation zu entwickeln, die Daten von mindestens einem Ortungsverfahren verarbeiten kann.

Die Abkürzung GPS (Global Positioning System) wird im allgemeinen Sprachgebrauch als Überbegriff für satellitengestützte Ortung verwendet. Genau genommen ist das GPS aber nicht der Überbegriff, sondern das US-amerikanische globale Navigations satellitensystem (GNSS) (EUROPEAN GLOBAL NAVIGATION SATELLITE SYSTEMS AGENCY, 2017). Neben dem GPS gibt es die drei weiteren GNSS Galileo (europäisch), GLONASS (russisch) und BeiDou (chinesisch) (EUROPEAN GLOBAL NAVIGATION SATELLITE SYSTEMS AGENCY, 2017). Die Funktionsweise und angewandten Prinzipien sind bei allen GNSS vergleichbar ähnlich und werden im Folgenden vereinfacht skizziert.

Bei der satellitengestützten Ortung wird mithilfe von Trilateration eine dreidimensionale Positionsbestimmung durch die Kombination mehrerer Satellitensignale oberflächennah global ermöglicht (RIZOS, 2020). Die Grundlage zur Ortung bildet dabei ein Netz aus Satelliten in erdnahen Umlaufbahnen, die durchgehend Daten zur Erde senden (RIZOS, 2020). Die Satelliten enthalten eine Atomuhr, aus der die genaue Uhrzeit am Satelliten ausgelesen und mit weiteren Informationen zur Erde gesendet wird (ALLAN UND WEISS, 1980). Das Empfangsgerät auf der Erde vergleicht das empfangene Signal des Satelliten mit der eigenen Uhrzeit und errechnet mithilfe der Lichtgeschwindigkeit den zurückgelegten Weg des Satellitensignals, um den Abstand zum Satelliten zu ermitteln. Der errechnete Abstand eines Satelliten ermittelt nun die Position des Nutzers irgendwo auf einer Kugel mit dem Radius der gemessenen Entfernung zum Satelliten und dem Satelliten als Mittelpunkt der Kugel (RIZOS, 2020). Damit die eindeutige Position bestimmt werden kann, werden daher mindestens drei simultan gemessene Entfernungen von drei unterschiedlichen Satelliten benötigt (RIZOS, 2020). Die Anzahl der Schnittflächen und somit die mögliche Position des Nutzers wird dadurch auf zwei reduziert, wobei eine davon zumeist eine unmögliche Position nicht auf der Erdoberfläche ist und so die eindeutige Position des Empfängers auf der Erde näherungsweise bestimmt werden kann (RIZOS, 2020). Für eine einheitliche Positionierung mit nur einer Schnittfläche wird ein viertes Satellitensignal benötigt (RIZOS, 2020). Mit jedem weiteren Signal wird die gemeinsame Schnittfläche kleiner und die Position kann bis auf wenige Meter genauer ermittelt werden.

Eine Reihe von Störfaktoren können dabei Einfluss auf die Satellitensignale nehmen und eine genaue Positionierung verfälschen. Daher werden die Signale in einem Post-Processing-Prozess nochmals überarbeitet. Die zu beachtenden Störfaktoren sind nach KARAIM ET AL. (2018) beispielsweise:

- Diverse Ursachen können Fehler bei der Zeitbestimmung durch die Uhren der Satelliten und der Empfänger auslösen. Dadurch entsteht eine fehlerhafte Abstandsbestimmung.
- Die Verzögerung und Verzerrung des Signals aufgrund der Brechung des Strahlungswinkels eines Signals an der Ionosphäre. Dies kann Fehler in der Abstandsbestimmung auslösen.
- Die Refraktion in der Troposphäre aufgrund von trockenen Gasen und Wasserdampf löst eine Verzögerung des Signals auf und beeinflusst somit die Abstandsbestimmung.
- Die Signalreflektion durch Strukturen, die den Empfänger umgeben, wie zum Beispiel Bebauung oder glatte Oberflächen. Dadurch kommt das Signal über mehrere Wege am Empfänger an, wird mehrmals zu unterschiedlichen Zeitpunkten empfangen und führt zu Mehrdeutigkeiten.

Zur Verbesserung der Präzision können zusätzliche stationäre Empfangsstationen genutzt werden, die zur Berechnung als konstante Orientierungspunkte genutzt werden. Landesvermessungsämter nutzen dies, um eine hohe Genauigkeit zu erreichen. In Deutschland wird dazu von den Landesvermessungsämtern das SAPOS®-System als einheitliches, bundesweites Referenzsystemnetz verwendet, durch das Dienste mit einer Genauigkeit von einem Zentimeter bereitgestellt werden können (RIECKEN UND KURTENBACH, 2017). Es nutzt die Signale von GNSS, GPS, GLONASS und Galileo sowie Signale lokaler Erweiterungssysteme (LGLN). Um dies zu gewährleisten, gibt es bundesweit ein flächendeckendes Netz von über 270 permanent registrierenden GNSS-Referenzstationen (LGLN).

Für das Erreichen einer Genauigkeit von 1m ohne stationäre Empfangsstationen, werden zusätzliche Informationen benötigt, die nicht über herkömmliche Satelliten übertragen werden. Dazu werden Daten geostationärer Satelliten verarbeitet, die dazu führen, dass Zuverlässigkeit, Genauigkeit und Verfügbarkeit des Ortungssignals erhöht werden. Diese Systeme werden als SBAS (Satellite Based Augmentation System) oder auf Deutsch „satellitenbasiertes Ergänzungssystem“ bezeichnet und werden zusätzlich zu den herkömmlichen GNSS genutzt. Dafür gibt es diverse Systeme der „großen Staaten“, wie das WAAS (US-amerikanisch), das EGNOS (europäisch) oder das SDCM (russisch) (EUROPEAN GLOBAL NAVIGATION SATELLITE SYSTEMS AGENCY, 2019).

3 Aktueller Forschungsstand

Zeitgleich mit der Entwicklung der in Kapitel 2.2 beschriebenen Visualisierungstechnologien haben sich Forschungsgebiete herausgebildet, entsprechende Studien liegen vor. Erste Erkenntnisse wurden Ende des vorigen Jahrhunderts veröffentlicht. Dazu gehören die grundlegenden Ausführungen von AZUMA (1997), wie in Kapitel 2.2 beschrieben. Im Rahmen seiner Forschungsarbeit ermittelte er sechs Anwendungsgebiete, in denen sich Augmented Reality (AR) etablieren könnte: Medizin; Instandhaltung und Reparatur; Anmerkungen und Kommentare in digitale Dokumente einblenden (annotations); Robotik; Unterhaltungsindustrie; Verteidigungswesen (AZUMA, 1997). Tatsächlich wurden in diesen sechs Segmenten Forschungen im Bereich von AR und MR durchgeführt. Darüber hinaus erfuhr die Entwicklung der in Kapitel 2.2 beschriebenen Visualisierungstechnologien in den letzten Jahren einen Aufschwung in weiteren Anwendungsgebieten. Insgesamt ergab sich von 2012 bis 2018 bei den bewilligten Patenten für AR Technologien eine konstante jährliche Wachstumsrate von 82% (EVANGELISTA ET AL., 2020). Die nachfolgenden Ausführungen geben einen Überblick zum Stand der Forschung.

Im Bereich der *Medizin* wurde eine Literaturrecherche durchgeführt, die zu dem Ergebnis kam, dass AR eingesetzt werden kann, um den Körper des Patienten sowie relevante medizinische Daten gleichzeitig in derselben physischen Umgebung sichtbar zu machen (SIELHORST ET AL., 2008). TREPPER ET AL. (2017) führte in diesem Zusammenhang eine Studie durch, inwiefern die HoloLens für die Schönheitschirurgie eingesetzt werden kann. Der Mehrwert entsteht dadurch, dass das Einblenden von Hologrammen in die reale Umgebung ergänzende, komplexe Daten in das Sichtfeld der Chirurgen projiziert, während er durchgehend freie Hände zum Arbeiten hat (TREPPER ET AL., 2017). Dabei wurde der Nutzen der HoloLens mit zwei weiteren Brillen verglichen, zum einen der Google Glass als AR-Gerät und zum anderen der Oculus Rift als VR-Gerät (TREPPER ET AL., 2017). Das Ergebnis der Studie war, dass sich die HoloLens durch ihre komfortable Form und ihre robuste Akkulaufzeit sowie durch die Funktionalität, mit freien Händen Hologramme in das Sichtfeld projizieren und manipulieren zu können, deutlich besser für Operationen eignet (TREPPER ET AL., 2017). Eine weitere Studie hat den Einsatz von AR in der orthopädischen Unfallchirurgie untersucht (NEGRILLO-CÁRDENAS ET AL., 2020). Die Studie kam zu dem Ergebnis, dass AR dem Arzt während der Operation helfen kann, indem entsprechende, zusätzliche Daten eingeblendet werden können (NEGRILLO-CÁRDENAS ET AL., 2020). Darüber hinaus liegen weitere Studien zu unterschiedlichen Forschungsfragen vor. Mit Hinblick auf AR lassen die Studien erkennen, dass AR in der Medizin einsetzbar ist und maßgebliche Vorteile dadurch entstehen, dass Virtualität und Realität gleichzeitig dargestellt werden, insbesondere wenn das Gerät in Form einer Brille am Kopf befestigt ist und frei bewegliche Hände ermöglicht.

Im Bereich der *Instandhaltung und Reparatur* wird Augmented Reality bei komplexeren Maschinen eingesetzt, um Qualität sicherzustellen und Kosten zu senken. Auf der „Second IEEE International Conference on Internet-of-Things Design and Implementation (IoTDI)“ im Jahr 2017 wurde die Anwendung ARIOT vorgestellt. Sie besteht aus einem tragbaren IoT AR-System in Form eines Kopfdisplays mit Video-Durchsicht und kann mit Komponenten der zu wartenden Maschine kommunizieren, um dadurch die zugehörigen Echtzeitdaten zu den Komponenten zu visualisieren (WIJESOORIYA ET AL., 2017). Das System hilft den Technikern durch seine verbesserte Anleitung dabei, ihre Produktivität zu steigern und die Inspektion effektiver durchzuführen sowie deren Sicherheit zu erhöhen (WIJESOORIYA ET AL., 2017). Eine weitere Anwendung ist das von ZHU ET AL. (2015) entwickelte Autorensystem ACAAR, durch das für umgebungsintelligente, AR-unterstützte Wartungssysteme der Kontext selbstständig definiert und bestimmt werden kann. Dadurch ist es möglich, Inhalte wie Texte oder CAD Modelle mit Rauminformationen zu kombinieren und die Beziehungen zwischen den

Inhalten und dem Wartungsobjekt zu definieren und zu dokumentieren (ZHU ET AL., 2015). Über eine Nutzerstudie wurde das System anschließend getestet (ZHU ET AL., 2015). Die Studie resultierte darin, dass alle Probanden das Autorensystem zur Erstellung von AR-Anwendungen für Wartungen und Reparaturen für sinnvoll erachteten (ZHU ET AL., 2015). In einer dritten Studie ermittelten HENDERSON UND FEINER (2011) den Nutzen einer AR-Dokumentation für die Wartung und Reparatur im Turm gepanzerter Fahrzeuge. Der für diese Studie entwickelte Prototyp nutzt ein am Kopf getragenes Display, das die natürliche Sicht des Mechanikers um Texte, Beschriftungen, Markierungen und animierte Sequenzen zu erweitert (HENDERSON UND FEINER, 2011). Das Ziel bestand darin, das Verständnis zur Aufgabe, die Lokalisierung sowie die Ausführung zu unterstützen (HENDERSON UND FEINER, 2011). Innerhalb der Studie untersuchten mehrere professionelle Servicetechniker das System bei der Ausführung von 18 Aufgaben unter realen Bedingungen im Feld (HENDERSON UND FEINER, 2011). Das Ergebnis der Studie war, dass die Mechaniker durch das AR-System die Aufgaben schneller lokalisieren konnten und das System als intuitiv und zufriedenstellend empfanden (HENDERSON UND FEINER, 2011).

Der von AZUMA (1997) definierte Anwendungsbereich *Anmerkungen und Kommentare in digitale Dokumente einblenden* umfasst sämtliche Aufgabenstellungen, um Schrift auf reale Objekte zu projizieren und Anmerkungen zu formulieren. Die Einsatzgebiete sind beispielsweise im Bereich Werbung, Marketing, Design, Inspektion und Dokumentation im weitesten Sinne zu finden. Exemplarisch werden an dieser Stelle die Studien von GARCÍA-PEREIRA ET AL. (2020) und MADSEN ET AL. (2016) zitiert. Bei ersterer wurde eine Anwendung für die Inspektion von Fertighäusern entwickelt, mit der Anmerkungen und Vermerke direkt in der Gebäudeansicht gemacht werden können (GARCÍA-PEREIRA ET AL., 2020), während bei letzterer mehrere Strategien zur Darstellung von Beschriftungen untersucht wurden (MADSEN ET AL., 2016). Beide Studien haben gezeigt, dass diverse Vermerke und Anmerkungen gut mit der Realität verbunden werden können und für informative AR-Anwendungen essenziell sind (GARCÍA-PEREIRA ET AL., 2020; MADSEN ET AL., 2016). Außerdem belegte die Studie von GARCÍA-PEREIRA ET AL. (2020), dass Anmerkungen anhand von Objekten der Realität mithilfe einer AR-Anwendung formuliert und gespeichert werden können (GARCÍA-PEREIRA ET AL., 2020).

Im Bereich der *Robotik* gibt es diverse Entwicklungen und Forschungsarbeiten zur Steuerung und Manipulation der Roboter mithilfe von Augmented Reality, teilweise wird hier als Medium die Microsoft HoloLens verwendet. Die Forschungsarbeit von NEVES ET AL. (2018) fokussiert sich auf den Einsatzbereich von AR mit der HoloLens, um die Pfade und Bewegungsabläufe von industriellen Robotern in dem jeweils realen Umfeld sichtbar zu machen und zu manipulieren. Innerhalb dieser Forschungsarbeit wurden zwei fertige und sofort einsatzbereite Anwendungen entwickelt und deren Vorteile ermittelt (NEVES ET AL., 2018). Die erste Anwendung visualisiert die vordefinierten Pfade der Roboter im realen Umfeld. Dies bietet dem Nutzer die Möglichkeit, den Pfad zu verfolgen und Probleme während der Ausführung zu identifizieren (NEVES ET AL., 2018). Mit der zweiten Anwendung können die Roboterpfade direkt anhand der HoloLens über die reale Umgebung erstellt und angepasst werden (NEVES ET AL., 2018). Diese Anwendungen der HoloLens ermöglichen es, ohne Programmiererfahrung komplexe Aufgaben an den Robotern zu lösen und die Komplexität durch die Kombination von Realität und Virtualität für den Endnutzer zu reduzieren (NEVES ET AL., 2018). Eine weitere Studie beschäftigt sich mit der Steuerung von Robotern über eine AR-Anwendung mithilfe der HoloLens anstelle eines Joysticks (MANRING ET AL., 2020). Dabei kann der Nutzer über manuelles oder automatisches Steuern einen holografischen Roboter entsprechend kontrollieren und steuern, während der reale Roboter zeitgleich reagiert (MANRING ET AL., 2020). Die Studie hat gezeigt, dass die Robotersteuerung durch die Kombination und Interaktion von Virtualität und Realität mithilfe der HoloLens deutlich intuitiver und effektiver erfolgt, da der Nutzer den entsprechend gewählten Pfad des Roboters in der Realität direkt einsehen und verstehen kann (MANRING ET AL., 2020). Er sieht somit

nicht erst mit der Bewegung des Roboters, wohin er sich bewegt und ob dies der richtige Weg ist, sondern kann dies realitätsnah und ohne einen Transformationsaufwand im Kopf simulieren. Die Untersuchung von SHA ET AL. (2019) beschäftigt sich mit dem Einsatz von Augmented Reality in Form der HoloLens im laufenden Produktionsprozess durch Simulation des Kontrollsystems über die Dynamiken und Bewegungsabläufe des virtuellen Roboters. Damit werden Algorithmen überprüft, Wege optimiert und Serienabläufe vorbereitet (SHA ET AL., 2019). Da dieses System die Situation und das Verhalten des Roboterkontrollsystems in der echten Umgebung durch die HoloLens simuliert, kann der Einfluss bestimmter Parameter auf das System schnell und effektiv in der realen Umgebung überprüft werden (SHA ET AL., 2019). Dies ermöglicht eine erhebliche Kostensenkung des Herstellungsprozesses und eine Effizienzsteigerung des Roboterkontrollsystems, weil diverse Funktionalitätstest vor der Herstellung und Einführung des Systems in der Produktivumgebung durchgeführt werden können (SHA ET AL., 2019).

In der *Unterhaltungsindustrie* ist die Nutzung von Augmented Reality weit fortgeschritten, um besonders im Gaming-Bereich ein Höchstmaß an Realitätsnähe zu erreichen. Ein bekanntes Beispiel für AR im Bereich des Gamings ist das Spiel Pokémon Go, bei dem virtuelle Pokémons in der realen Umgebung gesucht und eingefangen werden (SARI, 2020). Ältere Beispiele sind das Sportspiel AR² Hockey (OHSHIMA ET AL., 1998) und das Schießspiel RV-Border Guards (OHSHIMA ET AL., 1999). Die in diesem Bereich durchgeführten Studien haben den Schwerpunkt vorwiegend in der Wirkung der jeweiligen Spiele und sind daher für die vorliegende Arbeit nicht relevant.

Im Bereich *Verteidigungswesen* finden sich diverse Forschungsergebnisse. Beispiele hierfür sind Anwendungen für eine erleichterte Planung der Missionen durch eine Visualisierung. Die Verknüpfung von realer Umgebung mit missionskritischen Daten können Missionsplanungen der Air Force durch Simulation der Missionen unterstützen und dadurch die Erfolgsrate erhöhen (ALEXANDER ET AL., 2019). Die Studie kam zu dem Ergebnis, dass AR-Technologien das Potential haben, die Durchführung der Missionen zu verbessern und Verluste zu minimieren (ALEXANDER ET AL., 2019). Nutzern werden über die HoloLens virtuelle Informationen über die Mission eingeblendet und das visuelle Erlebnis wird mit weiteren Nutzern geteilt (ALEXANDER ET AL., 2019). Die reale Umgebung bleibt weiterhin sichtbar und die Hände bleiben frei (ALEXANDER ET AL., 2019). Weitere Studien haben die Simulation von Einsatzvorbereitung und Training zum Gegenstand (FAN ET AL., 2019; ULLO ET AL., 2019). Diese Studien kommen zu dem Ergebnis, dass die Trainingsform dem realen Einsatz sehr nah kommt und dadurch Ressourcen eingespart werden (FAN ET AL., 2019; ULLO ET AL., 2019). Ein weiterer Aspekt ist der Schutz der Soldaten (FAN ET AL., 2019; ULLO ET AL., 2019). Durch die Simulation von Gefechtssituationen und den Einsatz von Sprengstoff kann der Umgang in solchen Situationen realitätsnah erlernt werden, ohne dabei einer Gefahr ausgesetzt zu werden (FAN ET AL., 2019).

Zusätzlich zu den von AZUMA (1997) definierten Anwendungsgebieten haben sich andere Anwendungsgebiete für Augmented Reality etabliert, daraus sind weitere Studienergebnisse entstanden. Eine vollumfängliche Darstellung aller Ergebnisse würde den Rahmen dieser Arbeit deutlich überschreiten. Der Fokus der nachfolgenden Ausführungen liegt daher auf den zentralen Erkenntnissen, nachgewiesen durch mindestens eine Studie.

Der Einsatz von AR im *Bildungswesen* ist Gegenstand weitreichender Forschungsarbeiten. Viele Studien versuchen zu ermitteln, inwieweit Schülern das Lernen erleichtert wird. Exemplarisch sei an dieser Stelle die Anwendung MantarayAR erwähnt. Es handelt sich um ein Statistik-Lernprogramm, welches College-Studenten das Erlernen von statistischen Grundlagen erleichtern sollte (CONLEY ET AL., 2020). Die in diesem Zusammenhang durchgeführte Studie prüfte die Auswirkungen auf die Schüler durch den Einsatz von Augmented Reality (CONLEY ET AL., 2020). Durch die Studie konnte belegt werden, dass die Studenten, die mit MantarayAR lernten, ein höheres Engagement zeigten und höhere

Lerngewinne erzielten als die Vergleichsgruppe ohne diese Anwendung (CONLEY ET AL., 2020). Auch HANID ET AL. (2020) und HERNANDEZ-DE-MENENDEZ ET AL. (2020) ermittelten bei ihren Berichten, dass AR beim Lernen unterstützt und einen Mehrwert dadurch schaffen kann, dass Schüler und Studenten eine Steigerung von Motivation und Engagement zeigten. Eine weitere Studie ermittelte den Lerneffekt von High School Schülern beim Erlernen des Elektromagnetismus mithilfe einer AR-Lernanwendung (IBÁÑEZ ET AL., 2014). Der Lerneffekt dieser Schüler wurde mit dem Lerneffekt einer Vergleichsgruppe verglichen, die den Stoff mithilfe einer Webanwendung gleichen Inhaltes erlernen sollten (IBÁÑEZ ET AL., 2014). Dabei kamen IBÁÑEZ ET AL. (2014) zu dem Ergebnis, dass die Schüler mit der AR-Anwendung einen höheren Lernerfolg erzielten als die Vergleichsgruppe. Als Grund wurde ermittelt, dass die Schüler eine positivere Stimmung und mehr Zufriedenheit beim Lernen zeigten, eine höhere Konzentration beim Lösen der Aufgaben aufwiesen sowie einen höheren und effektiveren Wissenszuwachs hatten (IBÁÑEZ ET AL., 2014). Bei visualisierungsbezogenen Fragen trat dieser Effekt besonders intensiv auf (IBÁÑEZ ET AL., 2014). Die Schüler empfanden die Interaktion mit realen Objekten, die bei der Anwendung ermöglicht wurde, als natürliche Aktivität und dies half ihnen, den Schaltkreis Schritt für Schritt aufzubauen und dessen physikalischen Eigenschaften in Echtzeit zu prüfen (IBÁÑEZ ET AL., 2014). Die Interaktivität und Realitätsnähe der Anwendung ermöglichte ein direktes Feedback zu jedem Arbeitsschritt (IBÁÑEZ ET AL., 2014). Als mögliche Gründe für das verbesserte Verständnis der elektromagnetischen Zusammenhänge mithilfe von AR vermuten IBÁÑEZ ET AL. (2014) einen Zusammenhang mit Konzentration, Kontrolle, autotelischer Erfahrung sowie Unterschiede im Aufwand der kognitiven Leistung zu Problemlösungen, die einen höheren Abstrahierungsgrad erfordern.

Der Einsatz von Augmented Reality in Zusammenhang mit der *Nutzung von Geoinformation* ist Gegenstand weiterer Forschungsarbeiten. Eine dieser Arbeiten untersucht den Nutzen von Augmented Reality bei der Feldarbeit in den Geowissenschaften für die Beobachtung und der präzisen Aufnahme benötigter geologischer Daten (GAZCÓN ET AL., 2018). Die hierfür entwickelte Anwendung ermöglicht es, Informationen mit Geotags zu registrieren und abfallende Ebenen aus entfernten Blickwinkeln zu messen sowie frühere Positionen zu visualisieren (GAZCÓN ET AL., 2018). Dies hilft dabei, die räumlichen Beziehungen der aufgenommenen Stichproben zu verstehen (GAZCÓN ET AL., 2018). Das Ergebnis der in diesem Zusammenhang durchgeführten Pilotstudie war, dass Experten in dem System ein hohes Potential sehen, ihre Feldarbeit zu ergänzen, da die reale Umgebung direkt in einem System mit virtuellen Informationen und Ergebnissen ergänzt werden kann (GAZCÓN ET AL., 2018). Eine Studie von LEE ET AL. (2015) untersucht die Visualisierung von Daten zu Bohrlöchern. Die AR-Visualisierung erfolgt auf Basis einer iPad-Anwendung (LEE ET AL., 2015). Durch die Kombination der drei Module Augmented Reality, Karte und Datenbank können die Bohrlöcher über eine Kamera lokalisiert und mithilfe des AR-Moduls in die reale Umgebung übertragen sowie entsprechend markiert werden (LEE ET AL., 2015). Zusätzlich werden die Standorte der Bohrlöcher in einer 2D-Karte angezeigt und die Daten über das Datenbankmodul direkt gespeichert (LEE ET AL., 2015). Zusammenfassend hat die Studie gezeigt, dass durch die intuitive Repräsentation der Daten im realen Umfeld nicht zugängliche Standorte besser verstanden werden können und die Anwendung daher sowohl Geologen als auch Ingenieure bei ihrer Arbeit unterstützt (LEE ET AL., 2015). Dennoch ist die Genauigkeit der Visualisierung in der Anwendung stark abhängig von der Qualität der vorliegenden Daten und einer präzisen Standortbestimmung des Nutzers (LEE ET AL., 2015). Eine dritte Studie fokussiert sich auf den Nutzen der Kombination von CityGML-Daten mit Augmented Reality im Bauwesen mit dem Fokus auf die Nachverfolgung von Position und Orientierung von Objekten mithilfe von günstigen, mobilen Standardgeräten, zum Beispiel Smartphones (BLUT UND BLANKENBACH, 2020). Dadurch können die Planung, die Durchführung sowie die Kommunikation der Entscheidungsträger bei dem Bau von Gebäuden oder Gebäudeteilen erleichtert werden (BLUT UND BLANKENBACH, 2020). Informationen werden intuitiv wahrgenommen, ohne dass die virtuellen Objekte oder Informationen von einer

zweidimensionalen Darstellung in die dreidimensionale, reale Welt transformiert werden müssen (BLUT UND BLANKENBACH, 2020). Aus dieser Studie resultierte das Ergebnis, dass der Anwendungsfall mit mobilen Endgeräten wie Smartphones möglich ist und Vorteile erzeugt (BLUT UND BLANKENBACH, 2020).

Im Bereich *Wasser- und Energieversorgung* stehen diverse Forschungsergebnisse zur Verfügung, die Visualisierungen von Daten der im Untergrund liegenden Objekte zum Gegenstand haben. Im Rahmen umfangreicher Forschungsarbeiten berichten SCHALL ET AL. (2010) über Erkenntnisse im Bereich von 3D-GIS und Augmented Reality. Bei diesem Forschungsprojekt wurde das Augmented Reality System *Vidente* für ein standort- und kontextbezogenes Handheld-Gerät entwickelt, um Wasserversorgungsnetze aus dem Untergrund auf die Erdoberfläche zu projizieren (SCHALL ET AL., 2010). Das Forschungsprojekt hatte zum Ziel, die Optimierung von Arbeitsprozessen im Bereich unterirdischer Versorgungsnetze zu untersuchen, indem durch die Kombination virtueller Objekte und der realen Umgebung eine intuitivere Informationsvermittlung stattfindet (SCHALL ET AL., 2010). Dadurch hat das System das Potential, Arbeitszeit einzusparen und die Effizienz mobiler Arbeitskräfte während des gesamten Prozesses in der Infrastruktur der Wasserversorgungsnetze zu steigern (SCHALL ET AL., 2010). Der gesamte Prozess umfasst Planung, Betrieb, Wartung, Inspektion, Fehlermanagement und Entscheidungsfindung (SCHALL ET AL., 2010). Eine weiterführende Forschungsarbeit mit Bezug auf das System *Vidente* kam zu dem Ergebnis, dass Vermessungsarbeiten von der unmittelbaren Visualisierung der raumbezogenen Objekte vor Ort profitieren, indem die bekannten und aufgenommenen Objekte in einem Kontext dargestellt werden können (SCHALL ET AL., 2013). Inzwischen bietet ein Partnerunternehmen des an der Entwicklung von *Vidente* beteiligten GIS-Unternehmens Grintec das Produkt *Augview* an (GRINTEC). Es zeigt Betriebsmittel eines Versorgungsnetzes als AR-Applikation auf dem Smartphone oder auf einer 2D-Karte an (GRINTEC). LI ET AL. (2019) führten eine Forschungsarbeit durch, bei der die unterirdische Infrastruktur von Wasserrohren mithilfe von Smartphones dargestellt werden sollte. Unter dem Namen *ARPV* wurde ein AR-basiertes System entwickelt, das die Sensorik der Smartphones nutzt, um den Standort und die Ausrichtung des Nutzers zu erkennen (LI ET AL., 2019). Die entsprechenden Graphiken der Wasserrohre werden auf dieser Basis in die Kamera eingeblendet (LI ET AL., 2019). Das System identifiziert dabei die Rohre vor Ort und zeigt sie mit den entsprechenden Attributen an (LI ET AL., 2019). Der in diesem Zusammenhang durchgeführter Test zeigte, dass das System eine bessere Visualisierung der Wasserrohre bietet und dadurch ein höheres Verständnis über die Wasserrohre ermöglicht als die konventionellen Methoden (LI ET AL., 2019). Zu diesem Anwendungsbereich stehen weitere Forschungsarbeiten zur Verfügung, die zu ähnlichen Ergebnissen führen. ZHANG ET AL. (2016) verglich beispielsweise zwei Methoden für ein ARGIS-basiertes Informationssystem für unterirdische Rohrleitungen und kam zu dem Ergebnis, dass mobile AR-Anwendungen beim Untersuchen und Managen von unterirdischen Rohren unterstützend wirken können, indem Rohrleitungen direkt vor Ort visualisiert erfasst werden können. LI ET AL. (2018) prüfte, inwiefern Echtzeitdaten von städtischen Rohrleitungen mit hoher Genauigkeit in einem AR-System gerendert und lagegenau visualisiert werden können. Dabei wurde eine neue Rendering-Methode entwickelt, die im Ergebnis die Lagegenauigkeit der Rohrleitungen in der Anwendung verbessert und zusätzlich ein dynamisches Rendering ermöglicht, das sich bei Bewegung des Nutzers anpasst (LI ET AL., 2018). FENAIS ET AL. (2019) entwickelten ein integriertes AR-GIS für das Abbilden und Erfassen von unterirdischer Infrastruktur mit mobilen Endgeräten mithilfe Cloud-basierter Daten. Das Ergebnis der Studie war, dass das System besonders als Kommunikations- und Informationssystem gut geeignet ist, da die Daten in der Cloud automatisch gespeichert, bearbeitet und in Echtzeit an mehreren mobilen Endgeräten mit der AR-Technologie abgebildet werden (FENAIS ET AL., 2019). In Summe validierten 86% der Nutzer das System als effizient für die Erfassung und den Informationsaustausch (FENAIS ET AL., 2019).

Die Literaturanalyse ergab, dass die in Kapitel 2.2 beschriebenen Visualisierungsmethoden Augmented Reality und Mixed Reality seit den 90er Jahren Gegenstand unterschiedlicher Forschungsprojekte ist. Vor dem Hintergrund der Forschungsfragen der vorliegenden Arbeit sind die wesentlichen Erkenntnisse aus dem derzeitigen Forschungsstand nachfolgend zusammengefasst. Der wesentliche Nutzen der Visualisierungsmethode der Augmented Reality ergibt sich aus der Möglichkeit, fiktive Daten in die Realität zu projizieren und dadurch die reale Umgebung mit relevanten, ergänzenden Daten zu überlagern. Dies erzeugt ein verbessertes Verständnis der tatsächlichen Lage von in der Realität nicht sichtbaren Objekten, sowie ein direkter Bezug von realen Objekten zu ergänzenden Informationen. Durch diesen Vorteil können Arbeitsabläufe schneller stattfinden und Kosten gesenkt werden. Die Studien aus der Baubranche und bei der Wasser- und Energieversorgung zeigen ebenfalls die Chancen von AR. Neben den Vorteilen von Augmented Reality ermitteln einige Studien wichtige Einschränkungen zur Nutzbarkeit. Eine dieser Einschränkungen liegt in der Abhängigkeit zur Datenqualität, eine weitere Einschränkung ergibt sich bei mangelnder Präzision des Standortes des Betrachters. Als Endgeräte werden in den Studien mobile Handgeräte wie Tablets, Smartphones oder am Kopf getragene Geräte wie die HoloLens erwähnt. Auffällig ist die Fokussierung auf die mobilen Handgeräte im Bereich des Bauwesens und der Wasser- und Energieversorgung, sodass die Nutzung der HoloLens in diesem Segment kaum erforscht ist.

4 Technologien im Vergleich

4.1 Anforderungen

Wie in Kapitel 2.2 ausgeführt, haben sich die Visualisierungstechnologien in vier Richtungen entwickelt. Für den weiteren Verlauf dieser Arbeit wird eine der vier Technologien Augmented Virtuality, Virtual Reality, Augmented Reality und Mixed Reality (HoloLens) ausgewählt.

Die Anforderungen für die Darstellung von Leitungsinformationen mithilfe visueller Techniken wurden aus dem Einsatzumfeld, den in Kapitel 2.1 beschriebenen rechtlichen Anforderungen und den in Kapitel 2.2 dargelegten Erkenntnissen über die Visualisierungstechnologien abgeleitet. Das Anforderungsprofil ist in folgende vier Kategorien gegliedert: Darstellung der realen Umgebung, Darstellung des virtuellen Objektes, Interpretationsfähigkeit des Objektes in der realen Umgebung und praktischer Einsatz.

Eine wesentliche Grundlage für den Einsatz der Visualisierungstechnologien ist die weitgehend originalgetreue Darstellung der Realität. Ausschlaggebend ist die Genauigkeit der zugrundeliegenden Datenbasis, nach der die 2D-Pläne erstellt werden. Eine präzise, in Zentimetern ausgedrückte Schwankungsbreite ist nicht definiert. Die virtuelle Darstellung muss demnach den gleichen Genauigkeitsgrad erreichen wie der klassische papierbasierte Plan, beziehungsweise die digitalisierten Daten, wie in Kapitel 2.1 ausgeführt. Da die reale Umgebung stets die Referenzinformation zur Lage einer Leitung ist, kann jede Abweichung in der Darstellung der Realität zu Fehlern führen. Wie in Kapitel 1.1 ausgeführt, können diese Fehler teilweise erhebliche Sachschäden und Gesundheitsgefährdungen verursachen. Die folgenden Anforderungen sind zu erfüllen:

- Originalgetreue Darstellung der realen Perspektive
- Zeitgleiche Darstellung der Realität bei Änderung der Perspektive
- Lagegenaue Darstellung des realen Umfeldes
- Erkennbarkeit relevanter Details des realen Umfeldes

Die zweite wesentliche Grundlage für den Einsatz der Visualisierungstechnologien betrifft die Darstellung der virtuellen Objekte. Analog zu der Darstellung des realen Umfeldes liegt in Abweichungen bei der Darstellung der virtuellen Objekte erhebliches Fehlerpotential mit denselben Risiken. Die folgenden Anforderungen sind für die Darstellung der virtuellen Objekte in ihrer realen Umgebung ausschlaggebend:

- Maßstabsgetreue Darstellung des virtuellen Objektes ohne perspektivische Verzerrung
- Zeitgleiche Darstellung der virtuellen Objekte bei Änderung der Perspektive
- Lagegenaue Darstellung des virtuellen Objektes in der realen Umgebung

Die dritte wesentliche Grundlage für den Einsatz der Visualisierungsmethoden ist die Anforderung, virtuell dargestellte Objekte eindeutig zu identifizieren und zu interpretieren. Fehler in diesem Bereich können ebenfalls Ursache für Schäden und Gesundheitsgefährdungen sein. Die Anforderungen sind:

- Erkennbarkeit der Leitungsart
- Darstellung wesentlicher Attribute
- Realitätsgetreue Darstellung des virtuellen Objektes
- Fehlerfreie Übertragbarkeit der Daten der virtuellen Objekte

Die vierte wesentliche Grundlage bezieht sich auf Anforderungen, die eine sichere und dauerhafte Nutzung der Technologie betreffen. Dazu gehören insbesondere:

- Darstellung folgt der Blickrichtung
- Möglichst uneingeschränkte Bewegungsfreiheit
- Vermeidung einer anhaltenden Belastung der Hände
- Sicherstellung der Akkulaufzeit
- einfache Interaktion des Benutzers mit der Anwendung
- Positionsunabhängigkeit des Anwenders beim Start der Anwendung
- Blickrichtungsunabhängigkeit des Anwenders beim Start der Anwendung

4.2 Vergleich

Aus den oben beschriebenen Anforderungen sind die entsprechenden technischen Anforderungen abgeleitet und in Tabelle 1 dargestellt.

Tabelle 1: Anforderungen und technische Anforderungen an die Technologie

Anforderung	Technische Anforderung
Realität	
Originalgetreue Darstellung der realen Perspektive	Verzerrungsfreie, optische Durchsicht
Zeitgleiche Darstellung der Realität bei Änderung der Perspektive	Verzerrungsfreie, optische Durchsicht
Lagegenaue Darstellung des realen Umfeldes	Verzerrungsfreie, optische Durchsicht
Erkennbarkeit relevanter Details des realen Umfeldes	Genügend Lichtdurchlass
Virtuelles Objekt	
Maßstabsgetreue Darstellung des virtuellen Objektes ohne perspektivische Verzerrung	Darstellungsgenauigkeit des virtuellen Objektes in einem Bereich von 1m bis 5m
Zeitgleiche Darstellung der virtuellen Objekte bei Änderung der Perspektive	Dynamisches Rendering des virtuellen Objektes in Echtzeit
Lagegenaue Verortung des virtuellen Objektes in der realen Umgebung	Anbindung an einen standortgenauen GPS-Empfänger
Identifizierbarkeit und Interpretation des virtuellen Objektes	
Erkennbarkeit der Leitungsart	Möglichkeit zur farblichen Differenzierung auf Basis der Leitungsart
Darstellung wesentlicher Attribute	Möglichkeit der Darstellung zusätzlicher Information
Realitätsgetreue Darstellung des virtuellen Objektes	Interaktion des virtuellen Objektes mit der realen Umgebung
Fehlerfreie Übertragbarkeit der Daten der virtuellen Objekte	Speichermedium in ausreichender Größe
Nutzbarkeit	
Darstellung folgt der Blickrichtung	Integrierter Bewegungssensor
Möglichst uneingeschränkte Bewegungsfreiheit	Gerät ist kabellos und autark
Vermeidung einer anhaltenden Belastung der Hände	Gerät ist am Kopf befestigt
Sicherstellung der Akkulaufzeit	Akkulaufzeit 3 Stunden
Einfache Interaktion des Benutzers mit der Anwendung	Intuitive Benutzeroberfläche mit möglichst wenig manuellen Eingriffen
Positionsunabhängigkeit des Anwenders beim Start der Anwendung	Anbindung an einen standortgenauen GPS-Empfänger
Blickrichtungsunabhängigkeit des Anwenders beim Start der Anwendung	Dynamisches Rendering des virtuellen Objektes in Echtzeit

Quelle: eigene Erhebung

Auf Basis der technischen Anforderungen gemäß Tabelle 1 werden nachfolgend die Visualisierungstechnologien bewertet. Die Grundlage der Bewertungen bilden die Ergebnisse der Kapitel 2 und Kapitel 3.

Als grundlegende Anforderung gilt, dass die Visualisierungsart die reale Umgebung über eine verzerrungsfreie, optische Durchsicht mit genügend Lichtdurchlass darstellen können muss, damit eine originalgetreue, lagegenaue und detailgetreue Darstellung der Realität ohne Verzögerung gewährleistet ist. Wie in Kapitel 2.2 beschrieben, verfügen sowohl die Technologie der Augmented Virtuality als auch die Technologie der Virtual Reality nicht über eine verzerrungsfreie, optische Durchsicht mit genügend Lichtdurchlass und können die Anforderung daher nicht erfüllen. Augmented Reality projiziert lediglich ein reales Objekt über eine Kamera in die virtuelle Umgebung und Virtual Reality verfügt über keine Durchsicht. Beide Technologien verfügen somit nicht über die Möglichkeit, die reale Umgebung darzustellen. Da die vorliegende Arbeit auf dieser Anforderung basiert, scheiden die beiden Technologien der Augmented Virtuality und der Virtual Reality bereits aus und werden im folgenden Vergleich nicht näher betrachtet. Die Technologie der Augmented Reality kann diese Anforderungen abhängig von dem jeweiligen Endgerät erfüllen, da es für diese Technologie sowohl Endgeräte mit optischer Durchsicht als auch Endgeräte über eine Kamera mit Video-Durchsicht gibt. Die Technologie der Mixed Reality über die Microsoft HoloLens kann die Anforderung erfüllen, da diese über eine optische Durchsicht verfügt.

Die nachfolgenden drei Anforderungen beziehen sich auf das virtuelle Objekt. Damit das virtuelle Objekt maßstabsgetreu ohne Verzerrung dargestellt werden kann und sich in Echtzeit bei Änderung der Perspektive lagerichtig anpasst, muss es in einem Bereich von 1 Meter bis 5 Meter lagerichtig und genau dargestellt werden können, sowie ein dynamisches Rendering möglich sein. Diese beiden Anforderungen können sowohl von der Technologie der Augmented Reality als auch von der Technologie der Mixed Reality erfüllt werden, sofern die Anwendung entsprechend entwickelt wurde und keine logischen Lücken bei der Anwendung vorliegen. Zusätzlich muss das virtuelle Objekt in der realen Umgebung lagegenau verortet werden können. Diese Anforderung kann nur durch die Möglichkeit der Anbindung eines hochgenauen GPS-Gerätes erfolgen, was für gewöhnlich nicht in dem benötigten Genauigkeitsgrad in den Endgeräten integriert ist. Beide Technologien verfügen aber über die Möglichkeit, programmatisch ein entsprechendes GPS-Gerät anzubinden und können diese Anforderung daher erfüllen.

Die Identifizierbarkeit und die Interpretation des virtuellen Objektes in der realen Umgebung werden durch die nächsten vier technischen Anforderungen gewährleistet. Ein wesentlicher Aspekt dabei ist, dass die Leitungsart erkennbar ist und wesentliche, zusätzliche Attribute dargestellt beziehungsweise eingeblendet werden können, damit der Nutzer sofort erkennt, um welches Objekt es sich handelt und er die nötigen Informationen abgreifen kann. Diese Anforderungen werden durch die technischen Anforderungen der Möglichkeit zur farblichen Differenzierung auf Basis der Leitungsart und der Möglichkeit der Darstellung zusätzlicher Information erfüllt. Sowohl die Technologie der Augmented Reality als auch der Mixed Reality können diese Anforderungen erfüllen. Eine weitere Anforderung dieser Kategorie ist die realitätsgetreue Darstellung des virtuellen Objektes in der realen Umgebung. Dies wird technisch über die Interaktion des virtuellen Objektes mit der realen Umgebung umgesetzt, damit die Objekte realitätsgetreu in die reale Welt projiziert werden können. Wie in Kapitel 2.2 herausgearbeitet wurde, ist die HoloLens mit dieser Funktion einzigartig und kann die Anforderung als einzige Technologie erfüllen, da sie über Sensorik verfügt, welche Oberflächen und Strukturen aus der realen Welt scannt und dadurch das virtuelle Objekt lagerichtig und logisch in der Realität platzieren kann. Die letzte Anforderung in dieser Kategorie ist die fehlerfreie Übertragbarkeit der virtuellen Objekte, damit diese korrekt in der Anwendung dargestellt werden können. Über ein entsprechendes

Speichermedium in ausreichender Größe kann diese Forderung technisch gewährleistet werden. Beide Technologien können diese Anforderung erfüllen, da sie über genügend Speicherplatz verfügen. Bei Augmented Reality ist dies jedoch abhängig vom jeweiligen Endgerät. Zusätzlich können die Daten bei beiden Technologien auch auf einen weiteren Computer ausgelagert werden und immer nur die benötigten Daten übertragen werden, wodurch sich eine hohe Speicherverfügbarkeit ergibt.

Die letzten fünf technischen Voraussetzungen sind der Kategorie der Nutzbarkeit des Gerätes zuzuordnen. Dabei gelten als wesentliche technische Anforderungen, dass das Gerät über einen integrierten Bewegungssensor verfügt, kabellos und autark ist sowie am Kopf befestigt ist. Hierüber wird eine uneingeschränkte Bewegungsfreiheit des Nutzers gewährleistet, während er seine Hände frei hat und die Darstellung direkt der Blickrichtung folgt. Diese drei technischen Anforderungen implizieren, dass es sich um ein mobiles Kopfgerät in Form einer Brille handeln muss. Sowohl die HoloLens für Mixed Reality als auch die Technologie der Augmented Reality können diese Anforderungen erfüllen. Die HoloLens ist ein am Kopf getragener holographischer Computer in Form einer Brille und Augmented Reality kann sowohl in Form von Handgeräten als auch in Form von Kopfgeräten genutzt werden. Somit ist die Erfüllung der Anforderung bei Augmented Reality abhängig vom entsprechenden Endgerät. Eine weitere technische Anforderung ist die Akkulaufzeit von drei Stunden am Stück, damit eine robuste und ausreichende Akkulaufzeit für einen Arbeitstag im Außendienst gewährleistet ist. Da das Gerät nicht dauerhaft den Tag über genutzt wird und während Zeiten der Nichtbenutzung ausgeschaltet wird, ergibt sich mit einer Akkulaufzeit von drei Stunden ein ausreichender Zeitraum. Abhängig vom jeweiligen Endgerät kann die Technologie der Augmented Reality diese Anforderungen erfüllen. Die HoloLens verfügt ebenfalls über eine ausreichende und robuste Akkulaufzeit und erfüllt daher ebenfalls die Anforderung. Der letzte zu berücksichtigende Punkt ist die Gewährleistung einer einfachen Interaktion des Benutzers mit der Anwendung. Dies wird technisch durch eine intuitive Benutzeroberfläche der Anwendung mit möglichst wenig manuellen Eingriffen ermöglicht. Da diese Anforderung nur programmatisch gewährleistet werden kann, können die Technologien diese Anforderungen erfüllen, sofern die Anwendung entsprechend logisch und intuitiv entwickelt wurde. Tabelle 2 zeigt die Übersicht der Umsetzbarkeit der zusammengefassten technischen Anforderungen nach der jeweiligen Technologie.

Tabelle 2: Erfüllung der technischen Anforderungen nach Visualisierungstechnologie

Technische Anforderung	Augmented Virtuality	Virtual Reality	Augmented Reality	Mixed Reality (HoloLens)
Verzerrungsfreie, optische Durchsicht mit genügend Lichtdurchlass	Nein	Nein	Abhängig von Endgerät	Ja
Darstellungsgenauigkeit des virtuellen Objektes in einem Bereich von 1m bis 5m	Nein	Nein	Ja	Ja
Dynamisches Rendering des virtuellen Objektes in Echtzeit	Nein	Nein	Ja	Ja
Möglichkeit zur farblichen Differenzierung auf Basis der Leitungsart	Nein	Ja	Ja	Ja
Möglichkeit der Darstellung zusätzlicher Information	Ja	Ja	Ja	Ja
Interaktion des virtuellen Objektes mit der realen Umgebung	Nein	Nein	Nein	Ja
Speichermedium in ausreichender Größe	Abhängig von Endgerät	Abhängig von Endgerät	Abhängig von Endgerät	Ja
Integrierter Bewegungssensor	Nein	Abhängig von Endgerät	Abhängig von Endgerät	Ja
Gerät ist kabellos und autark	Abhängig von Endgerät	Abhängig von Endgerät	Abhängig von Endgerät	Ja
Gerät ist am Kopf befestigt	Nein	Ja	Abhängig von Endgerät	Ja
Akkulaufzeit 3 Stunden	Abhängig von Endgerät	Abhängig von Endgerät	Abhängig von Endgerät	Ja
Intuitive Benutzeroberfläche mit möglichst wenig manuellen Eingriffen	Ja	Ja	Ja	Ja

Quelle: eigene Erhebung

Aus dem Vergleich der vier Visualisierungstechnologien geht hervor, dass Mixed Reality mit der HoloLens als Endgerät all einzige alle technischen Anforderungen erfüllt. Aufgrund der uneingeschränkten Bewegungsfreiheit und der realitätsgetreuen Projektion der Hologramme durch die Sensoren wird für den weiteren Verlauf der Arbeit unter den vier Technologien die HoloLens ausgewählt.

5 Methodik

5.1 Vorgehensweise

Wie in Kapitel 1.2 beschrieben, beziehen sich die Forschungsfragen im Rahmen dieser Arbeit auf den praktischen Einsatz einer neuen Visualisierungstechnologie. Auf Basis der in den vorherigen Kapiteln dargestellten Rahmenbedingungen, Anforderungen und zur Verfügung stehenden Technologien wurde die HoloLens als Technologieträger ausgewählt. Das nachfolgende Kapitel beschreibt die Entwicklung eines Prototyps im Bereich der Wasserversorgung. Die Aufgabenstellung besteht darin, eine Wasserleitung in einem definierten Untersuchungsgebiet auf die Erdoberfläche projiziert darzustellen. Dabei wird die tatsächliche Verlegetiefe der Leitung nicht miteinbezogen, da sie sonst nicht sichtbar wäre. Eine Ausprägung der Tiefe über ein Attribut würde den Entwicklungsaufwand stark erhöhen und so über den Rahmen dieser Arbeit hinaus gehen. Mit der Entwicklung des Prototyps soll die Validität der Forschungsfrage überprüft werden. Die folgenden Schritte beschreiben die Vorgehensweise:

1. Untersuchungsgebiet definieren
2. Pflichtenheft erstellen
3. Ausgangsdaten aufbereiten
4. Anwendung entwickeln
5. Ergebnisse vorstellen

Zur Entwicklung des Prototyps sind drei wesentliche Schritte notwendig. Dabei dienen die ersten beiden Schritte der Datenaufbereitung und der dritte Schritt der Entwicklung der Anwendung. Abbildung 6 zeigt schematisch die Vorgehensweise in der Übersicht.

Im ersten Schritt muss das als DWG-Datei bereitgestellte digitale Höhenmodell mithilfe der Software ArcGIS Pro in ein Höhenraster umgewandelt werden, damit es mit den zweidimensionalen Leitungsdaten überlagert und verschnitten werden kann. Die als Shapefiles bereitgestellten Ausgangsdaten werden über die Software FME verarbeitet und über die Verschneidung mit dem Höhenmodell auf die Oberfläche projiziert und in 3D Daten transformiert. Im nächsten Schritt wurden die nun in einer File Geodatabase vorliegenden Daten in die Software City Engine eingeladen, um sie in das für die Software Unity notwendige Datenformat FBX zu transformieren. Diese beiden Schritte werden in Kapitel 5.5 im Detail beschrieben und sind in Abbildung 6 auf der linken Hälfte dargestellt.

Der dritte Schritt ist die Entwicklung des Prototyps. Dies erfolgt mithilfe der Spiele-Entwicklungsumgebung Unity, da diese über eine einfach zu bedienende Benutzeroberfläche und durch für AR und MR zugeschnittene Toolkits eine erleichterte Entwicklung ermöglicht. Es ist lediglich die Programmierung von Skripten notwendig, die dann den entsprechenden Objekten hinzugefügt werden. Diese Skripte werden mit Visual Studio in der Programmiersprache C# geschrieben. Nach der Entwicklung in Unity wird die Anwendung in C++ konvertiert und als Visual Studio Projekt exportiert. Aus Visual Studio heraus wird die Anwendung anschließend als App auf die HoloLens übertragen. Dieser Schritt ist in Abbildung 6 auf der rechten Hälfte dargestellt und wird in Kapitel 5.6 im Detail beschrieben.

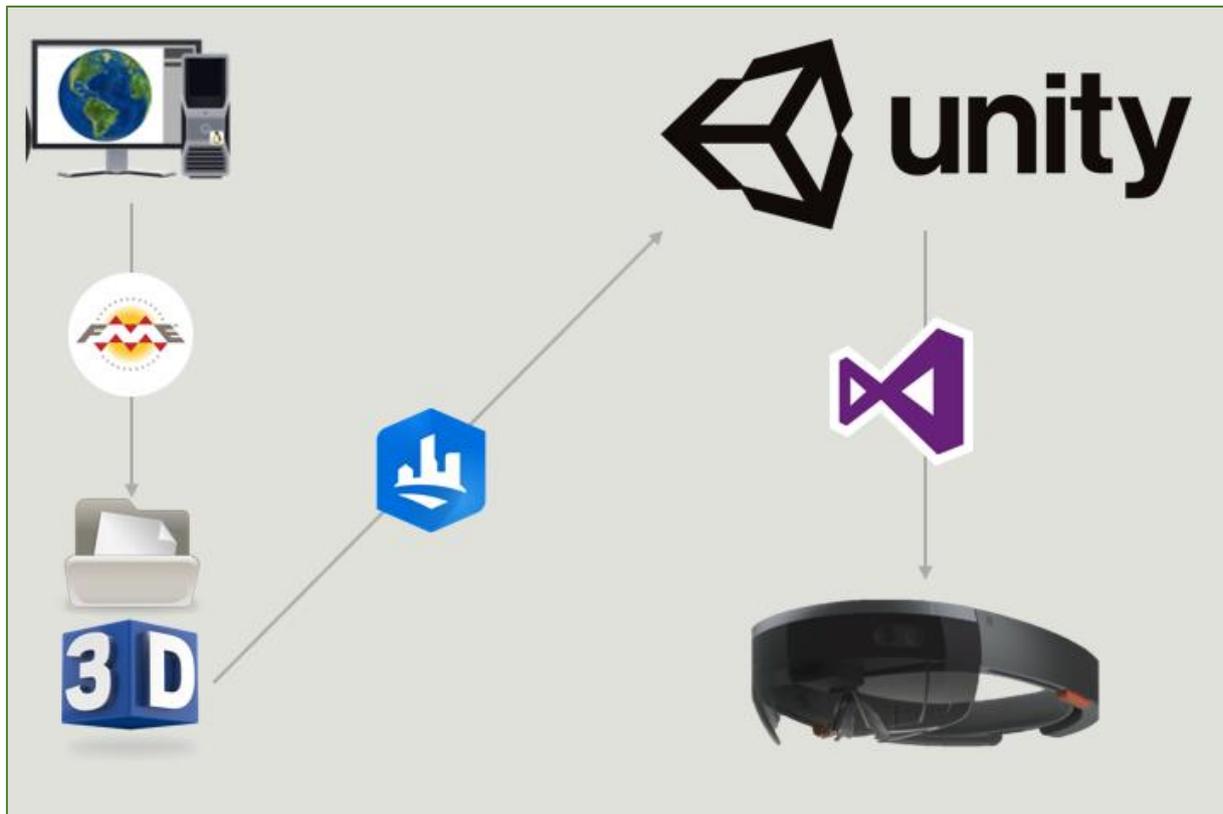


Abbildung 6: Vorgehensweise der Methodik

Quelle: eigene Darstellung nach KissCCO (2019), FREE ICON LIBRARY (2019), CLEANPNG.COM (2018), Screenshots aus der jeweiligen Software

5.2 Untersuchungsgebiet

Zur Auswahl eines geeigneten Studiengbietes werden die nachfolgend beschriebenen Anforderungen definiert. Der Arbeitsbereich der holographischen Darstellung wird auf maximal 5 Meter beschränkt, da die Visualisierungstechnologie bei größeren Entfernungen systembedingt eine zu große Abweichung aufweist. Um die Grenzbereiche ausreichend bewerten zu können, wird das Untersuchungsgebiet auf 10 x 20 m festgelegt. Die zugrundeliegende Berechnung geht davon aus, dass der Betrachter einen Blickwinkel von 180 Grad hat und auf jeder Seite eine Sicherheitsmarge von 5 Metern aufgeschlagen wird, siehe Abbildung 7.

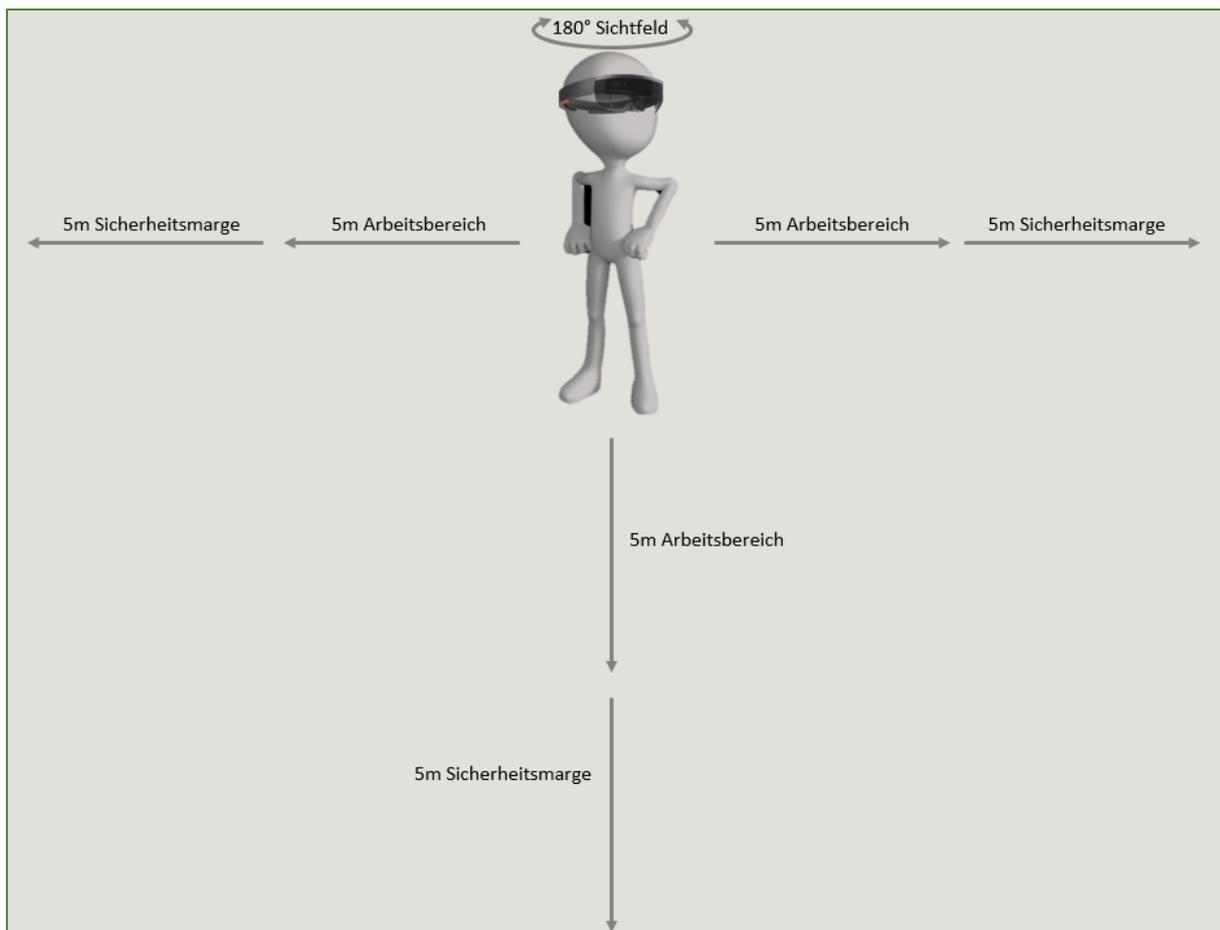


Abbildung 7: Ermittlung des Untersuchungsgebietes

Quelle: eigene Darstellung nach CLEANPNG.COM (2018) und THEDIGITALARTIST (2017)

Das Studiengbiet soll unweit des Entwicklungsstandortes der Autorin liegen. Weiterhin besteht eine Anforderung darin, dass sich in dem auszuwählenden Studiengbiet mindestens eine unterirdische Versorgungsleitung befindet und dafür ausreichend digitalisierte Daten zur Verfügung stehen. Eine Eingrenzung bei der Auswahl des Studienobjektes besteht bezüglich der zu verarbeitenden Datenmenge. Die Aufbereitung visueller Daten kann sehr umfangreiche Berechnungen nach sich ziehen. Für die nachfolgende Entwicklung und Datenaufbereitung muss daher ein einzelnes Objekt ausgewählt werden. Zusätzlich muss das Objekt im 2D-Plan lagegenau vorliegen, um eine Neuerfassung zu vermeiden.

Auf Basis der oben geschilderten Anforderungen fällt die Auswahl des Untersuchungsgebietes auf ein Gelände um das Betriebsgebäude der Firma BARAL. Wie in Abbildung 8 dargestellt, befindet sich das Betriebsgebäude im Gewerbepark Orschel-Hagen, nördlich der Stadt Reutlingen, Ludwig-Erhard-Straße 54. Das Studiengebiet ist in Abbildung 9 durch ein rotes Rechteck gekennzeichnet. Das ausgewählte Studienobjekt gehört zur Sparte Wasser, da dies lagegenau zur Verfügung steht. Abbildung 10 zeigt alle in dem ausgewählten Studiengebiet verfügbaren Leitungsnetze, die Sparte Wasser ist hellblau dargestellt. Aus diesem Grund muss das Objekt im Prototyp ebenfalls in blau dargestellt werden, um eine eindeutige Interpretation zu ermöglichen. Die Daten wurden von der Firma FairNetz GmbH als 2D Shapefiles im Koordinatensystem ETRS89, UTM Zone 32N zur Verfügung gestellt.

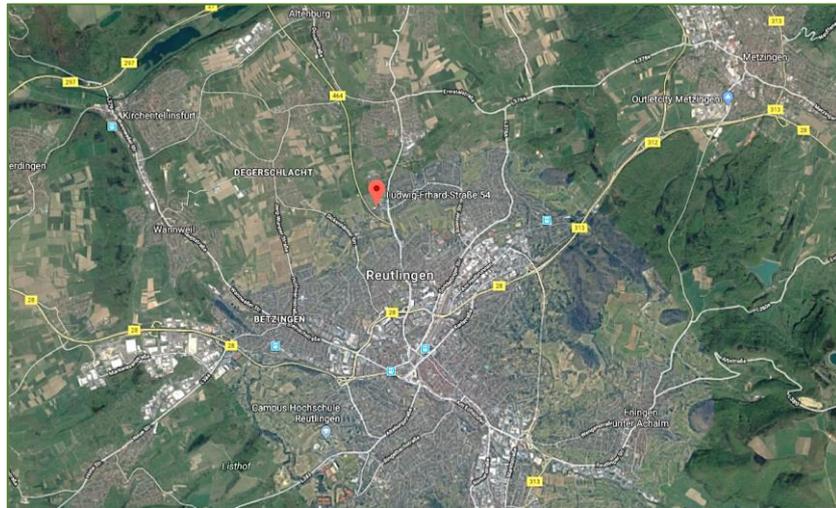


Abbildung 8: Untersuchungsgebiet

Quelle: GOOGLE INC



Abbildung 9: Abgrenzung des Untersuchungsgebietes

Quelle: GOOGLE INC



Abbildung 10: Karte des Untersuchungsgebietes mit Leitungen
Quelle: Eigene Darstellung mit Daten der FairNetz GmbH

5.3 Hardware und Software

5.3.1 Endgeräte

Wie in Kapitel 4.2 ausgeführt, wurde die HoloLens für die Entwicklung des Prototyps zur Darstellung der Leitungen bzw. Rohre ausgewählt. Dafür wurde die erste Generation der Microsoft HoloLens zur Verfügung gestellt. Es handelt sich um eine am Kopf getragene Brille, die gemäß der in Kapitel 3 vorgestellten Definition von Microsoft der Mixed Reality Kategorie zugeordnet wird. Sie ist der weltweit erste vollständig losgelöste holografische Computer (MICROSOFT, 2020B). Der Computer hat eine Intel-32-Bit Architektur mit TPM 2.0-Unterstützung sowie eine benutzerdefinierte holographische Verarbeitungseinheit (HPU 1,0) mit einem Arbeitsspeicher von 2 GB und 64 GB Flash (MICROSOFT, 2020B). Das ermöglicht eine schnelle Verarbeitung und Darstellung der Hologramme. Da dieser Computer ein Windows 10 Betriebssystem enthält, sind Applikationen der Universal Windows Plattform (UWP) lauffähig (MICROSOFT, 2020F). Aus diesem Grund ist der Prototyp als UWP-App bereitzustellen, was den Einsatz der dafür geeigneten Entwicklungswerkzeuge erfordert. Die Apps der Universal Windows Plattform sind universale Apps, die unabhängig vom Endgerät auf allen Windows 10 Betriebssystemen durch eine gemeinsame und einheitliche API-Oberfläche lauffähig sind (MICROSOFT, 2020F). UWP-Apps können in den Sprachen C#, C++, Visual Basic und JavaScript programmiert werden (MICROSOFT, 2020F).

Die Darstellung der virtuellen Objekte, im vorliegenden Fall eine Wasserleitung, erfolgt mithilfe moderner Optik und Sensorik. Die Objekte erscheinen in Form von 3D Hologrammen auf der sich vor dem Auge befindenden Mattscheibe. Dabei werden mehrere Sensoren, ein Infrarotlaser, eine hochauflösende Kamera, Beschleunigungsmesser und Mikrofone mit dem integrierten Computer der HoloLens kombiniert (TREPPER ET AL., 2017). Die Bilderzeugung erfolgt durch zwei hochauflösende 16:9 Lichtgeneratoren (TREPPER ET AL., 2017). Die menschliche Wahrnehmung erfolgt durch Reflektion auf der Netzhaut (TREPPER ET AL., 2017). Zusätzlich führt das augenbasierte Rendering durch eine automatische Pupillen-Entfernungs-Kalibrierung zu einer scharfen Abbildung des virtuellen Objektes (TREPPER ET AL., 2017). Eine Nachverfolgung des Blickes, Gesteneingaben und Sprachunterstützung gewährleisten eine Interaktion des Nutzers mit den Hologrammen (MICROSOFT, 2020B). Zudem wird die Realität über transparente, holographische Linsen dargestellt, wodurch die reale Umgebung verzerrungsfrei und in Echtzeit wahrgenommen wird. Die uneingeschränkte, reale Sicht ermöglicht eine gefahrlose Bewegung vor Ort. Die Dauer der Außendiensteinsätze mit der ersten Generation der HoloLens sind durch die Akkulaufzeit von zwei bis drei Stunden begrenzt (MICROSOFT, 2020B). In diesem Zusammenhang ist festzuhalten, dass die HoloLens während Zeiten der Nichtbenutzung ausgeschaltet wird, sodass sich im praktischen Einsatz deutlich längere Einsatzzeiträume ergeben. Die Unabhängigkeit des Computers führt dazu, dass keine weitere Abhängigkeit zu anderen Geräten besteht.

Weitere Komponenten der ersten Generation der HoloLens sind ein Stirnband sowie ein Riemen über dem Kopf zum Befestigen der HoloLens. Abbildung 11 zeigt die HoloLens im Einsatz. Sie kann auch zusammen mit einer gewöhnlichen, optischen Brille eingesetzt werden. Abbildung 12 zeigt die HoloLens mit ihren Komponenten.



Abbildung 11: Die HoloLens im Einsatz
Quelle: eigene Darstellung

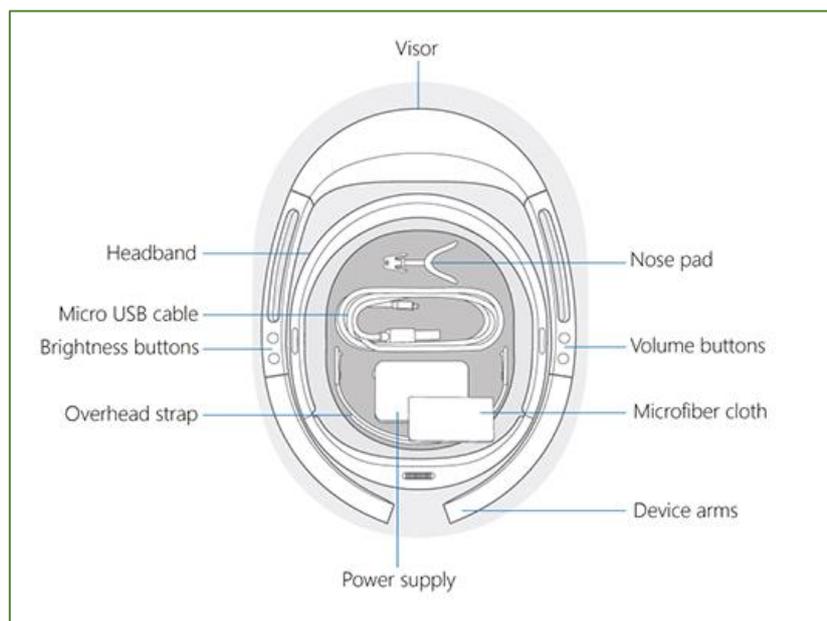


Abbildung 12: Die HoloLens und ihre Komponenten
Quelle: MICROSOFT (2020A)

Wie in Kapitel 4.2 erläutert, ist der Einsatz eines GPS-Systems nötig, um den Standort des Nutzers präzise zu ermitteln. Da die HoloLens über kein integriertes GPS verfügt, wird zusätzlich ein externes GPS-Gerät verwendet. Damit kann der Benutzer verortet werden und die Objekte werden lagerichtig platziert. Zum Einsatz kommt das Blumax GPS-710, da es mit den Maßen 90(L)x45(B)x12,5(H)mm und dem Gewicht von 57g leicht ist und so ermüdungsfrei mitgeführt werden kann (BLUMAX). Der GPS-Empfänger Blumax GPS-710 ist in Abbildung 13 dargestellt.



Abbildung 13: Das Blumax GPS-710

Quelle: eigene Darstellung

Nach Aussage des Herstellers ermöglicht das Gerät einen schnellen Empfang und eine Positionierung auch bei schlechten Verhältnissen (BLUMAX). Das GPS-Gerät empfängt 20 Kanäle und sendet die NMEA-Sätze GGA, GSA, GSV, RMC, VTG und GLL einmal pro Sekunde. NMEA ist ein Standard zur Protokollierung von GPS-Daten und wird in vielen gängigen GPS-Geräten eingesetzt (NMEA, 2011). Für die vorliegende Arbeit wird der NMEA-Satz GGA (Global Positioning System Fix Data, Time, Position and fix related data for a GPS receiver) benötigt. In der nachfolgenden Abbildung ist die Struktur des NMEA-Satzes dargestellt.

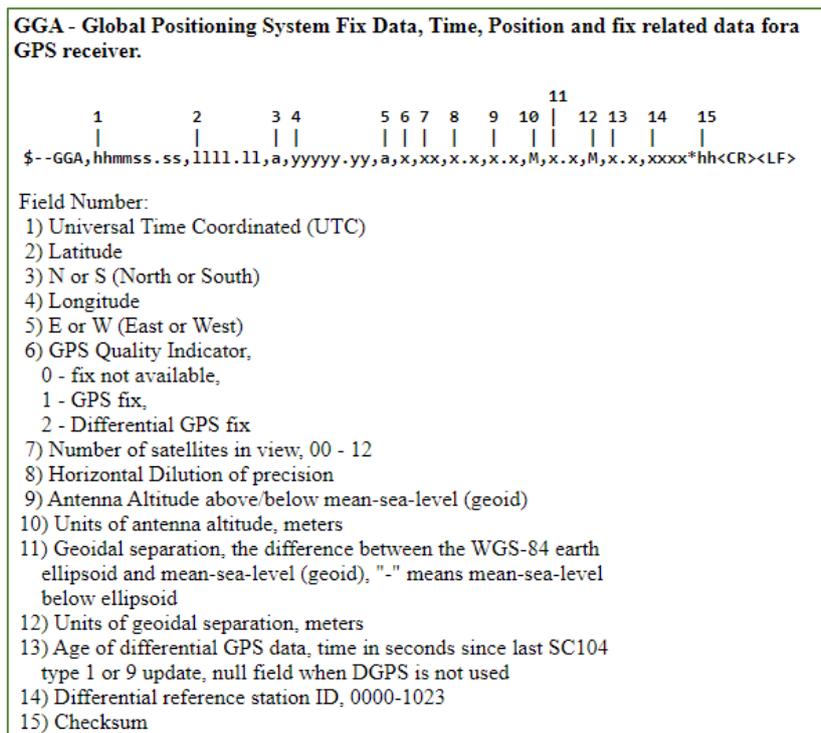


Abbildung 14: Der NMEA-Satz GGA

Quelle: NMEA (2013)

Der NMEA-Satz besteht aus 15 Feldern. Für die vorliegende Arbeit müssen die Werte für Latitude (Feld 2) und Longitude (Feld 4) übernommen werden, da dieser Schritt ein wesentlicher Bestandteil bei der Entwicklung des Prototyps ist. Die Felder 1, 3 und 5 können ignoriert werden, denn die Zeit wird für die zu entwickelnde Anwendung nicht benötigt und es ist bekannt, dass sich das Untersuchungsgebiet westlich des Nullmeridians auf der Nordhalbkugel befindet. Somit können die ausgegebenen positiven Koordinaten einfach übernommen werden. Die übrigen Felder beinhalten ebenfalls keine für die Anwendung relevanten Informationen. Neben dem NMEA-Satz ist die eindeutige Bestimmung des verwendeten Bezugssystems notwendig. Im vorliegenden Fall ist dies WGS-84, da es das Bezugssystem der GPS-Satelliten ist (LANDESAMT FÜR GEOINFORMATION UND LANDENTWICKLUNG BADEN-WÜRTTEMBERG, 2020). Das Datenformat lautet Grad, Minuten.

5.3.2 Software zur Datenaufbereitung

Zur Übertragung der Daten des Ausgangsformates 2D Shapefile in das Zielformat des 3D-Modells FBX werden mehrere Softwareprogramme eingesetzt. Die Auswahl richtet sich nach den im GIS-Bereich verbreitet eingesetzten Werkzeugen und der im Rahmen dieser Arbeit zur Verfügung gestellten Programme.

Das Softwareprogramm *ArcGIS Pro* ist das neue Desktop GIS des Unternehmens Esri und bietet zahlreiche Werkzeuge zur Erstellung professioneller Karten sowie zur Durchführung komplexer Analysen in 2D und 3D (ESRI INC, A). Zusätzlich stellt das Programm Werkzeuge zum Datenmanagement und zur Datenverarbeitung und -transformation zur Verfügung. Dabei können Esri-Ausgangsdaten in andere Datenformate sowie Datenformate fremder Hersteller in Esri-Datenformate transformiert werden.

Die Software *FME* (Feature Manipulation Engine) ist eine Datenintegrationsplattform des Unternehmens Safe Software mit Unterstützung räumlicher Daten (SAFE SOFTWARE INC., 2020A). Die Software beinhaltet Werkzeuge zur automatisierten Manipulation, Transformation und Konvertierung von Daten und Datenformaten (SAFE SOFTWARE INC., 2020A). Die übersichtliche Benutzeroberfläche bietet dabei die Möglichkeit, ganze und aus mehreren Teilarbeitsschritten bestehende Workflows in einer Übersicht abzubilden und so den Überblick komplexer Workflows zu behalten. Das Programm eignet sich daher gut für komplexere Datentransformationen, bei denen mehrere Zwischenschritte notwendig sind.

Das dritte für die Datenaufbereitung genutzte Softwareprogramm ist *CityEngine* des Unternehmens Esri. Es ist eine 3D-Modellierungssoftware zur Erstellung großer, interaktiver und immersiver städtischer Umgebungen in kürzerer Zeit als traditionelle Modellierungstechniken (ESRI INC, B). Die Software konsumiert unter anderem Esri-Datenformate und unterstützt die Anwender dabei, 3D-Modelle vorzubereiten und zu bearbeiten, um sie anschließend in Spiele-Engines exportieren zu können (ESRI INC, B).

5.3.3 Entwicklungsumgebungen

Wie in Kapitel 1.2 beschrieben, besteht ein wesentlicher Teil der vorliegenden Arbeit darin, einen Prototyp einer Anwendung zu entwickeln. Damit soll anhand eines vereinfachten Beispiels die

grundsätzliche Möglichkeit nachgewiesen werden, Mixed Reality für die Visualisierung von unterirdischen Leitungsnetzen vor Ort einzusetzen. Nachfolgend werden die eingesetzten Softwarewerkzeuge in ihren Grundzügen beschrieben, die detaillierte Erläuterung der Entwicklungsschritte erfolgt in Kapitel 5.6.

Als Hauptentwicklungsumgebung wurde die Software Unity von Unity Technologies in der Version 2019.4 ausgewählt. Microsoft empfiehlt, diese Entwicklungsumgebung für die Entwicklung von Anwendungen für die erste Generation der HoloLens zu nutzen (MICROSOFT, 2020A). Die Entwicklungsumgebung besitzt einen hohen Verbreitungsgrad und wird unter anderem zur Erstellung und Gestaltung von Computerspielen eingesetzt (THEIS, 2018). Daher orientiert sich die Bezeichnung von Objekten und Funktionen an dem Einsatzgebiet des Gamings, darzustellende Objekte werden beispielsweise *Game Objects* genannt. Im weiteren Verlauf der Ausführungen werden die Bezeichnungen der Entwicklungsumgebung jeweils im Original genannt. Neben dem Gaming-Bereich hat sich diese Entwicklungsumgebung für die Entwicklung von 3D-Anwendungen, insbesondere im Bereich von Virtual Reality, Augmented Reality und Mixed Reality etabliert. Nach UNITY TECHNOLOGIES (2020) erfordert die Software folgende Systemvoraussetzungen an die Hardware für Windows:

- Windows 7 (SP1+) oder Windows 10 als 64-Bit Version
- CPU mit X64 Architektur und Unterstützung des SSE2 Befehlssatzes
- DX10, DX11 und DX12 fähige GPU

Die nachfolgenden Ausführungen beschreiben den Aufbau der Benutzeroberfläche sowie die für diese Arbeit relevanten Komponenten der Software, um die Nachvollziehbarkeit der Arbeitsschritte zu verbessern.

Das Entwicklungsprogramm stellt mit dem *Unity Editor* die zentrale Entwicklungsoberfläche zur Verfügung. Der *Unity Editor* enthält eine Menüleiste am oberen Rand, womit die jeweiligen Funktionen aufgerufen werden (THEIS, 2018). Der Hauptbildschirm ist in mehrere Bereiche unterteilt, sogenannte Views, die frei angeordnet werden können (THEIS, 2018). Dies ermöglicht eine optimierte Übersicht und bildet damit eine wichtige Funktion, besonders bei steigender Anzahl der darzustellenden Elemente. Die großen Views in der Mitte des Editors sind die *Scene View* und die *Game View*. In der *Scene View* werden die *Game Objects*, zum Beispiel die Leitungsstränge, in der Umgebung positioniert. Die *Game View* bietet die Vorschau auf die jeweilige Umgebung (Theis, 2018). Am unteren Rand des Editors befindet sich die *Projekt View*, sie zeigt die vom jeweiligen Projekt verwendeten Datenordner. Einer dieser Datenordner enthält die darzustellenden Elemente des Projektes und ist mit dem Namen *Assets* gekennzeichnet (THEIS, 2018). Zu den *Assets* gehören die *Materials* zur Darstellung der Textur, die *Game Objects*, die *Scenes*, die *Scripts* und die *Plugins*. Die *Hierarchy View* am linken Rand des Editors beinhaltet die hierarchisch aufgelisteten *Game Objects* der *Scene* (THEIS, 2018). Eine weitere Oberfläche des Editors ist die *Inspector View* am rechten Rand. Sie bietet die Bearbeitung der Eigenschaften einzelner *Game Objects* und das Hinzufügen zusätzlicher Komponenten. Es werden kontextabhängig jeweils die Details der in der *Hierarchy View* ausgewählten *Game Objects* angezeigt (THEIS, 2018). Ergänzend befindet sich am unteren Rand des Editors die *Console View*. Sie zeigt Kontrollausgaben des Programmes sowie Fehlermeldungen und stellt damit ein wichtiges Werkzeug zur Qualitätssicherung dar. Abbildung 15 zeigt den im Rahmen dieser Arbeit verwendeten Aufbau der Oberfläche.

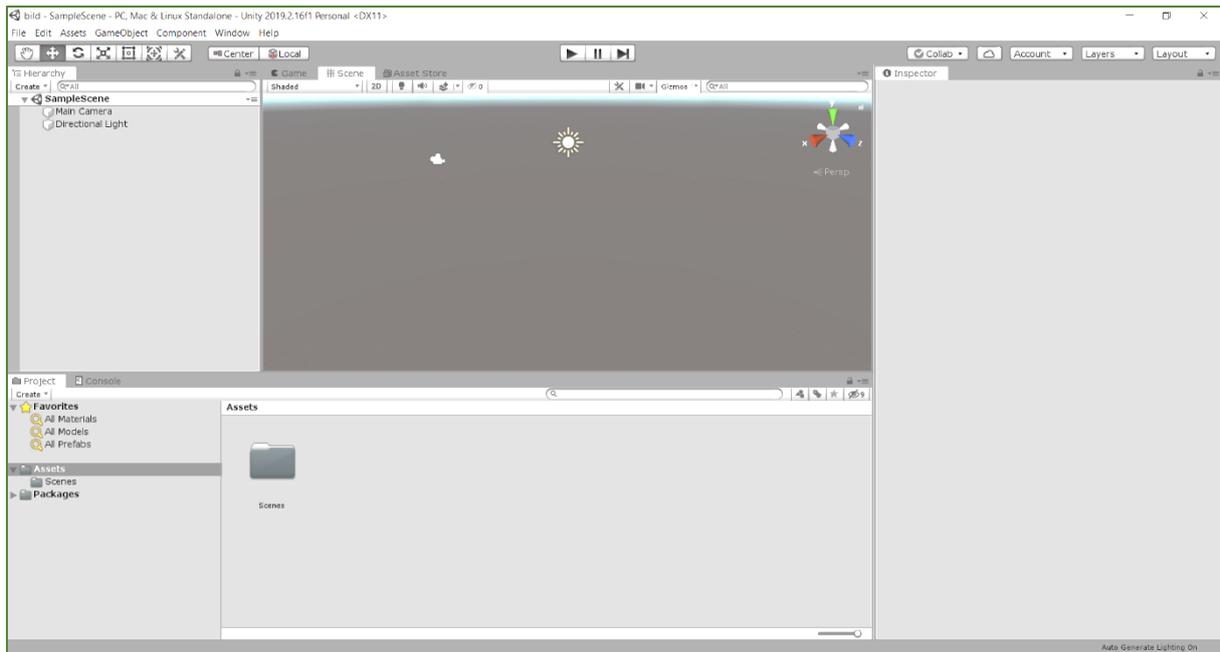


Abbildung 15: Aufbau der Oberfläche des Programms Unity
Quelle: Eigene Darstellung

Die Entwicklungsumgebung Unity kann ausschließlich 3D-Modelle der Formate FBX, DAE (Collada), 3DS, DXF und OBJ verarbeiten (UNITY TECHNOLOGIES, 2019). FBX ist dabei das Austauschformat von Unity (ESRI INC., 2020). Für die vorliegende Entwicklungsaufgabe wird FBX genutzt, da dieses Modell aus Esri Ausgangsdaten über Esri Software erstellt werden kann, sodass die Koordinaten des Mittelpunkts beibehalten werden. Das FBX-Format ist das systemeigene 3D-Format des MotionBuilder von Autodesk, ein Programm zur Erstellung, Bearbeitung und Verschmelzung von Motion Capture- und Keyframe-Animationen (AUTODESK INC., 2020). Durch FBX-Dateien ist die Interoperabilität zwischen den meisten 3D-Softwareprodukten gewährleistet, unabhängig vom Anbieter der Software oder der Computerplattform (AUTODESK INC., 2020). Das in der vorliegenden Arbeit genutzte Programm City Engine kann 3D Modelle in solche FBX-Files exportieren (ESRI INC., 2020).

Die Hauptentwicklungsumgebung Unity dient der Gestaltung der Anwendung. Dabei werden sogenannte C#-Skripte erstellt. Das sind Routinen, die direkt den *Game Objects* zugeordnet werden, wie beispielsweise Bewegungsabläufe der *Game Objects*. Im Rahmen dieser Arbeit öffnet Unity zur Erstellung der Routinen die Entwicklungsumgebung Visual Studio von Microsoft, um die Skripte in dem Editor von Visual Studio zu schreiben. Am Ende des Entwicklungsprozesses wird Visual Studio erneut eingesetzt, um eine lauffähige UWP-App zu erstellen. Dies beinhaltet auch die Funktionalität, die App auf der HoloLens bereit zu stellen. Abbildung 16 zeigt die Oberfläche von Visual Studio. Nach MICROSOFT (2020E) sind von der entsprechenden Hardware folgende Systemanforderungen für Windows Desktop zu erfüllen:

- Windows 7 SP1, Windows 8.1 oder Windows 10 Betriebssystem in Version 1703 oder höher
- 1,8-GHz-Prozessor, empfohlen wird Quad-Core oder besser
- Mindestens 2GB RAM, empfohlen wird 8 GB RAM
- 20-50 GB freier Speicher für normale Installation
- Grafikkarte mit Unterstützung einer Auflösung von mindestens 720p. Empfohlen wird die Auflösung WXGA oder höher.

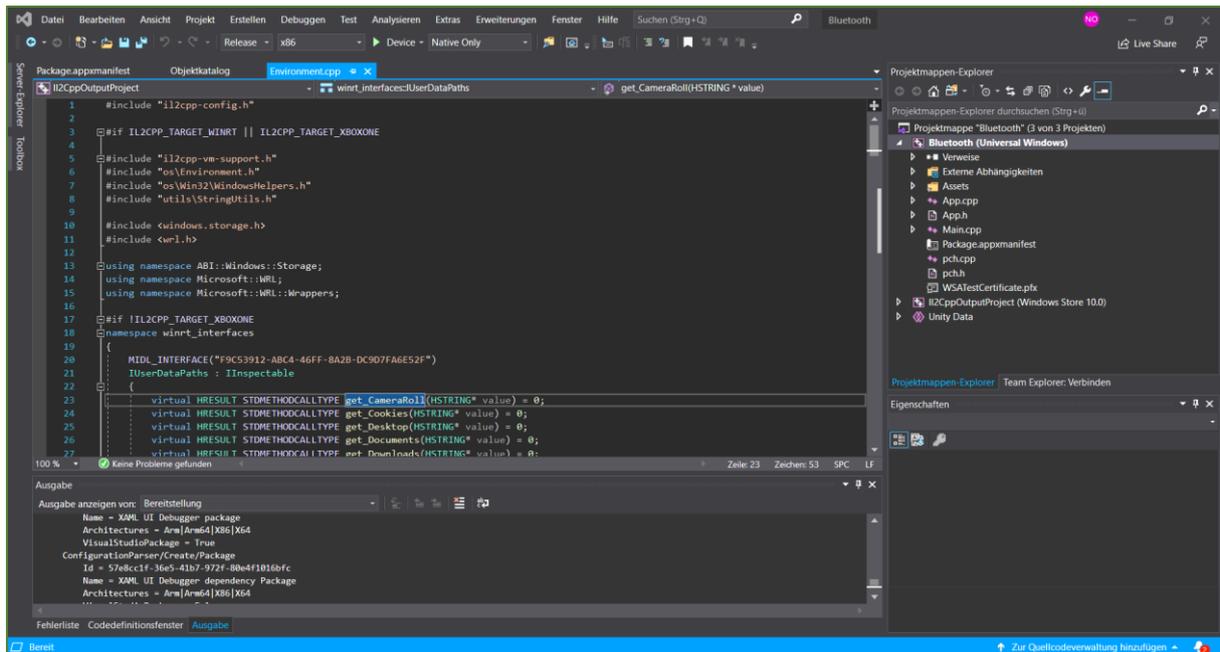


Abbildung 16: Oberfläche von Visual Studio
Quelle: Eigene Darstellung

Aus den Systemvoraussetzungen der Entwicklungsumgebungen wurde als Entwickler-PC ein Dell Precision 5540 zur Verfügung gestellt. Er enthält das Betriebssystem Windows 10 mit neuesten Updates und ein 64-B-System. Zusätzlich ist ein Intel Core i7 Prozessor der neunten Generation (Intel® Core™ i7-9850H CPU) mit 2.6GHz und 16 GB RAM eingebaut. Die im Entwickler-PC enthaltene Grafikkarte ist die NVIDIA Quadro T1000. Der verfügbare Speicher beträgt 250 GB. Der genutzte Entwickler-PC erfüllt somit mit genügend Puffer die optimalen Systemanforderungen beider Entwicklungsumgebungen.

5.4 Pflichtenheft

Im Bereich der Softwareentwicklung gehört die Erstellung eines Pflichtenheftes regelmäßig zu den Bestandteilen der Projektarbeit. Je nach Aufgabenstellung kann ein Pflichtenheft sehr umfangreich sein, indem es das gesamte Arbeitskonzept enthält. Dem gegenüber wird der Begriff Pflichtenheft auch für die einfache Zusammenfassung der technischen Spezifikationen genutzt. Der Einsatzbereich eines Pflichtenheftes kann sich von der Projektidee über Projektvorbereitung, Angebotserstellung bis hin zur Projektdurchführung erstrecken. Eine allgemeinverbindliche Vorlage zur Struktur eines Pflichtenheftes für eine Softwareentwicklung wurde im Rahmen wissenschaftlicher Arbeiten nicht definiert. Es finden sich jedoch je nach Aufgabenstellung diverse Varianten, die sich an dem jeweiligen Projekt orientieren. Für den Rahmen dieser Arbeit dient das Pflichtenheft zur Dokumentation des angestrebten Ergebnisses eines Prototyps. Die Inhalte des nachfolgenden Pflichtenheftes leiten sich aus den Ergebnissen der vorangegangenen Kapitel ab. In den Kapiteln 1 bis 4 sind Inhalte aufbereitet, die teilweise Bestandteil von Pflichtenheften sein können. Zur Vermeidung von Redundanz wurde auf Basis der vorliegenden Aufgabenstellung die folgende Struktur für das Pflichtenheft erstellt:

1. Aufgabenstellung
2. Rahmenbedingungen
3. Technische Spezifikationen
4. Kernfunktionen
5. Qualitätsanforderungen
6. Benutzeroberfläche
7. Testszenarien

Die nachfolgende Tabelle stellt das Pflichtenheft für den zu entwickelten Prototypen dar, gegliedert nach der zuvor festgelegten Struktur. Die Angaben in der Spalte Merkmale enthalten lediglich Stichpunkte und verweisen auf die Ausführungen der vorangegangenen Kapitel.

Tabelle 3: Pflichtenheft

Nr.	Inhalt	Merkmale	Kapitel-Referenz
1.	Aufgabenstellung	- Erstellung des Prototyps einer lauffähigen App zur Visualisierung von Leitungsnetzen vor Ort	Kapitel 1.2
2.	Rahmenbedingungen	- Beschränkung auf die Darstellung eines Wasserrohrstranges - Einsatz von Mixed Reality - Übernahme vorgegebener 2D-Leitungsdaten - Nutzung einer gängigen Softwareentwicklungsumgebung - Anbindung eines definierten, externen GPS-Gerätes - Arbeitsumfang definiert durch die Anforderungen an eine Masterarbeit - Vorgehensweise nach den wissenschaftlichen Prinzipien einer Masterarbeit	- Kapitel 5.2 - Kapitel 4 - Kapitel 5.2 - Kapitel 5.3.3 - Kapitel 5.3.1
3.	Technische Spezifikationen	- Hardware: ○ HoloLens, 1. Generation ○ Entwicklungs-PC Dell Precision 5540	- Kapitel 5.3.1 - Kapitel 5.3.3

		<ul style="list-style-type: none"> ○ GPS-Gerät Blumax GPS-710 - Entwicklungsumgebungen: <ul style="list-style-type: none"> ○ Unity ○ Visual Studio 	<ul style="list-style-type: none"> - Kapitel 5.3.1 - Kapitel 5.3.3 - Kapitel 5.3.3
4.	Kern-funktionen	<ul style="list-style-type: none"> - Darstellung des virtuellen Wasserrohrstranges in der realen Umgebung - Verwendung der Farbe Blau für das darzustellende Objekt - Lagegenaue Darstellung in einem Abstandsbereich vom Betrachter mit den Ausmaßen von 1-5 Meter - Dynamisches Rendering des Wasserrohrstranges in Echtzeit - Anpassung des virtuellen Objektes an die Perspektive des Betrachters - Positionsunabhängigkeit des Anwenders - Blickrichtungsunabhängigkeit des Anwenders - Mobilität des Betrachters unter Verzicht auf den Anschluss an einen PC während der Anwendung vor Ort 	<ul style="list-style-type: none"> - Kapitel 2.2 und 5.2 - Kapitel 5.2 - Kapitel 4
5.	Qualitäts-anforderungen	<ul style="list-style-type: none"> - Optische Erkennbarkeit des Objektes - Fehlerfreie Interpretation des Objektes - Fehlerfreie Datentransformation von 2D in 3D - Verzerrungsfreie, optische Durchsicht mit genügend Lichtdurchlass - Benutzeroberfläche ist intuitiv 	<ul style="list-style-type: none"> - Kapitel 4 - Kapitel 4 - Kapitel 5.3.2 - Kapitel 4 - Kapitel 4
6.	Benutzer-oberfläche	<ul style="list-style-type: none"> - Aktivierung der App über die Benutzeroberfläche der ausgewählten Hardware - Weitgehend ermüdungsfreie Nutzung 	<ul style="list-style-type: none"> - Kapitel 4 - Kapitel 4
7.	Testszenarios	<ul style="list-style-type: none"> - TestszENARIO 1: ohne Anbindung an das GPS-Gerät zur Prüfung der grundsätzlichen Funktionalität. Nutzer nimmt beim Start der Anwendung eine fest definierte Position ein und prüft die Sichtbarkeit und Lagegenauigkeit des Objektes. Anwender kann jede beliebige Blickrichtung nutzen. - TestszENARIO 2: mit Anbindung an das GPS-Gerät zur Prüfung der Anwendbarkeit im praktischen Einsatz. Nutzer kann innerhalb des definierten Bereichs jede beliebige Position einnehmen und jede beliebige Blickrichtung nutzen. 	

Quelle: Eigene Erhebung

5.5 Datenaufbereitung

Wie in Kapitel 5.3.2 beschrieben, arbeitet Unity mit 3D Datenmodellen. Dem gegenüber liegen die Leitungsdaten in zweidimensionaler Form vor und müssen daher transformiert werden. Für die Erstellung des Prototyps ist dieser Transformationsprozess nachfolgend dargelegt.

Im ersten Schritt werden die zweidimensional vorliegenden Linienobjekte in 3D Multipatch Features umgewandelt. Die dazu notwendige Höheninformation wird aus einem digitalen Geländemodell (DGM) entnommen, da die Daten auf die Erdoberfläche des Untersuchungsgebietes projiziert werden müssen. Die Darstellung mit der tatsächlichen Tiefenangabe würde bedeuten, dass die Leitung unterhalb der Erdoberfläche liegt und damit in der HoloLens einen Darstellungsfehler erzeugt. Entweder ist das Objekt gar nicht sichtbar oder die angezeigte Lage entspricht nicht der Realität. Das DGM wurde für das festgelegte Untersuchungsgebiet von der Stadt Reutlingen im DWG-Format mit 2m Auflösung im Gauß-Krüger Koordinatensystem (DHDN GK Zone 3) bereitgestellt. Dieses DGM ist ein 3D-Modell, wodurch die in 3D vorliegenden Daten nicht unmittelbar mit den zweidimensionalen Ausgangsdaten der Leitungen verschnitten werden können. Zusätzlich sind die zugrundeliegenden Koordinatensysteme der zur Verfügung gestellten Daten unterschiedlich. Daher muss eine Datentransformation erfolgen, die sowohl die Umrechnung des 3D-Modells in ein Höhenraster vornimmt als auch die Koordinatentransformation von DHDN GK Zone 3 in das Koordinatensystem ETRS89, UTM Zone 32N durchführt. Die folgenden Schritte sind dazu notwendig und erfolgen mit der Software ArcGIS Pro:

1. Mit dem Werkzeug *CAD to Geodatabase* der Software ArcGIS Pro wird die DWG-Datei in eine Geodatabase umgewandelt. Dabei wird das vorliegende Gauß-Krüger-Koordinatensystem mit dem EPSG-Code 31467 angegeben.
2. Anschließend wird das Werkzeug *Create TIN* der Software ArcGIS Pro angewandt, um ein Höhenmodell basierend auf den CAD-Daten zu erzeugen. Dabei wird die Polygon-Feature-Klasse aus der zuvor erzeugten Geodatabase als Input verwendet, als Koordinatensystem wird ebenfalls der EPSG-Code 31467 angegeben.
3. Durch das Werkzeug *TIN to Raster* der Software ArcGIS Pro wird anschließend das Höhenraster erstellt.

Die nachfolgende Verschneidung der Daten mit dem Höhenmodell und deren Umwandlung in 3D Multipatch Daten wird in mehreren Schritten mit der Software FME durchgeführt. Das Höhenraster wird dabei zunächst mit dem Transformer *Reprojector* in das Koordinatensystem der Leitungsdaten transformiert (ETRS89.UTM-32N). Ein weiterer wichtiger Schritt besteht darin, die Linienfeatures der Leitungsdaten mit dem *Deaggregator* in Singlepart Features umzuwandeln, bevor die Daten verschnitten werden. Multipart Features können andernfalls Fehler im Prozess der Linienbildung erzeugen. Anschließend werden den neuen Linienfeatures mithilfe des *Counters* jeweils neue, eindeutige IDs (LineID) vergeben. Mit dem *Chopper* werden sie danach in ihre Stützpunkte aufgetrennt, um jedem Stützpunkt die entsprechende Höhe zuzuordnen zu können. Diese werden anschließend mit dem *Counter* durchgezählt, damit jeder Stützpunkt eine eindeutige ID erhalten kann. Damit ist die Voraussetzung geschaffen, die Stützpunkte und das Höhenraster mit dem *PointOnRasterValueExtractor* miteinander zu verschneiden. Durch diesen Schritt werden die Informationen des Rasters inklusive der Höhe als Listenattribut an die Punkte angehängt. Um die richtige Reihenfolge der Stützpunkte sicherzustellen, werden sie mit dem *Sorter* nach ihrer zuvor erstellten ID sowie nach der LineID sortiert. Danach werden die Punkte mit dem *3DForcer* in 3D Objekte umgewandelt. Dabei wird als Z-Wert die Höhe aus dem Listenattribut `_band{}.value` herangezogen. Zu

diesem Wert werden 20 cm addiert, um sicherzustellen, dass alle Objekte über der Oberfläche liegen und mit der Erdoberfläche keine Schnittpunkte aufweisen. Abbildung 17 zeigt die am *3DForcer* vorgenommenen Einstellungen zur Umwandlung der 2D-Punkte in 3D-Punkte.

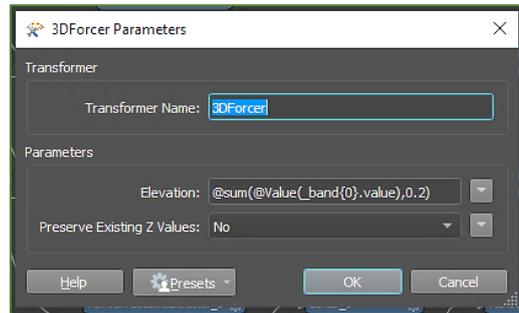


Abbildung 17: Einstellungen des Transformers *3DForcer*
Quelle: Eigene Darstellung

Die zuvor beschriebenen Arbeitsschritte führen zu der Erstellung integrierter dreidimensionaler Stützpunkte des darzustellenden Objektes. Diese Stützpunkte müssen in weiteren Bearbeitungsschritten zu dem entsprechenden Objekt zusammengesetzt werden. Zunächst werden die 3D-Stützpunkte zu einer 3D-Linie verbunden. Danach werden Multipatch Features erstellt, um die Linien in der HoloLens darstellen zu können. Die nachfolgend beschriebenen Transformer dienen dazu, die Multipatches zu erstellen. Dazu werden die Stützpunkte der Linien mit dem *LineBuilder* wieder zu Linien zusammengefügt. Mit dem *Bufferer* werden danach aus den Linien die Multipatch Features erstellt, Abbildung 18 zeigt die dafür vorgenommenen Einstellungen.

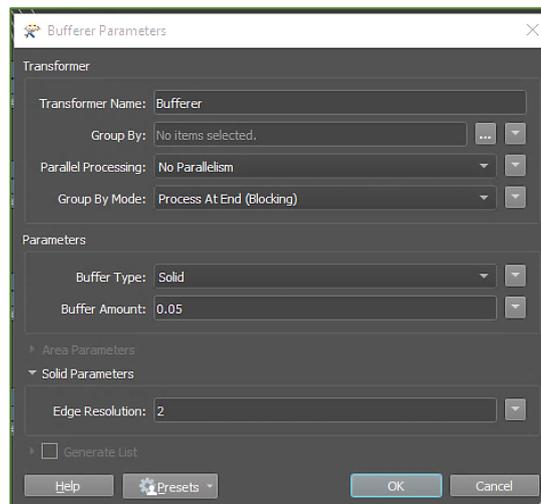


Abbildung 18: Einstellungen des Transformers *Bufferer*
Quelle: Eigene Darstellung

Mit dem *AttributeKeeper* als letzten Transformer wurden alle überflüssigen Attribute entfernt, sodass nur noch die Attribute LBEZ, BYEAR, LTYP, OBJ_ID in die Ziel-Feature-Klassen übernommen wurden. Abbildung 19 zeigt den vollständigen FME Workbench.

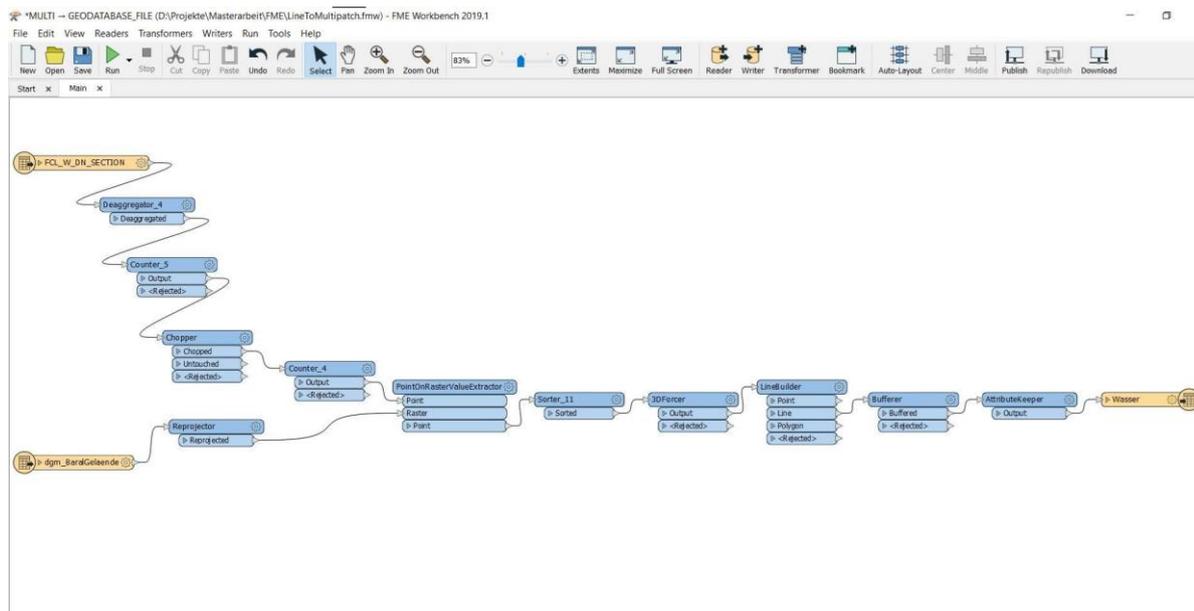


Abbildung 19: Gesamter FME-Workbench

Quelle: Eigene Darstellung

Der finale Schritt der Datenaufbereitung für die Nutzung in der HoloLens besteht aus der Generierung einer FBX-Datei aus den zuvor erzeugten Multipatch-Daten. Bei der FBX-Datei handelt es sich um ein Datenformat, das von der Entwicklungsumgebung Unity bearbeitet werden kann. Die Umwandlung in eine FBX-Datei wird mit der Software CityEngine durchgeführt. Dazu werden das entsprechende Höhenraster aus ArcGIS Pro als TIF exportiert und in CityEngine als Terrain importiert. Zusätzlich werden die erstellten Multipatch-Daten aus der Geodatabase als File GDB importiert.

Die eigentliche Datenumwandlung in die FBX-Datei erfolgt mit dem Exportbefehl. Die Eingabesequenz muss für jedes Feature erneut wiederholt werden und in der Entwicklungsumgebung Unity separat bearbeitet werden. Dieses Verfahren erzeugt einen hohen Arbeitsaufwand und Speicherbedarf. Für den zu erstellenden Prototypen erfolgt daher eine Beschränkung auf ein einzelnes Feature.

Mit den oben beschriebenen Abläufen erfolgt die vollständige Transformation der Daten in ein 3D-Modell. Damit ist die Grundvoraussetzung für die weitere Verarbeitung in der Entwicklungsumgebung Unity geschaffen. Das dritte operative Teilziel dieser Arbeit besteht darin, zu prüfen, inwieweit die bestehende Datengrundlage für die Nutzung in einer Mixed Reality Umgebung ausreichend ist. An dieser Stelle ist festzustellen, dass die Aufbereitung der Ausgangsdaten für die Mixed Reality Anwendung erfolgreich durchgeführt werden konnte und daher der erste Punkt dieses Teilziels positiv bestätigt werden kann. Zusätzlich wird darauf hingewiesen, dass diese Vorgehensweise nur für den Prototypen im Rahmen dieser Arbeit auf Basis der spezifischen Datenausgangslage Gültigkeit besitzt. Abhängig von den Ausgangsquellen, verwendeten Tools und Zielformaten entsteht teilweise erheblicher Anpassungsbedarf und es ist davon auszugehen, dass nicht alle Ausgangsformate in entsprechende Zielformate umgesetzt werden können. Ergänzend ist anzumerken, dass der hohe Anteil an manuellen Arbeitsschritten und Datenberechnungen ein signifikantes Fehlerpotential besitzt. Das Verfahren wird dadurch sehr aufwändig und erfordert für umfangreichere Netze eine entsprechende Automatisierung.

5.6 Entwicklung

Wie in Kapitel 1.2 ausgeführt, ist das Ziel der Entwicklung, einen lauffähigen Prototyp in Form einer App zu erstellen, mit der eine Wasserrohrleitung als virtuelles Hologramm lagerichtig in der realen Umgebung gezeigt werden kann. Für den Prototyp wurde das in Kapitel 5.2 definierte Gebiet ausgewählt. Der Verlauf der ausgewählten Wasserrohrleitung wurde den zur Verfügung gestellten Netzdaten entnommen und ist in Abbildung 20 mit einer hervorgehobenen, türkisenen Linie gekennzeichnet.

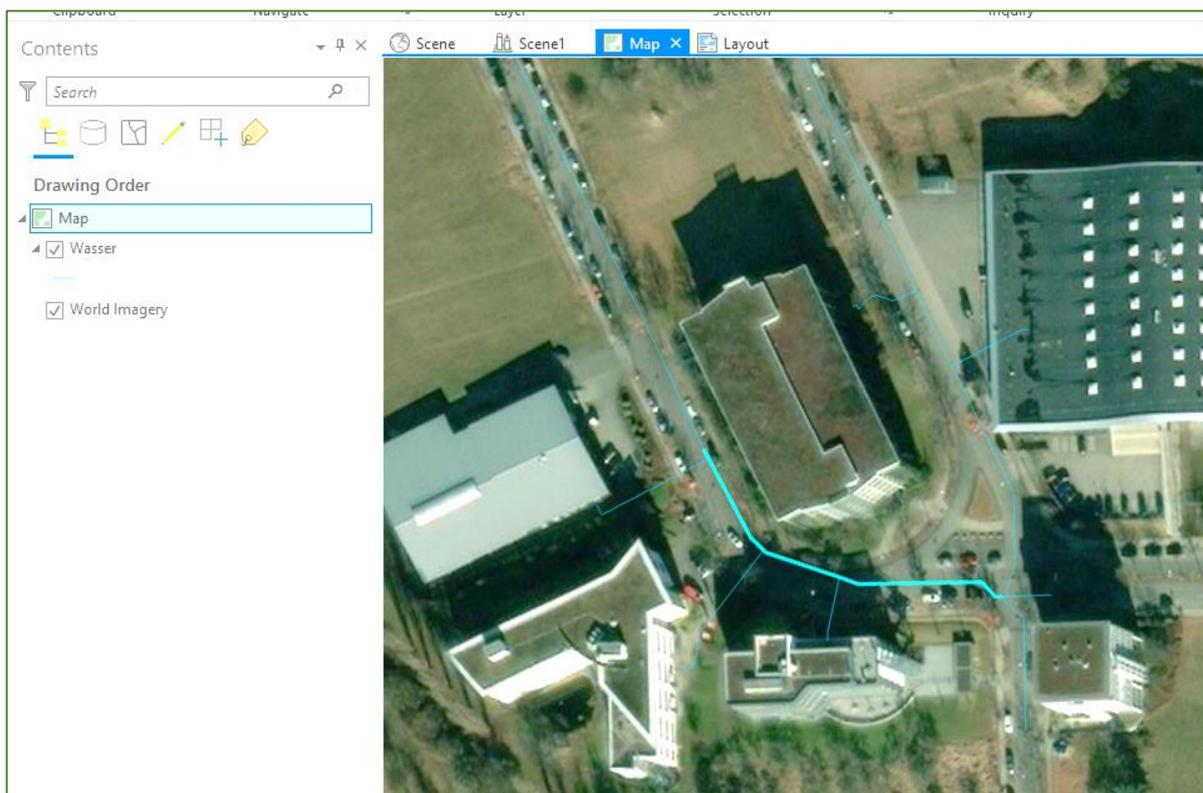


Abbildung 20: Ausgewählter Strang der Sparte Wasser
Quelle: Eigene Darstellung

Die Teilaufgaben der Entwicklung bestehen nach Aufbereitung der Daten darin, das Objekt sichtbar zu machen und lagerichtig einzubetten. Die Sichtbarkeit wird durch die Übernahme der aufbereiteten Daten und Darstellung mithilfe der HoloLens Technologie erzeugt. Die lagerichtige Einbettung des Objektes erfolgt durch die Verknüpfung mit dem in Kapitel 5.3.1 beschriebenen GPS-Empfänger. Dabei ist zu beachten, dass der darzustellende Rohrleitungsstrang virtuell auf die Erdoberfläche projiziert werden muss, da eine unterirdische Darstellung mit der eingenommenen Perspektive zu einer lagefalschen Darstellung führt und teilweise nicht sichtbar ist. Demzufolge wird die Verlegetiefe des Kabelstranges bei der Darstellung im Prototyp nicht berücksichtigt.

Zur Entwicklung des Prototyps sind die nachfolgend beschriebenen Phasen notwendig. In der ersten Phase wird das Objekt in die Entwicklungsumgebung Unity geladen und die für die geplante Visualisierung erforderlichen Einstellungen in Unity werden vorgenommen. Die zweite Phase beinhaltet das Schreiben des Skriptes für die lagerichtige Positionierung des Wasserrohrstranges und

die Zuordnung zu dem Objekt. Während der dritten Phase wird ein Visual Studio Projekt erstellt, das die Grundlage für das nachfolgende Debugging bildet. Während des Debuggings werden in einem iterativen Verfahren Fehler ermittelt und Korrekturen in Phase zwei vorgenommen. In Phase vier wird der Prototyp in Form der finalen App erstellt und in die HoloLens übertragen. Diese Entwicklungsschritte sind in den nachfolgenden Ausführungen detailliert beschrieben und über ein Flussdiagramm in Anhang 13 skizziert.

Die erste Phase beginnt damit, die FBX-Datei in die Hauptentwicklungsumgebung Unity zu laden. Dabei wird die Datei im ersten Schritt im Ordner *Assets* abgelegt. Anschließend wird das Objekt in die *Hierarchy View* per Drag&Drop überführt und entpackt, damit es in der *Scene View* dargestellt werden kann. Unity nutzt kein geografisches, sondern ein kartesisches Koordinatensystem. Aus diesem Grund werden die geografischen Koordinaten beim Einladen in die *Scene View* nicht von Unity übernommen. Das Objekt erhält zunächst die Koordinaten des Ursprungs (0, 0, 0), die richtige Positionierung im Verhältnis zur Kameraposition erfolgt später durch die GPS-Anbindung über ein Skript. Die Farbkennzeichnung des Objektes aus CityEngine hat den Wert grau und enthält noch kein *Material*, das heißt keine Textur. Da es sich um ein Wasserrohr handelt, wird für die Textur die Farbe Blau ausgewählt. Dieses neu erstellte *Material* wird anschließend per Drag&Drop in die *Hierarchy View* auf das bestehende *Material* des Objektes übertragen, wodurch automatisch die blaue Farbe angenommen wird. In der *Inspector View* ist die blaue Farbe sichtbar, bei Selektion des Objektes wechselt die Farbe auf orange, siehe Abbildung 21.

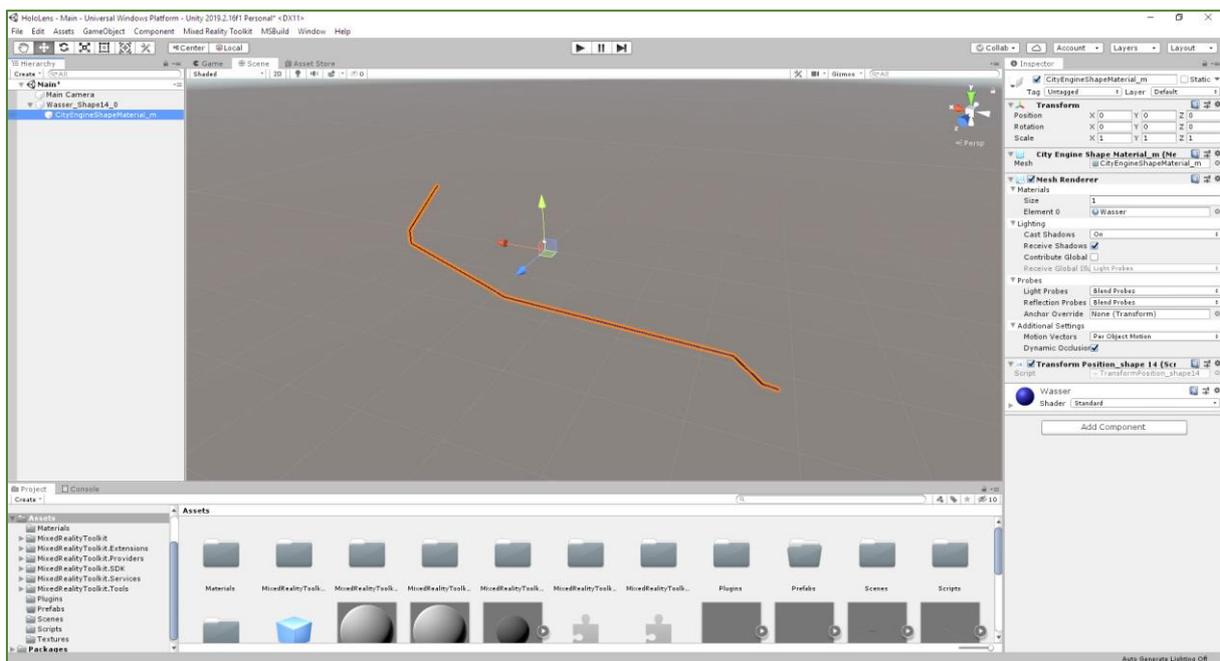


Abbildung 21: In Unity importierter Wasserstrang mit seinen Einstellungen in der Inspector View

Quelle: Eigene Darstellung

Im nächsten Schritt der ersten Phase erfolgen die Einstellungen für die *Main Camera* in der *Inspector View* in Unity (siehe Abbildung 22). Mit dem Objekt *Main Camera* wird in Unity die Perspektive des Betrachters dargestellt. Die Position der *Main Camera* muss angepasst werden, da sie sich am Kopf des Nutzers befindet und somit höher liegt als das virtuelle Objekt. Im vorliegenden Anwendungsfall wurde die Größe des Nutzers mit 1,70 m Augenhöhe angenommen. Gemäß des in Unity verwendeten,

kartesischen Koordinatensystems wird dieser Wert übersetzt mit $X=0$, $Y=1.7$, $Z=0$ und in das Feld *Position* eingetragen. Weitere Anpassungen werden in dem Feld *Camera* vorgenommen. Für *Clear Flags* wird anstatt der standardmäßig ausgewählten *Skybox* die Option *Solid Color* ausgewählt, um den Hintergrund einheitlich und ohne künstlichen Horizont darzustellen. Die *Background Color* wird dabei auf schwarz umgestellt. Diese beiden Einstellungen müssen gemäß Microsoft für Mixed Reality Anwendungen gemacht werden (MICROSOFT, 2020D). Alle übrigen Einstellungen werden bei den Default-Werten belassen. Damit ist die erste Phase der Entwicklung abgeschlossen.

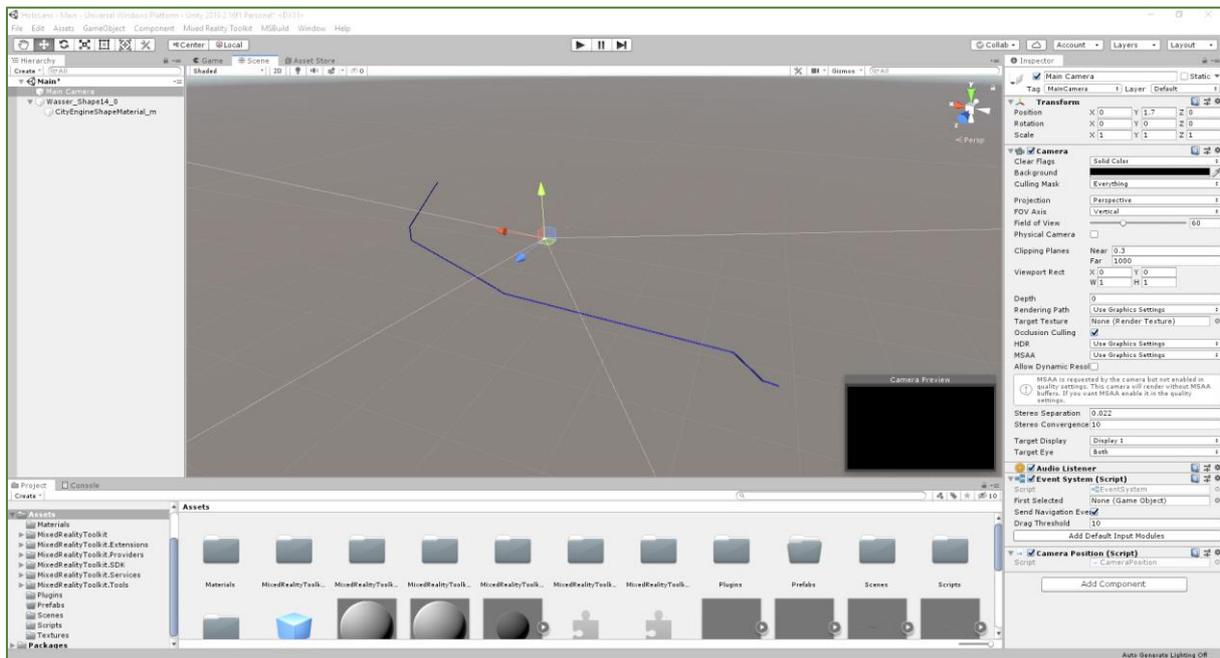


Abbildung 22: Einstellungen der Main Camera
Quelle: Eigene Darstellung

In der zweiten Phase der Entwicklung erfolgt die Erstellung des Skriptes zur lagerichtigen Positionierung des Wasserrohrstranges, der gesamte Skript Code befindet sich in Anhang 14. Dieses Skript wird in der Programmiersprache C# geschrieben und muss im Unity-Editor vollständig lauffähig sein, damit die nachfolgende Kompilierung erfolgen kann. Zusätzlich muss dieses Skript mit Unterstützung der Windows Runtime lauffähig sein, da die UWP-App auf der HoloLens Teile der Windows Runtime benötigt. Im ersten Schritt wird im Ordner *Assets* -> *Scripts* in der *Project View* eine neue Skriptvorlage erstellt, geöffnet und dem Objekt über dessen *Inspector View* zugeordnet. In Schritt zwei erfolgt die Verbindung des GPS-Empfängers mit dem Computer sowie mit der HoloLens und der Empfang der Daten über das Bluetooth-Verbindungsprotokoll. Hierbei ist zu berücksichtigen, dass die HoloLens für die Bluetooth-Verbindung über den Code einige Assemblies der Windows Runtime benötigt, die vom Unity Editor nicht unterstützt werden. Dies führt dazu, dass die Ausführung des Skriptes im Unity-Editor an dieser Stelle abbrechen würde, da diese Assemblies nicht erkannt werden. Um das Skript in beiden Umgebungen lauffähig zu realisieren, muss das Skript mit dieser Situation umgehen können. Daher muss es unterscheiden, in welcher Umgebung das Skript jeweils ausgeführt wird. Für beide Fälle werden Bedingungen geschaffen, die bei Erfüllung zur Ausführung des entsprechenden Code-Blocks führen. Bei Nichterfüllung wird der jeweilige Code-Block übersprungen. Die Umsetzung wird über If-Bedingungen realisiert. Die Bedingung `#if UNITY_EDITOR ... #endif` prüft,

ob der Code im Unity-Editor ausgeführt wird. Die Bedingung `#if ENABLE_WINMD_SUPPORT ... #endif` prüft, ob der Code mit Windows Runtime Unterstützung ausgeführt wird.

Die Umsetzung der Anbindung im Unity Editor erfolgt über einen Serial Port. Über die Bluetooth-Einstellungen des Entwickler-PCs wird das GPS-Gerät bei einem COM-Port registriert, der dann über den Code angesprochen wird. Dafür werden zunächst den für die Bluetooth-Anbindung und den Datenempfang relevanten, globalen Variablen Werte zugeordnet (Abbildung 23, Zeilen 26 bis 33). Dazu gehören `serialPort`, `portName`, `baudRate`, `readTimeOut` und `handshake`, die in der Start-Funktion aufgerufen werden.

```
22 public class TransformPosition : MonoBehaviour
23 {
24
25
26 #if UNITY_EDITOR
27     // Variablen setzen
28     SerialPort serialPort = null;
29     string portName = "COM3";
30     int baudRate = 115200;
31     int readTimeOut = 10000;
32     string handshake = "2";
33 #endif
```

Abbildung 23: Code-Ausschnitt: Definition der Variablen im Unity-Editor
Quelle: Eigene Darstellung

Im Anschluss an die Variablendeklaration folgt der Code zur Bluetooth-Anbindung und dem Empfang der Daten. Zunächst wird der Port, unter dem das GPS-Gerät registriert ist, erstellt und geöffnet, indem zuerst die Variablen definiert und dem richtigen Objekt zugeordnet werden (Abbildung 24, Zeilen 50 bis 54). Über `serialPort.Open()` wird der Port anschließend geöffnet und über eine Logmeldung in der Konsole wird das erfolgreiche Öffnen des Ports mitgeteilt (Abbildung 24, Zeilen 55 und 56). Zum Abfangen möglicher Fehler wird der Block durch eine Try-Catch-Anweisung umrahmt (Abbildung 24, Zeilen 44 bis 63). Kann der Block im Try-Ausdruck nicht ausgeführt werden, läuft der Code in den Catch-Ausdruck, in dem Logmeldungen zur Beschreibung des Fehlers realisiert sind. Im Anschluss werden die Daten über einen Buffer empfangen und gespeichert. Hierfür werden zunächst zwei Variablen für den Buffer und die gelesenen Bytes definiert und danach wird der Datenempfang und das Abspeichern in einer Try-Catch-Anweisung implementiert, damit mögliche Fehler abgefangen werden können (Abbildung 24, Zeilen 68 bis 90). Dabei wird die Variable `bytesRead` neu belegt (Abbildung 24, Zeile 74) und über eine If-Bedingung geprüft, ob die gelesenen Bytes größer Null sind, um zu prüfen ob Daten gelesen werden (Abbildung 24, Zeilen 76 bis 84). Wird dies erfüllt, werden die Daten mithilfe einer For-Schleife gelesen und in einem String gespeichert (Abbildung 24, Zeilen 78 bis 83). Damit sind die Bluetooth-Anbindung und die Datenübertragung des GPS-Empfängers im Unity-Editor abgeschlossen.

```

41 #if UNITY_EDITOR
42 void Start()
43 {
44     try
45     {
46         // Port für die Datenübertragung öffnen.
47         // Nachricht, wenn der Port geöffnet werden konnte.
48         // Wenn nicht -> Catch-Block und Nachricht, dass es nicht funktioniert hat.
49
50         serialPort = new SerialPort();
51         serialPort.PortName = portName;
52         serialPort.BaudRate = baudRate;
53         serialPort.ReadTimeout = readTimeOut;
54         serialPort.Handshake = (Handshake)Enum.Parse(typeof(Handshake), handshake, true);
55         serialPort.Open();
56         Debug.Log("start getting gps information");
57     }
58
59     catch (Exception e)
60     {
61         Debug.Log("start getting gps information failed");
62         Debug.Log(e.Message);
63     }
64
65     // Daten empfangen.
66     // Hierfür erstellen eines Buffers, in den die Daten gelesen werden.
67
68     Byte[] buffer = new Byte[bufferSize];
69     int bytesRead = 0;
70
71     try
72     {
73
74         bytesRead = serialPort.Read(buffer, 0, bufferSize);
75
76         if (bytesRead > 0)
77         {
78             for (int i = 0; i < bufferSize; i++)
79             {
80                 StringBuilder sb = new StringBuilder(serialPort.ReadLine());
81                 string data = sb.ToString();
82             }
83         }
84     }
85
86     catch (Exception e)
87     {
88         UnityEngine.Debug.Log(e.Message);
89     }
90 }
91 #endif

```

Abbildung 24: Code-Ausschnitt: Bluetooth-Anbindung und Datenempfang im Unity-Editor
Quelle: Eigene Darstellung

Die Umsetzung der Anbindung des GPS-Empfängers und der Datenübertragung in der Windows Runtime Umgebung erfolgt in mehreren Stufen. In der ersten Stufe wird der GPS-Empfänger manuell über die Bluetooth-Einstellungen der HoloLens gekoppelt. Über weitere sechs Stufen erfolgt die Erstellung des entsprechenden Code-Blocks über einen StreamSocket. Dafür werden zunächst die entsprechend benötigten Assemblies der Windows Runtime, die vom Unity Editor nicht unterstützt werden, global geladen (siehe Abbildung 25).

```

8  #if ENABLE_WINMD_SUPPORT
9  using Windows.Devices.Enumeration;
10 using Windows.Devices.Bluetooth.Rfcomm;
11 using Windows.Devices.Bluetooth;
12 using Windows.Networking.Sockets;
13 using System.Linq;
14 using Windows.Storage.Streams;
15 using Windows.Devices.SerialCommunication;
16 #endif
17

```

Abbildung 25: Code-Ausschnitt: Assemblies für Windows Runtime
Quelle: Eigene Darstellung

Die nächste Stufe beinhaltet die Definition der Variable für den StreamSocket (Abbildung 26, Zeile 95). In der dritten Stufe werden alle am Gerät registrierten Bluetooth-Geräte über die Funktion `await DeviceInformation.FindAllAsync()` gesucht und in der Variable `deviceInfoCollection` abgespeichert (Abbildung 26, Zeile 97). Im Anschluss wird das benötigte Gerät, in diesem Fall das GPS-Gerät, aus der Liste extrahiert, in der Konsole ausgegeben und in der Variable `service` gespeichert (Abbildung 26, Zeilen 100 bis 103). Danach wird der für den Datenempfang benötigte Socket definiert, mit dem Gerät verbunden und die erfolgreiche Verbindung in der Konsole ausgegeben (Abbildung 26, Zeilen 106 bis 108). In der letzten Stufe werden die Daten empfangen und in einem Array `dataArray` gespeichert (Abbildung 26, Zeilen 111 bis 115). Damit ist die Bluetooth-Anbindung und der Datenempfang über die Windows Runtime abgeschlossen.

```

89 #if ENABLE_WINMD_SUPPORT
90
91
92     async void Start ()
93     {
94
95         StreamSocket _socket;
96         // Zunächst Suchen aller an der HoloLens registrierten Bluetooth Geräte
97         DeviceInformationCollection deviceInfoCollection = await DeviceInformation.FindAllAsync(RfcommDeviceService.GetDeviceSelector(RfcommServiceId.SerialPort));
98
99         // Das gewünschte GPS-Gerät aus der Liste extrahieren
100        Debug.Log("<--> target Bluetooth BR Device initialized:" + deviceInfoCollection[0].Name + ", " + deviceInfoCollection[0].Id);
101        var deviceInfo = deviceInfoCollection[0];
102        var service = await RfcommDeviceService.FromIdAsync(deviceInfo.Id);
103        Debug.Log(service.ConnectionServiceName + ", " + service.ConnectionHostName);
104
105        // Socket für den Empfang der Daten erstellen und zum Bluetooth-Gerät verbinden
106        _socket = new StreamSocket();
107        await _socket.ConnectAsync(service.ConnectionHostName, service.ConnectionServiceName);
108        Debug.Log("Connected");
109
110        // Daten empfangen und speichern
111        var dataReader = new DataReader(_socket.InputStream);
112        Debug.Log("dataReader created");
113        uint s = await dataReader.LoadAsync(256);
114        string dataString = dataReader.ReadString(s);
115        string[] dataArray = dataString.Split(new[] { Environment.NewLine }, StringSplitOptions.None);
116
117    }
118
119 #endif
120

```

Abbildung 26: Code-Ausschnitt: Bluetooth-Anbindung und Datenempfang für Windows Runtime Unterstützung
Quelle: Eigene Darstellung

Im dritten Schritt der zweiten Phase zur Erstellung des Skriptes erfolgt die Filterung und Aufbereitung aller empfangenen Daten in einer globalen Funktion `NmeaToLatLon (string[]dataSplit, string data) {}` für die Positionierung des Objektes. Diese Funktion wird aus der Start-Funktion ausgelagert, da der zu erstellende Code in dieser Funktion nicht unterscheiden muss, in welcher Umgebung das Skript ausgeführt wird. Die Bestimmung der Objektposition erfolgt in Abhängigkeit zur Position des Betrachters. Um diesen Sachverhalt zu verdeutlichen, ist an dieser Stelle darauf hinzuweisen, dass der Betrachter stets die Koordinaten 0,1.7,0 erhält und damit einen fest definierten Standort besitzt. Der Standort des Objektes wird relativ von der Position des Betrachters bestimmt. Diese Vorgehensweise entspricht einer fundamentalen Abweichung der sonst üblichen geografischen Darstellungsweise, bei der die Objekte stets einen fest definierten, absoluten Standort haben. Durch diese Veränderung wird eine bekannte, absolute Standortinformation in eine relative, perspektivenabhängige

Standortinformation umgewandelt. Dies dient dazu, die Position des Objektes immer vom Betrachter aus genau bestimmen zu können und so Abweichungen zu verhindern, die aus der Entfernung des Betrachters zum Startpunkt resultieren. Dazu wird eine Distanzberechnung zwischen der GPS-Position des Betrachters und den Mittelpunktkoordinaten des Objektes jeweils für Latitude und Longitude durchgeführt, damit das Objekt relativ zur Kameraposition, also zur Position des Nutzers, positioniert werden kann. Die benötigte Formel wurde aus KOMPf (2020) entnommen und ist in Abbildung 27 dargestellt.

```
mit distance: Entfernung in km
dx = 71.5 * (lon1 - lon2)
dy = 111.3 * (lat1 - lat2)
lat1, lat2, lon1, lon2: Breite, Länge in Grad
```

Abbildung 27: Formel zur Berechnung der Distanz
Quelle: KOMPf (2020)

Dabei entspricht dx der x-Position und dy der z-Position des Objektes in Unity. Die Höhe des Objektes wird auf null gesetzt, damit es direkt auf den Boden projiziert wird. Lat1 und Lon1 sind die Koordinaten des Leitungsstranges und Lat2 und Lon2 sind die GPS-Koordinaten des Nutzers.

Vor der Implementierung der Distanzbestimmung in die Funktion `NmeaToLatLon (string[] dataSplit, string data) {}` muss der empfangene String zunächst an den Kommas aufgetrennt werden, um die benötigten Koordinaten im richtigen Format aus dem Satz zu extrahieren. Alle durch ein Komma getrennten Elemente werden in eine Liste übertragen und mit Indizes versehen. Anschließend können die benötigten Indizes zwei und vier für Latitude und Longitude extrahiert werden und in das richtige Format Grad, Dezimal umgerechnet werden (Abbildung 28, Zeilen 128 bis 135). Zusätzlich müssen für die Mittelpunktkoordinaten des Leitungsstrangs Variablen definiert werden. Die Koordinaten befinden sich in der Log-Datei des Objektes, die beim Transformieren des Objektes aus CityEngine in das FBX-Format erstellt wurde. Sie befinden sich bereits im richtigen Format: grad, dezimal. Für die vorliegende Arbeit erfolgt dieser Schritt manuell, da er nur einmalig stattfinden muss. Die Koordinaten werden als `double lat = 48.5176` und `double lon = 9.198945` am Anfang des Skriptes definiert (Zeilen 33 und 34 im Code). Die Programmierung des automatisierten Auslesens würde über den Rahmen dieser Arbeit hinaus gehen. Nach Bestimmung der Koordinaten des Betrachters und des Objektes kann die Position des Objektes in Unity relativ zur Kameraposition berechnet werden, um die Position in kartesische Koordinaten umzurechnen. Dabei erfolgt zunächst die Distanzbestimmung auf Basis der in Abbildung 27 dargestellten Formel (Abbildung 28, Zeilen 142 und 143). Dabei sind `lat` und `lon` die Koordinaten des Objektes, während `latCamera` und `lonCamera` die Koordinaten des Betrachters sind. Aus der Distanzbestimmung erfolgt die Positionsbestimmung des Objektes, die Position wird über die `return {}` Funktion in Form eines Vektors ausgegeben (Abbildung 28, Zeilen 148 bis 153).

```

121 public Vector3 NmeaToLatLon(string[] dataSplit, string data)
122
123 {
124     Debug.Log(data);
125
126     // Splittten des Strings und extrahieren der Koordinaten aus dem GPS-Gerät
127
128     string latMin = (dataSplit[2].Replace('.', ',')).Substring(2);
129     string latDegree = (dataSplit[2].Replace('.', ',')).Substring(0, 2);
130
131     string lonMin = (dataSplit[4].Replace('.', ',')).Substring(3);
132     string lonDegree = (dataSplit[4].Replace('.', ',')).Substring(0, 3);
133
134     double latCamera = double.Parse(latMin) / 60 + double.Parse(latDegree);
135     double lonCamera = double.Parse(lonMin) / 60 + double.Parse(lonDegree);
136
137     Debug.Log("data readed Latitude: " + latCamera + " data readed Longitude: " + lonCamera);
138
139     string[] position = { Convert.ToString(latCamera), Convert.ToString(lonCamera) };
140
141     // Berechnen der Distanz zwischen den GPS-Daten und dem Objekt
142     var longitude = 71.5 * (lon - lonCamera) * 1000;
143     var latitude = 111.3 * (lat - latCamera) * 1000;
144
145     Debug.Log(longitude + ", " + latitude);
146
147     // Ausgabe des Vektors der Position des Objektes
148     return new Vector3
149     {
150         x = (float)longitude,
151         y = 0,
152         z = (float)latitude
153     };
154 }

```

Abbildung 28: Code-Ausschnitt: Bestimmung der Position des Objektes

Quelle: Eigene Darstellung

Der bisher erstellte Code in den Zeilen 121 bis 154 dient der Bestimmung der relativen Position des Objektes in Form eines Vektors. Auf diesen Vektor wird beim Ausführen der Start-Funktion zugegriffen, sobald das GPS-Gerät erkannt wurde und Daten empfangen werden können. An dieser Stelle wird erneut auf die entsprechende Umgebung Rücksicht genommen, in der das Skript läuft. Daher wurde der hierfür notwendige Code in die jeweiligen Startfunktionen für den Unity-Editor (Abbildung 29, Zeilen 87-110) und für die Windows Runtime Unterstützung (Abbildung 30, Zeilen 149-172) eingebunden. Welche der beiden Blöcke das Programm durchläuft, hängt von der Erkennung der jeweiligen Umgebung ab, wie im ersten Schritt der Phase zwei beschrieben wurde.

Wie in Kapitel 5.3.1 ausgeführt, wird im Rahmen der vorliegenden Arbeit nur der GGA-Satz benötigt, der an der Zeichenfolge \$GPGGA am Anfang des Satzes erkennbar ist. Daher werden die Daten in beiden Umgebungen zunächst danach gefiltert, ob die Zeile mit \$GPGGA startet. Anschließend wird die Funktion zur Positionsbestimmung ausgeführt.

Im Unity-Editor wird der neue Code-Block innerhalb der For-Schleife zum Datenempfang eingefügt. Nachdem die Daten im String data gespeichert werden, folgt eine If-Bedingung, in der geprüft wird, ob der Satz mit \$GPGGA startet. Wenn ja, werden diese Daten am Komma aufgesplittet und in dem Array dataSplit gespeichert (Abbildung 29, Zeilen 89 bis 90), damit im Anschluss über eine weitere If-Bedingung geprüft werden kann, ob GPS-Daten empfangen wurden und Latitude und Longitude in den Satz geschrieben wurden. Dafür wird geprüft, ob der Index zwei des Arrays, welcher der Latitude entspricht, mehr als zwei Ziffern hat (Abbildung 29, Zeile 95). Ist dies nicht der Fall, konnten keine Koordinaten empfangen werden und es wird der Code innerhalb des Else-Ausdruck ausgeführt, in dem eine entsprechende Logmeldung implementiert ist (Abbildung 29, Zeilen 105 bis 109). Konnten die Daten erfolgreich empfangen werden, wird der Code innerhalb des If-Ausdrucks ausgeführt (Abbildung

29, Zeilen 95 bis 103). Dort wird zunächst die Funktion `NmeaToLatLon` mit den notwendigen, zuvor gespeicherten Daten ausgeführt (Abbildung 29, Zeile 97). Anschließend wird die Position des Objektes über die Funktion `transform.Position = itemPosition` ausgeführt (Abbildung 29, Zeile 100). Damit ist die Positionierung des Objektes im Unity Editor abgeschlossen.

```

74     try
75     {
76
77         bytesRead = serialPort.Read(buffer, 0, bufferSize);
78
79         if (bytesRead > 0)
80         {
81             for (int i = 0; i < bufferSize; i++)
82             {
83                 StringBuilder sb = new StringBuilder(serialPort.ReadLine());
84                 string data = sb.ToString();
85
86                 // Filtern nach dem String, der den benötigten Satz enthält
87                 if (data.StartsWith("$GPGGA"))
88                 {
89                     char[] separator = new char[] { ',' };
90                     string[] dataSplit = data.Split(separator, StringSplitOptions.None);
91
92                     // Prüfen, ob Daten empfangen wurden.
93                     // Wenn ja, Aufruf der Funktion zur Positionierung.
94                     // Wenn nein, Meldung, dass keine Daten empfangen wurden.
95                     if (dataSplit[2].Length > 2)
96                     {
97                         Vector3 itemPosition = NmeaToLatLon(dataSplit, data);
98
99                         // setzen der Position des Objektes
100                        transform.position = itemPosition;
101
102                        Debug.Log("position of water object" + transform.position);
103                    }
104                }
105                else
106                {
107                    Debug.Log(data);
108                    Debug.Log("Es können momentan keine GPS Daten empfangen werden");
109                }
110            }
111        }
112    }
113    catch (Exception e)
114    {
115        UnityEngine.Debug.Log(e.Message);
116    }
117 }
118 }
119 #endif

```

Abbildung 29: Code-Ausschnitt: Positionierung des Objektes für den Unity-Editor

Quelle: Eigene Darstellung

Für die Windows Runtime Unterstützung wird der neue Code-Block zur Positionierung des Objektes unterhalb des Code-Blocks für den Datenempfang eingefügt. Es werden hier zunächst die Variablen definiert, die für die anschließende For-Schleife benötigt werden, das sind die Länge des Arrays `length` als integer und ein Integer `i` (Abbildung 30, Zeilen 149 bis 150). Innerhalb der nachfolgenden For-Schleife (Abbildung 30, Zeilen 152 bis 172) wird derselbe Code-Block eingefügt, der im Unity-Editor in den Zeilen 87 bis 110 auf Abbildung 29 eingebunden wurde. Es wird zunächst nach dem benötigten Satz gefiltert (Abbildung 30, Zeile 154), anschließend erfolgt die Prüfung, ob Daten empfangen wurden (Abbildung 30, Zeilen 156 bis 159). Wenn ja, wird die Funktion `NmeaToLatLon` ausgeführt (Abbildung 30, Zeile 161) und das Objekt wird über die Funktion `transform.Position = itemPosition` neu positioniert (Abbildung 30, Zeile 162). Wenn keine Daten empfangen wurden, wird eine entsprechende Konsolenmeldung ausgegeben (Abbildung 30, Zeilen 166 bis 171). Damit ist die Positionierung des Objektes für die Windows Runtime Unterstützung abgeschlossen.

```

147     string[] dataArray = dataString.Split(new[] { Environment.NewLine }, StringSplitOptions.None);
148
149     int length = dataArray.Length;
150     int i;
151
152     for(i = 0; i < length; i++)
153     {
154         if (dataArray[i].StartsWith("$GPGGA"))
155         {
156             char[] separator = new char[] { ',' };
157             string[] dataSplit = dataArray[i].Split(separator, StringSplitOptions.None);
158
159             if (dataSplit[2].Length > 2)
160             {
161                 Vector3 itemPosition = NmeaToLatLon(dataSplit, dataArray[i]);
162                 transform.position = itemPosition;
163                 Debug.Log("position of water object" + transform.position);
164             }
165         }
166         else
167         {
168             Debug.Log(dataArray[i]);
169             Debug.Log("Es können momentan keine GPS Daten empfangen werden");
170         }
171     }
172 }
173
174 }
175
176 #endif
177

```

Abbildung 30: Positionierung des Objektes für Windows Runtime Unterstützung
Quelle: Eigene Darstellung

Phase 2 der Entwicklung ist abgeschlossen, die lagerichtige Positionierung des Wasserrohrstranges liegt applikationsgerecht vor. Die Funktionsweise des gesamten Skripts ist in einem Flussdiagramm in Anhang 15 schematisch dargestellt.

Die dritte Phase der Entwicklung des Prototyps ist die Kompilierung aus der Hauptentwicklungsumgebung Unity in ein Visual Studio Projekt für UWP in der Programmiersprache C++. Die notwendigen Einstellungen werden über *Files -> Build Settings* vorgenommen, dazu gehören Festlegung von Hardware, Architektur, Kompressionsmethode sowie Auswahl der entsprechenden Release-Stände. Für die Hauptentwicklungsumgebung wurden die vom System vorgegebenen Default-Werte beibehalten, siehe Abbildung 31.

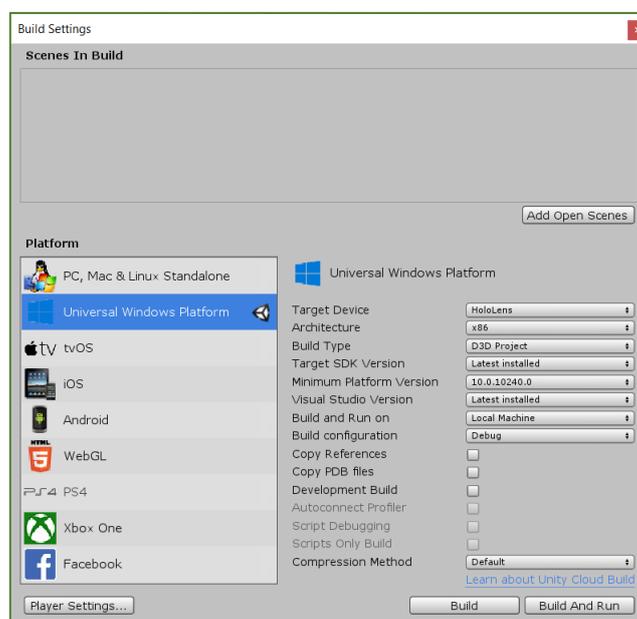


Abbildung 31: Build-Einstellungen im Unity-Editor
Quelle: Eigene Darstellung

Anschließend wird eine Verbindung zu dem GPS-Empfänger hergestellt, falls diese im Rahmen der oben genannten Entwicklungstätigkeiten unterbrochen wurde. In diesem Zusammenhang ist es notwendig, den Zugriff der Anwendung auf den GPS-Empfänger freizuschalten. Auf Basis der für diese Arbeit eingesetzten Komponenten wird hierzu eine zusätzliche Einstellung in der Datei *AppManifest.xml* vorgenommen. Diese Datei wird durch einen Doppelklick auf der linken Seite im Projektmappen-Explorer in Visual Studio geöffnet und bei *Funktionen* wird bei *Bluetooth* das Häkchen gesetzt, wie die nachfolgende Abbildung zeigt.

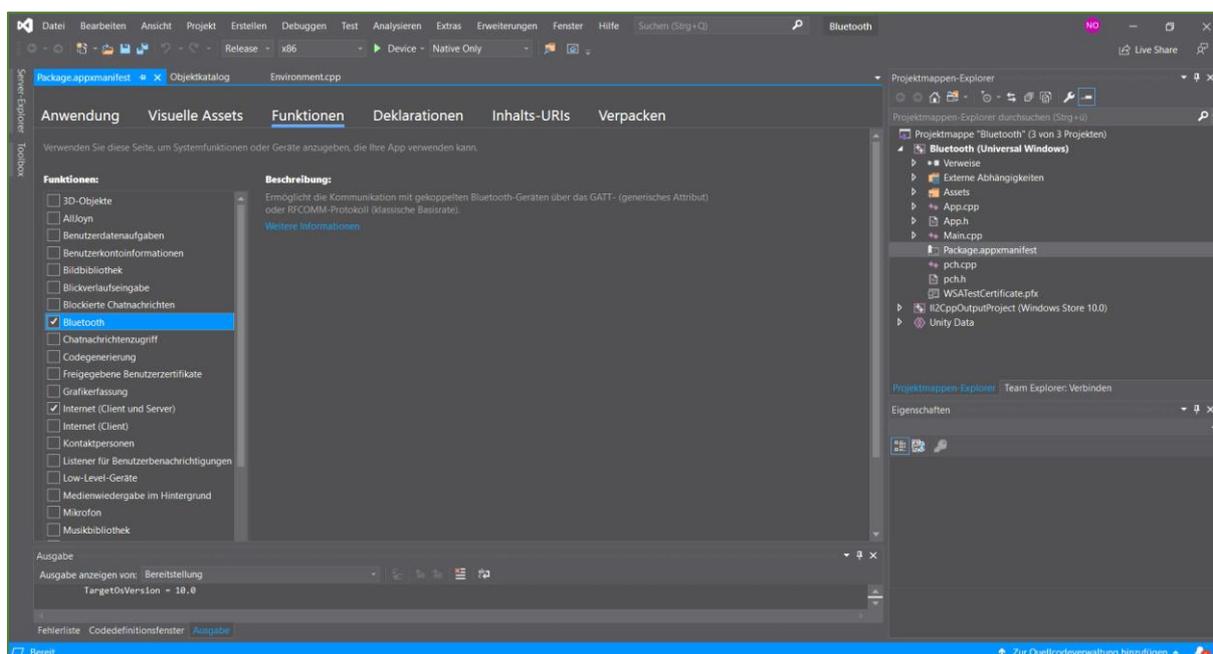


Abbildung 32: Einstellungen in der *AppManifest.xml*

Quelle: Eigene Darstellung

Das durch die bisherigen Entwicklungsschritten entstandene Visual Studio Projekt bildet die Grundlage für die nachfolgende Testphase. Dazu gehören das Debugging, die Bearbeitung von Fehlermeldungen, die Durchführung eines Testszenarios zu Laborbedingungen ohne GPS-Einsatz und die Durchführung eines Test-Szenarios mit GPS-Einsatz. Die Durchführung beider Testszenarios ist ausschlaggebend für das Ergebnis dieser Arbeit, denn das erste Testszenario beantwortet die Frage nach der grundsätzlichen Durchführbarkeit, während das zweite Testszenario die Frage nach der praktischen Einsatzfähigkeit beantworten soll. Vor dem Start des Debuggings wird die HoloLens an den Entwickler-PC angeschlossen und die Anwendung des Prototyps wird in Visual Studio mit den Einstellungen *Debug*, *x86*, *Device* und *Native Only* gestartet, Abbildung 33.

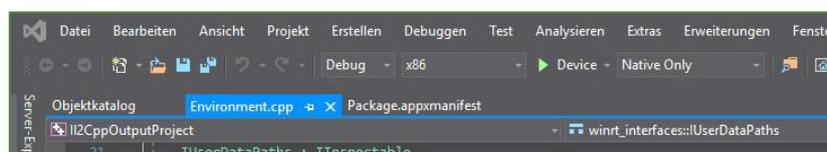


Abbildung 33: Debugging-Einstellungen in Visual Studio

Quelle: Eigene Darstellung

Nach dem Start erscheinen in der Konsole in Visual Studio die entsprechenden Meldungen. Die erste Prüfung besteht darin, Fehlermeldungen zu identifizieren sowie die Ursache zu ermitteln und zu beseitigen. Typische Fehlermeldungen weisen auf fehlende Assemblies, nicht unterstützte Versionen, Zugriffsbeschränkungen, unzureichende Kapazität, fehlende Netzwerkanbindung, etc. hin. Diese Fehlermeldungen sind in der vorliegenden Arbeit aufgrund der Eingrenzung von Umgebung und Objekt nicht aufgetreten. Bei umfangreicheren Anwendungen können diese Fehlerquellen einen erheblichen Korrekturaufwand erzeugen. Dies gilt sowohl für die Entwicklungsschritte als auch für die Pflege und Wartung in einer produktiven Anwendung.

Nach Bearbeitung der systemseitigen Fehlermeldungen erfolgt die Durchführung des ersten Testszenarios. Dieses soll die grundsätzliche Funktionalität der Applikation überprüfen. Hierzu ist es notwendig, statt der durch ein GPS-System erhobenen Positionsdaten fest definierte Standortwerte des Betrachters festzulegen und zu verwenden. Zu diesem Zweck muss der Betrachter die so festgelegte Position physisch einnehmen und die Anwendung bei jedem Testdurchgang von diesem Startpunkt aus starten. Dafür wurde ein markanter Punkt ausgewählt, es handelt sich um die Kreuzung zweier Wege vor dem Parkhaus. Durch die Vermeidung der Verwendung variabler GPS-Daten werden im ersten Testszenario Fehler vermieden, die durch das GPS-System entstehen können. Das lässt bei Abweichungen den Schluss zu, dass diese Abweichungen Gründe außerhalb der GPS-Anbindung haben müssen. Zur Durchführung dieses Testszenarios ist das zuvor erstellte Skript entsprechend anzupassen. Dazu gehören die Eliminierung sowohl der GPS-Anbindung als auch des Transformierens der empfangenen Daten sowie das Hinzufügen von zwei Variablen für Latitude und Longitude des festgelegten Standortes. Abbildung 34 zeigt das für dieses Testszenario erstellte Skript. Analog der Entwicklungsphasen des Prototyps sind für die Testvariante alle Schritte der Phase 3 zu wiederholen.

```
1 using UnityEngine;
2 using System.IO.Ports;
3 using System;
4 using System.Text;
5 using System.Collections.Generic;
6
7 public class TransformPosition : MonoBehaviour
8 {
9
10 // Definition der Mittelpunktskoordinaten des Objekts
11 // und der Koordinaten der Kamera beim Start der App
12 double latObject = 48.516976;
13 double lonObject = 9.199235;
14 double latCamera = 48.516917;
15 double lonCamera = 9.199331;
16
17 void Start ()
18 {
19 // Aufrufen der Funktion zum Berechnen der Position
20 // und Festlegen der entsprechenden Position
21 try
22 {
23 Vector3 itemPosition = NmeaToLatLon();
24 transform.position = itemPosition;
25 Debug.Log("position of water object: " + transform.position);
26 }
27 catch (Exception e)
28 {
29 Debug.Log(e.Message);
30 }
31 }
32
33 // Berechnen der Position des Objektes bzw
34 // der Entfernung des Objektes zur Kameraposition
35 public Vector3 NmeaToLatLon()
36 {
37 Debug.Log("data received Latitude: " + latCamera + " data received Longitude: " + lonCamera);
38 string[] position = { Convert.ToString(latCamera), Convert.ToString(lonCamera) };
39 var longitude = 71.5 * (lonCamera - lonObject) * 1000;
40 var latitude = 111 * (latCamera - latObject) * 1000;
41 Debug.Log(longitude + ", " + latitude);
42 return new Vector3
43 {
44 x = (float)longitude,
45 y = (float)-1.5,
46 z = (float)latitude
47 };
48 }
49 }
```

Abbildung 34: Angepasstes Skript für erstes Testszenario

Quelle: Eigene Darstellung

Das zweite Testszenario erweitert die Erkenntnisse aus dem ersten Testszenario um die Komponente der GPS-Anbindung. Dabei wird geprüft, inwieweit Datentransfer und Datenaufbereitung sach- und lagerichtig erfolgen. Für die Testdurchführung wird der im Rahmen dieser Arbeit entwickelte Prototyp inklusive GPS-Anbindung verwendet, der gesamte Skriptcode ist Anhang 14 zu entnehmen. Die Funktionsweise des Skriptes ist in einem Flussdiagramm in Anhang 15 dargestellt. Der Betrachter nimmt exakt die gleiche Position ein, wie in der vorangegangenen Testreihe ohne GPS-Einsatz. Der wesentliche Unterschied dieser beiden Testverfahren besteht darin, dass bei dem zweiten Test die Positionsdaten des Betrachters von dem GPS-Empfänger zur Verfügung gestellt werden, während bei dem ersten Testszenario diese Positionsdaten manuell eingegeben werden.

Die Entwicklung des Prototyps wird durch die Übertragung der Anwendung auf die HoloLens abgeschlossen, sodass eine autark arbeitende App entsteht. Die Nutzung eines PCs ist in diesem Szenario nicht erforderlich. Hierzu wird aus Visual Studio ein AppPackage zum Querladen der App erstellt und über das Device Portal der HoloLens auf diese übertragen. Durch den letzten Schritt steht die Anwendung in der App-Übersicht der HoloLens zur Verfügung und die Entwicklung der für diese Arbeit verwendeten Applikation ist abgeschlossen.

5.7 Ergebnisse

Im Folgenden werden die ermittelten Ergebnisse beim praktischen Einsatz des Prototyps dokumentiert. Dabei werden zunächst die Ergebnisse aus dem ersten Testszenario mit fest definiertem Standort des Betrachters aufgezeigt. Im Anschluss werden die Ergebnisse des zweiten Testszenarios mit Einbindung des GPS-Gerätes beschrieben. Abschließend werden die Ergebnisse mit dem Pflichtenheft abgeglichen und beschrieben, inwieweit die dort definierten Ziele erreicht werden konnten.

Beide Testszenarien haben gemeinsam, dass die Anwendung des jeweiligen Testszenarios durch Gestik im Startmenü der HoloLens gestartet wird. Zuvor muss sichergestellt sein, dass die Blickrichtung des Anwenders in Richtung Süden zeigt. Der Grund dafür liegt darin, dass die Kamerablickrichtung zur Vereinfachung der Entwicklung in Richtung Süden festgelegt wurde. In beiden Testszenarien nimmt der Betrachter eine zuvor festgelegte Startposition ein und verändert diese Position im weiteren Testverlauf, während die Anwendung läuft.

Nach Beendigung des Startvorgangs wird bereits der Strang des Wasserrohrs angezeigt. Der Betrachtungsbereich ist gemäß den Ausführungen in Kapitel 5.1 auf eine Fläche von 5x5 Meter definiert. In Abbildung 35 ist das Wasserrohrnetz als hellblaue Linie dargestellt, die Startposition des Betrachters ist durch einen roten Punkt gekennzeichnet und der rote Pfeil zeigt die Blickrichtung beim Start der Anwendung.



Abbildung 35: Startposition für das Testen der App
Quelle: Eigene Darstellung, Bild der HoloLens von CLEANPNG.COM (2018)

Die technischen Spezifikationen sind für beide Testszenarien gleich, mit Ausnahme des Einsatzes eines GPS-Gerätes, das lediglich in Testszenario 2 verwendet wird. Bei dem eingesetzten Entwickler-PC handelt es sich um das Dell Notebook Precision 5540. Als GPS-Empfänger wurde das Blumax GPS-710 eingesetzt, die entsprechenden Spezifikationen sind in Kapitel 5.3.1 beschrieben. Zum Einsatz kam die erste Generation der HoloLens mit Windows 10 Betriebssystem. Die eingesetzten Entwicklungsumgebungen waren Unity 2019.4 und Visual Studio 2019. Somit entspricht die verwendete Software der in Kapitel 5.4 spezifizierten Anforderungen.

Das Testscenario 1 beginnt mit dem Start der Anwendung. Der Betrachter hat die zuvor definierte Position eingenommen und die HoloLens aufgesetzt. Der Wasserrohrstrang wird virtuell eingeblendet und der Anwender sieht das virtuelle Objekt und die Umgebung.

Punkt 4 des Pflichtenheftes enthält die Dokumentation der erforderlichen Kernfunktionen. Die nachfolgende Ergebnisdarstellung für Testscenario 1 folgt derselben Struktur. Nach dem Starten der Anwendung ist erkennbar, dass der Strang in das Sichtfeld direkt auf den Boden projiziert wurde. Das virtuelle Objekt erscheint in blauer Farbe, siehe Abbildung 36.

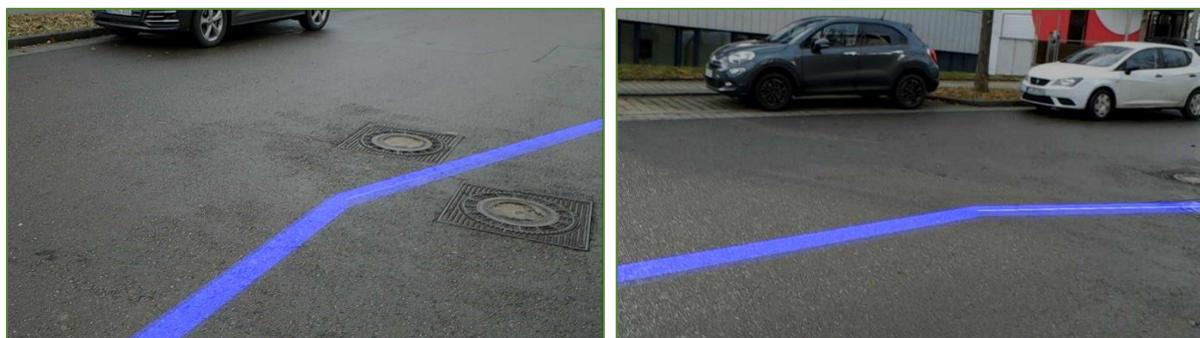


Abbildung 36: Screenshots aus der App: geringe Entfernung zum Betrachter
Quelle: Eigene Darstellung

Die Darstellung erfolgt zunächst lagegenau innerhalb des definierten Bereichs und entspricht damit der Realität. Sobald der Blick des Anwenders diesen Bereich verlässt, nimmt die Lagegenauigkeit proportional mit der Entfernung ab. Abbildung 37 zeigt, wie sich das Rendering des Objektes in größerer Entfernung zur Kameraposition verschlechtert. Dabei zeigt der linke Teil der Abbildung den virtuellen Verlauf des Stranges östlich des Startpunktes. Das tatsächliche Objekt folgt dem Verlauf der abbiegenden Straße, anstatt bereits früher abzuknicken. Dem gegenüber zeigt der rechte Teil der Abbildung den virtuellen Verlauf des Stranges westlich des Startpunktes. Dort befindet sich der Strang auf der Straße und folgt der Kurve, während das virtuelle Objekt mit zunehmender Entfernung zur Kamera auf dem Grünstreifen dargestellt wird. Diese Darstellung entspricht nicht der Realität und dadurch kann diese Information außerhalb des definierten Bereiches nicht genutzt werden.



Abbildung 37: Screenshots aus der App: größere Entfernung zum Betrachter
Quelle: Eigene Darstellung

Verändert der Betrachter seine Position in Blickrichtung und verkürzt dadurch die Entfernung, bewegt sich der Bereich, der zuvor außerhalb des Fünf-Meter-Limits lag, in den definierten Sichtbereich. Abbildung 38 zeigt, dass dadurch die Lagegenauigkeit erhöht wird. Allerdings erreicht die Genauigkeit nicht denselben Grad, der bei der ursprünglichen Betrachtung mit definierter Position vorliegt.

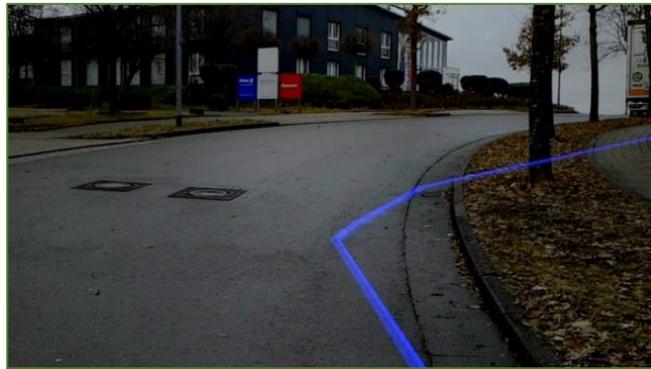


Abbildung 38: Screenshot aus der App: größere Entfernung zur Startposition
Quelle: Eigene Darstellung

Der Grund dieser Darstellungsungenauigkeit liegt darin, dass die Startposition des Anwenders Grundlage für die Positionsbestimmung des virtuellen Objektes ist. Bei Positionsänderung des Anwenders ändert sich die Berechnung nicht, weil die Anwendung keine Daten zur Positionsveränderung des Anwenders erhält. Demzufolge ist festzuhalten, dass die Lagegenauigkeit des virtuellen Objektes in Szenario eins nur dann gegeben ist, solange sich die Position des Betrachters nicht von dem fest definierten Ausgangspunkt entfernt. Verändert der Betrachter seine Blickrichtung, ohne dabei seine vorher definierte Position zu verlassen, wird das virtuelle Objekt weiterhin lagerichtig angezeigt. Somit erfolgt die korrekte Anpassung der Darstellung bei dem entwickelten Prototyp bei Änderung der Blickrichtung. Eine Einschränkung besteht darin, dass bei der Entwicklung des Prototyps festgelegt wurde, die vorgegebene Blickrichtung nach Süden während des Startvorgangs der Anwendung sicherzustellen. Die HoloLens arbeitet mit einem unabhängigen Akku und die Anwendung läuft vollständig auf dem Betriebssystem der HoloLens, sodass eine Anbindung an den PC nicht notwendig ist. Daher ist die Anforderung der Mobilität erfüllt.

Wie im Pflichtenheft in Tabelle 3 dokumentiert, sind neben den Kernfunktionen auch Qualitätsanforderungen zu erfüllen. Die Ergebnisse im Testszenario 1 werden nachfolgend dokumentiert. Wie aus Abbildung 36 ersichtlich, ist das virtuelle Objekt deutlich zu erkennen. Testszenario 1 wurde bei unterschiedlichen Lichtverhältnissen durchgeführt. Dazu gehörten direkte Sonneneinstrahlung bei wolkenlosem Himmel zur Mittagszeit, vollständig bedeckter Himmel am frühen Nachmittag und diffuses Licht morgens. In allen Fällen war das virtuelle Objekt erkennbar, Einschränkungen bei der Erkennbarkeit waren bei sehr hellen Lichtverhältnissen zu verzeichnen. Das Objekt war in diesen Fällen deutlich heller, jedoch noch ausreichend erkennbar. Mit zunehmender Abdunklung der Lichtverhältnisse erhöhte sich die Erkennbarkeit des virtuellen Objektes. Form und Farbe erlaubten jeweils eine fehlerfreie Interpretation. Die Datentransformation von 2D in 3D war erfolgreich. Dies ist an der Durchgängigkeit des virtuellen Objektes zu erkennen. Im Falle einer fehlerhaften Transformation wäre das Objekt nicht durchgängig sichtbar gewesen. Die Durchsicht durch die HoloLens erfolgte verzerrungsfrei. Die bei schlechten Lichtverhältnissen durchgeführten Tests zeigten, dass die optische Dämpfung, die durch die Linsen hervorgerufen wird, vernachlässigt werden konnte. Die Benutzeroberfläche wird als intuitiv bedienbar wahrgenommen, da nur zum Start

zwei Eingabebefehle in Form von zwei Gesten per Handzeichen notwendig sind. Im weiteren Verlauf findet keine menügestützte Eingabe statt. Zum Ausschalten wird ein Druckknopf an der HoloLens betätigt. Demzufolge ist damit ebenfalls die Forderung aus Punkt 6 des Pflichtenheftes erfüllt, die Anwendung direkt über die Benutzeroberfläche der HoloLens zu starten. Durch die vorhandene Befestigungstechnik war die HoloLens angenehm zu tragen und erzeugte während der durchschnittlich einstündigen Tests keinerlei Ermüdung im Bereich der Augen oder des Nackens.

Insgesamt wurden 12 Praxistests durchgeführt, davon drei Praxistests zwischen 8:00 Uhr und 11:00 Uhr vormittags, zwei Tests in den Abendstunden zwischen 17:00 Uhr und Einsetzen der Dämmerung sowie sechs Tests tagsüber zwischen 12:00 Uhr und 14:00 Uhr. Die Testreihen vormittags, mittags und abends erfolgten jeweils bei guten sowie bei eingeschränkten Sichtverhältnissen. Aufgrund der Witterung während der Testphase konnten keine Tests bei Niederschlag, bei Nebel und bei Temperaturen unter 0°C durchgeführt werden.

Zusammenfassend lässt sich festhalten, dass in Testszenario 1 wesentliche Kernfunktionen und wesentliche Qualitätsanforderungen erfüllt werden konnten. Basierend auf einer fest definierten Position und eines fest definierten Blickfeldes von fünf mal fünf Meter konnte eine ausreichende Lagegenauigkeit erzielt werden. Inwieweit die Anforderungen bei Veränderung der Position des Anwenders erfüllt werden, wurde in Testszenario 2 ermittelt.

Tabelle 4: Ergebnisbewertung basierend auf dem Pflichtenheft

Nr.	Inhalt	Details	Testergebnis
1.	Aufgabenstellung	- Erstellung eines Prototyps einer lauffähigen App zur Visualisierung von Leitungsnetzen vor Ort	
2.	Rahmenbedingungen	- Beschränkung auf die Darstellung eines Wasserrohrstranges - Einsatz von Mixed Reality - Übernahme vorgegebener 2D-Leitungsdaten - Nutzung einer gängigen Softwareentwicklungsumgebung - Anbindung eines definierten, externen GPS-Gerätes - Arbeitsumfang definiert durch die Anforderungen an eine Masterarbeit - Vorgehensweise nach den wissenschaftlichen Prinzipien einer Masterarbeit	Rahmenbedingungen vorhanden
3.	Technische Spezifikationen	- Hardware: <ul style="list-style-type: none"> ○ HoloLens, 1. Generation ○ Entwicklungs-PC Dell Precision 5540 ○ GPS-Gerät Blumax GPS-710 - Entwicklungsumgebungen: <ul style="list-style-type: none"> ○ Unity ○ Visual Studio 	- erfüllt - erfüllt
4.	Kernfunktionen	- Darstellung des virtuellen Wasserrohrstranges in der realen Umgebung - Verwendung der Farbe Blau für das darzustellende Objekt - Darstellung lagegenau in Länge und Breite in einem Bereich von 1m bis 5m	- erfüllt - erfüllt - erfüllt

		<ul style="list-style-type: none"> - Dynamisches Rendering des Wasserrohrstranges in Echtzeit - Anpassung des virtuellen Objektes an die Perspektive des Betrachters - Positionsunabhängigkeit des Betrachters - Blickrichtungsunabhängigkeit des Betrachters - Mobilität des Betrachters unter Verzicht auf den Anschluss an einen PC während der Anwendung vor Ort 	<ul style="list-style-type: none"> - erfüllt - erfüllt - eingeschränkt - eingeschränkt - erfüllt - erfüllt
5.	Qualitätsanforderungen	<ul style="list-style-type: none"> - Optische Erkennbarkeit des Objektes - Fehlerfreie Interpretation des Objektes - Fehlerfreie Datentransformation von 2D in 3D - Verzerrungsfreie, optische Durchsicht mit genügend Lichtdurchlass - Benutzeroberfläche ist intuitiv 	<ul style="list-style-type: none"> - erfüllt - erfüllt - erfüllt - erfüllt - erfüllt
6.	Benutzeroberfläche	<ul style="list-style-type: none"> - Aktivierung der App über die Benutzeroberfläche der ausgewählten Hardware - Weitgehend ermüdungsfreie Nutzung 	<ul style="list-style-type: none"> - erfüllt - erfüllt
7.	Testszenerien	<ul style="list-style-type: none"> - Testszenerio 1: ohne Anbindung an das GPS-Gerät zur Prüfung der grundsätzlichen Funktionalität. Nutzer nimmt beim Start der Anwendung eine fest definierte Position ein und prüft die Sichtbarkeit sowie Lagegenauigkeit des Objektes. Anwender kann jede beliebige Blickrichtung nutzen. - Testszenerio 2: mit Anbindung an das GPS-Gerät zur Prüfung der Anwendbarkeit im praktischen Einsatz. Nutzer kann innerhalb des definierten Bereichs jede beliebige Position einnehmen und jede beliebige Blickrichtung nutzen. 	<p>Zwölf Testdurchgänge an unterschiedlichen Witterungsbedingungen und Tageszeiten</p> <p>Separate Testreihe, siehe Ausführungen Testszenerio 2</p>

Quelle: Eigene Erhebung

Das zweite Testszenerio beginnt gleichermaßen mit dem Start der Anwendung. Der Betrachter hat die zuvor definierte Position eingenommen und die HoloLens aufgesetzt. Der Wasserrohrstrang wird virtuell eingeblendet und der Anwender sieht das virtuelle Objekt und die Umgebung. Zusätzlich wurde das GPS-Gerät gestartet und die Positionsdaten des Betrachters übertragen. Die Übertragung der Daten war zunächst erfolgreich, während im weiteren Testverlauf die Übertragung vielfach unterbrochen wurde. Der Grund für diese Unterbrechung war nicht zu ermitteln, es kann sich um eine Fehlfunktion des im Test eingesetzten Gerätes handeln oder die Qualität des GPS-Signals war stellenweise nicht ausreichend zur Positionsberechnung. Die Unterbrechungen führten dazu, dass einige Testdurchläufe mehrfach wiederholt werden mussten, um die Kernfunktionen und Qualitätsanforderungen zu prüfen. Der gesamte Ablauf des Testszenerios 2 entspricht, abgesehen von den Abbrüchen, dem gleichen Vorgehen wie bei Testszenerio 1.

Die Projektion entspricht dem gleichen Ergebnis wie in Testszenerio 1. Bezüglich der Lagegenauigkeit sind erhebliche Abweichungen festzustellen. Die Positionierung des virtuellen Objektes wich im Vergleich zu den Ergebnissen aus Testszenerio 1 sowie zwischen den einzelnen Testdurchgängen in Testszenerio 2 erheblich voneinander ab. Dieser Umstand ist darauf zurückzuführen, dass die vom GPS-

Gerät übertragenen Positionsdaten beim Start der Anwendung variierten, obwohl die zuvor definierte Position des Betrachters beim Start nicht verändert wurde. Insgesamt wurden neun Testdurchgänge durchgeführt, bei lediglich einem Testdurchgang war die berechnete Position identisch mit der tatsächlichen Position. In diesem Fall ist das Ergebnis analog der Ergebnisse in Testszenario 1. Die Lagegenauigkeit außerhalb des definierten Betrachtungsfeldes nimmt mit der Entfernung zur ursprünglich definierten Position ab. Ergänzend ist darauf hinzuweisen, dass die Positionsberechnung per GPS in der Anwendung nur während der Startsequenz stattfindet. Die laufende Berechnung einer sich verändernden Position des Betrachters ist nicht Gegenstand dieser Arbeit und findet daher keinen Niederschlag in der Anwendung. In Testszenario 2 zeigten alle Funktionen, die nicht von dem Einsatz des GPS-Gerätes abhängen, die gleichen Ergebnisse wie in Testszenario 1. Dies sind die Kernfunktionen Farbkennzeichnung, Rendering in Echtzeit, perspektivische Anpassung, Mobilität sowie alle Qualitätsanforderungen und die Anforderungen an die Benutzeroberfläche.

Insgesamt wurden neun Praxistests durchgeführt, in einem Fall erfolgte eine korrekte Positionsbestimmung durch das GPS-Gerät. Dieser Testdurchgang fand bei leichter Bewölkung zwischen 12:00 Uhr und 14:00 Uhr statt. In drei Fällen wich die berechnete Position um mehr als 10 Meter von der tatsächlichen Position ab, diese Tests wurden abgebrochen. In fünf Fällen konnte das GPS-Gerät keine Position berechnen, diese Tests wurden ebenfalls abgebrochen.

Zusammenfassend lässt sich feststellen, dass die Übertragung der Daten des GPS-Gerätes korrekt erfolgte. Die Position des Anwenders wurde von der Anwendung auf Basis der übertragenen GPS-Daten korrekt wiedergegeben. Somit ist die Nutzung von GPS-Daten bei dem Einsatz von Mixed Reality erfolgreich nachgewiesen. Mit Bezug auf die Lagegenauigkeit konnte das Testszenario 2 aufgrund der Testabbrüche keine repräsentativen Ergebnisse liefern. Daraus ergibt sich die Erkenntnis, dass die Nutzbarkeit von GPS-Positionsdaten in Zusammenhang mit Mixed Reality eine höhere Präzision und Zuverlässigkeit der übertragenen Positionsdaten voraussetzt. Die nachfolgende Tabelle fasst die Ergebnisse des erfolgreich durchgeführten Testdurchganges zusammen.

Tabelle 5: Bewertung der Ergebnisse von Testszenario 2

Nr.	Inhalt	Details	Testergebnis
1.	Aufgabenstellung	- Erstellung eines Prototyps einer lauffähigen App zur Visualisierung von Leitungsnetzen vor Ort	
2.	Rahmenbedingungen	- Beschränkung auf die Darstellung eines Wasserrohrstranges - Einsatz von Mixed Reality - Übernahme vorgegebener 2D-Leitungsdaten - Nutzung einer gängigen Softwareentwicklungsumgebung - Anbindung eines definierten, externen GPS-Gerätes - Arbeitsumfang definiert durch die Anforderungen an eine Masterarbeit - Vorgehensweise nach den wissenschaftlichen Prinzipien einer Masterarbeit	Rahmenbedingungen vorhanden
3.	Technische Spezifikationen	- Hardware: ○ HoloLens, 1. Generation ○ Entwicklungs-PC Dell Precision 5540 ○ GPS-Gerät Blumax GPS-710 - Entwicklungsumgebungen:	- erfüllt - erfüllt

		<ul style="list-style-type: none"> ○ Unity ○ Visual Studio 	
4.	Kern-funktionen	<ul style="list-style-type: none"> - Darstellung des virtuellen Wasserrohrstranges in der realen Umgebung - Verwendung der Farbe Blau für das darzustellende Objekt - Darstellung lagegenau in Länge und Breite in einem Bereich von 1m bis 5m - Dynamisches Rendering des Wasserrohrstranges in Echtzeit - Anpassung des virtuellen Objektes an die Perspektive des Betrachters - Positionsunabhängigkeit des Betrachters - Blickrichtungsunabhängigkeit des Betrachters - Mobilität des Betrachters unter Verzicht auf den Anschluss an einen PC während der Anwendung vor Ort 	<ul style="list-style-type: none"> - erfüllt - erfüllt - eingeschränkt - erfüllt - erfüllt - eingeschränkt - eingeschränkt - erfüllt - erfüllt
5.	Qualitätsanforderungen	<ul style="list-style-type: none"> - Optische Erkennbarkeit des Objektes - Fehlerfreie Interpretation des Objektes - Fehlerfreie Datentransformation von 2D in 3D - Verzerrungsfreie, optische Durchsicht mit genügend Lichtdurchlass - Benutzeroberfläche ist intuitiv 	<ul style="list-style-type: none"> - erfüllt - erfüllt - erfüllt - erfüllt - erfüllt
6.	Benutzeroberfläche	<ul style="list-style-type: none"> - Aktivierung der App über die Benutzeroberfläche der ausgewählten Hardware - Weitgehend ermüdungsfreie Nutzung 	<ul style="list-style-type: none"> - erfüllt - erfüllt
7.	Testszenarien	<ul style="list-style-type: none"> - Testszenario 1: ohne Anbindung an das GPS-Gerät zur Prüfung der grundsätzlichen Funktionalität. Nutzer nimmt beim Start der Anwendung eine fest definierte Position ein und prüft die Sichtbarkeit und Lagegenauigkeit des Objektes. Anwender kann jede beliebige Blickrichtung nutzen. - Testszenario 2: mit Anbindung an das GPS-Gerät zur Prüfung der Anwendbarkeit im praktischen Einsatz. Nutzer kann innerhalb des definierten Bereichs jede beliebige Position einnehmen und jede beliebige Blickrichtung nutzen. 	<p>Separate Testreihe, siehe Ausführungen Testszenario 1</p> <p>Acht Testdurchgänge abgebrochen, ein Testdurchgang erfolgreich</p>

Quelle: Eigene Erhebung

6 Diskussion

Zunächst erfolgt die Bewertung der Vorgehensweise. Die zur Überprüfung der These ausgewählte Vorgehensweise bestand aus den folgenden Schritten. Als erstes wurden die entsprechenden Grundlagen zu den rechtlichen Rahmenbedingungen, den Visualisierungstechnologien und der satellitengestützten Ortung ermittelt. Im Anschluss erfolgte die Literaturrecherche zu dem Stand der Forschung bezüglich der Nutzbarkeit von Mixed Reality zur Visualisierung unterirdischer Leitungsnetze. Danach wurden die wesentlichen Mixed Reality Technologien miteinander verglichen. Das daraus entstehende Anforderungsprofil bildete die Basis zur Auswahl der Technologie zur Entwicklung des Prototyps. Der nächste Schritt bestand darin, das Untersuchungsgebiet klar zu definieren und die für die Entwicklung und Anwendung benötigten Hardware- und Softwarekomponenten zu definieren. Im Anschluss erfolgte die Erstellung des Pflichtenheftes. Auf dieser Grundlage wurden die Entwicklungsschritte bestehend aus der Datenaufbereitung und -übertragung, der Entwicklung der Anwendung sowie der Installation des Prototyps auf dem Endgerät durchgeführt. Der Prototyp wurde in zwei Testszenarien geprüft und die Ergebnisse wurden anhand der Vorgaben des Pflichtenheftes dokumentiert.

Die Vorgehensweise dieser Arbeit erwies sich als zielführend, die operativen Teilfragen wurden beantwortet. Die Voranstellung der Erarbeitung der rechtlichen Grundlagen war wichtig, um diese Erkenntnisse in den Anforderungskatalog einfließen zu lassen. Dieser Schritt führte zu dem Ergebnis, dass als Basis für die Definition von Anforderungen an die Leitungsdokumentation hauptsächlich die technischen Regeln zu berücksichtigen sind. Weiterhin wurden die Grundlagen zu den Visualisierungstechnologien und der satellitengestützten Ortung ermittelt. Die Ergebnisse aus Kapitel 2 beantworten die erste operative Teilfrage dieser Arbeit.

Die nachfolgende Literaturrecherche zum Einsatz von Mixed Reality in der aktuellen Forschung hat ergeben, dass ein deutlich erkennbarer Nutzen durch den Einsatz von Mixed Reality vorhanden ist. Die Ausführungen in Kapitel 3 weisen bereits darauf hin, dass ein Teil dieses Nutzens auch bei der Visualisierung unterirdischer Leitungsnetze gegeben ist. Die Ergebnisse aus Kapitel 3 bestätigen die zweite Forschungsfrage und legen damit die Voraussetzung für die weitere Bearbeitung der Leitfrage.

Der anschließende Vergleich der Technologien hatte zum Ziel, die für die Erstellung des Prototyps geeignete Technologie auszuwählen. Auf Basis der wesentlichen Parameter und mithilfe des in Kapitel 4.1 dokumentierten Anforderungskataloges wurden vier Technologien verglichen und die HoloLens ausgewählt. Die Vorgehensweise in Kapitel 4 beantwortet damit die dritte operative Teilfrage.

Mit der präzisen Definition des Untersuchungsgebietes wurde die Voraussetzung geschaffen, einen Prototyp innerhalb des vorgegebenen Rahmens dieser Arbeit zu entwickeln. Zunächst wurde mit der Definition der Rahmenbedingungen ein klar abgegrenztes Untersuchungsgebiet erzeugt. Dazu gehörte auch der Verzicht auf die Einbindung ergänzender Attribute, die Limitierung auf einen einzelnen Strang der Sparte Wasser und die Begrenzung der Nutzung von GPS-Daten. Zur Vorbereitung der Entwicklung wurde die Aufbereitung der zur Verfügung gestellten Daten erfolgreich durchgeführt. Diese Aufbereitung erfolgte durch manuelle Schritte und muss bei zunehmender Komplexität automatisiert werden. Der Einsatz der HoloLens erfolgte aus technischer Sicht ebenfalls problemlos. Auf dieser Basis fand die Entwicklung des Prototyps mithilfe der Entwicklungsumgebungen Unity und Visual Studio statt. Die dabei eingesetzte Hardware und Software erzeugte im Rahmen dieser Arbeit keine Limitierungen oder unerwartete Verhaltensweisen, sodass die Datenaufbereitung und die Applikationsentwicklung inklusive der Einbindung eines GPS-Empfängers erfolgreich abgeschlossen

wurden. Das geplante virtuelle Objekt war innerhalb der Projektionsebene sichtbar. Mit diesem Ergebnis ist die vierte operative Teilfrage bestätigt.

Abschließende Tests und der Vergleich der Ergebnisse mit den Anforderungen des Pflichtenheftes waren notwendig, um aus den gewonnenen Erkenntnissen Rückschlüsse auf die Validität der These zu ziehen. Zu diesem Zweck wurden zwei Testszenarien definiert, das erste ohne GPS-Einsatz und das zweite mit GPS-Einsatz. Wie in Kapitel 5.7 dargestellt, haben die Ergebnisse der Testdurchläufe im ersten Testszenario bestätigt, dass die meisten der im Pflichtenheft definierten Kernfunktionen erfüllt wurden. Einschränkungen ergaben sich bei der Forderung zur Blickrichtungs- und Positionsunabhängigkeit beim Starten der Anwendung. Diese Einschränkung liegt in der zuvor festgelegten Position. Dieser Schritt war notwendig, um die Entwicklungsarbeit im begrenzten Rahmen der vorliegenden Arbeit zu halten. Im praktischen Einsatz wäre die Erfüllung dieser Funktion eine zwingende Voraussetzung, aber für die Überprüfung der These hat diese Einschränkung keine Auswirkung. Die Testergebnisse zeigten weiterhin, dass im Rahmen des Untersuchungsgebietes sämtliche Qualitätsanforderungen erfüllt wurden. Gleiches gilt für die Benutzeroberfläche. Das zweite Testszenario beinhaltete die Anbindung des GPS-Empfängers. Während dieser Testdurchläufe traten erhebliche Ausfälle auf, da die übertragenen GPS-Informationen teilweise unvollständig oder massiv fehlerhaft waren. Es war in diesem Zusammenhang nicht feststellbar, inwieweit die Ausfälle durch das Gerät, die Umgebung oder die Qualität des GPS-Signals verursacht wurden. Die Ergebnisse mit Bezug auf die GPS-Anbindung sind daher nicht als repräsentativ einzustufen. Vor diesem Hintergrund ist die fünfte operative Teilfrage für das erste Testszenario validiert, für das zweite Testszenario kann sie nicht validiert werden.

Zusammenfassend ist festzustellen, dass die Vorgehensweise und die erzeugten Ergebnisse dieser Arbeit die These bestätigen, dass die bildgebende Technologie der Mixed Reality für die Visualisierung von Leitungsnetzen angewendet werden kann und der Nachweis in Form eines Prototyps möglich ist. Dieser Prototyp erfüllt die geforderten Funktionen, lediglich bei der GPS-Anbindung ist keine finale Aussage möglich.

7 Ausblick

Wie in den vorangegangenen Kapiteln dargelegt, wurde die These bestätigt, dass die bildgebende Technologie der Mixed Reality für die Visualisierung von Leitungsnetzen angewendet werden kann und der Nachweis in Form eines Prototyps möglich ist. Diese Ergebnisse konnten im Rahmen dieser Arbeit nur dadurch erzielt werden, dass wesentliche Einschränkungen in Bezug auf das Untersuchungsgebiet, die Funktionalität, die Komplexität und die GPS-Anbindung festgelegt wurden. Insofern ist die Bestätigung das Ergebnis einer wissenschaftlichen Arbeit unter Laborbedingungen. Zusätzlich sind die Ergebnisse direkt abhängig von der Qualität der zur Verfügung gestellten Datenbasis. Das Ergebnis zeigt, dass eine Applikation definitiv entwickelbar ist, das vorliegende Ergebnis ist jedoch noch sehr weit von einer praktikablen Anwendung entfernt. Ausgehend von dem Zweck einer solchen Anwendung, Kosten zu senken und Sicherheit zu erhöhen, sind weitere Forschungsarbeiten sinnvoll.

Für weitere Entwicklungen in diesem Forschungsgebiet sind zunächst grundsätzliche Anforderungen zu erfüllen. Dazu gehört die Ausweitung des Untersuchungsgebietes in der Fläche, in der Anzahl der Objekte sowie in der Vielfältigkeit der Gewerke. Eine zeitgemäße Applikation sollte außerdem dazu in der Lage sein, die entsprechenden Ausgangsdaten aus diversen, unterschiedlichen Formaten und Quellen vollkommen automatisiert aufzubereiten und in die Applikation zu übertragen. Zur Gewährleistung der Mobilität könnte sich daraus die Anforderung an den Einsatz einer drahtlosen Datenübertragungstechnologie ergeben. Weiterhin sind im nächsten Entwicklungsschritt Attribute hinzuzufügen, die Auskünfte analog der bisher gängigen ausgedruckten und zweidimensionalen Pläne geben. Dazu gehören Angaben zur Verlegetiefe, zu Dimension und Material, und weitere. Ein zusätzliches wichtiges Entwicklungsgebiet ist die vollständige Einbettung einer hochgenauen Positionierungstechnologie, beispielsweise die Ergänzung eines gängigen GPS-Empfängers mit SAPOS. Angelehnt an die Anwendung des klassischen papierbasierten Planes darf das GPS-gestützte Visualisierungssystem keinerlei Einschränkungen bei der Position des Betrachters sowie seiner Blickrichtung haben. Diese Funktionalität kann erreicht werden, indem die Anwendung im Sekundentakt die Position und Blickrichtung neu berechnet. Die dafür gängigen Technologien finden sich im Bereich der hochgenauen Ortung und entsprechender Sensorik. Sobald eine in Teilen verfügbare Technologie eingesetzt werden kann, besteht eine weitere Entwicklungsaufgabe darin, die Schnittstelle Mensch-Maschine zu optimieren. Hierbei können Technologien eingesetzt werden, die auf Basis einer intuitiven Benutzeroberfläche, Gestik-, Blick- und Sprachsteuerung arbeiten.

Die Forschungsergebnisse zum Einsatz virtueller Techniken in anderen Branchen sind sehr vielversprechend. Die gegenwärtigen Probleme und Risiken, die sich beim Einsatz von 2D-Plänen ergeben, können mit der Mixed-Reality-basierenden Visualisierung vielfach vermieden oder zumindest reduziert werden. Aufgrund der mittlerweile weit verbreiteten Trennung zwischen digitaler Datenspeicherung und -bearbeitung auf der einen Seite sowie der Visualisierung auf der anderen Seite, sind die Grundlagen für Mixed Reality bereits vorhanden. Dies spart erhebliche Kosten, trägt zur Vermeidung von Unfällen auf der Baustelle bei und leistet einen hohen Beitrag zur Gewährleistung der Versorgungssicherheit.

Literatur

- ALEXANDER, S. A., ROZO, J. S., DONADIO, B. T., TENHUNDFELD, N. L., VISSER, E. J. D. & TOSSELL, C. C. Transforming the Air Force Mission Planning Process with Virtual and Augmented Reality. 2019 Systems and Information Engineering Design Symposium (SIEDS), 26-26 April 2019 2019. 1-4.
- ALLAN, D. & WEISS, M. 1980. *Accurate Time and Frequency Transfer During Common-View of a GPS Satellite*.
- AUTODESK INC. 2020. *Autodesk FBX-Dateien* [Online]. Verfügbar unter: <https://knowledge.autodesk.com/de/support/3ds-max/learn-explore/caas/CloudHelp/cloudhelp/2019/DEU/3DSMax-Data-Exchange/files/GUID-26E80277-1645-4C4E-A6B2-44399376490F-htm.html> [Zugriff am 26.05.2020].
- AZUMA, R., BAILLOT, Y., BEHRINGER, R., FEINER, S., JULIER, S. & MACINTYRE, B. 2001. Recent advances in augmented reality. *IEEE Computer Graphics and Applications*, 21, 34-47.
- AZUMA, R. T. 1997. A Survey of Augmented Reality. *Presence: Teleoperators and Virtual Environments*, 6, 355-385.
- BAE, Y., CHA, B. & RYU, J. 2020. Calibration and Evaluation for Visuo-haptic Collocation in Haptic Augmented Virtuality Systems. *International Journal of Control, Automation and Systems*, 18, 1335-1342.
- BAREL, A. 2017. *The differences between VR, AR & MR* [Online]. Verfügbar unter: <https://www.linkedin.com/pulse/differences-between-vr-ar-mr-avi-barel> [Zugriff am 05.05.2020].
- BLUMAX Bluetooth GPS Receiver. User's Guide. GPS-710.
- BLUT, C. & BLANKENBACH, J. 2020. Three-dimensional CityGML building models in mobile augmented reality: a smartphone-based pose tracking system. *International Journal of Digital Earth*.
- BUNDESAMT FÜR JUSTIZ. 2015. *Gesetz über die Elektrizitäts- und Gasversorgung (Energiewirtschaftsgesetz - EnWG). § 49 Anforderungen an Energieanlagen*. [Online]. Verfügbar unter: https://www.gesetze-im-internet.de/enwg_2005/__49.html [Zugriff am 10.02.2020].
- CLEANPNG.COM. 2018. *Erstellen Von Microsoft HoloLens Von Windows 10 Die Windows-Mixed Reality - Microsoft PNG - 800*400* [Online]. Verfügbar unter: <https://de.cleanpng.com/png-czryat/> [Zugriff am 18.09.2019].
- CONLEY, Q., ATKINSON, R. K., NGUYEN, F. & NELSON, B. C. 2020. MantarayAR: Leveraging augmented reality to teach probability and sampling. *Computers & Education*, 153, 103895.
- DELL. 2020. *Trademarks* [Online]. Verfügbar unter: <https://www.microsoft.com/en-us/legal/intellectualproperty/trademarks/usage/general> [Zugriff am 25.06.2020].
- DEUTSCHES INSTITUT FÜR NORMUNG 1975. Planwerke für die Versorgungswirtschaft, die Wasserwirtschaft und für Fernleitungen. *DIN 2425, Teil 1*.
- DVGW 2010. Arbeitsblatt GW 120: Netzdokumentation in Versorgungsunternehmen.
- ESRI INC. a. *ArcGIS Pro* [Online]. Verfügbar unter: <https://www.esri.de/arcgis/produkte/arcgis-pro> [Zugriff am 10.02.2020].
- ESRI INC. 2017. *Copyrights and Trademarks* [Online]. Verfügbar unter: <https://www.esri.com/de-de/legal/copyright-trademarks> [Zugriff am 25.06.2020].
- ESRI INC. b. *ArcGIS CityEngine*. [Online]. Verfügbar unter: <https://www.esri.com/de-de/store/city-engine> [Zugriff am 10.02.2020].
- ESRI INC. 2020. *Export FBX (Autodesk)* [Online]. Verfügbar unter: <https://doc.arcgis.com/en/cityengine/latest/help/help-export-fbx.htm> [Zugriff am 20.06.2020].

- EUROPEAN GLOBAL NAVIGATION SATELLITE SYSTEMS AGENCY. 2017. *What is GNSS?* [Online]. Verfügbar unter: <https://www.gsa.europa.eu/european-gnss/what-gnss> [Zugriff am 20.05.2020].
- EUROPEAN GLOBAL NAVIGATION SATELLITE SYSTEMS AGENCY. 2019. *What is SBAS?* [Online]. Verfügbar unter: <https://www.gsa.europa.eu/european-gnss/what-gnss/what-sbas> [Zugriff am 20.05.2020].
- EVANGELISTA, A., ARDITO, L., BOCCACCIO, A., FIORENTINO, M., MESSENI PETRUZZELLI, A. & UVA, A. E. 2020. Unveiling the technological trends of augmented reality: A patent analysis. *Computers in Industry*, 118, 103221.
- FAN, L., CHEN, J., MIAO, Y., REN, J. & WANG, Y. Multi-vehicle Cooperative Military Training Simulation System Based on Augmented Reality. 2019 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct), 10-18 Oct. 2019 2019. 128-133.
- FENAI, A., ARIARATNAM, S. T., AYER, S. K. & SMILOVSKY, N. 2019. Integrating geographic information systems and augmented reality for mapping underground utilities. *Infrastructures*, 4.
- FREE ICON LIBRARY. 2019. *3D Icon #255145* [Online]. Verfügbar unter: <http://icon-library.com/icon/3d-icon-12.html> [Zugriff am 18.09.2019].
- GARCÍA-PEREIRA, I., PORTALÉS, C., GIMENO, J. & CASAS, S. 2020. A collaborative augmented reality annotation tool for the inspection of prefabricated buildings. *Multimedia Tools and Applications*, 79, 6483-6501.
- GAZCÓN, N. F., NAGEL, J. M. T. B., ERNESTO A. & CASTRO, S. M. 2018. Fieldwork in Geosciences assisted by ARGeo: A mobile Augmented Reality system. *Computers and Geosciences*, 121, 30-38.
- GHULOUM, H. 3D hologram technology in learning environment. Proceedings of Informing Science & IT Education Conference (InSITE) 2010.
- GOOGLE INC. *Google Maps* [Online]. Verfügbar unter: <https://www.google.de/maps/> [Zugriff am 10.02.2020].
- GRINTEC. *Augview - Mobiles augmented Reality GIS* [Online]. Verfügbar unter <https://www.grintec.com/Augview> [Zugriff am 29.07.2019].
- HANID, M. F. A., MOHAMAD SAID, M. N. H. & YAHAYA, N. 2020. Learning Strategies Using Augmented Reality Technology in Education: Meta-Analysis. *Universal Journal of Educational Research*, 8, 51-56.
- HENDERSON, S. & FEINER, S. 2011. Exploring the Benefits of Augmented Reality Documentation for Maintenance and Repair. *IEEE Transactions on Visualization and Computer Graphics*, 17, 1355-1368.
- HERNANDEZ-DE-MENENDEZ, M., ESCOBAR DÍAZ, C. & MORALES-MENENDEZ, R. 2020. Technologies for the future of learning: state of the art. *International Journal on Interactive Design and Manufacturing (IJIDeM)*, 14, 683-695.
- HUANG, W., SUN, M. & LI, S. 2016. A 3D GIS-based interactive registration mechanism for outdoor augmented reality system. *Expert Systems With Applications*, 55, 48-58.
- IBÁÑEZ, M. B., DI SERIO, Á., VILLARÁN, D. & DELGADO KLOOS, C. 2014. Experimenting with electromagnetism using augmented reality: Impact on flow student experience and educational effectiveness. *Computers & Education*, 71, 1-13.
- KARAIM, M., ESHEIKH, M. & NOURELDIN, A. 2018. GNSS Error Sources. In: RUSTAMOV, R. B. & HASHIMOV, A. M. (eds.) *Multifunctional Operation and Application of GPS*. London: IntechOpen.
- KISSCCO. 2019. *Computer Icons ArcGIS Geographic Information System ArcMap User* [Online]. Verfügbar unter: <https://www.kisscco.com/clipart/computer-icons-arcgis-geographic-information-syste-b5qg3z/> [Zugriff am 18.09.2019].
- KOMPF, M. 2020. *Entfernungsberechnung* [Online]. Verfügbar unter: <https://www.kompf.de/gps/distcalc.html> [Zugriff am 09.03.2020].
- LANDESAMT FÜR GEOINFORMATION UND LANDENTWICKLUNG BADEN-WÜRTTEMBERG. 2020. *Koordinatenref.system* [Online]. Verfügbar unter: <http://www.sapos-bw.de/bezugssysteme.php> [Zugriff am 29.05.2020].

- LATTA, J. N. & OBERG, D. J. 1994. A conceptual virtual reality model. *IEEE Computer Graphics and Applications*, 14, 23-29.
- LEE, S., SUH, J. & PARK, H.-D. 2015. BoreholeAR: A mobile tablet application for effective borehole database visualization using an augmented reality technology. *Computers and Geosciences*, 76, 41-49.
- LGLN. SAPOS(R)-Dienste [Online]. Verfügbar unter: https://www.lgln.niedersachsen.de/startseite/online_angebote_amp_services/webdienste/sapos/sapos-services-und-bereitstellung-143814.html [Zugriff am 20.05.2020].
- LI, M. C., AMERUDIN, S. & MOHAMED YUSOF, Z. 2019. Development of Augmented Reality Pipeline Visualiser (ARPV) Application for Visualising Underground Water Pipeline. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, XLII-4/W16, 365-373.
- LI, W., HAN, Y., LIU, Y., ZHU, C., REN, Y., WANG, Y. & CHEN, G. 2018. Real-Time Location-Based Rendering of Urban Underground Pipelines. *ISPRS International Journal of Geo-Information*, 7, 32.
- MADSEN, J. B., TATZQERN, M., MADSEN, C. B., SCHMALSTIEG, D. & KALKOFEN, D. 2016. Temporal Coherence Strategies for Augmented Reality Labeling. *IEEE Transactions on Visualization and Computer Graphics*, 22, 1415-1423.
- MANRING, L., PEDERSON, J., POTTS, D., BOARDMAN, B., MASCARENAS, D., HARDEN, T. & CATTANEO, A. Augmented Reality for Interactive Robot Control. 2020 Cham. Springer International Publishing, 11-18.
- MICROSOFT. 2020a. *Erlernen von Tools und Architektur* [Online]. Verfügbar unter: <https://docs.microsoft.com/de-de/windows/mixed-reality/development> [Zugriff am 29.02.2020].
- MICROSOFT. 2020b. *HoloLens-Hardware (1. Generation)* [Online]. Verfügbar unter: <https://docs.microsoft.com/de-de/hololens/hololens1-hardware> [Zugriff am 29.02.2020].
- MICROSOFT. 2020c. *Microsoft Trademark & Brand Guidelines* [Online]. Verfügbar unter: <https://www.microsoft.com/en-us/legal/intellectualproperty/trademarks/usage/general> [Zugriff am 25.06.2020].
- MICROSOFT. 2020d. *MR Basics 100: Getting started with Unity* [Online]. Verfügbar unter: <https://docs.microsoft.com/de-de/windows/mixed-reality/holograms-100> [Zugriff am 21.12.2019].
- MICROSOFT. 2020e. *Systemanforderungen der Visual Studio-Produktfamilie 2019* [Online]. Verfügbar unter: <https://docs.microsoft.com/de-de/visualstudio/releases/2019/system-requirements> [Zugriff am 25.06.2020].
- MICROSOFT. 2020f. *What's a Universal Windows Platform (UWP) app?* [Online]. Verfügbar unter: <https://docs.microsoft.com/de-de/windows/uwp/get-started/universal-application-platform-guide> [Zugriff am 29.05.2020].
- MILGRAM, P. & COLQUHOUN, H. 2001. A Taxonomy of Real and Virtual World Display Integration.
- MILGRAM, P. & KISHINO, F. 1994. A Taxonomy of Mixed Reality Visual Displays. *IEICE Trans. Information Systems*, vol. E77-D, no. 12, 1321-1329.
- NEGRILLO-CÁRDENAS, J., JIMÉNEZ-PÉREZ, J.-R. & FEITO, F. R. 2020. The role of virtual and augmented reality in orthopedic trauma surgery: From diagnosis to rehabilitation. *Computer Methods and Programs in Biomedicine*, 191, 105407.
- NEVES, J., SERRARIO, D. & PIRES, J. N. 2018. Application of mixed reality in robot manipulator programming. *Industrial Robot: An International Journal*, 45, 784-793.
- NMEA. 2011. *NMEA.de. Infos zur NMEA Schnittstelle an Bord.* [Online]. Verfügbar unter: <http://www.nmea.de/index.html> [Zugriff am 29.02.2020].
- NMEA. 2013. *NMEA 0183 Datensätze* [Online]. Verfügbar unter: <http://www.nmea.de/nmea0183datensaetze.html> [Zugriff am 29.02.2020].
- OHSHIMA, T., SATOH, K., YAMAMOTO, H. & TAMURA, H. 1998. AR2 Hockey: A Case Study of Collaborative Augmented Reality. *Proceedings of the Virtual Reality Annual International Symposium*. IEEE Computer Society.

- OHSHIMA, T., SATOH, K., YAMAMOTO, H. & TAMURA, H. 1999. RV-Border Guards : A Multi-player Mixed Reality Entertainment. *Transactions of the Virtual Reality Society of Japan*, 4, 699-705.
- REGENBRECHT, H., OTT, C., WAGNER, M., LUM, T., KOHLER, P., WILKE, W. & MUELLER, E. 2003. *An augmented virtuality approach to 3D videoconferencing*.
- RIECKEN, J. & KURTENBACH, E. 2017. Der Satellitenpositionierungsdienst der deutschen Landesvermessung - SAPOS(R). *Zeitschrift für Geodäsie, Geoinformation und Landmanagement*, 142, 1-8.
- RIZOS, C. 2020. Trends in GPS Technology & Applications.
- SAFE SOFTWARE INC. 2020a. *Power the Flow of Data with FME* [Online]. Verfügbar unter: <https://www.safe.com/fme/> [Zugriff am 29.05.2020].
- SAFE SOFTWARE INC. 2020b. *Safe Software Trademark Usage Guidelines* [Online]. Verfügbar unter: <https://www.safe.com/legal/terms-and-conditions/trademark-usage-guidelines/> [Zugriff am 25.6.2020].
- SARI, F. 2020. Spatial disparities in accessibility to recreational amenities: the case of Pokémon GO. *Annals of Regional Science*, 64, 389-419.
- SCHALL, G., JUNGHANNS, S. & SCHMALSTIEG, D. 2010. Vidente - 3D Visualization of Underground Infrastructure using Handheld Augmented Reality. *Geohydroinformatics: Integrating GIS and Water*. CRC Press.
- SCHALL, G., MENDEZ, E., KRUIJFF, E., VEAS, E., JUNGHANNS, S., REITINGER, B. & SCHMALSTIEG, D. 2009. Handheld Augmented Reality for underground infrastructure visualization. *Personal and Ubiquitous Computing*, 13, 281-291.
- SCHALL, G., ZOLLMANN, S. & REITMAYR, G. 2013. Smart Vidente: advances in mobile augmented reality for interactive visualization of underground infrastructure. *Personal and Ubiquitous Computing*, 17, 1533-1549.
- SCHEU, M. 2015. Netzinformationssysteme und die Herausforderungen der Energiewende. *Zeitschrift für Geodäsie, Geoinformation und Landmanagement*, 140, 381-385.
- SCHNEIDER, M. 2013. *Einführung in die Physiologie des Menschen*, Springer-Verlag.
- SEDLMAJER, K., BAMBUŠEK, D. & BERAN, V. 2019. Effective Remote Drone Control using Augmented Virtuality. *Proceedings of the 3rd International Conference on Computer-Human Interaction Research and Applications - Volume 1: CHIRA*. Wien.
- SHA, X., JIA, Z., SUN, W., HAO, Y., XIAO, X. & HU, H. Development of Mixed Reality Robot Control System Based on HoloLens. 2019 Cham. Springer International Publishing, 571-581.
- SIELHORST, T., FEUERSTEIN, M. & NAVAB, N. 2008. Advanced Medical Displays: A Literature Review of Augmented Reality. *Journal of Display Technology*, 4, 451-467.
- STEIN, P. D.-I. D. 2003. *DEMO Grabenloser Leitungsbau - Vorwort* [Online]. Verfügbar unter: <http://www.unitracc.com/know-how/fachbuecher/demo-grabenloser-leitungsbau/vorwort> [Zugriff am 18.06.2020].
- SUTHERLAND, I. E. 1968. A head-mounted three dimensional display. *Proceedings of the December 9-11, 1968, fall joint computer conference, part I*. San Francisco, California: Association for Computing Machinery.
- TERNIER, S., KLEMKE, R., KALZ, M., ULZEN, P. & SPECHT, M. 2012. AR Learn: Augmented reality meets augmented virtuality. *Journal of Cheminformatics - J Cheminf*, 18.
- THEDIGITALARTIST. 2017. *charakter-3d-mann-konzeptionellen-2732490* [Online]. Verfügbar unter: <https://pixabay.com/de/illustrations/charakter-3d-mann-konzeptionellen-2732490/> [Zugriff am 20.05.2020].
- THEIS, T. 2018. *Einstieg in Unity. 2D- und 3D-Spiele entwickeln.*, Bonn, Rheinwerk Verlag.
- TREPPER, M. O., RUDY, H. L., LEFKOWITZ, A., WEIMER, K. A., MARKS, S. M., STERN, C. S. & GARFEIN, E. S. 2017. Mixed reality with hololens: Where Virtual Reality Meets Augmented Reality in the Operating Room. *Plastic and Reconstructive Surgery.*, 140, 1066-1070.
- ULLO, S. L., PIEDIMONTE, P., LECCESE, F. & DE FRANCESCO, E. 2019. A step toward the standardization of maintenance and training services in C4I military systems with Mixed Reality application. *Measurement*, 138, 149-156.

-
- UNITY TECHNOLOGIES. 2019. *Supported Model file formats* [Online]. Verfügbar unter: <https://docs.unity3d.com/2019.4/Documentation/Manual/3D-formats.html> [Zugriff am 29.05.2020].
- UNITY TECHNOLOGIES. 2020. *System requirements for Unity 2019.4* [Online]. Verfügbar unter: <https://docs.unity3d.com/Manual/system-requirements.html> [Zugriff am 25.06.2020].
- VDE 2010. VDE-AR-N 4201: Netzdokumentation. In: (FNN), F. N. N. (ed.).
- WANG, W., WU, X., CHEN, G. & CHEN, Z. 2018. Holo3DGIS: Leveraging Microsoft HoloLens in 3D Geographic Information. *International Journal of Geo-Information*, 7, 1-16.
- WEIß INGENIEURE. *Professionell projektiert. Optimal realisiert. Ausgewählte Referenzprojekte von Weiß Beratende Ingenieure.* [Online]. Verfügbar unter: <https://www.weiss-ingenieure.de/leistungen/referenzen/> [Zugriff am 02.05.2020].
- WIJESOORIYA, I., WIJewardana, D., SILVA, T. D., GAMAGE, C. & JAYaweera, M. Demo Abstract: Enhanced Real-Time Machine Inspection with Mobile Augmented Reality for Maintenance and Repair. 2017 IEEE/ACM Second International Conference on Internet-of-Things Design and Implementation (IoTDI), 18-21 April 2017 2017. 287-288.
- ZHANG, X., HAN, Y., HAO, D. & LV, Z. 2016. ARGIS-based outdoor underground pipeline information system. *Journal of Visual Communication and Image Representation*, 40, 779-790.
- ZHU, J., ONG, S. K. & NEE, A. Y. C. 2015. A context-aware augmented reality assisted maintenance system. *International Journal of Computer Integrated Manufacturing*, 28, 213-225.

Anhang

Anhang 1: Muster einer manuellen Aufnahmeskizze der Sparte Strom

Leistungs- u. Materialnachweis, Einmessungsriß

Elektro- Hausanschluss Blatt 1

Name: Bernd Mustermann	Bauzeit von: 27.04.2009	bis: 29.04.2009
Ort: Musterhausen	Serviceauftrag Strom: 4711	
	Ortsnetz-Nr.: 657	
Straße und Haus-Nr.: Hauptstraße 17	Straßenschlüssel: 47123	

Pauschalleistung:		Material:	Nummer:	Einh.	Menge:
1.1	Grundbetrag Elektro: <input type="checkbox"/> 1.1.1 Grundbetrag Elektro (vorverlegt): <input type="checkbox"/>	NAY2Y-J 4 x 50 mm ²	17 850	m	14,0
1.2	Grundbetrag Elektro u. Gas: <input type="checkbox"/> Anteil Elektro: <input type="checkbox"/> Anteil Gas siehe Blatt 2	NAY2Y-J 4 x	17 895	m	
1.3	Längenabhängiger Betrag: Grabenlänge auf dem Privatgrundstück m durch Baufirma: m durch Eigenleistung: m	Trassenwarnband 150 mm breit	28 173	m	
1.4	Teilverfüllung (mind. 0,30 m Überdeckung) + Warnband durch Baufirma m	Vortriebsrohr 50 x 2,4 x 1000 mm, rot	26 107	m	
2.	Kabelverlegung bis 30 m Grabenlänge, einschließlich Mauerdurchbruch: <input type="checkbox"/>	Kabelschutzrohr d = 110 mm, Länge 6 m	26 056	m	
3.	Elektronontage: HA-Kasten und HA-Muffe montieren: <input type="checkbox"/>	Kunststoffrohr flexibel 36 mm	30 099	m	5,0
Weitere Eigenleistungen durch den Kunden: Mauerdurchführung und Abdichtung bzw. Mehrsparteneinführung: <input type="checkbox"/> Tiefbau im öffentlichen Bereich: <input type="checkbox"/>		Schrumpfschlauch, kleberbeschichtet, d = 26 - 60 mm	24 325	St.	
HA-Kasten montiert am: 29.04.2009		Paketset mit Muffe und Schraubklemmring 4 x 150/70 mm ²	26 001	St.	
Name: Meier Unterschrift: <i>Meier</i>		Kabelhausanschlusskasten 3 x 100	27 754	St.	1
HA-Muffe montiert am: 29.04.2009		Sicherung NH00 63A	27 867	St.	
Name: Schulze Unterschrift: <i>Schulze</i>					
		Erstprüfung nach BGV A3 am: 29.04.2009			
		Name: Müller Unterschrift: <i>Müller</i>			

Skizze:

Datum: 29.04.2009	Ort: Musterhausen	Baufirma: MSM	Bauaufsicht: <i>Müller</i>
-----------------------------	-----------------------------	-------------------------	-------------------------------

Abbildung 39: Manuelle Aufnahmeskizze Strom
Quelle: VDE (2010)

Anhang 2: Muster einer digitalen Aufnahmeskizze der Sparte Strom

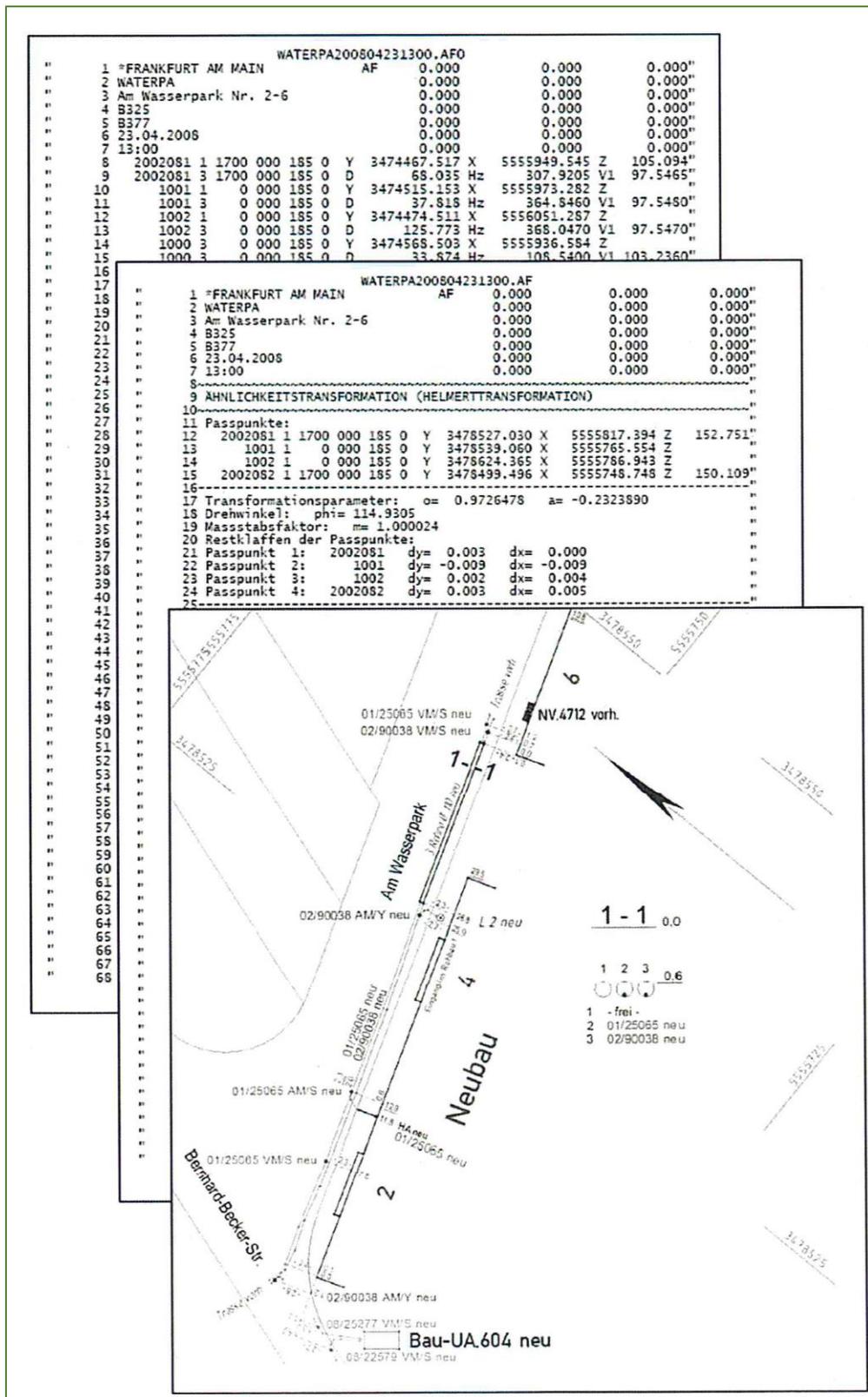


Abbildung 40: Digitale Aufnahmeskizze Strom
Quelle: VDE (2010)

Anhang 3: Muster eines Bestandplans Strom in Mehrstrichdarstellung

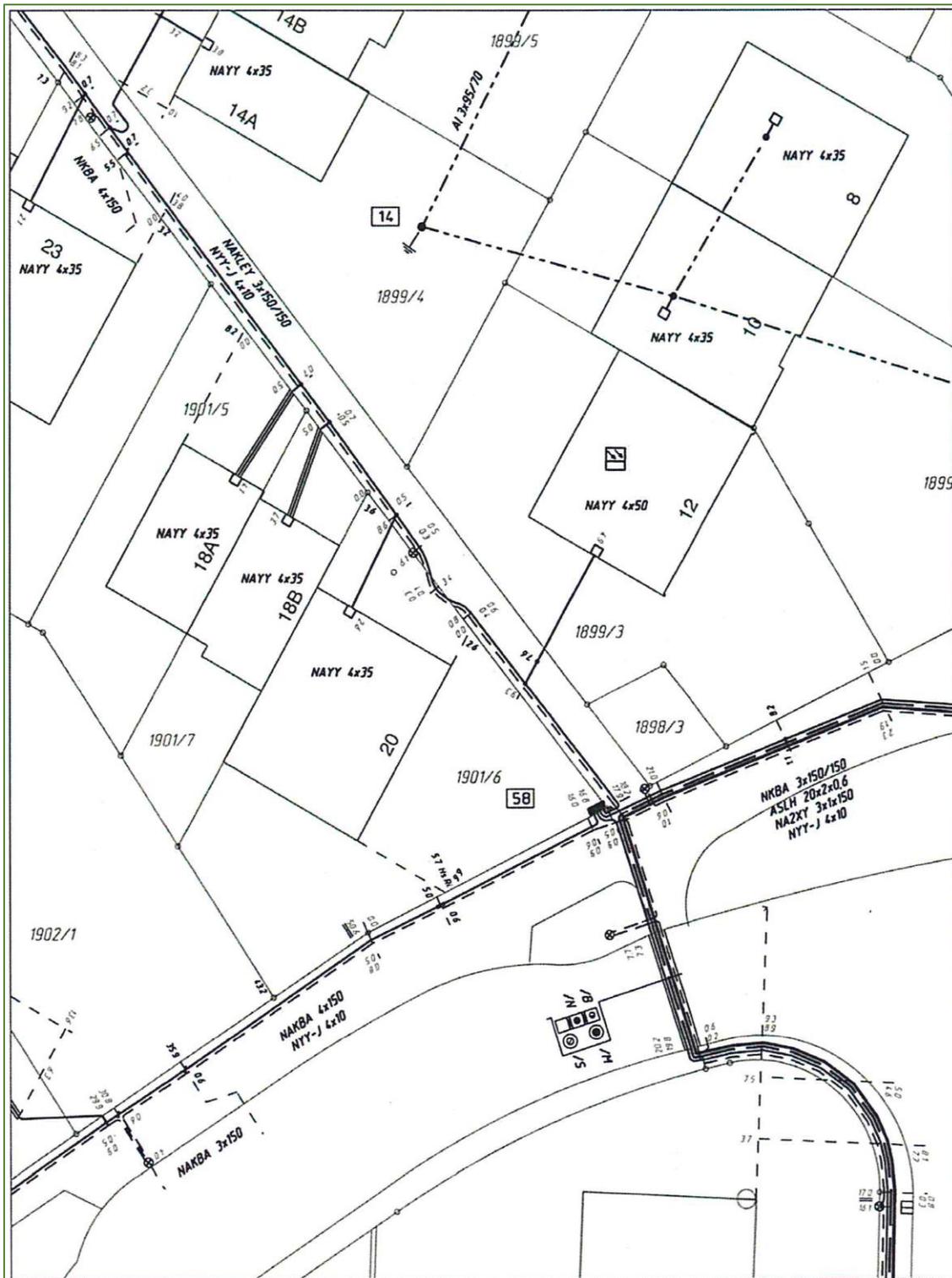


Abbildung 41: Bestandsplan Strom in Mehrstrichdarstellung
Quelle: VDE (2010)

Angang 4: Muster eines Übersichtsplans Strom



Abbildung 42: Übersichtsplan Strom 1:5000
Quelle: VDE (2010)

Angang 5: Mindestanforderungen für die grafische Ausprägung der Pläne der Sparte Strom

Tabelle 6: Bestandteile und Inhalte der Netzdokumentation Strom, Teil 1

Dateninhalte		Aufnahmeskizze	Stromnetzplan (Netze mit Verteilungsfunktion)		Stromnetzplan (Netze mit Übertragungsfunktion)			
			Bestandsplan	Übersichtsplan	Bestandsplan	Profilplan	Übersichtsplan	
Geobasisdaten	Gebäude	X	X	(x)	X	X	(x)	
	Topografie (ohne Gebäude)	(x)	(x)	X	(x)	(x)	X	
	Grenzen	(x)	X	(x)	(x)		(x)	
	Gemarkungsname, Flur-, Flurstücksnummer	(x)	(x)	(x)	(x)		(x)	
	Straßenamen	X	X	(x)	x		(x)	
	Hausnummer	X	X	(x)	x		(x)	
	Ergänzende Bezeichnung (z. B. Gewinnbezeichnung, Kilometrierung von Verkehrswegen)	(x)	(x)	(x)	(x)	(x)	(x)	
Netzkanten	Hoch- und Höchstspannungsnetz	110 kV – 380 kV	x	x	x	x	x	
	Mittelspannungsnetz	<110 kV – >1 kV	x	x	x			
	Niederspannungsnetz	1 kV	x	x	x			
	Beleuchtungsnetz		x	x	x			
	Netzanschlussleitung		x	x	(x)			
	Kommunikationsnetz		x	x	x	(x)	x	
	Fremdleitungen		(x)	(x)	(x)	(x)	(x)	
	Leitungen	in Betrieb	x	x	x	x	x	x
		außer Betrieb	x	x	(x)	x	x	x
		stillgelegt	x	x		x	(x)	
Leitungsangaben (geführte Sachdaten)		x	(x)	(x)	(x)	(x)	(x)	

Quelle: VDE (2010)

Tabelle 7: Bestandteile und Inhalte der Netzdokumentation Strom, Teil 2

Dateninhalte		Aufnahmeskizze	Stromnetzplan (Netze mit Verteilungsfunktion)		Stromnetzplan (Netze mit Übertragungsfunktion)		
			Bestandsplan	Übersichtsplan	Bestandsplan	Profilplan	Übersichtsplan
Netzknoten	Erzeugung/UW/Stationen	Darstellung	x	x	x	x	x
		Anlagenbezeichnung	x	(x)	(x)	(x)	(x)
		Art	x	x	x	x	x
	KVS	Darstellung	x	x	x		
		Anlagenbezeichnung	x	(x)	(x)		
		Art	x	(x)	(x)		
	Netzanschluss	Darstellung	x	x	(x)		
	Muffe	Darstellung	x	x	(x)	x	x
	Mast	Darstellung	x	x	x	x	x
		Werkstoff	x	(x)	(x)	(x)	(x)
Bezeichnung		x	(x)	(x)	(x)	(x)	
Sonstiges	Maßangaben	Darstellung	x	(x)	(x)	(x)	
	Schutzrohr	Darstellung	x	x	(x)	x	x
		Dimension	x	(x)	(x)	(x)	(x)
		Werkstoff	x	x	(x)	(x)	x
Grabenprofil	Darstellung	(x)	(x)		(x)		

Legende
x Mindestanforderung
(x) Empfehlung

Quelle: VDE (2010)

Tabelle 8: Symbol- und Signaturkatalog Strom

Bedeutung	Symbol	Bedeutung	Symbol
Leitungen (alle Spannungs- und Funktionsebenen)	—	Netzanschluss (HA-Kasten, HA-Säule)	□
Netzanschlussleitung (alle Spannungs- und Funktionsebenen)	—	Muffe	●
Erzeugung allgemein	□	Abzweigmuffe	—●—
Erzeugung Wasserkraft	◻	Querschnitts- bzw. Materialwechsel	/
Erzeugung fossile Energieträger (z. B. Kohle, Gas, Öl)	◻	Mast oder Turmstation	○
Erzeugung Photovoltaik	◻	Mast sowie Dachständer	○
Erzeugung Windkraft	◻	Gittermast	□
Erzeugung HKW (z. B. BHKW)	◻	Leuchte	⊗
Station, UW, Schaltstation	○	Anode	▽
KVS	⏏	Erder	⏏
Überspannungsableiter	⏏		
Trassenband ggf. mit Beschriftung	▨		
Schutzrohr ggf. mit Beschriftung	▭		

Quelle: VDE (2010)

Anhang 6: Muster einer manuellen Aufnahmeskizze für Gas und Wasser

Aufnahmeskizze

Baustelle: Zehlendorf, Hauptstraße 16
Gemeinde/Ort, Ortsteil, Straße, Haus-Nr., Hausanschlussnummer

Hausanschluss Versorgungsleitung Sanierung Regelanlage Leckstelle

Netzgebiet: EMB Bl.-Nr.: 1/1 Rohrleitungsbau-Fa.: RBU

EQ-,TP-Nr.: 50123125

Projekt-Nr.:

Auftrags-Nr.: 41015664

gemessen am: 10.05.2005 durch (Name): Meier gez. durch: Mi

Bemerkung: Neubau (verkleinert)

Skizze

The sketch depicts a plan view of a building and its connection to a main street. The building is labeled '16' and has a gas meter 'Ga' with a diameter of '3,0'. A gas inlet 'GS' is shown on the main street 'Hauptstraße'. The main street has a diameter of 'd 90 PE'. A gas pipe with a diameter of 'd 32 PE' runs from the main street to the building. The sketch shows various pipe segments with dimensions: a vertical segment of '10,2', a horizontal segment of '9,9', a diagonal segment of '5,8', and a vertical segment of '2,2'. The main street has a diameter of 'D 98'. A north arrow is located on the left side of the sketch. A legend in the top right corner shows a symbol for 'GAS' and a checked box for 'Hausanschluss'.

Abbildung 43: Manuelle Aufnahmeskizze Gas und Wasser

Quelle: DVGW (2010)

Anhang 7: Muster einer digitalen Aufnahmeskizze für Gas und Wasser

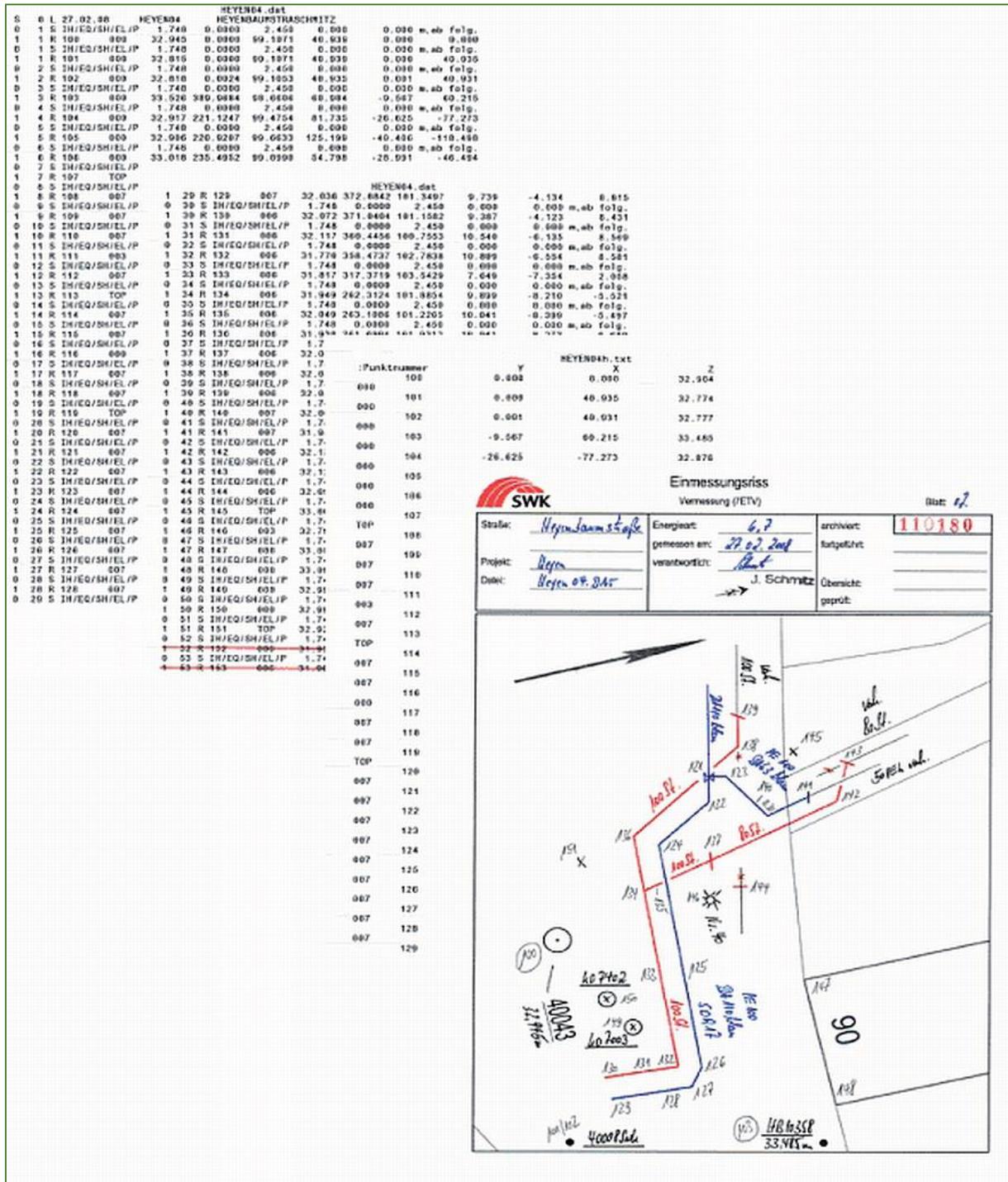


Abbildung 44: Digitale Aufnahmeskizze Gas und Wasser
Quelle: VDE (2010)

Anhang 8: Inhalte der Aufnahmeskizze für Gas und Wasser

Tabelle 9: Inhalte einer Aufnahmeskizze Gas und Wasser

Bestandteile und Inhalte (Legende: X muss : (X) kann)		Auf- nahme- skizze	Rohrnetzplan							
			Verteilnetze		Rohrfernleitungen					
			Be- stands- plan	Über- sichts- plan	Be- stands- plan	Be- stands- plan (Längs- schnitt)	Über- sichts- plan	Bau- plan	Rechts- er- werbs- plan	
Geobasis- daten	Gebäude	X	X	(X)	X		(X)	X	X	
	Topographie (ohne Gebäude)	(X)	(X)	(X)	(X)		(X)	(X)	(X)	
	Grenzen	(X)	(X)	(X)	(X)		(X)	(X)	X	
	Gemarkungs-, Flur-, Flurstücksbezeichnung	(X)	(X)		(X)			(X)	X	
	Straßennamen	X	X	(X)	X	(X)	(X)	X	X	
	Hausnummer	X	X		X			X	X	
	Eigentümergebungen							(X)	(X)	
	Verkehrswege inklusive Kilometrierung		(X)		(X)	(X)		(X)	(X)	
Maßangaben		X	(X)		(X)			(X)		
Netzdaten	Leitungen	Versorgungsleitungen	X	X	X	X	X	X	X	X
		Hausanschluss- leitungen	X	X	(X)					
		stillgelegte Leitungen	X	X		X				
		geplante Leitungen		(X)	(X)	(X)		(X)	X	X
	Leitungsschutz- streifen								X	
	Leitungsangaben	Leitungsart (Gas, Wasser)	(X)	(X)	(X)	(X)		(X)	(X)	(X)
		Dimension	X	X	(X)	X	X	(X)	X	X
		Material	X	X	(X)	X		(X)	X	(X)
		Verbindungsart	X	(X)		(X)	X			
		Rohrumhüllung	X	(X)		(X)			X	
		Rohrrinnenschutz	(X)	(X)		(X)			X	
		Verlegedatum	X	(X)	(X)	X	X	(X)		
		Rohroberkante m. u. NN (NHN)					X			
		Geländehöhe m. u. NN (NHN)					X			
		DP (Design Pressure)	(X)	(X)	(X)	X		(X)	X	
		MOP (Maximum Operating Pressure)		(X)	(X)	(X)	(X)	(X)		
	Mantelrohr	Dimension	X	X		X	X		X	(X)
		Material	X	(X)		X			X	(X)
	Einbauteile	Typ/Funktion	X	X		X	(X)		X	
		Nummer		(X)		(X)	(X)			
	Anlagen	Typ/Funktion	X	X	(X)	X		(X)	X	(X)
		Nummer		(X)	(X)	(X)		(X)		
		Name		(X)	(X)	(X)		(X)	(X)	
	Sonstiges	Reparaturstelle	X	(X)		(X)				
		Zustands-/ Schadensstelle		(X)	(X)	(X)	(X)	(X)		
		Rohrfolge mit Längenangaben	(X)			(X)				
		Lage zu fremden Versorgungsanlagen bei Kreuzungen, etc.	(X)	(X)		(X)	(X)		(X)	
Kabel		X	X	(X)	X		(X)	X	(X)	
kathodische Korro- sionsschutzanlagen		X	X	(X)	X		(X)	X	X	
Schilderpfähle	(X)	(X)		X			(X)	(X)		

Quelle: VDE (2010)

Anhang 9: Muster eines Bestandsplanes für Gas und Wasser

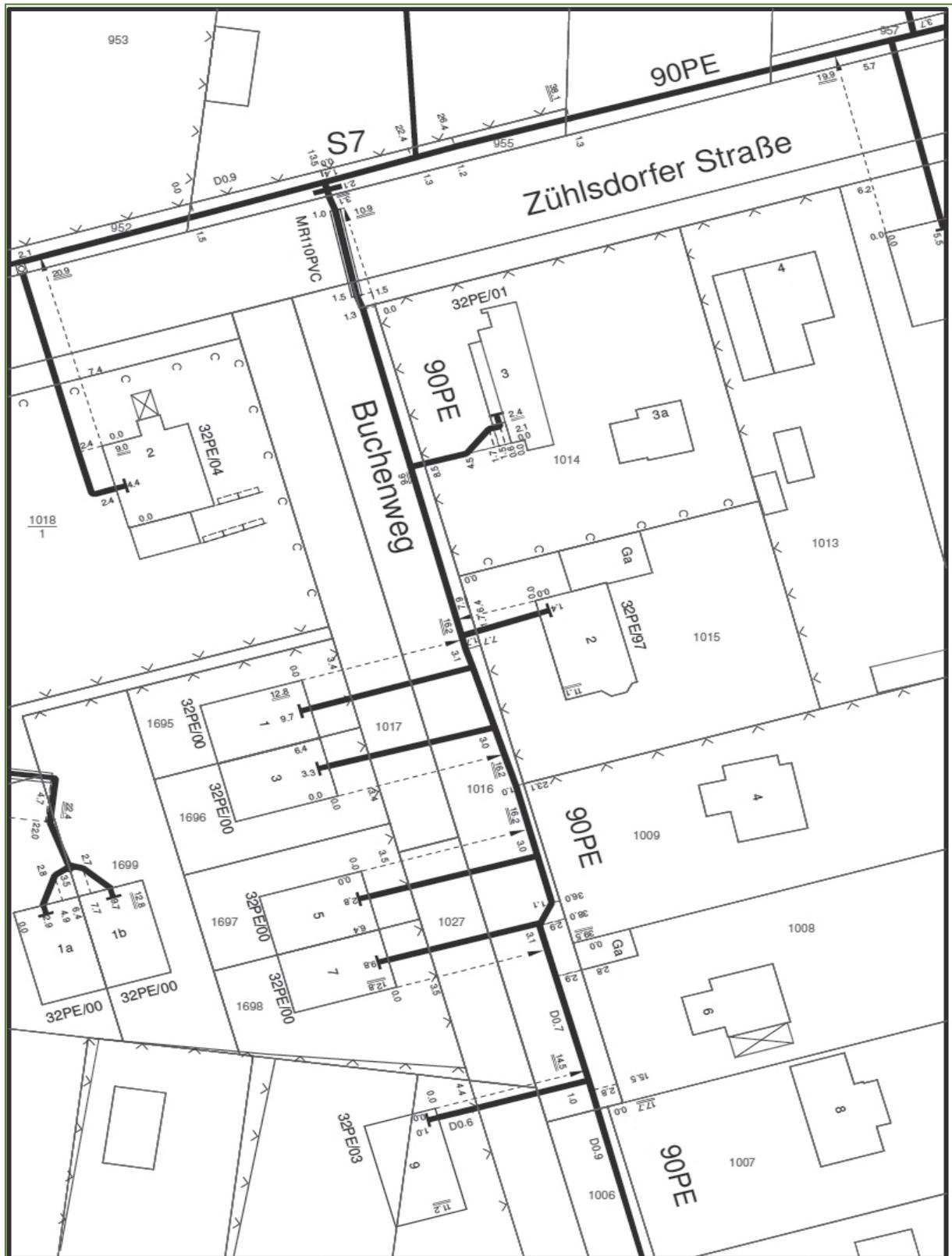


Abbildung 45: Bestandsplan Gas, 1:500

Quelle: VDE (2010)



Abbildung 46: Bestandsplan Wasser, 1:500
Quelle: VDE (2010)

Anhang 10: Muster eines Übersichtsplanes für Gas und Wasser

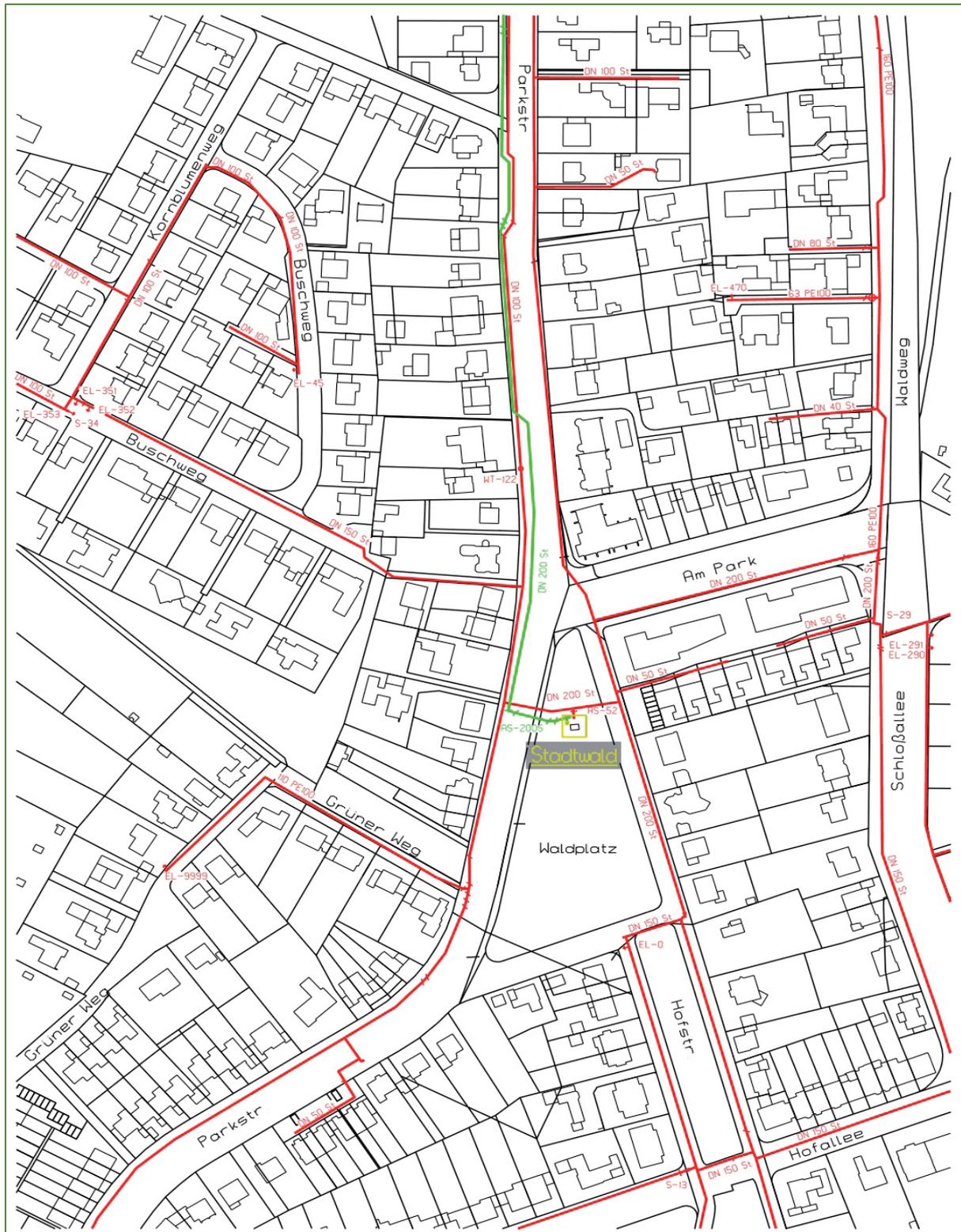


Abbildung 47: Übersichtsplan Gas, 1:5000
 Quelle: VDE (2010)

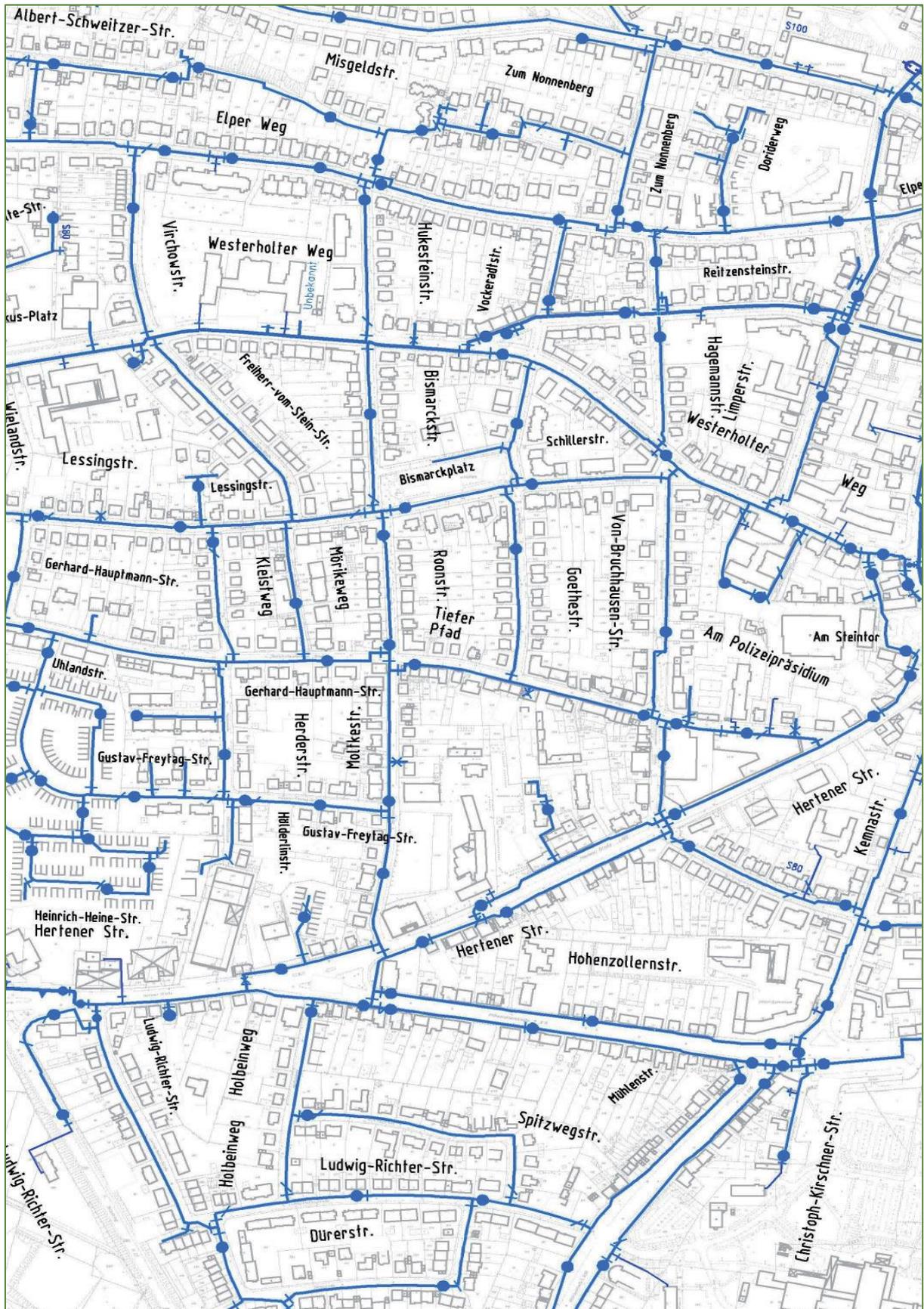


Abbildung 48: Übersichtsplan Wasser, 1:5000

Quelle: VDE (2010)

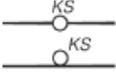
Anhang 11: grafische Vorschriften für Gas und Wasser

Tabelle 10: Gemeinsame grafische Vorschriften für Gas und Wasser

Leitung in Betrieb	
geplante Leitung	
unsichere Leitungslage	
Leitung stillgelegt	
Absperrarmatur	
Dehner	
Trennstelle	
Leitungsabschluss	
Druckregelanlage	
Rückflussverhinderer	
Mantelrohr	
Schutzstreifen	
Nennweiten/Material/Baujahr-Übergang	
Zählereinrichtung außerhalb von Gebäuden	
Kabel	
Kabelschrank	
Kathod. Korrosionsschutzanlage KKS	
KKS-Messstelle (verzogen, auf Rohr)	
KKS Anode	
Schilderpfahl	

Quelle: VDE (2010)

Tabelle 11: Zeichenvorschrift Gas

Ausbläser	
Gasströmungswächter	
Kondensatsammler im Rohr neben dem Rohr	
Blasenloch (z. B. Anbohrung mit DN 25)	

Quelle: VDE (2010)

Tabelle 12: Zeichenvorschrift Wasser

Entleerungsschacht	
Unterflurhydrant	
Überflurhydrant	
Entlüftung	

Quelle: VDE (2010)

Anhang 12: Flussdiagramm über die Datenaufbereitung

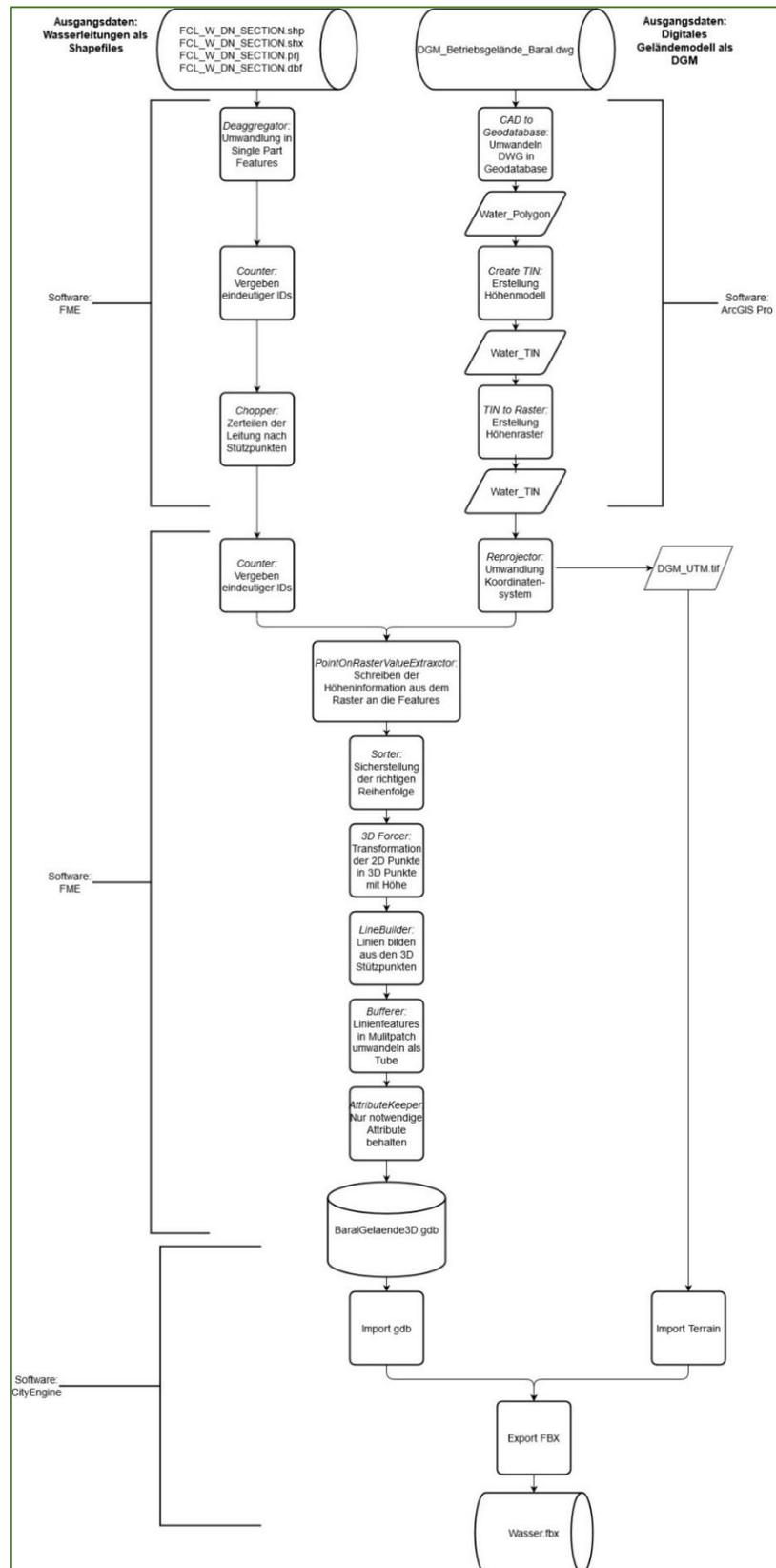


Abbildung 49: Flussdiagramm zur Datenaufbereitung
Quelle: Eigene Darstellung

Anhang 13: Flussdiagramm über die Entwicklung

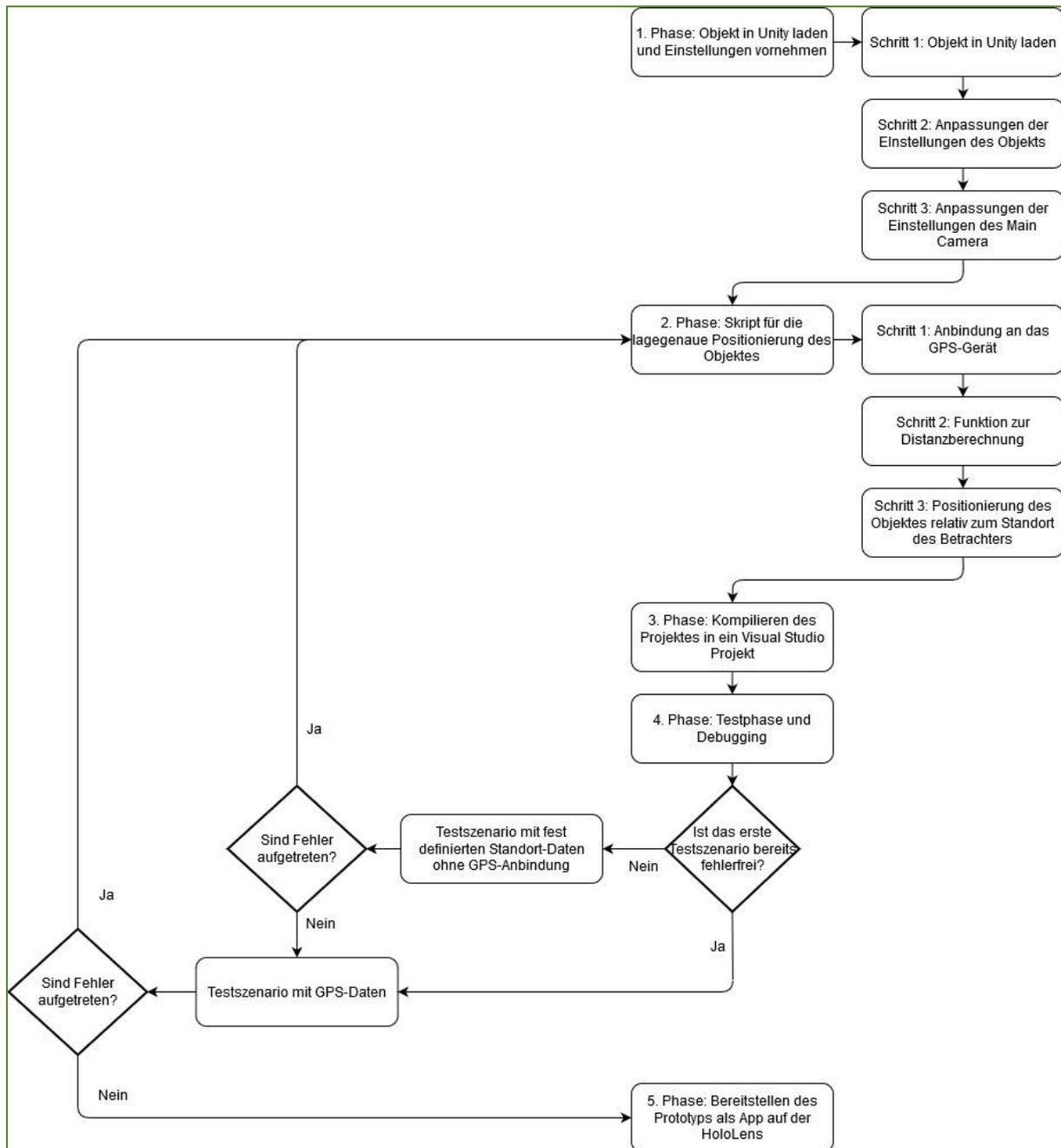


Abbildung 50: Flussdiagramm zur Entwicklung

Quelle: Eigene Darstellung

Anhang 14: Gesamter Scriptcode

```
1 using UnityEngine;
2 using System.IO.Ports;
3 using System;
4 using System.Text;
5 using System.Collections.Generic;
6
7
8 #if ENABLE_WINMD_SUPPORT
9 using Windows.Devices.Enumeration;
10 using Windows.Devices.Bluetooth.Rfcomm;
11 using Windows.Devices.Bluetooth;
12 using Windows.Networking.Sockets;
13 using System.Linq;
14 using Windows.Storage.Streams;
15 using Windows.Devices.SerialCommunication;
16 #endif
17
18 public class TransformPosition : MonoBehaviour
19 {
20
21
22 #if UNITY_EDITOR
23     // Variablen setzen
24     SerialPort serialPort = null;
25     string portName = "COM3";
26     int baudRate = 115200;
27     int readTimeOut = 10000;
28     string handshake = "2";
29 #endif
30
31     // Definition der Koordinaten des fbx-Modells:
32     // Koordinaten setzen nach dem offset des Exports (logfile of the model)
33     double lat = 48.5176;
34     double lon = 9.198945;
35     int bufferSize = 32;
36
37
38
39
40 #if UNITY_EDITOR
41     void Start()
42     {
43         try
44         {
45             // Port für die Datenübertragung öffnen.
46             // Nachricht, wenn der Port geöffnet werden konnte.
47             // Wenn nicht -> Catch-Block und Nachricht, dass es nicht funktioniert hat.
48
49             serialPort = new SerialPort();
50             serialPort.PortName = portName;
51             serialPort.BaudRate = baudRate;
52             serialPort.ReadTimeout = readTimeOut;
53             serialPort.Handshake = (Handshake)Enum.Parse(typeof(Handshake), handshake, true);
54             serialPort.Open();
55             Debug.Log("start getting gps information");
56         }
57
58         catch (Exception e)
59         {
60             Debug.Log("start getting gps information failed");
61             Debug.Log(e.Message);
62         }
63
64         // Daten empfangen.
65         // Hierfür erstellen eines Buffers, in den die Daten gelesen werden.
66
67         Byte[] buffer = new Byte[bufferSize];
68         int bytesRead = 0;
69
70         try
71         {
72
```

```

73     bytesRead = serialPort.Read(buffer, 0, bufferSize);
74
75     if (bytesRead > 0)
76     {
77         for (int i = 0; i < bufferSize; i++)
78         {
79             StringBuilder sb = new StringBuilder(serialPort.ReadLine());
80             string data = sb.ToString();
81
82             // Filtern nach dem String, der den benötigten Satz enthält
83             if (data.StartsWith("$GPGGA"))
84             {
85                 char[] separator = new char[] { ',' };
86                 string[] dataSplit = data.Split(separator, StringSplitOptions.None);
87
88                 // Prüfen, ob Daten empfangen wurden.
89                 // Wenn ja, Aufruf der Funktion zur Positionierung.
90                 // Wenn nein, Meldung, dass keine Daten empfangen wurden.
91                 if (dataSplit[2].Length > 2)
92                 {
93                     Vector3 itemPosition = NmeaToLatLon(dataSplit, data);
94
95                     // setzen der Position des Objektes
96                     transform.position = itemPosition;
97
98                     Debug.Log("position of water object" + transform.position);
99                 }
100
101                 else
102                 {
103                     Debug.Log(data);
104                     Debug.Log("Es können momentan keine GPS Daten empfangen werden");
105                 }
106             }
107         }
108     }
109 }
110 catch (Exception e)
111 {
112     UnityEngine.Debug.Log(e.Message);
113 }
114 }
115 #endif
116
117 #if ENABLE_WINMD_SUPPORT
118
119
120 async void Start()
121 {
122
123     StreamSocket _socket;
124     // Zunächst Suchen aller an der HoloLens registrierten Bluetooth Geräte
125     DeviceInformationCollection deviceInfoCollection = await DeviceInformation.
FindAllAsync(RfcommDeviceService.GetDeviceSelector(RfcommServiceId.SerialPort));
126
127     // Das gewünschte GPS-Gerät aus der Liste extrahieren
128     Debug.Log("--> target Bluetooth BR Device initialized:" + deviceInfoCollection[0].
Name + ", " + deviceInfoCollection[0].Id);
129     var deviceInfo = deviceInfoCollection[0];
130     var service = await RfcommDeviceService.FromIdAsync(deviceInfo.Id);
131     Debug.Log(service.ConnectionServiceName + ", " + service.ConnectionHostName);
132
133     // Socket für den Empfang der Daten erstellen und zum Bluetooth-Gerät verbinden
134     _socket = new StreamSocket();
135     await _socket.ConnectAsync(service.ConnectionHostName, service.
ConnectionServiceName);
136     Debug.Log("Connected");
137
138     // Daten empfangen und speichern
139     var dataReader = new DataReader(_socket.InputStream);
140     Debug.Log("dataReader created");
141     uint s = await dataReader.LoadAsync(256);
142     string dataString = dataReader.ReadString(s);
143     string[] dataArray = dataString.Split(new[] { Environment.NewLine },
StringSplitOptions.None);
144
145     int length = dataArray.Length;

```

```
146     int i;
147
148     for(i = 0; i < length; i++ )
149     {
150         if (dataArray[i].StartsWith("$GPGGA"))
151         {
152             char[] separator = new char[] { ',' };
153             string[] dataSplit = dataArray[i].Split(separator, StringSplitOptions.None);
154
155             if (dataSplit[2].Length > 2)
156             {
157                 Vector3 itemPosition = NmeaToLatLon(dataSplit, dataArray[i]);
158                 transform.position = itemPosition;
159                 Debug.Log("position of water object" + transform.position);
160             }
161
162             else
163             {
164                 Debug.Log(dataArray[i]);
165                 Debug.Log("Es können momentan keine GPS Daten empfangen werden");
166             }
167         }
168     }
169 }
170 }
171
172 #endif
173
174 public Vector3 NmeaToLatLon(string[] dataSplit, string data)
175 {
176     Debug.Log(data);
177
178     // Splitten des Strings und extrahieren der Koordinaten aus dem GPS-Gerät
179
180     string latMin = (dataSplit[2].Replace('.', ',')).Substring(2);
181     string latDegree = (dataSplit[2].Replace('.', ',')).Substring(0, 2);
182
183     string lonMin = (dataSplit[4].Replace('.', ',')).Substring(3);
184     string lonDegree = (dataSplit[4].Replace('.', ',')).Substring(0, 3);
185
186     double latCamera = double.Parse(latMin) / 60 + double.Parse(latDegree);
187     double lonCamera = double.Parse(lonMin) / 60 + double.Parse(lonDegree);
188
189     Debug.Log("data readed Latitude: " + latCamera + " data readed Longitude: " +
190 lonCamera);
191
192     string[] position = { Convert.ToString(latCamera), Convert.ToString(lonCamera) };
193
194     // Berechnen der Distanz zwischen den GPS-Daten und dem Objekt
195     var longitude = 71.5 * (lon - lonCamera) * 1000;
196     var latitude = 111.3 * (lat - latCamera) * 1000;
197
198     Debug.Log(longitude + ", " + latitude);
199
200     // Ausgabe des Vektors der Position des Objektes
201     return new Vector3
202     {
203         x = (float)longitude,
204         y = 0,
205         z = (float)latitude
206     };
207 }
208 }
```

Anhang 15: Flussdiagramm zur Funktionsweise des Skriptes zur Positionierung des Objektes

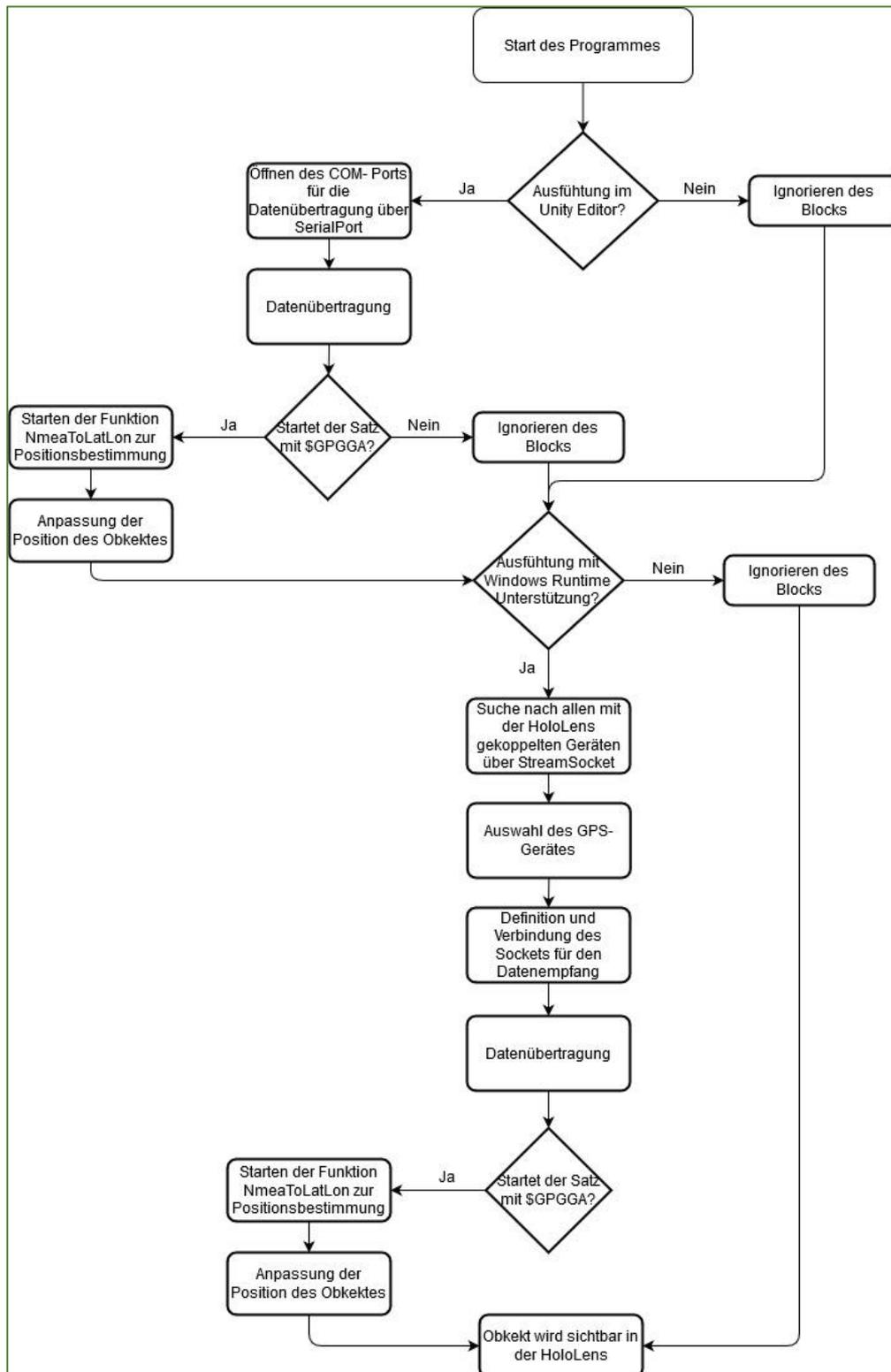


Abbildung 51: Flussdiagramm zur Funktionsweise des Skriptes zur Positionierung des Objektes

Quelle: Eigene Darstellung

Anhang 16: Rechtliche Hinweise

In dieser Arbeit werden vielfach Firmenbezeichnungen und Produktnamen genannt, die durch entsprechenden Urheberrecht geschützt und zu kennzeichnen sind. Diese Kennzeichnungen weisen teilweise Unterschiede in einzelnen Ländern auf, sodass an dieser Stelle lediglich die in Deutschland relevanten Kennzeichnungen und Hinweise erfolgen. Zur besseren Lesbarkeit wird im laufenden Text auf diese Kennzeichnungen verzichtet und stattdessen die nachfolgende Liste mit den rechtlich relevanten Hinweisen angefügt. Für den Fall von Einzelzitaten aus dieser Arbeit durch Dritte sind die rechtlich notwendigen Hinweise entsprechend einzufügen.

Hardware:

Notebook: DELL™ Precision 5540

Trademark-Hinweise sind aus DELL (2020) zu entnehmen.

Mixed Reality Gerät: Microsoft™ HoloLens™, 1. Generation

Trademark-Hinweise sind aus MICROSOFT (2020c) zu entnehmen.

GPS-Gerät: Blumax GPS-710

Software:

ArcGIS® Pro der Firma Esri®, Version 2.3

Trademark-Hinweise sind aus ESRI INC (2017) zu entnehmen.

CityEngine® der Firma Esri®, Version 2019.1

CityEngine® ist eine registrierte Trademark der Procedural AG™ und Lizenzvertrieben durch Esri®.

Trademark-Hinweise sind aus ESRI INC (2017) zu entnehmen.

FME® der Firma Safe Software™, Version 2019

Trademark-Hinweise sind aus SAFE SOFTWARE INC. (2020b) zu entnehmen.

Unity® der Firma Unity Technologies™, Version 2019.4

Trademark-Hinweise sind aus UNITY TECHNOLOGIES (2020) zu entnehmen.

Visual Studio™ 2019 der Firma Microsoft™

Trademark-Hinweise sind aus MICROSOFT (2020c) zu entnehmen.

Bei der Beschreibung der Schritte zur Softwareentwicklung werden vielfach Menüpunkte und Befehle der entsprechenden Softwareprodukte erwähnt. Zur Vermeidung von Missverständnissen sind diese Menüpunkte und Befehle jeweils durch *kursive Schrift* gekennzeichnet. An dieser Stelle wird darauf hingewiesen, dass es sich teilweise um geschützte Begriffe oder Namen handelt. Dasselbe Prinzip wird auch bei spezifischen Begriffen verwendet, die bei Erläuterungen der Literaturrecherche auftraten.