Christian Aluani, BSc

Utilizing augmented reality based guiding techniques in service design

Master's Thesis

to achieve the university degree of
Diplom-Ingenieur (DI) / Master Of Science (MSc)

submitted to
Graz University of Technology

Supervisor
Dr. Wolfgang Vorraber

Institute of Engineering and Business Informatics
Head: Univ.-Prof. Dipl-Ing. Dr.techn. Siegfried Vössner

Graz, April 2018

Affidavit

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present master's thesis.

_____   _____
           Date   Signature

Abstract

Augmented Reality (AR) is a up and coming technology which has the potential to change implemented work-flows in many different areas in a sustainable way. The goal of this thesis was to give a comprehensive overview of the technology and to conduct a case study implementing a new service that utilizes AR technology.

In the theoretical part a State-of-the-Art in AR is presented, covering different aspects like technological basics, available hardware and software and various use cases. Additionally, the theoretical background of service design and related tools are elaborated.

The practical part covers the implemented case study where a Proof-of-Concept for an AR-assisted remote consultation service was developed. This new service aims at providing inexperienced doctors or students visual guidance to correctly perform an ultrasound by using an Android smartphone and the Microsoft HoloLens.

Finally, the results of two evaluation sessions with selected domain experts are presented. Conclusions, limitations as well as potential for future work are derived from the evaluation process.

Contents

Contents

List of Figures

List of Figures

List of Tables

1 Introduction

Research and innovation are moving faster than ever and new technologies are surfacing nearly every single week. One of the currently most discussed innovations is Augmented Reality (AR). This technology enables people to seamlessly merge their surroundings with all kinds of virtual objects. Augmentation may be provided in many different forms and with many different gadgets, for example using head-mounted displays or hand-held mobile devices.

The endless amount of possibilities that rise together with AR spark sustainable interest among organizations in various areas. Industrial companies as well as public institutions have already recognized the great potentials of this exciting technology and invested in a broad range of research projects. The Institute of Engineering and Business Informatics (MBI) at the University of Technology Graz is one of the leading research groups in the field of AR with special focus to service design. The main driver for incorporating aspiring technologies like AR into new services is the potential to improve existing processes and greatly increase their effectiveness and efficiency in terms of quality, speed or costs.

In close cooperation with local hospitals in Graz, the use case of medical remote consultation was identified as highly promising for further AR research. Current consultation processes are dependent on the physical presence of a domain expert at the medical scene, for example in the examination room. Furthermore, there are basically only two real options for providing input and feedback to untrained staff:

- *Vocal communication.* Talking to the trainee and giving spoken commands. Although it is sufficient in some scenarios, it often lacks precision and is difficult to process for primarily visual types.
- *Physical intervention.* Personally intervening in the process, for example manually guiding the arm of the trainee. This method is very precise

1 Introduction
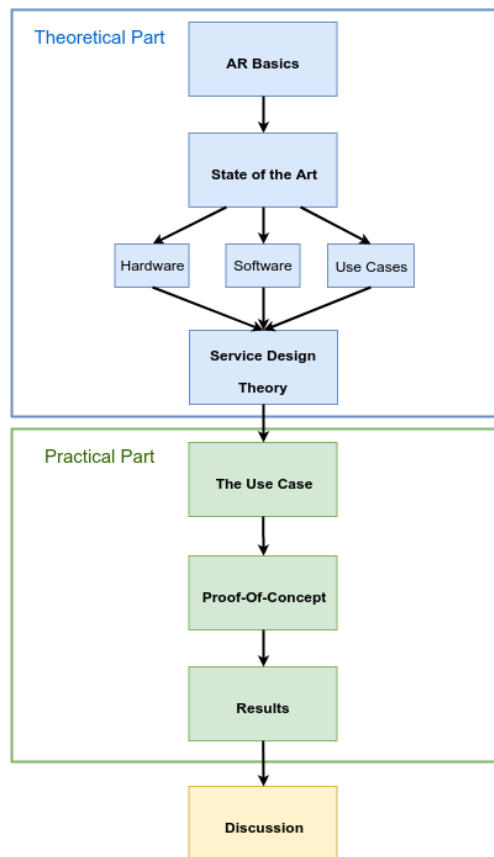
Figure 1: The Thesis Structure

and easy to understand, but takes away the personal autonomy in learning new techniques.

Augmented reality based guiding techniques could potentially generate advanced methods which improve the services and processes by a mile. For this reason, the main goal of this thesis is to identify potential improvement levers in the medical field by using AR based guiding techniques and to develop an enhanced service that integrates these technologies into the operations.

Figure 1 gives an overview over the basic structure of this thesis, which is topically split into a theoretical and a practical part. First of all, related research in AR is elaborated and a State-of-the-Art regarding hardware, soft-

ware and use cases are presented. In the second part important frameworks and tools for successful service design are introduced and their building blocks are described in detail.

The definition of a use case for the conducted case study and the practical application of the frameworks mark the start of the practical section. In the following chapters all elements of the implemented Proof-Of-Concept and the evaluation results are presented. Finally, all findings are discussed and potentials for future work are pointed out.

2 Related Work in Augmented Reality

This chapter covers related work and research in the area of Augmented Reality (AR). The goal is to evaluate the current State-of-the-Art in this promising technology. The first part of this chapter explains the main definitions and basic concepts of AR. Subsequently the different types of displays for AR are reviewed and categorized. Furthermore, the different possible techniques to position these displays in a scene are elaborated. Additionally, since it is a main part of the research of this thesis, some already explored input technologies for AR systems are presented. The last part of this section deals with current problems and challenges in the field of Augmented Reality.

Another major segment deals with the exploration of use cases already realized in research or economy. Applications in various segments including medicine, industry and entertainment are presented, with a special focus on Mobile Augmented Reality (MAR) as an emerging technology. Finally, currently available Software Platforms and SDKs are reviewed and the most common Head-Mounted Displays (HMDs) on the market are compared.

2.1 Definitions

The goal of this segment is to define the term **Augmented Reality (AR)** and position it in comparison to other concepts of its area. Due to emerging technologies and advances in computing and computer graphics the possibilities to enhance our "real world" with complementary dimensions are growing fast. Figure 2 depicts the categorization of different reality concepts from the real world to virtual worlds based on Schnabel et al., (2007). Scenery

Figure 2: Categories of reality and virtuality concepts, taken from Schnabel et al., (2007).

techniques which do not include any object from the real world and are completely generated by a computer are referred to as **Virtual Reality (VR)**. Opposite to that, **Mixed Reality** comprises all technologies that merge real and virtual worlds. Examples are Augmented Virtuality (displaying real objects in a virtual world) and Augmented Reality.

This thesis uses the definition for Augmented Reality originally provided by R. T. Azuma, (1997). The paper suggests that Augmented Reality "mixes the real world with virtual objects, so in contrast to Virtual Reality the user is not completely immersed into an artificial environment". (R. T. Azuma, 1997) In a perfect scenario the virtual objects are perfectly embedded into the surrounding to create the impression that they actually belong to the scenery.
For defining AR in a more formal way, R. T. Azuma, (1997) introduced three main characteristics of the concepts:

- AR combines real and virtual elements.

- AR is interactive and reactive in real-time.

- Within AR applications all objects and the user are registered in three-dimensional space.

One of the most important aspects is the requirement of a system to be interactive and reactive in real-time. The behavior of users is often unpredictable and presents a huge challenge for developers of AR systems as well as applications.

	Optical	**Video**	**Projective**
Mobile	Yes	Yes	Limited
Outdoor use	No	Yes	Limited
Resolution	Growing	Growing	Good
Field-of-view	Limited	Limited	Growing
Dynamic Refocus	No	No	Yes
Power economy	Limited	Bad	Limited
Cost	Expensive	Cheap	Moderate

Table 1: Different types of Augmented Reality displays, adapted from Van Krevelen and Poelman, (2010)

2.2 Augmented Reality Basics

This section addresses the technological aspects of Augmented Reality by introducing the different types of devices and displays that are currently State-of-the-Art. Other important topics include the various possibilities for positioning AR devices and the miscellaneous input methods for users.

2.2.1 Types of Displays

Since AR is a wide-spread field of technology there are different approaches for device design as well. The categorization provided in the following part is mainly adapted from R. Azuma et al., (2001). Table 1 gives a general overview over the diverse approaches for AR displays.

Optical Approach   One of the most common and advanced technologies are so called "optical" or "see-through" displays. They consist of a transparent screen merged with optical combiners (a combination of lenses and mirrors), which are partially reflective and partially transmissive to project the virtual objects into the scene. Mainly optical solutions are realized as binocular Head-Mounted Displays (HMDs) like the HoloLens (Microsoft Inc., 2016b), or as monocular smart glasses like the Google Glass. (Google

Inc., 2013)
The main advantages of this technology are: (R. Azuma et al., 2001)

- Realistic feeling and good depth information.

- Since the real world is processed with the human eye, there is no need for video stream synchronization.

- No limits due to resolution boundaries of the display.

- Safer than the video approach for HMDs, because in case of power loss the user is still able to see the surrounding environment.

- Cameras can never exactly imitate the viewing angle of the human eye. See-through displays are able to compensate this issue and provide no offset of the viewing field.

- An approach for optical devices developed by Holloway, (1995) allows the usage of the eye's center of rotation as viewpoint. Therefore the need for an additional eye-tracking camera can be neglected.

A noticeable drawback of see-through monitors is the low contrast and brightness of the projective elements, which makes an outdoor use difficult to realize.
Retinal Scanning Displays (RSDs) are a special form of optical displays, currently under research at Washington University and funded by the US military. A RSD projects high-resolution images directly onto the retina of the eye, allowing for a much higher contrast and brightness than combiner-based systems. Combined with the lightweight, energy efficient nature and the wide field-of-view, this technology overcomes the drawbacks of all current major displays and has the potential to establish future dominance on the market. Currently only monocular techniques are tested, but binocular solutions are proposed and still under development. (Schowengerdt et al., 2003)

Video Approach   Video-based devices are the wide-spread alternative on the market today, because they are relatively cheap and dominant in the Mobile AR segment. This category encompasses closed-view HMDs, which

are similar to VR devices, as well as static monitors or hand-held devices like smart phones and tablets. The main difference with the optical devices is that these systems enhance a video stream of the real world, captured by multiple cameras, with virtual elements.
This approach also has its own advantages:

- No "holographic problem". Graphical objects rendered into an optical display are always semi-transparent because of the light-blocking filters. With video displays virtual elements seem more realistic.

- Wider field-of-view because distortions in the optical field can be corrected virtually.

- Good brightness and contrast, easy to use with outdoor applications

- A video stream of the real world enables additional registration options (see section 2.2.4)

Obviously there are also some major drawbacks to consider when using video displays. Cameras can never imitate the viewing field of the human eye in a perfect way and the depth information is often not precise enough for activities that require fine motoric skills. Additionally the need for synchronization between multiple video data streams (real and virtual) generates a significant delay between the motions of a user and the actual displayed picture. Finally, the high amount of energy consumption limits the utilization significantly. (R. Azuma et al., 2001)

Projective Approach   In contrast to the optical and video solutions virtual objects are, as the name states, projected on specified areas with projective AR technologies. Special types of projectors and beamers are used to display graphical annotations or items on large surfaces like walls, floor, furniture or people. This augmented reality technique is mainly used in fixed installations, for example exhibitions in museums or sport events, since it requires a spatial setup (see section 2.2.2). To enable user interaction with projective AR installations, a combination of depth cameras and sensors like infrared or GPS are indispensable to capture the input.
Advantages of projective AR are:

- Most comfortable for a consumer. No special eye-wear or any other devices must be installed directly on a user.

- High resolution projections are possible due to the static nature of the system.

- Energy efficiency and computing power are not limiting factors because projective AR devices are stationary and can be connected to a high-performance PC.

- Well-suited if no complex interactions with users are required.

The benefits of projective AR that a spatial setup provides also leads to several negative effects. Use cases that require mobility or movement through different locations can practically not be realized with this approach, since it requires extensive hardware setup and special features in the environment. For this reason the significance of this technology is very limited and only suitable for very specific fields of application. (R. Azuma et al., 2001)

2.2.2 Categories of Display Positioning

Similar to the different display types for augmented reality experiences there are multiple ways to position those elements depending on the areas of application and selected technologies. The discussion below is also based on the previous work of R. Azuma et al., (2001). According to the authors an AR device can either be attached directly to the body of a person, carried as a mobile hand-held item or set up in a fixed position for example mounted on a bracket.

Head-Worn Head-mounted displays (HMDs) are closed-view video or optical see-through screens which are attached to the head of an user and directly placed in front of the eyes. This type of positioning has the benefit of free hands which allows a person to freely interact with its environment without any distraction. This is especially useful in industry or medical settings where operators are required to work with both hands simultaneously on the actual job. Since HMDs are mobile AR equipment,

power consumption and weight introduce potential problems. Workers that perform complex tasks over a longer period of time may be negatively influenced by heavy HMDs and are more likely to lose their focus. (R. Azuma et al., 2001)

Hand-held   Best known examples for hand-held AR items are smartphones and tablets when used in mobile AR (section 2.2.4). A user carries around the device and scans the environment with the built-in camera(s) and sensors like the gyroscope. The main advantage of hand-held devices is obviously the convenience for users. They do not need to buy additional hardware and are able to use AR applications at every location. Considerable drawbacks are that one hand is always blocked and the only possible choice of display is video. (R. Azuma et al., 2001)

Spatial   This category includes video screens, optical see-through displays, binocular videoscopes and projectors that are mounted on a stationary construction, for example a steel arm. Spatial positioning eliminates mobility of the AR device, but also allows the use of a hand-held device for example in medical settings where both hands of a person must be available. As previously mentioned in section 2.2.1, spatial positioning has limited application areas because every time the position of a device changes, the need for re-initialization of the whole application arises. (R. Azuma et al., 2001)

2.2.3 Input Technologies

The way people can interact with AR applications is manifold and depends on the device in use. A good overview of possible input technologies is provided in an article by Van Krevelen and Poelman, (2010) which is the basis for this section.

Augmented reality programs are able to receive and process **local** input as well as **remote** input. Local input comprises all interactions that are transmitted directly from the main user of the system. Remote input means every communication that is issued by a third party connected to the system

via an remote connection. The following local input methods are currently realized in most AR hardware:

Tangible UI and 3D Pointing   The well-known "Windows UI paradigm" for a two dimensional space, which is implemented on most PCs, is not suitable for AR systems. Augmented reality operates in 3D space and requires motions based on the principle called "six degrees of freedom". Tangible user interfaces (TUIs) fulfill this requirement and allow precise pointing in 3D environments. Examples for TUIs are paddles, pens or wands that can even be replaced with virtual objects by the AR device when they have markers or special features on them. Current research focuses on direct manipulation of virtual objects by a TUI, which is a computationally demanding problem. (Van Krevelen and Poelman, 2010)

Haptic UI   Simple TUIs only operate in a unidirectional manner and are not able to provide feedback to the user. Haptic user interfaces (HUIs) are designed to overcome this gap by enhancing TUIs with bidirectional communication. The two basic types of HUIs are force and tactile feedback. **Force feedback** interfaces simulate impact or resistance through a joystick or steering wheel. The PHANTOM haptic device (Massie and Salisbury, 1994) is a well-developed force feedback HUI and is mainly used in teleoperation. **Tactile feedback** in AR is realized with "smart gloves" that are able to inform a user about the surface of a currently touched virtual object or its temperature. (Van Krevelen and Poelman, 2010)

Visual UI and gesture recognition   To eliminate the need for physical aids and tools an visual user interface can be used to track human hands. This approach is based on compute graphic methods to analyze hand movements that are captured with a camera. Gesture recognition is a technique that is widely implemented and used as the main input method for projective AR devices to reduce the dependency on additional sensors. (Van Krevelen and Poelman, 2010)

Gaze tracking   For gaze input, tiny cameras and sensors are focused on the users eyes to keep track of the viewing angle. It allows the limitation of user interactions on the currently focused object. To guarantee exact functionality a calibration to user specific biometry data (pupil distance etc.) needs to be performed before every use. (Van Krevelen and Poelman, 2010)

Aural UI and Text Input   Aural user interfaces or speech recognition is a practical input concept to keep the hands of a person free for work tasks. Voice commands can be used for examples to zoom on objects or step through a set of instructions. Text input on the other hand is a complicated task to realize if used in 3D environments. The currently best solutions available are speech-to-text conversion in combination with pen-based handwriting recognition as a fall back option. (Van Krevelen and Poelman, 2010)

Remote input   is still an area of extensive research and therefore not adapted as a standard in publicly available AR devices. Already developed solutions are realized as annotations provided by an external observer, which are integrated into the video stream recorded from the AR device and sent over a network connection. Example applications are developed by Andersen et al., (2016), Volante et al., (2017) or Shenai et al., (2011) and are further elaborated in section 2.3.

2.2.4 Problems and Challenges

Despite the great advances in the field of AR in the last couple of years, there are still areas that present complex issues and need further research. R. T. Azuma, (1997) and Huang et al., (2013) outline the most difficult processes and limitations currently present for AR applications. The following segment summarizes their findings in a holistic manner.

Registration Problem   The correct alignment of virtual objects in the real environment remains the limiting factor in AR, especially for use cases that require precise operations. Considering a biopsy procedure where a "virtual tumor" is augmented on the patient, perfect registration is absolutely essential. If the virtual object is not exactly placed on the real equivalent and merged with the surroundings the biopsy will fail. This problem is especially serious with optical approaches, because there is no possibility for post-processing of the video data to correct inconsistencies. The reasons for registration errors may have static or dynamic sources.
Static errors occur even when the viewpoint is completely still. Examples include optical distortions, calibration errors of the tracking system, mechanical misalignments of the hardware or incorrect viewing parameters (field of view, tracker-to-eye position, interpupilary distance).
Dynamic errors result from movements of the user which may cause a lag between tracking data and the rendering of virtual objects. This means that graphical elements are sometimes not able to follow the users movements if they are too fast. Dynamic problems can be mitigated with stronger hardware, more computing ressources, prediction of future locations or enhanced digital synchronization of video streams (which is only possible using video-based devices). (R. T. Azuma, 1997)

Tracking   Most registration problems are directly related with the methods used to track the real world and the position of a user in it. **Sensor-based tracking** relies on the data from different sensors either built-in or connected to an augmented reality device. Table Y compares the most common sensor types to provide a principal overview. (Huang et al., 2013)

- *Inertial sensors*: Use data retrieved from acceleration. Example is the gyroscope that utilizes double integration (two measurements over a short period of time) to obtain position and angle information. High accuracy can only be achieved with frequent calibration.

- *Magnetic sensors*: Tracking based on the magnetic field of the earth. May be exposed to external inferences.

- *Electromagnetic sensors*: Position tracking based on electronic signals, for example GPS, bluetooth or Wi-Fi. Has limited accuracy and problems

Type	Sensors	Coverage	Accuracy
inertial	accelerometer	anywhere	0.01m
	gyroscope	almost anywhere	0.2 deg.
magnetic	magnetometer	almost anywhere	0.5 deg.
electro-magnetic	GPS	almost anywhere	cca. 15m
	Wi-Fi	cca. 100m	2.5m
	Bluetooth	cca. 10m	1 - 10m
	Infrared	cca. 6m	cca. 0.1m
ultrasonic	ultrasonic	10m	cca. 0.01m

Table 2: Comparison of sensor types, adapted from (Huang et al., 2013)

with shielding indoors or in covered areas.

- *Ultrasonic sensors*: Provide high accuracy based on ultra-sound signals, but not widely adapted and rather expensive.

Vision-based tracking on the other hand uses video data captured by different cameras and applies various processing tools: (Huang et al., 2013)

- *Marker-based method*: Fiducials (markers) are attached to objects and act as artifical features. They are easily trackable because of their predefined properties like color, size or shape. Only applicable in indoor applications or in small areas.

- *Nature feature method*: Computer vision algorithms are used to track point and region features in image sequences. To avoid drift and lag problems, so called "frame-by-frame" techniques are applied. Algorithms include the comparison with reference images, machine learning techniques or principal component analysis. Limited to environments with noticeable and outstanding objects that are easy to identify.

Vision-based approaches often have a high computational overhead and are prone to image distortions, whereas sensor-based methods often lack in accuracy. To overcome the individual shortcomings **hybrid methods** are developed which for example combine GPS and gyroscope data with natural feature tracking. Microsoft developed a hybrid approach for the HoloLens

(Microsoft Inc., 2016b) with special cameras and depth sensors to generate a digitalized image of the complete environment, which facilitates object tracking and registration. (Huang et al., 2013)

Interactiveness A main characteristic of augmented reality is interactiveness (as stated in section 2.1). AR systems work in real-time and must be able to react to user actions and interactions in a very short period of time. Special effects, for examples in blockbuster movies or TV shows, are easier to realize because every scene and movement of the people involved in the production is carefully planned and executed. For real-life applications the unpredictable nature of human behavior presents high obstacles for AR hardware and software. (R. Azuma et al., 2001)

UI Limitations and Social Acceptance To explore wider fields of application for AR it is necessary to develop more flexible ways for displaying information to a user and advanced opportunities to query data. But even if a system would be perfectly developed and affordable for mass-use at some point in the future, there are some concerns that possibly prevent people from integrating the technology into their daily lives. The biggest reservations that many users have about AR are comfort, fashion and privacy concerns. Security is a highly sensible issue, because AR devices collect lots of private data like video streams for object recognition or position information and transmitting it over a network connection, which increases the risk of data-theft. (R. Azuma et al., 2001)

Special Case: Mobile Augmented Reality Recent advances in modern technology make it possible to use mass-products like smartphones or tablet computers as devices for Mobile Augmented Reality (MAR). Factors that enable MAR applications without the need for extra hardware include built-in high resolution cameras and sensors as well as new generations of computational resources. Furthermore, the possibilities of mobile cloud computing, network caching and device-to-device communication allow much more powerful AR experiences on smartphones.
Since mobile devices are restricted in many areas such as storage, power and

networking due to platform specific boundaries, additional problems arise which are discussed briefly in the following part: (Huang et al., 2013)

- **Limited Storage and Network Capacity:** Mobile devices have restricted storage space available which is often too small for complex applications with many different virtual objects. Cloud storages combined with remote computing may overcome this issue, but often require high network bandwidth and produces significant overhead. A slow internet connection makes the whole application useless.

- **Energy Efficiency vs. Performance:** Processors in mobile devices are designed for lower power consumption and interval usage. MAR applications on the other hand require long-term communication between camera, sensors, network and computing resources and are therefore draining the batteries quickly.

- **Low-level MAR libraries:** Tracking unknown environments is even harder to do on mobile environments. Native libraries for object and position tracking or computation offloading would greatly facilitate the development of new MAR programs.

- **Lack of "killer apps":** Most MAR experiences developed so far are demonstrations for research purposes or create only a short hype. Sustainable real-world applications to assist everyday activities need to be built for increased popularity.

2.3 Applications and Use Cases

After the exploration and discussion of the basic properties of AR in the previous segments, the following part focuses on different application areas for AR systems. Concrete examples and work of many companies and research groups over the last decade are presented and discussed. Figure 3 shows the categorization of use cases used in this thesis. For reasons of simplification, some use cases which may be classified in multiple categories are assigned to the most suitable one. The individual research results are presented in random order and different levels of detail inside their respective class,
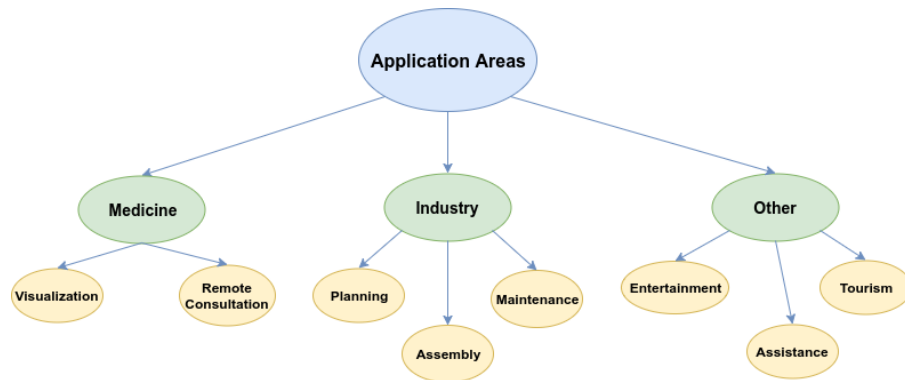
Figure 3: Fields of application for AR systems.

which is subject to the personal impression of the author and does not reflect scientific relevance in any way.

2.3.1 Medicine

Medical applications are one of the most frequent subjects of AR research. Applications can potentially facilitate work routines of medical staff and emergency responders. This leads to more effectiveness of processes, improved quality of procedures and can ultimately save lives. Remote consultation is an interesting field of study, since the lack of qualified personnel is a critical factor especially in developing countries.

Visualization   Augmented Reality can be utilized as visualization and training aid for various medical procedures. Additional vital data that is collected from different non-invasive techniques like magnetic resonance imaging (MRI), computed tomography scans (CT) or ultrasound, may be rendered in real-time on to the patient. This introduces a kind of X-Ray vision to a doctor for example during surgery or training processes. Another field of application is using AR as a guidance tool for precision tasks, for example by *displaying drill holes* for brain surgery or instructions for a *tumor biopsy*. A **UNC Chapel Hill research group** (Ohbuchi, Bajura, and Fuchs, 1998) adopted one of these visualization ideas early on. The authors conducted

a case study to test a "3D stethoscope" which used simple AR techniques to display a fetus inside a pregnant woman's womb. Ultrasound data was used to simulate movements and vital signs.

The exploratory study performed by **Vorraber, Voessner, et al., (2014)** is a recent example for AR-based visualization techniques. They utilized Google Glass to display vital signs (heart rate, temperature, oxygen rate) next to the patient in real-time during a surgical procedure. This allowed the surgeon to concentrate solely on the procedure and eliminated the need to move the field of view away from the patient to check the backup monitors.

Remote Consultation   In medical training personal guidance and instructions from a domain expert to a novice doctor are one of the most important and effective ways for education. Since time and personnel resources are limiting factors for the physical presence of an instructor, remote consultation is gaining popularity very fast. Another gap that virtual guidance can overcome is the small number of experts for newly developed and extensively complex procedures, especially in developing countries. Experienced medical staff may assist untrained surgeons in difficult processes without the need to travel for example to war zones. *Augmented Reality* techniques could help to elevate remote consultation solutions to the next level. Instructors can demonstrate techniques directly on the patient and learners do not need to shift their focus away to another screen.

The following list represents interesting and promising research projects in this field of study:

- **Augmented Telemedicine Platform: (Anton et al., 2017)** The authors developed a telemedicine framework to connect a physician at a hospital with a first responder that is with the patient.
  The *AR client* is set up on the patient side, preferably in an ambulance vehicle to facilitate the hardware setup. This module collects texture, body surface and depth information of the scenery (with a Microsoft Kinect 3D camera) as well as vital signs from the patient. This data is transmitted via "named pipes", precisely a RGB video pipe, a depth information pipe and a data pipe, to the module at the hospital. It also features multiple *projectors* which allow OpenGL based rendering of incoming information.

The **VR client** is a 3D touch display set up at the hospital that renders the different data streams received from the AR client by parsing it with an OpenGL library. An experienced physician is able to interact with the video, for example zoom onto different body areas for further examination, and can add annotations to the video, which are transferred back and rendered onto the patient.

To allow secure communication a **communication module** was created to act as a middleman between the AR and VR client. It uses a WebRTC framework, which allows automatically encrypted peer-to-peer communication, and synchronizes the video stream and depth data.

Results of this study show that if an emergency doctor is not available right away, this remote system could be a valuable addition to ambulances. The augmented instructions were far more accurate than over the phone. The most noticeable drawback was a slight video and audio delay, which could be lowered by using CPU overuse detection.

- **Virtual Interactive Presence and Augmented Reality (VIPAR): (Shenai et al., 2011)** VIPAR is a system developed for telepresence in surgical settings by combining local real world images and remote assistance to a composite field of interest. Both stations, local and remote, are equipped with a *binocular videoscope* which consists of two HD color video cameras to record the local viewing field, a blue light emitting diode (LED) array for background subtraction and high-definition binocular viewers to display the common virtual field. Figure 4 shows a schematic representation of the whole VIPAR system.

  The authors tested their system in an experimental setup where a novice neurosurgeon performed a procedure on a human cadaver head and neck. The local videoscope was mounted on a steel arm over the "patient". To correctly align the remote input from the assisting expert in the combined virtual field, a marker-base method (section 2.2.4) was used to facilitate registration and tracking.

  Results: The local surgeon felt that the instructions from the remote physician were helpful and in reverse the remote surgeon felt confident in his ability to guide the trainee precisely through the VIPAR system. Even though the camera resolution was quite good, very small anatomical structures could not be visualized correctly. A MRI image
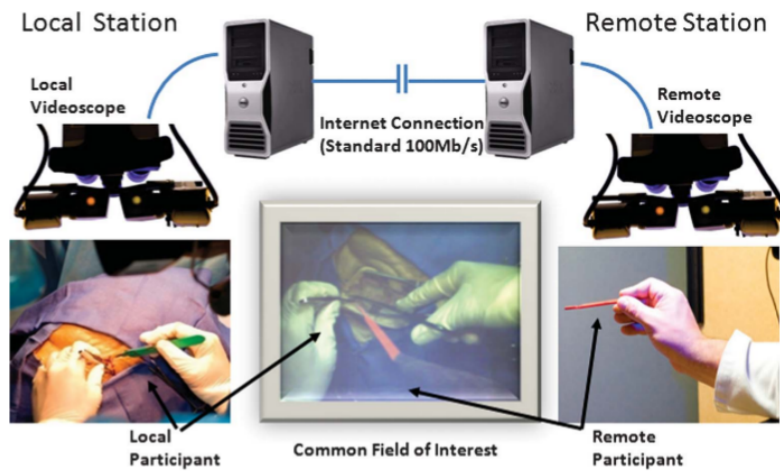
Figure 4: Schematic representation of the VIPAR system, taken from Shenai et al., (2011).

was used to overcome the resulting "blind spots". Visual delays, which are caused by the frame rate of the cameras, were detectable when rapid movements occurred but did not influence the procedure in any negative way. Another observation that could be taken from the experiment was the importance of *audio instructions* as addition to the visual guidance, since they add indispensable context. The expert on the remote station experienced mental fatigue after some time because VIPAR could not provide tactile feedback. A considerable problem that emerged during the procedure was the insufficient depth mapping, which means that the "remote hands" did overlay the local surgeons hands at some occasions.

- **Real-Time Augmented Reality and Wearable Computing Devices: (Ponce et al., 2014)** This study combined *smart glasses* with the *VIPAR* system, which has been discussed in the previous paragraph, in a real surgical setting. The different nature of the experiment implies a slightly changed hardware setup than the one used by Shenai et al., (2011). The local surgeon used a Google Glass to display the merged visual field instead of a binocular videoscope. On the remote side the videoscope was replaced by a standard display and a 3D depth camera to record the experts hands.

Results: The surgery was successful, but lasted at least 45 minutes longer than a standard procedure. Technical difficulties with the Google Glass led to a sustainable delay. Google Glass did not support secure network access in general and was dependent on Hangouts for aural communication, which did not meet the necessary confidentiality properties regarding the procedure. Therefore the direct audio transmission was conducted with a Skype conference call. The mobility advantage provided by smart glasses was greatly reduced by the low battery life and the image quality on the local surgeon's end. Fine details of the surgical field were not visible well enough and led the trainee to use contact lenses to improve the sight. Another issue that is closely linked to the image quality is the divergent line of sight. The viewing angle of Google Glass forced the surgeon to look down in an unnatural fashion to provide good pictures to the remote expert. As opposed to that, to see the virtual field in the heads-up display the gaze needed to be in an upward direction, away from the actual surgical field.

As a general conclusion the authors were confident that the VIPAR system does not work well with smart glasses as technical aid.

- **VIPAR in Global Surgical Education: (Davis et al., 2016)** The adaption of telesurgical systems, like robots that can be controlled remotely, in developing countries is impeded due to cost, available network resources and the need for skilled technical personnel at the site. Advanced telepresence system can have a huge impact in global surgical education, as demonstrated in this paper by the use of a sophisticated iteration of the VIPAR system as education tool between doctors in Vietnam and the US.

  The hardware of the advanced VIPAR system developed by the authors consisted of two iPad Air 2 (Apple Inc., 2017) with a wireless 3G network connection. On the local site the tablet was mounted on a steel arm in front of the surgeon and the foward-facing camera provided 1080p HD video data. For most remote instructions the hands of the expert were filmed on a white background and a complementary pen could be used to draw 2D instructions on the tablet screen.

  Results: All test runs were executed successfully in terms of stability and video quality. No noticeable delays were detected during the
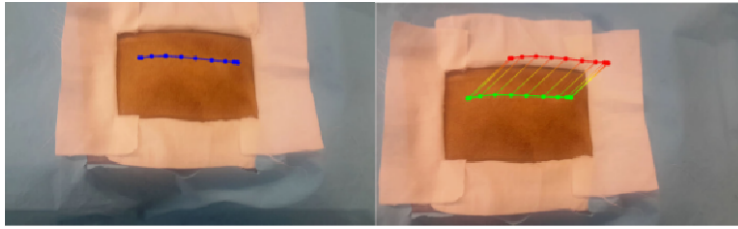
Figure 5: Incision line annotation anchoring process with descriptor matching, taken from Shenai et al., (2011).

procedures, which was confirmed with off-line video analysis that revealed an average delay of 275ms over 14,000 km. Additionally, spatial accuracy was sufficient and differences were undetectable by the surgeons. With the modernized hardware setup, VIPAR has high potential as additional surgical training tool in global environments, because of its relatively low acquisition costs and fast setup at any site. On the other hand the use of iPads caused problems in depth representation and lack of tangible feedback. Unstable network connection in crisis regions or developing countries also poses a noteworthy threat to the implementation.

- **Virtual annotations of the surgical field: (Andersen et al., 2016)** This paper describes another approach to realize remote surgical assistance. An AR transparent display was created, where a tablet was precisely aligned with the viewing field of the local surgeon to create the illusion of a see-through display perfectly merged with the surroundings. This setup helped to maintain the eye-hand coordination that is required to perform exact surgical procedures. Another innovation proposed by the authors are *"anchored annotations"*, a concept that automatically keeps the virtual objects created by the remote surgeon in place when the viewing field changes. When the trainee moves the display all annotations are automatically positioned, scaled and rotated with respect to their position in the initial frame.
  The anchoring algorithm: Figure 5 demonstrates the anchoring process on a virtual incision line.

  1. Using the *mentor system* the expert places annotation objects, which are either transparent images of surgical instruments, lines,

shapes or text, onto a frozen frame. The corresponding data objects (type of annotation and point-based positioning information) are transmitted to the trainee tablet together with the current frame.

2. The *trainee system* performs the image processing and anchoring calculations every time a new object arrives or the viewing field changes. Because the framework is marker-less, the first step of the process is feature detection. An OpenCV implementation of the ORB algorithm (Rublee et al., 2011) is used to determine descriptors for each feature point in the reference scene.

3. Each future frame involves the extraction process from step 2 as preparation. For every point of the image the current descriptors are compared against the reference descriptors by using an all-pair brute-force method, where the pair with the smallest distance is selected. Finally the homography between the matched feature descriptors, which represents the mapping of a point in the reference frame to the point in the current frame, is calculated based on the RANSAC algorithm (Fischler and Bolles, 1981), and the annotation is transformed and re-rendered accordingly.

The anchoring process was tested in a live surgical setting and performed well for repositioning by an average error rate of around 10% of the total frames. Occlusion of the scene, which means that new real objects are added, led to problems if not enough features could be detected. Finally deformation of the surgical field, for example when opening a wound with a retractor, broke the algorithm and produced high error rates due to the changing set of natural features. The computing power of the tablet was sufficient for the task, limiting latency errors to very fast movements only. Depth information and tactile issues remain the same as with the previously discussed studies.

2.3.2 Industry

Augmented Reality applications are not only tested in medical settings, but there is also active research in various industrial sectors. Workers in those areas are often confronted with complex tasks which consist of a
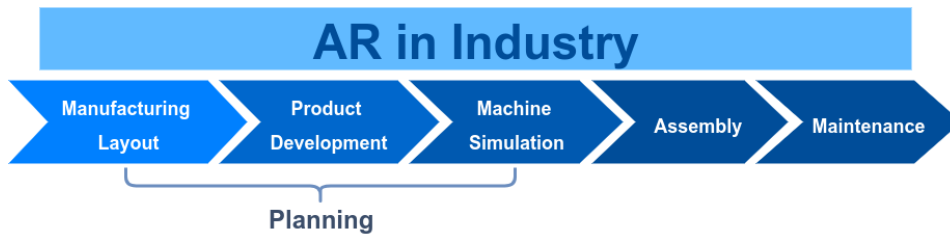
Figure 6: AR in the industrial value chain.

substantial number of steps and instructions. These circumstances lead to long familiarization and initial training routines which not only impair the overall productivity, but also occupy many skilled workers as trainers. Figure 6 demonstrates the versatility of AR throughout an industrial "value chain". It has the potential to facilitate many of these industrial tasks, shorten the learning curve and extend the capabilities of the involved technicians. The following section presents promising AR applications tested mainly in production industry scenarios.

Planning   For simplification this thesis couples all industrial tasks which involve some kind of planning activities together in one topic, as outlined in figure 6. One of the main advantages of AR in planning tasks is the possibility to test designs and models in real-time directly applied to their subsequent environment without the need for physical production up front. This approach provides great flexibility, immediate feedback and saves time and ultimately costs.(Ong, Yuan, and Nee, 2008)

- **Manufacturing layout:** Process development is getting faster and improvement cycles become even shorter, which rises the need for efficient tools to plan sophisticated manufacturing sites. AR provides new methods to improve the provisioning of industry systems by making direct interaction with real sites and technologies available. The necessity for physically produced models or whole virtual worlds for VR is no longer present. *Dangelmaier et al., (2005)* proposed an AR application to assist users in system modeling with virtual planning objects. They also implemented improved tools for model validation and optimization of the results.

- **Product development:** After a complete review process, many products are modified and adjusted in advanced design or even production phases, which often leads to exploding development costs. *Mixed prototyping*, an AR-based simulation strategy for design review and validation, can be used to project virtual prototypes into real environments and perform detailed assessments of functional, optical or ergonomic criteria. This is especially helpful for product engineers to test new ideas without investing enormous amounts of money into physical prototypes.

  - *Persiani, Liverani, and De Crescenzio, (2001)* researched AR prototyping methods in cockpit construction. They included pilots into the development process by directly applying their feedback and experiences to the protoype.

  - *Friedrich, Jahn, and Schmidt, (2002)* presented "ARVIKA", a mixed prototyping tool used for extensive usability testing in the automotive industry.

  - *Frund et al., (2004)* proposed a collaborative multi-user AR-system for prototype design, which enabled experts that are physically separated from each other to work together on a common virtual model for the first time.

  - *Lee and Park, (2005)* extended mixed prototyping concepts with a tangible UI. "Augmented foam" was used to create prototypes. Special sensors included in the interface provided advanced interaction methods and tangible feedback to users.

- **Telerobotics:** Bley et al., (2004) and Chong et al., (2006) developed a AR systems to practice control decisions and programming for a robot. The provided instructions were performed on a virtual equivalent placed in the real scene. This helped to prevent errors during program execution and to find more efficient solutions. AR could also be used to provide additional visual information to robot-assisted procedures in poor visual conditions.

- **Machine simulation:** J. Zhang, Ong, and Nee, (2006) demonstrated the use of augmented materials to test industrial cutting machines.
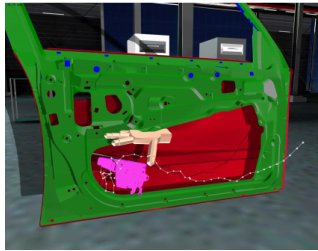
Figure 7: Doorlock assembly procedure, taken from Reiners et al., (1998).

The outcome of the cutting process performed by the real machine was simulated on a previously unknown virtual workpiece. The system avoided possible tool collisions and dangerous trial runs with uncertain outcome.

Assembly   Assembly can be described as the action of putting together complex machinery from interchangeable standardized parts. Classical assembly instructions are often textual, as paper or electronic manuals, which force the worker to constantly switch the field of view between the working area and the manual. AR can be used to display step-by-step instructions projected near or directly onto the objects which need to be assembled. Most of the examples presented below rely on markers to recognize assembly objects. Natural feature detection to coordinate projection of correct information onto the corresponding object remains a challenge. Rich sensor models which presume a certain understanding of the environments could evolve as the method of choice.

- **Boeing Airplane Assembly: (Curtis et al., 1999)** Engineers from Boeing developed one of the first AR assembly applications. It was able to display instructions and diagrams in front of the workers to provide guidance for the assembling of aircraft parts and cables.

- **Door Lock Assembly: (Reiners et al., 1998)** Reiners et al., (1998) proposed another early AR-based training program to teach users to assemble car locks into doors by rendering direction and positioning information onto the parts as depicted in figure 7

- **Furniture Assembling: (Zauner et al., 2003)** Augmented reality is
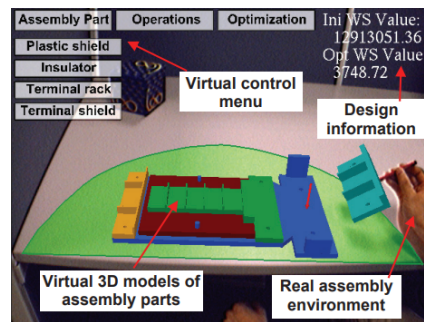
Figure 8: AR assembly planning environment, taken from Ong, Pang, and Nee, (2007).

used by the authors to visualize a step-by-step manual for furniture assembly. The system utilized ARToolkit (section 2.4) for object recognition and tracking.

- **Virtual assembly panel: (Yuan, Ong, and Nee, 2008)** Yuan, Ong, and Nee, (2008) implemented an interactive virtual panel to step through predefined assembly instructions without the need for markers or sensors. The proposed solution only works in well-known environments.

- **Assembly process planning: (Reinhart and Patron, 2003) and (Ong, Pang, and Nee, 2007)** Both papers describe AR systems for assembly planning. They were designed for engineers to assist in the process of designing and planning a product assembly and its assembly sequence. Figure 8 shows an exemplary user interface for AR assembly planning.

- **AR sensing device for assembly workstations: (Dalle Mura, Dini, and Failli, 2016)** In this very recent study the authors outlined a system to detect human errors in the assembly process. A force sensor was installed under a workbench to alert the AR device in case of faulty operations, which then advises the user by giving visual feedback.

- **Mobile Augmented Reality Technical Assistance (MARTA): (Volkswagen AG, 2017)** Multi-national car manufacturers like Volkswagen have already recognized the potential of AR for assembly processes as well. They introduced the tablet-based system MARTA in 2015, which labels car parts clearly and shows precise work instructions directly on the car itself, as can be seen in figure 9.

Figure 9: Mobile Augmented Reality Technical Assistance (MARTA), taken from Volkswagen AG, (2017).

Maintenance Similar to assembly tasks, AR can be used to display relevant information and step-by-step instructions directly onto complex machinery. This could potentially speed up maintenance and repair processes since it eliminates the need to consult another expert or look at a paper manual. Currently only prototypes were developed because there are still issues with tracking and sensing in dynamic surroundings.

- **Aircraft maintenance:** Neumann and Majoros, (1998) presented a program for aircraft maintenance and repair, which was able to highlight the location of fasteners and attachment hardware that must be removed in the display.

- **Food quality control:** A large food producer commissioned Najjar, Thompson, and Ockerman, (1997) to improve the quality control process. They proposed a system using a head-mounted display which showed the inspectors suitable quality criteria based on the currently visible type of food.

- **Further examples:**

  - *Underground pipe inspection* (Lawson and Pretlove, 1998)
  - *Device diagnostics* (Sundareswaran et al., 1999)
  - *Railway maintenance* (Azpiazu et al., 2011)

Figure 10: MAR for archeological sites, taken from Zoellner et al., (2009).

2.3.3 Other

Not only does AR spark interest in many business fields, as described in the previous sections, but is also gaining more and more popularity for daily life applications. Mobile AR (presented in section 2.2.4) introduces mixed reality applications to smartphones and is therefore able to reach the mass market for private end-users. This part describes research projects as well as commercially developed MAR applications for private consumers.

Tourism and Navigation: Traditional navigation software sometimes works not good enough to provide ideal orientation for a user. Similarly tourists who visit places of interest, for example cultural heritage sites or museums, may have trouble imagining what an attraction looked like or felt like in the past. MAR has great upside for visualizing additional information into real world settings:

- *Höllerer et al., (2001)* proposed a navigation system for the Columbia University campus, which overlays buildings and objects with additional information and directions.

- *Narzt et al., (2006)* introduced an augmented navigation system designed for car displays.

- *Dahne and Karigiannis, (2002)* presented the "Archeoguide" app to show tourists computer models of ancient historic buildings at their original sites in Olympia (Greece) (see figure 10).

- *Föckler et al., (2005)* implemented an enhanced museum experience. Guidance and extra information are provided when phone is focused

on an exhibit.

Entertainment and Advertisement: Entertainment applications like games or social media extensions are very popular choices for applied AR. In contrast to most other fields discussed in this thesis, there not only research prototypes are realized, but products are already available, which proved to be commercially successful.

- *Renevier et al., (2005)* developed one of the first MAR entertainment applications, a connected multiplayer *3D chess game* which was augmented onto a real table.

- *Morrison et al., (2009)* created a AR system which allows the creation of *virtual objects upon a paper map* when markers are focused with the camera.

- *Olsson et al., (2013)* specialized their research on advertisement and *enhanced shopping experiences*. The MAR application showed for example user ratings from online sources or allergy information to the scanned product and was tested in a shopping-center in Sweden.

- *Q. Zhang et al., (2017)* conducted a very recent study which tested a MAR program for *virtual prescription eyeglass try-on*. An image sequence of a user's face is scanned and the selected eyeglass frame is correctly placed and automatically re-aligned. Finally the lenses are merged into the frame, based on the individual medical prescription data. The system is able to change the appearance of the eyes, for example shrinking them when lenses for nearsightedness are selected, as well as rendering shadows and reflections for a realistic feeling.

- *Sekhavat, (2017)* introduced the first *privacy preserving clothes try-on* MAR application. It was designed as an alternative to expensive and privacy-impairing virtual try-on rooms in some stores. Based on universal human body models the program calculates the correct appearance with the help of images and measures of a person. Limiting factor are the clothes, because every garment needs to be virtualized and equipped with according markers in every store.

- *Snapchat (Snap Inc., 2017) and Instagram (Instagram Inc., 2017)* intro-

Figure 11: Snapchat Face-Filters. "snapchat-xmen-filter-mustache-filter" by Jamie is licensed under CC BY 2.0

duced *face filters* to their respective social media apps. This AR extensions allow users to place different animated objects on their face while filming themselves with their phone or tablet. The feature rapidly became one of the most used functionalities on both apps and the amount of available virtual objects is nearly exploding. This development indicates the massive potential of mixed reality for entertainment purposes. Figure 11 shows example face filters which can be used in Snapchat.

- *Pokémon GO (The Pokémon Company International Inc., 2016)* is another highly successful example of commercial MAR applications. It is a location-based game where GPS and network access is used to project virtual monsters (Pokémons) at specified real-world locations. Main goal for a user is to visit the indicated places and catch as many monsters as possible. Multi-player support is also realized by simulated AR battles between monsters of two different players. Pokémon GO triggered a huge hype in 2016, which brought still existing problems of MAR to the surface. Players engaged with the game had *reduced awareness* for real world situations, for example causing traffic accidents or committing home invasion because some monsters were located on private property.

Education In the near future many institutions could use AR for educational purposes to increase the interest of students for various topics. *Freitas and Campos, (2008)* used 3D models of cars or airplanes projected into a

class room to *demonstrate transportation basics*. *Andujar, Mejias, and Marquez, (2011)* tested an *AR remote lab* in an educational facility to save costs and enable more students to participate in laboratory courses. AR was used to replace switches of a physical device board with virtual ones. Students could interact with the virtual controls and simulate their experiments without occupying the actual machine. *Ibanez, Castro, and Kloos, (2017)* conducted a study to examine the influence of projective AR systems on the motivation of students. A simulation of a charged particle in a constant magnetic field to demonstrate the *physical principle of electromagnetism* was augmented into the room. Results showed that the interactive system, specifically the possibility to view the particle movements from any angle, significantly raised the interest of the students.

2.4 Available Hardware and Software

After reviewing the AR basics and fields of applications, the last section of this chapter is devoted to software toolkits and hardware solutions currently available on the market.

2.4.1 Head-Mounted Displays

As already presented in section 2.2, there are many different types of AR hardware. Since smartphones, tablets and projectors are already well-known and can be classified as "secondary" hardware for AR, this thesis only discusses head-mounted displays. Furthermore, the selection is limited to HMDs which are commercially produced for the mass-market and are officially announced by the manufacturer (as of the 10th November 2017). Table 3 shows a comparison between selected HMDs. The currently available HMDs can be distinguished into *full-AR systems and smart-glasses*. Full AR-systems, for example the HoloLens, are rather expensive and heavy to wear, but provide strong computing performance, large storage space, good resolution and advanced object tracking. In addition the user-interfaces are sophisticated and offer a wide selection of input methods, including gaze tracking.

Name	Price	Pro	Con
HoloLens (Microsoft Inc., 2016b)	ca. 3,000€	Binocular, Depth-Mapping, Unity (Java), Storage, Connectivity	Medium FOV, Comfort (Heavy)
ODG R7 (Osterhout Design Group Inc., 2016)	ca. 2,500€	Smart-Glasses (lightweight), Android-based UI, Storage, Connectivity	Resolution, Energy
Vuzix M300 (Vuzix Inc., 2016)	ca. 1,700€	Smart-Glasses, Sensors, Android 6.0	Monocular, Small FOV, SDK Support
Meta 2 (Metavision Inc., 2017)	ca. 1,400€	Binocular, Wide FOV (90deg), High Resolution, Computing Power	**NOT** wireless, Comfort
Moverio Pro (Epson Inc., 2015)	ca. 700€	Cheap, Binocular Smart-Glasses, Android 4.0	Low Storage, Weak CPU, Resolution, SDK support
Recon Jet Pro (Recon Instruments Inc., 2017)	ca. 550€	Cheap, Smart-Glasses, Brightness	Monocular, SDK Support, Low Storage, Resolution, Battery
GlassUp F4 (GlassUp, 2017)	TBA (Demo)	Design, Battery life, Sensors, Linux OS	Monocular, Low Storage, Resolution
Vravana Totem (Vravana Inc., 2017)	TBA (Demo)	120deg FOV, Strong Cameras, Open SDK Support	**NOT** wireless, Video Display, Comfort

Table 3: Comparison of head-mounted displays

Smart-glasses on the other hand are constructed like regular glasses and therefore more comfortable and lightweight. Monocular solutions are more affordable, while binocular systems are not much cheaper than full AR-systems. General advantages are the high contrast and brightness, which facilitates outdoor-use, and the Android-based UIs that allow inserts from standard smartphone apps. Substantial drawbacks of smart-glasses are the small field-of-view, limited display resolution and deficient depth sensing. Storage and computing capacities as well as battery life are also restricted. Furthermore, advanced input methods like gaze tracking or voice input are not well-supported.

The two announced models labeled as "Demo" in Table 3 promise a greater FOV, advanced computing power and camera-sensor systems. Resolution, storage and battery life (the Vravana Totem is completely wired) still remain issues. **RSDs** are still in prototype stage and therefore very expensive (up to 9,000€) to work with.

2.4.2 Software Toolkits

The wide range of hardware and UIs makes the development of AR applications a complex and time consuming task. Working with object tracking, positioning algorithms and computer vision techniques poses additional challenges to developers. The main goal of software toolkits and SDKs is to provide standardized methods and general starting points for as many different applications as possible. Some manufacturers of HMDs ship their own libraries for a specific product and do not provide free access to the development tools. For this reason, many publicly available software toolkits focus on MAR applications because they are currently easier to deploy. This thesis covers the most important commercial SDKs which are (partially) free for public use and some open-source projects from research groups:

- **AR Toolkit: (Kato and Billinghurst, 1999)** First marker-based platform that provides specific fiducials and targets on which objects can be placed. Served as foundation for next generation SDKs and was used for many early AR prototypes in research.

- **DART: (MacIntyre et al., 2003)** DART is a low-level support library that was designed to manage and connect different sensors, trackers and cameras.

- **Studierstube (Schmalstieg et al., 2002) and Open-Tracker (Reitmayr and Schmalstieg, 2001):** The authors developed a hardware- and platform-independent AR system at Vienna University of Technology. In combination with AR Toolkit and Open-Tracker, a tool to manage tracking devices, it serves as widely-used platform for AR prototypes.

- **DWARF: (Bauer et al., 2001)** DWARF is a peer-to-peer component based service management framework. It encapsulates different services (for example optical tracking) as standalone modules which interact as cooperative elements on any remote server.

- **Unity3D: (Unity Technologies SF, 2017)** Unity is the most successful game engine and scripting platform for 3D applications. It supports a wide number of operating systems and rendering engines (for example Direct3D, OpenGL, WebGL, etc.) and provides extensive scripting in C#. Since AR applications operate mainly in 3D space, Unity serves as standard tool for creating mixed-reality experiences. Unity is partially free of charge and provides detailed documentation and tutorials.

- **Vuforia AR Platform: (PTC Inc., 2017)** The Vuforia SDK is a well-developed commercial AR platform for mobile devices and selected HMDs, for example HoloLens or ODG R-7. Since the latest release it is natively included with Unity as standard package. The Vuforia engine serves as a static client-side library that provides applications with additional AR functionalities. Main benefit is the extensive object scanning and custom image recognition technology, which allows user-defined items to serve as "anchor" for virtual information, for example projecting additional content onto a magazine page.

- **Google Tango: (Alphabet Inc., 2017)** Tango is a JAVA-based API developed by Google for Android phones. It combines motion tracking, area learning and depth perception to create AR content for smartphone

apps.

- **Microsoft AR Toolkits: (Microsoft Inc., 2017b)** Microsoft offers different open-source auxiliary tools to connect and extend basic AR systems. An interesting project is the *"RoomAliveToolkit"*, which is a collection of Unity scripts to enable dynamic projection mapping experiences for stationary installations.

- **TaleBlazer: (Medlock-Walton, 2012)** TaleBlazer is an open-source AR game editor from the Massachusetts Institute of Technology (MIT). It is designed as a simple drag-and-drop editor to create simple location-based games for Android and iOS devices.

3 Related Work in Service Design

To develop and deploy successful AR applications, the technical side discussed in chapter 2 is only one of many factors. Careful analysis and design of a system ensures high quality output and efficient implementation processes. This chapter presents the theoretical backgrounds of some methods and tools, which were developed to facilitate the planning process of a service and are applied in the case study (chapter 4) of the thesis.

3.1 Service Design Theory

In literature there exists no common definition or clearly specified language for Service Design. Generally speaking, Service Design can be seen as an interdisciplinary approach that is constantly improving and bundles different techniques and tools from various fields, for example development, management and process engineering. (Stickdorn, Schneider, et al., 2011)

> *"When you have two coffee shops right next to each other, that each sell the exact same coffee at the exact same price, service design is what make you walk into the one and not the other."*

31 volts service design, 2008

The quote above perfectly describes the main goal of the service design process, which is to create or innovate systems to make them satisfactory to the user in a holistic manner. Additionally, it ensures the effectiveness, efficiency and distinctiveness from the service deliverer's point of view. (Stickdorn, Schneider, et al., 2011)

3.1.1 Basic Principles of Service Design Thinking

Due to the lack of a distinct definition, this thesis uses the often cited description by Moritz, (2005): "Service Design helps to innovate (create new) or improve (existing) services to make them more useful, usable, desirable for clients and efficient as well as effective for organisations. It's a new, holistic, multi-disciplinary, integrative field." On the other hand Mager, (2006) and Stickdorn and Schwarzenberger, (2016) outlined the process of service design thinking by the application of five core principles. Some basic definitions are required to correctly interpret the following section:

- **Touch point:** Every contact point in time between a user and the service provider.
- **Service Evidence:** Tangible artifact or resource related to a service process.
- **Service Period:** Current state of a service. Pre-service (get in touch with the system), service (actually experience a system) or post-service (rate the system).

The principles described below try to describe a possible approach to view the way of thinking required for successful service design: (Mager, 2006) and (Stickdorn and Schwarzenberger, 2016)

1. User centered: Services are created through interactions between provider and user, which means that customer participation early on in the project is important. To fully understand specific habits, needs, behavior and motivation of users, it is crucial to look beyond statistical analyses and empirical profiles. Every person has a different background and level of knowledge that influences their individual service experience. The user-centered approach tries to make this knowledge usable by implementing a common understanding between all stakeholders involved.

2. Co-creative: After establishing a "service user language" the different service propositions, one for each user group, need to be elaborated. This idea about how to create or improve a system needs cooperative work and creativity. Co-creative means that the generation and evaluation of

ideas should be done in representative groups of stakeholders to simulate correlations in the final service. This principle forms the basis for a lasting customer and employee satisfaction.

3. Sequencing: Services have a dynamic nature and take place over a specified range of time. The understanding of this "service rhythm" is indispensable to provide a soothing experience to a customer. In service design thinking complex systems are stripped down to touchpoints that are structured in service periods. Each interaction should be carefully arranged and "tell the story" throughout the whole process. A well-established rhythm results in constant user interest and commitment.

4. Evidencing: Many services take place "behind the scenes" because they either function as assisting activity or are designed to be discreet. By making intangible services visible to the customer, Service Design Thinking tries to create emotional associations with them. This may be achieved by small souvenirs or other physical objects which are aligned with the touchpoint process and revive positive memories of the service.

5. Holistic: Customers perceive a service and its outcome with all their senses, therefore keeping an eye on the bigger picture of the whole service environment is crucial to a great user experience. Varying emotions may influence customer decisions at different touchpoints and could change the course of a whole service journey. The holistic approach in Service Design thinking is used to take possible sequence changes into account and create a satisfying system, no matter which journey is taken.

3.1.2 Iterative Service Design Process

Designing and creating services, systems or products is an extremely complex process that is naturally dynamic and non-linear. Stickdorn and Schwarzenberger, (2016) developed a basic structure to break down this process into four separate phases, which collude in an iterative manner. This

means that there may always be a need to take one or more steps back or even restart from scratch. There are many different tools and methods from various sources which may be used during each step.
The four phases are: (Stickdorn and Schwarzenberger, 2016)

1. **Discovery:** In the discovering phase the first task for a service designer is to get an overview of the organizational structure of a company and the involved stakeholders, which is further used to determine if the environment is already prepared for a successful design thinking process. The second important task is to get a detailed understanding of the *real problem* the service tries to solve and prepare proper visualizations for the involved parties.

2. **Definition:** The generative stage of the process focuses on developing and testing different ideas for the service. A main principle of service design thinking is *not to avoid mistakes, but rather learn from as many problems as possible*. The key to efficient development is to explore the various options early on instead of sliding in extra iterations after the launch, which does not only negatively influence the companies reputation but also often results in exploding costs.

3. **Reflection:** Together with creation, this stage is probably the one which is repeated most during the whole process. Testing intangible services is substantially harder than prototype evaluation for physical products. The main task for a service designer is to prototype a vision of the future service in environments which are close to the reality. Roleplay methods, staging concepts or storyboards are good ways to create a mental picture that reflects the emotions connected to the service.

4. **Implementation:** The last step is to implement the tested service concepts. Extensive change management is required to deploy and monitor improvements to the service. The service designer needs to provide a clear vision of the concept to all employees, because their motivation and engagement is crucial for service implementation. Blueprints are a standard method to keep track of changes made to the system and their repercussions. Ideally, another exploration phase follows a change thus completing the iterative service design process.

Figure 12: Use Case Technology Mapping, adapted from Vorraber, Neubacher, et al., (2018).

3.2 Applied Tools and Frameworks for Service Design

In the course of realizing a well-structured service design process in this thesis, various standardized frameworks and tools are applied in chapter 4. This section explains the theoretical background of the most important methods used for the exploration phase and elaborates the interconnections between them.

3.2.1 The UCTM Framework

The *Use Case Technology Mapping* orignally introduced by Vorraber, Voessner, et al., (2014) and adapted by Vorraber, Neubacher, et al., (2018) serves as foundation for the practical part of this thesis. The authors described a two-phase approach to discover and analyze possible use cases and applicable technologies for a service, positioning it as tool for the *"Exploration", "Creation" and "Reflection"* phases of service design thinking. Figure 12 depicts a visualization of the UCTM framework.

The *human-centered and process-driven approach* focuses on identifying needs of individuals or organizations. At first extensive Requirements Engineering (section 3.2.2) is performed and the results are formulated as use cases. In

combination with process analysis, different levers for improvement are identified. The *technology-driven approach* on the other hand concentrates on innovation strategies and research. Current and future technological developments are analyzed and formalized as functions and tools.
Finally, the authors developed a mapping process to create service concepts, which classifies the obtained results and combines the most encouraging use cases with the best suited technologies. (Vorraber, Neubacher, et al., 2018)

3.2.2 Requirements Engineering

Requirements Engineering is a technique utilized in the UCTM framework to collect consumer demands and needs in a systematic way. In this thesis the following formal definition is used:

> *"Requirements-engineering (RE) is a cooperative, iterative and incremental process. It's main goal is to ensure that all essential requirements of a system are well-known and understood in a detailed manner. Furthermore RE is used to ensure that all known requirements are well-documented and that the involved stakeholders reach an agreement concerning all of them."*

Rupp, Simon, and Hocker, 2009

Requirements   A requirement is a statement about a property or performance of a product, a process or the persons involved in the process. This very basic definition does not really reflect the multidimensional nature of the term. A desired feature of a software expressed by the customer verbally often differs from the actual documented specification in the project, which means that problems occurring in the final stages of a product or service development cycle can often be traced back to the wrong formalization of a requirement in the first place. Therefore, one of the most important functions of RE is to correctly derive implications from verbal task descriptions and convert them into well-documented requirements. Since there exist different possible categorizations of requirements in relevant literature (Nascimento

et al., (2012), Salado and Nilchiani, (2014) and others) this thesis follows the abstract proposal by Rupp, Simon, and Hocker, (2009). First of all they can be distinguished by **type**. *Functional requirements* describe actions which should be performed by a system, interactions between users and the system or specifications about general agreements regarding functional behavior. *Non-functional requirements* on the other hand cover all needs that are not directly related to the functionality of a service. Examples are specifications about the user interface, technological details, legal concerns, qualitative aspects or financial boundaries.

Another possible distinction is the **"legally binding nature"**. *Obligations* are legally required and without them the system is not allowed to operate. *Wishes* are not legally necessary, but it is very desirable to realize them and may only be neglected in exceptional situations. Finally, *intentions* include future developments and standards in technology which must be considered in later iterations, but are not absolutely essential at the moment. (Rupp, Simon, and Hocker, 2009)

Stakeholders   A stakeholder of a system is a person or organization who has direct or indirect influence on requirements of the respective service or system. Stakeholders are the most important source of information for the definition of goals, demands and boundary conditions of the service. Given this setting, it is highly important as a first step to identify all involved stakeholders. A good way to categorize the resulting actors is the **"Motivation-Influence" matrix**, which defines four different types of stakeholders and proposes according rules for interaction with people from each category. (Rupp, Simon, and Hocker, 2009)

An extended form of this matrix was proposed by Vorraber and Voessner, (2011) and is discussed in section 3.2.4.

The RE Process   The process of requirements engineering consists of four parts, which are performed in sequential order. During the **survey** phase all necessary data is collected to get a holistic view of the system and its environment and a complete understanding of the desires and problems of all stakeholders. Since this phase is very similar to the "exploration" stage of the service design thinking process (section 3.1.2), identical tools

for example interviews, questionnaires or video analysis may be used.

Documentation of requirements in a correct and complete way is the most complex and challenging activity in RE. By using the collected data from the survey phase as a foundation, the general task of this step is the exact formalization and formulation of all requirements for the system.

After all requirements are documented, it is crucial to verify the quality of the obtained results in the **evaluation** step. Reviews and inspections should be performed together with relevant stakeholders to validate all specifications against test simulations. Requirements that do not completely fulfill the necessary quality criteria need to be reworked or eliminated.

Finally, all requirement-related activities need to be traced and managed throughout the project. The continuous **management** process provides standardized methods to organize all relevant information and RE documents in a feasible way, even for huge amounts of data. Additionally some requirements may change over time or need to be refined, therefore this task also includes frequent communication with all stakeholders. (Rupp, Simon, and Hocker, 2009)

Problems in RE   As stated above, requirements engineering is a very complex activity and there are potential pitfalls that one has to watch out for: (Rupp, Simon, and Hocker, 2009)

- **Unclear objectives:** If many different user groups are interacting with the system, the main functionalities of the service may be different for each individual group. If not all stakeholders are reflected accordingly, some important system objectives may be missed.
- **High complexity:** Internal processes of companies are getting more and more complex. Following this trend, supporting systems and tools also register a noticeable increase in complexity, leading to mutual dependencies between different specifications.
- **Language barriers:** Due to progressing globalism, projects are often organized and implemented internationally. Problems in communication between stakeholders may lead to vague descriptions of requirements.
- **Constantly changing goals:** The world continuously spins and the surrounding environment of a system changes over time. Stakeholders may change their minds frequently, making it hard to fulfill the
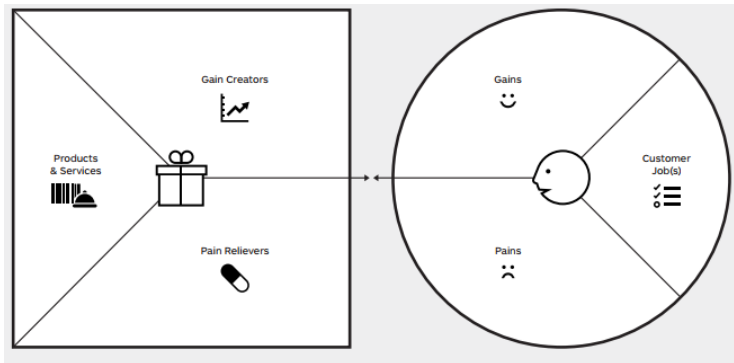
Figure 13: Value proposition canvas, taken from Kyhnau and Nielsen, (2015) and Osterwalder, Pigneur, et al., (2014).

requirements.

- **Bad documentation quality:** A well-formulated requirement is complete, clear, correct, coordinated between all stakeholders, consistent, realizable, necessary and testable. Implementation should not start before all specifications fulfill this criteria.

3.2.3 Value Proposition Canvas

The value proposition canvas is a strategy tool proposed by Osterwalder, Pigneur, et al., (2014) to analyze and visualize the value possibly created by a service or product. It focuses on the customer and product segments of the high-level *Business Model Canvas*. (Osterwalder and Pigneur, 2010) This method is a practical implementation of the user-centering and co-creation principles (section 3.1.1) of service design thinking. Figure 13 shows the building blocks of the value proposition canvas.

The customer profile on the right side of the canvas focuses on the potential customers. *"Customer jobs"* are all activities a person performs to reach a specific goal. *"Pains"* are problems or negative emotions from the customer's point-of-view, which arise while executing the customer jobs. *"Gains"* on the other hand describe positive emotions, dreams and benefits of a person if the service is realized.

The value proposition represents the value that is created by the proposed

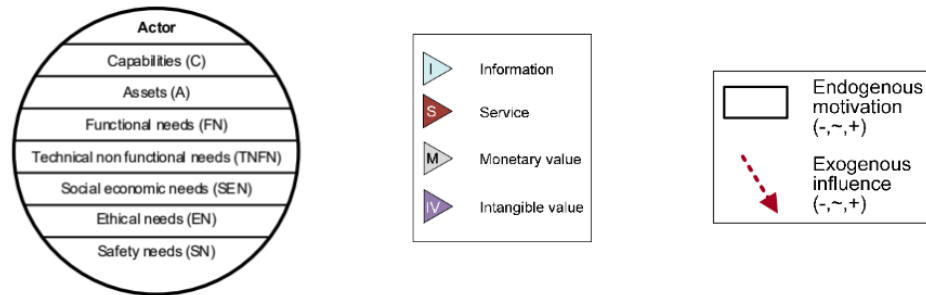Figure 14: V2 value network notation basic building blocks, taken from Vorraber and Voessner, (2011) and Vorraber, (2016).

service. As a first step, all small sub-services and products offered to the customer are listed as *"Products and Services"*. *"Pain relievers"* and *"gain creators"* are concrete solutions and differentiations from competitors, which are provided by the service to make the life of a customer easier.

3.2.4 The V² Value Network Notation

Different economic stakeholders are working together in a network-like structure to create benefits for the customer in nearly every service today. To model the complex value stream, Biem and Caswell, (2008) introduced a basic technique to represent it as value networks. However, their methodology is not able to take forces into account that influence the different value-creating entities, for example highly motivated stakeholders are more likely to produce more value in better quality, whereas discouraged agents may slow down the whole network. To close this gap, Vorraber and Voessner, (2011) proposed a more advanced modeling technique, the "V² value network notation". The basic building blocks of this notation are economic entities, transfer objects and forces, which are depicted in figure 14.

An **economic entity** is formally described as an actor for whom it is possible to design an individual business plan. This entity can either be a firm, a business unit or an individual and is described by specific characteristics. The standard notation introduced by Biem and Caswell, (2008) includes:

- *Capabilities:* Process or activities executed by the entity in the value network.

- *Assets:* Static values (goods or knowledge) which are permanently connected with the entity.

The following adaptions were proposed by Vorraber, (2016):

- *Functional needs:* Functionalities that are necessary to fulfill an activity.

- *Technical non-functional needs:* Demands associated with non-functional requirements of a system.

- *Social economic needs:* Economic needs of actors in terms of how they are seen by others, for example to present oneself as innovative and successful.

- *Ethical needs:* Need for conformity of actions with the ethical standards of a stakeholder.

- *Safety and legal needs:* Protection in terms of personal safety of an actor during service usage and respecting legal restrictions.

Economic entities generate and transfer different types of value inside the network. These so called **transfer objects** are represented in the notation as arrows and are distinguished in the context of service systems in the following way: (Biem and Caswell, 2008)

- *Information:* Transfer of a trading good.

- *Service:* Delivery from a service provider that changes the assets of a recipient in any way.

Vorraber and Voessner, (2011) introduced two more transfer objects:

- *Monetary value:* Transfer of money or financial equivalents.

- *Intangible value:* Model for customer experience, for example positive ratings for a service.

As already mentioned, special **forces** that influence the value generation process in the network are core elements in the "V2" notation. Vorraber and Voessner, (2011) proposed the concept of endogenous motivation and exogenous influence to model these forces accordingly. Figure 15 shows the
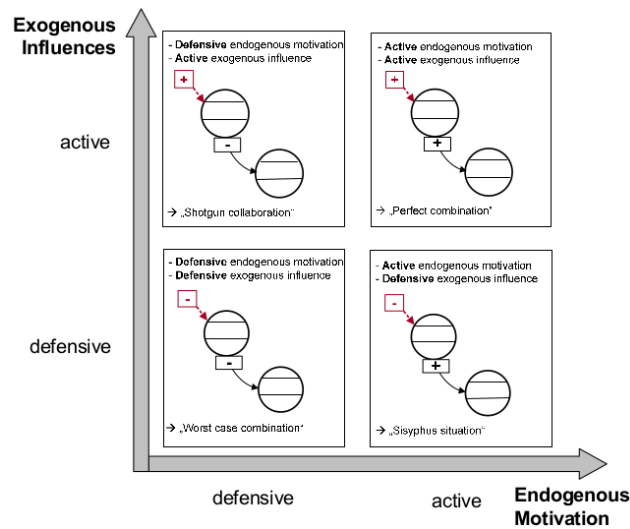
Figure 15: Endogenous motivation and exogenous influence matrix, taken from Vorraber and Voessner, (2011).

adapted "Motivation-Influence" matrix which reflects the interplay of the two forces.

Endogenous Motivation   The level of motivation the employees of an agent in the network bring to the service system. It is based on the "expectancy theory" by Vroom, (2005), which describes the fact that employees tend to adapt their behavior and motivation at the workplace to generate the highest possible reward, for example a promotion. This means that workers are investing more energy and resources into tasks that promise a usable outcome for them. There are three types of endogenous motivation. Employees follow a **defensive** strategy, if they only execute value tasks which are not conflicting with their personal goals. A **neutral** attitude results in collaboratively and timely execution of value tasks, but personal goals are still more important to the employees. **Active** motivation means that value responsibilities have a high priority for the employees and receive a lot of attention. (Vorraber and Voessner, 2011)

Exogenous Influences   External forces from the management of a company, for example direct orders, or other sources that ease or complicate the involvement of an agent in the value network are classified as exogenous influences. These forces may have several reasons like strategic decisions or legal regulations. Influences can occur in the same forms as motivation. **Defensive** forces are impeding the value task, while **neutral** ones are neither smoothing nor complicating the value network activity. Finally, **active** influence tends to encourage value network activities, for example by implementing special reward programs. (Vorraber and Voessner, 2011)

Implications   According to the matrix in figure 15, endogenous motivation and exogenous influences mostly occur as a combination and their nature lead to specific implications for the whole network: (Vorraber and Voessner, 2011)

- *Worst Case*: In this scenario, defensive motivation and influence come together. Agents with these circumstances may become severe bottlenecks for the network as a whole and the business case is very likely to fail.
- *Sisyphus Situation*: If active motivation comes upon defensive influence, considerable efforts are required to create a positive performance for the actor. Department internal projects often generate lots of motivation for the involved employees, but concerns from the top management may decrease the belief in the idea.
- *Shotgun collaboration*: The combination of defensive motivation and active influence on the other hand poses a considerably greater threat to the business case. Employees are forced on value network tasks by an enthusiastic management, but do not understand the reason behind them or are not able to see a benefit emerging for themselves. Such agents are likely to become energy draining partners for other actors in the network.
- *Perfect combination*: The ideal scenario of active motivation and influence. Actors that can fall back on this constellation are very valuable and can be high performers in the network.

The V² value network notation is an important tool for the strategic analysis of a use case and will therefore be applied within this thesis 4.

4 Case Study

This chapter describes the practical implementation of the tools and frameworks introduced in chapters 2 and 3. At first a detailed documentation of the performed use case selection process, based on the UCTM framework, is provided. This section also includes a Value Proposition Canvas as well as a V² Value Network analysis. In the next step an extensive description of the multiple parts developed as a Proof-Of-Concept (POC) is provided. This part covers aspects like the general system structure, network architecture, UI design and implementation details regarding the software. Finally, the chosen evaluation process for the POC is documented and the consequential results are discussed.

4.1 The Use Case

The definition of a suitable use case for this case study, based on the findings described in the previous chapters, presented itself as one of the main challenges early on. To ensure a high practical value in the future, it is especially critical to conduct extensive requirements engineering in close cooperation with potential users of the new technology and domain experts in the chosen field. Therefore the first step in the decision process was to narrow down the main application areas presented in section 2.3 to a specific starting point for the analysis. A long-term partnership between the MBI and a hospital located in Graz tipped the scales in favor of the medical field. The UCTM framework (see section 3.2.1) provides a well-structured process to facilitate the problem of use case definition. The following section demonstrates the implementation of these techniques in a medical setting with focus on AR.

4.1.1 Technology Driven Approach

As already described in section 3.2.1, the first part of the UCTM framework is the technology driven approach, which aims at innovation and research. The rise of publicly available Augmented Reality devices quickly sparked interest in the medical sector, since it enables new levels of remote consultation to doctors and other medical staff. After extensive research and discussion based on the findings presented in section 2.4 the decision was made to use a **Microsoft HoloLens** as tool for this Proof Of Concept. Main reasons for the decision have been the hardware and processing power of the system as well as the easy accessibility for the project, since the MBI has one item available at all times.

Table 4 gives a short overview of the most important hardware specifications of the HoloLens.

4.1.2 Human-centered and Process-driven Approach

The second part of the UCTM framework is the human-centered and process-driven approach to identify needs and requirements. At the beginning of the process a panel consisting of relevant stakeholders and experts from both medical and technical areas was formed.

Step 1 - Use Case Scouting: Throughout a number of panel meetings a profound use case scouting was performed. Three main use cases in the medical field were identified as interesting areas for AR assisted remote consultation:

1. *Pre-operative anamnesis*

2. *Ultrasound procedures*

3. *Advanced Life Support courses*

After some discussion the second use case regarding ultrasound was chosen as most promising.

Category	Specifications
Optics	See-through holographic lenses (waveguides)
	2 HD 16:9 light engines
	Automatic pupilary distance calibration
	Holographic Resolution: 2.3M total light points
Sensors	1 IMU, 4 environment understanding cameras
	1 depth camera, 1 2 MP photo and HD video camera
	4 Microphones
	1 ambient light sensor
Human Understanding	Spatial Sound, Voice support
	Gaze tracking
	Gesture recognition
IO / Connectivity	Built-in speakers, Audio 3.5mm jack
	Wi-Fi 802.11ac
	Micro USB 2.0
	Bluetooth 4.1 LE
Power / Battery	Active Use: 2-3 hours
	Standby: Up to 2 weeks
	Passive cooling (no fans)
Processors	Intel 32 bit architecture with TPM 2.0 support
	Microsoft Holographic Processing Unit (HPU 1.0)
Memory	64GB Flash Drive, 2GB RAM
Weight	579g
OS	Windows 10

Table 4: Overview of HoloLens Hardware Details, adapted from Microsoft Inc., (2016a)

Step 2 - Process Analysis: For the selected field of application, *ultrasound procedures*, the underlying processes were analyzed to detect potential areas for improvement.

Typical process types for ultrasound procedures are:

1. Emergency screening at the ER

2. Control examinations at the hospital or at a local practitioners office

3. Training procedures for inexperienced staff

After the identification of the most important underlying processes, a requirements engineering process was performed to detect possible improvement levers.

Step 3 - Improvement Levers: The following levers for an improved ultrasound consultation were discovered during the requirements engineering process:

- *Improved knowledge transfer* with AR assisted remote consultation. Using a HMD and input from an expert the position and orientation of a ultrasound probe could be displayed in the field-of-view of an inexperienced doctor or student. It allows the trainee to learn the necessary motion sequences on their own, without disturbing mechanical influences from a supervisor.

- *Improvments related to health policy*. Due to increasing specialization, centralization and restriction of working hours the availability of expert knowledge at a specific time at a specific place is very limited. AR assisted remote consultation eliminates the need for physical presence of an expert at the training site and has therefore sustainable practical relevance.

4.1.3 Use Case Specification Scheme

For the concrete realization of the Proof Of Concept, it is absolutely necessary to formulate the results and findings of the UCTM framework in a precise and standardized manner. The *Use Case Specification Scheme* provides

a well-defined way to achieve this goal by giving an holistic overview of the complete use case and the underlying process steps. In this scenario there are two actor groups using the system: trainees and medical experts. Experts can be classified as active users because they directly interact with the system and actively provide input. The trainees on the other hand can be seen as passive users, because they are only receiving information from the system and react accordingly. To provide a working framework some preconditions and additional constraints must be met. For example the trainee must be equipped with a HoloLens and be physically present at the scene. The expert on the other hand needs the live images from the ultrasound machine to provide help. All details of the specification are listed in table 5.

4.1.4 Value Proposition Canvas

The theoretical background of the value proposition canvas is described in chapter 3.2.3. Now the canvas is applied to the use case specified above to get an even better understanding of the value created by the new service. This can be used to project the practical relevance of the business model for the future.

The first part of the canvas is the *customer profile*. Together with medical experts some notable customer jobs, gains and pains regarding ultrasound examinations were defined. As an example, one of the most important customer job certainly is the correct identification of diseases during the process. This job may cause pains for an untrained doctor who performs the procedure, because some parts of the human body are hard to scan and require a very precise position and angle of the probe. A possible gain that a customer probably expects from the new service could be an improved procedure accuracy in terms of handling and output.

The other side of the canvas represents the *value map* of the service. The Proof-Of-Concept includes products like a HoloLens application for the trainee. It creates a gain for the customer by providing intuitive guiding and learning. This feature may serve as a pain reliever because it provides clear visual instructions regarding position and angle of the probe. The full canvas was evaluated by two domain experts during extensive feedback

4 Case Study

Name	AR-assisted remote ultrasound training
Short Description	A trainee performs an ultrasound procedure wearing a AR-HMD and gets remote assistance from an expert working from a different location.
Actors	Trainee, Medical Expert
Preconditions	- Activated HoloLens on the trainee side - Desktop PC with connected input device on the expert side - Available network connection on both sides - Activated ultrasound scanner with live images visible on both sides
Trigger	Trainee starts ultrasound procedure
Typical Process	1. Trainee asks expert for help. 2. Expert starts system and establishes a connection to the HoloLens. A live stream of both trainees field-of-view and ultrasound images are visible. 3. Expert specifies location on patients body as starting point. 4. Information is visualized in front of trainee. Trainee adjusts probe to appropriate location. 5. Expert may change location or move on to next step. 6. Expert indicates necessary rotation of probe to see correct image by moving input device. 7. Trainee sees virtual moving probe blended into field-of-view. Trainee adjusts probe orientation 8. Next part of the body (Step 3) or end of procedure.
Additional Constraints	- HoloLens must be charged at all times - Additional audio communication necessary

Table 5: Use Case Specification Scheme
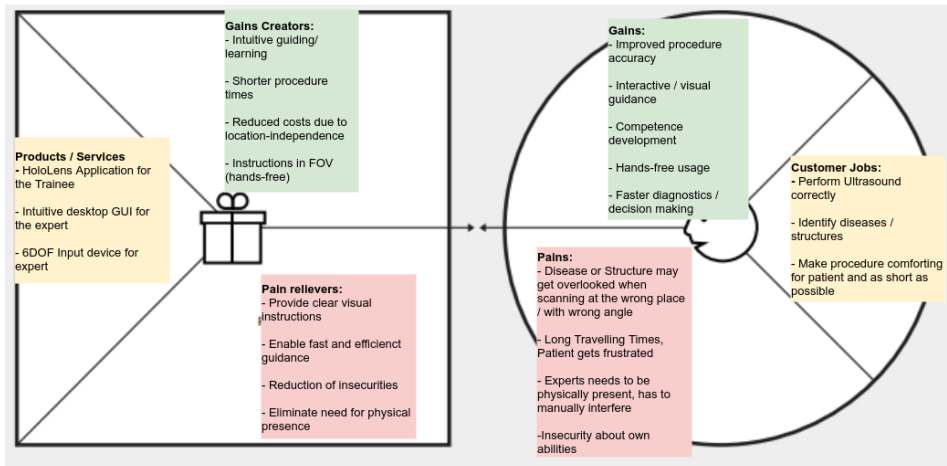
Figure 16: Value Proposition Canvas for Remote AR Ultrasound Consultation

sessions (see section 4.3) and is depicted in figure 16.

4.1.5 Value Network Analysis

As explained in chapter 3.2.4, the V² Value Network Notation is a powerful tool to model forces between participants of a system. The first step of applying the concept to the context of ultrasound AR remote consultation was the definition of all relevant actors:

- **Task-related:**
  - Patient
  - Trainee
  - Medical Expert

- **Organization-related:**
  - Hospital IT-Department
  - Hospital Management

- **Environment-related:**
  - Ultrasound Machine Manufacturers
  - Insurance Companies

Actor	Endogenous Motivation	Exogenous Influence	Expert Remarks
Hospital Management	Neutral (˜)	Info not available	*Positive*: Save costs (no physical presence of expert necessary), Students learn faster *Negative:* Image Problem (may suggest unawareness), Additional time for enrollment required
Hospital IT-Department	Positive (+)	Neutral (˜)	IT employees motivated because of new technology, superiors maybe skeptical because of necessary work and support
Expert	Neutral (˜)	Positive (+)	Get positive enforcement from superiors. Are able to assist remotely, but earn less money and barrier to call for help lower
Trainee	Positive (+)	Positive (+)	Biggest profiteer. Gets visual guidance and advanced support from experts. Encouraged by management to use the system.

Table 6: Value Network Notation - Endogenous Motivation and Exogneous influences

As a next step all capabilities, assets and different types of needs of these actors were defined. For reasons of simplicity it was assumed that in practice all capabilities and assets related to the new service are set within the IT-department of a hospital. Finally, all in-port and out-port transfer objects between the individual actors were elaborated. Figure 17 shows the complete results of the definition phase.

Another important aspect to correctly map all aspects of the value network are *endogenous motivation* and *exogenous influence* forces that affect specific actors and therefore the whole network. Similarly to the Value Proposition Canvas, two domain experts validated the whole network and provided insights which allowed to deduce these forces in a exact way. Table 6 shows the results of the evaluation workshops.

The resulting network is depicted in figure 18.

4.2 Proof-Of-Concept

This chapter is elaborating on the Proof-Of-Concept developed within the case study, which is based on the detailed use case specification in section 4.1.3. In contrast to the previous, mostly methodical chapters, the focus of this part is strongly placed on technical aspects. It aims to provide insights into the thought processes and design decisions at every stage of the project. At first a general technical system overview is provided, which includes components as well as communication patterns. The remaining sections follow up with algorithms, software design patterns and technologies used to realize all components and aim to illustrate the product at all stages throughout the development processes.

4.2.1 Technical System Overview

In order to enable the implementation of a functional prototype for the use case "Remote ultrasound guiding through a HoloLens", a technical system overview is required. This analysis maps the abstracted components and

Actors	Capabilities	Assets	Needs							In-port transfer objects	Out-port transfer objects
			Functional Needs	**Non-functional Needs**							
				Technical non-functional needs	Social economic needs	Social human needs	Ethical Needs	Safety Needs	Legal Needs		
Patient	Provide feedback about procedure comfort	Medical Records	Thorough examination	Short examination times	-	-	-	Not getting hurt during the examination	-	Service (Ultrasound)	Monetary Transfer (Mgmt)
Trainee	Execution of ultrasound procedure	Basic medical knowledge	Medical guidance	Low latency of the service, Wearing comfort of the HMD, Short set-up time	Wants to be perceived as competent by the patient	Help the patients	Respect privacy of patient	Not getting hurt or sick by using the service	Complying with legal regulations while using the service	Information / IV from expert, Monetary Value from hospital	Service to patient, Information to expert
Expert	Provide guidance to trainee for ultrasound procedure	High profile medical expertise	Tool for guidance	Low latency, Intuitive usage of the service, Short set-up time	Wants to be perceived as expert in the medical field	Help the patients	Respect privacy of patient	Not getting hurt or sick by using the service	Complying with legal regulations while using the service	Information from trainee, Monetary Value from hospital	Information / Intangible Values to trainee
Hospital IT-Department	Provide IT-infrastructure and services	IT-Experts, IT-Equipment (incl. HoloLens Service)	Fulfillment of IT-infrastructure regulations	Easy set-up and maintenance, Low influence on IT-infrastructure (bandwith utilization)	Want to be seen as innovative and competent	Facilitate work for the employees	-	-	Complying with legal regulations while providing the service	Monetary Value from hospital	Services to trainee and experts
Hospital Management	Provide monetary budget to actors, Define medical framework conditions, Provide information to experts	Monetary Budgets, Medical Know-How, Management Competences	Reports of procedure execution	Service completely integrated into the system, No touchpoints,	Want to be seen as innovative and always in control	Provide high-quality environment for patients and staff	Avoid violations of ethical standards by the servie	-	Avoid legal fallouts due to the service	Monetary Value from insurance	Monetary Value to IT-department, expert, trainee, ultrasound manufacturers
Ultrasound Manufacturers	Provide ultrasound machine, Maintenance	Technological Know-How, Staff	Seamless integration of the service	No side-effects of the service regarding the machines	-	-	-	-	-	Monetary Value from hospital	Service to trainee
Insurance Companies	Collect insurance fees and transfer of monetary values	Financial Know-How, Administrative Structure	Reports of provided services	-	-	Provide health services to all patients	-	-	-	Monetary Value from patient	Monetary Value to hospital

Figure 17: Value Network Notation Definitions for Remote AR Ultrasound Consultation

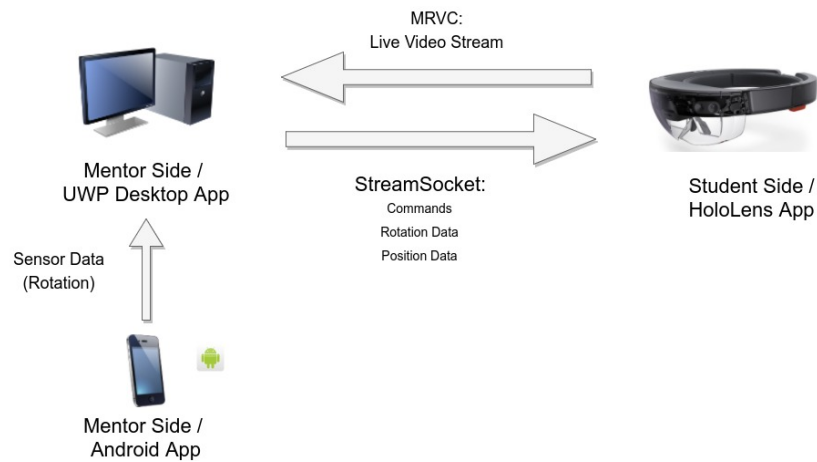Figure 18: Value Network Notation for Remote AR Ultrasound Consultation

Figure 19: Proof-Of-Concept - System Overview

the data flow between them. Figure 19 shows the resulting diagram.

System components   As depicted in figure 19 the system as a whole consists of individual components which are connected through different network interfaces.

At the highest level two main elements can be distinguished:

- **The remote (mentor) side:** In the final setting, a domain expert uses the applications provided with this component to guide a trainee remotely through an ultrasound procedure. It includes a live video feed received from the HoloLens and provides the ability to transfer different kinds of data into the trainees field-of-view in real-time.

- **The local (trainee) side:** This versatile component acts as a rendering and presenting tool for the trainee. The HoloLens application captures live video data on the one hand and displays the remote input in a convenient way to facilitate the learning process.

The mentor side itself is formed from two separate applications. An Universal Windows Platform (UWP) desktop application serves as the core of the architecture. It implements a sophisticated user interface and also acts as port between the other applications.

Additionally an Android smartphone, together with a specifically developed native application, is used as a tool to enable input with six degrees of freedom (6DOF). This data is transfered over the desktop application to the HoloLens. Details regarding the implementation of all components are provided in the following sections.

Live Streaming One of the "backbones" of the Proof-Of-Concept is the availability of a stable, low-latency live video stream from the HoloLens. Since a remote expert should provide guidance and feedback in real-time, it is crucial to deliver a first-person-view from the trainee wearing the HMD. As a first step to realize this feature, it was important to list all requirements, which need to be fulfilled without restrictions:

- Low latency network connection (lower than 300ms)

- Sufficient resolution (min. 360p (634x360))

- Standard Video encoding

- Stability while running on the HoloLens

- Rendering of Holograms into the RGB video stream

- Support for Windows UWP desktop applications at the receiving end

After extensive research and enforcement of the restrictions stated above, two possible solutions emerged: An open-source framework (WebRTC) and an AddOn by Microsoft (Mixed Remote View Compositor).

WebRTC: WebRTC is an open-source project originally created by Justin Uberti and Peter Thatcher in 2011. It provides applications with Real-Time Communications (RTC) capabilities via simple APIs. The main advantage of this framework is the flexibility provided by enabling common protocols on all platforms. After gaining support by companies like Google, Mozilla and Opera, WebRTC was standardized by the World Wide Web Consortium (W3C) in 2016. (WebRTC, 2017b)
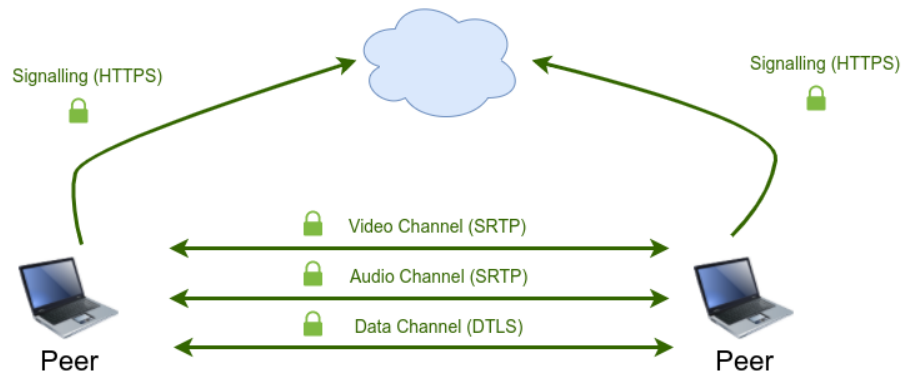
Figure 20: WebRTC Architecture Overview - Adapted from WebRTC, (2017a)

WebRTC uses a complete Peer-to-Peer approach and only relies on external server communication to identify the communicating partners. Different channels for video, audio and additional data are used to achieve the best possible quality for the stream. Figure 20 gives an overview over the basic WebRTC architecture.

Even though the framework looked promising in the beginning, some limitations regarding Windows and the HoloLens emerged early in the development process. Since WebRTC was mainly released for web-based solutions, a special plugin designed for use in native C# applications is required. Additionally, Unity (see section 4.2.3) only ships with the early .NET version 3.5, which does not support external plugins like WebRTC or other advanced functionalities for media streaming. Implementing a so called "Stub Plugin", which basically is a .NET 3.5 container translating code from later .NET versions, presented a large challenge and would have overstepped the bounds of this thesis. *For this reason, the decision to go with Microsoft's MRVC was made.*

Mixed Remote View Compositor (MRVC): The MRVC project was released by Microsoft in early 2017 as part of the "Mixed Reality Companion Kit", which is a set of scripts and tools to facilitate the work with augmented reality applications and hardware. It is based on low-level Media Foundation elements and a native C++ network interface for communication. Figure 21 shows the most relevant parts of the framework used in this thesis. Even
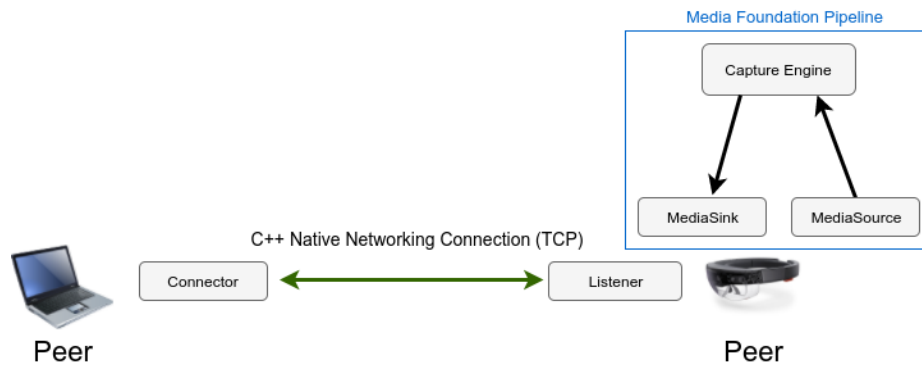
Figure 21: Mixed Remote Compositor View Overview - Adapted from Microsoft Inc., (2017c)

though the source code is mainly native C++, Microsoft delivers the MRVC project with C# AddOns for Unity and Windows UWP, which makes it an ideal fit. The following components of the framework are most relevant to the purpose of this project: (Microsoft Inc., 2017c) and (Microsoft Inc., 2017a)

- **Connector:** The *Connector* class is implemented in the UWP desktop application on the mentor side and serves as an endpoint to establish a network connection to the remote peer. It configures all necessary parameters, such as port and ip address and issues a connect request to a listener.

- **Listener:** The *Listener* class on the other hand is implemented at the side of the capture device, in this case the HoloLens. It listens on a specific port and creates a connection object when a connect request from the Connector is received. This connection is a type of an low-level C++ TCP socket stream and may be used by both parties to send and receive data.

- **Media Foundation Pipeline:** Media Foundation is an extensive multimedia platform and API for Windows and bundles robust media capture and rendering APIs. The *capture pipeline* is a set of cross-linked functionalities which are specifically designed to capture different types of video and audio sources.

- **Media Source:** Objects that generate media data. In this context the RGB camera of the HoloLens as well as the holographic engine represent media sources.
- **Media Sink:** Objects that receive media streams from one or more media sources. This class prepares the raw data for playback by encoding and compressing it.
- **Capture Engine:** Implements functionality to connect sources with sinks and performs spatial calculations. It combines the raw RGB video frames with the holographic data with respect to the current orientation of the device. The final source data is then provided to an appropriate video rendering sink and sent over the network.

The above mentioned components are used in custom classes called "Stream Manager" on both sides. Those wrappers ensure full interoperability and provide additional functionalities like rendering, error handling scenarios and automatic reconnect features (see sections 4.2.2 and 4.2.3). The resulting video stream is encoded in MPEG1 video format with a minimum resolution of 360p at an update rate of 30fps. Due to technical reasons, an embedded audio stream is not available.

Additional Data Transfer   In addition to the live streaming service, it is also necessary to implement another connection between the system components that allows arbitrary data transfer. The following types of data need to be sent over the network:

- Different types of commands from the desktop app to the HoloLens
- Position data from the desktop app to the HoloLens
- Rotation data from an Android phone to the HoloLens

As already depicted in figure 19, the first strategic decision was to avoid a direct connection from the Android phone to the HoloLens. This is because of the limited computing and networking power of the HoloLens, which handles already one video and one data connection to the UWP application. Another reason is the fundamental view of the desktop user interface as the core control panel and the smartphone as additional input source for this

UI.

The problem of varying software platforms, which already occured for the live streaming feature, turned out to be even more challenging when dealing with communication between three different systems. Therefore a standardized networking solution is needed that works with .NET 4.5 (UWP desktop), C# Scripting (HoloLens) and Java (Android) is required.

After some research, the Javascript Object Notation (JSON) surfaced as a solution that is applicable on all three platforms. JSON is a lightweight, compact and easy to read data format, which facilitates data transfer between applications. The two main building blocks of the notation are *Objects* and *Arrays*. Similar to object-oriented programming patterns, JSON Objects can be seen as a collection of related variables with varying data types. JSON Arrays on the other hand may function as list of objects or variables with the same data type.

The outstanding characteristic of JSON that makes it a perfect fit for this purpose is fast **serialization**. All JSON data blocks are encoded to strings, a primitive data type available on every platform and in every programming language, and can be easily parsed after the reception. Each of the platforms used for the POC implements different methods to deal with JSON strings:

- **UWP Desktop (C# .NET):** C# with .NET provides the built-in *"DataContractJsonSerializer"* class that allows the transformation of every class instance into a JSON string. As a first step, the class itself has to be labeled as *"DataContract"* and its members as *"DataMembers"*. Nested objects can be achieved by using other DataContract classes as members. Raw data types like byte arrays are not supported. Finally, the object instances may be encoded or parsed with the appropriate serializer.

- **Unity HoloLens (C# Scripting):** The Unity engine ships with the *"JsonUtility"* class, which works similar to the .NET implementation described above. The main difference is the labeling of classes, because the individual members do not require special attributes. Declaring the entire class as *"Serializable"* is sufficient.

- **Android (Java):** The Java platform for Android provides the simplest interface for working with JSON data. *JSONObject* and *JSONArray* are

```
[DataContract]
10 references
internal class Message
{
    [DataMember]
    internal string type;

    [DataMember]
    internal MessageData data;
}

[DataContract]
5 references
internal class MessageData
{
    [DataMember]
    internal Rotation rotation;
}
```

```
[Serializable]
0 references
public class Message
{
    public String type;
    public MessageData data;
}

[Serializable]
1 reference
public class MessageData
{
    public Rotation rotation;
}
```

```
JSONObject message = new JSONObject();
message.put( name: "type", value: "COORD");

JSONObject data = new JSONObject();
JSONObject rotation = new JSONObject();
rotation.put( name: "x", rot[0]);
rotation.put( name: "y", rot[1]);
rotation.put( name: "z", rot[2]);
rotation.put( name: "w", rot[3]);
data.put( name: "rotation", rotation);
```

```
{
    "type": "",
    "data": {
        "rotation": {
            "x": null,
            "y": null,
            "z": null,
            "w": null
        }
    }
}
```

Figure 22: JSON Implementation - .NET (top left), Unity (bottom left), Android (top right) and the resulting JSON string

available as separate data type and may be used directly without the need for conversion. To parse a received string, it simply needs to be provided as argument to the constructor of either class.

Figure 22 shows an example of the different implementation patterns described above and applied within the project, which result in the same JSON output string.

Network Infrastructure: After defining the underlying data structure, it was necessary to determine the network infrastructure to enable frequent data transfer. A very important requirement when working with people is privacy and data protection. Since the selected use case requires the recording of highly personal information like ultra sound images, all data

must be protected from external access. For this reason, the applied concept works in an isolated local network and does not need a connection to the internet.

TCP/IP sockets listening on specific ports prove to be a robust and fast solution for continuous data transfer available on all platforms. For the POC a socket connection was spanned between the UWP app and the Android app and between the HoloLens and the UWP app (see figure 19). Technical details of the implementation and limitations for each platform are described in sections 4.2.2 and 4.2.3.

4.2.2 Expert Side

This section covers the remote parts of the implementation, hereafter referred to as expert side. It consists of an UWP desktop application and an Android mobile application. The following explanations focus on the technical aspects, applied concepts and different iterations of both products.

MBIRemote UWP Desktop App   The core tool developed for this Proof-Of-Concept is without a doubt the Universal Windows Platform (UWP) desktop application. It serves as control station for the remote expert and provides many different functionalities, as described later in this section. The choice to use UWP as platform and .NET as development framework had several (technical) reasons:

1. Microsoft Windows is the world market leader in commercial OS distributions and is widely spread across public health facilities.

2. The Microsoft HoloLens, which is the main communication partner during the service, is powered by an earlier version of .NET.

3. MRVC Live Streaming (see section 4.2.1) is easy to integrate with UWP.

4. UWP allows deployment of the same code base to many different devices that run Windows 10, for example tablets or surface books.

Programming Pattern: Using .NET for UWP, it is straight forward to implement one of the most common patterns in software development:

Model-View-Controller (MVC). The general idea behind the MVC system is the strict separation between UI layout, business logic and data. Using this principles and guidelines makes the code base a lot easier to maintain and to modify in future iterations.

MVC was implemented in the "MBIRemote" application in the following way:

- *SocketManager and StreamManager (Model):* As their name already suggests, these two classes are responsible for handling the live streaming service and additional data sockets. They manage network communication to external services and administrate all types of dynamic data used. In accordance with the MVC pattern, there is no direct communication between these instances and the GUI.

- *MainPage Controller:* This class serves as link between the GUI and the model. It handles user input, retrieves the according information from the streaming service or the data sockets and updates the GUI. It is important to mention that the running instance is operating on a different thread then the models, to avoid visible freezes during expensive operations.

- *General User Interface (View):* The View is realized with the Windows Presentation Foundation (WPF) graphical framework. It displays the retrieved data in a appealing fashion to the user and also listens for user inputs, which are then transmitted to the controller.

Due to the object oriented nature of the used programming language C#, the elements mentioned above are all implemented as individual classes.

UI Design: As mentioned above, the user interface was realized with WPF, which uses the Extensible Application Markup Language (XAML) to define graphical objects. XAML is based on XML and provides a variety of standard UI elements to choose from. For reasons of simplicity the design of MBIRemote application is inspired by the standard "Windows 10 look" for store apps. Due to the spiral development process carried out within the project (described in detail in the documentation for the Android application in the next section), it was necessary to modify the user interface multiple times. Figure 23 shows the final mock-up after the last iteration. The UI is composed of the following modules:
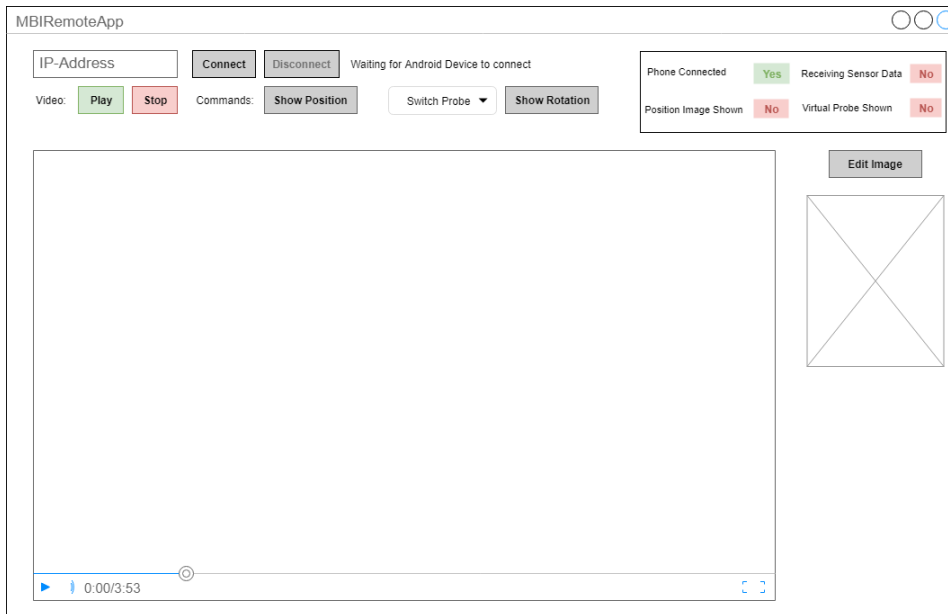
Figure 23: Final Mock-up of the MBIRemote Desktop App GUI

- *Connection Area:* Positioned at the top left corner. Contains all tools that are associated with external communication.

- *Media Area:* Below the connection tools. Elements to render and control the live video data.

- *Dashboard:* Located at the top right corner. Status indicators signaling the current state of the application to the user.

- *Command Area:* Below the dashboard. Buttons and other tools designed for active user interaction.

Functionalities: After taking a closer look on the layout of the system, the focus is now shifted towards the actual functionality provided to a user. This comprises automatically prepared and rendered information as well as methods designed for processing different kinds of input.

The subsequent part is structured in the form of a typical use case and highlights each feature individually and if necessary, illustrates them with screenshots.

4 Case Study

#1 Connecting to the HoloLens:

After the start up process is completed, all UI elements are disabled except the connection area. To use all other tools it is necessary to connect to external devices. A connection to the HoloLens requires the IP-Address of the device, which needs to be determined beforehand and is entered into the respective input field. If the attempt to connect was successful, the media and position controls are enabled (see #3 and #4). If the attempt fails the user is notified via a pop-up message. The whole process is portrayed in figure 24.

Figure 24: Connecting to the HoloLens application (left) and error message (right)

#2 Connecting to the Android phone:

To enable the rotation features, a connected Android phone is required. In contrast to the HoloLens, a connection attempt is not initiated by this application. In the top left corner, right next to the "Disconnect" button, the PC's own IP address is displayed and needs to be entered into the corresponding field in the Android app. A successful connection is indicated in the dashboard and the rotation feature is available as well (see feature #5).

#3 Starting the Live Stream:

With the media controls, located below the connection panel, the user is able to control the live stream from the HoloLens. After the connection is successfully established, the "Play" button is enabled. Pressing this button triggers a request to the MRVC engine, which initializes the hardware and starts retrieving frames over the media pipeline.

The resulting stream of data can not be displayed directly by the application, because the media player element used in the UI is only able to play back video data from a URI. In .NET *scheme handlers* provide the ability to register custom media extensions or URIs and link them to a data source. For this application a new scheme handler *"mrvc://"* was created and subsequently used as input for the player.

Using the "Stop" button, the user may interrupt the broadcast at any time. On restart, the resources do **not** need to be re-initialized.

In the case of "passive" interruptions of the stream, either caused by the HoloLens power management system or an unstable network, the system automatically attempts to restart as fast as possible.

#4 Indicating the Starting Position:

Determining the ideal initial position of the probe on the human body is both crucial for a successful ultrasound procedure and also hard to learn at the same time. The realization of this particular feature has changed the most throughout the development process. Some technical limitations (see next sections for more details) and the feedback from domain experts forced many adaptions over time. The final solution is a image-based one and basically relies on the drawing of a human torso.

On the far right side of the user interface the drawing mentioned above is visible for the expert at any time. At first it does not contain any hint for the trainee and must be edited by clicking the according button (see figure 25).

This triggers a second window, which opens up in the foreground. As can be seen in figure 26, the content of this frame consists of a large verison of the drawing and a toolbar. At this moment, the expert is able to indicate the correct starting position of the probe for a specific procedure by drawing on the provided outline. The type of drawing instrument as well as the color is freely selectable. In the case of a mistake, parts of the contents can be erased or the whole canvas may be cleared entirely. Finally, the changes may be saved or discarded and the window is closed.

In case the modifications were saved, the thumbnail displayed in the command area has been updated. At this point the instructions can be displayed to the trainee by executing the "Show Position" command. As long as the
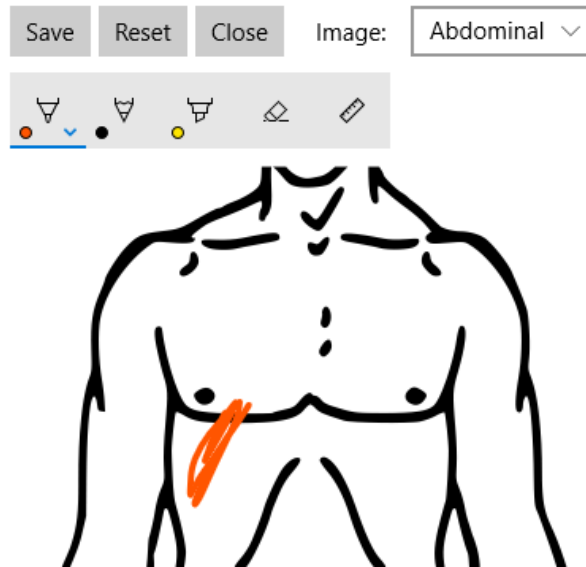
Figure 25: Drawing instructions on the position image

image is shown in the HoloLens, the corresponding panel in the dashboard is switched on. The drawing can be hidden from the trainee by clicking the same button once again.
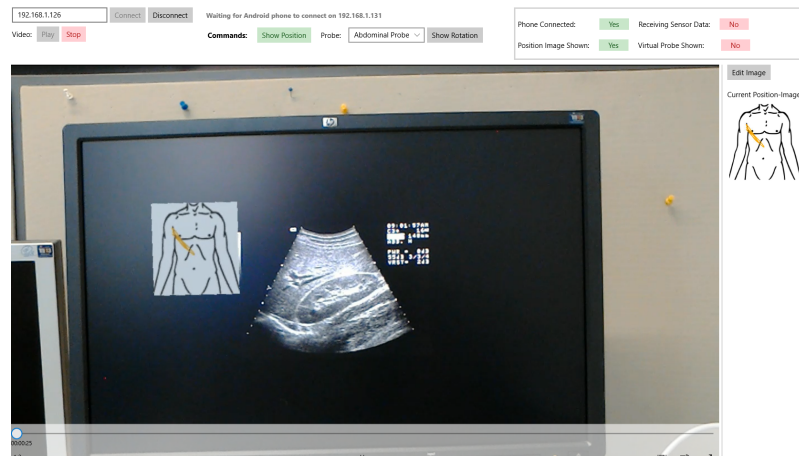

Figure 26: Finished position image shown displayed in the HoloLens

#5 Indication of the probe angle:

The last, and at the same time, most important feature of this application is the indication of the probe angle. Ultrasound procedures require very precise movements to acquire the desired result. A shift in the scanning angle by as little as 10 degrees may result in completely different images and ultimately lead to overlooked diseases. For this reason, this Proof-Of-Concept aims to facilitate the remote guidance for this matter by displaying a virtual ultrasound probe in front of the trainee. Since examiners, after the probe is placed on the body, are mainly viewing the screen and not to the patient, the virtual object is floating peripherally in the field-of-view.

To make this object visible in the HoloLens, the expert needs to trigger the button labeled "Show Rotation". To move the probe the built-in rotation sensors from the Android phone are used, which makes the solution elegant and affordable at the same time.

The virtual object rotates in the same way as the phone itself, as long as the data transfer from the Android application is enabled. The dashboard indicates if the application is currently receiving data. Finally, the virtual probe can be frozen in the current position by pressing the "stop" button on the phone. If the "Show Rotation" event from the GUI is triggered once again, the probe will be hidden and the rotation is reset.

Figure 27: Virtual abdominal probe displayed in the HoloLens

#6 Switching between probes:

To facilitate the understanding of the mechanisms described above, an additional feature was implemented. A drop-down field is located next to the button which enables the virtual probe. By default the virtual object is shaped like a broad probe which is mainly used for abdominal scans. For other regions that require a smaller and more precise field of operation, for example cervical arteries or the heart, a narrow object can be selected. To allow the exchange to take effect, the probe needs to be hidden for the trainee first. The newly selected object will be visible from the next activation. The different models are presented in figure 29

MBIRemote Android App   One of the standout values added by this new remote consultation service is the ability to provide the input for the virtual objects by simply moving a standard smartphone. Since iOS-based phones do not provide easy access to its orientation sensors, the decision was made to develop an Android application instead. This section explains the technical and mathematical fundamentals used for the application and outlines all different iterations that emerged out of the thought process during development.

Introduction to Android sensors: Every up-to-date Android phone, developed and shipped over the past two or three years, comes with many different built-in sensors. There are three basic categories of supported sensors: (Android Developers, 2017)

1. *Motion sensors:* Measure acceleration and rotation forces.
2. *Environmental sensors:* Meter environmental parameters like temperature or air pressure.
3. *Position sensors:* Give physical position data, for example based on the magnetic field.

For this PoC mainly the motion and position sensors are of further interest. The "SensorManager" framework provides a central access point to all kinds

of sensor data. It defines various hardware and software sensor interfaces that can be registered for use in the app. All active sensors provide data at steady intervals, depending on the defined settings and used hardware. The following specific sensors were used throughout this project: (Android Developers, 2017)

- *Accelerometer:* Measures the acceleration force in m/s² on all three physical axes, including gravity.

- *Linear Acceleration:* Software sensor which basically provides the same data as the accelerometer, but <u>without</u> gravity.

- *Gyroscope:* Measures the device's rate of rotation around all three axes in rad/s.

- *Magnetometer:* Measures the ambient geomagnetic field for all physical axes in μT.

- *Rotation Vectors:* Software fused sensors that combine accelerometer, gyroscope and/or magnetometer data to provide the device's orientation on all physical axes relative to a starting angle.

All sensors provide different strengths and weaknesses and their data can also be combined in many ways (see implementation details below).

Iterations in the development process: As already mentioned earlier, the application was re-engineered a couple of times because of its complex nature. The subsequent descriptions trace the evolution from the first idea to a final product.

Iteration 1: Linear Movement of the device
The first iteration was based on the idea to move a virtual probe directly on a patient's body by transmitting the linear movement of the phone on all three axes (x, y, z) to the HoloLens.
This problem turned out to be a very hard one, since that all suitable Android sensors for this problem are only providing acceleration data at frequent intervals. To calculate the traveled distance over a certain portion

of time one can use the following relations:

$$v(t) = \int a(t)dt$$

$$s(t) = \int v(t)dt$$

$$s(t) = \int\int a(t)dt$$

Velocity is the indefinite integral of an acceleration function over time. Furthermore, displacement is the indefinite integral of a velocity function over time. This means that displacement can be calculated as the second indefinite integral of an acceleration function over time.

Applying this relationship to the actual problem means to determine the current speed at time t by integrating the data points obtained from the accelerometer. The current displacement of the device at time t can subsequently be calculated by using "forward propagation". This means integrating the previously calculated speed over time and adding the resulting change to the last position:

$$s(t+1) = s(t) + \int v(t)dt$$

Since the acceleration measurements itself contain small errors, integrating these values twice results in huge deviations (around 30% after 2 seconds), even when applying sophisticated filters to the data. In other terms, this means that when moving the device 15cm in one direction, the calculations are up to 5cm off and the complete approach can not be realized.

Iteration 2: Linear Movement by tilt
To remove complexity the second idea was to move the probe around by tilting the phone. This means that as long as the phone remains in a tilted state, the virtual object moves in the specific direction until it moves back into the starting position.

In contrast to actual linear movement of the device, it is much easier to obtain far more reliable results with tilt, because it provides some significant simplifications of the model presented above. On the one hand, this is due

to the condition that the virtual object is assumed to be moving at constant speed. Therefore it is not necessary to integrate the acceleration data to calculate the displacement. On the other hand this approach reduces the complexity to two axes (x and y), because tilt is not available around the z-axis.

The tricky part of the implementation was to detect constant tilt forces with the accelerometer. When the phone is held at the same angle the acceleration on the axis converges to zero, because it is not actually moving. By combining accelerometer and gyroscope information, the rotation differences over time may serve as reference frame.

After collecting feedback for this idea from a medical expert it turned out that it was not usable in practical settings, because the minimum distance for HoloLens projections is 80cm and ultrasound scans often take place far closer to the patient.

Iteration 3: Rotation

Finally, transmitting only the device's rotation to the virtual probe to indicate the angle surfaced as the most feasible solution. This can be achieved by using the gyroscope angular velocities as a basis to get the actual rotation values around the axes in radians. The gyroscope itself however has some disadvantages too. First of all it has no idea where it is in relation to the world, so it is dependent on a calibration step to determine a starting point. Second of all, it drifts over time by a couple degrees due to the calculation errors of the integration process described in the first iteration.

The accelerometer on the other hand is affected by gravity and therefore well-suited to determine the long-term orientation of the device. The noise on fast movements however prevents accurate data over short periods of time.

The logical solution is to *fuse* both sensors to cancel their respective disadvantages out. The rotation vector sensor provides such a fusion by implementing a **complementary filter**. The name is derived from the principle of using a low-pass and a high-pass filter as a complement to each other.

- *High-pass filter:* Applied to the gyroscope data. Values <u>below</u> a specific threshold are discarded. This provides accurate short-term rotation changes.

- *Low-pass filter:* Applied to the accelerometer data. Values <u>above</u> a specific threshold are discarded. This provides accurate long-term position data and corrects the gyroscope drift.

Mathematically the complementary filter is expressed as a weighted sum of both data streams

$$\theta_{t+1} = k * (\theta_t + gyr) + (1 - k) * acc$$

where k is the weighting factor that represents the high-pass and (1-k) is the factor that represents the low-pass filter. Figure X shows the effect of a complementary filter on specific measurements.
The resulting data stream from the rotation vector sensor is then passed through a custom median filter to smooth the motions. Rotation updates are transfered over the data socket to the desktop application only every 100ms to prevent massive overhead.

The final application: The final application provides a single screen where all actions take place. It includes the following features:

- *Connect:* After starting the app, the user is asked to enter the IP-address of the PC running the MBIRemote desktop app, which can be obtained from the GUI itself (see figure 28). It is important to mention that the phone needs to be connected to the same network as the PC. By clicking the "Connect" button, an attempt to open up the socket is made. If unsuccessful, an according message is displayed.

- *Calibration:* The calibration process of all sensors takes place directly after the connection was successfully established. A message alerts the user to not move the phone during the process.

- *Start the data transfer:* The user is able to kick-off the data transfer over the socket by pressing the "Start" button, which is prominently located on the screen. The background process includes:

  1. Properly register all sensors

  2. Process and filter the data

  3. Display the current rotation on the screen
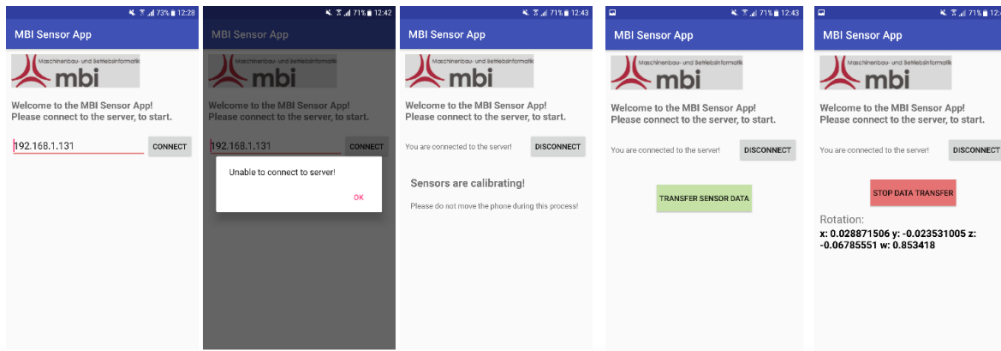
  4. JSON Serialization

Figure 28: Android MBIRemote App User Interface

     5. Transmission over the stream socket

- *Stop the data transfer:* After the data transmission has started, the "Start" button is switched to "Stop". A click stops the background process described above. The sensors however remain calibrated.

- *Disconnect:* The disconnect action terminates all background tasks, resets the calibration data and closes the socket.

Figure 28 shows all possible states of the user interface.

4.2.3 Trainee Side

This section focuses on the receiving end of the service, which is used by the trainee during an ultrasound scan. It describes all parts of the HoloLens application and how they are connected to the applications used by the expert (see section 4.2.2).

HoloLens Application   The operating system running on the Microsoft HoloLens is a very slimmed version of Windows 10 and therefore requires UWP apps. Using common .NET / C# and the UI frameworks described in section 4.2.2 is not an option, because they do not support 3D object rendering which is absolutely necessary for HoloLens applications.

Unity3D (Unity Technologies SF, 2017) is a runtime and development environment specialized in 3D games that provides all necessary tools for HoloLens development. It's core functionalities include 3D object manipulation and rendering combined with C# based scripting. All projects can be compiled to UWP solutions and are therefore perfectly compatible with the HoloLens OS.

The "HoloRemote" project implemented for this PoC consists of multiple modules:

- 3D Objects
    - Virtual Probes
    - Image Display

- Scripts
    - Stream Manager
    - Socket Manager
    - Position Manager
    - Rotation Manager

3D Objects: To model HoloLens enviroments, a standard Unity scene set up with a light source and a adjusted camera is needed. The camera object represents the physical HoloLens depth camera and moves with the user. All 3D objects are "GameObjects" initially placed in front of the virtual camera lense.

Every GameObject can be composed from various components, but at least require a position and rotation component. These empty GameObjects are mainly used as containers to group multiple objects together. Other possible modules include meshes and materials (for visual 3D objects), C# scripts or colliders. To model the 3D objects used in this project, only two different types of GameObjects were needed:

- *Cube (Image display):*
  A standard cube was used to display images. The received image data is loaded as custom texture which is assigned to the cube.

- *Custom 3D object (Probes):*
  Both types of probes (heart and abdominal) are based on the same 3D model file, which is loaded as custom mesh for their respective GameObjects. To achieve the desired result, both elements are grouped

together and scaled in a different way. The associated script (see "Rotation Manager" in the next paragraph) is assigned to the outside container.
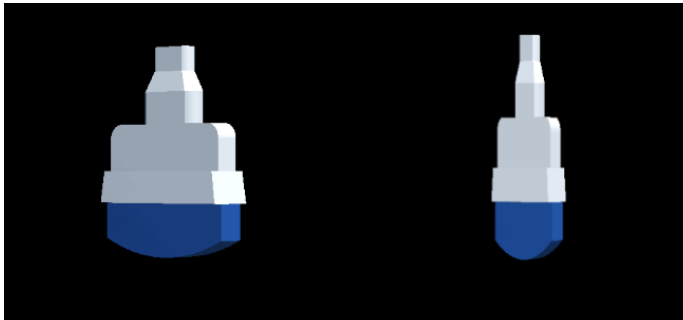
The final objects can be seen in figure 29.

Figure 29: 3D Models of ultrasound probes

Scripts: Initially all Unity GameObjects are static and not interacting with the scene. To add functionality multiple scripts based on the Unity "MonoBehaviour" class can be attached to each object. *MonoBehaviour* implements lifecycle methods that are automatically called by the game engine at the according time, for example every frame. Custom methods can be called from these lifecycle events or manually from other scripts. To realize the MBIRemote service, four different scripts were implemented:

- *Stream Manager:*
  This script, as the name suggests, manages the live stream from the HoloLens to the desktop application. It acts as intermediary between the MRVC plugin from Microsoft and the HoloLens hardware modules. First a Listener that waits for a peer connection is created and hooked up with different event handlers. On a successful connection, the capture engine is intialized and the stream is started. Finally, after the the connection is closed, all allocated resources are freed and the listener is re-started.

- *Socket Manager:*
  The socket manager handles all additional data transfer necessary for the service. Together with basic networking functionalities a handler is implemented to delegate received image data, position data and rotation information to other scripts. This is realized by keeping object references to all instantiated GameObjects and passing all data to appropriate functions from the attached scripts. Additionally, this script also implements helper classes to transform the collected JSON messages to C# objects (see section 4.2.1).

- *Position Manager:*
  This script is attached to the "Image Container" GameObject described above. It provides functionality to position the cube and to load custom textures from the received image data, which are base64 encoded strings. The positioning algorithm is based on the current gaze of the user by accessing position vectors of the main camera. In order to prevent occlusion of the user's field-of-view, the object is placed around 1,5 meters in forward direction (z axis) and 30 centimeters to the left side (x axis), relative to the camera's point-of-view (the Unity coordinate system is described below).

- *Rotation Manager:*
  The rotation manager is used to control the virtual probes. It is attached to the container wrapping both GameObjects, which prevents code duplicates. The first part of the provided functionality is the selection of the current probe. Since both objects are initially hidden and only one probe can be display at the same time, the probe currently selected by the user is positioned and shown. The positioning works quite similar to the process developed for the "Image Container", except the probes are shifted 30 centimeters towards the right side on the x axis.
  Applying the actual rotation angle to the selected probe is the core element of this script. Unity and Android use *quaternions* to describe rotation, because in contrast to euler angles they do not suffer from gimbal lock. Gimbal lock occurs when two axis are driven into a parallel configuration leading to the loss of one dimension of freedom, which would happen frequently with 6DOF input devices.
  Unfortunately Android and Unity have different definitions of their

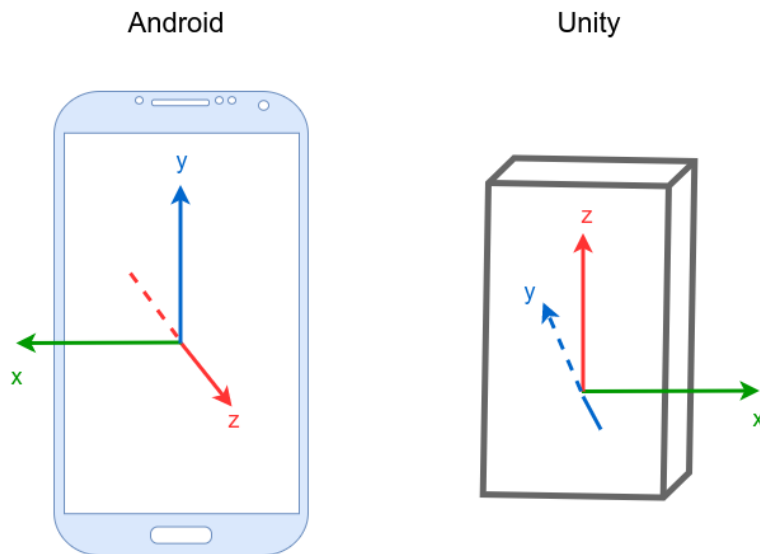Android   Unity

Figure 30: Comparison of Android and Unity coordinate systems

coordinate systems, one being left-handed and the other one being right-handed (as can be seen in figure 30). To correctly carry over the rotation information, all values needed to be inverted and the z and y axis needed to be swapped.

Additionally, a special requirement was to make sure that the probe is always directly facing the camera, regardless of head movements, to allow precise guidance from the expert's point-of-view. The HoloLens world space is designed to allow 360 degree views of objects, therefore it was necessary to compensate movements of the user. The requirement was fulfilled by updating the objects position and rotation on every frame with respect to the current position of the main camera.

4.3 Testing and Evaluation

To ensure that the high quality standards defined for this project are met at all times, evaluation and testing were very important aspects throughout the

whole process. This section describes the different types of tests conducted to evaluate the PoC and highlights all results that emerged from them.

Spiral Development and Continous Evaluation: The concept of spiral development served as the foundation of the whole implementation process for the PoC. Every development cycle involves four basic steps: (Boehm, 1988)

1. Definition of objectives and requirements for the module under development

2. Evaluation of alternatives and risk analysis

3. Implementation and testing of the intermediary product

4. Planning of the next cycle (module)

Every application was periodically reviewed by members of the institute and if necessary, in close cooperation with external domain experts. All evaluated aspects, the according test results and derived design changes (some of them already mentioned in sections 4.2.2 and 4.2.3) are documented below:

Stream Quality: The first prototype involving a live streaming service was realized with WebRTC. The following criteria was evaluated (ordered by importance):

- Fusion and positioning of holograms in the video stream
- Latency
- Stability
- Video Quality
- Power consumption

Results: WebRTC performed well with power consumption and stability, but had issues regarding latency and especially the hologram fusion. As a consequence the design was changed to rely on MRVC, which provided good results in those areas. Power consumption and stability emerged as small issues and required temporary solutions like external power banks.

Position Accuracy: The initial design included the possibility to indicate the trajectory of a probe by moving it directly along the patients body. Therefore *sensor accuracy and translation of linear acceleration to displacement* were tested.

Results: The sensor accuracy as well as the displacement calculations were not accurate enough to provide the expected results (error margin up to 80%). Additionally, the HoloLens does not provide the ability to project holograms at a closer distance than 80 centimeters and the domain experts pointed out that doctors performing the procedure are not looking at the patient most of the time. As a result, the design was changed to indicate the position of the probe by drawing on body outlines that are displayed to the trainee.

Rotation Accuracy: Similar test cases as for positioning were conducted for the rotation and angle functionalities. Since Android offers multiple hardware and software sensors, evaluating sensor accuracies was the main goal of this tests. Likewise the correct conversion to another coordinate system and the compensation algorithm for head movements were evaluated periodically.

Results:

- *Sensor accuracy*
    - *Gyroscope:* Good short-term accuracy, drifting over time (around 15-20 degrees after 30s)
    - *Accelerometer:* Sufficient long-term accuracy, weak on fast changes (around 30% error)
    - *Magnetic Rotation Vector:* Combination of gyro, accelerometer and magnetometer. Sufficient accuracy, but needs frequent calibration and prone to magnetic disturbances
    - *Rotation Vector:* Combination of gyro and accelerometer. Most accurate and sufficient stability.
- *Coordinate system conversion:* Using rotation matrices and euler angles resulted in inaccurate data, gimbal lock and rough motions including sudden jumps in angles. Quaternions provided smoother movements and accurate conversions.

- *Head-movement compensation:* Combining the received rotation input and the current transform of the main camera resulted in over-compensation and a drift of about 25 to 30 degrees. Re-setting the object rotation to face the camera first and applying the new angle afterwards reduced the drift to a maximum of 5 degrees, which was classified as sufficient by the domain experts.

Evaluation of the final product In additional to the continuous testing cycles described above, two expert sessions to evaluate the finished product and the theoretical models from section 4.1 were conducted. This test cases were not focused on technical details, but rather on usability and subjective perception. The interviewed domain experts were:

- **Expert A:** Experienced surgeon from a local hospital. Well-trained in using new technologies.
- **Expert B:** General practitioner located near the city of Graz. No experience with AR applications prior to the session.

Both sessions were planned and executed in the exact same way. First the expert were introduced to the service and asked to interact not only with separate modules but with the system as a whole and from both perspectives (expert and trainee). A third person served as counterpart for the procedure. Following the evaluation each expert was asked for his overall impression and suggestions for improvement of the service. Finally, the experts were presented the prepared Value Proposition Canvas and Value Network and were asked for feedback.

Results: In general, both experts offered mainly positive feedback after using the whole system. Expert A highlighted the practical relevance and proposed other possible use cases for further research. Expert B explicitly emphasized the easy and intuitive handling of the system. Both experts agreed that IT support would be necessary if the system would be deployed in real-life settings.
The following list documents detailed feedback expressed during the sessions:

- **User Interface:** Both experts classified the user interfaces as easy to understand and appealing. Expert A suggested the implementation of a dashboard to monitor the current state of the application and the inclusion of different body outlines to choose from.

- **Live Stream:** According to Expert A the quality of the live stream was sufficient. Expert B noted a small offset between the image and the actual field-of-view of the user. It was also mentioned that a continuous capture of the physical probe could be helpful to determine the initial angle.

- **Input Accuracy:** Expert A noticed an offset while operating with the Android smartphone. Based on this feedback the compensation algorithm was revised and adapted for the next session. Expert B confirmed the improvements and stated that the input accuracy is sufficient for ultrasound procedures.

- **Wearing Comfort:** While Expert A did not mention any discomfort wearing the HoloLens, Expert B experienced some difficulties adjusting the HMD correctly.

Regarding the evaluation of the prepared models both experts provided additional insights and inputs, but confirmed the overall correctness of the assumptions made for both VPC and Value Network. The detailed feedback can be found in section 4.1.

5 Conclusion and Outlook

Augmented Reality is a fast developing technology set to become a sustainable part of workflows across various fields in the near future. With research and technological progress advancing, more and more possible use cases and applications emerge. The primary goal of this thesis was to investigate the technology as well as existing and possible future use cases and to analyze the possibility to utilize this knowledge in service design.

In the theoretical part a study was conducted to give a comprehensive overview over the current State-of-the-Art in Augmented Reality. First of all technical basics were examined with different kinds of displays as a starting point. Both optical and video approaches proved to be excellent solutions depending on the area of application. The next part dealt with different types of display positioning and worked out head-mounted displays as well as hand-held or spatial mobile devices as most used solutions. Furthermore, different input methods for AR devices were presented with "six degrees of freedom" pointing, gaze tracking, gesture recognition and voice input as the most important ones. Last but not least challenges like sensor-based or visual tracking and the registration problem were pointed out and possible solutions were outlined.

The following section centered around different use cases already addressed by research groups or industrial companies. As a first large area of research the medical field was determined. Prototypes to project patient data like medical records, ultrasound images, X-ray images or vital parameters directly onto the human body as additional information source for different procedures proved to be very effective in various studies. Remote consultation on the other hand emerged as a topic of active research, which attempts to help inexperienced doctors or surgeons working for example in crisis regions to perform a procedure. A domain expert could help remotely

by providing visual instructions directly into the field-of-view. Industrial applications were found as another interesting area for Augmented Reality. Visual instructions for maintenance or assembly tasks and virtual prototypes in product planning are among the most researched topics. Other fields of application include tourism and navigation, entertainment and advertisement as well as education.

The last part of the AR study focused on available hardware and software. A comparison between different head-mounted displays regarding price and features was provided and the most important open source software toolkits were described.

Finally, the theoretical background of different models and approaches used in service design was elaborated. At first the basic principles of "Service Design Thinking" and their application in an iterative approach were described. Secondly important auxiliary tools for successfully designing a new service were presented. The UCTM framework, requirements engineering, the Value Proposition Canvas and the Value Network Notation are notable representatives of these methodologies.

For the practical part of this thesis a case study with the goal to design a new service with Augmented Reality technologies was implemented. In close cooperation with experts from a local hospital in the city of Graz a proof-of-concept for remote ultrasound consultation was developed. The final product consists of a desktop utility and an Android application (as 6DOF input device) for an expert as well as a visual application designed for the Microsoft HoloLens.

During the development process the UCTM framework, the Value Proposition Canvas and the Value Network Notation were successfully applied and validated by designated domain experts. The service itself was evaluated and tested continuously throughout the whole process and two expert feedback sessions were conducted to collect feedback regarding the finalized product.

Although the proof-of-concept was overall received very well, some notable limitations and potentials for improvement surfaced during the evaluation process:

- The wearing comfort of the HoloLens is not sufficient for long-lasting procedures.

- The possible area of projection of holograms is limited using the HoloLens, for example objects need to be placed at least 80cm away from the user.

- Live streaming drains a lot of power and therefore may be interrupted by the HoloLens' internal power manager. Also a stable and fast internet connection is required to maintain a low latency.

- Android's positioning and rotation sensor accuracy is sufficient for ultrasound procedures, but may present a error too large for high precision tasks.

- The prototype was only tested by a small number of people and was not evaluated in real-world scenarios.

Looking at the limitations stated above, some areas for future work can be derived. Evaluating the service in a real-world setting with a larger number of participants is clearly the most important task. Another possible improvement would be the migration of the desktop application to a mobile user interface to further increase the portability of the service, but requires the new mobile network standard 5G. Last but not least, the service could be extended to other areas. Examples include remote consultation for other medical procedures like endoscopies or a virtual guiding tool for medical conferences, which would eliminate the need to fly in remote experts.

Finally, it is important to state that Augmented Reality has the potential to change the way of working as it is known at this time, but it's future is highly dependent on further advances in hardware and infrastructure. Cheaper AR technologies with lighter displays and a longer battery lifetime as well as faster and more stable network connections are necessary for the technology finally succeeding in the mass-market.

Bibliography

31 volts service design (2008). *One line of service design by Marc Fonteijn*. URL: http://www.31volts.com/2008/03/one-line-of-service-design/ (visited on 11/14/2017).

Alphabet Inc. (2017). *Google Tango*. URL: https://developers.google.com/tango/ (visited on 11/14/2017).

Andersen, Daniel et al. (2016). "Virtual annotations of the surgical field through an augmented reality transparent display". In: *The Visual Computer* 32.11, pp. 1481–1498.

Android Developers (2017). *Sensors Overview*. URL: https://developer.android.com/guide/topics/sensors/sensors_overview.html (visited on 12/04/2017).

Andujar, Jose Manuel, Andres Mejias, and Marco Antonio Marquez (2011). "Augmented reality for the improvement of remote laboratories: an augmented remote laboratory". In: *IEEE transactions on education* 54.3, pp. 492–500.

Anton, David et al. (2017). "Augmented Telemedicine Platform for Real-Time Remote Medical Consultation". In: *International Conference on Multimedia Modeling*. Springer, pp. 77–89.

Apple Inc. (2017). *Apple IPad*. URL: https://www.apple.com/ipad-9.7/ (visited on 11/07/2017).

Azpiazu, J et al. (2011). "Remote support for maintenance tasks by the use of Augmented Reality: the ManuVAR project". In: *CARVI 2011: IX Congress on virtual reality applications*.

Azuma, Ronald T (1997). "A survey of augmented reality". In: *Presence: Teleoperators and virtual environments* 6.4, pp. 355–385.

Azuma, Ronald et al. (2001). "Recent advances in augmented reality". In: *IEEE computer graphics and applications* 21.6, pp. 34–47.

Bibliography

Bauer, Martin et al. (2001). "Design of a component-based augmented reality framework". In: *Augmented Reality, 2001. Proceedings. IEEE and ACM International Symposium on*. IEEE, pp. 45–54.

Biem, Alain and Nathan Caswell (2008). "A value network model for strategic analysis". In: *Hawaii International Conference on System Sciences, Proceedings of the 41st Annual*. IEEE, pp. 361–361.

Bley, H et al. (2004). "Appropriate human involvement in assembly and disassembly". In: *CIRP Annals-Manufacturing Technology* 53.2, pp. 487–509.

Boehm, Barry W. (1988). "A spiral model of software development and enhancement". In: *Computer* 21.5, pp. 61–72.

Chong, JWS et al. (2006). "Robot programming using augmented reality (RPAR): an interactive methodology". In: *2006 Singapore-MIT Alliance Symposium*.

Curtis, Dan et al. (1999). "Several devils in the details: making an AR application work in the airplane factory". In: *Proceedings of the international workshop on Augmented reality: placing artificial objects in real scenes: placing artificial objects in real scenes*. AK Peters, Ltd., pp. 47–60.

Dahne, Patrick and John N Karigiannis (2002). "Archeoguide: System architecture of a mobile outdoor augmented reality system". In: *Mixed and Augmented Reality, 2002. ISMAR 2002. Proceedings. International Symposium on*. IEEE, pp. 263–264.

Dalle Mura, Michela, Gino Dini, and Franco Failli (2016). "An integrated environment based on augmented reality and sensing device for manual assembly workstations". In: *Procedia CIRP* 41, pp. 340–345.

Dangelmaier, Wilhelm et al. (2005). "Virtual and augmented reality support for discrete manufacturing system simulation". In: *Computers in Industry* 56.4, pp. 371–383.

Davis, Matthew Christopher et al. (2016). "Virtual interactive presence in global surgical education: international collaboration through augmented reality". In: *World neurosurgery* 86, pp. 103–111.

Epson Inc. (2015). *Epson Moverio Pro BT-2000*. URL: https://epson.com/moverio-augmented-reality-headsets-industrial-applications (visited on 11/13/2017).

Fischler, Martin A and Robert C Bolles (1981). "Random sample consensus: a paradigm for model fitting with applications to image analysis and

automated cartography". In: *Communications of the ACM* 24.6, pp. 381–395.

Föckler, Paul et al. (2005). "PhoneGuide: museum guidance supported by on-device object recognition on mobile phones". In: *Proceedings of the 4th international conference on Mobile and ubiquitous multimedia*. ACM, pp. 3–10.

Freitas, Rubina and Pedro Campos (2008). "SMART: a SysteM of Augmented Reality for Teaching 2 nd grade students". In: *Proceedings of the 22nd British HCI Group Annual Conference on People and Computers: Culture, Creativity, Interaction-Volume 2*. BCS Learning & Development Ltd., pp. 27–30.

Friedrich, Wolfgang, D Jahn, and L Schmidt (2002). "ARVIKA-Augmented Reality for Development, Production and Service." In: *ISMAR*. Vol. 2002, pp. 3–4.

Frund, J et al. (2004). "Cooperative design support within automobile advance development using augmented reality technology". In: *Computer Supported Cooperative Work in Design, 2004. Proceedings. The 8th International Conference on*. Vol. 2. IEEE, pp. 492–497.

GlassUp (2017). *GlassUp F4*. URL: http://www.glassup.com/en/f4/ (visited on 11/13/2017).

Google Inc. (2013). *Google Glass*. URL: https://developers.google.com/glass/ (visited on 10/31/2017).

Höllerer, Tobias et al. (2001). "User interface management techniques for collaborative mobile augmented reality". In: *Computers & Graphics* 25.5, pp. 799–810.

Holloway, Richard Lee (1995). "Registration errors in augmented reality systems". PhD thesis. University of North Carolina at Chapel Hill.

Huang, Zhanpeng et al. (2013). "Mobile augmented reality survey: a bottom-up approach". In: *arXiv preprint arXiv:1309.4413*.

Ibanez, Maria-Blanca, Antonio J de Castro, and Carlos Delgado Kloos (2017). "An Empirical Study of the Use of an Augmented Reality Simulator in a Face-to-Face Physics Course". In: *Advanced Learning Technologies (ICALT), 2017 IEEE 17th International Conference on*. IEEE, pp. 469–471.

Instagram Inc. (2017). *Instagram*. URL: https://www.instagram.com/ (visited on 11/09/2017).

Kato, Hirokazu and Mark Billinghurst (1999). "Marker tracking and hmd calibration for a video-based augmented reality conferencing system".

In: *Augmented Reality, 1999.(IWAR'99) Proceedings. 2nd IEEE and ACM International Workshop on*. IEEE, pp. 85–94.

Kyhnau, Jan and Christian Nielsen (2015). "Review Of: Value Proposition Design". In: *Journal of Business Models* 3.1, pp. 81–92.

Lawson, Shaun W and John RG Pretlove (1998). "Augmented reality for underground pipe inspection and maintenance". In: *Photonics East (ISAM, VVDC, IEMB)*. International Society for Optics and Photonics, pp. 98–104.

Lee, Woohun and Jun Park (2005). "Augmented foam: A tangible augmented reality for product design". In: *Mixed and Augmented Reality, 2005. Proceedings. Fourth IEEE and ACM International Symposium on*. IEEE, pp. 106–109.

MacIntyre, Blair et al. (2003). "DART: The designer's augmented reality toolkit". In: *Mixed and Augmented Reality, 2003. Proceedings. The Second IEEE and ACM International Symposium on*. IEEE, pp. 329–330.

Mager, Birgit (2006). "Service design basics". In: *Cologne: Kcln International School of Design*.

Massie, Thomas H, J Kenneth Salisbury, et al. (1994). "The phantom haptic interface: A device for probing virtual objects". In: *Proceedings of the ASME winter annual meeting, symposium on haptic interfaces for virtual environment and teleoperator systems*. Vol. 55. 1. Chicago, IL, pp. 295–300.

Medlock-Walton, Michael Paul (2012). "TaleBlazer: a platform for creating multiplayer location based games". PhD thesis. Massachusetts Institute of Technology.

Metavision Inc. (2017). *Meta 2*. URL: https://buy.metavision.com/#specifications (visited on 11/13/2017).

Microsoft Inc. (2016a). *HoloLens Hardware Details*. URL: https://developer.microsoft.com/en-us/windows/mixed-reality/hololens_hardware_details (visited on 02/23/2018).

Microsoft Inc. (2016b). *Microsoft HoloLens*. URL: https://www.microsoft.com/en-us/hololens (visited on 10/31/2017).

Microsoft Inc. (2017a). *About Media Foundation*. URL: https://msdn.microsoft.com/en-us/library/windows/desktop/ms696274(v=vs.85).aspx (visited on 11/11/2017).

Microsoft Inc. (2017b). *Microsoft AR Toolkits*. URL: https://github.com/Microsoft/ (visited on 11/14/2017).

Microsoft Inc. (2017c). *Mixed Remote View Compositor*. URL: https://github.
com/Microsoft/MixedRealityCompanionKit/tree/master/MixedRemoteViewCompositor
(visited on 11/10/2017).

Moritz, S (2005). *Service Design. A practical access to an evolving field. Viitattu
10.4. 2014.*

Morrison, Ann et al. (2009). "Like bees around the hive: a comparative
study of a mobile augmented reality map". In: *Proceedings of the SIGCHI
Conference on Human Factors in Computing Systems.* ACM, pp. 1889–1898.

Najjar, Lawrence J, J Christopher Thompson, and Jennifer J Ockerman
(1997). "A wearable computer for quality assurance inspectors in a food
processing plant". In: *Wearable Computers, 1997. Digest of Papers., First
International Symposium on.* IEEE, pp. 163–164.

Narzt, Wolfgang et al. (2006). "Augmented reality navigation systems". In:
Universal Access in the Information Society 4.3, pp. 177–187.

Nascimento, Paula et al. (2012). "An approach to requirements categoriza-
tion using Kano's model and crowds". In: *Computer Supported Cooperative
Work in Design (CSCWD), 2012 IEEE 16th International Conference on.* IEEE,
pp. 387–392.

Neumann, Ulrich and Anthony Majoros (1998). "Cognitive, performance,
and systems issues for augmented reality applications in manufacturing
and maintenance". In: *Virtual Reality Annual International Symposium,
1998. Proceedings., IEEE 1998.* IEEE, pp. 4–11.

Ohbuchi, Ryutarou, Mike Bajura, and Henry Fuchs (1998). "Case study:
Observing a volume rendered fetus within a pregnant patient". In:
Visualization: Proceedings of the IEEE Conference on Visualization. Vol. 5.
IEEE Computer Society Press.

Olsson, Thomas et al. (2013). "Expected user experience of mobile aug-
mented reality services: a user study in the context of shopping centres".
In: *Personal and ubiquitous computing* 17.2, pp. 287–304.

Ong, SK, Yu Pang, and AYC Nee (2007). "Augmented reality aided assembly
design and planning". In: *CIRP Annals-Manufacturing Technology* 56.1,
pp. 49–52.

Ong, SK, ML Yuan, and AYC Nee (2008). "Augmented reality applications
in manufacturing: a survey". In: *International journal of production research*
46.10, pp. 2707–2742.

Osterhout Design Group Inc. (2016). *ODG R7.* URL: https://www.osterhoutgroup.
com/r-7-glasses-system.html (visited on 11/10/2017).

Osterwalder, Alexander and Yves Pigneur (2010). *Business model generation: a handbook for visionaries, game changers, and challengers*. John Wiley & Sons.

Osterwalder, Alexander, Yves Pigneur, et al. (2014). *Value proposition design: How to create products and services customers want*. John Wiley & Sons.

Persiani, F, A Liverani, and F De Crescenzio (2001). "Augmented Reality approach to vehicle prototyping". In: *IIEC, Amman, Jordan*.

Ponce, Brent A et al. (2014). "Emerging technology in surgical education: combining real-time augmented reality and wearable computing devices". In: *Orthopedics* 37.11, pp. 751–757.

PTC Inc. (2017). *Vuforia AR SDK*. URL: https://library.vuforia.com/ (visited on 11/14/2017).

Recon Instruments Inc. (2017). *Recon Jet Pro*. URL: https://www.reconinstruments.com/enterprise/jet-pro/ (visited on 11/13/2017).

Reiners, Dirk et al. (1998). "Augmented reality for construction tasks: Door-lock assembly". In: *Proc. IEEE and ACM IWAR* 98.1, pp. 31–46.

Reinhart, G and C Patron (2003). "Integrating augmented reality in the assembly domain-fundamentals, benefits and applications". In: *CIRP Annals-Manufacturing Technology* 52.1, pp. 5–8.

Reitmayr, Gerhard and Dieter Schmalstieg (2001). "Opentracker-an open software architecture for reconfigurable tracking based on xml". In: *Virtual Reality, 2001. Proceedings. IEEE*. IEEE, pp. 285–286.

Renevier, Philippe et al. (2005). "Generic interaction techniques for mobile collaborative mixed systems". In: *Computer-Aided Design of User Interfaces IV*, pp. 309–322.

Rublee, Ethan et al. (2011). "ORB: An efficient alternative to SIFT or SURF". In: *Computer Vision (ICCV), 2011 IEEE international conference on*. IEEE, pp. 2564–2571.

Rupp, Chris, Matthias Simon, and Florian Hocker (2009). "Requirements engineering und management". In: *HMD Praxis der Wirtschaftsinformatik* 46.3, pp. 94–103.

Salado, Alejandro and Roshanak Nilchiani (2014). "A Categorization Model of Requirements Based on Max-Neef's Model of Human Needs". In: *Systems Engineering* 17.3, pp. 348–360.

Schmalstieg, Dieter et al. (2002). "The studierstube augmented reality project". In: *Presence: Teleoperators and Virtual Environments* 11.1, pp. 33–54.

Schnabel, Marc Aurel et al. (2007). "From virtuality to reality and back". In: *Proceedings of the International Association of Societies of Design Research* 1, p. 15.

Schowengerdt, Brian T et al. (2003). "Binocular retinal scanning laser display with integrated focus cues for ocular accommodation". In: *Stereoscopic Displays and Virtual Reality Systems X. Proceedings of SPIE-IS&T Electronic Imaging. SPIE* 5006, pp. 1–9.

Sekhavat, Yoones A (2017). "Privacy preserving cloth try-on using mobile augmented reality". In: *IEEE Transactions on Multimedia* 19.5, pp. 1041–1049.

Shenai, Mahesh B et al. (2011). "Virtual interactive presence and augmented reality (VIPAR) for remote surgical assistance". In: *Operative Neurosurgery* 68.suppl_1, ons200–ons207.

Snap Inc. (2017). *Snapchat*. URL: https://www.snapchat.com/ (visited on 11/09/2017).

Stickdorn, Marc, Jakob Schneider, et al. (2011). *This is service design thinking: Basics, tools, cases*. Wiley Hoboken, NJ.

Stickdorn, Marc and Klaus Schwarzenberger (2016). "Service design in tourism". In: *Entrep. Tour. Unternehmerisches Denk. Erfolgskonzepte Aus Prax* 2261, pp. 835–848.

Sundareswaran, Venkataraman et al. (1999). "A Distributed System for Device Diagnostics Utilizing Augmented Reality, 3D Audio, and Speech Recognition". In: *Proceedings of HCI International (the 8th International Conference on Human-Computer Interaction) on Human-Computer Interaction: Ergonomics and User Interfaces-Volume I-Volume I*. L. Erlbaum Associates Inc., pp. 466–470.

The Pokémon Company International Inc. (2016). *Pokémon Go*. URL: https://www.pokemongo.com/en-us/ (visited on 11/09/2017).

Unity Technologies SF (2017). *Unity 3D*. URL: https://unity3d.com/ (visited on 11/14/2017).

Van Krevelen, DWF and Ronald Poelman (2010). "A survey of augmented reality technologies, applications and limitations". In: *International Journal of Virtual Reality* 9.2, p. 1.

Volante, Jackson W et al. (2017). "Augmented Reality Technician Assistance Program". In:

Volkswagen AG (2017). *Mobile Augmented Reality Technical Assistance (MARTA)*. URL: https://www.volkswagenag.com/en/group/research/virtual-technologies.html (visited on 11/09/2017).

Vorraber, Wolfgang (2016). *Including ethical aspects in business model- and service design*. Internal Paper of the Department of Engineering and Business Informatics at Graz University of Technology.

Vorraber, Wolfgang, Dietmar Neubacher, et al. (2018). "UCTM - An Ambidextrous Service Innovation Framework: A Bottom-up Approach to Combine Human- and Technology-centered Service Design". Internal Paper of the Department of Engineering and Business Informatics at Graz University of Technology.

Vorraber, Wolfgang and Siegfried Voessner (2011). "Modeling Endogenous Motivation and Exogenous Influences in Value Networks of Information Service Systems". In: *Journal of convergence information technology* 6.8, pp. 356–363.

Vorraber, Wolfgang, Siegfried Voessner, et al. (2014). "Medical applications of near-eye display devices: an exploratory study". In: *International Journal of Surgery* 12.12, pp. 1266–1272.

Vravana Inc. (2017). *Vravana Totem*. URL: https://vrvana.com/#specifications (visited on 11/13/2017).

Vroom, Victor H (2005). "On the origins of expectancy theory". In: *Great minds in management: The process of theory development*, pp. 239–258.

Vuzix Inc. (2016). *Vuzix M300*. URL: https://www.vuzix.com/Products/m300-smart-glasses (visited on 11/13/2017).

WebRTC (2017a). *WebRTC Basics and Overview*. URL: http://io13webrtc.appspot.com/#55 (visited on 11/09/2017).

WebRTC (2017b). *WebRTC real-time communication framework*. URL: https://webrtc.org (visited on 11/09/2017).

Yuan, ML, SK Ong, and AYC Nee (2008). "Augmented reality for assembly guidance using a virtual interactive tool". In: *International Journal of Production Research* 46.7, pp. 1745–1767.

Zauner, Jürgen et al. (2003). "Authoring of a mixed reality assembly instructor for hierarchical structures". In: *Proceedings of the 2nd IEEE/ACM International Symposium on Mixed and Augmented Reality*. IEEE Computer Society, p. 237.

Zhang, J, SK Ong, and AYC Nee (2006). "A volumetric model-based CNC simulation and monitoring system in augmented environments". In: *Cyberworlds, 2006. CW'06. International Conference on*. IEEE, pp. 33–42.

Zhang, Qian et al. (2017). "A Virtual Try-On System for Prescription Eyeglasses". In: *IEEE Computer Graphics and Applications* 37.4, pp. 84–93.

Zoellner, Michael et al. (2009). "An augmented reality presentation system for remote cultural heritage sites". In: *Proceedings of the 10th International Symposium on Virtual Reality, Archaeology and Cultural Heritage VAST*, pp. 112–116.