Augmented Reality-based Assistance and Automated Planning for Neurosurgical Interventions

zur Erlangung des akademischen Grades eines

Doktors der Ingenieurwissenschaften

von der KIT-Fakultät für Informatik
des Karlsruher Instituts für Technologie (KIT)

genehmigte
Dissertation

von

Christian-Peter Kunz
aus Hanau

Tag der mündlichen Prüfung: 21.10.2021

Erster Gutachter: Prof. Dr.-Ing. Heinz Wörn
Zweite Gutachterin: Jun.-Prof. Dr. Franziska Mathis-Ullrich
Dritter Gutachter: Prof. Dr. med. Christian Rainer Wirtz

Institute for Anthropomatics and Robotics (IAR) -
Health Robotics and Automation (HERA)
KIT Department of Informatics
Karlsruhe Institute of Technology
Engler-Bunte-Ring 8
76131 Karlsruhe

Acknowledgment

A large number of people have contributed to the success of this work. First and foremost, I would like to thank my supervisor Jun.-Prof. Dr. Franziska Mathis-Ullrich, who has contributed significantly to the success of this thesis. Big thanks also go to Prof. Dr. Heinz Wörn for his constant advice and help during my doctorate.

I would also like to thank the neurosurgeons of the University Hospital Ulm/Guenzburg for the pleasant cooperation and the work we achieved together. I would especially like to thank PD Dr. med. Michal Hlâvac, Dr. Max Schneider, PD MUDr. Dr. med. Andrej Pal'a and Prof. Dr. med. Christian Rainer Wirtz.

Special thanks goes to the best team I can think of from the Health Robotics and Automation Lab and its members, Paul Scheikl, Nikola Fischer, Christian Marzi and Balazs Gyenes. Furthermore, I would like to thank all the members of the Institute for Anthropomatics and Robotics (IAR) - Intelligent Process Automation and Robotics (IPR), who always provided constructive advice. I want to particularly mention Prof. Dr. Björn Hein, who was always available to answer questions and open up new perspectives to me.

A big thanks also goes to a number of undergraduates for their contributions, which are Birgit Jickeli, Paulina Maurer, Maximilian Gerst, Paul Maas, Vera Genten and Fabian Kees. It was real fun to work with you all and I am happy about the interesting solutions that were found. I also want to thank my two assistants Steffen Peikert and Pit Henrich for their great work. They have brought many new perspectives to my work and opened up new paths for me. The discussions with you were very inspiring.

A very special thanks goes to Jean-Marie Teikitohe, our workshop foreman. He was always available when something had to be constructed, be it test stands, phantoms or the catheter navigation aid. He was a great help during the time this work was written.

Finally, I would like to thank the great people around me who have supported me during the sometimes challenging time of this PhD. First and foremost, my partner Anni and my grandparents, without whom I would not have made it this far. To all my friends who have always been by my side.

Für Rolf und Marga und Ute und Anni

Abstract

Augmented Reality-based Assistance and Automated Planning for Neurosurgical Interventions

Neurosurgical interventions are very challenging due to the delicate structures of the human brain. The complete brain is composed of tissue which must be damaged as little as possible. Unfortunately, this is not always possible, as many surgeries require penetration of tissue to reach a target area. Examples include biopsies, the ventricular puncture, deep brain stimulation and tumour ablation. It is especially important to prevent high-risk areas from being injured, as this can lead directly to major neurological impairments, e.g. loss of speech. Optimized surgical planning can prevent high-risk areas from being injured in order to prevent neurological damage.

In this work, novel methods are presented that support the complete surgical workflow by performing fully automated surgical planning and subsequent scene registration to provide intraoperative assistance to the surgeon based on augmented reality (AR).

In a first step, the patient's image data is segmented to create patient models that accurately represent the anatomy. A segmentation accuracy of 98% (F1-score) is reached. The system uses statistical shape models in a consecutive stage to determine the catheter placement trajectories for the left and right ventricles completely autonomously. The Kocher's point is found in 98.4% of cases. The correct target point inside the ventricular system is obtained in 95.4%. This leads to an overall rate of correctly planned trajectories of 93.9%. This planning is enhanced by a patient specific risk map-based entry point adjustment to find the optimal risk-reduced path to a target.

Furthermore, methods are presented that enable robust and accurate tracking of the patient during a surgical procedure. Possible markers for robust patient tracking were investigated. As a result, Vuforia markers have been selected as they best meet the requirements. In addition, infrared marker tracking was implemented on the AR-glasses (HoloLens) to address the shortcomings of the relatively large and not freely usable two-dimensional Vuforia markers. The surgical requirements for sterilisability and high tracking accuracy are met by the infrared markers.

A marker system is presented that supports the transition from the non-sterile to the sterile surgical phase. The remounting accuracy is 0.18 ± 0.06 mm. Vuforia and infrared markers were attached to the marker system. Four registration methods are presented to register the marker system to the patient. Three of the methods support a user in manual registration to match the patient models as closely as possible to the real patient. A game controller, a pointer or gestures can be used to do so. The last method uses a multi-level surface matching approach to perform the registration automatically. The registration method using the game controller achieves the highest accuracy with a value of 2.71 ± 1.18 mm.

The generated patient models and the determined puncture trajectories are used to assist the surgeon intraoperatively by visualizing these hidden structures. The proposed methods increase the puncture accuracy of linear trajectories in relation to the optimal entry and target points. This can lead to a lower penetration rate of high-risk areas inside the brain and to a better outcome for the patient.

First, experiments after the standard of care were conducted. A mean error of 6.6 ± 3.1 mm and a success rate of 72.7% was achieved. This corresponds to the values communicated in the state of the art. In the first stage an AR guidance is provided to the user to assist during neurosurgical interventions. The mean error was determined to be 4.8 ± 2.5 mm and a 9 percent better success rate of 81.7% could be achieved. In a follow-up stage, the challenge of a well-chosen initial puncture angle was addressed by introducing a catheter navigation aid in combination with the AR system. The mean error was improved to 3.1 ± 1.8 mm and a success rate of 98% was reached. Still the challenge of a manual manipulation remained. To provide a more stable manipulation in space, the navigation aid was mounted on a robot that was telemanipulated with a game controller. The controller was used to set the correct angle. A mean error of 1.9 ± 1.2 mm was reached. All catheters could be placed correctly. In the successive stages, a higher accuracy and precision were achieved in each case.

Zusammenfassung

Augmented Reality-based Assistance and Automated Planning for Neurosurgical Interventions

Neurochirurgische Eingriffe sind aufgrund der empfindlichen Strukturen des menschlichen Gehirns eine große Herausforderung. Das gesamte Gehirn besteht aus sensiblem Gewebe, das so wenig wie möglich beeinträchtigt werden sollte. Dennoch ist dies nicht immer möglich, da es bei vielen Operation nötig ist, durch Gewebe hindurch zum Zielgebiet zu gelangen. Beispiele sind Biopsien, die Ventrikelpunktion, tiefe Hirnstimulation und Tumorentfernungen. Dabei muss speziell verhindert werden, dass Hochrisikobereiche verletzt werden, da dies zu erheblichen neurologischen Beeinträchtigungen führen kann, z.B. dem Verlust der Sprachfähigkeit. Eine optimierte Planung der Operation kann hierbei verhindern, dass Hochrisikobereiche verletzt werden, um so neurologische Schäden zu verhindern.

In dieser Arbeit wird ein neuartiges System vorgestellt, das den kompletten chirurgischen Arbeitsablauf abdeckt, indem eine vollautomatische Operationsplanung und anschließende Szenenregistrierung ausgeführt wird, um dem Chirurgen eine intraoperative Assistenz auf Basis von Augmented Reality (AR) zu ermöglichen.

In einem ersten Schritt werden die Bilddaten des Patienten segmentiert, um Patientenmodelle zu erstellen, die die Anatomie des zu Operierenden möglichst exakt darstellen. Die vorgestellten Methoden erreichen eine Segmentierungsgenauigkeit von 98% (F1-Score). In einem weiteren Schritt verwendet das System *Statistical Shape Models*, um die Punktionstrajektorie für den linken und rechten Ventrikel automatisiert zu bestimmen. Der Kocher Punkt wird in 98,4% der Fälle gefunden. Der korrekte Zielpunkt innerhalb des Ventrikelsystems wird in 95,4% bestimmt. Dies führt insgesamt zu 93,9% korrekt geplanten Trajektorien. Diese Planung wird durch eine patientenspezifische, Risikokarten-basierte Eintrittspunktbestimmung erweitert, um den optimalen risikoreduzierten Pfad für einen Patienten zu finden.

In dieser Arbeit werden Methoden vorgestellt, die ein robustes und akkurates Tracking des Patienten während eines chirurgischen Eingriffs ermöglichen. Verschiedene Marker für eine robuste Patientenverfolgung wurden untersucht. Als Ergebnis sind Vuforia Marker ausgewählt worden, da diese die Anforderungen am besten abdecken.

Zusätzlich wurde ein Infrarot-Marker Tracking auf der zur Unterstützung verwendeten AR-Brille (HoloLens) implementiert, um die Unzulänglichkeiten der relativ großen und nicht frei nutzbaren zwei-dimensionalen Vuforia Marker zu adressieren. Die chirurgischen Anforderungen an Sterilisierbarkeit und eine hohe Trackinggenauigkeit werden durch die Infrarot Marker erfüllt. Es wird ein Markersystem vorgestellt, das den Prozesswechsel von der unsterilen zur sterilen Operationsphase unterstützt. Die Wiederholgenau-

igkeit bei erneuter Montage beträgt 0,18 $\pm$ 0,06 mm. Das Markersystem kann Vuforia- und Infrarotmarker aufnehmen.

Es werden vier Registrierungsmethoden vorgestellt, um das Markersystem auf den Patienten zu registrieren. Drei der Verfahren unterstützen einen Benutzer bei der manuellen Registrierung, um die Patientenmodelle möglichst exakt mit dem realen Patienten in Einklang zu bringen. Verwendet werden kann ein Spielecontroller, ein Pointer oder Gesten. Das letzte Verfahren verwendet einen mehrstufigen *surface matching* Ansatz, um die Registrierung automatisiert auszuführen. Die Registrierungsmethode mithilfe des Spielecontrollers erreicht die höchste Genauigkeit mit einem Wert von 2,71 $\pm$ 1,18 mm.

Die erstellten Patientenmodelle und die ermittelten Punktionstrajektorien werden verwendet, um den Chirurgen intraoperativ durch Visualisierung dieser verborgenen Strukturen zu unterstützen. Die vorgestellten Methoden erhöhen die Punktionsgenauigkeit von linearen Trajektorien in Bezug auf die optimalen Eintritts- und Zielpunkte. Dies kann zu einer geringeren Penetrationsrate von Hochrisikobereichen im Gehirn und zu einem besseren Ergebnis für den Patienten führen.

Zunächst wurden Experimente nach dem Behandlungsstandard durchgeführt. Dabei wurde ein mittlerer Fehler von 6,6 $\pm$ 3,1 mm und eine Erfolgsrate von 72,7% festgestellt. Diese Werte liegen im Bereich, der im Stand der Forschung kommuniziert wird. In der ersten Stufe wird dem Anwender eine virtuelle Führung zur Unterstützung bei neurochirurgischen Eingriffen zur Verfügung gestellt. Dabei wurde ein mittlerer Fehler von 4,8 $\pm$ 2,5 mm und eine um neun Prozent bessere Erfolgsrate von 81,7% ermittelt. In einer aufbauenden Stufe wurde die Herausforderung eines gut gewählten initialen Punktionswinkels durch die Einführung einer Katheternavigationshilfe in Kombination mit dem AR-System adressiert. Der mittlere Fehler konnte auf 3,1 $\pm$ 1,8 mm gesenkt werden und eine Erfolgsrate von 98% wurde erreicht. Dennoch blieb die Herausforderung einer manuellen Manipulation bestehen. Um eine stabilere Manipulation im Raum zu ermöglichen, wurde die Navigationshilfe an einen Roboter montiert, der mit einem Spielecontroller telemanipuliert werden kann. Der Controller wurde verwendet, um den korrekten Winkel einzustellen. Es wurde ein mittlerer Fehler von 1,9 $\pm$ 1,2 mm erreicht. Alle Katheter konnten korrekt platziert werden. In den darauffolgenden Schritten wurde jeweils eine höhere Genauigkeit und Präzision erreicht.

Contents

1. Introduction

Supporting a surgeon during a neurosurgical procedure using augmented reality (AR) demands various high requirements from the assistance system, particularly on the accuracy of the registration and execution of the procedure, which is often in the millimeter range. In addition, there is the problem that the physician often cannot see inside the patient's head during certain procedures like biopsies, ventricular punctures and deep brain stimulation, which makes these types of interventions particularly challenging. Therefore, one of the most important factors is an accurate registration of the patient to the AR system, so that holograms can be displayed at the correct position. It is desirable to automate the operation planning so that a surgeon can focus on the execution of the procedure and patient counseling. In a first step anatomical structures of interest are determined that can then be used for automated determination of a valid path through the brain. At the moment these are mainly linear paths, due to a lack of flexible instruments that can perform non-linear paths. After the operation planning and an accurate scene registration, intraoperative assistance can support a surgeon during demanding procedures.

1.1. Research Questions

In this work an automated operation planning for neurosurgical interventions is presented, as well as accurate patient registration and tracking methods to enable an intraoperative assistance to support a surgeon using augmented reality on the example of the ventricular puncture.

The structure of this thesis is depicted in Figure 1.1.

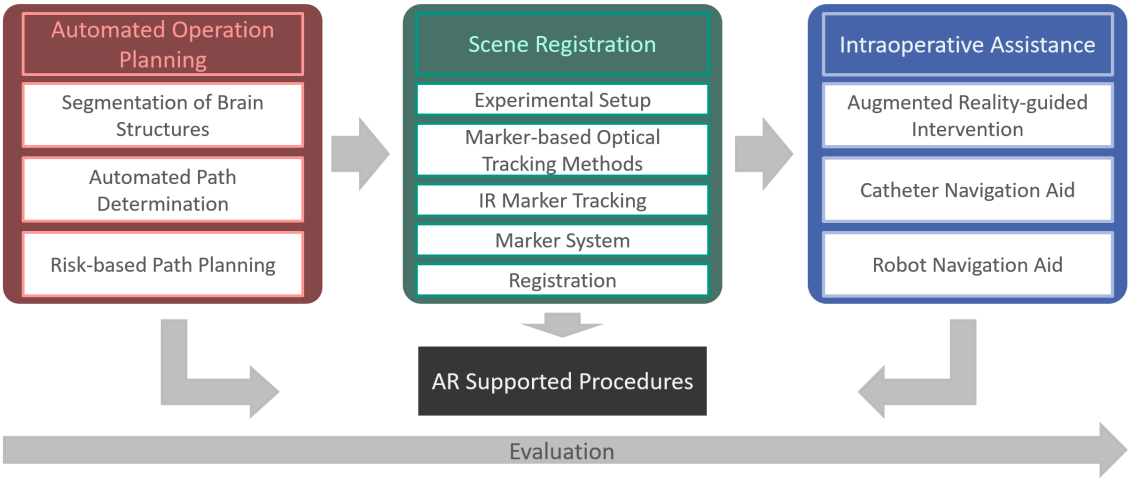


Figure 1.1.: The three pillars of this work to enable AR supported interventions.

The following research questions arise:

- **How can suitable medical knowledge be automatically retrieved and effectively applied to plan linear paths for neurosurgical interventions like ventriculostomies and biopsies?**
  Planning a neurosurgical intervention can be time-consuming and prone to errors if done manually. This time could be used for a neurosurgeon to prepare for the operation or take care of the patient. Annotation of image data, e.g. a CT scan of a patient is very time consuming. Automated methods can be used to extract the important structures from patient image data with a higher level of robustness. Especially in emergency situations, this is time critical. Therefore, the time needed to execute all necessary methods must be minimized. The automated determination of puncture trajectories could help to find the optimal paths to a target with minimal risk.

  A first goal of this thesis is to automatically derive the patient models and the optimal puncture trajectory for ventriculostomies based on medical knowledge. For biopsies, a risk-based trajectory planner is investigated that can help a neurosurgeon to derive the optimal path to perform a tumor biopsy. Methods are being investigated to automate these tasks to minimize the effort required of manually performed tasks.

- **Which algorithmic methods should be applied to perform an accurate neurosurgical scene registration?**
  Augmented reality could support a surgeon during demanding neurosurgical interventions. The most important aspect hereby is the correct alignment of the virtual holograms of the augmented reality scene with the real world. The scene is thereby constantly changing, as the wearer of the augmented reality device or head mounted display is normally constantly moving, in the mm- to m-range. The objects or persons in the scene can also change their positions. This could be the patient, operating room staff or neurosurgical instruments. All entities that are processed within the augmented reality scene need to be constantly tracked as well. It is thereby especially challenging to track the patient in the operating room because in the standard of care the patient is fully covered with sterile drape to minimize the risk of an infection. Only the operation situs remains not covered.

  In this thesis, it will be investigated, which methods are needed to perform a robust and accurate scene registration for neurosurgical applications of AR. All possible error sources need to be identified and minimized.

- **How can these methods and the results of the automated operation planning be used to implement an augmented reality-based intraoperative assistance?**

  The presentation of the holographic visualization in neurosurgical interventions has presumably a high influence on the performance of the system. The system consists of hard- and software components that need to be perfectly integrated to enable a usable augmented reality system. A drawback when using AR guidance to puncture the ventricular system or perform a biopsy is that it needs to be performed manually, which could be the largest error source of the system. An additional important aspect is the acceptance of the system from neurosurgeons who are the target group

to use such a system. It is therefore important to be in constant exchange with medical experts to meet their clinical requirements and match the surgical workflow.

It will be investigated how a virtual scene needs to be designed to reach a good acceptance of the system and a better outcome for the patient. Further it will be examined how the system can be advanced to overcome errors from the manual placement of the catheter.

From these three research questions, a key question can be derived:
Can augmented reality help to increase the accuracy of ventriculostomies and biopsies?

1.2. Contributions

This work addresses the above described research questions and presents novel and robust methods to automate tasks for operation planning and scene registration to enable intraoperative assistance for neurosurgical interventions.

The main contributions of this work are listed below and are further examined in this thesis:

- **Automated Operation Planning: Automated segmentation of the ventricular system, the skin and the skull.**
  To perform an automated operation planning, vital structures of the human head need to be segmented in a first step as reference marks for further algorithms. For the ventriculostomy these are the skin, the skull and the ventricular system. In this work methods are presented for automated segmentation of these structures [113].
- **Automated Operation Planning: Automated determination of the puncture path on the example of the ventriculostomy.**
  For safe execution of the ventriculostomy a path must be determined based on landmarks as in the standard of care. Therefore, two points on every side must be determined: the Kocher's point and the target point inside the ventricular system. The insertion path is defined between these two points. This work presents an approach to plan the path for the ventricular puncture based on medical knowledge [119].
- **Automated Operation Planning: Risk-based path planning.**
  In the standard of care no patient-specific inner structures of the human brain are considered. The Kocher's point is chosen in a way that as little damage as possible is done to vital structures of the human brain, it is based on experience values. However, the best way would be a patient specific approach by using a personalized risk structure map representing high-risk values of the brain. We present methods to use extracted risk maps from different image modalities to provide a puncture path with minimized risk [118] [117].
- **Scene Registration: Implementation of an infrared marker tracking on the HoloLens.**
  With the release of the research mode of the HoloLens (Microsoft, USA) it was made possible to access all sensor streams and to implement a tracking of infrared marker targets. Two dimensional markers have several disadvantages. For example

they are not sterilizable and are often large because of their quadratic form. In this work we present a tracking algorithm that is able to track infrared spheres directly with the HoloLens, even when moving [116]

- **Scene Registration: Development of a marker system.**
  During neurosurgical procedures the patient is normally fully covered with sterile sheets. Only the OP situs is visible through a recess in the sheet. This makes it hard to track a patient directly, e.g. through facial landmarks. To address this problem a custom build marker system is developed that is mounted on the patient's forehead [114]. It can be mounted non-invasively by using a formable mass that hardens out and an elastic band to hold it in place. Different markers and their attachments have been evaluated in regard to tracking accuracy and performance. The marker system can be customized to hold Vuforia, infrared or Aruco markers. The marker tracking methods Aruco and AprilTags were evaluated in regard to their tracking accuracy [112].

- **Scene Registration: Marker to patient registration.**
  To enable a high-quality hologram visualization for intraoperative assistance an accurate and precise marker to patient registration needs to be performed. We present four approaches to perform manual and automated registration. The manual registration can be done using a game controller, a pointer or hand and head gestures. The automated registration is done via surface matching.

- **Intraoperative Assistance: Augmented reality-guided ventricular puncture.**
  In the basic variant of the augmented reality support, the holograms are offered as a visual assistance to the surgeon to help them puncture the ventricular system with higher accuracy and precision [190]. A surgeon can choose from different visualization templates to superimpose the ventricular system, the skin and the skull over the real patient anatomy. The automatically derived optimal puncture path is also visualized to the surgeon for the best possible catheter placement [119].

- **Intraoperative Assistance: Navigated ventricular puncture.**
  In the second stage of the assistance a handheld catheter navigation aid is used to enable enhanced augmented reality support to the surgeon by visualizing the correct insertion angle.

- **Intraoperative Assistance: Robot-guided intervention.**
  The final stage utilizes a robot to control the catheter navigation aid to provide a more stable angle visualization.

1.3. Outline

The following thesis is structured in eight chapters. These chapters include the following:

- Chapter 2 gives an overview of the medical fundamentals important for this work. Imaging modalities as well as relevant surgical procedures are presented.

- Chapter 3 gives an overview of the related work regarding augmented reality in medicine and systems that are comparable to the proposed system of this work.

- Chapter 4 describes methods to enable automated operation planning. Algorithms for automated segmentation of vital structures are presented that are used for visualization and to perform a automatic path determination on the basis of statistical shape models. As an extension to this, a path determination based on high-risk structures of the human brain is presented.
- Chapter 5 investigates methods for scene registration to properly track the patient. The marker system is registered to the patient to superimpose the holograms at the correct position.
- Chapter 6 describes the three assistive stages of the AR guidance.
- Chapter 7 discusses the presented methods and the research questions.
- Chapter 8 gives a summary of the presented work and provides an outlook of future work and challenges.

2. Medical Fundamentals

2.1. Imaging Modalities

The diagnostic imaging of choice regarding the brain is Magnetic Resonance Imaging (MRI), which offers better contrast between gray and white matter than computed tomography (CT). CT scans provide, with the help of contrast agents, a clear visual differentiation of the surface of the brain, ventricles, tumors and blood vessels [78].

A voxel describes a cuboid image region, defined by the *height*, *width*, and *depth*. The resulting image of a MRI or a CT consists of a multitude of voxels arranged in layers. The voxel domain is defined by Equation 2.1, wherein x, y and z are the position of a voxel p_j.

$$p_j = \{(x, y, z)^\intercal \in [0, \mathit{width}] \times [0, \mathit{height}] \times [0, \mathit{depth}] \subset \mathbb{R}^3 \tag{2.1}$$

2.1.1. Magnetic Resonance Imaging

Magnetic resonance imaging (MRI) makes use of the concept of nuclear magnetic resonance (NRM). It allows physicians to image the insides of the body without, in contrast to CT, using ionizing radiation. An example of a modern MRI scanner of the university hospital in Guenzburg is shown in Figure 2.1. Hydrogen nuclei are spinning, which makes them magnetic around their axis of rotation. Within tissue, the spinning axes of hydrogen nuclei are randomly aligned. Upon application of a strong magnetic field, the spin axes have the tendency to align, oscillating around the direction of the field. This oscillation emits radio frequency (RF) waves but decreases as the spin axis aligns closer to the direction of the field. To force the spin axis out of its stable position (excitation), an RF pulse is applied. After the pulse, the spin axis slowly returns to its previous alignment with the magnetic field (relaxation). During this process, the nuclei emit RF waves. From this signal, MRI images are generated. The speed of the relaxation is tissue dependent and is slower for large molecules or dense tissue. This results in contrasts between tissues in the resulting MR images [144].

The sequence of RF pulses can be varied to create different types of images. Most commonly T1-weighted and T2-weighted sequences are recorded. Cerebrospinal fluid (CSF) appears dark on T1 and bright on T2-weighted images. They are sometimes used with the contrast agent Gadolinium [144]. FLAIR (Fluid Attenuated Inversion Recovery) sequence highlights abnormalities, normal CSF and white matter remain dark [187]. SWI (Susceptibility Weighted Imaging) is a sensitive method to show the amount of iron in the brain and can help to discover multiple sclerosis, stroke, trauma, vascular malformations and tumors [80]. A brain mask highlights all brain tissue regions. This can either be done

by a medical expert or by a machine learning algorithm that evaluates MRI images [107]. Functional Magnetic Resonance Imaging (fMRI) is suited to visualize neuronal activity in the brain [144]. Diffusion Tensor imaging (DTI) provides a good visualization of nerve fiber bundles [125].

Figure 2.1.: Magnetic resonance imaging scanner (Magnetoma Vida 3 Tesla, Siemens Healthineers, Erlangen) in the university hospital Guenzburg

2.1.2. Computed Tomography

X-ray computed tomography (CT) was the first of the modern slice-imaging modalities, introduced in 1972 by Sir Godfrey Newbold Hounsfield and Allan McLeod Cormack, who received the Nobel prize in medicine in 1979 for their invention. Since then, the technology has been improved, especially with the introduction of spiral scanning, which provides true volume imaging instead of slice-by-slice imaging. The invention of array detector technology in the 1990s allows imaging of the whole body in 5 to 20s while providing sub-millimeter isotropic resolution. CT is the standard for fast 3D diagnostics of trauma patients, where it cannot be clarified before the measurement, whether a MRI may be performed at all [95, 29]. An example of a modern CT scanner located at the university hospital in Guenzburg is shown in Figure 2.2.

The values of all voxels p_j of a CT scan describe the mean attenuation of the scanned tissue. The Hounsfield scale maps corresponding tissue types onto a scale representing Hounsfield units (HU). Example tissue values are visualized in Table 2.1.

2.2. Relevant Types of Neurosurgical Interventions

In the following the relevant types of neurosurgical interventions are presented that are used as clinical scenarios in this work.

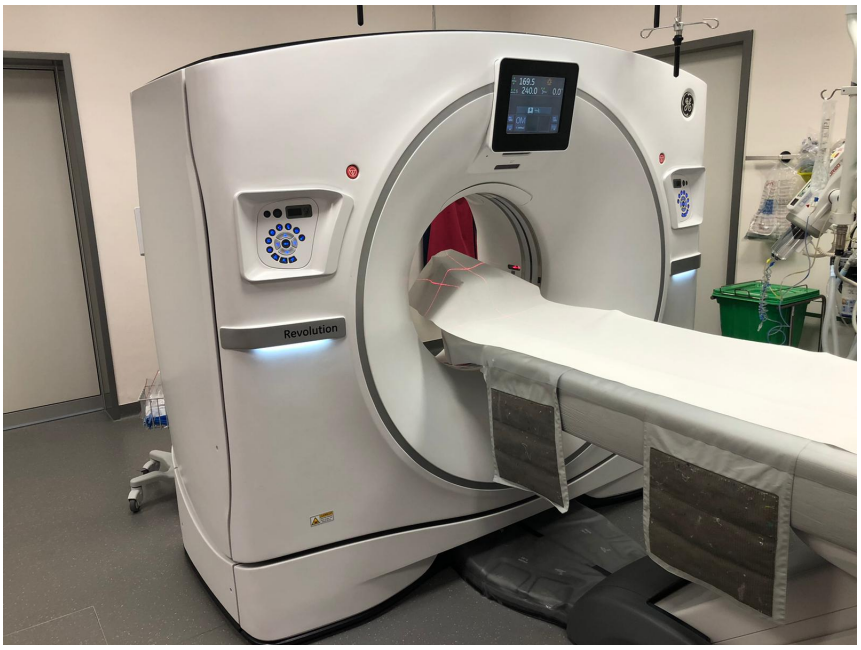


Figure 2.2.: Computed tomography scanner (Revolution CT, GE Healthcare, Great Britain) in the university hospital Guenzburg.

Table 2.1.: Hounsfield unit scale with mapped tissue types.

Tissue	HU
Air	-1000
Lung	-600 to -400
Fat	-100 to -60
Water	0-14
CSF	0-14
Muscle	10 to 40
White matter	20 to 30
Blood	30 to 45
Grey matter	37 to 45
Soft tissue	40 to 80
Bone	400 to 3000

2.2.1. Ventriculostomy

The ventricular puncture or ventriculostomy is a standard procedure in neurosurgery. In Germany 25'000 ventriculostomies are performed each year as a standalone operation or as part of a more complex intervention. The ventricular system consists of multiple ventricles, the two large lateral ventricles, the third and fourth ventricle, which are all interconnected. The ventricular system is also connected to the spinal cord and the subarachnoid space. A schematic drawing is visualized in Figure 2.3 a. It lies approximately 4-5 cm deep in the center of the brain and has an elongated shape. The shape of the ventricles can vary substantially, but normally they are 1-2 cm wide and symmetric. Due to pathologies, this can differ. The ventricular puncture is done when there is an overpressure inside (Figure 2.3 b) the ventricular system or for the diagnosis of cerebrospinal fluid.

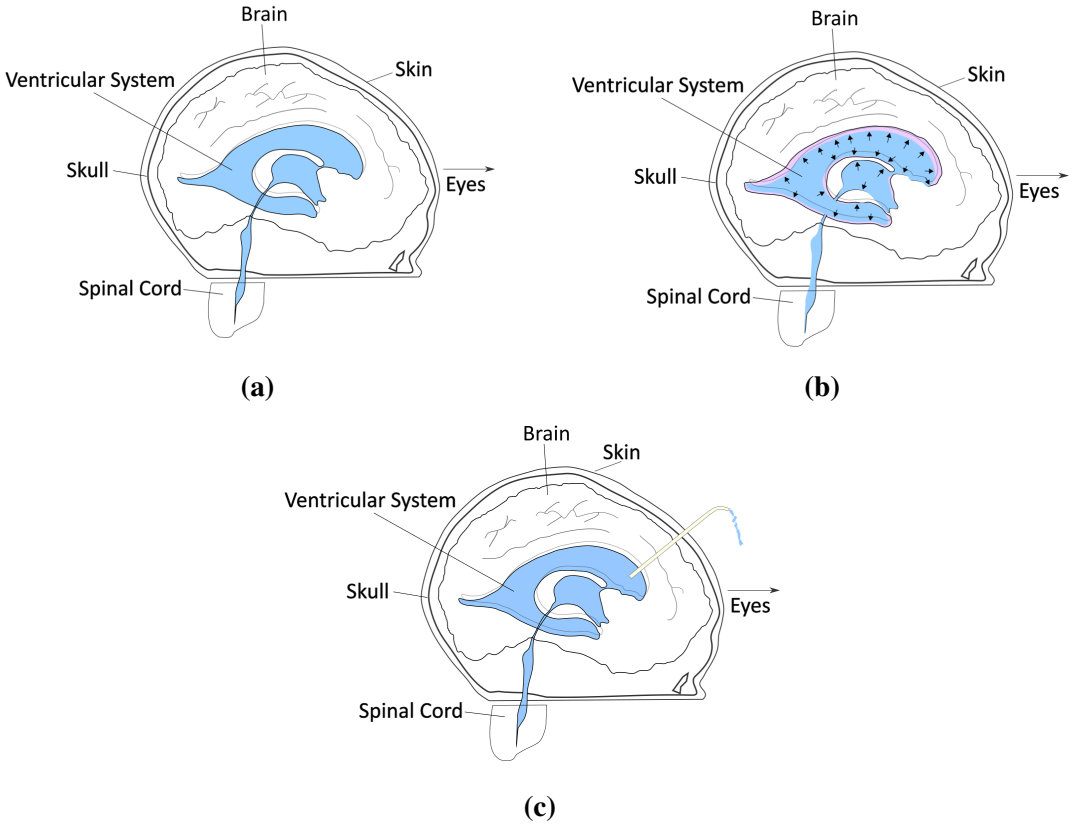


Figure 2.3.: a) Non pathological ventricular system. b) Hydrocephalus, overpressure in the ventricular system. The ventricles are widened. c) Ventriculostomy.

The ventricular puncture is often indicated in emergency situations, e.g. after an accident and if a person suffers from a traumatic brain injury or intracranial hemorrhage. The procedure can also become necessary after a stroke. When a patient is delivered to the hospital a mandatory head CT scan is made, if an injury to the brain is suspected. When widened ventricles are detected, a ventriculostomy is performed as fast as possible. The longer the pressure is induced to the brain, the more damage is dealt to it. High pressure leads to severe damage to the brain and can lead to death.

The goal is to place a neurosurgical catheter inside the ventricular system to release the overpressure from the ventricle (Figure 2.3 c). Ventricular puncture is often performed using freehand technique. The neurosurgeon orients according to anatomical landmarks, like the ears and the nose, because the patient is fully covered for sterility. The alternative is to use a stereotactic frame or with the help of a navigation system, e.g. Brainlab. Using an additional navigation software is very time-consuming, because the intervention needs to be planned. In emergency situations the manual placement is preferred.

During ventriculostomy, first an incision is made at the Kocher's point on the left or right hemisphere, then the skull is opened through a burr hole trepanation. Then the *dura mater*, the meninges, is opened and the puncture is done by aiming with a catheter towards the ventricular system. The optimal target point inside the ventricle is the *foramen of monro* - the connection to the third ventricle. The Kocher's point is chosen in a way that no high risk areas of the brain are penetrated, such as the motor or speech cortex.

An illustration of the catheter placement is depicted in Figure 2.4 a.

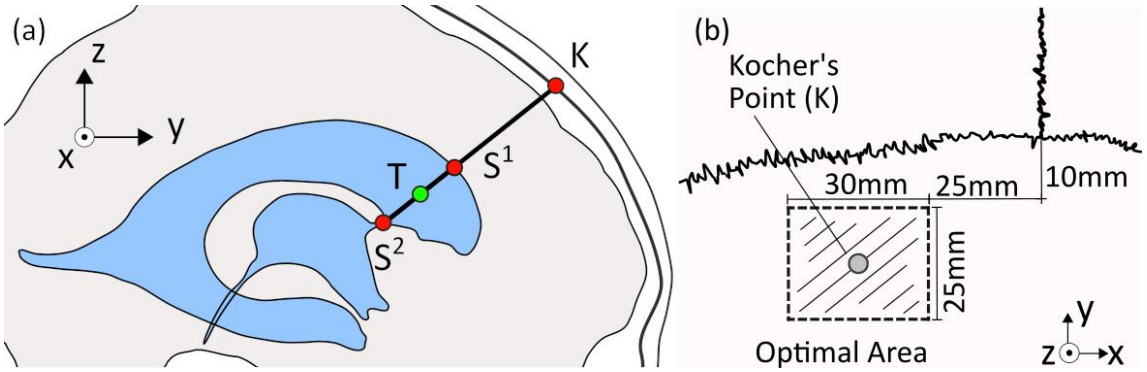


Figure 2.4.: Construction of the insertion vector and optimal Kocher area.

There are different definitions of the Kocher's point in literature. In this work, a Kocher's area is assumed, as depicted in Figure 2.4 b. The optimal point lies within an area of 1 - 3.5 cm in front of the *coronal suture* and 2.5 to 5.5 cm left and right to the midline.

The placement of the catheter has to be performed as accurately as possible to prevent any unnecessary damage to the delicate brain structures. A wrongfully placed catheter may result in the loss of brain function or death. The first puncture attempt for catheter placement in ventriculostomy is only successful in 40-77% [224, 209, 128, 94, 88, 86, 172, 66]. A placement is rated as successful if cerebrospinal fluid is released out of the catheter. If the catheter placement was not successful at the first trial the brain is punctured again until the placement is successful, resulting in extended damage to brain tissue.

In the following work the ventricular puncture is used as the clinical scenario in Sections 4.1 and 4.2. The three stages of intraoperative assistance presented in chapter 6 is also carried out using the ventriculostomy.

2.2.2. Biopsies of Brain Tumors

A biopsy describes the extraction of a tissue sample for subsequent characterization of the tissue type. The removed tissue can be examined under a microscope and often chemical analyses are performed. The pathologist can draw conclusions about pathological changes in the fine-tissue structure. A biopsy is often used to clarify whether tumors are malignant or benign [19, 130, 225, 135, 226, 73, 218].

In the following work the example of tumor biopsies on the brain is examined in Section 4.3.

3. Theoretic Background and Related Work

3.1. AR Concepts and Technologies

There are different definitions of augmented reality (AR). In the literature, the definition of Azuma [14] is the most commonly used. According to Azuma, an AR system has three basic properties:

- A combination of real and virtual worlds.
- Registration of real and virtual objects in 3D.
- Interactivity in real time.

In order to fulfill these characteristics, various components are required to work in unison. These components may include displays, tracking systems, cameras, projectors, graphics computers, and other specialized devices. Following the first two features of Azuma's definition, this section first describes various types of displays used to combine real and virtual worlds, followed by tracking systems that track the pose of the user and other real-world objects to correctly overlay the virtual objects.

3.1.1. Displays to Enable Augmented Reality

In order to overlay reality with virtual elements, various types of displays can be used. They can be categorized according to their positioning in 3D space into head-mounted displays, hand-mounted displays and spatial displays. A distinction can also be made in the visual representation of AR. These include video-transparent displays, optical-transparent displays, projectors, and holograms [205].

3.1.1.1. Head-mounted Displays

Head-mounted displays are special glasses that allow graphics to be visualized in the user's field of vision, so that both hands can be used for the actual task. There are basically two different types of head-mounted displays: video see-through displays and optical see-through displays [33].

Video see-through displays are the most inexpensive and straightforward to implement. They do not offer a direct view of the real world. A camera is used instead to capture an image of the outside world. This image is placed on the projection screen and virtual objects are superimposed. Both, the real world and the virtual world are fully digitized

and merged into one view. This makes it easier to synchronize the two and visual delays can be avoided. It is also possible to adjust the brightness and contrast of virtual objects to match the environment. However, there are also disadvantages, like the reduced resolution of reality, the limited field of view and the lack of orientation [216].

Optical see-through displays enable the user to see the real world directly through a transparent projection surface. In addition, virtual objects are displayed within this projection surface using a semi-transparent mirror placed at an angle in front of the user's eyes. These displays have two major advantages over video see-through displays: first, the user can perceive the environment without experiencing reduced resolution, time delay, or limitations due to lens distortion or eye shift. Second, the user does not go blind if the device is disconnected from the power supply [24, 216].

3.1.1.2. Hand-mounted Displays

Hand-mounted displays are devices that are held in the user's hand and provide the projection surface. Examples of such devices include smartphones and tablets. AR with handheld devices is an attractive option for many companies, as many people already own one of these devices. This way, no special hardware is needed and only the software has to be developed. This makes it a perfect fit for the entertainment industry, where a wide audience needs to be reached. However, in other fields such as medicine and military, handheld displays are impractical, because the user does not have both hands free. In addition, the size of smartphones is not optimal for many use cases, and devices with larger displays quickly become uncomfortable to carry [33].

3.1.1.3. Spatial Displays

Spatial AR separates user and technology by using projectors or holograms to directly overlay the real environment with virtual content. The advantage of this is that the user does not have to wear or carry a device. In addition, not only a single person but a group of users can experience the same AR projection at the same time. A major disadvantage, however, is that it is limited to non-mobile applications. Thus, spatial AR is mainly used in museums, laboratories and medicine [33].

3.1.2. Tracking Techniques

An AR system must ensure that the virtual content is positioned accurately in the real world so that it appears as being part of the physical environment. The challenge here is to ensure that the virtual objects remain aligned with the position and orientation of the real objects as the user moves or changes their viewpoint. There are several tracking techniques that help meet this requirement. A distinction can be made between sensor-based tracking and vision-based tracking.

3.1.2.1. Vision-based Tracking

Vision-based tracking utilizes a camera and computer vision algorithms. These algorithms can compute the position of the camera relative to the objects based on the images

provided. To detect virtual objects, artificial markers can be placed in the scene. These markers are easily recognized, because they are simple geometric shapes such as squares or circles. For reliable object detection, however, the markers must have the appropriate brightness and a clear view. A detailed state of the art is provided in Section 5.2.1.

3.1.2.2. Sensor-based Tracking

Sensor-based tracking methods use active sensors including magnetic, acoustic, inertial or mechanical sensors to track the user's position. In magnetic tracking systems, a magnetic field is generated. The relative position between the source of the magnetic field and a receiver can be measured. This is a low-cost solution, but does not provide the required accuracy for use in AR.
Acoustic tracking methods utilize an ultrasonic transmitter attached to the tracked object and acoustic sensors placed in the environment. It measures the time between sending and receiving the ultrasound and uses it to calculate the relative positions. Since the speed of sound is relatively slow, these systems have slower response than other sensor-based methods.
Inertial tracking methods use an accelerometer and a mechanical gyroscope to measure the acceleration and rotation of an object. Since this does not require an external reference point, it can be easily applied in a variety of environments. However, small errors can accumulate over time.
Mechanical tracking methods require the tracked object to be connected to a reference point using a robotic arm that consists of mechanical links. By measuring the angle between these links, the position of the tracked object can be determined. This allows for very accurate measurements, but requires a robotic arm. Since all of these methods use active sensors, special hardware is always required, making them often difficult to use in practical AR applications [173].

3.2. Augmented Reality in Medicine

Augmented reality has been used in multiple works and fields. Since the focus of this work is on neurosurgery, a general overview of the state of the art with respect to AR in the field of neurosurgery is given. Additionally, other fields are investigated.

3.2.1. AR in Neurosurgery

One of the most successful areas in medicine where AR has been applied is neurosurgery. This is based on the fact that the skull serves as a rigid structure and fixed point between taking several patient scans during surgery. Additionally, the displacement and deformation of the brain is limited. These observations facilitate the implementation of robust tracking in real time. On the other hand, the surgical field is small, and even minor errors can lead to lasting consequences. Therefore, surgeons need to develop the ability to visualize the structures under the skin. In this process, AR can help the surgeon by making these structures visible. This is especially important in the field of minimally invasive neurosurgery, where the field of view is even more limited [147].

The field of neurosurgery is also where the first AR-assisted procedure took place [22]. In 1989, Roberts et al. [178] developed an AR system that integrates CT images with the surgical microscope. Spatial registration of the head and microscope is required to correctly position the CT image. To do this, the relative position of the head with respect to the microscope is first measured by manually focusing the microscope on three points of the patient's head. The absolute position of the microscope is determined using an ultrasound range finder. A number of reports followed, in which various AR systems for neurosurgical procedures were presented.

King et al. [102] presented another early surgical navigation system using AR in 1999. This system also uses the surgical microscope. The surgeon's view is enhanced by 3D structural information. Since the surgical microscope magnifies the view, precise alignment and tracking is even more important. They use bone-implanted fiducials and a dental attachment for tracking. During a clinical study, their evaluation showed an error of 0.5-1.5 mm, which increases to 2-4 mm when brain deformation occurs.

In 2002, Kawamata et al. [101] described a system that adds AR to images of an endoscope. 3D images of the tumor and adjacent structures are superimposed on the endoscopic images in real time. The color of the tumor image changes based on the distance between the tip of the endoscope and the tumor. Optical tracking was performed using two sets of infrared LEDs and a goggle frame with reference markers. One set of infrared LEDs is attached to the patient's head, the other to the endoscope. Clinical tests during 12 tumor surgeries showed that their AR navigation system is a promising tool for tumor surgeries.

Seven years later, in 2009, DEX-Ray, an AR neuronavigation system, was developed by Kockro et al. [109]. It includes a handheld pointer with a lipstick-sized camera mounted on top of the pointer. The video stream from the camera is overlaid with multimodal 3D graphics and landmarks. These are obtained during neurosurgical planning and displayed on a separate screen. The hand-held probe serves as both navigation and interaction device for the 3D graphics. They evaluated the registration accuracy of the system in 12 surgeries of tumor and vascular cases. The mean registration error was 1.2 mm. According to their overall assessment, especially in the macroscopic phase of surgery, the display of 3D information of internal structures improves the navigation experience. In deep and narrow corridors, lighting conditions and camera resolution limit this experience.

In 2013, Inoue et al. [90] developed an AR neuronavigation system that uses webcams for tracking. Their solution is cheaper than others that require more specialized hardware. In addition to imaging tumors and vessels, the system can also perform tractography, a technique for visualizing neural pathways, that were generated using the open-source 3D Slicer software [41], [62]. The system was tested on three patients and the results showed its usefulness in brain surgery.

Deng et al. [53] presented a hand-mounted solution for neurosurgery. In 2014, they introduced an AR system that relies on a tablet as a handheld display. They use two different tracking systems for patient and device localization, both based on artificial markers. They performed a preclinical study and a clinical study that demonstrated the feasibility of the system. Similarly, Watanabe et al. [223] also used a tablet PC in 2016 to overlay a 3D brain model on a patient's head. The brain model was obtained from magnetic resonance imaging and computer tomography. Tracking was performed using a Vicon (Vicon Motion Systems, UK) system with six cameras. The system has been evaluated during six

tumor resection surgeries and the researchers concluded that the system is accurate to within 1 mm and can help surgeons plan skin incisions and locate superficial tumors.

In 2015, Tabrizi and Mahvash [206] introduced an AR system that superimposes regions of interest onto the skull and brain of the patient using a projector. They used images of segmented brain tumors to define regions of interest. Five fiducial markers were placed on the patient's head around the tumor region to perform the registration. Before surgery, magnetic resonance images were taken with the attached markers. Using the magnetic resonance images, a 3D model was reconstructed, in which the fiducial markers were visible as well. The fiducial markers of the virtual model could now be overlaid onto the physical markers. The projector was then fixed to this position. With the help of this AR system, five surgeries were successfully performed leading to a mean projection error of 1.2 ± 0.54 mm. Tabrizi and Mahvash concluded that the system is accurate enough for the use in neurosurgery.

Cutolo et al. [48] developed a head-mounted, see-through AR system in 2017 to improve surgery for complex neurological lesions. A patient-specific head mannequin containing an anatomically realistic brain phantom with tumors was used to test the system. They concluded that the novel system is user-friendly and can be successfully used for making skin incisions, conducting craniotomies, and for lesion targeting.

AR systems that have been used in the neurosurgical context are often expensive and customized. This is due to the lack of research on the use of low-cost commercial AR systems in the operating room [110]. Different research groups tested commercial available low-cost devices in 2018 and 2019.

In 2018, Maruyama et al. [146] were the first to use a surgical navigation system with available smart glasses (Moverio BT-200, Seiko Epson Corporation, Suwa, Japan) in clinical practice. Patient models were created from preoperative magnetic resonance images and superimposed on the see-through glasses. These were models of the scalp, brain tumors, and vessels of the brain surface [203]. Attaching optical markers to the glasses and the patient's head provided accurate navigation. Tracking was enabled by the use of two motion capture cameras. The technology was used in two brain tumor surgeries. The results show that hands-free neuronavigation in the surgical field was possible. The targeting error in two surgeries ranged from 0.6 to 4.9 mm.

In 2019, van Doormaal et al. [214] used smart glasses (HoloLens, Microsoft, Redmond, Washington) in a neuronavigation system. During surgery of three patients the registration errors with holographic neuronavigation and with conventional neuronavigation were measured. The error with holographic neuronavigation was 4.4 ± 2.5 mm compared to an error of 3.6 ± 0.5 mm with conventional neuronavigation. Their conclusion was that holographic neuronavigation can be used in the operating room in the future, but that some improvements need to be made before this can be used as a routine in clinical practice.

HoloLens was also used by Kubben and Sinlae [110]. The surgeons can perform special hand gestures to interact with the system without touching it. All available hand gestures were tested under two different lighting conditions (general theater light and theater light with surgical light). Also, different types of surgical gloves were used in testing the device. Speech recognition was also evaluated against background noise in the form of two nurses talking to each other. The results showed that all gestures were recognized under both lighting conditions and regardless of the gloves worn. Speech recognition and visibility of the holographic images were both good. However, the brightness of

the display had to be properly adjusted for the images to be visible. Overall, the results showed that the system is stable and works reliably in surgical environments with sterile garments.

In 2018, Gibby et al. [76] evaluated Microsoft HoloLens for guiding pedicle screw placement. During preoperative planning, virtual trajectories were created for ideal access into the pedicle. CT images and the planned trajectories were displayed using HoloLens in combination with the Novarad [43] OpenSight application. This specialized application for AR in surgery offers automatic image registration, advanced image processing, and automatic review and logging. Following the projected trajectories, the surgeon inserted spinal needles until they made contact with the bone. Another CT was then performed to compare the actual needle trajectory with the planned one. A phantom evaluation showed that the deviation was approximately circular with an average radius of 2.5 mm. Clinical studies have not been performed to date.

3.2.2. Medical Education and Training

As a surgeon, a thorough knowledge of human anatomy is essential. Traditionally, students use textbooks with visualizations to gain this knowledge. However, this can be challenging because 2D images are fixed to one viewpoint, making it difficult to understand the spatial context. The use of AR has been proposed to solve these problems and increase the effectiveness of the learning process [141]. There are indications that AR can improve students' motivation to learn [195].

The magic mirror developed by Ma et al. [141] is an example of such an educational AR system. The user faces an RGB-D camera that records his or her movements. The video stream is displayed on a computer screen and superimposed with a 3D model of his anatomy. A specific organ can be selected to get additional medical images and text information. In this way, the user can interactively explore the human anatomy.

Using a mobile device to display a 3D model of an anatomical structure is another common approach to medical education. To identify which model to display, the mobile device recognizes a specific marker. These markers can also be included in textbooks. The user can view the objects from all sides and even take a look inside. Examples of these systems have been proposed by Jamali et al. [91], Wang et al. [222] and Kurniawan et al. [120].

Learning in a medical context refers not only to theoretical knowledge, but also to practical experience. Complex skills must be acquired to successfully perform medical procedures. These skills include navigating complex physiological systems, developing expertise, and collaborative skills necessary for multidisciplinary medical practice. Medical training in a real workplace environment, however, is often not possible due to safety, time, and cost constraints. In addition, trainees may be discouraged by the complexity of the workplace environment. AR provides an opportunity to create a safe, appropriate, and cost-effective training environment where the trainee is allowed to make mistakes without fear of consequences [96]. Several AR training systems are described below.

ProMIS is an AR simulator [168] that can be used to train skills for laparoscopic surgery. The simulator consists of a torso-shaped mannequin with three camera tracking systems to identify instruments inside the mannequin. To do this, the instruments must be marked

with colored tape. To train a procedure, a special tray must be inserted into the mannequin. This tray can contain physical components that give real haptic feedback when using the instruments. In addition, virtual objects can be overlaid to enrich the simulation. In order to process the data collected during the procedure, the simulator is connected to a portable computer. The data includes time, path length, and uniformity of motion during each task. Upon completion of a training module, the statistics are shown on the computer display and a full video of the procedure is saved. The tasks that can be trained with the simulator include navigation, object positioning, sewing, knot tying, and sharp dissection [168]. Botden et al. [26] evaluated the system by analyzing questionnaires from 55 participants who completed basic tasks as well as suturing and knot-tying tasks. The questionnaires included questions about realism, haptics, and didactic value of the simulator. The results showed that ProMIS is a useful training system for laparoscopy.

Lahanas et al. [122] created another AR simulator for training basic laparoscopic tasks. The novel system includes a box trainer, a camera, and a set of laparoscopic tools. Custom-built sensors were attached to the tools to allow interaction with virtual objects. For training basic skills such as hand-eye coordination and depth perception, instrument navigation, peg transfer, and clipping tasks were developed. To evaluate the system, they divided subjects into novices and experts according to their experience in laparoscopic surgery. Execution time, tool path length, and two task-specific errors were recorded. Their results show that there is a large difference between the two groups in terms of smoothness and economy of motion. Therefore, the proposed system has the potential to recognize users with different experience in laparoscopic surgery. Unlike ProMIS, the new AR-based system allows users to interact with rigid and deformable virtual models in real time.

Endotracheal intubation is a common procedure in which a tube is inserted through the patient's mouth into the trachea. This is often a life-saving procedure, but it is difficult to perform. Traditionally, mannequins such as the Human Patient Simulator (CAE Healthcare, Inc., USA) have been used to train this procedure. Hamza-Lup et al. [81] uses AR to enhance the Human Patient Simulator. The trainee wears an optical, see-through head-mounted display that superimposes a 3D model of the underlying anatomy. An optical tracking system is used to track the trainee's head, the human patient simulator, and the inserted tube. Their conclusion was that the system can accurately superimpose the 3D model, but further studies are needed to fully evaluate the system.

Shenai et al. [198] are taking a different approach to medical education with their developed system called VIPAR (virtual interactive presence and augmented reality). The VIPAR system enables an experienced surgeon to remotely assist the operating surgeon. Two stations are set up for this purpose. The local station is set up above the surgical field in the operating room. The remote station is set up above a blue screen. They both consist of two cameras and a high-resolution display. The blue screen in the remote operating room allows easy segmentation of the remote surgeon's hands and instruments. A video stream of these elements is then displayed in the local surgeon's view in real time. Correspondingly, the video stream recorded at the local station is shown to the remote surgeon. Additionally, preoperatively recorded MRI images were added to the virtual fields of the two surgeons. The system was tested on a fixed-formalin cadaver head and neck. The remote surgeon was able to observe the procedure and provide instructions to the local surgeon. In addition, the enhanced MRI images provided spatial guidance. Since this system is already used in surgeries, it is not only for medical education but can also be considered as a surgical AR system.

3.2.3. Other Fields

In the following other medical fields where augmented reality is used are summarized below.

3.2.3.1. Orthopedic Surgery

Orthopedic surgery is a common target area of AR in medicine, as it addresses organs in the human body that have little ability to move and deform. AR enables the direct visualization of reconstructions on the patient's body. This makes it easier for the surgeon to focus, as he or she does not have to look at a distant display to see the planned reconstruction.

A mobile C-arm produces diagnostic images using X-ray during minimal invasive surgery. In 2010, Navab et al. [157] mounted a camera and a double mirror system on a C-arm, making it possible to fuse images obtained by the camera with X-ray images. Von der Heide [220] tested the system in a clinical trial. Their results show that the new system can reduce X-ray pulses by 48%. Abe et al. [5] designed a head-mounted AR guidance system called VIPAR (virtual protractor with augmented reality) in 2013, which visualizes a needle trajectory to the surgeon during percutaneous vertebroplasty. Wu et al. [227] proposed another AR system for vertebroplasty one year later called ARCASS (augmented reality computer-assisted spine surgery). They used skin markers for tracking and overlay a preoperatively created 3D model on the patient's body with the help of a camera and a projector. The system significantly reduced the time for finding a suitable entry point and reduced the radiation exposure.

Ogawa et al. [162] described the new system AR-HIP in a pilot study in 2018. It assists the surgeon during total hip arthroplasty for acetabular cup placement. The physician can see an image of the acetabular cup superimposed in the surgical field as well as its placement angle through a smartphone. The AR-HIP system delivers more accurate results than the goniometer. Tsukada et al. [211] adopted the system for total knee arthroplasty in 2019, superimposing the tibial axis, the aiming varus angle and posterior slope angle. The achieved results were similar to Ogawa et al. [162]. In 2018, Liu et al. [129] proposed an AR system for hip surfacing, an alternative for total hip replacement. Instead of markers they used a depth camera for registration. The preoperative data is shown by a commercial AR headset to assist the surgeon in hole drilling. They found a similar error as in other commercial computer-assisted orthopedic systems.

Müller et al. [153] were the first to use 3D intraoperative fluoroscopy in combination with a head-mounted AR system. Fiducial markers were used for registration during pedicle screw navigation. Ma et al. [140] proposed another system for pedicle screw navigation. They used ultrasound for registration and an integral video overlay device and a half-silvered mirror to superimpose 3D graphics of the planning path and the spine. Their results showed a sufficient targeting accuracy of the system and a reduced radiation exposure.

3.2.3.2. Abdominal Surgery

One of the first AR experiments in the abdominal field was performed as early as 1992 by Bajura et al. [16]. They used an electromagnetic tracking system. Real world images

of a camera were superimposed with ultrasound images. Due to the limited computer power the superimposed images lagged behind and their resolution was too low. Onda et al. [165] developed an AR system for hepatobiliary and pancreatic open surgery (laparotomy). Infrared emitting diode markers were positioned at anatomical landmarks of visceral organs and tracked by an optical location sensor. The reconstructed 3D graphics were superimposed onto the real-time images of the rigid scope. The system provided an improvement in terms of safety, accuracy and efficiency during surgeries.

Kang et al. [97] developed a real-time stereoscopic AR system that can be used for laparoscopic surgery in 2013, consisting of a stereoscopic vision system and a laparoscopic ultrasound system. They use an optical tracking system and overlay live ultrasound images on a stereoscopic video in real-time. Müller et al. [154] developed an AR system to simplify the creation of a percutaneous renal access to treat kidney stones in 2013. Non-invasive fiducial markers are used for tracking and the preoperative CT image of the kidney is superimposed onto the video stream of the tablet computer positioned above the patient. The accuracy of the system is measured to be 2.5 mm.

In 2016, Mahmoud et al. [143] displayed preoperative anatomical models on the patient's body with a tablet PC using a visual simultaneous localization and mapping (SLAM) approach. For the registration no markers are used, but 4-6 anchor points need to be selected manually. They concluded that the system only requires minimal interaction and is easier to set up compared to marker-based systems. Lau et al. [124] developed an AR system that integrates ultrasound images and laparoscopic images in 2019. The overlaid images are shown to the surgeon on a screen. They use an electromagnetic tracking system. Their results show that their system allows the surgeon to correct depth and direction of resection without needing to look away from the laparoscopic screen.

3.3. Challenges and Limitations

Despite the great benefits AR offers, there are challenges and limitations that still need to be addressed for AR to become a standard in medical procedures. Difficulties identified in several studies include system latency that does not meet the AR criterion of real-time capability, inaccurate image registration, and insufficient depth perception [147]. Concerns regarding cost-effectiveness, the increasing amount of data generated by AR, and the increasing technical complexity of using an AR system are also discussed in the scientific community.

Deng et al. [53] and Watanabe et al. [223] mentioned in their studies that they struggled with system lags. When they moved their tablet from one location to another, they noticed a delay in updating the projected image on the tablet screen. Watanabe et al. stated that this delay was 400 ms and needs to be resolved in the future. Deng et al. observed that this delay causes an alignment error that increases with the speed of the tablet's movement. Kang et al. [97] also noted a time delay when using their AR system for laparoscopic surgery. This delay, however, was measured to be 144 ± 19 ms, which is small enough to allow surgeons to perform laparoscopic procedures smoothly. The problem of system latency is highlighted in previous publications. It is constantly being addressed through advancements in processing algorithms and parallel computing on CPU and GPU in conjunction with continuous hardware improvements [22].

The registration error is defined as the difference between the physical object and the superimposed image. It is important that this error is minimal so that the surgeon can rely on this information to make surgical decisions. Therefore, accurate registration is essential for a usable AR system and is still one of the biggest challenges in AR [129]. Especially in procedures involving complicated deformities and soft tissues, problems with registration frequently arise [5]. For example, in the study by Ma et al. [140] potentially occurring large registration errors are reported as a system limitation. They pointed to the low resolution of the ultrasound images they used for registration as a reason for this. Tabrizi and Mahvash [206] also referred to registration problems, mentioning that registration was not possible after a brain shift and that further developments of their system should allow automatic registration.

Moreover, there are difficulties in generating correct 3D and depth perception [129]. In AR-based surgery, the most common approach is to overlay a 2D image of the real scene with the 2D augmentation. This approach removes all depth perception cues such as occlusions and motion parallax [22]. This partially negates the advantage of AR in providing information to the surgeon in an intuitive manner. To circumvent this problem, several visualization processing techniques have been tested. One example is the use of specific color coding, where the color of the object changes depending on the distance from the camera. Similarly, the transparency of the colors can be changed the deeper the structures are located. When choosing a visualization technique to improve depth perception, it should be noted that crowding of the surgical image must be avoided [147].

With AR, it is possible to visualize a wide range of medical data at the same time. But this can also lead to a crowded view and be more distracting and misleading than helpful [40]. Therefore, the amount of data displayed to the surgeon must be balanced to avoid obstructing the surgical view. The information displayed should be useful and minimalistic. In addition, the surgeon should be able to decide whether to show or hide the displayed information. Switching between different sets of displayed information could also be useful [217].

Liu et al. [129] found that AR systems increase technical complexity compared to traditional systems. Different AR technologies such as headsets, tablets, complex tracking systems, voice commands, and gaze control are new potential failure points. Interruptions such as misunderstood commands or hardware failures can disrupt the surgical workflow. Surgeons run the risk of relying too much on the system and being unable to perform the surgery without it.

Cost is another factor addressed in several studies. The majority of AR systems developed for surgery are custom prototypes built with inexpensive equipment. These systems, however, are not suitable for widespread use [110] [147] [40]. Producing a well-tested, market-ready system would be costly. Consideration should be given to whether these costs are justified by the potential benefits of such a system. Chytas et al. [40] state that close collaboration between scientists, clinicians, and industry is needed to prove that cost-effective AR systems are possible.

3.4. Assisted Ventricular Punctures

Lollis and Roberts [132] use a robot to perform the ventricular puncture (VP). The VP is automated, but the planning of the trajectory is not. They claim that the operation

is successful on every attempt. Thomale et al. [208] suggests guidance assistance by smartphones or mini-tablets. The results have not yet been evaluated in a large number of patients. For patients with common sizes of ventricular systems, the assistance is valid. For extremely narrow ventricular systems, they recommend navigated positioning. It also takes trained surgeons to use the software, as the learning rate of the surgeon increases with multiple surgeries. The trajectory is also planned in advance. A further study with a similar approach is given by Sarrafzadeh et al. [186].

Azeem and Origitano [12] and Gil et al. [77] introduce frameless neuronavigational systems. Since the trajectory can be planned in advance but also be adjusted during surgery, these systems show very good results. The disadvantage of such devices is that they are very expensive. This method can be suitable for infants with very small VS.

Strowitzki et al. [204] demonstrated ultrasound-guided catheter placement. It results in a higher accuracy of catheter tip placement, but does not lead to fewer placement attempts. Gautschi et al. [72] compared three different methods: a non-guided version versus a neuro-navigated and XperCT-guided (Philips Medical Systems, Germany) method. Their results show that the guided methods improve the accuracy of the VP. The challenge is still the time required to plan the trajectory and process the images.

The use of additional navigation aids is acceptable for neurosurgeons if the normal surgical workflow is not extended more than 10 minutes [166].

3.5. Systems Comparable to the Proposed System

Van Gestel and Frantz et. al. [215] showed a similar approach to support a neurosurgeon during ventriculostomies with augmented reality. Frantz et.al. [68] presented a detailed evaluation of the possibilities of Vuforia marker tracking in neurosurgery. They show a system that tracks the patient with an infrared marker target and superimposes the patient models over the patient. No automated operation planning is performed and no customizable marker system is presented. In addition, no navigation aid or robot was used.

Li et.al. [127] presented AR-guided ventriculostomies utilizing a Microsoft HoloLens. No active patient tracking was used and no automated planning was performed.

Azimi et al. [13] proposed a system for ventriculostomies and tracking the patient with Vuforia markers. The marker to patient registration is done by attaching fiducials to the patients skull prior to the CT and registering them afterwards to the marker. No automated planning is performed and the system was not evaluated in a clinical environment.

4. Automated Operation Planning

This chapter presents methods for automated operation planning on the example of the ventricular puncture and tumor biopsies as depicted in Figure 4.1.

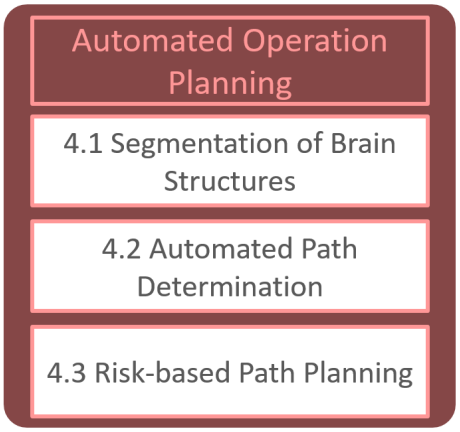


Figure 4.1.: Overview of chapter 4.

In Section 4.1 we introduce an approach to segment vital structures that are needed to automatically determine the puncture path. In Section 4.2 the segmentation results are used to automatically derive the path for the ventricular puncture. In Section 4.3 this approach is expanded by taking several risk structures inside the human brain into consideration.

4.1. Volumetric Segmentation of CT Head Images

Accurate and detailed information about the size, shape, and position of the region to be punctured is required to provide a detailed visualization to the surgeon. To perform the automated planning of the procedure and to determine the puncture paths, the ventricular system, the skin and the skull are used. These structures are automatically derived from preoperative CT images. Results of existing solutions did not yield the needed quality and puncture accuracy. Often large parts of additional structures around the ventricular system were also derived from the scans. Segmenting the ventricular system is not an easy task, because the boundaries are hard to determine in the CT scans, even for a medical expert. The question arises where the ventricular system ends and where the boundary and other structures of the brain begin. In addition, there is a wide variety of pathologies that also need to be considered. The challenge of segmenting CT scans is directly correlated to their low dynamic range and an unfavorable signal-to-noise ratio. CTs are scans of a real patient in discrete space and it is obvious that with a slice thickness of approximately 1 mm a lot of information from continuous space is not digitized. The solution and challenges are discussed in detail in the following chapter. The extraction of the ventricular system, the skin and the skull should be automated in a stream-lined process that requires as few manual steps as possible.

4.1.1. Related Work

The patient models must be segmented out of CT data, because this is very often the only image modality available, especially in emergency situations. This process is still considered to be challenging because the anatomy can greatly differ from patient to patient. Additionally, CT data has an unfavorable signal to noise ratio.

Point-based segmentation conducts a pointwise classification of each pixel. For each pixel, its intensity is compared with thresholds to determine if the pixel belongs to a segment or not. Multiple thresholds are used to divide the image into multiple segments. Selecting thresholds can be done manually or automatically. The multilevel Otsu method [167] can be used to automatically select the thresholds. The method determines the optimal thresholds for classifying each pixel in an image-based on a specific discrimination criterion, such as threshold variance between and within segments. Contour-based segmentation classifies pixels using an initial user-defined approximation that is automatically optimised. Kass et al. [99] interpreted a contour as an energy minimization problem of a curve on an image. The internal energy addresses the rigidity of the curve, commonly defined as a function of the derivative of the curve. The external energy uses image information to lead the curve to the wanted energetic minimum, often defined as an edge filter like the sobel operator. For examples of different possible definitions see [99]. Xu and Prince [228] minimize the chance of convergence to a local minimum by defining additional control forces or using alternative contour optimising techniques such as the gradient vector flow method.

Model-based segmentation performs the classification of pixels by using prior knowledge, like active shape models as described by Cootes et al. [42]. Annotated training data is used to create a geometric median shape, on which possible deformations are later performed. Using the median shape and performing its deformation using knowledge obtained by a principle component analysis [58], an object extraction in the image is achieved. This analysis is performed on the variance matrix of the shape that is created with the annotated training data. This method can be fully automated and produces excellent extractions for many cases. It performs well with low quality data and produces smooth results. The main limitation with this method is the amount of annotated data that is needed to train the model.

Qian et al.[171] used k-means clustering and selects the largest cluster that represents the ventricular system. A similar approach improves the segmentation using morphological operators [189].

Further, Chen et al. [36] have shown an extension to low-level segmentation techniques using template matching . Lundervold and Lundervold [137] also discuss the application of machine learning to perform the segmentation of medical datasets.

An early example of the use of gradient length is from a 1988 paper by Levoy [126]. In this paper, a simple function is used to assign opacity values to voxels depending on their intensity values and gradient lengths. Kniss et al. [108] also present transfer functions which are based on intensity values and gradient lengths. The user can use graphical user interface widgets on the two-dimensional histogram, which determine the visualization of the data. Lan et al. [123], on the other hand, deal with the improvement of such two-dimensional transfer functions based on scalar values and gradients (SG-TF for short). More precisely, it is a matter of eliminating the overlapping of different areas, which always occurs in such transfer functions. There also exist more complex one-dimensional

transfer functions, such as the procedure presented by Drebin et al. [57]. In this approach, the voxels are classified by probabilities based on their intensity values. Depending on this classification, color values are assigned to the voxels. The method is well suited to detect relatively large structures, but it is unclear whether the relatively narrow ventricular system would be detected. Another idea to segment structures based on spatial information is to use region growing. For example, Huang and Ma [87] present such a region growing method. The user can choose a point of interest in the volume, called the seed. All 26 neighbors of the seed are visited and a cost function comparing the corresponding value of the visited voxel and the seed voxel is used to decide whether they belong to the region or not. If they are part of the structure, their neighbors are also visited and all matching voxels are added to the region. This process repeats until all voxels are found or some other internal termination criterion is met. In comparison, Chen et al. [35] not only use a seed-based procedure, but also add a sketch-based procedure in front of it. First, the user selects a set of intensity values in the histogram that are of interest to them. Then they can draw and mark a region of interest directly in the volume. The program then cuts away all parts of the volume that lie outside the selected region. Now the user can set their seed as in the previously presented procedure. This makes it easier for the user, as they can get to their point of interest faster without having to iterate through various cross-section images first. Furthermore, it saves time for the user if not exactly familiar with the dataset and the region to be visualized. Correa and Ma [44], on the other hand, show in their work an approach based on the relative size of the features to be visualized. For this purpose, they use the so-called scale-space, which is calculated for the volume, in order to subsequently apply a transfer function based on size. This assigns color and occlusion to the corresponding sizes of the features of the volume.

In another publication, Correa and Ma [45] describe a method based on the occlusion of the voxels. For this purpose, they consider the environment of individual voxels and calculate the occlusion depending on this. The results are stored in a two-dimensional histogram in combination with the intensity values of the voxels. Another work by Correa and Ma [46] deals with transfer functions depending on the visibility of individual voxels. The visibility of each voxel is calculated depending on the viewpoint on the volume by accumulating the opacity from the camera viewpoint to the voxel. A histogram is then created over the visibility values. On this, a transfer function can be created to determine the optical visualization. These histograms were improved again by Correa and Ma [47] in an extended work. They introduced multidimensional visibility histograms that, for example, also take gradient length into account. Furthermore, they present two methods for computing viewpoint independent visibility histograms. One is an omni-directional visibility histogram, where the visibility of all possible viewpoints is calculated. Furthermore, a radial visibility histogram is calculated, where radial rays are used.

Tzeng et al. [212] use machine learning to represent structures of interest to the user. A neural network and a support vector machine are used. Volumes with highlighted regions serve as input for the method. The program then takes the intensity values, length of the gradients and intensity values of the neighbors of all highlighted voxels as input to find a meaningful segmentation. Soundararajan and Schultz [200] also present a method where the user can mark directly in the volume which areas are of interest. The work deals with the comparison of five different supervised classification techniques that have to classify the volume data based on the markings. Gaussian Naive Bayes, k Nearest Neighbor, Support Vector Machines, Random Forests and Neural Networks were compared, with

the result that Random Forests is best suited for the task. The researchers found that it was fast, robust, easy to use, and produced sufficiently good results.

Sereda et al. [193] build their work on the low-high (LH) histograms presented by Serlie et al. [196] and show how they can be used to classify objects. For visualization, Sereda et al. use a three-dimensional transfer function that uses the two LH values and the gradient length. The researchers then use region growing to identify structures even more clearly. Here, the cost function is based on the LH histogram. They describe this as significantly better than cost functions based on the intensity value and the gradient length, since edges can be detected better despite overlaps.

In a later work Sereda et al. [194] present a hierarchical clustering method. Here, in a set of clusters, the two that are most similar in the selected comparison procedure are always merged. A combination of two such comparison methods is presented. First, the spatial proximity is considered, where the direct neighbors of two clusters are counted. Second, proximity in LH space is examined. The boxes of the LH histogram serve as starting clusters.

Nguyen et al. [159] also present a clustering-based method. First, the gradients of the volume are calculated in a preprocessing step. Hong's method [85] was used for this purpose. The LH values are determined using Heun's method, a modified Euler method. Here, a weighting is calculated depending on the distance traveled during interpolation for the low and high values, respectively. A LH histogram is then generated from the LH values and their weights. The user can then choose between a two-step and a three-step clustering procedure. The result of the first two steps are clusters where all voxels have similar LH values and are also spatially close to each other. In the optional third and final clustering step, hierarchical clustering is performed. The user can then decide how many clusters are needed. Depending on this, the hierarchical clustering is reversed and the clusters are separated again until the desired number of clusters is reached.

4.1.2. Methods for CT Image Data Segmentation to Retrieve the Ventricular System

Each discrete layer l_n of a head CT scan can be defined by a matrix $G \in \mathbb{R}^{n \times m}$, where $g_{i,j} = I(i,j)$ representing the intensity of a certain pixel $p_{i,j}$. All layers of a CT together can be interpreted as a three dimensional volume as they are a pile of two dimensional images. CT scanners provide only intensity values representing Hounsfield Units. Therefore, they are visualized as grey-scale images.

The segmentation pipeline is composed of four successive stages, as illustrated in Figure 4.2. In the first step the subarachnoid space is removed to remove unfavorable areas of cerebrospinal fluid near the skull. This step is only used for extraction of the ventricular system. A very important step is the preprocessing of the CT data to reduce noise. Subsequently, voxels of interest are selected and in the last step appropriate clusters are identified. The last two steps are adapted to the specific structure to be segmented.

The CT scans are converted to the 'nrrd' file format without any change of the data itself. 'nrrd' represents a volumetric model that has a height (n), width (m) and depth (k). The coordinate domain is therefore defined as

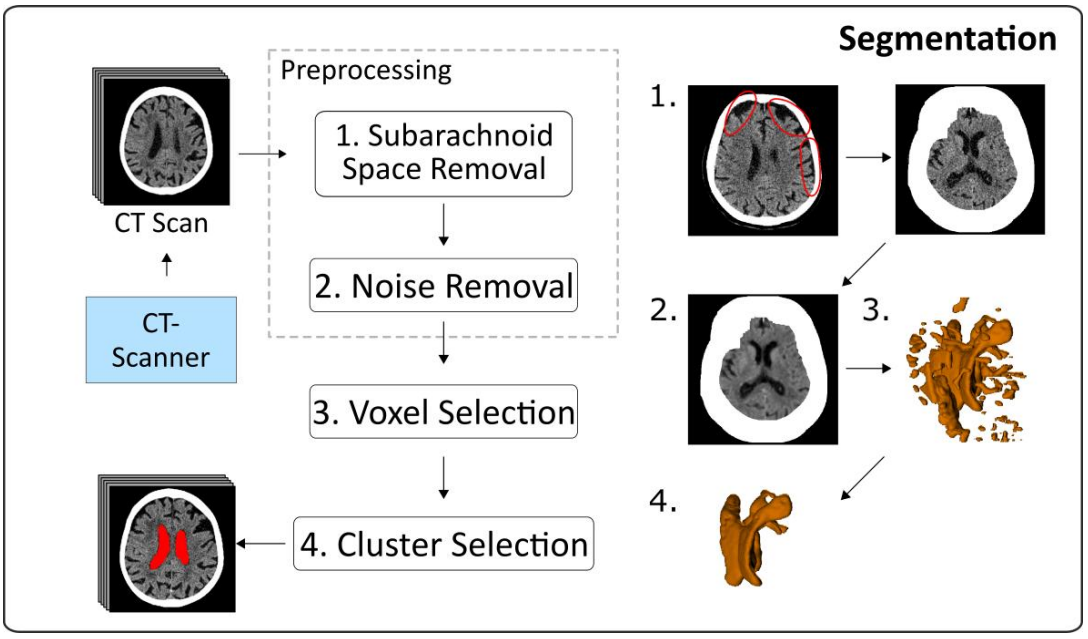


Figure 4.2.: Segmentation pipeline. Adapted from [119].

$$\mathcal{V}_{domain} = \{(x, y, z)^\intercal \in [1, n] \times [1, m] \times [1, k] \subset \mathbb{N}^3\} \tag{4.1}$$

We define the mapping function for the content of the region, the intensity values of the CT scan, as follow:

$$Map : \mathbb{N}^3 \to \mathbb{Z} \tag{4.2}$$

The region $\mathcal{V}$ can be defined as the graph of the function Map over $\mathcal{V}_{domain}$

$$\mathcal{V} = \{(v, Map(v))^\intercal = (v_{position}, v_{value})^\intercal \mid v \in \mathcal{V}_{domain}\} \subset \mathbb{N}^3 \times \mathbb{Z} \tag{4.3}$$

No alignment of the CT-data is needed. The position and rotation of the patient is not restricted by our approach. CT scanners produce their own coordinate system, and the patient can be in different positions in the scan. However, during the intraoperative assistance the holograms need to be registered to the patient in the HoloLens coordinate system.

In general there are two operators that can be applied to a voxel, a point-based operator and a local operator. The point-based operator is considering the individual voxel and its Hounsfield Unit. This is useful for the segmentation of large structures that have a very unique density, like bone. The local operator also takes the neighborhood of a voxel into account and can be used to segment more fractured structures like the ventricular system. This is also used for segmentation of the skin, as it is characterized by its neighborhood to air.

4.1.2.1. Subarachnoid Space Removal

Degenerative diseases of the brain can cause large bodies of fluid that accumulate between the brain and the skull, in the so called subarachnoid space. A safe area is defined as

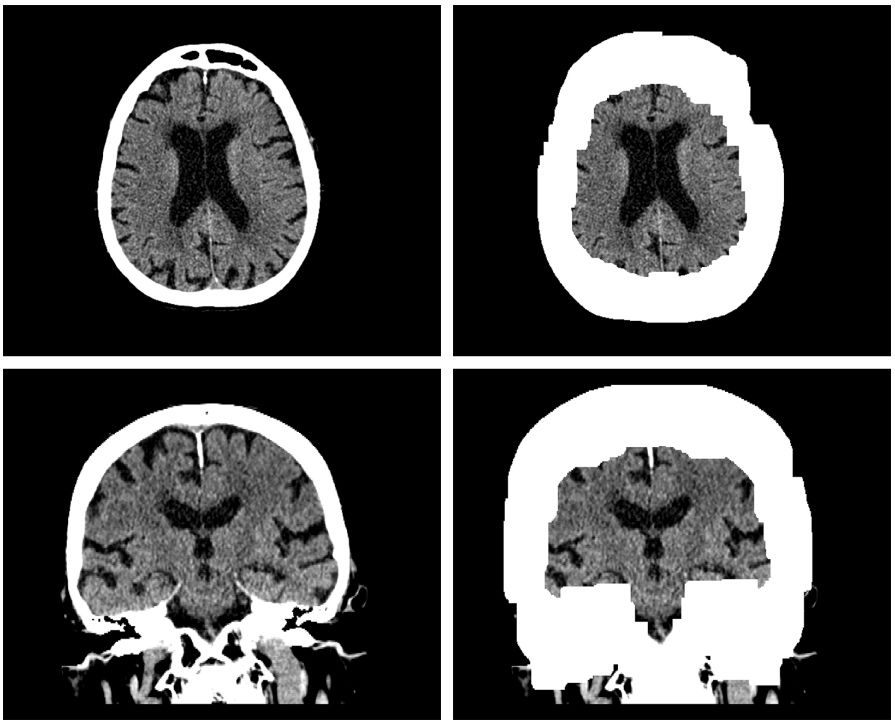


Figure 4.3.: Removal of the subarachnoid space.

large amounts of fluid near the skull are excluded. The presented approach takes medical knowledge into account. It is assumed that the ventricular system is the largest cluster of cerebrospinal fluid inside the human brain and that the ventricular system is approximately centered in the brain.

To remove these so-called subarachnoid space in the image, a bone-growing operator is used to grow the skeletal structure of the skull:

$$Bone_r^{local} : \mathcal{V} \to \mathbb{Z}, Bone_r^{local}(s) = \begin{cases} 900 & |\{n \in N_{s,r} \mid n_{value} \geq 900\}| \geq 1 \\ s_{value} & else \end{cases} \tag{4.4}$$

This approach can be processed in parallel and improves the overall results. It uses a local operator taking also the neighborhood $N_{s,r}$ of a voxel into account. The neighborhood $N_{s,r}$ is defined as a set of all voxels around a selected voxel s within radius r, the maximum distance between s and the surrounding voxels. It is possible to calculate the distance with different norms, e.g. the infinity or Euclidean norm. A radius of $r = 1$ voxel is chosen and the growing is performed for $\frac{max(n,m,k)}{30}$ iterations, where n, m and k are the number of voxels along each dimension of the nrrd volume. The number of iterations was determined empirically taking the size of the volume and the position of the ventricular system into account.

In Figure 4.3 an example of a subarachnoid space removal is shown that was generated utilizing the bone-growing operator.

4.1.2.2. Noise Removal

A bilateral filter is used to reduce the salt and pepper noise by smoothing the image data. The bilateral filter is a local operator and can be defined as follows:

$$
\begin{aligned}
&Bilateral_r : \mathcal{V} \to \mathbb{Z}, Bilateral_r(s) = \\
&\frac{1}{W} \sum_{n \in N_{s,r}^{\infty}} G_\alpha(|n_{value} - s_{value}|) G_\beta(\|n_{position} - s_{position}\|) n_{value} \\
&W = \sum_{n \in N_{s,r}^{\infty}} G_\alpha(|n_{value} - s_{value}|) G_\beta(\|n_{position} - s_{position}\|) \\
&G_\sigma(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x}{\sigma})^2}
\end{aligned}
\tag{4.5}
$$

The bilateral filter uses a Gaussian function G_σ for smoothing of intensities and spatial differences of an input image while preserving its edges. The filter is parameterized with the two parameters σ_α and σ_β. σ_α is the spatial parameter and σ_β controls the affected range. Other implementations of the bilateral filter use independent parameters. Here, we use a fitting parametrization of radius $r = 2$. We choose $\sigma_\alpha = r$ and $\sigma_\beta = 10r$ to simplify the calculation.

4.1.2.3. Voxel Selection

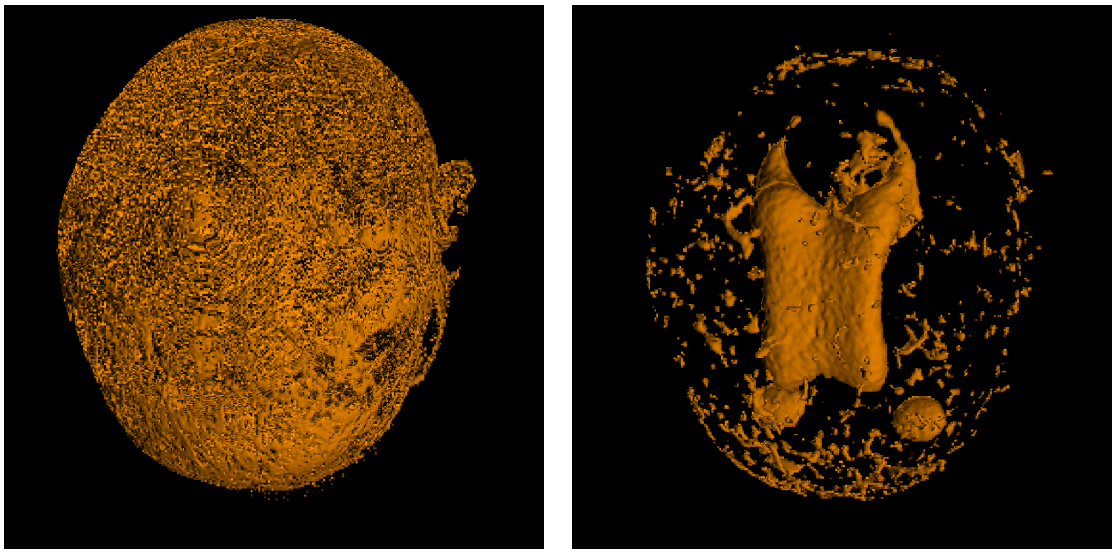
In this step, all voxels are identified that potentially belong to the ventricular system. The ventricular system is filled with cerebrospinal fluid, which has a Hounsfield Unit of 0 to 14, as described in Section 2.1.2, Table 2.1. This may vary for certain pathologies, for example, when the ventricular system contains blood. The challenge is that the complete brain is filled with small areas of cerebrospinal fluid as depicted in Figure 4.4 a). The figure shows the selection of all voxels that represent cerebrospinal fluid. The local operator makes a pre-selection of cerebrospinal fluid accumulations with a certain size. We select all voxel clusters with the desired Hounsfield Unit and create a binary mask for voxels potentially belong to the ventricular system as visualized in Figure 4.4 b). When comparing Figure 4.4 a) and b) it can be seen that the pre-selection leads to a significant lower number of possible clusters.

We apply the binarization function defined in Equation 4.6 to each voxel s in $\mathcal{V}$. The binarization function is a local operator that takes the neighborhood $\alpha N_{s,r}$ of a voxel into account. We choose $r = 2$ and $\alpha = 0.5$, which means that 50% of the neighboring voxels with radius 2 must have a Hounsfield Unit between the min and max values, 0 and 14, respectively.

The parametrization of $r = 2$ was chosen empirically and filters most of the small accumulations of cerebrospinal fluid.

$$
\begin{aligned}
&VS^{local}_{min,max,\alpha,r} : \mathcal{V} \to \{0, 1\}, \\
&\quad VS^{local}_{min,max,\alpha,r}(s) = \begin{cases} 1 & |\{n \in N_{s,r} \mid min \leq n_{value} \leq max\}| \geq \alpha |N_{s,r}| \\ 0 & else \end{cases}
\end{aligned}
\tag{4.6}
$$

All results are stored in the binary mask M_{CSF}^{mask} and written to a nrrd file. Selected voxels that could belong to the ventricular system are stored with value 1 in the mask and all remaining voxels are 0.



(a) **(b)**

Figure 4.4.: Difference between a point (a) and a local operator (b) to determine accumulations of cerebrospinal fluid. The clusters shown in (b) are the 1-clusters selected in the image data as described in the 'Cluster Selection' section.

4.1.2.4. Cluster Selection

After the removal of the areas filled with cerebrospinal fluid we assume that the ventricular system is the largest cluster that is left in the brain scan. For the ventriculostomy mainly the left and right large lateral ventricles are of interest. Therefore, we focus on the extraction of these two structures. It can often appear that the ventricular system is not represented as one connected cluster in the CT data, but as two separated ones.

Using Equation 4.6 voxels of larger groups of cerebrospinal fluid are identified and preselected. In a next step, connected voxels are grouped to clusters. A cluster can be described as an accumulation of linked voxels, where a path can be drawn from any voxel in the cluster to any other voxel without gaps and voxels can be passed more than once. This can be formalized by defining a discrete cluster using a path P and its connectivity. We define the connectivity with the help of the neighborhood $\mathcal{N}_1$ with radius $r = 1$, such that for each point $p_i \in P$ that is not a starting or end point, the following and previous points p_{i+1} and p_{i-1} are in the neighborhood of $r = 1$.

We can now establish a formal definition:

$$\begin{aligned} M \subseteq \mathcal{V} \text{ is a discrete cluster } \iff \\ \forall a, b \in M : \exists P = \{a = p_1, \cdots, b = p_n\} \subseteq M \text{ with} \\ \forall i \in \{2, \cdots, n-1\} : p_{i-1}, p_{i+1} \in N_{p_i,1}^{\infty} \end{aligned} \tag{4.7}$$

We can now define discrete 1-clusters in Equation 4.8:

$$M_{VS} \text{ is a discrete 1-cluster } \Leftrightarrow M_{VS} \text{ is a cluster and } VS^{local}_{min,max,\alpha,r} = \{1\} \tag{4.8}$$

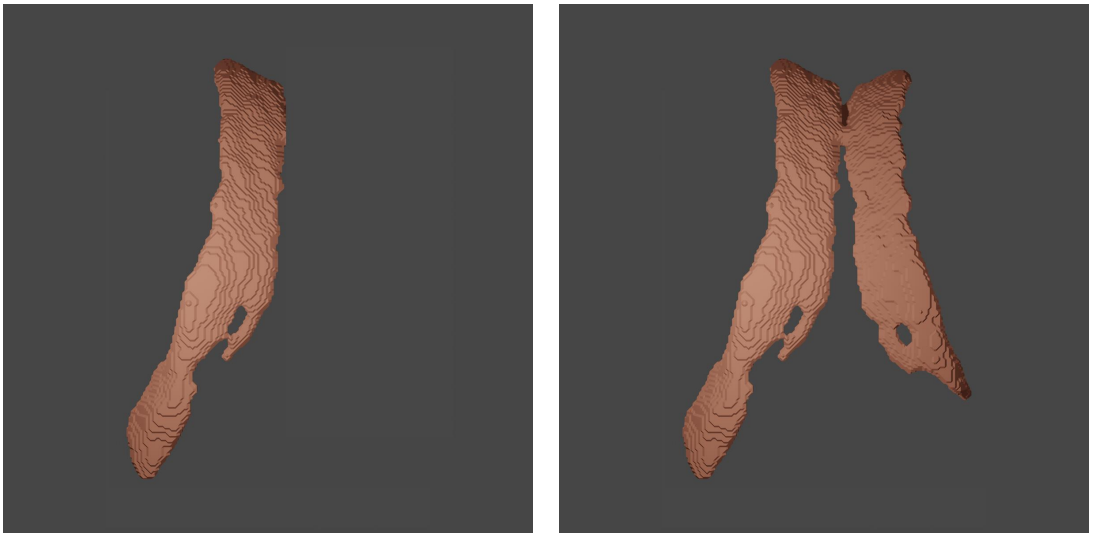
On this basis, all 1-clusters can be defined as follows:

$$C_1 = \{M_{VS} \subseteq \mathcal{V} \mid M_{VS} \text{ is a discrete 1-cluster}\} \tag{4.9}$$

With the medical knowledge we have, we can now select the 1-clusters of interest, which represent the ventricular system. Normally this is the largest cluster of all 1-clusters. It happens often that the two large lateral ventricles are separated in the CT image data due to noise. Therefore, all clusters of a certain size of the largest cluster M_{max} are selected, which is determined by counting the overall voxel count per cluster. The clusters are selected using Equation 4.10. The parameter $\beta = 0.5$ is chosen, which means that all clusters of 50 percent of the size of the largest cluster are chosen. Normally this selects the two lateral ventricles and sometimes the third ventricle as well.

$$C_{VS} = \{M \in C_{VS} \mid |M| \geq \beta |M_{max}|\} \tag{4.10}$$

Figure 4.4 b) shows all grouped 1-clusters, while in Figure 4.5 a) the largest cluster and in b) all clusters of 50% of the size of the largest cluster are visualized.



(a) **(b)**

Figure 4.5.: (a) Selected left ventricle, when just the largest cluster is taken, b) left and right ventricles are extracted, when all clusters of 50% size of the largest cluster are selected.

4.1.2.5. Challenges

In the following subsections challenges in the segmentation of the patient imaging data are presented and discussed.

Large Accumulations of Fluid in the Subarachnoid Space

There can be large accumulations in the subarachnoid space between the skull and the brain as depicted in Figure 4.3. In a healthy brain, the ventricular system is the largest cluster filled with cerebrospinal fluid. In the described case this can change so that the subarachnoid space becomes the largest cluster, which is challenging to the here presented approach. A solution is presented in Section 4.1.2.1.

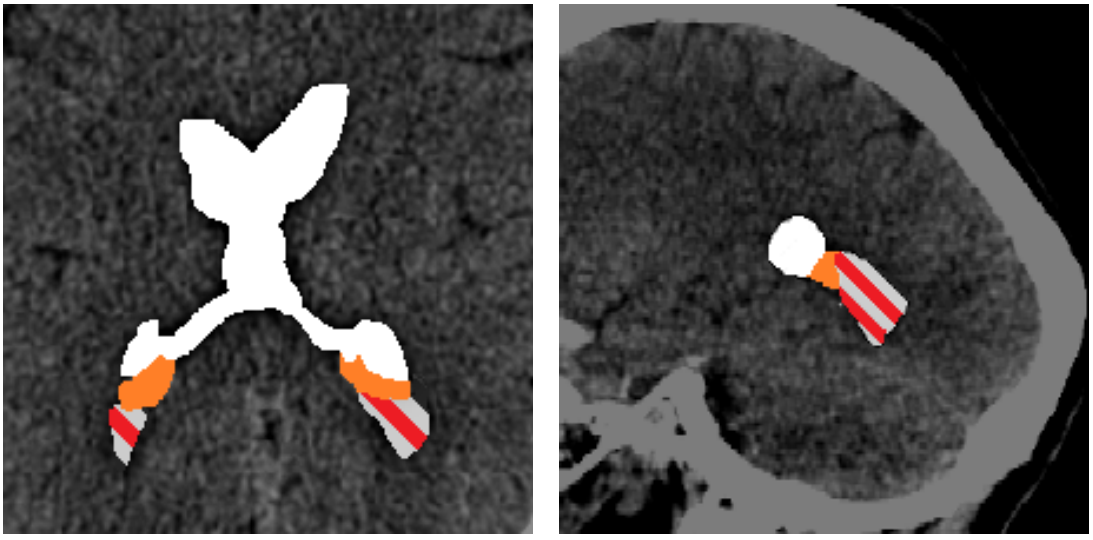
Blood Clots Inside the Ventricular System

Due to certain pathologies like intracranial hemorrhage or stroke, and others, there can be blood clots inside the brain that can also accumulate in a damaged ventricular system. This can change the Hounsfield Unit of the areas to be segmented. A solution is presented in Section 4.1.6.

Calcification of the Choroideus Plexus

All four ventricular systems have one *choroideus plexus*. This structure is responsible to create new cerebrospinal fluid. Due to certain pathologies these areas can calcify. These calcified areas have the same HU as bone, which can lead to the situation that the segmentation algorithm stops and these areas are classified as their own cluster. The calcified regions are located in the last quarter of both large lateral ventricular systems one and two and prevent the ventricular horns from being segmented. However, these areas are not of importance for the ventricular puncture and can therefore be neglected, because the catheter is placed in the first quarter of the ventricle.

An example of a ventricular system with calcified regions is depicted in Figure 4.6 a and b.



(a) **(b)**

Figure 4.6.: Calcified areas inside the ventricular system in orange. Cut-off areas of the horns striped in red/white. a) Top-down view. b) Side view.

Large Difference in Size of the Two Lateral Ventricles

The proposed method of this work selects the largest cluster and the second largest cluster

of a certain size of the largest one. The standard parametrization of the size parameter β in Equation 4.10 is 0.75, as it takes all clusters that are 75% of the size of the largest cluster. Problems occur when one lateral ventricle is much smaller than the other one. When this happens a neurosurgeon needs to adjust the value for β manually. An example is shown in Figure 4.4 a) and b). When just clusters of 75% size of the largest cluster are selected only the left ventricle is taken. Changing the parameter β to 0.5 leads to a correct selection of both lateral ventricles.

4.1.3. Evaluation of the Volumetric Segmentation

The proposed method for the automated segmentation of the ventricular system was evaluated on 26 randomly selected CT datasets, with and without pathologies. Results of the 26 segmented patient scans are shown in Figure 4.7.

4.1.3.1. Qualitative Evaluation of the Segmentation Results

For qualitative evaluation of the extraction results, medical experts (n=4) were asked to review the results and complete a questionnaire. The experts were asked to investigate all 26 extracted ventricular systems. First the results were visualized in a custom 3D viewer as a volumetric model where it was possible to change the view. Then the experts used MITK to review the extracted regions slice-wise as a mask overlay of the original CT data. In this way, a very accurate pixel-wise evaluation was possible. After the review of the CT scans, the experts were asked to complete a questionnaire. They were asked to rate the quality of the results according to the following questions:

1. **How good is the quality of the segmentation on a scale from 1-5?**
2. **Are all relevant areas included?**
3. **How is the quality of the surface details of the extracted region?**
4. **What problems exist?**

The survey was conducted on the basis of the 26 dataset extractions that are depicted in Figure 4.7. At this stage most of them were segmented correctly, still some challenges remained, e.g. segmentation of blood clots. For the first question the experts were asked to provide a rating from very good (5) to very poor (1). The other questions could be answered freely.

Question 1: The quality of the segmentation was evaluated as very good by the experts. They provided a rating of 4.5/5 points on average.

Question 2: All relevant areas are included in 24 of 26 CT datasets. For some, it was necessary to adjust the second cluster selection to a lower percentage of the largest cluster. One challenge that remained were ventricular systems with blood. The problem was solved by an expansion shown in Section 4.1.6. For the quantitative evaluation presented in Section 4.1.3.2 this problem was solved and the results are based on the adapted version.

Question 3: The surface of the ventricles were rated as too irregular extracted, which could be improved in the future. The reason for this is the low resolution of the CT scans. The

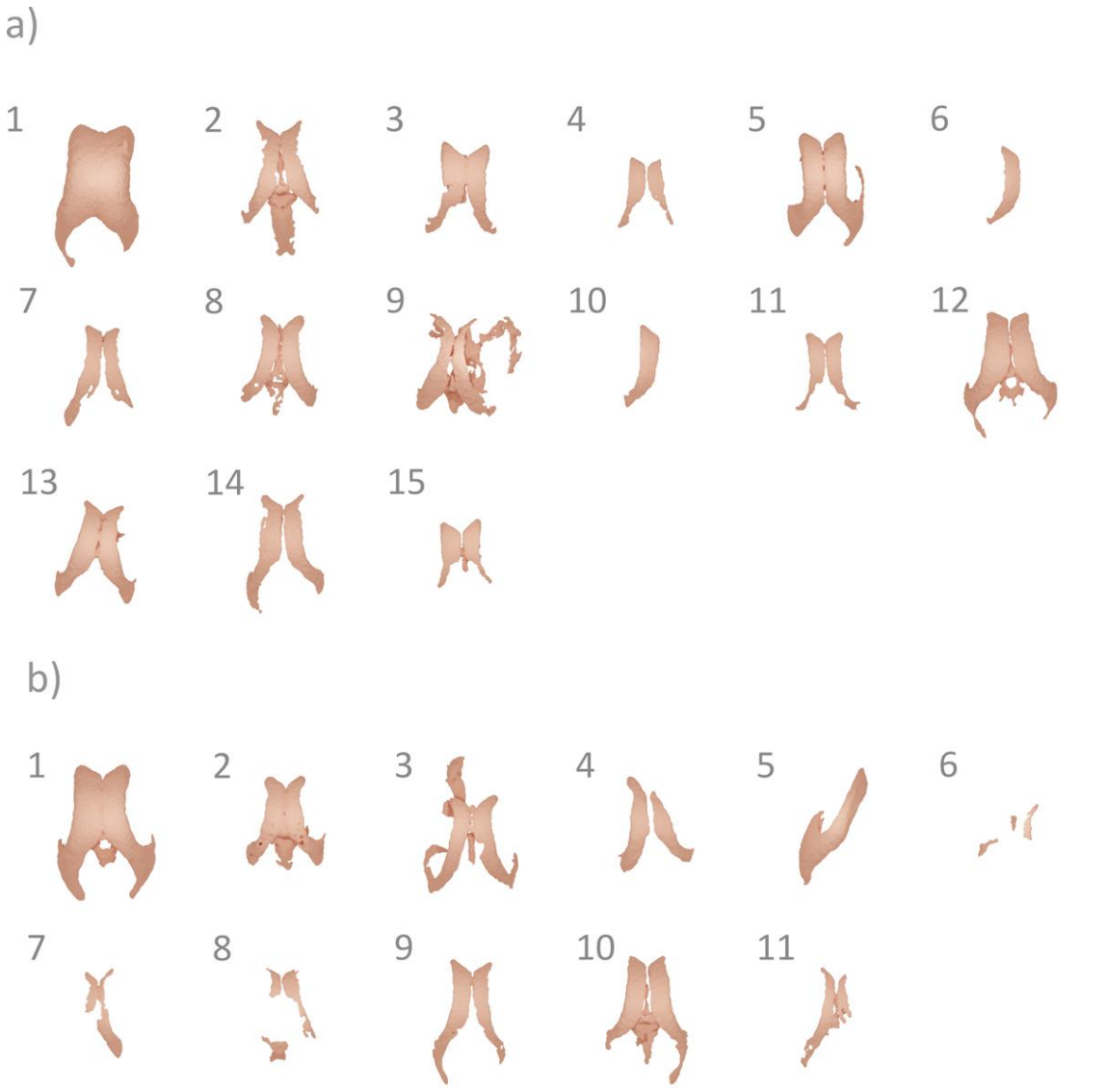


Figure 4.7.: Segmentation results of 26 CT scans: a) Example of extracted ventricular systems without pathology, b) Ventricular systems with pathology (1,2: Cerebral atrophy, 3: Blood reabsorbed post bleeding, status post endovascular coiling of an aneurysm, 4: Deformation due to hemorrhage, 5: Prominent midline shift, 6: Hydrocephalus intraventricular blood clot, status post endovascular coil, 7: Midline shift, 8,9,10: Slim ventricles, 11: Slim ventricles due to chronic subdural hematoma. Adapted from [113].

Table 4.1.: Comparison between automatic and manual segmentation of (a) the complete and (b) the partial ventricular system.

	VS1		VS2		VS3		VS4	
	(a)	(b)	(a)	(b)	(a)	(b)	(a)	(b)
F1-Score	0.93	0.95	0.91	0.99	0.85	0.99	0.84	0.99

data is visualized without interpolation, which would lead to a higher resolution. When creating surface models a more detailed surface could be interpolated.

Question 4: It is challenging to segment ventricular systems with certain pathologies. Just looking at the segmentation result is sometimes not enough to understand the situation completely. Therefore, it is necessary to check the 2D slices of the CT scan. Our system works fully automated. Some pathologies make it necessary to manually adapt some parameters. The largest challenge remained ventricles with blood clots. The problem is that blood or other liquids entering the ventricular system due to injuries or strokes change the density of the liquid inside the ventricular system. The extracted regions are sufficiently suitable for an AR visualization.

4.1.3.2. Quantitative Evaluation of the Segmentation Accuracy

To evaluate the approach in regard to its segmentation accuracy the generated results are compared to manually segmented head CT scans. The manual segmentation was done by two expert neurosurgeons. One challenge in segmenting the ventricular system is calcifications at the posterior areas, which divides it for the algorithm into two separate clusters, as depicted in Figure 4.6. However, the intervention is performed on the frontal regions of the ventricular system, which are segmented sufficiently well by the proposed method. The segmentation errors of the ventricular system due to calcifications can be neglected for use in our system. We evaluate the segmentation quality for the complete ventricular system and only relevant sections excluding the posterior sections. The Dice Coefficient (F1-Score) is determined to perform a quantitative analysis. The ground truth is determined through manual segmentation and compared to the algorithm results.

The results are depicted in Table 4.1.

Most calculations of the proposed method can be performed massively parallel on a GPU. The extraction of the ventricular system was tested on two different systems on all 26 CT datasets. The results of the evaluation are depicted in Table 4.2. The datasets were processed 5 times in a row to get the average calculation time. The segmentation has a runtime of 2.3 ± 0.2 s per structure, which makes it a very fast and reliable algorithm.

Table 4.2.: Performance of our implementation on different systems.

CPU	GPU	RAM	OS	Avg. Time	Used RAM
Intel Core i7-8700K	Nvidia GeForce 1070 Ti	32GB DDR4	Ubuntu 16.04 LTS	2.3 ± 0.2 s	500MB
Intel Xeon E5-2670	Nvidia GeForce 1060	16GB DDR3	Ubuntu 18.04 LTS	3.8 ± 0.3 s	500MB

4.1.4. Segmentation of the Skin

The proposed method is also applicable for the segmentation of the skin and the skull. The segmentation of skin is challenging, due to the large HU range from approximately -750 to 200. Other tissue types lie within the same range. The use of just point- or region-based operators for the selection of the skin would also select the entire brain matter, blood vessels, blood, cerebrospinal fluid and other. The skin has the property that it is the most outer layer of the human body, which means that it is always in the vicinity of air. This information can be used to implement a new filter taking the neighborhood to air into account. The basic idea is to select voxels in direct neighborhood of air. Voxels that represent air can easily and reliably be identified in a CT scan due to its unique HU of -1000. Equation 4.6 is therefore adapted and expanded to:

$$Skin_r^{local} : \mathcal{V} \to \{0,1\}, Skin_r^{local}(s) = \begin{cases} 1 & s_{value} \in [-750, 200] \wedge |\{n \in N_{s,r}^{\infty} \mid n_{value} \leq -800\}| \geq 1 \\ 0 & else \end{cases} \tag{4.11}$$

Again, s represents the selected voxel, and we iterate over the complete volume. s_{value} is the value of s, its intensity represented as HU. $n \in N_{s,r}$ are neighboring voxels of s and n_{value} their intensity values. We select all voxels s that are in a neighborhood of $n_{value} < -800$. A lower value of -800 was chosen because voxels near air merge with skin voxels in the scan data. For the calculation of the neighborhood, a radius $r = 3$ is chosen as the skin thickness is approximately 2 voxels.

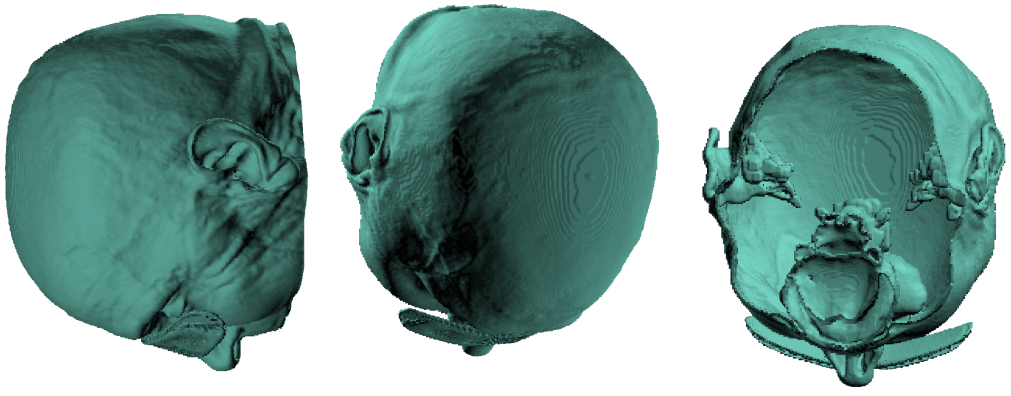


Figure 4.8.: A skin extraction from a head CT scan using a local operator.

For the segmentation of the skin no preprocessing of the CT data is needed and no cluster selection must be performed. Looking at Figure 4.8, also areas inside the head are extracted. These are the skin tissue of the ear canal and the sinuses. These structures are neither required for this work, nor is it a problem that they are displayed. Additionally, these structures represent areas of risk that should be avoided. Especially the ear canal is problematic. When punctured on the way to a target it can contaminate the puncture canal and lead to an infection.

The segmented skin is later used in Section 5.5.1.4 for the registration of the patient to the marker system.

4.1.5. Segmentation of the Skull

The extraction of the skull is needed as an input for the algorithms that automatically determine the puncture trajectories presented in Section 4.2. The segmentation of bone is the most simple one, because bone has a very unique HU of 300 to 1900. A simple point-based operator can be used to perform the selection of the appropriate voxels. We also do not need any filtering or cluster selection.

$$Binary^{point}_{min,max} : \mathcal{V} \rightarrow \{0,1\}, Binary^{point}_{min,max}(s) = \begin{cases} 1 & 300 \leq \hat{s}_{value} \leq 1900 \\ 0 & else \end{cases} \tag{4.12}$$

Here we use a point-based operator only taking the selected voxel into account. The values for $min = 300$ and $max = 1900$ were chosen appropriate to the HUs. A segmented skull is depicted in Figure 4.9.

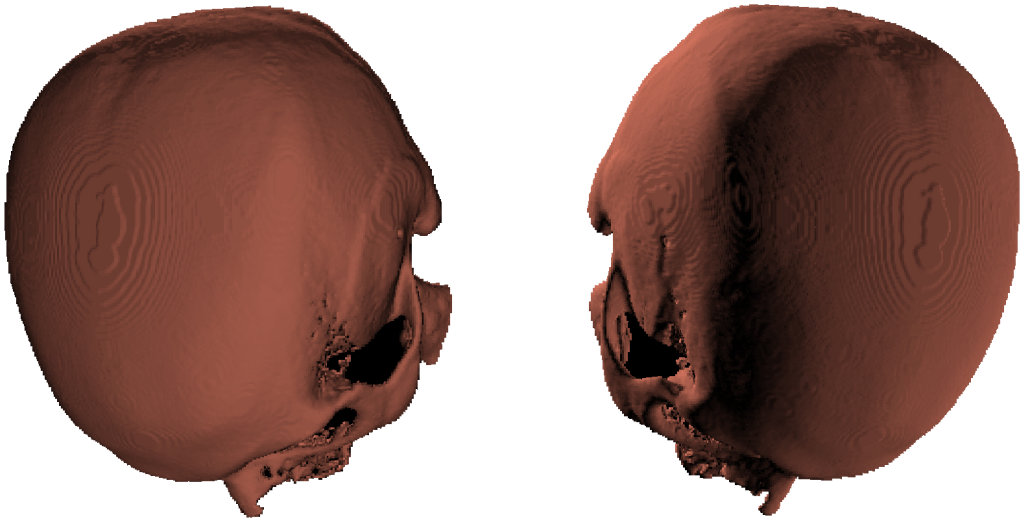


Figure 4.9.: Exemplary extraction of a skull from a patient head CT using the point-based operator.

4.1.6. Segmentation of the Ventricular System with Blood Clots

In brain injury, like hemorrhages, diseased ventricles may contain blood, so that they cannot be derived with a single HU range. For segmentation of the correct voxels, the range of cerebrospinal fluid $[0, 14]$ and additionally a range of $[60, 80]$ that represents blood infused cerebrospinal fluid is used. This leads to the intervals $I = [min_1, max_1] \cup [min_2, max_2] = [0, 14] \cup [60, 80]$ which leads to a correct selection of the desired voxels.

For the segmentation of ventricular systems containing blood Equation 4.11 is adapted to the following equation:

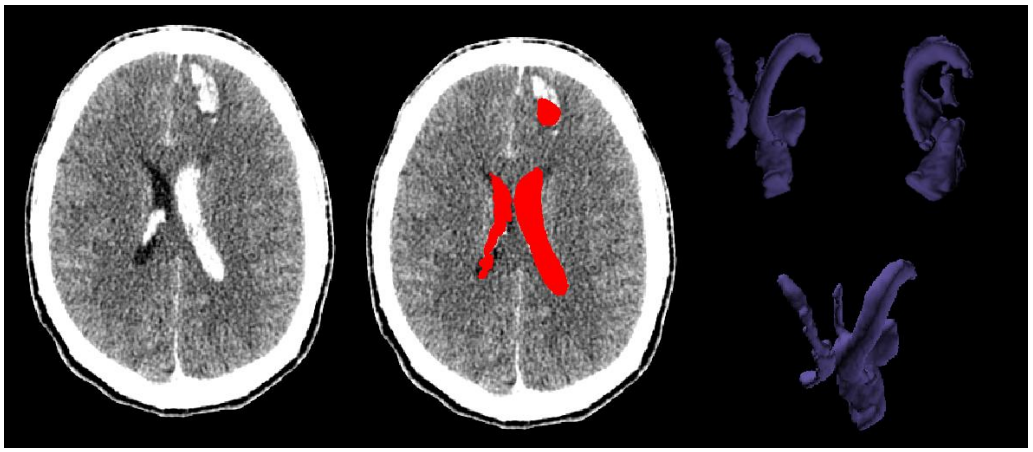


Figure 4.10.: Segmentation of a ventricular system with blood clots.

$$VS^{blood}(s) = \begin{cases} 1 & |\{n \in \mathcal{N}_2(s) \mid n_{val} \in I\}| \geq \alpha|\mathcal{N}_2| \\ 0 & else \end{cases} \tag{4.13}$$

It should be noted that the complete blood infiltration is selected. A surgeon needs to examine the segmentation result carefully. An extraction of a blood infused ventricular system is shown in Figure 4.10.

4.1.7. Discussion

We have presented an approach that is able to extract the ventricular system, the skin and the skull from head CT scans with appropriate designed local or point-based operators. We therefore take the neighborhood of a voxel into account, except for the extraction of the skull. The presented method has a very fast runtime of just a few seconds due to its execution on the GPU. Several pathologies can lead to erroneous segmentation results. This can be the case for intraventricular blood clots (hyperdens), subdural collections with consecutive brain shift and deformations of the ventricular system. The methods were implemented in two stages, while the second introduced expansions to solve the challenging scenarios. In some cases one ventricle is much smaller than the other one. This leads to a segmentation where just one ventricle is found in the cluster selection step. An example can be seen in Figure 4.5. This case was identified as a challenge and discussed in Section 4.1.2.5. A manual parametrization of the algorithm is necessary in such cases. The parameter β in Equation 4.8 must be parameterized differently, e.g. with 0.25, to find the second cluster correctly.

The most challenging situation occurs when the ventricles are infused with blood. We have presented an expansion of our approach to correctly segment the ventricular system and blood clots as a whole. This comes at the cost of loosing automation so that not every case is covered fully automated. A medical expert verifies the current case and can easily start an adapted version of the segmentation as presented in Section 4.1.6. Due to the very fast implementation, this takes 2-3 seconds per structure.

We performed a qualitative and quantitative analysis of the presented approach. We reach a F1-Score of 95-99% for the relevant areas. CT scans are limited in their resolution of

approximately 1 mm slice thickness. Additionally, they have salt and pepper noise. To overcome this problem preprocessing is an important step to generate good segmentation results. The medical experts stated that the surface of the ventricular system was too irregular. Before the volumetric extractions are used as models in the augmented reality system they are converted to surface models to realize a proper rendering on the HoloLens. It is possible to smooth the surface, but this needs to be carefully executed, because we do not want any additional data points that are not present in reality.

We presented solutions to all identified challenges in segmenting the ventricular system. In addition, we expanded the proposed method to segment the skin and the skull. We were able to use a local operator considering the neighborhood of air for skin segmentation. The skull can be segmented using a point-based operator. An improved version capable of segmenting ventricular systems with blood clots was also demonstrated.

The segmentation time of any structure is 2.3 ± 0.2 s. The runtime does not depend on the structure to be segmented, since all voxels in the CT scan must be examined for each structure. The time needed for structure segmentation depends mainly on the resolution of the CT scan. All three structures can be segmented in 6.9 ± 0.6 s.

4.2. Automated Path Planning

In the previous section, methods for automated determination of the skin, skull and ventricular system are presented. The extracted models can be visualized to support a surgeon during an intervention. A surgeon can see the underlying skull of a patient and determine the Kocher's points with this additional information. The superimposed else hidden ventricular system can help to target a specific point. For further support, the visualization of the optimal points could help to improve the accuracy of the catheter placement and lead to a higher puncture success rate, from which the patient would benefit. It is desirable to determine these points automatically to always get an optimal solution. In this section, an automated process for puncture path determination is presented that includes the determination of the entry and target points.

4.2.1. Related Work of Autonomous Neurosurgical Planning Methods

There are only a few papers that address the automatic planning of ventricular punctures. Beckett et al. [18] perform the trajectory planning for ventricular puncture. Their approach uses classic pattern recognition tasks. They use a small annotated dataset of CT data with labeled trajectories. This dataset is used as the base to create dataset templates for different rotations. Using a threshold, the skull is extracted out of the CT scan. For each template the sum of squared differences (SSD) is calculated. To perform faster trajectory planning, they use two classes of templates. For alignment of the skulls the template with the smallest SSD is used. Two registrations are performed, first a linear and a non linear afterwards. The Kocher's point (K) and the target point in the VS are mapped to the CT scan once the registration falls below a certain threshold using the root mean square (RMS) metric. When the first class of templates is insufficient, the second class of templates is tried. They also evaluate the length of the trajectory, which is between 6 and 7 cm. The authors determine the Kocher's point so that an approximately perpendicular insertion ends in the intended target point. They report results of 99% reliability on scans without pathologies, and of 95% for intracranial hemorrhage. However, the authors do not clearly explain the technical details how the mapping was performed, e.g. the calculation of the error metric and the exact matching procedure. Rehman et al. [174] validate that the correct orientation for the placement of the catheter is perpendicular to the Kocher's point.

4.2.2. Methods for Automated Determination of the Puncture Path

The process of the automated puncture path determination is depicted in Figure 4.11. The segmented patient models created using the method described in the previous Section 4.1 are taken as input. The puncture path is generated based on this patient-specific data by identifying the Kocher's points on the skull of the patient and favorable target points inside the VS. To find these target points statistical shape models (SSMs) are utilized.

In the context of this work, anatomical structures are considered. Each of these structures can be regarded as a shape model. We consider a shape to be all geometrical information

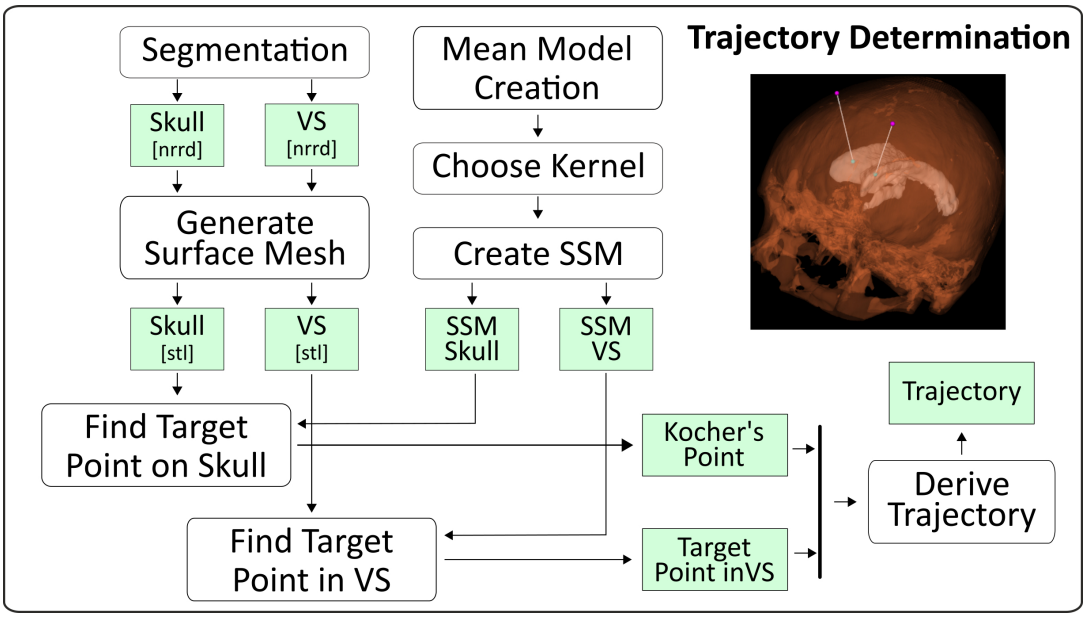


Figure 4.11.: Process that depicts the proposed method for the automated path planning. Adapted from [119].

of an object that remains when location, scale and rotational effects are filtered out [139]. Shapes can be grouped to shape families or classes. For example all ventricular systems form a shape family. We neglect the size for our shape family definition, because the size of a shape often correlates to a certain shape family. A good example is the anatomy of children in comparison to adults. The anatomy of children differs in many ways, so it makes sense to define own shape families for them. In this work we only consider adult humans.

The SSMs are generated by the creation of a mean model in combination with a kernel. Within the statistical shape models of the ventricular system and the skull the optimal target points are defined. The shape model fitting adjusts the statistical shape models to the current patient-specific anatomy. The target points of the SSM are moved accordingly to the shape fitting. The path is then derived as the vector from the Kocher's point to the target point inside the ventricular system.

4.2.2.1. Creation of the Statistical Shape Model

A statistical shape model is created in three steps. First a mean model is generated out of sample data. In a second step a kernel is chosen to represent the shape deformations. In the last step, the mean model and the kernel are combined to form the statistical shape model.

Preprocessing - Generation of Surface Models

The surface meshes of the ventricular system and the skull are generated from the volume models that have been created using the proposed methods in Sections 4.1.2 and 4.1.5 from real patient data. The marching cubes [134] algorithm was used to generate *stl* surface models from the volumetric segmentations. All 26 CT datasets were automatically segmented and manually validated. For each dataset the ventricular system and the skull

were obtained. The surfaces of the ventricular systems were smoothed, to retrieve more generalized models. A stronger smoothing was used than for the surface models used in the intraoperative assistance. The segmentation results of the skull were taken without smoothing, because the surfaces were smooth enough and extracted without outliers.

General Process to Create a Statistical Shape Model

Partial information presented in this paragraph are taken from the online course by Luethi et al. [136]. To create a statistical shape model a gaussian process $GP(\mu, k)$ is derived from sample data. All possible shape variations of a shape class are modeled through this process. The gaussian process is defined through a mean function $\mu(x)$ and a covariance function $k(x, x')$ that is derived from the shape deformations $U = u_1, ..., u_j$ found in the dataset. The shape deformations U are vector fields that represent the deformation in relation to the reference shape. Any input shape can be selected as the reference shape. Normally, simply the first mesh is used.

The mean is calculated using the following equation:

$$\mu(x) = \overline{u}(x) = \frac{1}{n}\sum_{i=1}^{i} u_{\mathrm{i}}(x) \tag{4.14}$$

The covariance function is estimated using

$$k(x, x') = \frac{1}{N-1}\sum_{i=1}^{N}(u_i(x) - \bar{u}(x))(u_i(x') - \bar{u}(x')) \tag{4.15}$$

The mean function and the covariance function are then used to define the gaussian process $g \sim GP(\mu, k)$.

The statistical shape model can then be defined as

$$S = \overline{x} + g(x) \tag{4.16}$$

Mean Model Creation

Given the general process for creation of a statistical shape model the mean shape models of the ventricular system $\overline{x}_{vs}$ and the skull $\overline{x}_{skull}$ are generated from $n = 26$ surface models from real patient data using the following equations.

$$\overline{x}_{VS} = \frac{1}{n}\sum_{i=1}^{n} VS_{\mathrm{i}} \tag{4.17}$$

$$\overline{x}_{skull} = \frac{1}{n}\sum_{i=1}^{n} Skull_{\mathrm{i}} \tag{4.18}$$

Kernel Selection and Parametrization

The ideal way of constructing a kernel is from a large number of sample data. In our case only 26 datasets where available, which is not sufficient to get a kernel that is reflecting all shape variations. For this reason it was decided to use a standard gaussian kernel. The challenge is that the covariance matrix that is generated by Equation 4.15 can become very large when using meshes that have a high resolution, which makes it impossible to compute them on a computer. Therefore a low rank approximation of the gaussian process is estimated using the Karhunen-Loéve expansion [98] [131] [92].

To reduce the dimensionality of the data, the gaussian process can be written as

$$g = \mu + \sum_{i=1}^{n} \sqrt{d_i}\varphi_i\alpha_i,\ \alpha_i \sim N(0,1) \tag{4.19}$$

where n is the number of sample shapes used to derive the gaussian process.

The basic concept is that any shape that is modeled by the gaussian process can also be derived by taking the mean shape and adding up a restricted number of eigenpairs (eigenvalues mulitplied with its eigenvectors) $\sqrt{d_i}\varphi_i$ multiplied by a coefficient α_i. The coefficient α is thereby probabilistic and following a standard normal distribution. It represents exactly the same shape class as the gaussian process but can be processed on a computer.

Any new shape g_{new} is defined only through the chosen coefficients α. The decomposition of the gaussian process into its Karhunen-Loéve representation is done using the Principal Component Analysis (PCA). The covariance function is often referred to as the kernel.

Any new shape deformation x_{new} of a shape class can be described by a valid parametrization of Equation 4.20.

$$x_{new} = \overline{x} + \sum_{i=1}^{d} \alpha_i G_i \tag{4.20}$$

This is basically the execution of the low rank gaussian process. $\overline{x}$ is the according mean model and G_i is the decomposed covariance function and α the coefficients of the shape model [25]. The deformation field of the model is defined through the number d of chosen basis vectors [138]. They are also referred to as the degrees of freedom of the shape model. The higher the number of the basis vectors the more shape variations are possible.

For the shape model of the skull $d = 50$ and for the ventricular system $d = 200$. The ventricular system has a much larger variety of possible shapes. The gaussian kernel is parameterized with $s = 10$ and $\sigma = 40$. The parameter s is affecting the variance of the possible deformation. The parameter σ takes effect on the smoothness of the vector field. Normally shapes do not change the direction abruptly, σ is a radius in which smoothness is applied.

For the skull a symmetrized kernel was chosen, as usually the left and right sides are symmetric with only small deviations. For the ventricular system this is normally also the case but can vary a lot due to certain pathologies or anatomical specialties. Therefore a non symmetric kernel was chosen for the ventricular system.

Kocher's and Ventricular System Target Point Definition

In the last step to complete the creation of the statistical shape model the Kocher's points and the target points in the ventricular system are defined.

The optimal target point lies within the ventricular system. However, it is not possible to define a point that is not part of the mesh. The point on the surface of the ventricular system is chosen in a way that it lies directly in the vector between the Kocher's points and the target point inside the ventricular system. When the point on the ventricle is derived within the fitting process the vector between that point and the Kocher's point is enlarged until the other side of the ventricle is hit. Then the middle of these points is taken as visualized in Figure 2.4.

4.2.2.2. Shape Model Fitting

In the following the process step of the model fitting is described in detail. The shape model fitting takes as input the segmented patient models of the ventricular system and the skull. The SSMs of the ventricular system x_{vs}^{ssm} and the skull x_{skull}^{ssm} are deformed according to their parametrization of the kernel to fit the current patient's anatomy. The defined target points on the SSMs are moved during this fitting process to receive the target points on the skull and on the ventricular system to automatically retrieve the optimal puncture trajectory.

Preprocessing

Any surface model defines its own coordinate space and has a certain translation, rotation, scale and deformation. When a SSM is fitted to a certain shape model of the same class all of these four aspects need to be aligned correctly. The scaling needs to be chosen correctly when the SSM is created. Translation and rotation are aligned during the shape model fitting and in a last step the deformation is fitted. The more these four aspects of SSM and target shape differ, the more difficult it gets to fit them. The scaling of a SSM is determined by the data it was created from. In our case these were the above mentioned 26 patient scans. The SSMs are slightly enlarged for later shrinking over the target shape like a vacuum bag.

To perform a successful fitting the SSM and the target shape need to have a good inital matching. If this is not the case the algorithm often runs into a local minimum instead of finding the global minimum, equaling the best fit of both meshes. The SSMs are generated in a way that the coordinate source is in the center of the mean ventricular system and the skull. The patient models are also extracted with coordinate source in the center. The proposed approach for patient model extraction is invariant to the used coordinate space. The patient scan can have any orientation and translation in space. However, the extracted patient models are oriented in the same way as in the patient scan. Also the scanned area differs from scan to scan, sometimes the scans are cut under the nose, others under the forehead. In this work, it is assumed that the patient scans are manually oriented correctly and that they always image the same area of the head. The coordinate system of the target shape (patient model) is kept. The SSMs are translated into this coordinate space.

Fitting Step

The fitting step is executed by utilizing algorithms that perform shape alignment through

point set matching. Shape model fitting of the skull and ventricular system are performed sequentially.

Generally, this can be formulated as a regression problem. In the standard regression problem a given point set is taken as input. The goal is then to search for a function that best fits these points. The regression problem for statistical shape modeling can be formulated accordingly. A reference shape that consists of $X = \{x_1, ..., x_n\}$ points is matched to a target shape with $P = \{p_1, ..., p_n\}$ points of the same shape family.

As the target shape may be of a different shape the possible deformation needs to be found. We do this to move the defined Kocher's and target points accordingly. Several algorithms have been implemented to perform the shape matching and will be presented in the following.

Different versions of the *Iterative Closest Point (ICP)* and the *Markov Chain Monte Carlo (MCMC)* method have been implemented and evaluated.

- Rigid Iterative Closest Point
- Flexible Iterative Closest Point
- Combined Iterative Closest Point
- Markov Chain Monte Carlo

In the following the current patient anatomy or patient scan is referred to as the target mesh. The statistical shape model is referred to as the source mesh.

Rigid Iterative Closest Point: A certain amount of points is sampled on the target mesh to reduce the computation time. Then the closest point to the source mesh is determined and iteratively the distances of all points to the target mesh are calculated. The rigid version of the *ICP* method is not deforming as it only changes the rotation and translation. Throughout the iterations of the algorithm, the target and source mesh converge to the best match. The chosen number of iterations was twenty as no improvement was detectable after this amount.

Flexible Iterative Closest Point: The flexible version of the *ICP* method is very similar to the rigid version except it additionally deforms and tries to match the target mesh. This is also called the calculation of the posterior model. The number of iterations were set to 10. No improvement was detected with higher iteration numbers.

Combined Iterative Closest Point: Here both *ICP* methods were combined. First a rigid *ICP* is executed with 10 iterations and afterwards the flexible *ICP* with 6 iterations. Even though better matching results are possible with a higher iteration count, this would need a significant higher amount of time and no large improvements are detectable.

Markov Chain Monte Carlo: In the beginning also a limited amount of points on the target mesh are chosen and the closest points to the shape model are derived to perform an initial mapping of both meshes. A random walk is performed that guesses the posterior model. Then the acceptance / rejection ratio is evaluated and either the new or old proposal is used. Every proposal is added to the Markov chain. The best fit is taken as the posterior shape after a defined number of iterations.

The algorithms yielded different results regarding the fitting accuracy and the runtime. The results are described in the following Section 4.2.3.

4.2.2.3. Implementation

All implementations regarding the statistical shape models and the fitting algorithms were done using the Scalismo library or the standalone tool Scalismo Lab version 0.90. The annotation of the optimal points were done in MITK. The surface models were generated using a python script utilizing the ITK framework [105]. The marching cubes algorithm was used to construct the surface of the ventricular system and the skulls.

4.2.3. Evaluation

In this subsection the evaluation of the automated path planning is presented.

4.2.3.1. Experimental Validation of the Comparison of the Fitting Algorithms

To evaluate the four different methods, exemplary fittings are performed. For this purpose, five representative datasets were selected. The fitting methods evaluation was performed on the ventricular systems of the patient models, as they represent the more challenging structure. On each of these five patient models fitting of the statistical shape model was performed using all four methods. The algorithm was executed 5 subsequent times in a row and the mean error and runtime was measured. It must be stated that the algorithms always result in the same posterior model, as the input was always the same and the same parametrization was used. The runtime differed slightly.

All experiments were performed on one workstation with Intel(R) Core(TM) i5-8500, 8GB DDR4, Ubuntu 18.04 LTS and calculated on the CPU.

4.2.3.2. Results of the Fitting Algorithm Evaluation

The results of the evaluation of the four presented methods are depicted in Table 4.3.

Table 4.3.: Evaluation of the fitting methods.

Method	Accuracy (mm)	Runtime (s)	Iterations
Rigid ICP	5.1	9.0±1.1	100
Flexible ICP	6.1	140±2.4	100
Combined ICP	4.76	14.8±0.2	10 (rigid) /6 (flexible)
MCMC	5.6	24±0.8 / 85±1.8	100/5000

The fitting with the combined *ICP* resulted in the best fitting accuracy. The runtime was also sufficient with 15 seconds. In the further experiments the combined *ICP* method was taken for the shape model fitting.

4.2.3.3. Experimental Validation of the Shape Fitting

The proposed method is evaluated on 33 randomly selected real patient CT datasets. Planning was conducted for the left and right hemispheres resulting in a total of 66 plannings.

The datasets contained patients with and without pathologies. The plannings of the puncture path were carried out by determing the Kocher's (K) and target (T) points inside the ventricular system as depicted in Figure 2.4. The results of the automated planning were compared to manually performed plannings. The manual annotation was done by two expert neurosurgeons (1 neurosurgeon, 1 neurosurgical resident). For the determination of the Kocher's point an area of 30×25 mm around it was rated as an acceptable placement as shown in Section 2.2.1. Points inside the ventricular system were rated as correct when they were placed in a way that it would have come to a drainage of cerebrospinal fluid and were located in the first third. The distance to the annotated optimal points were determined for every automatically derived point resulting in an accuracy value. All derived points are rated by an expert neurosurgeon if they can be categorized as valid or not.

A path is evaluated as properly planned if both points are correctly derived.

4.2.3.4. Results of the Shape Fitting

The results of the automated path planning are visualized in Figure 4.12.

The Kocher's points are derived through the statistical shape model fitting correctly in 98.4% of cases. The target points inside the ventricular system can be determined correctly in 95.4%. This leads to a rate of 93.9% properly planned paths. The mean accuracy of the determination of the Kocher's points is 8.4±4.5 mm ($n = 66$). The absolute error and the resulting errors in x-, y-, and z-directions are visualized in Figure 4.12 a) to d). An error value of 4.9±2.3 mm is found for the derivation of the target points inside the ventricular system. The error values of all axes are depicted in Figure 4.12 e) to h). The runtime of the proposed method is 57.0±3.4 s. A detailed listing of the runtime of the different steps is given in Table 4.4.

Process step	Time (s)
Nrrd to stl conversion	19±1.2 s
Kocher's point determination	18±0.7 s
Target point determination inside the VS	15±1.1 s
Trajectory calculation and model generation	5±0.4 s
All	57.0±3.4 s

Table 4.4.: Time measurements of all process steps.

4.2.4. Discussion

The mean models of the statistical shape models are generated from 26 real patient scans. The best way would be to also calculate the used kernel from the sample data. In this work the number of sample datasets was not sufficient enough. Therefore, a standard gaussian kernel was used that was parameterized accordingly. In the future more sample data should be used to generate the statistical shape model from these patient models. This could lead to better fitting results and a better representation of the possible shape variations.

The target patient scans were partially not aligned correctly to the midline and different parts of the skull were present. Three groups were identified. Scans were cutted under the

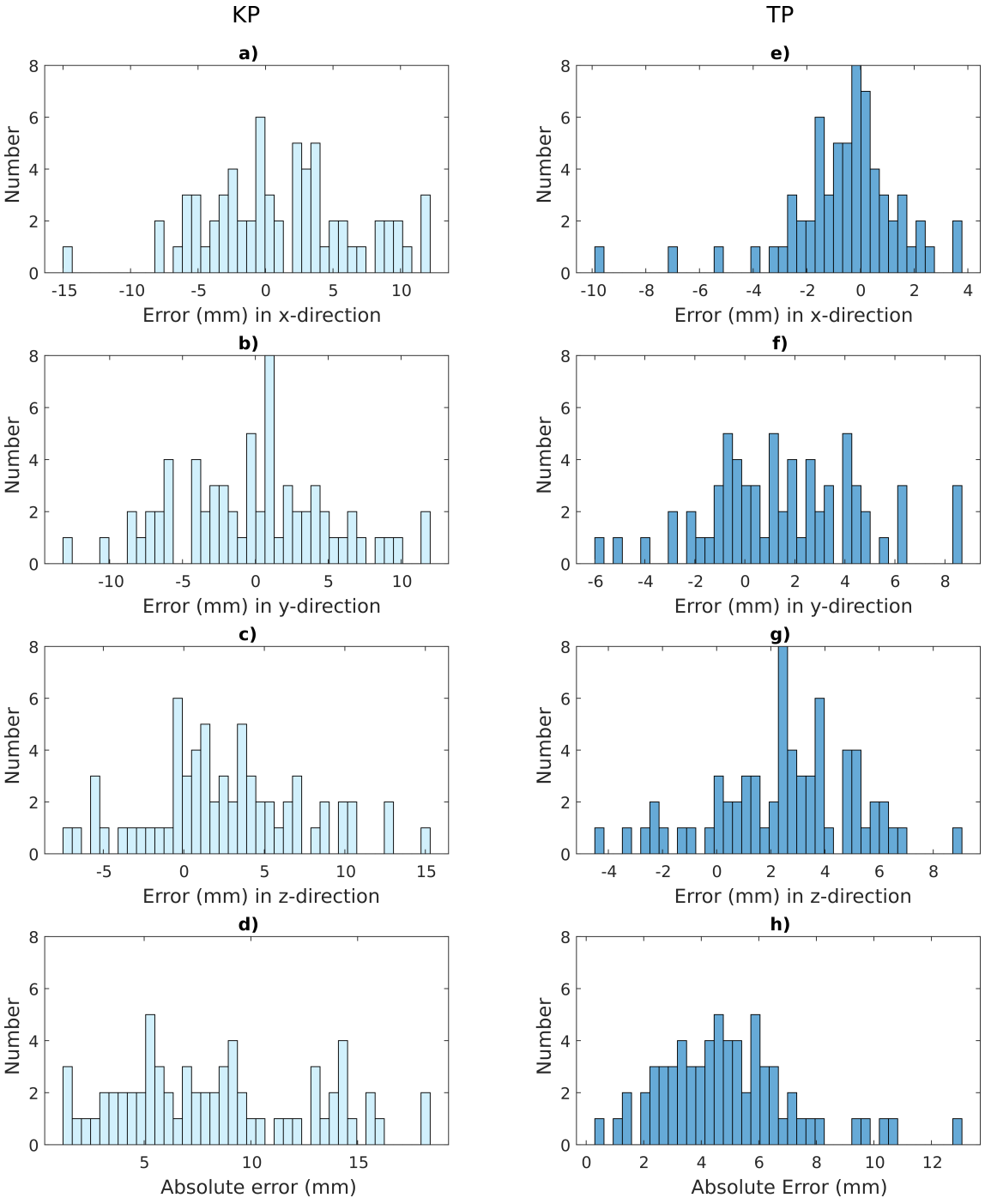


Figure 4.12.: (a) Error distance of automated derived Kocher's points on the skull to the manually annotated points: in x-direction, (b) y-direction, (c) z-direction and (d) absolute error. (e) Error distance of automated determined target points inside the VS to the annotated ground truth: in x-direction, (f) y-direction, (g) z-direction and (h) absolute error. All values given in mm. Adapted from [119].

nose, over the nose or the complete patient head was present. In this work, the scans were cutted manually over the eyes and rotated correctly. This could be accounted for during the scan procedure, so that always the same area of the head is scanned. However, in clinical reality the scanned area can differ a lot. Moreover, the patient head is sometimes only scanned partially to reduce the amount of radiation.

To address the different groups of scans also three different statistical shape models could be used. A patient scan could be fitted with all three different statistical shape models and afterwards an error metric could be determined, e.g. the mean squares error (MSE). The results of the fitting with the lowest error could then be chosen.

The statistical shape model of the ventricular system was modeled as a whole. This can lead to limitations when the ventricular system is not symmetric due to pathologies, e.g. when one ventricle is much larger than the other one. An expansion could be to model the left and right ventricles separately in two statistical shape models. This could improve the fitting reliability, especially for certain pathologies. A prerequisite for this would be that the left and right lateral ventricles are separated correctly in the segmentation step of this work presented in Section 4.1. Due to the limited resolution of the CT scan often the two lateral ventricles are segmented as a connected cluster.

Another promising approach would be the use of machine learning methods. Neural nets have been presented to automatically determine landmarks in CT or MRI data. The searched points could be directly determined with a neural net. However it is not clear if the Kocher's points and the target points inside the ventricular system could be determined with sufficient accuracy. Machine learning methods are good to find landmarks in the data, like the nose, ears and the eyes of a patient scan. This could be used to automate the preprocessing step that was manually done in this work. The scans could be oriented and cutted automatically, which is the largest source of error when using statistical shape models. The Kocher's points could also be constructed via the landmarks, as they can be defined as 11 cm above the nasion and 2-3 cm to the left or right lateral side towards the ears. The target point in the ventricular system could be constructed from the foramen of monro to the Kocher's points as this is a very unique landmark in the CT data.

Machine learning methods have been used in various works for the detection of landmarks in medical imaging data [230, 71, 49, 232, 17, 75, 163].

4.3. Expanded Path Planning based on Risk Structures

Surgical interventions can take up to several hours resulting in a high workload for the performing neurosurgeon, while they are required to work with highest accuracy and dexterity. Patient data is normally only visualized in 2D, which makes it necessary that the surgeon has a good spatial imagination in 3D. Navigated interventions are the standard of care in neurosurgery and technical aids are commonly used, like microscopes, endoscopes and nerve monitoring. Path planning is done on the basis of learned medical knowledge and experience of a surgeon, but often does not take the patient individual anatomy and brain structures into account. Basically, the complete brain is a risk structure. For certain interventions like the puncturing of the ventricular system or biopsies it is inviteable to move the instrument through brain tissue to reach the target. The goal then is to reduce the affected tissue to a minimum and avoid areas of very high risk. There are a number of different high risk areas inside the human brain. These are the blood vessels, nerve fiber tracts, eloquent areas, the ventricular system, the midline between both brain hemispheres and air-filled areas. These areas are gathered using different image modalities that were presented in Section 2.1.1 and 2.1.2. An expansion of the automated path planning is presented by additionally taking high risk areas of the brain into account.

The surgeon is provided with an assistive aid for planning of neurosurgical interventions. The risk structures are visualized to the surgeon in a volumetric view to enable fast, intuitive and reliable risk presentation to perform the planning. The software tool can be used to plan linear paths by selecting a good entry point on the skull to reach the target with minimized overall risk. The access route avoids high risk structures and minimizes the trauma to healthy tissue. The various risk structures were incorporated into a holistic model. All structures can be individually and freely weighted to each other. The neurosurgeons are provided with a comprehensive and interactive decision support tool to perform access planning on the skin, cranial or cortical surface.

4.3.1. Related Work

An automated safe-path planning tool for deep brain stimulation (DBS) was introduced by Brunenberg et al. [28]. For each voxel of risk structures they compute the Euclidean distance to the candidate trajectory and choose the trajectory that is most distant. The disadvantage of this approach is the long calculation time. Navkar et al. [158] describe a method that calculates the corresponding risk for trajectories from a puncture point on the head surface to a target point. They project the risk onto the skull surface but use only blood vessels as risk structures. The risk calculation takes into account the path length and the proximity to risk structures. The following work of Rincón-Nigro et al. [177] presents a planning system that enables almost real-time visualization of the risk map. This makes it possible to change the target of the trajectory and recalculate the risk in real time.

Shamir et al. [197] pair strict constraints such as a maximum path length with various cost functions. The risk of each voxel is computed according to the neighboring risk structures, and a sum of weights along the candidate trajectory is used to calculate the associated risk. A table provides the neurosurgeon with information about the risk parameters of the selected entry point and trajectory. Considered risk structures are blood vessels and the

surgical working volume, being defined as a cone-shaped sleeve. Different methods of risk calculation are also compared and described.

De León-Cuevas et al. [50] presented a user interface that visualizes the risk of several linear trajectories to a user. Two main steps are performed for risk allocation: a weighted sum that depends on the distance of a single voxel from other risk structures, and a second criterion taking into account other risk factors, such as the length of the trajectory with fuzzy rules.

Herghelegiu et al. [84] present a biopsy planner that visualizes trajectories between manually chosen entry points and the target in an augmented 2D cross-section view. A needle pathway distance graph was added to determine positions and cross-sections of highest risk. Bériault et al. [30] enhance the calculation times by including only those voxels that lie within the cylindrical trajectory. The major disadvantage of this approach is that the neighborhood is not included and thus the risk calculation is not affected by risk structures in close proximity. However, all of the above studies include a risk projection to the outer surface of the skull, either in a flat map representation or for only a section of the skull. Furthermore, not one of the studies fully addresses the complete surgical planning process.

Building on this research, Essert et al. [60] use a global trajectory optimization strategy that simultaneously combines cost functions of strict and soft constraints with user-dependent weights of these cost functions. Each case of the trajectories computation takes two minutes. As risk structures blood vessels and the ventricular system are used.

In the IEEE Visualization Contest 2010 [39], which addressed the topic of multimodal visualization for neurosurgical planning, the winners Diepenbrock et al. [55] present a system for planning brain tumor resections. They examine the interaction of blood vessels, functional areas and nerve fiber tracts, which are used as risk modalities, with the tumor. Particular importance is placed on the visualization of this interaction. The introduced system allows the surgeon to define the access route interactively by using the 3D views and to measure distances for evaluation of the possible access routes and assessment of associated risk. The radius of the circular access path can be selected by the user. A probe view can be used to inspect the trajectory in the first-person perspective.

Vaillancourt et al. [213] also present work on visualization and interaction of risk structures and consider fiber tracts according to their location in the brain and in relation to the tumor. In particular, this work focuses on fiber tracts as risk structures and only marginally accounts for blood vessels and white and gray matter brain tissue. Apart from the use of appealing visualizations, however, these studies do not provide a projection of risk to the skull surface.

Other related work addresses trajectory planning. Zhang et al. [231] incorporate eloquent brain areas into their visualization using diffusion tensor imaging (DTI) and functional magnetic resonance imaging (fMRI) data. While they visualize these data, they do not calculate risk measures. Related to this work is that of Vaillancourt et al. [213] and it extends the state of the art by the use of DTI fiber tracts based on multiple volumes of interest. De Momi et al. [51] present an automatic trajectory planner that maximizes the distance to blood vessels while avoiding sulci. Optimal trajectories are proposed based on manually defined entry and target areas. No individual weighting of different risk structures is considered, ignoring the case where a voxel is part of multiple risk structures.

Segato et al. [191] presents an approach for robot-driven curvilinear trajectories to plan deep brain stimulation treatments. To calculate the optimal path from an entry area on non-linear paths, the eloquent areas, nerve fiber tracts, the ventricular system and the tumor are considered, incorporating a safety distance to these structures.

Villanueva-Naquid et al. [219] introduce methods for a faster calculation of an optimal trajectory. A genetic algorithm is used to solve the safest trajectory search problem by avoiding an exhaustive search. Using this solution is limited to the use of blood vessels and the ventricular system.

A detailed comparison of the aforementioned works is presented in Table 4.5 and compared to the proposed method in this work.

4.3.1.1. Commercial Systems

There are several commercial navigation systems in the field of image-guided neurosurgery. The systems support by tracking the position of the surgeon's tools and displaying their position on pre- or intraoperative image volumes. The major competitors are *Stryker* (Stryker Corporation, Kalamazoo, Michigan, USA), *Brainlab* (Brainlab AG, Munich, Germany) and *Medtronic* (Medtronic plc, Dublin, Ireland), accounting for up to three quarters of the market share [6]. All of them use various (semi-) automatic registration and segmentation algorithms for interaction and highlighting of risk structures.

Key developments in this area involve the integration of robotic applications to ensure safe intervention based on a pre-planned trajectory. *Zimmer Biomet* (Zimmer Biomet Holdings, Inc., Warsaw, Indiana, USA) introduced a system intended for robotic cranial interventions, the neurosurgical robot ROSA One®, combined with an optical tracking system. It supports cranial interventions that require surgical planning with preoperative data, patient registration and precise frameless stereotactic instrument positioning and handling. The ROSA system provides haptic features that allow the neurosurgeon to manually guide the instruments within limits defined during planning. Another specialized navigation system is *B.Braun*'s (B.Braun AG, Melsungen, Germany) OrthoPilot® for orthopedic surgery, especially knee and hip arthroplasty. A special feature is navigation without the use of preoperative images, which enables fast surgery and no pre-registration, but offers less accuracy. The Mako system from *Stryker* is specifically designed for orthopedic knee and hip surgery. These planning and navigation systems have the capability to ensure a better surgical outcome by minimizing soft tissue and bone trauma, increasing accuracy, as well as minimizing the risk of serious error by limiting movement. The Swiss company *CAScination* (CAScination AG, Bern, Switzerland) focuses on targeted otologic procedures and laparoscopic percutaneous surgery planning.

4.3.2. Requirements Analysis

To implement a reliable, robust and usable system for usage in clinical practice it is important to understand the needs of the target group which consists mostly of surgeons. The requirements for a surgical planning tool are identified by conducting a qualitative user study with medical experts ($n = 5$; 1 neurosurgical resident, 4 neurosurgeons). A prototype of the system was implemented and evaluated by these medical experts. Subsequently, they were asked to answer a questionnaire.

Table 4.5.: Related work comparison: Surgical workflow mapping (C-I), integration of multiple structures into a holistic model (C-II), weighting of each structure's risk (C-III), projection of results onto the surface (C-IV), provision of a safety margin around tool or risk structure (C-V), applied soft (SC) and hard (HC) constraints (C-VI), and tool access point planning (C-VII) [118].

Method	C-I	C-II	C-III	C-IV	C-V	C-VI	C-VII
[28]	yes	3 (ventricles (V), blood vessels (BV) gyri (G))	no	no	yes	SC: minimize distance to V and BV HC: entry points on frontal lobe gyri	cortical surface projection
[158]	no	1 (BV)	no	yes	yes	SC: path length (PL), minimize distance to BV (on surface projection)	skin surface projection
[213]	no	3 (fiber tracts (FT), BV, functional areas (FA))	no	no	yes	none	none
[55]	yes	4 (FT, FA, BV, bone (B))	no	no	yes	none	exemplary trajectory only
[84]	no	1 (BV)	no	partly	yes	SC: minimize distance to BV	manual picking of region for possible entry points
[197]	no	2 (BV, V)	yes	partly	yes	SC: minimize distance to BV and V	skin surface projection
[60]	no	2	yes (sulci (S) and V)	yes	yes	SC: minimize PL and distance to S and V, optimize electrode placement, HC: maximum PL, risk avoidance	upper skin surface projection, gyri projection on the skin surface
[177]	no	1 (BV)	yes	yes	yes	SC: Minimize PL	skin surface projection
[231]	no	3 (BV, FT, FA)	no	no	no	none	none
[50]	no	2 (V, BV)	yes	partly	yes	HC: no crossing of brain midline, PL <=90 mm, no direct hit of V and BV	skin surface projection
[191]	no	3 (V, cerebral cortex BV)	yes	no	yes	SC: minimize hits of risk structures, minimize maximum curvature, minimize PL, smooth path, orientation of electrodes shape	none
[30]	yes	3 (V, S, BV)	yes	partly	partly	SC: avoid subcortical BV, minimize overlap with caudate nucleus, minimal overlap of cortical gray matter HC: entry point in frontal lobe, avoid midline, avoid penetration of risk structures	cortical surface projection
[51]	partly	4 (V, thalamus, cerebelum, BV)	partly	no	no	SC: minimize distance to BV and insertion angle HC: avoid penetration of risk structures, maximum distance between electrodes, maximum curvature	depending on image resolution
[219]	partly	2 (BV, V)	yes	no	no	none	trajectory count depending on genetic algorithm used
Presented Approach [118]	yes	4 (BV, V, FA, FT), not limited	yes	yes	yes	SC: Minimize path length and trajectory risk HC: maximum PL avoid penetration of risk structures	cortical or skin surface

The neurosurgeons were asked to answer the following questions freely:

1. **Do you use software for planning and during surgery? If yes, which one?**
2. **What image modalities do you use in general and which are available for planning?**
3. **What are high risk structures and how important would you rate them?**
4. **How much time do you generally have for planning a cranial approach?**
5. **What features should a tool offer to support the surgical planning process?**

The risk structures were rated as very important, important, neutrally important, unimportant or very unimportant. The answers were as follows:

Do you use software for planning and during surgery? If yes, which one?
All of the questioned medical experts use navigation software to plan and perform surgical interventions. They stated that planning tools are an essential part of their surgical workflow. Usage of these assistance tools is the standard of care. All questioned surgeons use the BrainSuite (Brainlab GmbH, Munich, Germany). Note: all questioned neurosurgeons are at the same clinic.

What image modalities do you use in general and which are available for planning?
Most frequently, an isotropic thin-slice MRI with T1, T2 sequence and a contrast-enhanced image is obtained to detect and visualize blood vessels. Dependent on the situation, other image modalities are used to create additional datasets. This depends primarily on whether the hospital deals primarily with emergency situations or performs more planned procedures. For tumor treatment, fMRI and DTI scans are often performed, especially if the tumor is in a critical location, e.g. near an eloquent area. Regular CTs are prepared as a standard in emergency situations, because they can be obtained very fast, and several pathologies can be seen like dilated ventricles. MRI T2 FLAIR sequences are used to separate tissue-bound from free liquids, e.g. to distinguish between cerebrospinal fluid and lesions in a patient scan.

What are high risk structures and how important would you rate them?
In general, all structures inside the human head are risk structures, but some are more vital than others. The brain can reorganize when parts are damaged. This is not possible or very difficult for certain areas. Therefore, these vital structures need to be avoided during surgery.

Eloquent areas were rated as most vital, with the motor cortex as the most important structure. In second place is the language center, followed by the blood vessels. Avoidance of nerve fiber tracts is considered necessary, although the determination of their location is challenging. Sometimes contact cannot be avoided depending on the selected target point in the brain. The surgical puncture path length should be as short as possible and was rated as important. Damage to healthy tissue should be minimized. Cosmetic aspects were rated as least important from a neurological-functional point of view and should be considered whenever possible. Incisions in the face should be avoided and existing scars should be reused whenever possible.

How much time do you generally have for planning a cranial approach?
Each cranial procedure is planned based on individual patient imaging data. Typically this includes the entry and target point that form the path.

Generally it can be stated that the time for planning depends strongly on the type of procedure. Also the individual experience of a surgeon plays an important role. It is very likely that an experienced neurosurgeon needs less time for planning. Standard procedures, like biopsies (e.g. tumor site), take an average planning time of three to ten minutes depending on the target location. Deep-seated targets need more time for planning due to higher complexity. For more challenging procedures, such as resection of a tumor, the planning time is reported to be ten to thirty minutes.

Depending on the current surgical situation a re-planning of the path can become necessary.

What features should a tool offer to support the surgical planning process?
The medical experts expressed their need for a shorter planning time. It is also desirable to minimize the risk of a path and to maximize the distance to risk structures leading to a safe planning procedure. It was also asked which degree of automation is considered useful and acceptable for a surgeon. Automation should be limited to a certain level and only recommendations should be provided. It is then in the decision of a surgeon to accept a planning or adjust the proposal. The planning tool should map the surgical workflow and should be easily adaptable to changing situations. The system must depict detailed information about the patient and the used image data to interpret the accuracy of the proposed surgical planning correctly. This can be the slice thickness and used resolution.

The survey with medical experts lead to several requirements that are listed in Table 4.6 and 4.7.

Two different requirement groups have been identified. On the one hand, there is a strong need for a good usability and design of the system. Requirements of this category are referred to as design requirements. On the other hand, the system needs to provide functionalities that are gathered as functional requirements.

Table 4.6.: Design requirements of the system for risk-based path planning gathered in the user study with medical experts. RD = design requirement

ID	Requirement
RD1	User-centered design
RD2	Integration of surgical planning workflow
RD3	Interactive coronal, axial and sagittal planes
RD4	Interactive 3D view
RD5	Pick target point in 2D view
RD6	Pick entry point in 3D view
RD7	Show cross-hair of current position
RD8	Display information about dataset dimensions
RD9	Trajectory visualisation
RD10	Augmented views
RD10.1	Display segmented risk structures in 2D views
RD10.2	Display segmented risk structures in 3D view
RD11	Planning on skin and cortical surface

4.3.3. Dataset and Segmentation of Risk Structures

In this work, a multimodal dataset was needed to obtain as many risk structures as possible to test the system capabilities. Thus, an open-source dataset from the visualization contest 2010 [9] was used as a starting point and partially modified and expanded, because no available dataset matched all the requirements.

The blood vessels were segmented out of an added digital subtraction angiography (DSA) scan that was not part of the original IEEE dataset. The image data was obtained from a different patient and matched to the IEEE dataset by adjusting the scan accordingly. In this way a virtual patient was created. A modified Frangi Vesselness Filter [67] was used to perform the segmentation. The ventricular system was switched and smaller tumors

Table 4.7.: Functional requirements of the system for risk-based path planning gathered in the user study with medical experts. RF = functional requirement

ID	Requirement
RF1	Dynamic risk weight change
RF2	Modular architecture for including arbitrary structures
RF3	Integration of path length
RF4	Integration of strict & soft constraints
RF5	Save trajectory information
RF5.1	Human-readable data interface
RF5.2	Machine-readable data interface
RF6	Integration into the Robot Operating System (ROS)
RF7	Fast update time

were defined as masks. This was done to create several different scenarios. The original tumor mask was used as well. Eloquent areas were extracted from the fMRI sequence with a threshold approach. The segmentation of the nerve fiber tracts was provided in the IEEE dataset. The skull, skin and the ventricular system were extracted from the patient scan using the algorithm that was introduced in Section 4.1 and [113]. The segmentations are stored as *nifti* or *nrrd* files [201] [54]. Our system is capable to work with both. The volumetric file formats are used for faster and easier processing.

It is necessary that all image modalities are scaled, rotated and translated correctly to each other, so a proper registration needs to be performed. Each voxel of every image modality then represents the same physical location. This can make it necessary that data is partially sampled from voxels nearby, e.g. if the resolution is smaller than the target modality. Different algorithms are used that are dependent on the modality, e.g. the head-and-hat and iterative closest point algorithm.

The segmented risk structures are shown in Figure 4.13 and together form the risk maps M_i.

4.3.4. Methods for Risk-based Trajectory Determination - Risk Calculation

The proposed system is designed to help neurosurgeons find an optimal path to a target in the human brain with minimized risk. The obtained image data of a patient is presented to the surgeon in three 2D views as well as one 3D view. The data is augmented with segmented risk structures derived from that data. To select a target in the desired structure, e.g. the ventricular system or tumorous tissue, the surgeon can use the interactive tool. Vertices of surface model meshes (*.stl*) of the skin, skull (cranial) or brain (cortical) are used as possible entry points. The system then calculates all risks of all possible paths to a selected target and stores it as a skin projection map. The number of calculated paths equals the number of vertices. A high level view of the system workflow is presented in Figure 4.14.

The risk of these paths is calculated based on an arbitrary number of anatomical structures inside the brain. These may include the ventricular system, nerve fiber tracts, eloquent areas, and blood vessels. All represent vital structures of the brain. Multimodal medical 3D

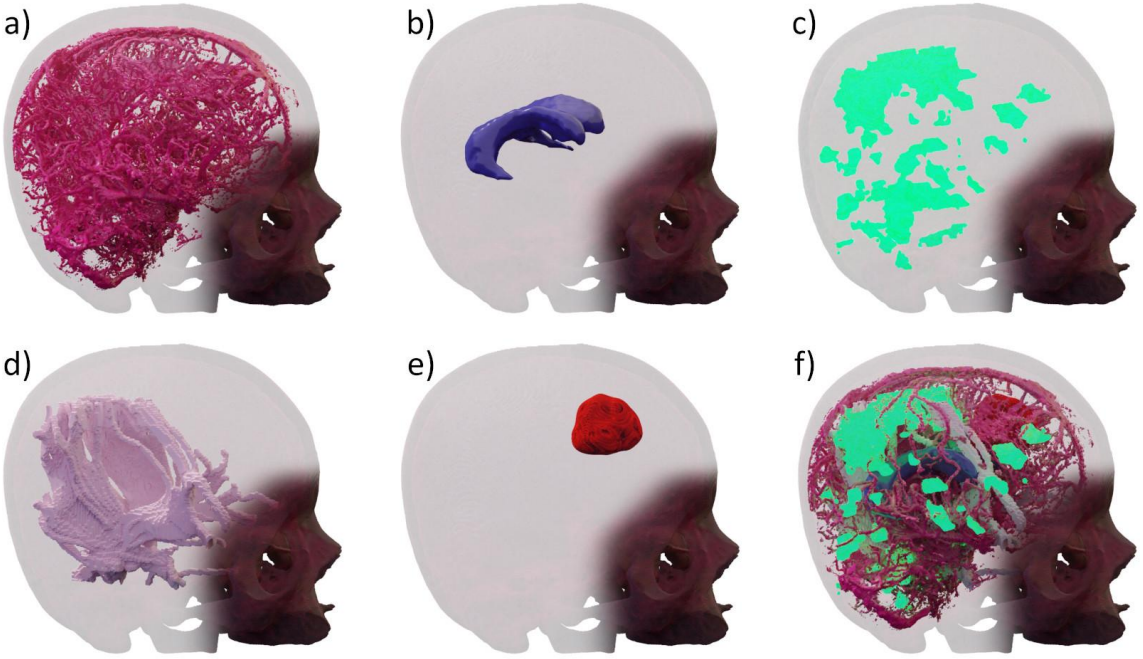


Figure 4.13.: Visualization of the different risk structures segmented out of the dataset. a) Blood vessels, b) Ventricular system, c) Eloquent areas, d) Nerve fiber tracts, e) Tumor mask, f) All risk structures. Adapted from [118].

imaging techniques are used on the patient to acquire these data (e.g. CT, MRI, fMRI, and DTI). Three-dimensional risk maps are used to store each high-risk area of the brain individually. A neurosurgeon must avoid each of these structures if possible to reach the chosen target. Through the modular architecture, it is possible to add additional risk structures at any time to the risk maps such as sulci, gyri, hemorrhages and special eloquent areas. This mainly depends on the pathology of a patient and availability of image data.

Each risk structure i is read from its segmentation file and stored in memory as a third-order tensor (3D matrix) M_i to enable fast processing. For evaluation of the system the aforementioned dataset is used.

Trajectory planning is performed by selecting a suitable path T_j with the lowest overall risk, while j is the number of vertices of the surface model mesh.

A medical expert can define a weighting factor w_i for every risk map M_i as a reflectance of its importance as an anatomical structure. Each risk structure can be considered in the overall risk calculation or switched off. This can be of special interest to integrate less important structures like cosmetic aspects. It can also become necessary when no path can be found with zero risk, which means that not all risk structures can be avoided. In that case the goal is to minimize the risk and the damaged tissue. The marching cubes algorithm is used to generate surface models of the skin, skull or brain surface.

The total risk of a path is then projected onto one of these surfaces. As a standard, the surface of the skull is used. Exemplary surface projections are illustrated in Figure 4.23. The colors represent the risk values from blue (low risk) to red (high risk).

A target point inside the brain is then manually chosen by the neurosurgeon, e.g. the center of a tumor. Each surface model has a certain resolution in form of the vertices count. From each vertex to the target a path is generated and the risk is calculated. To

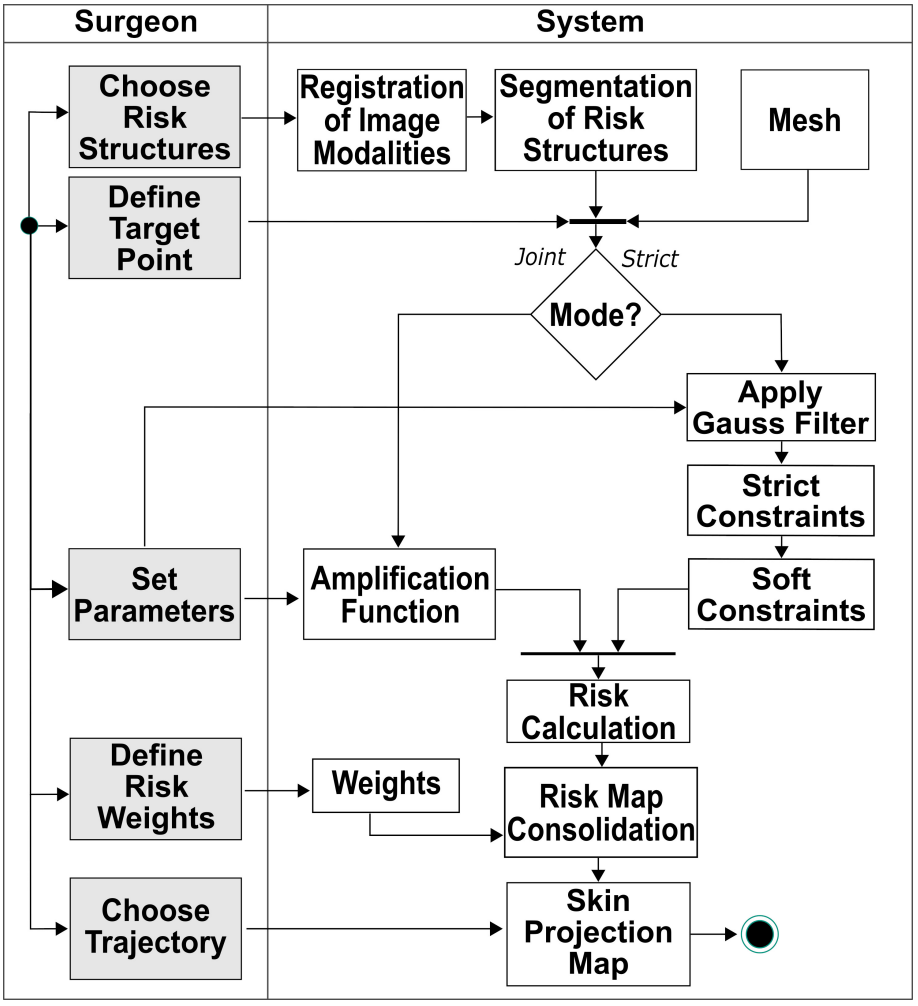


Figure 4.14.: Activity diagram of the proposed system, visualizing the workflow and the user interactions [118].

calculate the total risk of a trajectory all voxels must be determined that lie on the path by using the Bresenham algorithm [27] due to its fast computational performance. The set of voxels B_{T_j} of a path is identified.

In a next step, the medical expert can choose a weight for each of the selected risk structures. As a rule, all risk structures are weighted equally. This lies within the choice of the medical expert. So it is possible to assign lower values to risk structures that were rated less important. Additionally a weight can be given for the path length. This weight needs to be carefully chosen, because it is desirable to minimize the path. This can lead to the problem that low risk paths are set closer to risk structures.

The system offers two modes for risk calculation: *joint* and *strict* mode. Depending on the selected mode, the risk is calculated differently.

The *joint* mode represents the basic mode of the system. The trajectory risks are projected onto the chosen surface and amplified to create a safe area around the risk structures.

The risk calculation is done by adding up all risk values of all voxels at position (k, l, m) that are part of a trajectory T_j. Risk values are weighted by the chosen factor. The total trajectory risk $r(M_i, T_j)$ is calculated separately for each risk structure i and its risk map M_i.

$$r(M_i, T_j) = \sum_{(k,l,m) \in B_{T_j}} m^i_{klm} \tag{4.21}$$

The total consolidated risk $\hat{R}(T_j)$ of a trajectory T_j, including all chosen risk structures, is computed as a weighted sum as in

$$\hat{R}(T_j) = \sum_i r(M_i, T_j) \cdot w_i \tag{4.22}$$

The final risk of a path is then normalized in relation to the maximum risk of all trajectories:

$$R(T_j) = \frac{\hat{R}(T_j)}{\max_k \hat{R}(T_k)}. \tag{4.23}$$

The total risks of all paths are projected onto the cranial, cortical, or skin surface and stored as the vertex color value of the surface mesh. Values between 0 and 1 are possible and mapped to a color scale from blue to red. It is possible that no calculated path has a risk of 0, but the path with maximum risk has always the risk value of 1. To create a safe area around high risk trajectories, an amplification function can be applied to all paths changing the values of the projection map. The amplification function as shown in Eq. 4.24 is used to intensify these values, where C is the amplification parameter, set by a user.

$$\mathscr{R}_{joint}(T_j) = 1 - e^{-C \cdot R(T_j)} \tag{4.24}$$

The *strict* mode is a further development of the *joint* mode. The projection map gives a surgeon a very intuitive interface to plan an intervention. Creating a safety area on the 2D projection is fast and robust. However, this neglects the structures in its volumetric form. Therefore, as an alternative approach the safety area creation can be performed on the volumetric data by applying a Gaussian filter on the risk maps M_i. The Gaussian filter was modified because it is not desirable to reduce the risk value of a risk structure. The goal is to create a safe perimeter by extending the risk structure to a certain amount by maintaining the original structures.

By applying the modified Gaussian filter to every risk map M_i, new dilated risk maps G_i are created.

We define the Gaussian filter as in

$$K_g(k,l,m) = \frac{1}{(2\pi)^{\frac{3}{2}}\sigma^3} \exp(\frac{k^2+l^2+m^2}{2\sigma^2}) \tag{4.25}$$

with k, l and m being the dimensions of the kernel of the filter and σ being the strength factor of the Gaussian filter. The intensity of a voxel in the risk maps M_i is derived by $I_{M_i}(p)$ at position $p(x|y|z) \in [0, width] \times [0, height] \times [0, depth] \subset \mathbb{R}^3$.

The new intensity of voxel p after applied Gauss filter can be defined as

$$I_{G_i}(p) = I_{M_i}(p) * K_g(k,l,m) \tag{4.26}$$

When applying the Gaussian filter to the risk maps M_i the dilated risk maps G_i are created. The 3D matrix G_i has the entries $g_i = \max\{I_{G_i}(x,y,z), I_{M_i}(x,y,z)\}$, where the discrete position of a voxel is x, y, z. A dilated risk map of blood vessels is shown in Figure 4.15c. All risk structures that have been segmented in the patient data were marked as 1 in all risk maps M_i. The dilated risk around the structures values in G_i range from < 1 to 0 in a specific radius in relation to the chosen value of σ. The original mask is untouched. A surgeon chooses the value of σ for every risk structure individually. The larger a risk structure is the larger the value of σ is chosen. A good parametrization for the motor cortex is $\sigma = 6$ and $\sigma = 2$ for blood vessels. The values were determined empirically.

A risk value represents a certain hit probability. The safe area was created to react to a number of uncertainties like brain shift and tool deformations when inserted into the human brain. The size of the safe margin should be created in a way that it compensates for these uncertainties. Several works have investigated the expected brain shift and deformations caused by the inserted tools [83, 74, 161, 199, 151, 175, 160, 192].

The safe margin can be adjusted accordingly.

In *strict* mode *hard* and *soft* constraints are implemented. Risks that must be avoided are classified as hard constraints (HC). They are applied first on every path. If a path violates a hard constraint it gets blocked and marked with a relative risk of 1, else the risk calculation is done based on the risk values of all risk maps, i.e. $r_{hc}(\cdot,\cdot) \in \{0,1\}$). All risks that must be minimized are grouped as soft constraints (SC).

The first hard constraint (HC1) forbids the direct hit of a risk structure. This is the case if a voxel at position g_i has the value of 1 in any risk map. The second hard constraint defines a maximum length of a path that must not be undercut (HC2).

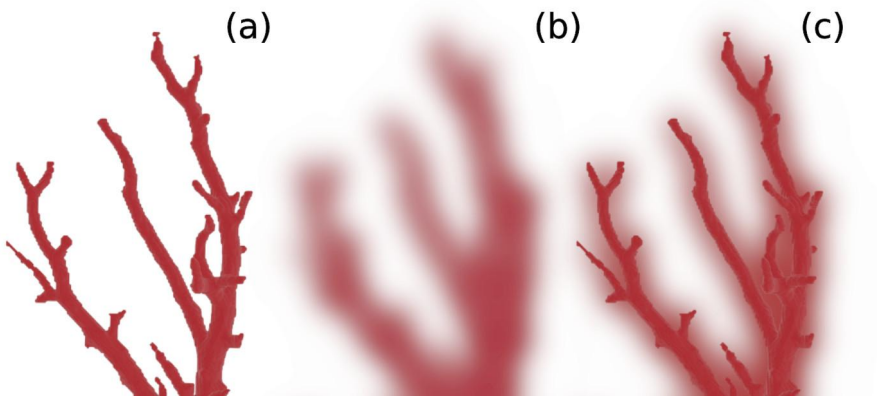


Figure 4.15.: (a) Visualization of the blood vessel risk map M_i, (b) the same risk map M_i after applying the Gaussian filter and (c) the dilated risk map G_i after applying the maximum function [118].

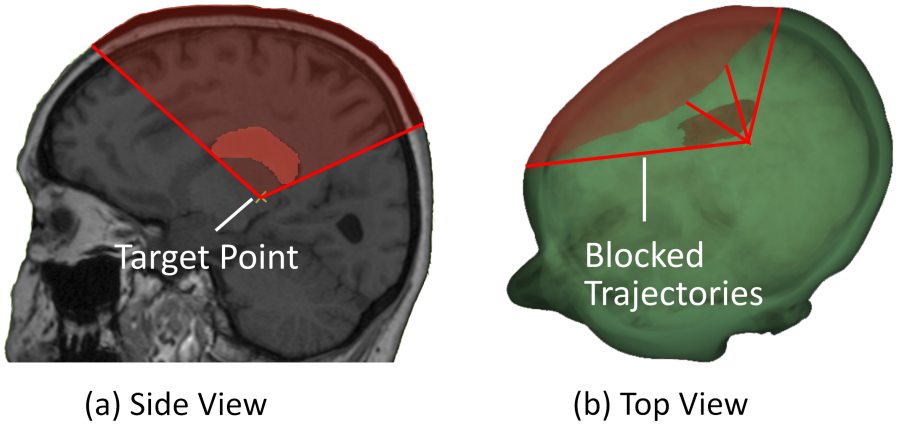


Figure 4.16.: Visualization of blocked trajectories in strict mode [118].

Paths that are blocked due to the violation of HC1 are depicted in Figure 4.16.

A parameter that must be minimized is the length of a path (SC1), because the goal is to damage as little healthy brain tissue as possible. To minimize the risk of penetrating a risk structure the distance to them must be maximized (SC2). Therefore, small path deviations and brain shift can be compensated.

A linear relationship is assumed between the length of a path T_j and the associated risk. SC1 is defined as a normalized risk function R_l, where $l_{min} = \min_k length(T_k)$ and $l_{max} = \max_k length(T_k)$ are the longest and shortest paths.

$$R_l(T_j) = \frac{length(T_j) - l_{min}}{l_{max} - l_{min}} \tag{4.27}$$

The risk of a path considering all risk maps (for the time being without considering the path length) is defined as a normalized risk function R_d as in

$$\hat{R}_d(T_j) = \sum_i r(G_i, T_j) \cdot w_i$$
$$R_d(T_j) = \frac{\hat{R}_d(T_j)}{\max_k \hat{R}_d(T_k)} \tag{4.28}$$

with the weights w_i predefined by a surgeon.

The risk of a trajectory is normalized with the maximum risk of all trajectories $\max_k \hat{R}_d(T_k)$. This leads to a normalization of the risk values to a range of 0 to 1.

The total risk of a trajectory in strict mode $\mathscr{R}_s$ for a given trajectory T_j is defined as

$$\mathscr{R}_s(T_j) = \begin{cases} 1 & |\{M_i | r_h(M_i, T_j) = 1\}| \geq 1 \\ \frac{1}{2}(R_l(T_j) + R_d(T_j)) & \textit{else} \end{cases} \tag{4.29}$$

The total risk values are also in *strict* mode projected onto the chosen surface, to provide an intuitive planning visualization to a surgeon. The risk calculation for *strict* and *joint* mode are exemplary visualized in Figure 4.17.

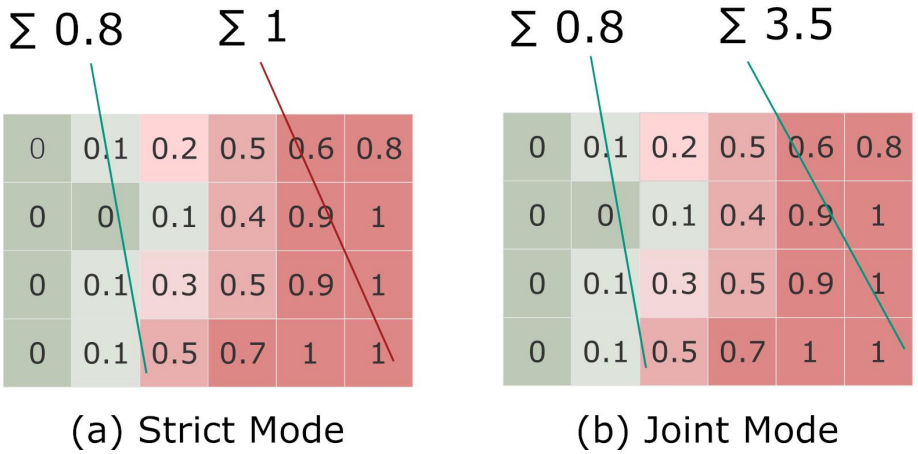


Figure 4.17.: Visualization of the different calculation methods for the total risk in (a) strict and (b) joint mode. Adapted from [118].

The calculation of the different risks per structure and trajectory is executed separately. For every risk map M_i or G_i the risk per path is determined and stored in its own data structure. This step was implemented to enable a faster recalculation when weights are changed intraoperatively. An alternative approach would have been to calculate the total risk of a trajectory at once. This renders the initial risk calculation of all trajectories faster. However, a recalculation with updated weights is much slower. In the proposed system architecture only the trajectory risk of structures with updated weights must be recalculated to determine the new total weight of all trajectories.

An expanded architecture diagram of the system is depicted in Figure 4.18.

A surgeon can then, in a final step inspect the risk projection map and select a suitable entry point to a chosen target. It is also possible to check the trajectory in a fly-along visualization.

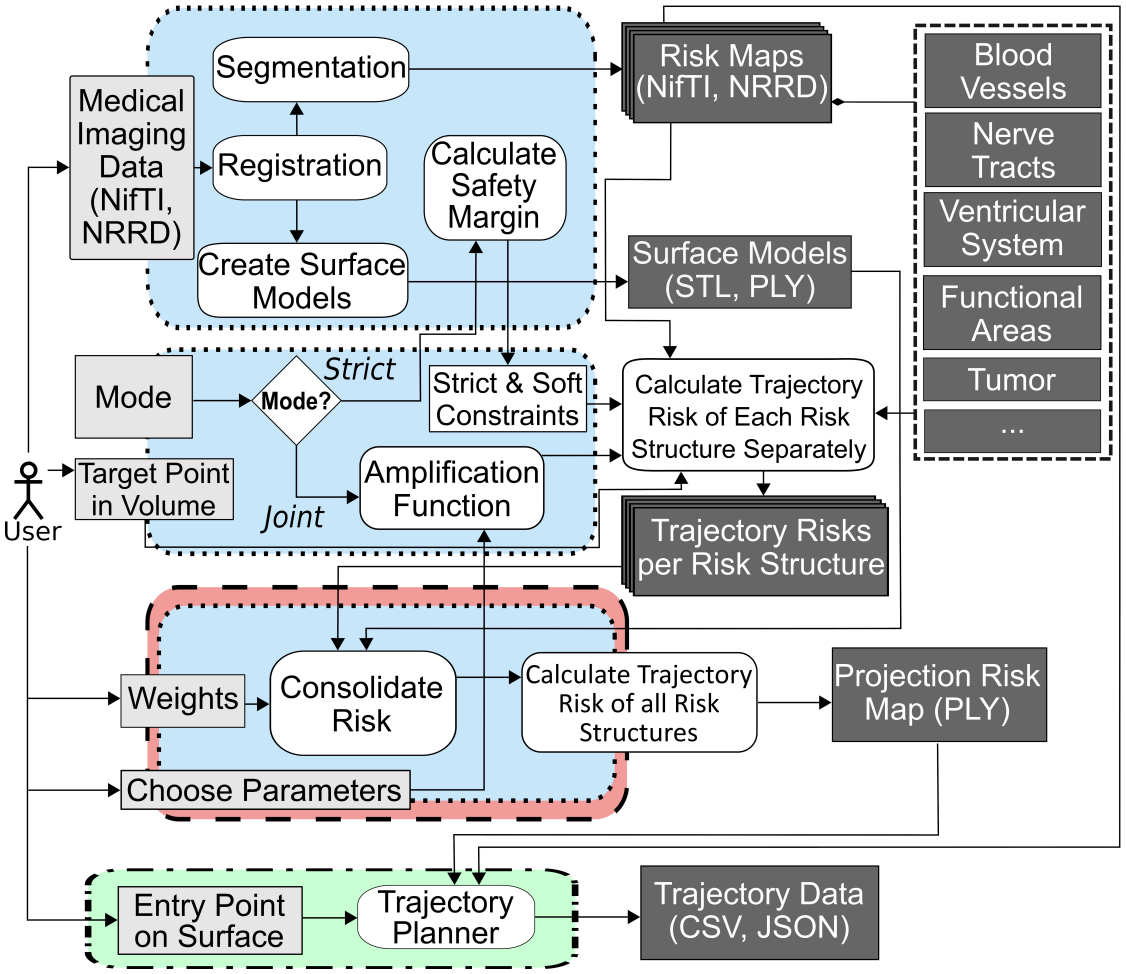


Figure 4.18.: Architecture of the planning tool divided in its main components. Actions are displayed in rectangles with rounded corners, while data and parameters are marked by a rectangular outline. Input data are marked light grey, stored output data are dark grey. Initial risk computation, update of risks, and saving a trajectory are indicated by blue, red and light green backgrounds, respectively. Adapted from [118].

4.3.4.1. Implementation

The presented planning tool was implemented as a standalone prototype utilizing Qt, VTK (Visualization Toolkit) [106] and ITK (Insight Segmentation and Registration Toolkit) [105]. To create the dilated risk maps G_i a Gaussian filter of ITK was used, the "Smoothing Recursive Gaussian Image Filter". For determination of the voxels that lie on a trajectory the Bresenham algorithm [27] was used.

4.3.5. Integration as Robot Operating System Packages

The implemented tool was used to initially test the risk-based path planning and to obtain first insights. It was decided to provide two independent modes to a surgeon to choose from. This was done to provide an easy and understandable way to interact with the system. However, the system can not be used in automated processes and the two defined

modes are not customizable. It is desirable that the different modules of each mode can be integrated freely and that they can be switched on and off to create custom modes to allow for a more flexible system. To cover these requirements, we integrate the solution in the Robot Operating System (ROS) as individual nodes or sevices.

The risk maps and consolidated risk maps have a size of round about 25 MB in the presented scenario which relates to the CT image size. This presents a certain challenge as it is not easy to process these amounts of data within a system in real time. ROS is a distributed system and it is also challenging to exchange the risk maps between multiple systems. The data storage was implemented using a shared memory approach, to enable a fast way for read and write operations.

The shared memory can only be accessed from the local machine where it was created on. If another device on the network needs to access the risk maps, they must be requested via an interface. However, it takes time to distribute and synchronize large amounts of data over network. Therefore all processing nodes should run on one device. The risk maps, the consolidated risk map, the paths, and the projection risk maps are stored in shared memory. The shared memory can be accessed at high speed to process large amounts of data only limited through the system capabilities.

The risk calculation is executed the same way as described in the previous chapter. The target point is provided to a ROS node that listens for incoming messages. The system then starts to calculate the trajectory risks of all possible paths from the surface to the target. Paths are derived from every vertex of the surface mesh to the target point. The system can then determine the trajectories with minimal weight. A surgeon can subsequently inspect the results in the visualization interface RVIZ. They can then inspect the risk maps and fly along each trajectory and choose the final path or directly the path with minimum risk.

An RVIZ example scene is depicted in Figure 4.19.

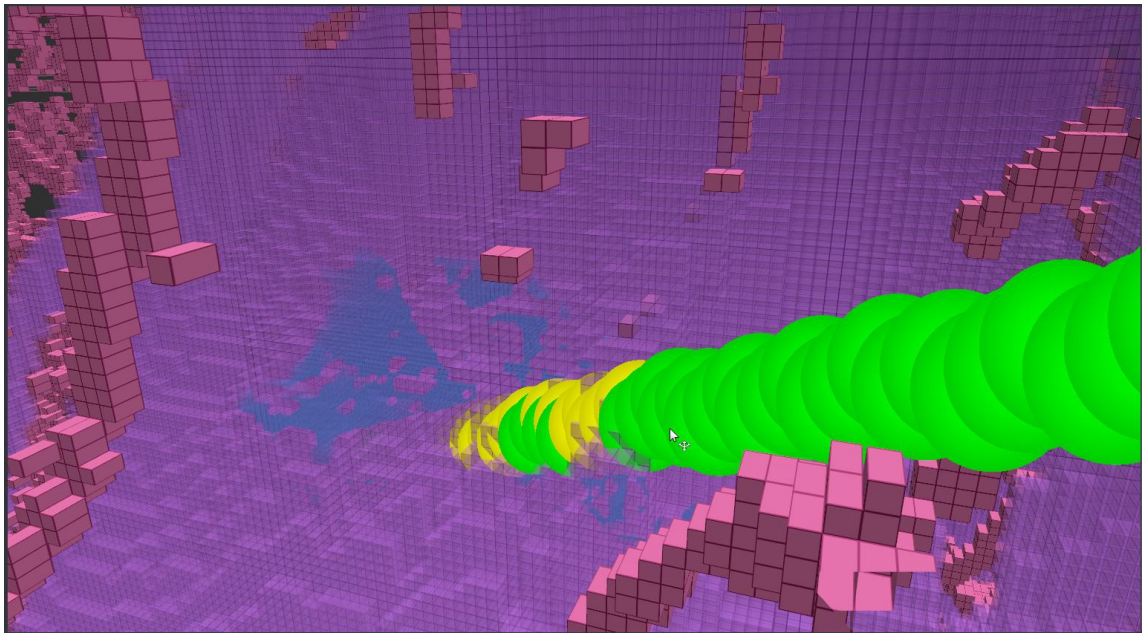


Figure 4.19.: RVIZ visualization of a trajectory with depicted risk per voxel within a color scale from green to red [118].

The architecture of the ROS implementation is depicted in Figure 4.20. The central ROS node is "Data Management" where all data is managed, like the risk maps, consolidated risk maps and the projection risk map. All functionality is implemented in ROS as described in this section. As a visualization interface RVIZ, the ROS visualization tool, is used. The manual interaction capabilities such as selecting a point on the surface is not implemented as the ROS integration is used in automated processes.

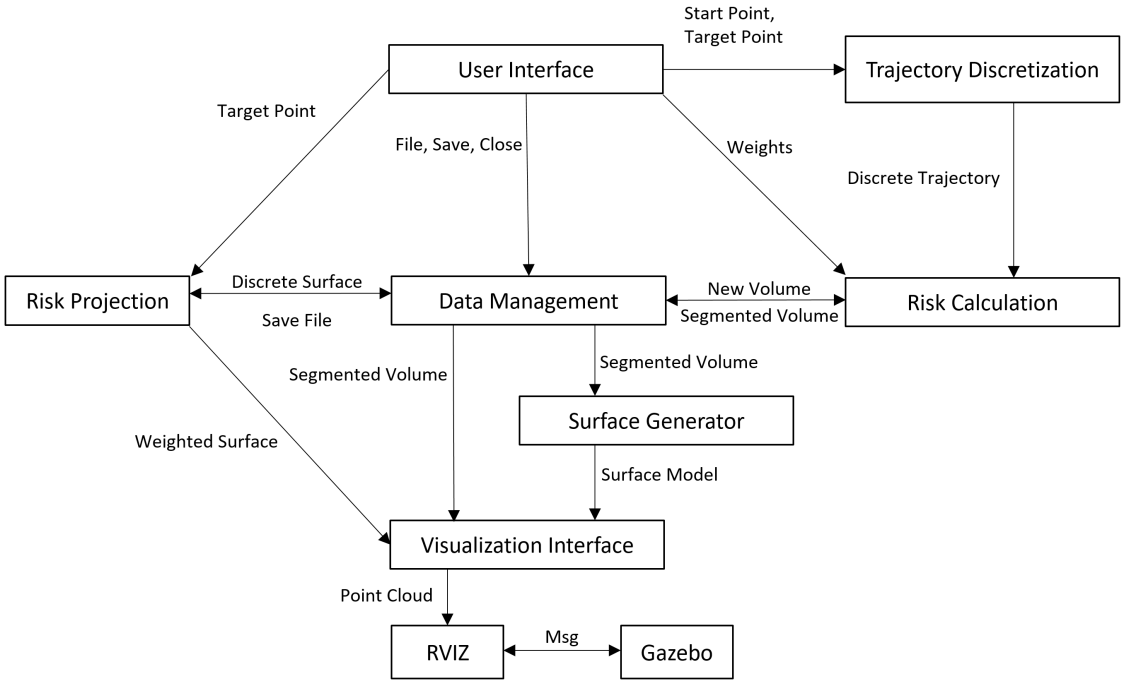


Figure 4.20.: Architecture of the proposed method in ROS.

4.3.6. Evaluation of Risk-based Path Planning

To evaluate the proposed system a quantitative analysis was performed. First, the risk-based trajectory determination is compared to the standard of care, where a surgeon chooses trajectories without tool support. Second, the performance of the system is evaluated. At last, the correctness of the calculations is evaluated and the fulfillment of the requirements is checked. After the implementation of the final system a second qualitative user study was performed. The aforementioned IEEE 2010 data visualization contest dataset was used during evaluation.

4.3.6.1. Comparison to Standard of Care

The proposed neurosurgical planning tool was evaluated by performing a quantitative analysis with expert neurosurgeons ($n = 3$; 1 neurosurgical resident, 2 neurosurgeons). They were asked to plan neurosurgical procedures using a biopsy as an example. Therefore, four realistic target areas were defined inside the brain. Again, the IEEE data visualization contest 2010 dataset (case 2) was used. The four targets are depicted in Figure 4.21. They were given the task of planning a linear trajectory to perform a biopsy of the

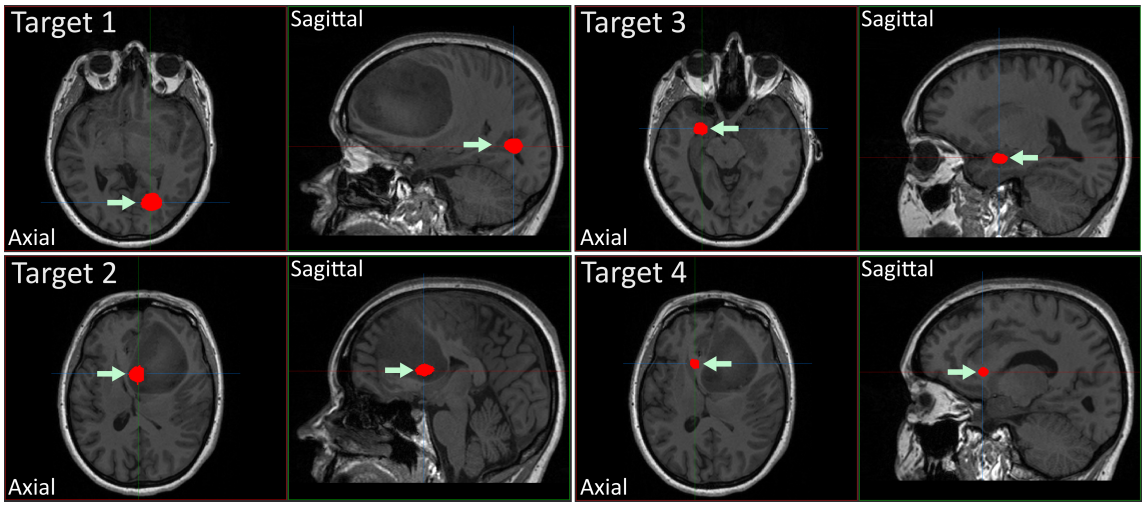


Figure 4.21.: Four targets that are defined in the open source dataset. Adapted from [118].

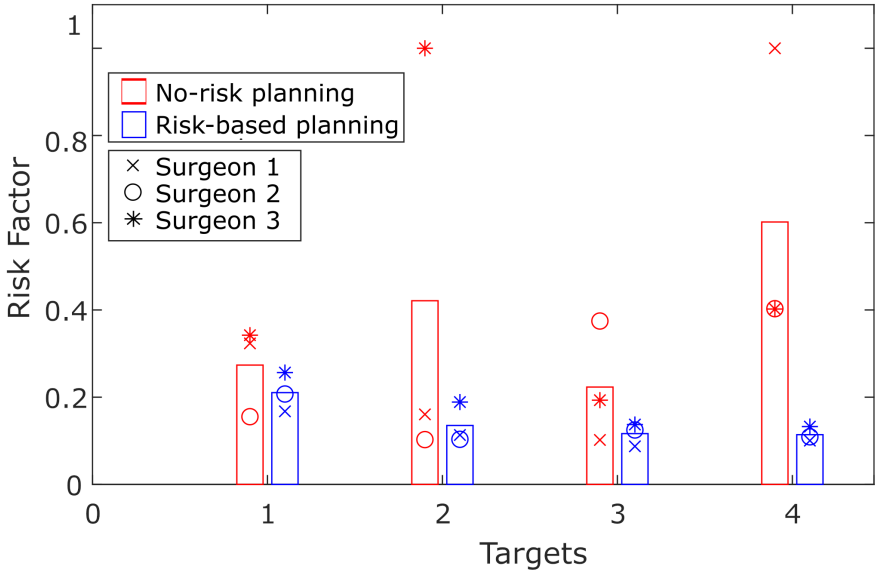


Figure 4.22.: Results of the quantitative analyses [118].

defined targets. The medical experts were free to choose an exact target point within these areas and as it made medical sense.

Initially, the medical experts were asked to perform the planning without the help of visible risk structures and the support of our tool. In a second experiment, they solved the same task using the switched-on risk projection and risk calculation. Thus, each medical expert performed four planning sessions without visible risk structures and four risk-based planning sessions. Each planning was done using the same interface in our tool in order to obtain comparable results. The results are visualized in Figure 4.22.

In almost all cases, risk-based planning resulted in a lower overall trajectory risk. Only in one case was the risk value slightly lower (surgeon 2, target 2). The mean relative risk across all four targets was 0.38 for risk-free planning and 0.14 for risk-based planning. No units are provided for the risk value, as this metric is derived to compare trajectories in a specific patient case scenario.

Two risk structures were perforated directly when no risk planning was used for target 2 and 4. No risk structure was hit when the risk-based planning was used.

4.3.6.2. Computational Performance

The risk-based path planning was investigated in terms of its runtime behavior. Several time measurements were performed. The results are depicted in Table 4.8. The time for initial calculation is determined by measuring the time taken for the generation of the consolidated risk map and the projection risk map, where all trajectory risks are determined. Next, the update time for replanning all trajectory risks with changed weights is measured. The evaluation is carried out in the *joint* mode, since this mode is more computationally intensive.

Table 4.8.: Time measurements of the runtime behavior of the presented approach for low (1'215 vertices, 60 KB), medium (43'280 vertices, 2.7 MB) and high (440'764 vertices, 21.5 MB) surface mesh resolution and according number of trajectories (starting point at each vertex). Initial computation time: Calculation of all individual risk maps, consolidated risk maps and the projection risk map. Update time: Recalculation with changed weights. Measurements were done with 1-4 risk structures [118].

Number of RS	Low	Medium	High
Initial computation time			
1	$4.5s \pm 0.3s$	$11s \pm 0.8s$	$145s \pm 4s$
2	$4.9s \pm 0.3s$	$15s \pm 1.2s$	$365s \pm 5s$
3	$12s \pm 0.7s$	$21.5s \pm 1.5s$	$355s \pm 6s$
4	$12.3s \pm 1s$	$27s \pm 2s$	$485s \pm 8s$
Update time			
1	$0.4s \pm 0.1s$	$1s \pm 0.2s$	$5.3s \pm 0.3s$
2	$0.5s \pm 0.1s$	$1.2s \pm 0.2s$	$5.7s \pm 0.4s$
3	$0.6s \pm 0.1s$	$1.3s \pm 0.2s$	$6s \pm 0.6s$
4	$0.8s \pm 0.1s$	$1.4s \pm 0.2s$	$6.1s \pm 0.7s$

The runtime was evaluated in regard to the number of used risk structures. One additional risk map is being calculated per risk structure and used in further calculations. The more risk structures are used the more computational expensive is the calculation of the trajectory risks. As shown in Table 4.8 the effort grows linearly. Furthermore, different resolutions of the surface mesh are investigated, while each vertex represents a possible starting point of a trajectory. The number of vertices equals the number of calculated trajectories. We evaluated three different meshes with a low (1'215 vertices, 60 KB), medium (43'280 vertices, 2.7 MB) and high (440'764 vertices, 21.5 MB) vertex count. In the proposed tool, this can be selected under 'Mesh Resolution', see Figure 4.24-(III).

Many scenarios are covered by the medium resolution surface model. When investigating the distance between the vertices, a value of 1 to 4 mm can be obtained. Modern CT scanners have a spatial resolution of round about 0.7 mm as found in the used dataset. This resolution can vary, with the best spatial resolution being 0.15-0.3 mm. The high mesh surface has a vertex distance of 0.4 to 1 mm.

Currently, medium- and high-resolution surface models cover all possible scenarios, even when high accuracy is required, e.g. for nerve biopsies. We evaluated the high vertex count to show that our approach is capable of processing any number of vertices, constrained only by the capabilities of the workstation used and the maximum possible resolution of modern imaging technologies. The computation utilizes parallel processing on the CPU. The results of the performance evaluation are summarized in Table 4.8. The evaluation was conducted on a workstation laptop with the following configuration: Intel Core i7 6820HQ, 16GB DDR4 RAM, SSD.

4.3.6.3. Correctness

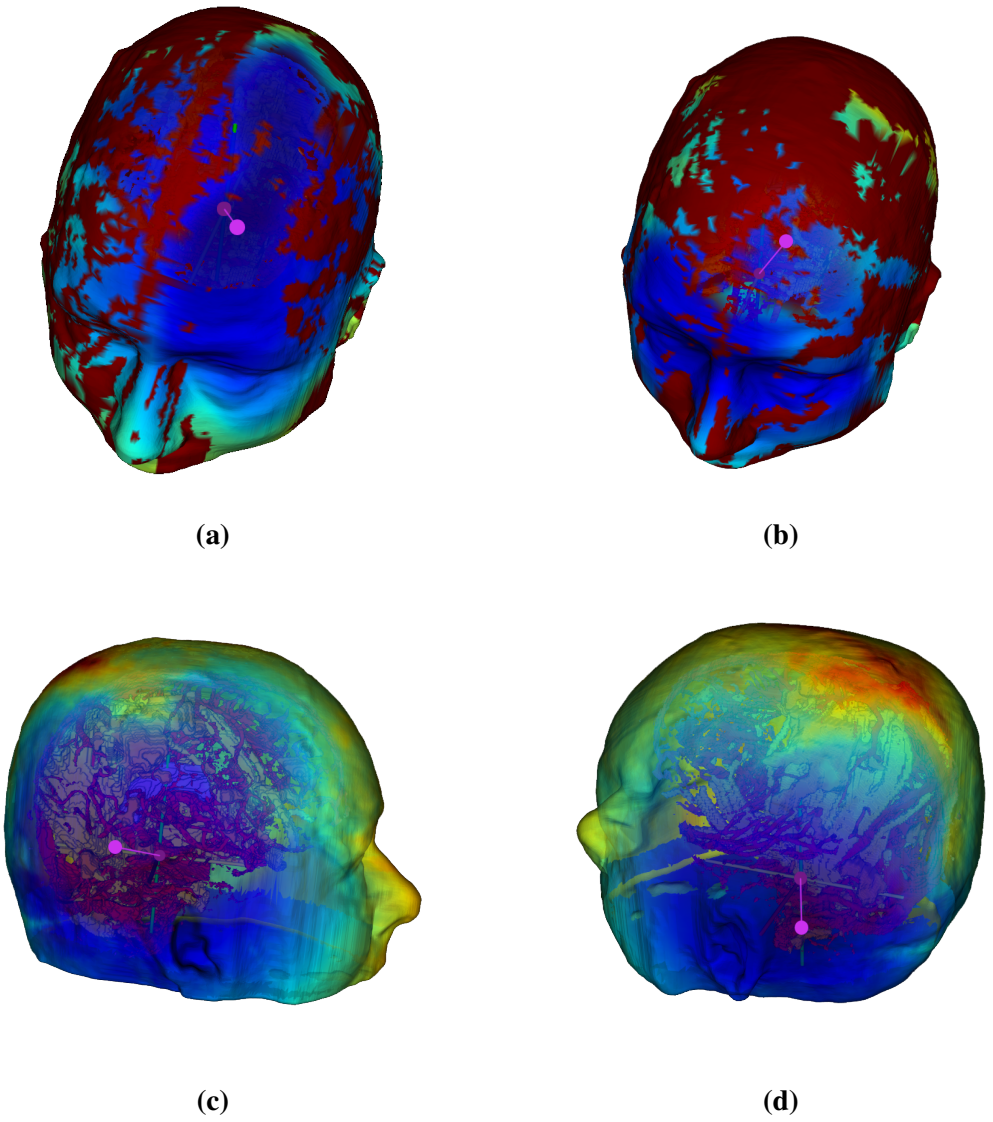
The algorithmic correctness is analyzed with a unit test and additionally verified manually to have a second independent method for risk calculation. Four defined paths are investigated that are depicted in Figure 4.23. The risks of these four paths are calculated as follows: First the voxels that lie on a path are derived by using the Bresenham [27] algorithm. Then the risk of a path is calculated for every risk structure separately and later on added up to the total path risk. Two paths are calculated in joint mode and two in strict mode. This equals the risk calculation presented in Equations 4.24 and 4.29. The risks are weighted equally with 1 and normalized with the maximum risk of all paths. These results are then compared to the results that were derived with the proposed method. The external validation yielded the same results as with the proposed method. Detailed results are depicted in Table 4.9. Figure 4.23 shows the corresponding paths, projection maps and entry points.

Table 4.9.: Correctness verification of the path risk calculation. The overall path risk is calculated from the 3D risk maps of the blood vessels (BV), the eloquent areas (EA), the nerve fiber tracts (NFT), the ventricular system (VS) and the path length (PT). The risks of the proposed method and the unit test are depicted.

Path	1	2	3	4
Entry Point	$(108, 129, 286)$	$(102, 124, 288)$	$(9, 320, 149)$	$(152, 322, 108)$
Target Point	$(95, 192, 234)$	$(81, 192, 146)$	$(60, 286, 120)$	$(114, 291, 109)$
Mode	Strict	Strict	Joint	Joint
BV	0.1337	1	1.80	1.0796
EA	0.0133	9.65×10^{-7}	0.9295	0.2810
NFT	0.0025	0.0186	0.027	0.9359
VS	0.0003	7.97×10^{-14}	8.46×10^{-8}	9.12×10^{-24}
PT (mm)	41.21	75.57	55.35	40.56
Risk - Unit Test	0.0299	1	0.03	0.013
Risk - Proposed Method	0.0299	1	0.03	0.013

4.3.6.4. Requirements Verification

The implemented standalone tool for risk-based path planning is visualized in Figure 4.24. The individual GUI elements are numbered from I to X and are explained in detail below. Their functionality is mapped to the appropriate requirements. We designed the



(a) **(b)**

(c) **(d)**

Figure 4.23.: The four chosen paths of the correctness validation. Results are depicted in Table 4.9.

proposed approach in close cooperation with medical experts to meet all clinical requirements (RD1) and match the surgical planning workflow (RD2) as depicted in Tables 4.6 and 4.7.

A surgeon can investigate the patient scans (RD3) in the coronal (Figure 4.24 I-a), axial (Figure 4.24 I-b), and sagittal (Figure 4.24 I-c) planes and in the 3D-view (RD4, Figure 4.24 II). The 3D view visualizes the chosen surface mesh. It is possible to load surface meshes of any resolution (Figure 4.24 III). The risk of each trajectory is projected onto this surface by coloring each vertex according to its risk value within a blue to red color range. Surface meshes of the skin, cranial or cortical surface can be chosen (RD11, Figure 4.24 II). It is possible to change the opacity of the projection on the surface model as desired (Figure 4.24 IV). The dataset dimensions (RD8) are displayed in the left upper corner of the 2D planes (Figure 4.24 V-a) and in the right corner of the 3D visualization plane (Figure 4.24 V-b).

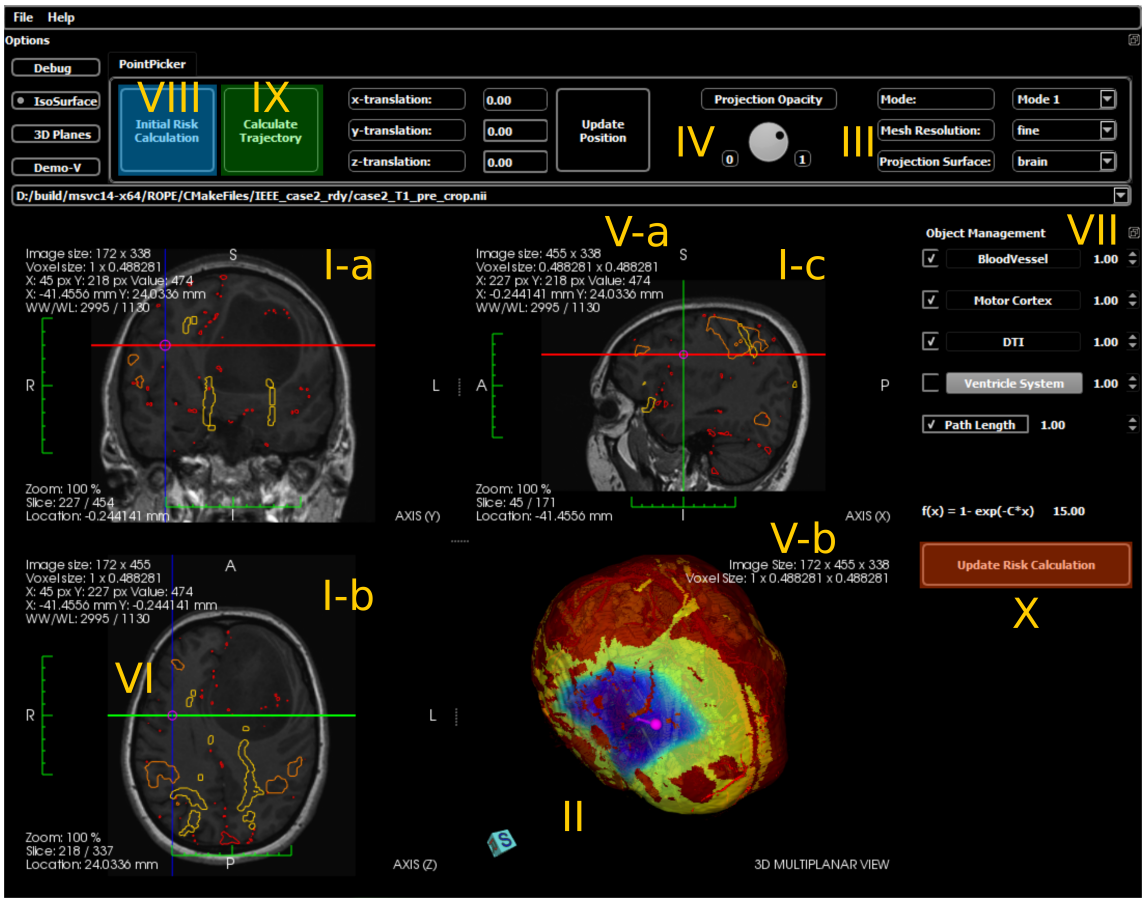


Figure 4.24.: Prototype of the implemented tool for the risk-based path planning. Adapted from [118].

The target point inside the brain can be selected in the 2D planes by simply clicking on the desired point (RD5, Figure 4.24 VI). The target point defines the end point of every trajectory. This can be for example where a catheter needs to be placed or a biopsy needs to be performed. The target point is also visualized in the 3D plane. In the 3D view, the entry point on the surface model can be chosen (RD6, Figure 4.24 II). A crosshair in all three 2D planes displays the current position (RD7). The crosshair can be switched on and off as desired. The trajectory is visualized from the target point to the entry point in the 3D plane (RD9, Figure 4.24 II). The perspective in the 3D view can be interactively changed with the mouse.

The modular architecture allows to load an arbitrary number of risk structures (RF2). The loaded risk structures are superimposed over the patient data in the 2D views and visualized as voxels in the 3D view (RD10.1, RD10.2). The entry point planning can be performed on any registered surface mesh. Therefore, our approach is not limited to the brain. For each loaded risk structure an individual weight can be set by the surgeon (RF1, Figure 4.24 VII). The initial risk calculation can be started with the button "Initial Risk Calculation" (Figure 4.24 VIII). To do so, first the target point needs to be selected. The initial calculation determines all trajectory risks from all possible entry points on the chosen surface to the target point - the projection map, as seen in the 3D view. After selecting the entry point the trajectory can be set by clicking on the button "Calculate Trajectory" (Figure 4.24 IX). After changing the weights a fast recalculation can be started (see "Update Risk Calculation" button in Figure 4.24 X).

Hard and soft constraints as defined in Section 4.3.4 are implemented in *strict* mode (RF4). A maximum path length can be set (RF3) and a weight. The tool stores the calculated risk maps as nrrd-files and all trajectory information for re-use as a machine readable binary file (RF5.2) and a human understandable yaml-file (RF5.1). The full functionality was integrated in ROS (RF6) as described in Section 4.3.5. The system was designed in a way that the update time is fast (RF7) at the cost of a higher initial computation time. All requirements were implemented in the risk-based path planning tool and as ROS nodes.

4.3.7. Discussion

When the approach for risk-based path planning is used in a real clinical scenario, several considerations must be taken into account. A serious risk is brain shift, as it can lead to a significant displacement of the risk structures, and thus the proposed system would not correctly indicate the current situation. Brain shift occurs mainly when large sections of the skull surface are removed. However, in scenarios such as ventricular puncture or biopsies, only a small incision is made which leads to no significant brain shift. To account for small changes, the risk maps can be dilated with a gaussian filter. Alternatively, risk areas in the projection risk map can be extended with an amplification function on the surface. These generated safety margins can compensate for uncertainties.

A further challenge can be the unknown potential deformation of the used tool (e.g. a compliant catheter), which is compensated for using the same method. The behavior of a tool inserted into the brain remains an open research question and scientific data is lacking. One solution may be active instrument tracking or the use of stiffer tools. The patient movement can be tracked with the marker system that is presented in Section 5.4. Furthermore, a stereotactic frame could be used to fix the position of the patient head.

Entry point planning of procedures on the brain is a challenging task, as multiple fine structures in the human brain have to be considered, e.g. blood vessels, eloquent areas and nerve fiber tracts. Sometimes avoidance of all these high-risk structures during certain procedures is not possible. It is also difficult to weight the individual risk structures in relation to each other. Here, a medical expert has the freedom to adjust the weighting according to their expertise, to consider a structure to a greater or lesser extent.

Any loaded and registered number of risk structures are combined to a holistic risk map and projected onto the chosen surface. The system supports any registered surface. In this way it is possible to also plan on partial surfaces, which would also decrease the planning time. Therefore, it is important that the surface mesh only consists of possible entry points. When the skull for example is extracted it can happen that there are two layers that represent the skull bone, the inner and outer side. The planning would remain to work, but twice as many paths would be calculated. In this case, one side should be removed.

To date, the medium resolution surface models are mostly sufficient for accurate planning, considering the limited resolution of current imaging techniques. This was also confirmed during the on-site survey with medical experts. The resolution of the used surface models can be freely chosen and is only limited by the capabilities of the used workstation. The system was designed in a way that it meets the surgical workflow and fulfills all clinical requirements. Any registered risk structure can be added to the risk calculation, providing full flexibility. The entry point planning is performed intuitively on the projection risk

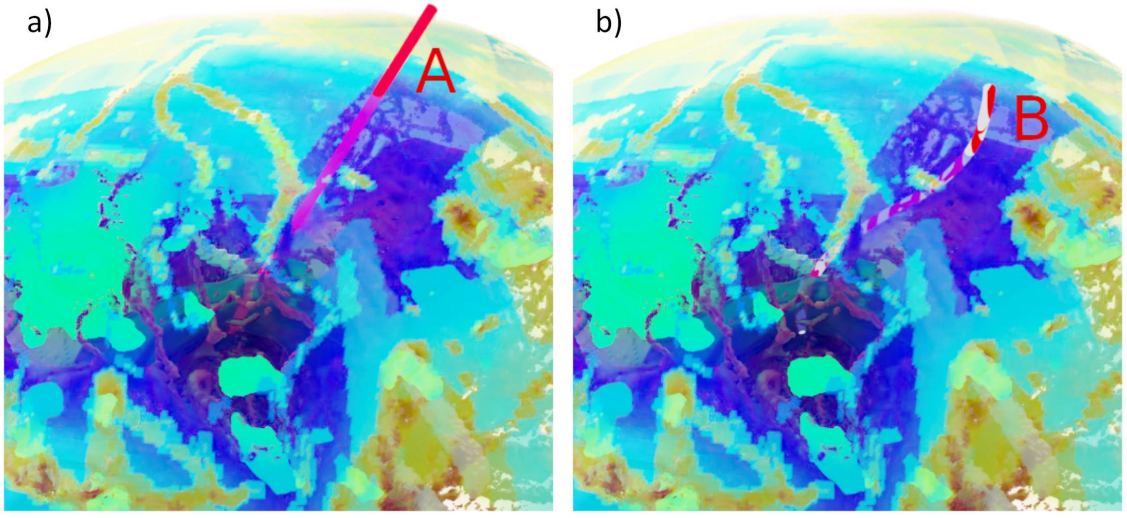


Figure 4.25.: Entry point planning with linear (a) and non-linear (b) trajectories that avoid risk structures [118].

map. The initial computation of the risk maps can take up to 27±2s on average (n = 20) when four of the most important structures are loaded and the medium surface models are used.

The update time is much faster, because the calculated risk maps are held in memory and do not need to be calculated again until a new target is chosen by the user. When the weights are updated, the recalculation time takes up to 1.4±0.2 s. The calculation time is directly related to the vertex count and the number of risk structures. The initial calculation can be started by assistant staff in the beginning and is only performed once. There is the potential to reduce processing time by implementing parallel processing on the GPU. This will be investigated in future research.

The quantitative evaluation performed with three medical experts shows a significant reduction in the risk of the selected trajectories. The neurosurgeons who participated in the evaluation used the tool for both plannings. First, they performed the entry port planning for all four targets without risk support. Then risk support was switched on and the planning was again performed for all four targets. Then all targets again with switched on risk support.

The initial experiments show a slightly faster planning time for the risk-based path planning. Also statements about a time reduction are limited, because normally the surgeons would have used their inhouse planning tool for the no-risk planning.

The main contribution of the proposed system is the risk mitigation. The system can automatically suggest the paths with lowest risk to the medical expert. The implementation in ROS enables integration of the functionality with other hard- and software. The proposed approach is able to compute the risk of any linear or non-linear path, as visualized in Figure 4.25. This allows the use of the proposed approach in combination with steerable needles or flexible instruments. The evaluation was limited to one dataset. In the future the approach will be tested on multiple multimodal datasets.

4.4. Chapter Conclusion

This chapter covers the automated path planning for ventriculostomies, which is the main clinical scenario of this work. Furthermore, an enhancement of the path planning was introduced based on risk structures. The main scenario here is the biopsy of brain tumors. The risk-based path planning can also be applied to the ventricular puncture to derive the optimal path towards the ventricular system.

To automatically derive the puncture path for ventriculostomy, structures in the patients head must be segmented. In Section 4.1 an algorithmic approach was presented for fast volumetric segmentation of the skin, the skull and the ventricular system from individual patient CT image data. A process has been proposed that consists of four steps. First the subarachnoid space is removed to delete unfavorable large accumulations between the skull and the brain. Then noise is removed and possible voxels are selected that could belong to the ventricular system. These voxels are grouped together to clusters and only clusters of a certain size are maintained. In the last step, the clusters of the ventricular system are selected, which are the one or two largest clusters that are found.

The voxel selection is performed using a binarization function that utilizes a local operator. The Hounsfield Unit of cerebrospinal fluid and the neighborhood of a voxel is thereby taken into account. The proposed method is also capable of segmenting the skull, the skin and blood infused ventricles. The skin is segmented with a local operator in regard to the neighborhood of air. The skull is segmented with a point-based operator, because bone tissue has a very unique Hounsfield Unit. The blood infused ventricles are segmented using two ranges in the local operator, so that not only voxels of cerebrospinal fluid are considered but also the Hounsfield Units that represent blood. The presented approach is capable to segment the structures very fast in six seconds. A high segmentation accuracy is reached, with a F1-score of over 95%, when pathological structures that are calcified are not considered.

Based on the segmented structures the puncture path is automatically determined with the utilization of statistical shape models. Mean models are generated from exemplary patient data that reflect a wide range of possible ventricular systems and skulls. These mean models are in a fitting step deformed to match the patients anatomy. The target points inside the ventricular system and the Kocher's points are defined on the mean models and moved according to the deformation. A correct Kocher's point is determined in 98.4% of cases and a target inside the ventricular system in 95.4%. This leads to a successful planning in 93.9%.

Consecutive to the path planning based on surgical knowledge, a risk-based path planning is presented that can minimize the risk of penetrating high risk structures of the brain. A risk map is presented that projects the risk of a linear trajectory to the cranial surface, by taking all risk structures on a path to a target into account. It was shown in an exemplary evaluation that a significant lower risk can be achieved and the penetration of risk structures can be prevented.

5. Scene Registration

In this chapter, scene registration methods are proposed to enable tracking of a patient for augmented reality support during neurosurgical procedures. The structure of the chapter is visualized in Figure 5.1.

Figure 5.1.: Overview of chapter 5.

First, the experimental setup is described that is used to evaluate the proposed methods of this work (Section 5.1). Next, an overview is given about possible optical tracking methods. Based on these, markers are presented that could be used to enable robust patient tracking (Section 5.2). Selected markers were evaluated, compared to each other and the results are presented. In Section 5.3, methods for tracking of infrared markers with the HoloLens are introduced. The markers are mounted to a marker system that can be attached to the patient (Section 5.4). Once the marker system is attached to the patient, a marker system to patient registration needs to be performed to enable tracking of the patient when only the marker system is visible (Section 5.5).

5.1. Experimental Setup

To evaluate the methods developed in this work, a custom-built medical phantom was created. Multiple measuring components were available at the laboratory to evaluate the proposed methods implemented during this work. The experiments were conducted in the digital operating room 'OP:Sense' as described in Section 5.1.2. All components are described in the following.

(a) (b)

Figure 5.2.: a) OpenHead side view. b) Front view.

Figure 5.3.: Ventricular systems of the OpenHead. a) View from above. b) Front view.

5.1.1. Phantoms: Synbone and OpenHead

Two head phantoms are used for evaluation of the implemented system. The first one is a commercially available phantom (Synbone, Switzerland) consisting of a plastic skull with foam forming the ventricular system and a detachable skin as depicted in Figure 5.6.

The second phantom 'OpenHead' was rendered from multiple patient scans to create a realistic virtual patient model. A side and front view of the CAD model of the OpenHead is depicted in Figure 5.2. Five ventricular systems from real patient data were modeled and can be attached to the inside of the OpenHead as visualized in Figure 5.3.

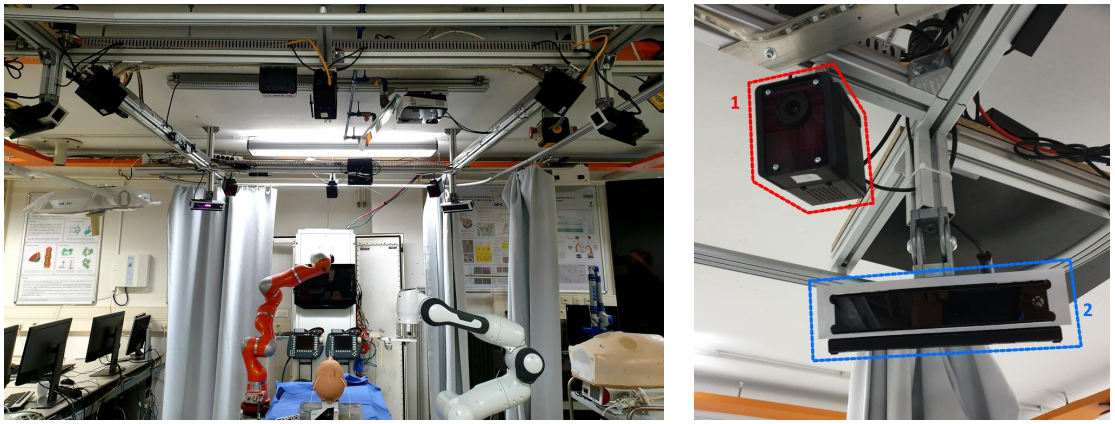
The OpenHead is printed from ASA filament on a Stratasys F370 3D printer (Stratasys, USA). The OpenHead has a detachable and interchangeable cap in the skull, that renders possible to open the skull. This allows it to be reused for multiple scenarios. A mold of each ventricular system is printed, so it is possible to model the ventricular systems from modeling clay for experiments. The molds can be integrated into the OpenHead. A visualization of the molds is depicted in Figure 5.4.

Figure 5.4.: Molds to model the ventricular systems from modeling clay.

5.1.2. OP:Sense

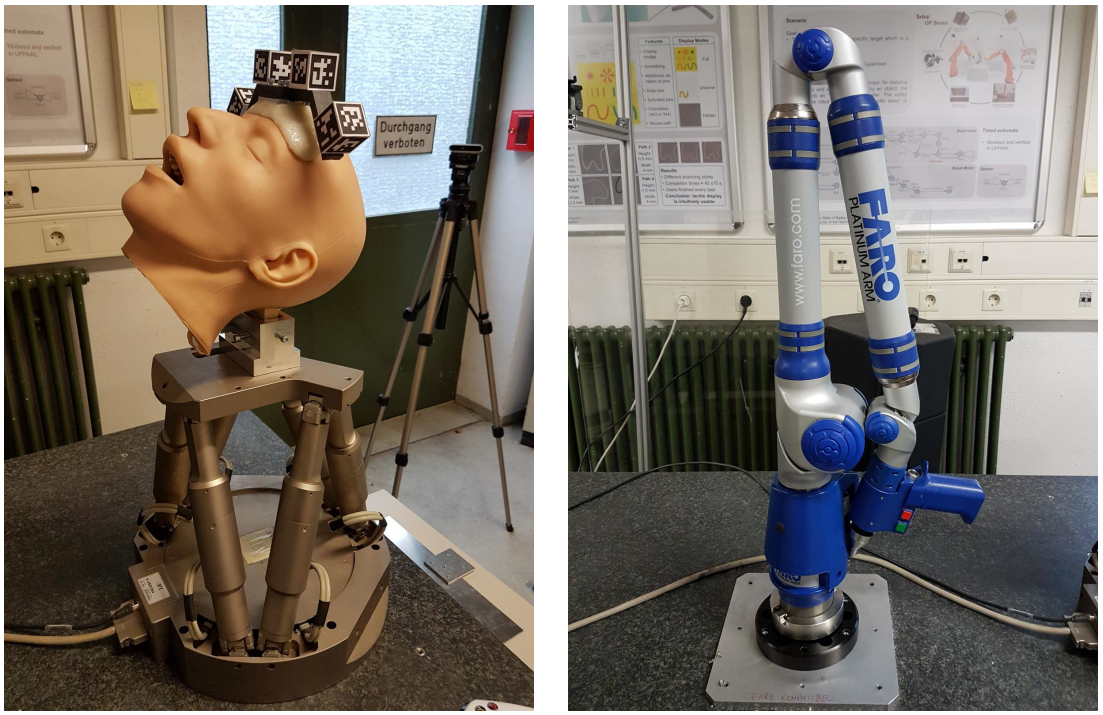
The digital operating room (OR) OP:Sense at the Health Robotics and Automation Lab of KIT includes a moveable OR table with two attached lightweight robots, a Kuka LWR4 and a Franka Panda as depicted in Figure 5.5 (a). The ceiling rack holds multiple optical sensors attached to it: six IR cameras as part of an ARTRACK 2 (ART, Germany) system to track retroreflective spheres and four Microsoft Kinect 2 cameras. The sensors are visualized in Figure 5.5 b).

The ART system (see Section 5.1.2 tracks infrared targets consisting of several infrared spheres that can be attached to any object with known transformation from marker target to object, e.g. surgical instruments. The Kinect 2 sensors provide an RGB stream and a point cloud of the digital OR. Additionally, Intel RealSense D435 and D415 cameras can be registered to the digital OR to capture near-field areas. The basic software framework of OP:Sense is the Robot Operating System (ROS), a middleware for robotic platforms. Many software packages of OP:Sense are implemented in ROS.



(a) **(b)**

Figure 5.5.: a) OP:Sense system setup. b) Sensor setup with six ART-TRACK 2 IR cameras (1-red) and four Microsoft Kinect sensors (2-blue) [115].



(a) **(b)**

Figure 5.6.: (a) Hexapod M-850 with attached Synbone phantom. (b) FARO Platinum measuring arm.

5.1.3. Hexapod

A Stewart platform 'Hexapod M-850' (PI, Karlsruhe, Germany) is additionally available. It is a high precision movement platform with a positioning accuracy of 2 μm. It allows for movement in a range of $\pm$ 50 mm in x and y direction and 30 mm in z. The Hexapod is placed on a planar measuring table as visualized in Figure 5.6 a).

5.1.4. FARO Measuring Arm

The FARO Platinum arm (FARO Technologies Inc., Germany) can be used to take high precision measurements in a defined work space with an indicated precision of ± 0.037 mm. The arm was used in combination wit the FARO CAM2 software. The FARO arm is mounted to a planar measuring table as depicted in Figure 5.6 b).

5.2. Marker-based Optical Tracking Methods

In this section optical tracking methods are presented that use planar or spherical markers, that are potentially trackable through the HoloLens. A selection of potential candidates was made and implemented on the HoloLens for further evaluation regarding runtime behavior and tracking accuracy.

When using marker-based tracking a marker M is attached to an object O. When the transformation T_M^O of the marker to the object is known, it is possible to only track the marker to infer the pose of the object. A good example of this is a surgical procedure, in which the patient is normally completely covered with sterile drape, except the operation site. During the sterile phase, a marker needs to be attached to track the patient, because no visual features are visible that could be tracked, e.g. the face. This is in contrast to markerless tracking, where image recognition methods can detect and localize objects by their characteristic features in the image, as shown for example in the work of Joseph Tan et al. [93].

5.2.1. Comparison of Optical Tracking Methods

In the following, potential tracking methods that can be used to determine an object's pose are listed and their operation is explained. Subsequently, further work is summarized that has dealt with the evaluation of these methods.

5.2.1.1. Aruco Marker

The work of Garrido-Jurado et al. from 2014 describes the development of a marker system for camera pose estimation [69]. By utilizing so-called Aruco markers, it is possible to automatically detect them in RGB images and determine their pose. Instead of using a predefined number of markers, the work introduced a general procedure for generating configurable marker dictionaries in size and number of bits. In addition, a solution is demonstrated to automatically detect the markers in the image and correct possible errors in the process.

The marker system is open source and supported by the image processing tool OpenCV [207]. The application of Aruco markers in optical tracking systems is demonstrated, for example, by a 2015 paper [59], where these markers were used to track the pose of an ultrasonic sensor and eventually control robot movements based on visual data (visual servoing). Aruco markers are characterized by two main components, the automatic dictionary generation and the marker recognition including an error correction. Within the automatic marker detection, individual rectangles in the image have to be detected and combined to a binary code to clearly identify a marker in the image. The algorithm receives a gray-scale image as input and extracts the edges in the first step. For this, a local adaptive thresholding method is first used, which is both more robust to different lighting conditions and also more suitable for real-time applications than, for example, a Canny edge detector (see [31]). For contour extraction, the Suzuki and Abe algorithm is run on the threshold image. Using the Douglas-Peucker algorithm [56], a polygonal approximation can be used to filter out all edges that are not part of a quadrilateral. For contours that are close to each other, only the outermost ones are used, so that finally only the marker

edge is considered. To identify the marker, its code must be determined, so each detected quadrilateral is examined for its inner region. Since markers are usually oblique in space, its perspective projection must be removed by computing the homography matrix. Since the marker is encoded in black and white, the Otsu thresholding method [167] can be applied here, which binarizes the image section.

The final step in marker identification is to decode the binarized image and filter out any erroneous detections. First, the results that do not have a black border are discarded. This means that all border bits must have the value zero. From all further valid results the binary code is extracted, whereby four different codes are possible per marker, since this can be present in four different rotations. If one of the codes is present in a dictionary D, it is declared as a valid marker. The search in the dictionary is accelerated here by an AVL tree, so that the complexity is logarithmic at this point $(O(4log_2(|D|)))$. In this case, |D| describes the size of the marker dictionary. The algorithm differs from others, such as ARToolKitPlus or ARTag (see below), mainly in error correction. If the binary code is not found in the dictionary, it is possible to detect and correct a certain number of erroneous bits. For a minimum distance $\hat{\tau}$ of any two markers in D, up to $\lfloor(\hat{\tau}-1)/2\rfloor$ erroneous bits can be determined [69]. The Aruco library additionally allows the detection of other marker families, such as those from ARTag or ARToolKit.

Finally, to determine the pose of the marker relative to the camera, an iterative minimization procedure, the Levenberg-Marquardt algorithm, is used. The goal is to minimize the reprojection error of the marker corners. To determine the corner points of the markers, a linear regression of the outer marker pixels takes place, allowing their intersection to be determined. ARTag, ARToolKit and ARToolKitPlus also use this method instead of alternative corner detection algorithms [69]. The determined vertices can then be used for the calculation of the spatial pose.

5.2.1.2. AprilTags

In the 2011 paper by Olson [164], a marker system is presented that aimed to improve performance in terms of robustness compared to previous systems such as ARTag or ARToolKit. In particular, illumination changes, occlusions and accuracy of pose estimation are addressed. As with the Aruco markers, the marker system consists of two components, the detector and the encoding system. The detector is implemented to have a low false negative rate but a high false positive rate [164]. With the help of an encoding system, this rate is to be reduced.

The flow of the system is similar to that of the Aruco marker detection. In the first step, the detector also starts with an edge detection, however, in this case the detection is gradient-based. In the algorithm, the direction and value of its gradient are determined for each pixel. Afterwards, pixels with similar values are grouped together (agglomerative clustering) according to the algorithm of Felzenszwalb [63]. This highlights extreme transitions in the image (i.e. edges). Like other gradient-based edge detection methods, this algorithm is susceptible to image noise. However, a low-pass filter can counteract this. Here, the loss of information does not affect the detection of markers, since a marker itself is characterized by significant edges and a low-pass filter has no effect on it.

The next step involves the detection of squares in the image from edges. For this, lines are generated from the clusters formed before using a weighted least squares method.

With additional directional information, derived from the gradient direction, a rule can be established for finding the line segments that eventually form a square. Thereby a line is linked with another one, if it is close enough and follows a winding direction. The distance is defined by a threshold parameter, which can be adjusted and thus has an influence on the robustness against occlusions. In addition, a high threshold results in low false negative rates, but also in high false positive rates [164]. The marker corners then correspond to the vertices of the squares. Since the line segments are computed by interpolation of many edge pixels, the estimation of a marker corner is accurate to a fraction of a pixel.

According to [164], the part for determining the squares takes most of the computation time and can only be accelerated by a lower image resolution. Accordingly, the frame rate of a camera only comes into play if the algorithm has at least the same frequency. This means that should the detection algorithm be able to process fewer images than available over a certain period of time, the frame rate provided by the camera has no influence on the total runtime. To identify the marker, a Direct Linear Transformation [82] (DLT) is used to calculate a homography that maps two-dimensional points of the marker coordinate system to the vertices of the detected square in the image coordinate system. The binary code derived from the black and white tiling within the marker is detected using a thresholding procedure similar to that used for Aruco markers. The coding scheme here is also a modified Lexicode [210], which can detect and fix bit errors depending on the size of the dictionary. The Lexicode system is characterized by the number of bits within a codeword and a minimum Hamming distance between two words. For example, the dictionary "Tag36h11" includes 36-bit coded markers with a minimum Hamming distance of 11.

It is particularly important when generating marker codes that the Hamming distance to other markers remains identical for rotations of the marker by multiples of 90 degrees in its plane, otherwise the marker could be assigned an incorrect identity.

Overall, a good coding system achieves the reduction of false positive detections in the previous step and allows the use of a variety of markers in one image. For example, inner squares would also count as potential markers in the first step, but these are eliminated by the decoding step.

5.2.1.3. Vuforia

Vuforia is a commercially available framework for augmented reality-based applications [170] from the company PTC Inc. (Boston, Massachusetts, USA). It offers multiple libraries to build augmented reality-based apps for the HoloLens, smartphones or tablets. It also features a marker tracking method based on features.

5.2.1.4. Infrared Marker Tracking

As mentioned in Chapter 1, infrared-based tracking offers another way to localize marker-based objects in space. For example, the system of ART (Advanced Realtime Tracking GmbH & Co. KG, Weilheim, Germany) or also the system of Vicon (Vicon Motion Systems Ltd UK, Yarnton, England) requires infrared cameras to be permanently installed in the room, which track spherical reflective markers attached to the objects. Moreover, infrared-based tracking can achieve an accuracy of up to 0.1 mm [7] and also works on

high frame rates (up to 100hz). To achieve this accuracy, multiple cameras are usually used in the scene, for example, up to six cameras with the ART system. Due to the technologies used and the high number of cameras, IR tracking systems require a high installation effort and entail costs of tens of thousands of dollars (see [34]). Nevertheless, infrared tracking systems have established themselves in medical application fields. The company Northern Digital (Northern Digital, Waterloo, Canada), for example, offers tracking systems specifically designed for medical purposes with various infrared cameras of the Polaris series. The Brainlab suite (Brainlab AG, Munich, Germany) also uses infrared markers for patient tracking.

5.2.1.5. ARAM

The 2014 paper by Belhaoua, Kornmann, and Radoux [20] presents the development of the augmented reality for Application on Medical field (ARAM) library for optical marker tracking, which is specifically targeted at medical application purposes, such as navigation-guided laparoscopies. Marker detection is performed via classical edge detection using the Canny edge detection, and subsequent Milgram contour closuring.

The runtime is accelerated via a GPU framework. Individual points of a detected edge are combined into a segment and, as with AprilTags, the Douglas-Peucker algorithm is used to interpolate a straight line. Marker code extraction is also performed using an adaptive thresholding method, always sampling a 7x7 pattern. The Aruco library is used for marker coding.

5.2.1.6. ARToolKit

One of the first marker systems was developed as early as 1999 by Kato and Billinghurst with the ARToolKit [100]. The planar square markers consist of a wide black border with an inner characteristic image stored in a database of valid patterns. It is fundamentally different from the other systems presented here in that a template matching algorithm is used for marker identification. As a result, high false positive rates and confusion between markers occur [69]. In addition, the system is sensitive to changing light conditions due to a globally set threshold used for initial edge detection.

Following the development of the ARTag system, a revised version, ARToolKitPlus [221], was released that also uses binary-encoded markers. Public further development of the project was stopped and continued privately by the Studierstube Tracker project [188]. In parallel, ARToolkit was further improved and the latest version is publicly available today as ARToolKitX. It provides its own tool for camera calibration and is not compatible with other calibration systems.

5.2.1.7. ARTag

ARTag markers were developed by Fiala in 2005 [64]. It takes the concept of the planar marker design of the ARToolKit markers and extends it with digitally encoded markers using an unique ID. This resulted in much better detection results in terms of false positive rate (0.0039%) [64]. In addition, binary coding speeds up the identification step.

In total, the system can detect 2002 different markers, each encoded over 36 bits. Similar to the AprilTags system, the detector searches for edges in the image based on gradients and finally forms them into quadrilaterals [164]. Thus, it is more robust to different lighting conditions. ARTag is no longer maintained and is not publicly available.

5.2.1.8. ChromaTag

ChromaTags, developed in 2017, differ from other marker systems by using additional color coding [52]. Mainly, the additional color information is used to reject false positive detections as fast as possible. ChromaTags are processed at 30fps (frames per second).

The color spectrum of the markers is in LAB color space 4. Large gradients between red and green in the A channel are used for initial recognition of the markers. Since this rarely occurs in the natural environment, the gradient can function well as a feature. The black and white edge of the marker implies high resolution in the L channel, which allows a polygon around the marker to be determined. Decoding is done using the B channel.

5.2.1.9. Moiré-based Markers

Another approach for a marker design is presented by Estaña in his 2006 work [61], where the interference of regular patterns (Moiré effect) is used to localize a marker. In this process, one of the patterns is free in space as a marker and is called the primary grid. A second pattern is then virtually superimposed on the camera image, resulting in interference of the patterns at the point where the primary and secondary grids intersect. Due to the high number of intersections, the position of the marker can be determined with high precision (up to $5\mu m$ accuracy).

The use case described in this paper is only for the two-dimensional case. However, in the outlook, Estaña describes a possible extension to the three-dimensional case. Interference with a secondary marker would be created at this point by virtually projecting spherical shells in space. However, practical considerations of this have shown that these overlays cannot be interpreted unambiguously using image recognition techniques, since the z-component is lost in the projection. The depth information could only be determined by approximation methods, which limits the accuracy to this step. For this reason, a comparable precision as in the two-dimensional case cannot be expected.

Alternatively, a marker system has been developed by the company Metria (Metria Innovation, Inc., Milwaukee, Wisconsin, USA) which also uses moiré effects for determining the position of objects in space. In this case, the marker is designed so that two patterns are offset and separated by a transparent glass layer, so that interferences occur on the marker at variable viewing angles. The developed system is not publicly available and can only be purchased. However, the work of [10] describes individual functionalities of the detection of these markers.

5.2.1.10. Other Marker-based Approaches

Most marker systems presented so far use square markers. In contrast, RUNETag [21] or Intersense [155] provide a circular marker design. In general, pose estimation of circular

markers requires the use of multiple markers, since in a single one only its identity is encoded and has only one correspondence point with the center [69]. This is also the case for Intersense markers. A RUNE tag marker contains several small salient circles that appear distorted as ellipses in the image and thus the pose can be computed.

It should be noted that in the detection algorithm of Intersense markers, the image resolution is reduced to 320x240 pixels for performance reasons [155].

The CALTag system [11] from 2010 presented mainly a solution for camera calibration, but the markers generated there can also be used for the purposes described here. The marker itself resembles a checkerboard pattern of individual square markers, allowing a large number of marker corner points to be used for pose estimation, promising a more precise result [52],[11].

5.2.1.11. Evaluation Methods

In addition to the marker systems presented above, there are also works that have dealt with their evaluation. In [133], for example, the accuracy of AprilTags is investigated. Among them, the influence of factors such as the camera distance, the number of markers and their position in the image is reviewed. For the evaluation series, a conventional webcam with a resolution of 640x480 pixels at 25 fps (frames per second) was used, calibrated with the classical method of OpenCV. Measurements were simulated with the Gazebo software and the marker was placed in the center of the camera image. In a test, the accuracy of the pose estimation was investigated for marker rotations around three axes and increasing camera distance. For rotations in the plane (i.e. around the z-axis of the camera), it was found that the accuracy of the pose estimation decreases with increasing z-component, but the error in the range from 1 to 4 m remains below 0.02° and 10 cm, respectively. Rotations in the plane had no influence on the accuracy. The results show that small sizes cause a larger error for rotations around the x and y axes (roll and pitch) and with increasing distance. However, larger values for roll and pitch are better detected. The results of another test, in which the influence of several markers on the measurement result were determined, showed significant improvements in the accuracy of the pose estimation already with two markers in one plane. Furthermore, the orientation error could be halved with four markers up to a distance of 5 meters [133]. In addition, it was examined whether the arrangement of the markers influences the result. For this purpose, the markers were not placed in one plane, but with different poses. This reduced the error, especially at greater distances. Finally, it was also noted that the position of the marker in the image has an influence on the pose estimation and has error variations up to 7.1 mm [133]. By simulating each test, the results are not affected by external confounding factors, such as changing light conditions. However, a statement about the marker size, where marker size is characterized by the edge length of the marker, was not made for the measurements in this work. Lastly, an experiment is mentioned in which a given set of markers was to be detected in order to determine the runtime in the process. For a 640x480 pixel image the detection time was 115 ms.

The work of Sagitov et al. [185] compares the marker systems ARTag, AprilTags and CALTag with respect to occlusions and rotations. The individual marker systems were systematically tested for their detection accuracy with rotations around three axes. For this purpose, a marker was fixed to the rotation axis in each case and rotated by specific

numbers of degrees. The coordinate system forming the three axes was aligned with respect to the camera. For AprilTags and CALTag, the publicly available libraries were used. ARTag markers were detected and decoded using the Aruco library. Precision was expressed in terms of detection rates, but how these were formed is not further specified. Overall, AprilTags and CALTag showed better results at different rotation angles around the axes. CALTag showed the best robustness to occlusions, justified by the grid arrangement of many individual marker elements. ARTag and AprilTags markers were only partially detected if only the interior of the marker was affected by occlusions. If outer marker edges were affected, the markers could no longer be detected.

The use of ARTag markers is described, for example, in a paper from 2017 (see [142]), which presented an optical tracking system for medical use. The system was primarily intended to present a low-cost but at the same time accurate marker tracking solution. ARTag markers can be attached to instruments, all of which can be simultaneously detected by the tracking system. Using a stereo camera with a resolution of 1024x768 pixels at a frame rate of 20 fps, localization results could be achieved with an accuracy of less than one millimeter. The marker size was 4 cm [142]. Similar to the medical application, the marker was attached to a pen intended to represent a surgical instrument. This was systematically moved at specified distances on a checkerboard pattern. With known pen position and registration between marker and pen tip, the detection result could be compared with the real position. On average, an accuracy of ± 0.4 mm was achieved.

In a 2004 paper, the accuracy of ARToolKit markers was investigated (see [4]). A commercial webcam with a resolution of 640 x 480 pixels at 15 fps was used. The orientation of the marker to the camera, i.e. its registration, was known. The marker length was 5.5 cm. A test sequence consisted of 250 measurements, and between each sequence the rotation around the y-axis and the distance of the marker were varied with step sizes of 10 cm and 2.5°, respectively. The standard deviation of the marker's position estimate was 2 ± 6 cm for a camera distance ranging from 20 cm to 60 cm. Overall, the best detection results were obtained with a camera distance between 20 cm and 50 cm and a y-rotation between 30° and 80°. Measurement series with rotations around the other axes were not performed.

In addition to the development of a library for the detection of ARAM markers, the second part of the work [20] focuses on the error analysis of the system, in particular on the investigation of the error in the primary step of the detection of the marker edges. In the range between 20 cm and 70 cm in z-direction (camera distance), error values in pose estimation in x- and y-direction of less than half a millimeter were obtained. The error of the z-component was less than one millimeter up to 60 cm. Camera resolution, marker size and lighting conditions remained unchanged during the measurements. The rotation error was between 0.05° and 0.22° in the mentioned range. Overall, it was found that the error grew with increasing distance of the marker from the camera. Furthermore, the error was investigated for different viewing angles in the range $\pm 40°$, with similar good results of error values below two millimeters and below half a degree. The runtime of the system was finally compared with that of the Aruco v2 library [69]. Here, the ARAM system showed better performance on average while capturing more than 1000 frames at a resolution of 640x480 pixels. Lastly, the paper noted that the use of multiple markers whose relative position to each other is considered known can counteract the occlusion problem.

The 2018 work [179] by Romero-Ramirez, Muños-Salinas, and Medina-Carnicer focuses on runtime analysis of Aruco markers. Higher image resolutions have become increasingly cost-effective over time, but they lead to ever-increasing computation times for marker detection. In the paper, an approach to speed up the detection algorithm is presented and subsequently compared with the runtime of the original method and other methods. Specifically, the modified method is intended to accelerate marker detection in video sequence. The segmentation step in the image and the extraction of the marker code account for the largest share of the computation time [179]. Image size and runtime of the local adaptive thresholding method for segmentation are in proportional relationship. Moreover, the pixel length of the detected contours in the image grows as the image size increases. This in turn increases the computation time, both in the subsequent contour filtering step and for generating the normalized square-shaped image from the contours. In order to accelerate the first step, a global thresholding method was used. For this purpose, the image is first downscaled in relation to the minimum marker size required for detection. Here, the marker size is related to the original image size and a factor that has to be determined. The paper shows a method to dynamically determine this parameter in a video sequence. In the thresholding method, a global threshold is adjusted until a marker is detected, or a maximum number of attempts is reached. A histogram of all pixel values of each detected marker is created and an optimal threshold value for the next frame is determined using the Otsu algorithm [167]. The threshold can be used under the assumption that consecutive images of a video sequence differ only minimally in their scenery. The subsequent contour extraction step is similar to that of the original process, whereby contours that do not reach a minimum are discarded. This minimum is also determined using the smallest possible pixel size of a defined marker. Next, an image pyramid is created, i.e. a staggering of images scaled to different sizes. The images are reduced by a factor of 2 until the length dimensions of the minimum marker size are reached. The staggered images are used for marker code determination by selecting the image of the pyramid for each edge, in which the edge length approximately corresponds to the minimum marker length. The further processing for code extraction corresponds to that of the original algorithm. Based on the results of this work, a global value of 32×32 pixels was determined for the minimum size of the extracted marker for any resolution starting from 480x640 pixels. Since the accuracy of the pose estimation depends on the detection of the marker corners and this becomes less accurate with increasing scaling, a subsequent upsampling is performed in the range of the initially estimated corner position. Thus, the precision of the pose estimation remains the same in the modified algorithm. In the subsequent evaluation of the modified method, different resolutions of a video sequence were used for runtime analysis. The Aruco v3 method achieves nearly 40 times the speedup of the conventional Aruco method when given an image resolution of 3840×2160 pixels. The running time for one iteration of the algorithm in this case was 2.755 ms for the newly developed method and, consequently, just under 110 ms for the conventional method [179].

Another work by Babinec et al. [15] focuses mainly on the analysis of the detection accuracy of Aruco markers depending on the camera system used. For this purpose, two cameras were selected, a webcam with a resolution of 1 MP and a network camera with the resolution of 0.44 MP. The latter is equipped with a zoomable lens, allowing a larger depth range to be covered dynamically. For the measurements, a marker size of 10 cm edge length was defined and initially checked whether a marker was detected. Two zoom settings were selected for the network camera. Finally, the position of the marker was

sampled by both cameras at distances of 20 - 150 cm in steps of 10 cm each. The error (the difference between detected and real marker position) is measured in each setting. The results show that webcam and network camera in a normal setting both have a small error of 0.5 cm up to 1.2 cm for a distance larger than 110 cm. The network camera in wide angle setting produces a much larger error of up to 6 cm. In a test to determine the influence of different lighting conditions, the network camera showed better performance under strong background light, as it is able to attenuate overexposed pixels. Lastly, this work also confirmed in an experiment the assumption that multiple markers can improve the detection result. They showed that the accuracy was almost twice as high in the above measurement setup when two markers were used instead of one.

5.2.1.12. Selection of Possible Tracking Methods for Use with the HoloLens

Analyzing the available marker tracking methods for usage on the HoloLens the Aruco v2, AprilTags and Vuforia libraries are the most promising methods and therefore selected for implementation on the HoloLens. These markers can be processed very fast and yield a high tracking accuracy. In addition, they were used in many scenarios. In the presented works, cameras similar to the HoloLens sensor have been used. Therefore, the assumption is made that the tracking accuracy and speed is comparable when processed on the HoloLens. The third version of the Aruco library was not published as this evaluation was carried out.

Additionally, in Section 5.3 an implementation and detailed evaluation of infrared marker tracking on the HoloLens is presented. This approach was possible after all sensor streams were made accessible through the research mode on the HoloLens in 2019.

5.2.2. Offline Evaluation of Aruco and AprilTags Marker Tracking

In this section a detailed evaluation of the Aruco and AprilTags marker tracking is presented. The findings are compared to the evaluation results of the commercially available Vuforia library from literature.

5.2.2.1. Experimental Design

The tracking methods are evaluated on two different sensor systems. First, a D435 RealSense camera from Intel was selected, which utilizes an OmniVision OV2740 sensor. It has multiple resolution options up to 1920x1080 pixels at 30 fps. This sensor is comparable to the front camera of the HoloLens 1. As a second camera system a PTgrey Blackfly S (BFS-U3-51S5C-C) industrial sensor with a resolution of 2448 x 2048 pixels at 75 fps with Computar V0828-MPY (LENS-080C7C) lens was used. The tracking methods were executed on a workstation with an Intel Core i5 CPU, 16 GB of RAM and a GeForce 1060 Ti GPU. The original libraries of Aruco v2 and AprilTags were used and executed under Ubuntu 16.04 on the described workstation. Ten measurements (n=10) were performed for each experiment and the mean error values are computed.

The markers were mounted to a high precision Stewart platform from PI (M-850) as presented in Section 5.1.3 with a movement accuracy of 2 μm to create a ground truth from known movement. The work presented in this section was published in [112].

5.2.2.2. Results

In the following the evaluation results are described.

Tracking Accuracy

The mean tracking accuracy of the Aruco library in relation to the two sensor systems is depicted in Tables 5.1 and 5.3. The results of the AprilTags tracking are summarized in Tables 5.2 and 5.4.

Table 5.1.: Tracking accuracy (mean error) of the Aruco library using the RealSense camera [112].

Aruco	Marker size	Translational system error [mm]			Rotational system error [°]		
Distance		40 cm	100 cm	200 cm	40 cm	100 cm	200 cm
	2 cm	0.715	-	-	4.864	-	-
	3 cm	0.696	3.53	-	1.255	4.664	-
	5 cm	0.270	0.911	8.607	1.035	3.788	4.134

Table 5.2.: Tracking accuracy (mean error) of the AprilTags library using the RealSense camera [112].

AprilTags	Marker size	Translational system error [mm]			Rotational system error [°]		
Distance		40 cm	100 cm	200 cm	40 cm	100 cm	200 cm
	2 cm	0.495	2.441	-	0.951	3.172	-
	3 cm	0.428	1.099	-	0.150	2.465	-
	5 cm	0.298	0.685	2.663	0.176	0.596	1.577

Table 5.3.: Tracking accuracy (mean error) of the Aruco marker library using the Blackfly S camera [112].

Aruco	Marker size	Translational system error [mm]			Rotational system error [°]		
Distance		40 cm	100 cm	200 cm	40 cm	100 cm	200 cm
	2 cm	0.596	5.642	-	2.315	2.505	-
	3 cm	0.435	1.402	22.097	1.221	1.836	4.782
	5 cm	0.234	1.036	9.768	1.079	1.389	4.144

Table 5.4.: Tracking accuracy (mean error) of the AprilTags marker library using the Blackfly S camera [112].

AprilTags	Marker size	Translational system error [mm]			Rotational system error [°]		
Distance		40 cm	100 cm	200 cm	40 cm	100 cm	200 cm
	2 cm	0.379	1.138	12.156	1.024	2.142	2.895
	3 cm	0.269	0.993	3.812	0.222	2.058	2.776
	5 cm	0.259	0.560	1.384	0.165	0.452	1.126

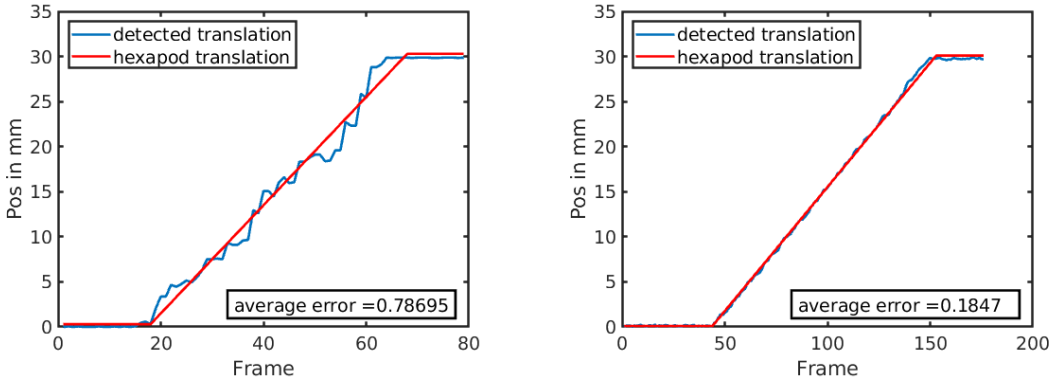


Figure 5.7.: a) Tracking curve of a single Aruco marker. b) Tracking curve of a 5x5 grid. Camera system: Blackfly S, marker size: 3 cm, distance to camera system: 40 cm [112]. The Hexapod movement is the ground truth.

In the range of 40 - 80 cm used for neurosurgical interventions a tracking accuracy of under 1 mm can be determined for the Aruco tracking method when a marker of size 5 cm is used in combination with an Intel RealSense camera. The AprilTags marker method achieves slightly higher accuracy values.

Both libraries support the tracking of multiple markers in one frame. It was further investigated how multiple tracked markers change the tracking accuracy and the jitter behavior. The results are depicted in Table 5.5 and directly compared to the results tracking one marker.

Table 5.5.: The mean rotational and translational localization error of the Aruco and AprilTags tracking method in relation to the number of markers used (grid). In the experiments the Blackfly S camera was used at a distance of 40 cm. The marker size was 3 cm. The tracking accuracy of one marker is depicted for comparison (Single) [112].

	Aruco				AprilTags			
Size	System error				System error			
	Translation [mm]		Rotation [deg]		Translation [mm]		Rotation [°]	
	Grid	Single	Grid	Single	Grid	Single	Grid	Single
2x2	0.249	0.435	0.617	1.221	0.173	0.269	0.203	0.222
5x5	0.159	0.435	0.403	1.221	0.161	0.269	0.074	0.222

In Figure 5.7 the tracking curve of a single Aruco marker a) and a 5x5 grid b) is depicted.

When multiple markers are used, the tracking curve can be smoothed by minimizing fluctuations.

Runtime Behaviour

The runtime behavior was determined by measuring the time needed to process one frame to detect one marker. A marker size of 5 cm was used, placed 40 cm away from the camera. The results are depicted in Table 5.6.

The Aruco tracking reaches 10.64 Hz and the AprilTags tracking 4.78 Hz using the Intel RealSense camera. The Aruco library is therefore twice as fast as the AprilTags tracking. When using the Blackfly camera, the tracking speed is halved.

Table 5.6.: Average detection time of one Aruco and AprilTags marker for both camera systems [112].

	Runtime [$\frac{sec}{frame}$]	
Marker system	RealSense	Blackfly S
Aruco	0.094 (10.64 Hz)	0.182 (5.49 Hz)
AprilTags	0.209 (4.78 Hz)	0.541 (1.85 Hz)

Both libraries support the tracking of multiple markers in one frame. The runtime behavior when tracking 25 markers on a 5x5 grid was investigated. The results are shown in Table 5.7. The number of markers tracked in one frame can be neglected as it took nearly the same amount of time to track one or 25 markers.

Table 5.7.: Average detection time of a 5x5 grid of Aruco and AprilTags markers for both camera systems [112].

5x5 grid	Runtime [$\frac{sec}{frame}$]	
Marker system	RealSense	Blackfly S
Aruco	0.108 (9.3 Hz)	0.24 (4.17 Hz)
AprilTags	0.209 (4.78 Hz)	0.541 (1.85 Hz)

5.2.3. Offline Evaluation and Comparison to the Vuforia Marker Tracking

The offline evaluation of the Aruco and AprilTags marker tracking show that the localization accuracy is dependent on the sensor quality, the distance to the sensor and the marker size.

The evaluation shows a mean tracking accuracy of 0.27$\pm$0.32 mm accuracy of the Aruco marker tracking, when a marker size of 5 cm is chosen within the relevant distance range for neurosurgical interventions. When using smaller markers of 3 cm size this accuracy reduces to 0.7$\pm$0.82 mm. The sensor quality is comparable to the HoloLens front camera. The AprilTags marker tracking often yields better results. A tracking accuracy of 0.3$\pm$0.24 mm can be determined for a marker size of 5 cm and a distance of 40 - 60 cm. An error of 0.43$\pm$0.37 mm was found for a marker size of 3 cm. The AprilTags marker tracking performs much better for larger distances and smaller marker sizes. However, this comes with the price of a slower runtime. Aruco performs twice as fast as the AprilTags marker tracking.

The Vuforia marker tracking was evaluated by Kiss et al. [104] and the accuracy was determined to be 0.31$\pm$0.38 mm up to a distance of 100 cm. A smartphone camera was used. One drawback when using the Vuforia marker tracking is, that this library is not open source. Additionally, large marker sizes need to be used. In the relevant distance of 40-80 cm this would be 10 cm. Smaller sizes are possible, but can lead to tracking issues. At the end of this evaluation a newer version of the Aruco marker tracking version 3 was released [179]. The authors state that it runs 40x faster than version 2. Based on the results of the offline evaluation it was decided to implement the Aruco and Vuforia marker tracking on the HoloLens.

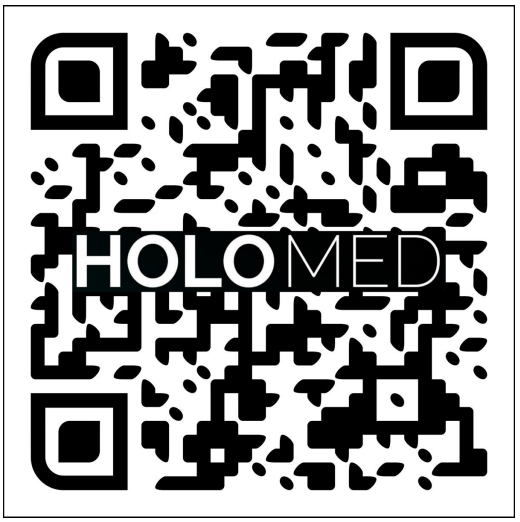


Figure 5.8.: Generated Vuforia Marker.

5.2.4. Implementation on the HoloLens

The Vuforia marker tracking was implemented over its Unity 3D plugin. A custom Vuforia marker was generated visualized in Figure 5.8. A stable tracking could be achieved with 25 fps.

HoloLens applications are implemented using Unity 3D, which uses the programming language C#, but the Aruco library is implemented in C++. In general, there are two ways to implement Aruco marker tracking on HoloLens, through a dll-import or winmd. Both solutions were successfully implemented on the HoloLens. Nevertheless, none of both approaches yielded a sufficient frame rate of 25 fps. The measured frame rates were low, i.e. under 5 fps. One possible explanation for this is the low performance of the HoloLens 1. For this reason it was decided to use the Vuforia marker tracking for patient tracking in further experiments.

5.2.5. Discussion

Multiple options exist for optical patient tracking based on planar or spherical markers. The most promising marker tracking method is the Aruco library. It yields a good tracking accuracy and has a fast runtime. Unfortunately, no satisfactory performance could be achieved on the HoloLens. The reason for this is presumably due to integration problems on the HoloLens. One bottle neck was the availability of new image frames. Theoretically, the Aruco library is able to handle a higher number of frames, but it took longer to capture new frames as expected. This leads to the situation that the algorithm is waiting for a new frame. The ideal case would be that a new frame arrives just before the algorithm has finished processing the last frame. This could be due to performance issues on the HoloLens.

Due to the low performance of the Aruco tracking on the HoloLens it was decided to use the Vuforia marker tracking to localize the patient during an intervention. The library provides a fast processing time and a high tracking accuracy. One disadvantage is that the library is not open source but is free for academic research and non-profit projects.

The current standard of care in tracking technologies in the OR are infrared sphere-based markers. They have a lot of advantages like sterilizability and robustness. After Microsoft made all streams accessible through the 'research mode' it became possible to track infrared spheres with the HoloLens. The solution is presented in the next section.

5.3. Infrared Marker Tracking

In this section the infrared marker tracking on the HoloLens is presented and a detailed evaluation is carried out.

5.3.1. Sensor Capabilities of the HoloLens

The HoloLens provides multiple sensors to track its surroundings. They are listed in Table 5.8. For a more detailed investigation on the sensor capabilities of the HoloLens see [89].

Sensor	Resolution
front camera (photo video) (*fc*)	1408 × 792 1344 × 756 1280 × 720 896 × 504
Four environmental cameras	
left left camera (*ll*) left front camera (*lf*) right front camera (*rf*) right right camera (*rr*)	160 × 480 160 × 480 160 × 480 160 × 480
Near field - Time of Flight (tof)	
short throw reflectivity (*str*) short throw depth (std)	448 × 450 448 × 450
Far field - Time of Flight (tof)	
long throw reflectivity (*ltr*) long throw depth (ltd)	448 × 450 448 × 450

Table 5.8.: Sensors of the HoloLens.

Most of the sensors are accessible over the research mode of the HoloLens (*ll*, *lf*, *rr*, *rf*, *str*, *std*). In general, there are two options to track infrared markers with the HoloLens. The *lf* and *rf* streams can be used or the time of flight sensor streams *str* and *std*. When using the time of flight sensor the *str* stream is mapped to the *std* stream. The *lf* and *rf* sensors form a stereo camera setup. The streams are depicted in Figure 5.9.

Both ways were implemented, because in the *tof* streams the infrared marker spheres are brighter, but the *lf* and the *rf* streams have a higher resolution of 640 x 480 and therefore offer the possibility of a higher tracking accuracy. The *tof* streams have a resolution of 448 x 450 pixels, only capturing data in a centrally located circular area, which leads to large areas of black pixels at the edges.

Infrared marker tracking is usually based on the capability to perform a binarization of the image to clearly identify the infrared markers in a frame. The IR markers light up brightly in the *str* stream. This works even when they are placed several meters away from the HoloLens. The maximum measurable distance of the *str* and *std* stream is up to 4 meters. When using the *lf* and *rf* streams the infrared markers have the same brightness as the surroundings, so no binarization would be possible. In the *str* stream this is easily possible as the infrared markers glow visibly bright.

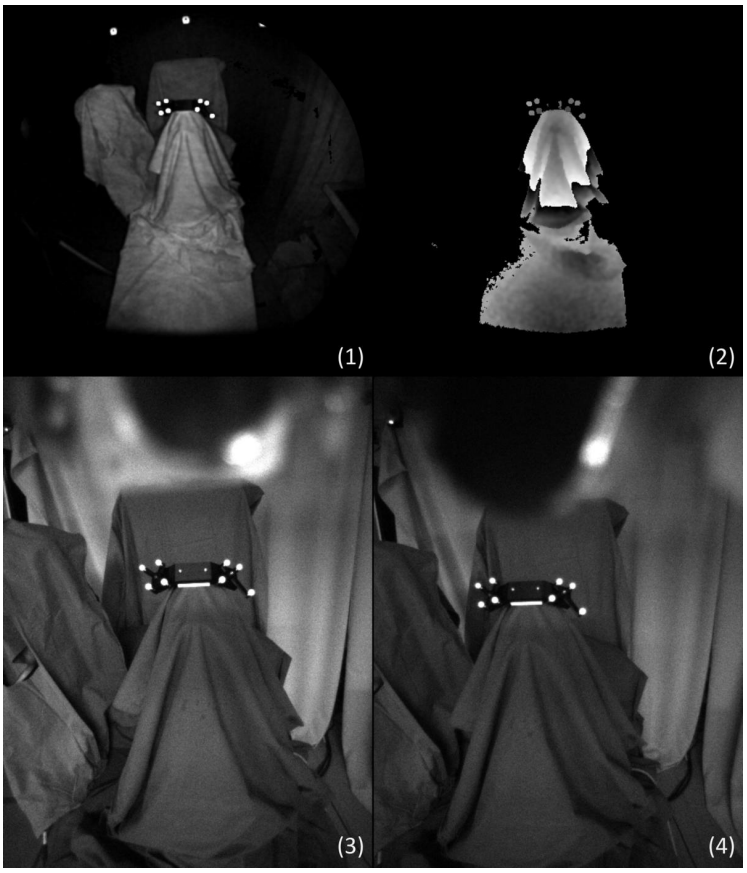


Figure 5.9.: Streams of the HoloLens: (1) *str* (2) *std*, (3) *lf* and (4) *rf*. In the upper part of subfigure (3) and (4) the additional light emitter sources can be seen. Adapted from [116].

The infrared markers need be illuminated with an additional infrared light emitting source in order to increase the contrast in the *lf* and *rf* streams. For this purpose, an IR light emitter was constructed consisting of two LEDs that provide a radiometric power of 530 mW and emit IR-radiation at a wavelength of 850 nanometer per LED. The LEDs need to be adjusted to the center of the *lf* and *rf* streams, because the emitted light needs to be reflected to the sensors. It is mounted on top of the HoloLens as depicted in Figure 5.10.

With the additional light emitters it is possible to perform a binarization of the frame to derive the positions of the infrared marker spheres. Also it must be stated that the overall scene light must not be too bright, else no tracking is possible.

In the following, the process of tracking the infrared spheres on the HoloLens is described.

5.3.2. Process of Infrared Marker Tracking

The goal is to determine the 3D positions of all infrared spheres in a frame. This alone would not enable the tracking of a patient, because it would just allow the determination of three degrees of freedom. To track a patient or any other object all six degrees of freedom need to be determined. To do so a marker target needs to be constructed, consisting of at least four infrared spheres. This makes it possible to track all six degrees of freedom or in other terms the pose of the marker target. The process to track the IR marker targets is shown in Figure 5.11. It will be explained in detail below.

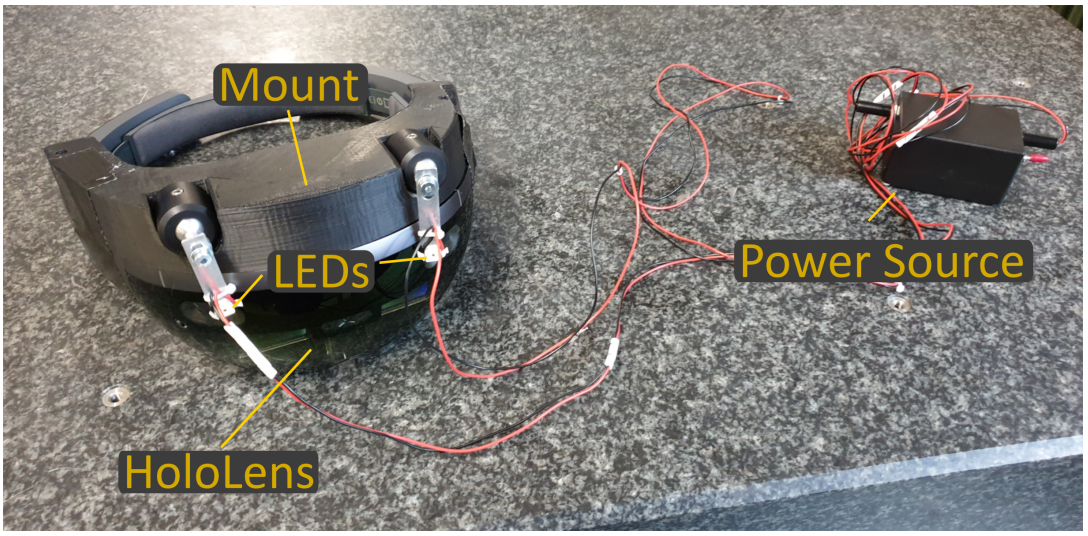


Figure 5.10.: LED mount for the HoloLens to brighten up the scene.

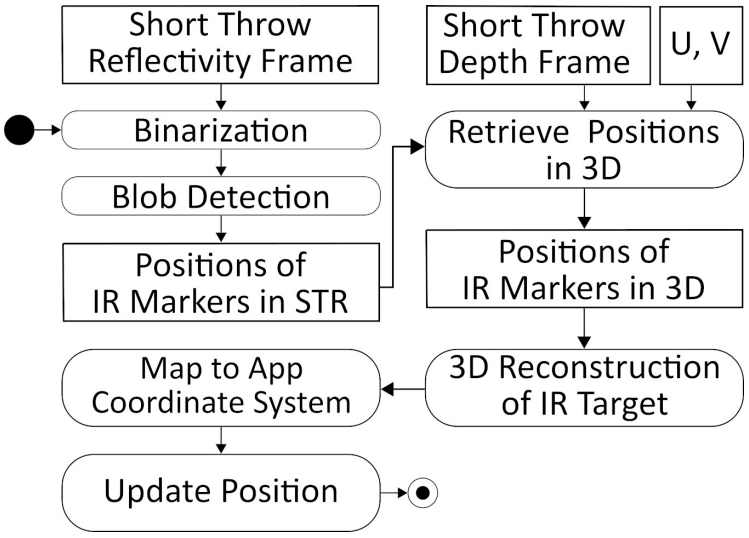


Figure 5.11.: Process for the determination of an IR marker target in the HoloLens streams for pose estimation [116].

In each update cycle of the HoloLens, the current *str* frame is acquired. First a binarization of this frame is performed. All pixels of the infrared spheres are assigned the value 255 (white), all other pixels are assigned the value 0 (black). This is possible because the infrared markers glow bright in the *str* frame. They cause brightness clipping due to their high light reflectance. To do so the binarization function depicted in Equation 5.1 is applied on each pixel p with value p_{val} of the *str* frame S_r. A threshold $T = [250, 255]$ is applied to each pixel to determine if it belongs to an infrared marker sphere.

$$\mathcal{B} : \mathcal{S}_r \rightarrow \{0, 1\}, \mathcal{B}(p) = \begin{cases} 1 & p_{val} \in T \\ 0 & else \end{cases} \tag{5.1}$$

After the acquisition of the binarized frame a blob detection is executed to eliminate all wrongfully marked white pixels. This can happen when there is additional clipping in the scene, e.g. through reflective areas. All these areas are filtered out. Normally these

areas do not form a perfect circle and can easily be deleted. All remaining spherical reflections are deleted in the 3D reconstruction phase. The 2D position of each detected blob is determined in the *str* frame as it is the center of the blob. Now in a next step the 3D position must be calculated. This is not possible by simply using the value of the *std* frame. Instead, an unprojection mapping must be performed to determine the 3D positions [X,Y,Z] of every center point in the *str* frame. To perform the unprojection mapping two additional matrices U and V are used, which can be queried during runtime for every frame. They describe the relationship between the *std* depth stream and the real 3D position in space of every pixel Z_{ij} of that frame.

The 3D positions are calculated as follows:

$$[X, Y, Z] = Z_{ij} \cdot [U_{ij}, V_{ij}, 1] \tag{5.2}$$

A more detailed description of the unprojection mapping is given by Labini et.al. [121].

Now, we have the 3D positions of all retroreflective spheres and maybe spherical reflections that are falsely detected as markers. Next, the 3D reconstruction of the complete infrared marker tracking is performed. To do so the registration or transformation matrix of every retroreflective sphere to each other needs to be known. The 3D reconstruction was done following an approach described in [70]. In a last step the derived transformation matrix needs to be transformed to the *app specific coordinate system*. After the complete process is finished the position of the marker target is updated for further processing.

5.3.3. Evaluation of the Infrared Marker Tracking

In the following the experiments for both methods are described. To measure the accuracy of the infrared tracking the marker targets are mounted on the OpenHead phantom and placed on the Hexapod, which enables the movement in space with a precise known ground truth. The movement accuracy of the Stewart platform is reported as $2\,\mu\mathrm{m}$. A translation is performed in all three axes, x, y and z. The rotational error was not investigated in this study.

The HoloLens was mounted on a stand and aligned with the marker system at a distance of $60\,\mathrm{cm}$. The evaluated recordings were performed directly on the HoloLens by utilizing the *HoloLensForCV* project [148]. The recording was started via voice command. Ten movements for each axis and ten combined movements over all axes were performed in n=42 experiments each. The Stewart platform was moved by a preselected distance, which is the vector length in space. Either variants of the IR tracking algorithms using the *tof* or *lf* and *rf* streams were evaluated offline. The positions of the infrared spheres are determined in the streams throughout the whole movement and compared to those of the Hexapod. The deviation of the vector length in space is determined to be the error.

In addition the runtime behavior of the infrared marker tracking was evaluated when executed on the HoloLens.

5.3.3.1. Results

When using the *tof* sensor, the average tracking accuracy is 0.76±0.65 mm. The results for the *lf* and *rf* streams are in a similar range with a tracking accuracy of 0.69±1.12 mm.

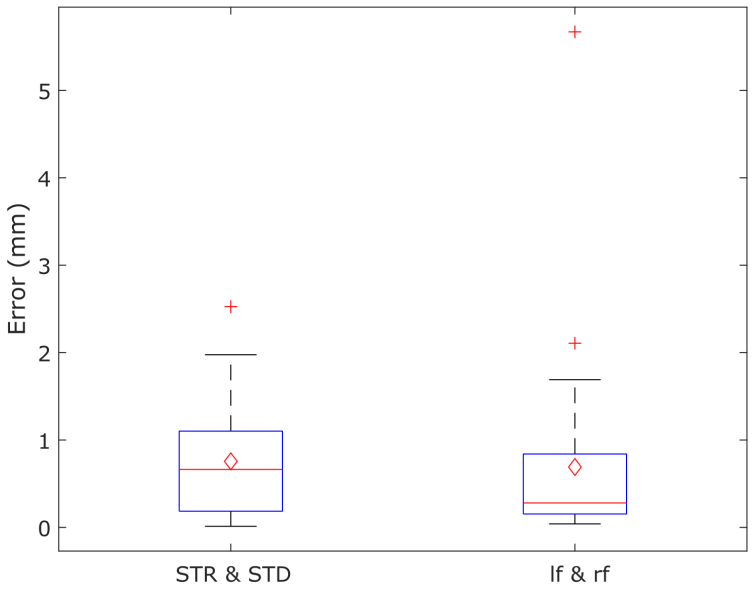


Figure 5.12.: Evaluation of the tracking accuracy (n=42) of the IR marker targets detected in the *tof* (left) and *lf* and *rf* (right) streams. Mean values are visualized as red diamond markers, outliers are depicted as red cross markers. Adapted from [116].

The results are visualized in Figure 5.12 as a standard box plot. The red line indicates the median, the blue box the range from the 25th percentile to the 75th percentile and the black lines the range from the minimum to the maximum. Outliers are depicted as red cross markers. Additionally, the mean value is visualized as a red diamond marker.

The blob detection can detect the centers of the blobs with sufficient accuracy.

It was decided to use the approach utilizing the *str* and *std* streams on the HoloLens. The accuracy is similar, but additional light emitters are needed when using *lf* and *rf* streams. This leads to a higher weight and low environmental light conditions are required.

Results of detected infrared spheres for both approaches are depicted in Figure 5.13.

Table 5.9.: Runtime evaluation of all process steps of the IR marker tracking [116].

Process step	Execution time in s
Binarization	0.000583
Blob detection	0.043253
Retrieve positions in 3D	0.000007
3D reconstruction of IR target	0.00103
Map to app coordinate system	0.000001
Update position	0.000001
All	0.044875 (22fps)
Framerate on HoloLens	55-60 fps
CPU utilization	60%-70%

The results of the runtime evaluation are depicted in Table 5.9. All process stages are investigated in regard to their runtime behavior. Most time is taken to perform the blob

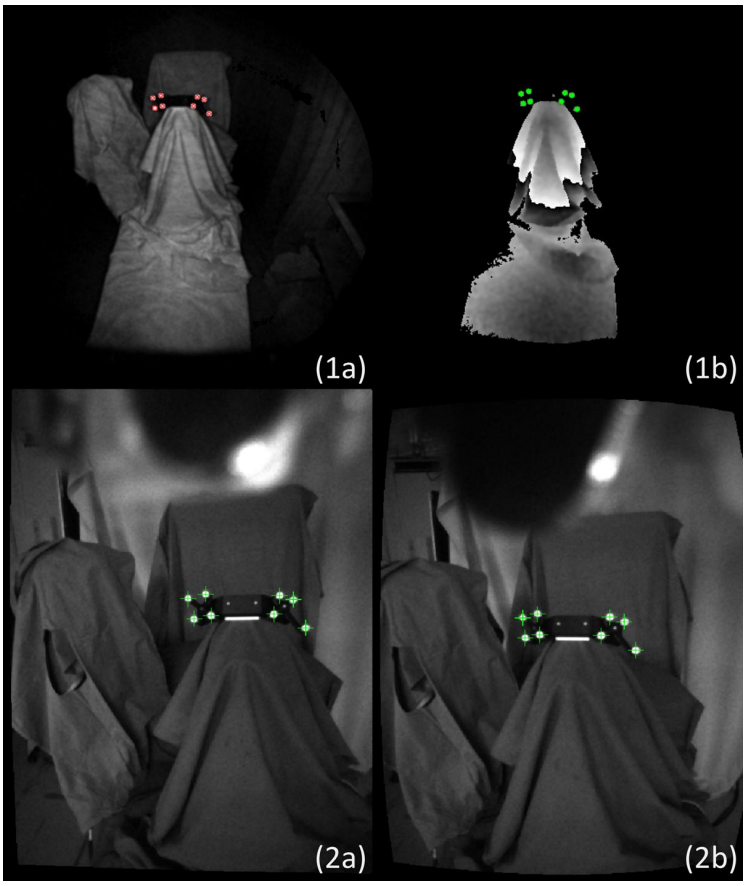


Figure 5.13.: Streams of the HoloLens with detected infrared spheres: (1a) *str* (1b) *std*, (2a) *lf* and (2b) *rf* [116].

detection with 0.043s per frame. Adding up all computation times one frame can be processed in 45 ms. This leads to a theoretic overall frame rate of 22 fps. However, on the HoloLens asynchronous multi-threading is used for the tracking, so that a higher frame rate of 55-60 fps can be reached. With active tracking performance reserves still remain, as the CPU of the HoloLens is only used at 60-70% capacity. It is even possible to track an additional Vuforia marker at the same time. A tracked marker target from HoloLens view is visualized in Figure 5.14.

5.3.4. Discussion

Two different options to enable an infrared marker tracking on the HoloLens were implemented and evaluated. The results indicate that the tracking of infrared marker targets with the HoloLens is possible with a high degree of accuracy. An accuracy of 0.76 mm was found when using the *str* and *std* frames and 0.69 mm when using the *lf* and *rf* streams in $n = 42$ measurements. The infrared spheres can be identified after binarization with a blob detection algorithm. The center points can be determined robustly and with high accuracy. A big disadvantage of the environmental cameras (*lf* and *rf*) is, that the infrared spheres do not glow bright in the frames without an additional light emitting source. This makes the HoloLens heavier and more difficult to handle. Also there is the problem, that the LEDs need to be precisely adjusted to the cameras and that they are visible in the streams as can be seen in Figure 5.14. A further problem is the transformation to the *app*

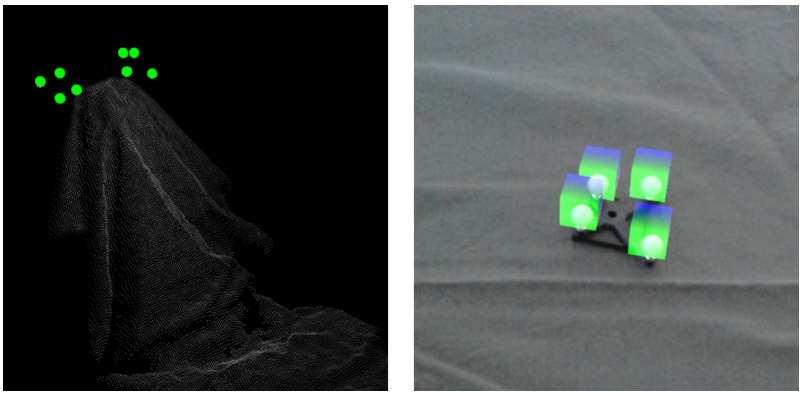


Figure 5.14.: IR markers in reconstructed point cloud after unprojection (left). IR target visualized in HoloLens (right) [116].

specific coordinate system. The HoloLens framework does not directly provide a possibility to perform this automatically as it is possible for the *tof* streams. The *str* and *std* streams have a lower resolution, which lead to a slightly lower accuracy in comparison to the environmental streams. Finally it was decided to use the *time of flight* sensor for tracking and the solution was implemented on the HoloLens.

The *str* and *std* streams run at 30 fps. It was evaluated that the algorithm can be processed with 22 fps. When the application is executed on the HoloLens a stable frame rate of 55-60 fps is reached. The reason is that the visor is updated in a dedicated thread. This frame rate decreases if the HoloLens is too heavily utilised. The pose of an infrared marker target is updated every 45 ms (22 fps). This can lead to a small but noticeable delay of the hologram movement, when the wearer of the HoloLens or the patient is moving. Further, the frame rate of the *std* stream can reduce to under 20 fps during fast movements for a short period of time, which is usually not the case in the scenarios presented in this work. Another limitation is that the tracking of the markers fluctuates by a certain amount. An extended Kalman filter can be used to smooth the tracked position. The blob detection is the process step that takes up most of the processing time. It is being investigated whether this can be optimized in the future.

Id	Requirement
R.MS.1	Fast Assembling Time
R.MS.2	Support of the surgical process change
R.MS.3	High Reattachment Accuracy
R.MS.4	Marker Visibility
R.MS.5	Low Weight
R.MS.6	Sterilizability
R.MS.7	Modularity

Table 5.10.: Requirements of the marker system.

5.4. Marker System

In the prior chapters, markers for patient tracking were presented. These markers must be rigidly attached to the patient to allow online tracking of movements. In the following a marker system is presented that can hold different types of markers and can be attached to a patients head non-invasively. A very important aspect for the application in neurosurgery is the need to support the process change from the non-sterile to the sterile surgical phase. It is not possible to use a non-sterile marker throughout the intervention to ensure hygienic protection. During the process change the patient is fully covered with sterile drape. Thus, a solution must be found for using a marker system in combination with this cover. The marker system was designed in a way that it consists of four major parts that are depicted in Figure 5.15.

The marker system is attached to the head of a patient and fixed to it non-invasively. The marker system to patient registration is performed during the non-sterile phase while the patient is not covered. After everything is prepared the patient is fully covered and only the surgical site remains visible. A coverage of the whole marker system would not be possible, because the marker system could not be tracked. Possibilities to use active LED markers under the foil were examined, but the sterile drape is not completely transparent, which makes it very hard to track any marker. Therefore, the marker system was designed in a way that the marker carrier can be removed after a successful registration. Then the patient and the marker base get fully covered. After that a sterile marker carrier is being reattached.

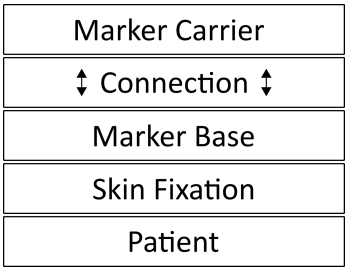


Figure 5.15.: General design of the marker system.

5.4.1. Requirements Analysis

The marker system must fulfill the requirements listed in Table 5.10. These are explained in more detail below:

Fast Assembling Time (R.MS.1): The assembly of the marker system should be able to be performed quickly. It must be easy to mount by medical personnel or neurosurgeons with as few additional aids or tools as possible. There should be only one way to mount the marker system, to avoid mistakes.

Support of the surgical process change (R.MS.2): The marker system must support the process change from the non-sterile to sterile surgical phase. The patient is prepared for the operation in the non-sterile phase. The marker system to patient registration can be performed in this step, while the patient is still completely visible. After the full coverage, this is not possible any more. As the markers must be visible during the intervention and the marker system must be attached to the human body, a solution must be found that the sterile drape can be combined with the system, e.g. between the marker head and the base.

High Reattachment Accuracy (R.MS.3): After registration of the marker system to the patient, the position of the marker head must be maintained. In order to support the process change from the non-sterile to the sterile phase the marker head needs to be replaced with a sterile one. After the change of the marker head a high reassembly accuracy (repeatability) of under 0.5 mm is required to keep the overall reassembly error as low as possible. During the process change a sterile drape is placed between the marker head and the base. It is not avoidable that the position may change because of the thickness of the drape. The thickness of the foil needs to be considered and the new position needs to be determined.

Marker Visibility (R.MS.4): The marker system should be designed in a way that the attached markers are always visible in the field of view of the HoloLens during an intervention to ensure that a patient can be tracked at any time by detecting the markers in every frame. The main scenario of this thesis is the ventricular puncture, therefore the main focus of a surgeon is the upper head and a view from the Kocher's points to the target points inside the ventricular system. A marker system should cover this scenario, but the installation room must be chosen so that it does not disturb a surgeon during a procedure and affect the work in any way. If possible, the marker system should allow 360° tracking around the patient.

Low Weight (R.MS.5): The weight plays a subordinate role, since the patient is sedated by the anesthesia during the operation. Nevertheless, the weight should be as low as possible. Biocompatible materials and smooth surfaces should be chosen so that a patient is not injured in any way.

Sterilizability (R.MS.6): Basically there are two possibilities to allow the use of a sterile marker system during an intervention. First the marker system can be build in a way that it can be sterilized, or a new marker system is used for an intervention. For the second option, two marker heads would be needed to cover the process change from non-sterile to the sterile phase, because it could not be guaranteed that the first marker remains uncontaminated during the preparation process. If possible, the marker system should be designed in a way that it is sterilizable. All parts should be designed reusable to keep the costs low.

Modularity (R.MS.7): The marker system should be able to accommodate different types of markers. At the current state these are Aruco, Vuforia and infrared markers. Therefore, different marker heads shall be designed. This lays the basis for a detailed evaluation of the different markers and their implementation on the HoloLens.

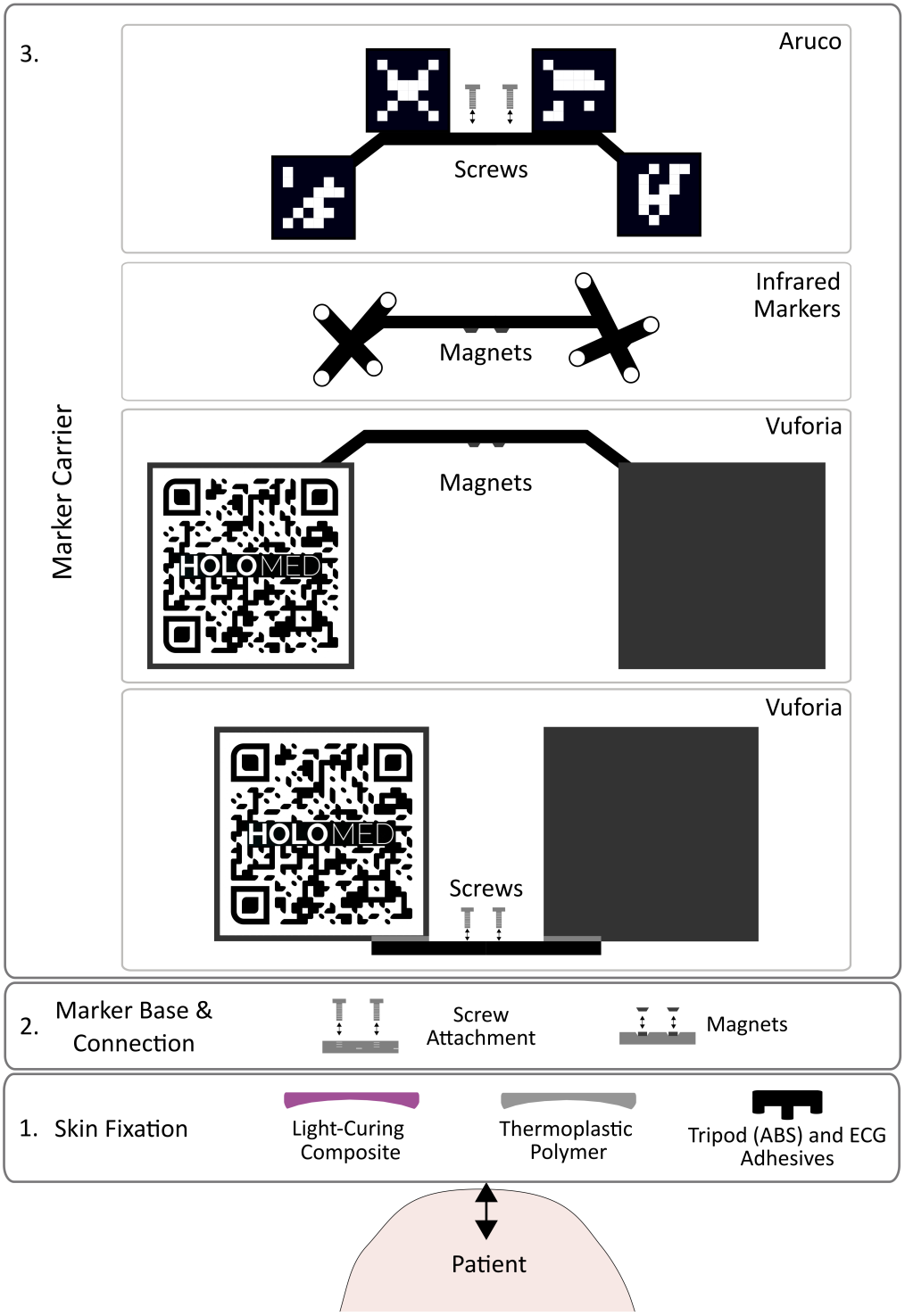


Figure 5.16.: Overview of the marker system and all possible configurations.

5.4.2. Methods

In Figure 5.16 the different configuration possibilities of the marker system are depicted. Basically three parts of the marker system can be interchanged: the skin fixation, the marker base to carrier connection and the marker carrier. Different options of these are detailed below.

5.4.2.1. Skin Fixation

The marker system is attached to the patient's forehead and must therefore be fixed to the skin to maintain a stable position. Three types of skin fixation options were investigated in this work: a thermoplastic polymer polycaprolactone, a light-curing composite and electrocardiogram (ECG) electrode pads on a printed tripod.

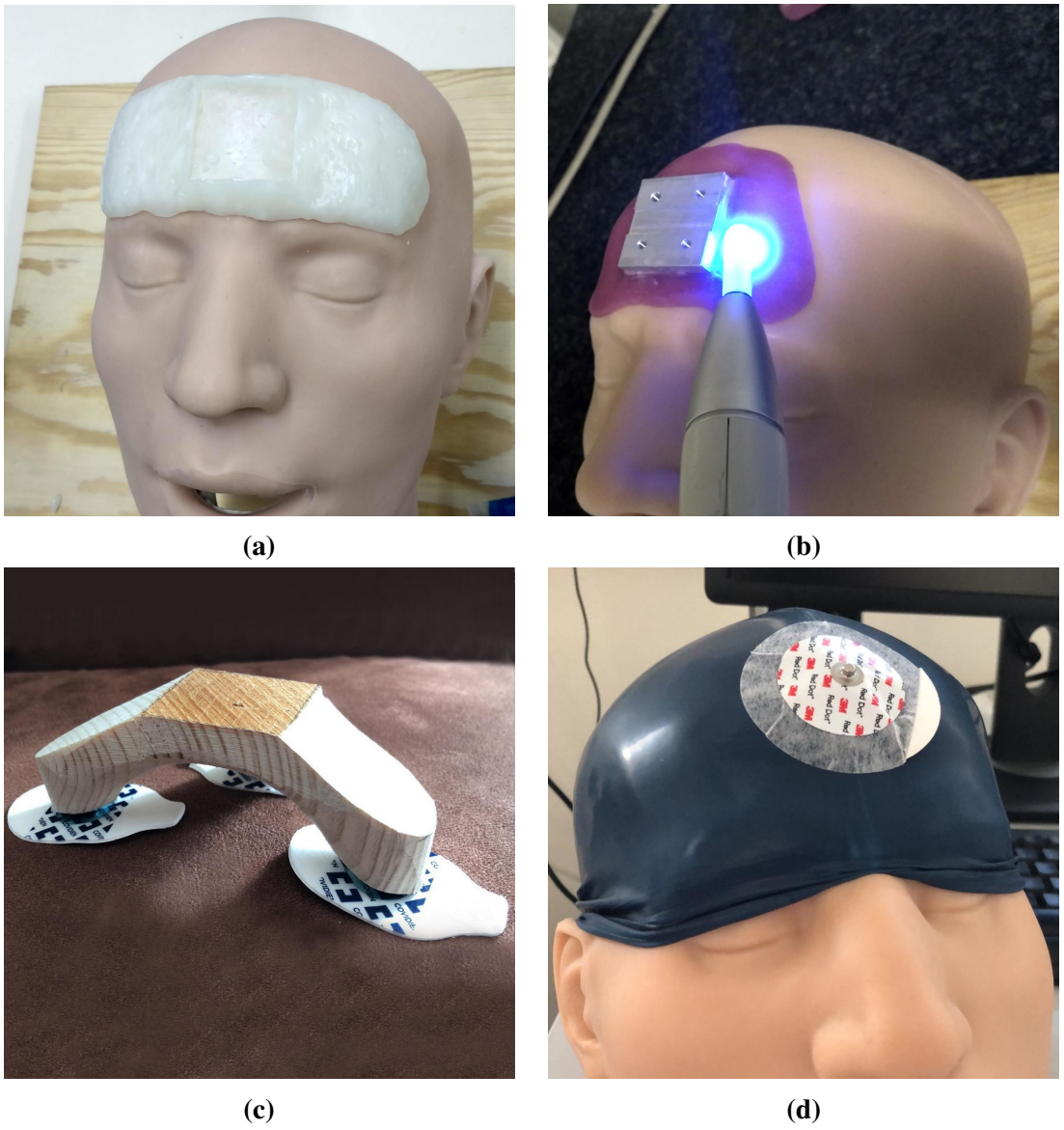
The thermoplastic polymer deforms at 42°C and hardens as it cools to room temperature. The used temperature is suitable to be used on human skin without causing any damage to it (see Figure 5.17 a). The light-curing composite is widely used in dentistry and is non-toxic. It hardens under ultraviolet (UV) light. In this work a dental UV polymerization light pen (Kent Dental Kentolight LED 800) was used, that emits light with 800 mw/cm with a wave length of 350 - 400 nm. Normally this pen is used for dental applications. Therefore, the effective light emitting area is quite small, which makes it necessary to move the pen over the skin fixation to cover the complete area. This increases the time required for hardening. The hardening process is depicted in Figure 5.17 b. The third way is to use ECG electrode pads which are normally used for electrocardiograms. They provide a standard of care way to attach objects to the human body. The electrode pads have a male snap (push button) on their top as depicted in Figure 5.17 c and d. Their counterpart, the female snaps, are glued to all three legs of the tripod. The ECG electrode pads are glued to the human head with the help of a template and the tripod is attached to it via the snaps.

The different skin fixation options are depicted in Figure 5.16-1.

5.4.2.2. Marker Base and Connection to Carrier

Two different options are available for the connection between the marker base and the carrier as shown in Figure 5.16-2. The base is chosen accordingly to the selected connection.

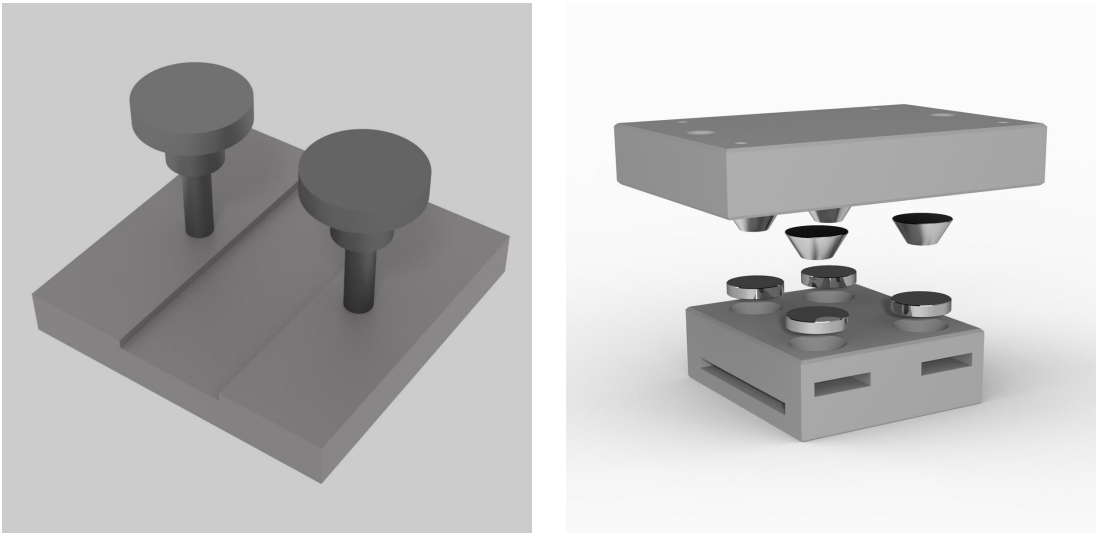
The first option is a screw attachment with knurled screws. A mounting plate with two screw threads is used as the base. Alternatively, a magnet attachment was developed in combination with a printed base that holds one side of the magnets, the other side of magnets must be attached to the marker carrier. See both options visualized in Figure 5.18. The base is pressed into the composite or polymer while still deformable to create an imprint. After hardening the base is removed and glued to the skin fixation and fastened with an elastic band to the patients head for a robust fit. When using the tripod, the base is screwed to it.



(a) **(b)**

(c) **(d)**

Figure 5.17.: a) Skin fixation with thermo polymer. b) Skin fixation with light-curing composite. c) Wooden prototype of the tripod with skin fixation with ECG electrode pads. d) Skin fixation with ECG pads on the Synbone phantom head.



(a) (b)

Figure 5.18.: a) Mounting plate that is used with the skin fixation options polymer and composite and screws. b) Magnet attachment.

5.4.2.3. Marker Carrier

For every possible marker technology a unique marker carrier was designed to optimally integrate them into the system. The general marker carrier design is similar. The carrier must be connected to the base over the chosen connection, via screws or magnets. When using the screw attachment the carrier has counterbore holes to accommodate the screws. The magnets are placed in recesses and glued to the carrier. The different marker carriers are depicted in Figure 5.16-3.

The Aruco and infrared marker carriers were designed in a way that the markers are also visible when standing left or right to the patient. Two different versions of the Vuforia marker carrier were tested. The first one has the markers on top, which can lead to a non-visibility in some recorded frames, but is not limiting the work space of the surgeon. In the second design the markers are placed beside the patients head. This leads to a better visibility but limits the workspace of the surgeon. A detailed evaluation is presented below in Section 5.4.3.

5.4.2.4. Sterilization Methods

High demands are placed on the hygienic status of medical products in order to keep the risk of infection as low as possible. The Federal Health Journal evaluates the hygiene requirements depending on the risk of infection for the patient and lists recommendations for prevention [1].

Accordingly, a distinction is made between three risk levels:

- Uncritical:
  Medical products coming into contact only with intact skin.
- Semi-critical:
  Medical products used with mucosa or pathologically changed skin.

- Critical:
  Medical products that are in direct contact with blood or internal tissues.

Sterilization is required in addition to cleaning and disinfection for the use of all products classified as semi-critical or critical. For a product to be considered sterile, a probability of finding a viable germ of 10^{-6} is defined in [3]. There are several standardized sterilization processes to meet this requirement, which are chosen depending on the application scenario and the nature of the product [79]:

- Heat sterilization procedure:
  Moist heat sterilization is the most commonly used process for reprocessing medical devices, for whose use and permissible process parameters there are European standards, which can be found in [2]. For dry heat processes higher temperatures and longer sterilization times are required and therefore they are used less often.
- Low temperature gas procedures:
  For thermally sensitive materials and components, sterilization with gases is the most common option. The main methods involve Ethylene Oxide (EO), Steam Formaldehyde or Gas Plasma. Ethylene oxide has the highest sterilization capacity and the best diffusion properties and is therefore often used for polymers.
- Sterilization with ionizing radiation:
  Ionizing radiation in the form of accelerated electrons or gamma rays is mainly used for the sterilization of disposables. Both processes incur high costs due to the high complexity of the devices and also require authorization.

5.4.3. Experimental Validation

The evaluation was performed using two configurations of the presented marker system. The first one uses the screw attachment and Vuforia markers, the second uses magnets and infrared marker targets. The marker systems were attached to the Synbone phantom head (Synbone, Switzerland) with an attached silicone skin as presented in Section 5.1.1. The two configurations are depicted in Figure 5.19.

The mounting error of the marker head after process change was determined with measurements on six reference points. Three reference points were located at the marker head and three on the phantom, respectively. The marker head was removed and reattached twenty times and measurements were taken with a high precision measuring arm (FARO platinum arm with FARO CAM2 software, FARO Technologies Inc., Germany). The FARO arm has an indicated precision of ± 0.037 mm.

The reference points on the marker system were numbered as M_i and on the patient as P_i ($i = 1, 2, 3$). The used coordinate system C_p was created over the points P_i on the phantom as it is important to use one fixed coordinate system for all measurements. The point P_1 is located on the forehead, P_2 on the cheek and P_3 on the chin, respectively. Reference points M_1, M_2 and M_3 are distributed over the marker head and marked with indentations to make it easier to place the spherical tip of the FARO arm. The position of points M_i are measured in the coordinate system C_p in all twenty experiments. To make sure that the coordinate system C_p has not changed in space the three points in P_i are measured as well. The mounting error is determined by measuring the position change of the points M_i in the phantom coordinate system.

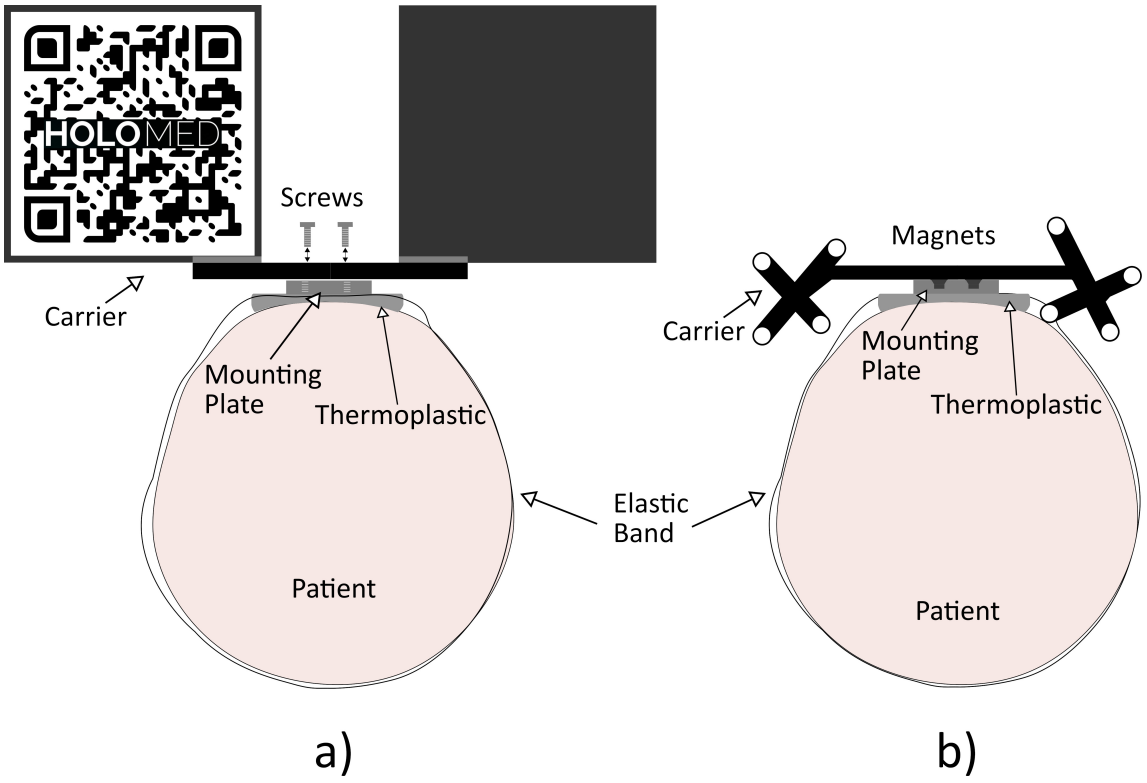


Figure 5.19.: The two evaluated configurations. a) Screw attachment and Vuforia marker. b) Magnet attachment and infrared markers.

Three different experimental setups (E_1, E_2, E_3) were evaluated for the first marker system configuration (screw attachment and Vuforia marker).

In the first experiment, E_1, a static setup was used and the marker head was removed and reattached twenty times and measurements were performed. This mimics the process change from the non-sterile to the sterile phase. In a second experiment, E_2, the phantom head was randomly moved in space after mounting it to a Stewart platform (Hexapod, PI GmbH, Germany). Within this setup the small movements of a sedated patient can be simulated. The last experiment, E_3, was used to simulate the behavior of the marker system under vibrations, like the ones that can occur during drilling of the burr hole trepanation. To do so a wooden plate was placed on the phantom head and a surgical drill was used to drill a hole into it. Thus, similar vibrations as those occurring during the burr hole trepanation when opening the skull to puncture the ventricular system were generated. The measurement uncertainty was obtained by performing twenty measurements with a fixed marker head.

The results indicate no change of the repeatability between experiments E_1, E_2 and E_3. Results are presented in detail in the next chapter. The second marker system configuration (magnets and infrared markers) is therefore only tested in experiment E_1. Additionally, the reattachment accuracy was evaluated with and without sterile drape between the marker base and the carrier. Ten measurements were performed on each measurement point.

Table 5.11.: Reattachment error on reference points M_1, M_2, M_3. E_1: Static phantom, no movement. E_2: Slight random motion. E_3: Drilling simulation. Configuration 1 was used as depicted in Figure 5.19-a. (All values in mm.)

	E_1	E_2	E_3	Mean(M_i)	SD(M_i)
M_1	0.17	0.16	0.16	0.16	0.06
M_2	0.21	0.18	0.16	0.19	0.06
M_3	0.23	0.16	0.17	0.19	0.06
Mean(E_i)	0.20	0.17	0.17	0.18	-
SD(E_i)	0.08	0.05	0.04	-	0.06

5.4.4. Results

The measurement uncertainty for all configurations was determined to be within 0.02 mm. For determination the tip of the FARO measurement arm was placed twenty times in a row at the same position. The measurement uncertainty is the maximum deviation over all measurements. The reason for this is that the measurement arm is manually manipulated in space and a human is not as accurate as a robot.

In the following the results for the first configuration (screw attachment and Vuforia marker) are presented. The results of all three experiments are summarized in Table 5.11. The carrier head could be reattached with an accuracy of 0.18±0.06 mm over all three experimental setups.

The results yielded nearly the same results for all three experimental setups. This indicates that small movements and vibrations do not severely change the position of the marker system.

The second configuration (Figure 5.19-b) was only evaluated with experimental setup E_1. It was additionally tested how the marker system behaved when a sterile drape was placed between the marker base and the carrier to simulate the coverage of the patient during the process change from the non-sterile to the sterile phase. A mean error of 0.018±0.01 mm was determined for the reattachment accuracy without sterile drape used. The mean errors and standard deviations for every single axis are as follows: x (mean: 0.005, std: 0.004), y-axis (mean: 0.008, std: 0.005), z-axis (mean: 0.014, std: 0.011). When using foil, the error increases to a value of 0.195±0.018 mm. The mean errors and standard deviations for every single axis are as follows: x (mean: 0.027, std: 0.016), y-axis (mean: 0.024, std: 0.012), z-axis (mean: 0.190, std: 0.018). In Figure 5.20 and 5.21 the detailed results over all three axes are illustrated.

It is noticeable that the main error is determined by the thickness of the sterile drape in z-direction. The sterile drape has a thickness of 0.15 mm. The thickness can be added in z-direction after the registration process to compensate for this error. By eliminating this error, the overall error when sterile drape is used drops to 0.054±0.019 mm and is thus in a similar range as during experiments without drape.

In the following, the requirements as summarized in Table 5.10 are checked for fulfillment and evaluated.

The preparation time of each work step is listed in Table 5.12. For configuration one (screw attachment and Vuforia marker), an assembly time of 380 s was determined and 346 s is required for configuration two. Most of the time is needed to prepare and harden

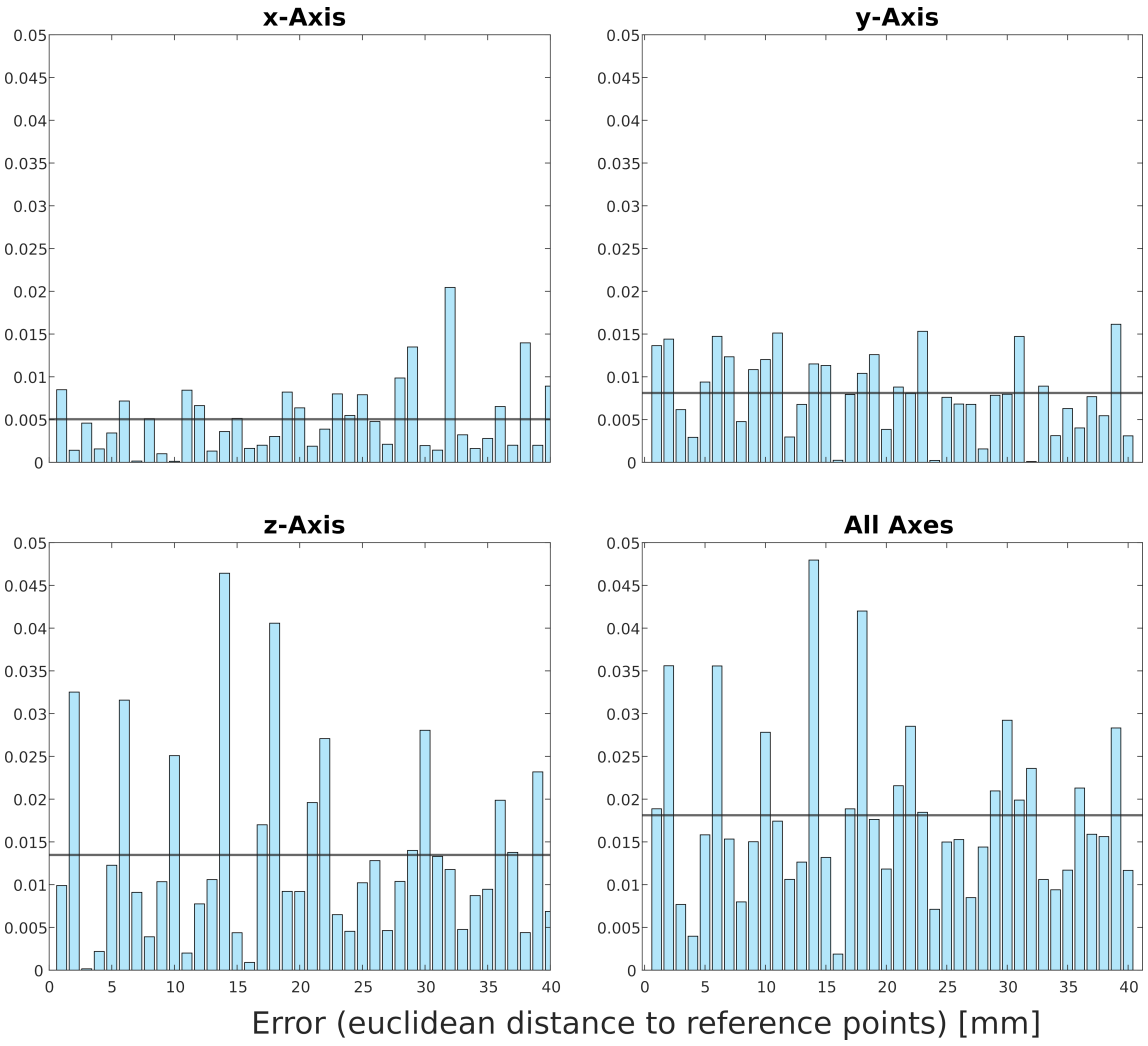


Figure 5.20.: Detailed evaluation results over all three axes without the use of sterile drape. The grey line describes the mean value.

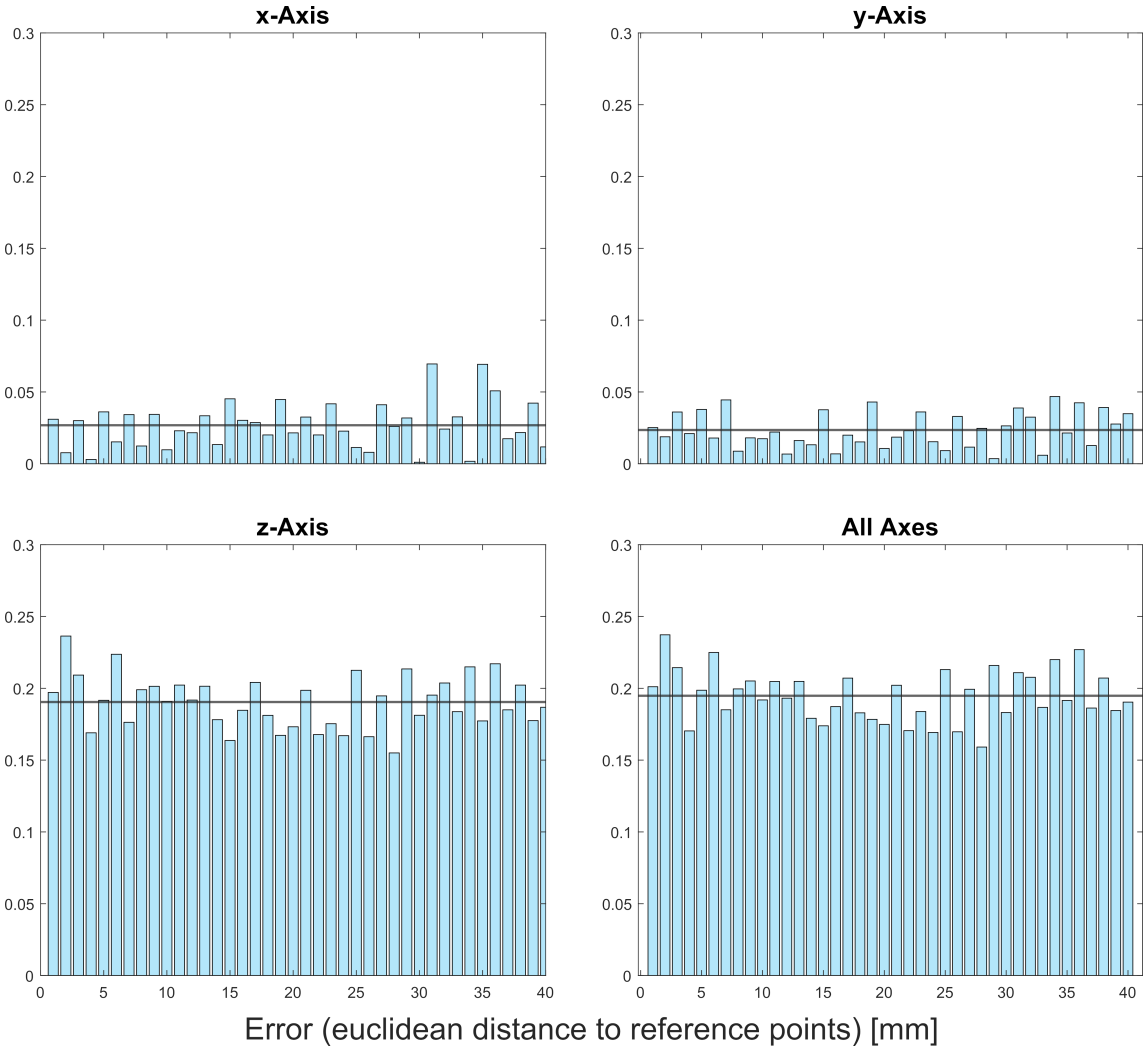


Figure 5.21.: Detailed evaluation results over all three axes with the use of sterile drape. The grey line describes the mean value.

Task	Time
Thermoplastic polymer	
Preparation of thermoplastic polymer	180 s
Press marker base into thermoplastic polymer	5 s
Hardening of thermoplastic polymer	120 s
Glue marker base into head fixation	15 s
Fix marker base to patient with elastic band	15 s
	335 s
Light-curing composite	
Preparation of light-curing composite	10 s
Press marker base into light-curing composite	5 s
Hardening of light-curing composite	300 s
Glue marker base into head fixation	15 s
Fix marker base to patient with elastic band	15 s
	345 s
ECG and tripod	
Glue marker base into head fixation	15 s
Attachment of the adhesive EKG strips to the tripod	10 s
Glue tripod to patient	15 s
Fix marker base to patient with elastic band	15 s
	55 s
Mount marker head with screw attachment	45 s
Mount marker head with magnet attachment	1 s

Table 5.12.: Time expenditure of each work step for mounting the marker system.

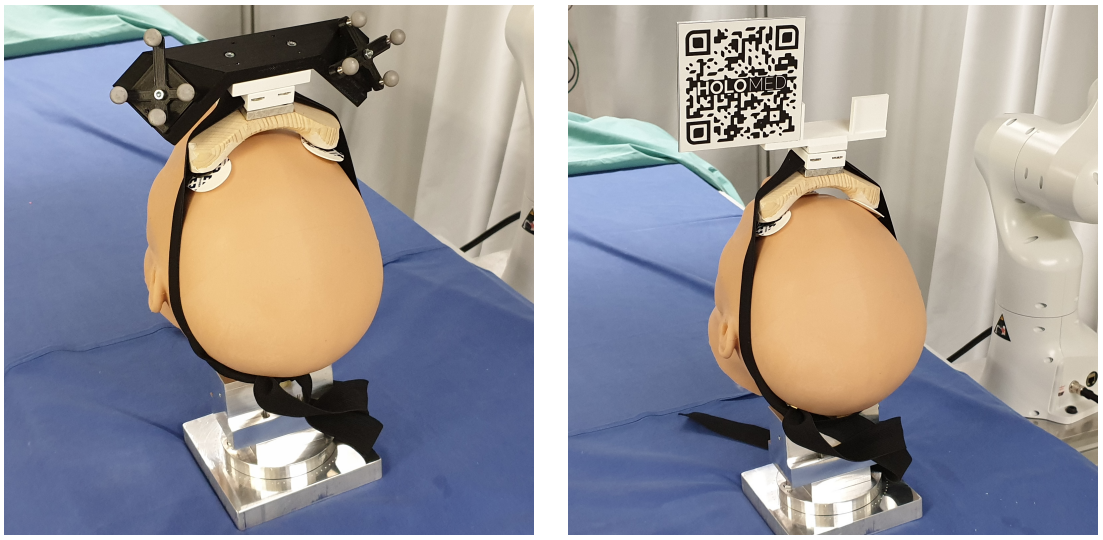
the skin fixation, which could be done prior to the intervention. Other materials like foam were also tested and first results seem promising. The preparation time of roughly 6 minutes is acceptable for planned interventions (R.MS.1).

To optimize the preparation time an alternative approach was examined that utilizes ECG electrodes to glue a tripod to the patient's forehead. The marker base is then attached to the upper side of the tripod with screws or glue and the marker system can be attached with both connection types (magnets and screws). This diminishes the issue of the time-consuming preparation and curing of the polymer and composite. The marker system mounting time can be reduced to 56 s when using the magnet attachment. Therefore, requirement R.MS.1 is fulfilled.

Two configurations using the tripod in combination with the magnet and screw attachment are depicted in Figure 5.22.

The marker system supports the process change (R.MS.2) from the non-sterile to the sterile phase. The sterile drape is thereby placed between the marker base and the connection to the marker carrier. As described above a sufficiently high reattachment accuracy is reached (R.MS.3).

Different marker carriers have been presented, that allow a placement of the markers slightly above the head. This can lead to the situation that the markers are not visible in every recorded frame, which makes it necessary for a surgeon to sometimes look up to recapture the markers with the AR glasses. An alternative design has been proposed



(a) **(b)**

Figure 5.22.: a) Marker system configuration with tripod, magnet attachment and infrared markers. b) Marker system configuration with tripod, screw attachment and Vuforia markers.

where the markers are placed beside the head of a patient, which results in a better overall coverage of the marker in the frames. However, this limits the workspace of a surgeon and makes it hard not to have contact with the marker, which may lead to a displacement of the marker system. In addition the coverage of a patient is not possible without greater changes of the sterile drape and therefore not feasible in daily clinical reality. For this reason, it was decided not to use this concept. The requirement R.MS.4 can nevertheless be evaluated as fulfilled, since the marker visibility is rated as good. The HoloLens uses the last known position of a marker when the tracking is interrupted, which guarantees a stable tracking even when the marker is not visible in every frame.

The marker system has a low weight (R.MS.5) of 210.9 g for the Aruco marker system, 193.4 g for the Vuforia marker system and 146.5 g for the infrared marker system. Currently, the marker system is not sterilizable as it is partly printed from ABS or ASA filament. The melting point of this material is with 100°C too low for the sterilization methods in Section 5.4.2.4. During process change a sterile marker carrier needs to be used. An alternative would be to manufacture the marker system completely from aluminium. This would be possible but would lead to a higher weight. It is also possible to print on aluminium to create sterilizable markers, which was already done in this work. Requirement R.MS.6 can therefore be seen as fulfillable. It was shown that the marker system is widely customizable and has a high modularity (R.MS.7) as it is capable of holding different types of markers. In general, the marker system can easily be adopted to any kind of marker.

5.4.5. Discussion

Tracking the patient is a crucial requirement when supporting a surgeon with augmented reality. Clinical reality scenarios can vary widely. A modular marker system design was therefore presented that is capable of holding different types of markers. The marker

system is basically divided into five parts: the skin fixation, the marker base, the marker carrier, the attachment that connects both with each other and the tracked marker. Several options were provided for each part. The marker system can be attached at any position of the human head as long as the marker has a good visibility to the HoloLens. The registration from the marker system to the patient is performed after the attachment. After that the marker needs to stay at the registered position.

The skin fixation that can be attached fastest uses ECG electrode pads in combination with a tripod base structure. It guarantees a stable connection to the patient and allows for fast and safe attachment. However, the polymer and the composite are preferable when time is no issue, as they prevent almost any movement of the marker base. Attachment of the ECG electrode pads is not simple as the three pads need to be secured on the correct locations on the forehead of the patient. This can become challenging due to shape variations of patient's foreheads. A simple template sheet with holes that mark the position of the pads can be used to mark the correct position on the forehead of the patient. An interesting option could be to create a fitting template sheet for every individual patient with the usage of statistical shape models, as presented earlier in Section 4.2. Generally, due to the design of the tripod it is easy to attach it to any human forehead.

All requirements were fulfilled. The marker system can be assembled fast, has a low weight, is sterilizable and modular in its design. It supports the process change from the non-sterile to the sterile surgical phase. Two connection options were presented, which both yield a good reattachment accuracy of under 0.2 mm. However, the magnet attachment has the advantage of a much faster attachment time and a slightly higher reattachment accuracy. The presented designs yield a good marker visibility.

5.5. Marker to Patient Registration

In the previous sections trackable markers were presented and a modular marker system that can hold these markers. Several possibilities have been shown to mount the marker system to the patients head and support the transition from the non-sterile to the sterile phase by reattaching a sterile marker carrier with sufficiently high accuracy of under 0.2 mm.

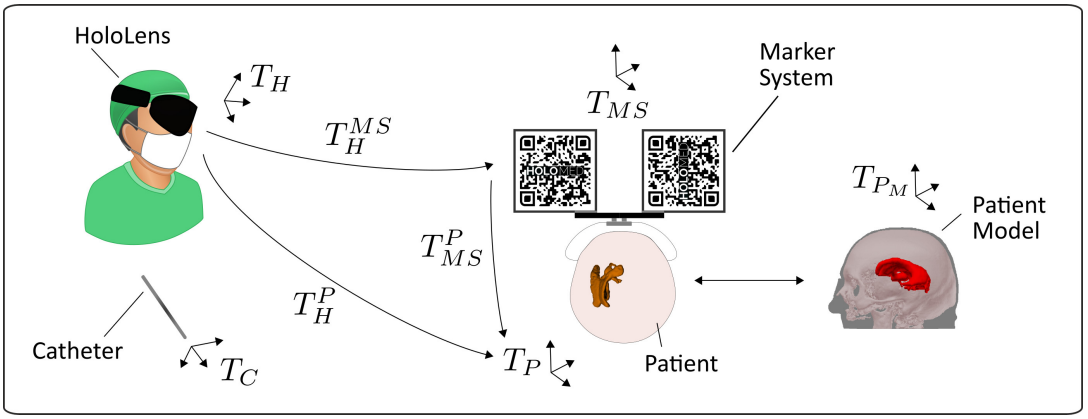


Figure 5.23.: Overview of the coordinate systems of the surgical scene.

The position of the marker system can be freely chosen, which makes it necessary to perform a registration from the marker system to the patient. Every involved entity in our experimental setup is described through its own coordinate system, this can be the surgeon, medical staff, one or more patients, surgical instruments, sensors, robots and so on. Sometimes several coordinate systems are grouped together by registering them to each other. This is the case for the HoloLens, where all sensors (see Table 5.8) represent their own coordinate system and are registered to the HoloLens coordinate system that is created at the spatial starting position in the room. For example the *short throw depth* streams represent coordinates in its own coordinate system and need to be registered to the HoloLens coordinate system with a transformation matrix. All coordinate systems of the investigated surgical scene are visualized in Figure 5.23.

It is assumed that the surgeon who wears the HoloLens T_H is present in the same coordinate system, which means that the eyes of a wearer see everything in the HoloLens perspective. It is necessary to perform a calibration to the user as described in Section 6.1.1.3. The patient model coordinate system T_{P_m} needs to be aligned with the patient coordinate system T_P. The coordinate system of the marker system T_{MS} has its origin in one of the used markers. If more than one marker is used these are registered to the main marker. Additionally, the catheter is represented with T_C. The transformation matrices represent the registration of the different coordinate systems to each other. T_H^{MS} is the transformation matrix from the HoloLens coordinate system to the marker system and T_{MS}^P defines the transformation from the marker system to the patient.

To enable the tracking of a patient during an intervention the position of the patient defined through the marker system needs to be constantly derived and updated. This is described through the transformation matrix T_H^P. The transformation to the marker system T_{MS} is determined simply by tracking the individual marker. To superimpose the patient models at the correct position the transformation T_{MS}^P needs to be determined. The patient coordinate system can be spanned from every point on the patient. In our case it is defined

by the patient models coordinate system that needs to be aligned with the patient. This is done in the non-sterile surgical phase while the patient's head is not covered.

In the following, four methods are presented to perform the registration from the marker system to the patient. Three manual methods and one automated one are examined and evaluated against each other.

5.5.1. Methods for Registration of the Marker System to the Patient

The registration is performed by placing the patient models over the real patient to derive the transformation $T_H^{P_{init}}$. Simultaneously, the marker system is tracked to determine T_H^{MS}. The transformation from the marker system to the patient is given by

$$T_{MS}^{P} = (T_H^{MS})^{-1} \cdot T_H^{P_{init}} \tag{5.3}$$

As shown in the previous section a sterile marker system can be reattached with very high accuracy. It remains stable even during small movements and vibrations. The marker system was designed in a way that it can be attached to any suitable position on the head of a patient. The position of the patient is then constantly derived as in

$$T_H^P = T_H^{MS} \cdot T_{MS}^P \tag{5.4}$$

The patient's head is completely covered during an intervention, except for the surgical site. Only the marker system is visible. The placement of the patient models over the real patient can be done with four different methods, that will be described below. The patient must not be moved during this process.

The generalized workflow is depicted in Figure 5.24.

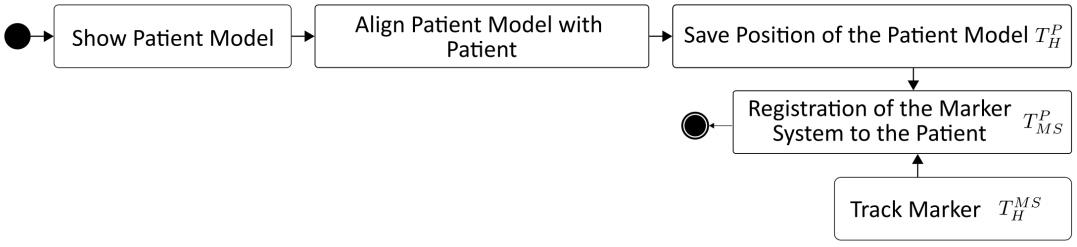


Figure 5.24.: Overview of the general workflow to perform the registration from the marker system to the patient.

5.5.1.1. Registration Using a Game Controller

The game controller-based registration is performed with an Microsoft XBox One Controller. The process is visualized in Figure 5.25. The attached marker system with unknown registration T_{MS}^P is tracked within the first recorded frame to perform a rough alignment of the patient models. It is roughly known where the marker system will be

attached to the patients forehead within a range of a few centimeters. The patient model can then be moved in all six degrees of freedom with the game controller. One axis at a time can be selected to change the translation or rotation. It was decided to perform one action at a time with only one joystick of the controller to keep the interaction simple and robust.

When the models are aligned with the real patient or phantom the registration is calculated using Equation 5.3, by tracking the used marker and taking the position of the placed models.

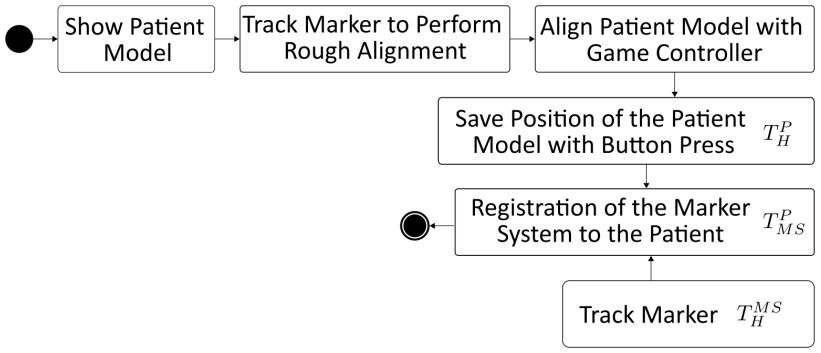
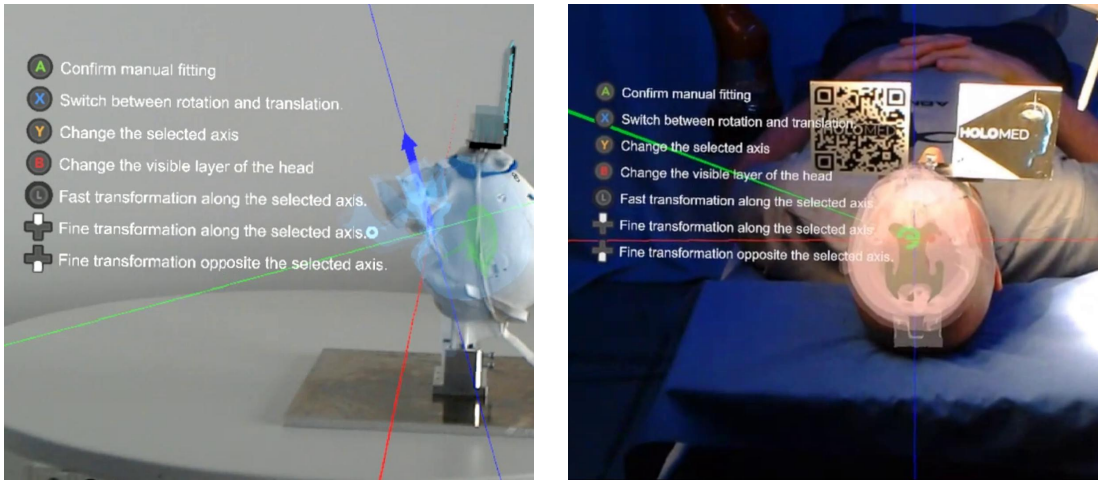


Figure 5.25.: Workflow to perform the registration with a XBox game controller.

In Figure 5.26 a and b the registration process from HoloLens view is depicted.



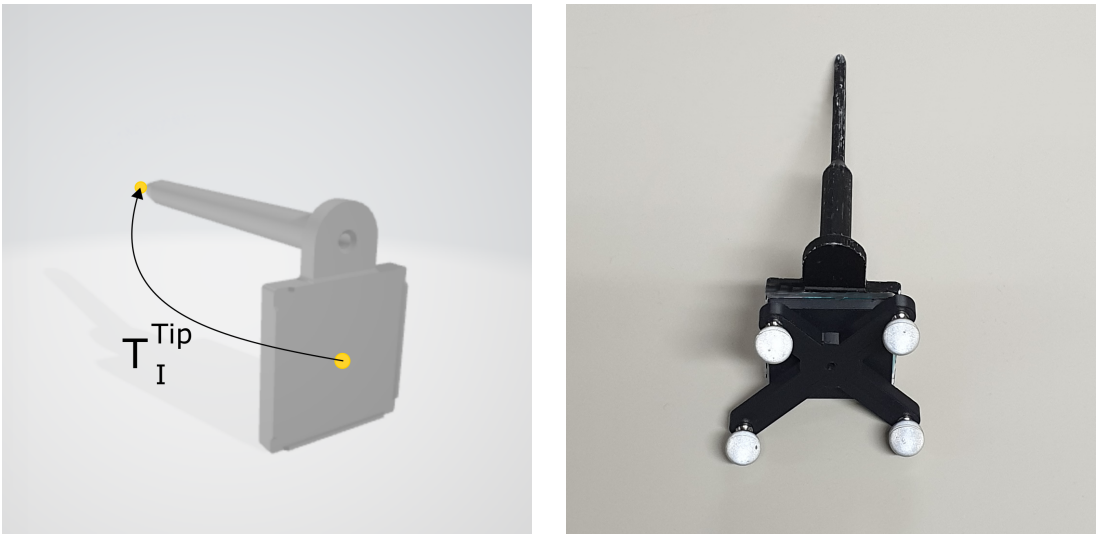
(a) **(b)**

Figure 5.26.: a) HoloLens view of the registration process on the OpenHead phantom. b) HoloLens view of the game controller-based registration process on a real person (non-clinical real world example).

5.5.1.2. Registration with Pointing Device

The second option to perform the registration of the patient model is utilizing a pointing device that is visualized in Figure 5.27 a and b. The pointing device is tracked with an attached infrared marker target. The tip of the pointing device is determined through

a known transformation matrix that is extracted from the CAD models. Figure 5.27 a shows the transformation from the mount position of the marker target to the tip. This corresponds to the source of the marker target. Depending on the marker target, additional transformations could be necessary, e.g. from one infrared sphere to the marker target source. A printed pointing device with attached marker target is depicted in Figure 5.27 b.



(a) **(b)**

Figure 5.27.: a) Transformation from the mount position of the marker target to the tip of the pointing device. b) Printed version of the pointing device with attached infrared marker target.

The workflow for registration with the pointing device is depicted in Figure 5.28.

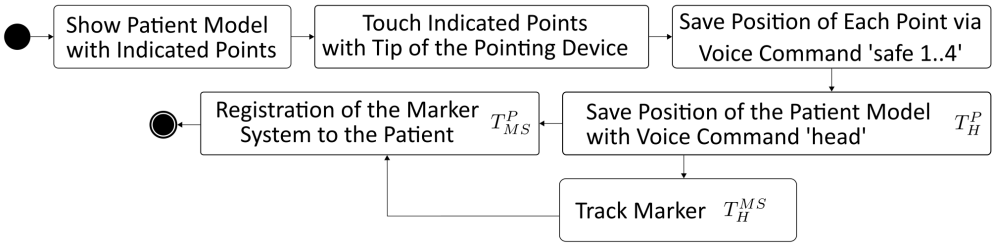
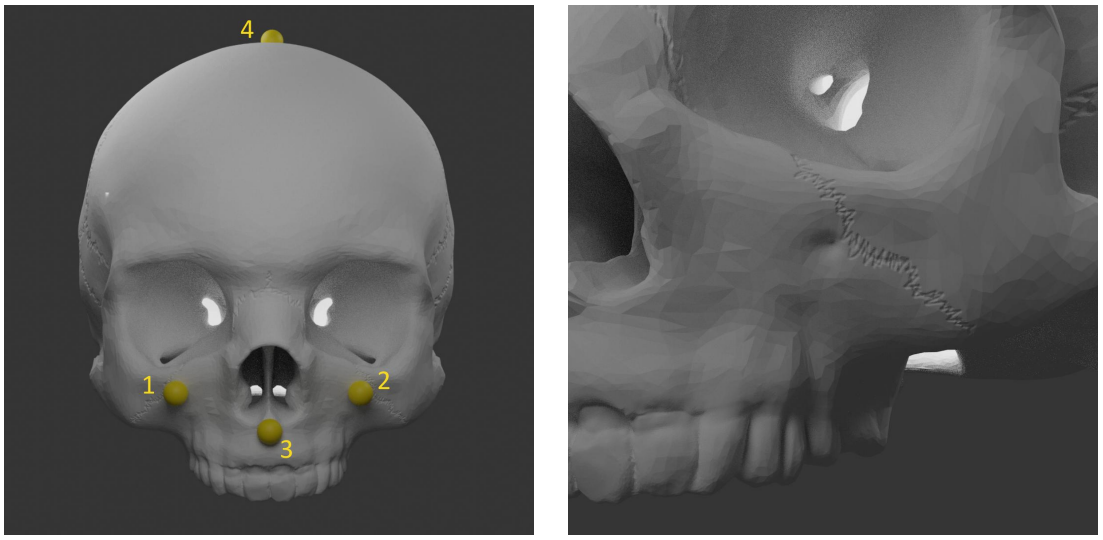


Figure 5.28.: Workflow of the registration with the pointing device.

In the registration step, the patient models with indicated registration points are visualized to the user as visualized in Figure 5.29. The user must then place the tip of the pointing device on the displayed points.

In our experimental setup this was performed on the OpenHead on recesses in the phantom. In a real world clinical environment this could be done with fiducials attached to the skull of a patient prior to the CT scan. The four indicated points must be touched and their positions are saved to the system. Since the surgeon does not have a free hand, they can use the voice command 'save' and the number of the point to save it. While all points are recorded, the patient models are moved according to the saved positions. At any time the registration can be visually checked. When the registration is satisfying, the position of



(a) (b)

Figure 5.29.: a) Indicated capture points on the OpenHead phantom numbered from 1 to 4 that are shown to the user. b) Detailed view of the OpenHead phantom with recess to which the tip can be pointed.

the patient model can be saved and the registration step is performed. The voice command 'head' is used to do so.

5.5.1.3. Gesture-based Registration

In the third method, patient models are placed with gestures over the real patient. The HoloLens is able to detect several gestures that can be processed and handled with the MRTK toolkit [149]. The gestures used in this work are the 'click' and 'grab' gesture shown in Figure 5.30 and Figure 5.31. Furthermore, the movement of the head and voice commands are used. The initial coarse alignment can be done by detecting the marker in the first frame. Alternatively, it is possible to use the 'click' gesture to perform the coarse alignment of the marker system to the patient. To do so the surgeon is positioned at the top of the operating table and performs the click near the top of the patient's head. The pose of the patient is roughly known as the patient is always lying on their back. The 'grab' gesture can then be used to grab the patient model and perform a finer translational alignment. To make the rotational fine adjustment an axis is selected using voice commands. Alternatively it would still be possible to use the game controller, however this registration method was designed to only use intuitive interaction. The rotation is then changed by moving the head to the left or the right side as shown in Figure 5.32.

The workflow is depicted in Figure 5.33.

5.5.1.4. Patient Model Alignment via Surface Matching

The three previous presented methods for the patient model alignment must be performed manually. However, it would be preferable to perform this step fully automatic in the future. Therefore, an automated process using a surface matching approach was implemented that is depicted in Figure 5.34.

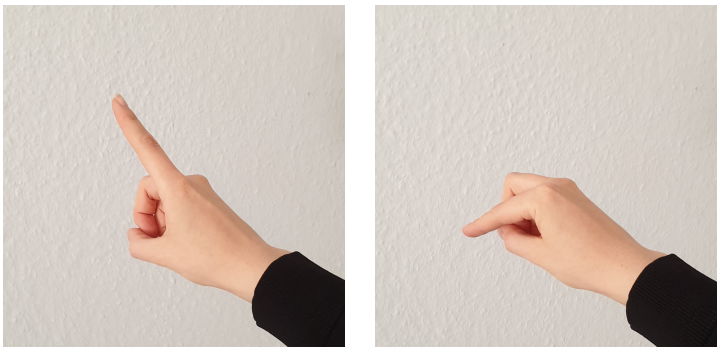


Figure 5.30.: Click gesture.

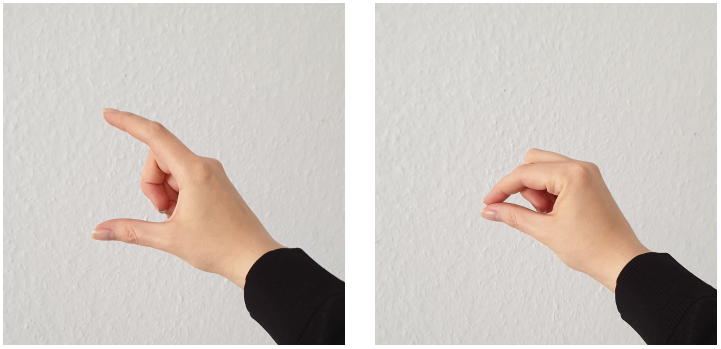


Figure 5.31.: Grab gesture.

Figure 5.32.: Head movement to the left and right side with HoloLens.

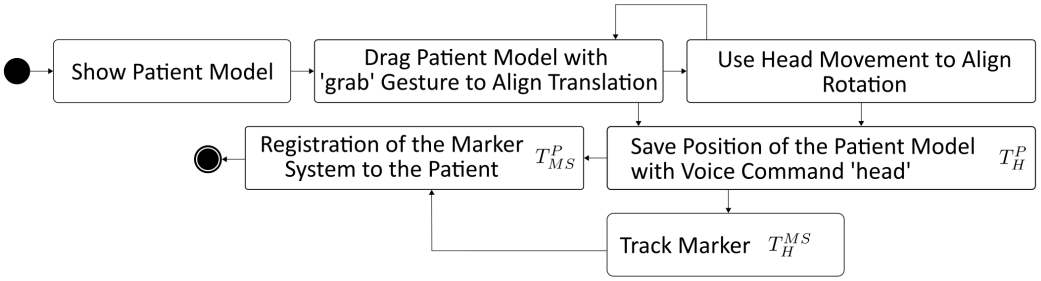


Figure 5.33.: Workflow of the gesture-based registration.

The approach is based on matching two point clouds, each forming a surface. The first point cloud is generated from the previously acquired patient models out of the patients CT data. The second point cloud is recorded from the sensor streams of the HoloLens in

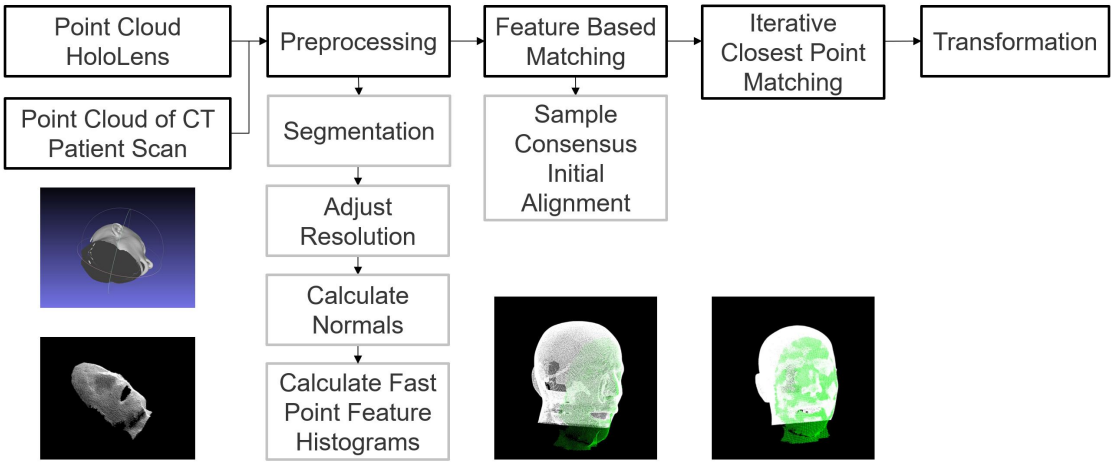


Figure 5.34.: Workflow of the registration using surface matching by utilizing depth cloud fusion.

the preoperative stage during registration. Within the experimental setup, the OpenHead phantom is used without the attached silicone skin.

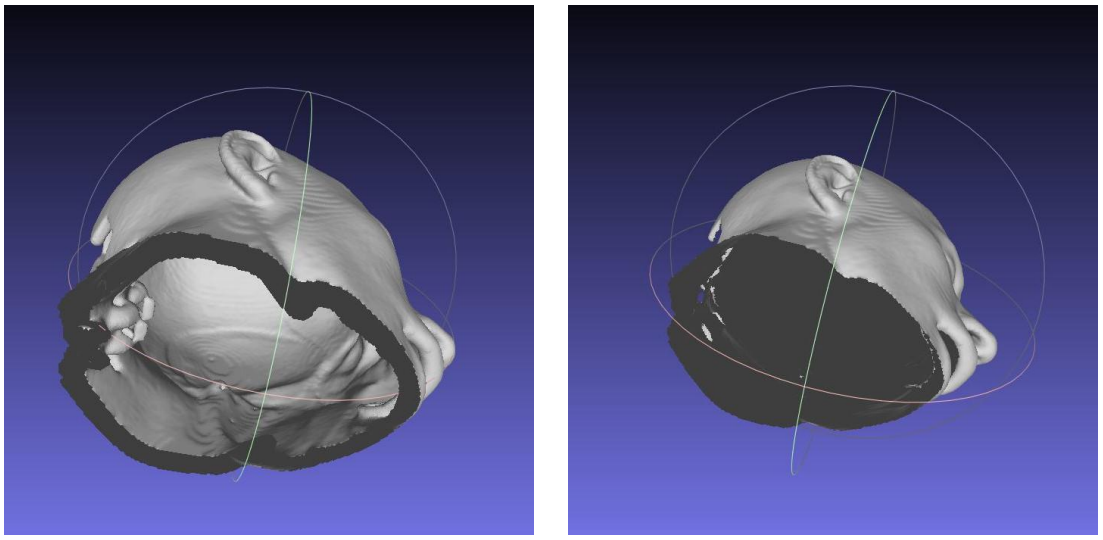
Point Cloud Acquisition: In Section 4.1, the segmentation of the patient models is described. The output of the segmentation are generated volumetric (*nrrd*) and surface (*stl*) models of the skull, skin and ventricular system of the patient. To match the surface of the patients head both point clouds need to be in the same file format, which in our case is the *polygon file format (ply)*. Therefore, the surface model of the patient's skull is converted to point cloud P_S in the *ply* file format. In this simple conversion every vertex of the surface model represents a point in the point cloud.

Normally the patient's skull and skin consist of an inner and outer layer due to a certain thickness of the tissue (Figure 5.37 a). The inner layer is removed manually in the data as visualized in Figure 5.37.

The point cloud P_S is aligned with the point cloud P_H, which is recorded from the HoloLens *short throw depth* and *short throw reflectivity* streams. The *short throw reflectivity* values are used to colorize the point cloud.

Preprocessing - Segmentation: If the entire acquired frame is used, it is often not possible to determine the correct position of the patient's head, because the surface matching approach runs into local minima. Therefore, the point cloud must be segmented so that only the patient's head remains. To do so, the OpenHead phantom is placed on a dark sheet in the segmentation step so that it can easily be determined in the *short throw reflectivity* frame using a threshold. An example of a recorded frame is depicted in Figure 5.36.

An alternative approach is the placement of a virtual box that covers the space of the patient's head. This is easily possible with the MRTK toolkit [149] and the HoloLens by placing a three-dimensional box in the scene that can be scaled and moved to the position of the patient's head. The box is placed in the HoloLens coordinate system and the point cloud P_H is present in the same coordinate space. All points that do not lie within this box are filtered out. A similar approach without augmented reality support was presented by Martin et al. [145].



(a) **(b)**

Figure 5.35.: a) Patient model with inner and outer layer. b) Patient model after inner layer was removed.

Figure 5.36.: Recorded point cloud of the short throw depth stream.

A further solution is to use a depth threshold to filter every point that is within a certain distance of the HoloLens. The wearer needs to focus on the head of the patient and the area of recording is displayed. A trough pass filter defined in Equation 5.5 is used to filter all values outside a certain range between 20 and 60 cm.

$$f : \mathbb{N} \to \mathbb{N}, T(p_{model}) = \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} = \begin{cases} \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} & -0,6 < p_z < -0.2 \\ \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} & else \end{cases} \tag{5.5}$$

Preprocessing - Adjust resolution: In order to shorten the computation time of the surface matching, an adjustment of the resolution is carried out. For this purpose, a downsampling is executed utilizing a three-dimensional voxel grid filter. The filter creates small three-dimensional voxels (i.e. boxes) in space with a certain side length over the point cloud. All points inside a voxel are approximated with their centroid, as in

$$c_v = \frac{1}{k} \sum_{i=0}^{k} (x_i, y_i, z_i) \tag{5.6}$$

Only the centroids in the respective voxels are used for the further calculation. An implementation in the Point Cloud Library [183] of the voxel grid filter is used [184].

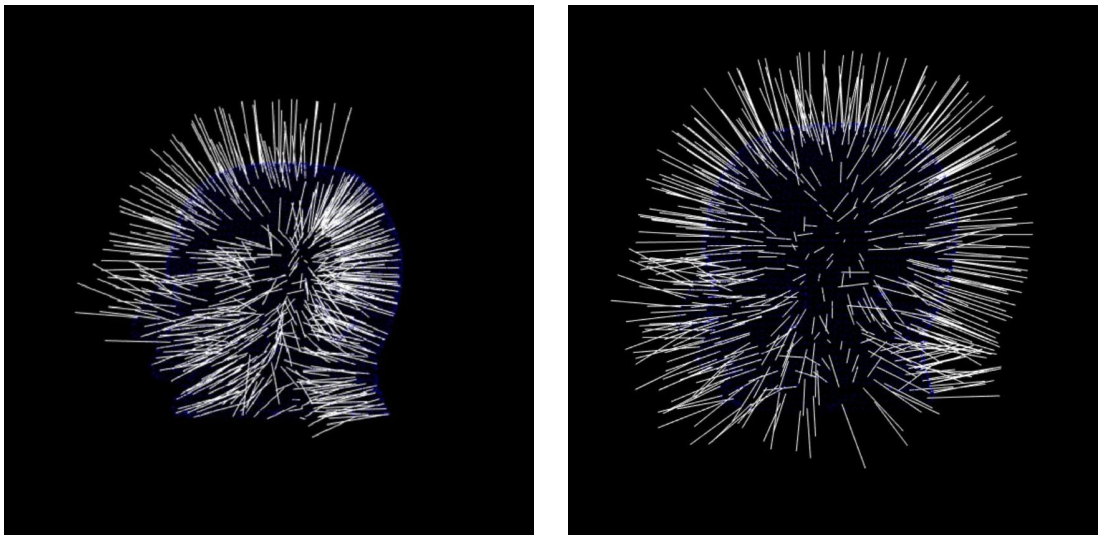
Preprocessing - Calculate Normals: Since knowledge about the normals of the point cloud is required for the surface matching procedure utilizing the *Sample Consensus Initial Alignment* algorithm, these are calculated beforehand using the PCL function *NormalEstimation*.

To determine the normal vector n_i of a point p_i of the point cloud, again a set K of points in a neighborhood N_r with radius r is selected and their centroid is calculated as in Equation 5.6. Points in a radius of 2.5 cm are chosen. Then the covariance matrix is calculated following Equation 5.7.

$$\begin{aligned} C = \frac{1}{k} \cdot \sum_{i=0}^{k} w_i \cdot (n_i - c_v) \cdot (n_i - c_v)^T \\ C \cdot \vec{v_j} = \lambda_j \cdot \vec{v_j} \\ j \in 0, 1, 2 \end{aligned} \tag{5.7}$$

Where $n_i \in N_r$ are all points in the neighborhood with radius r of a point p_i. The weight w_i represents a possible weight for n_i and normally equals 1. For a more detailed description see [182]. λ_i are the eigenvalues of the covariance matrix and $\vec{v_j}$ are the eigenvectors. The eigenvector corresponding to the smallest eigenvalue is an approximation of the normal n_i of point p_i.

Additionally, it must be checked if the normals are oriented correctly. This is done by setting a suitable view point. In our case it is suitable to take (0,1,0), which is the origin



(a) **(b)**

Figure 5.37.: The figures show the patient model converted to the *ply* format. a) Original orientation of the normals after conversion. b) Normals after orientation to the center of the model and the flip to the outside.

of the HoloLens coordinate system and an added meter above the ground. Thereby it is guaranteed that the normals are always flipped to the outside of the patient's head.

The normals of the point cloud that is generated from the patient scan must also be oriented correctly, because the normals are often not correctly aligned as depicted in Figure 5.37 a). To do so, the viewpoint is set to be directly in the middle of the patient model. All normals are then flipped inwards in the direction of the center of the head. In a second step, all normals are simply turned outwards. The results are depicted in Figure 5.37 b).

Preprocessing - Calculate Fast Point Feature Histograms: The feature-based matching to perform the initial alignment relies on fast point feature histograms (FPFH) that are calculated from the normals that were determined in the previous step utilizing the PCL function *FPFHEstimation*.

The approach determines a histogram for every point p_i in both point clouds P_S and P_H by calculating point features that represent geometric features taking the k-neighborhood (k neighboring points) of a point into account. For each pair of points in the k-neighborhood the difference is calculated by determining a quadruplet $(\alpha, \phi, \theta, d)$. All k quadruplets are combined into one histogram per point. A detailed description is provided in [181].

Feature-based Matching for Initial Alignment: To perform the feature-based matching the *Sample Consensus Initial Alignment* algorithm is used. A certain number of points in the point cloud to be registered are selected. For each of these points, the respective FPFH is compared with the point histograms of the target point cloud. A transformation matrix is calculated from the correspondences and executed. After the transformation, an error metric is calculated over all points by using the Huber penalty measure, as in

$$L_h(e_i) = \begin{cases} \frac{1}{2}e_i^2 & \|e_i\| \leq t_e \\ \frac{1}{2}t_e(2\|e_i\| - t_e) & \|e_i\| > t_e \end{cases} \tag{5.8}$$

where e_i is the error value and t_i a distance threshold of both point clouds [156]. This step is repeated and the transformation with the smallest error is saved [181]. In a last step, a non-linear local optimization is applied utilizing the Levenberg-Marquardt algorithm on the transformation [65].

Iterative Closest Point Matching: In our pipeline we use two matching approaches, the *Sample Consensus Initial Alignment* (SCIA) and the Iterative Closest Point (ICP) algorithm. The reason for this is that the first algorithm finds even in large scenes the position of the patient head, but the fine alignment sometimes fails and it can result in an offset of a few centimeters. The ICP algorithm is not suitable for the fitting of small objects into a large point cloud as it runs into local minima very often and gets stuck. It needs a good initial alignment. The combination of both a feature-based approach and ICP solves the issues both algorithms have. For a coarse alignment SCIA and for fine alignment ICP is used to obtain the best alignment results. ICP was first introduced in 1992 by [37] and [23]. Since then, different types of ICP have been proposed. A detailed survey about different ICP implementations are given in [180].

The algorithm ICP works iteratively. For each point in a point cloud, the nearest neighbor in the second point cloud is determined. In each iteration of the algorithm, the distance between the points is reduced by a change of the transformation parameters. After every iteration, an error value is calculated to determine the quality of the matching by using the least squares method. This procedure minimizes the euclidean distance between the two point clouds. In this work the standard ICP method of the PCL library is used.

Transformation: The matching result is stored in the transformation matrix T_H^P. The transformation describes the translation and rotation applied to each point of the source point cloud to match the target point cloud. For matching, the model's point cloud P_S was chosen as the origin cloud because the feature-based algorithm selects random points from the point cloud to be matched. If the HoloLens point cloud was chosen as the origin cloud instead, it could be that the algorithm uses irrelevant points for matching.

A matching procedure is visualized in Figure 5.41.

5.5.2. Evaluation of the Registration Methods

In the following a detailed evaluation of the presented registration methods is carried out. The four registration methods are evaluated and compared. The registration methods using a controller and gestures are evaluated with the same method. The evaluations are carried out on the OpenHead phantom skull with mounted Vuforia marker. The real phantom is fixated on a desk so it can not move. To measure the deviation on all axes a digital caliper is used to determine the misalignment of the digital model to the real OpenHead skull. Ten measurements are performed per axis to determine the error separately. The virtual skull is registered with the controller or gestures. The misalignment is measured using the digital caliper by estimating the error on the defined recesses of the OpenHead on all three axes.

To determine the registration error using the pointing device method, measurements are taken at all four recesses of the OpenHead. The pointing device is thereby tracked with an infrared marker target. The measured length between the points is compared to the known real length.

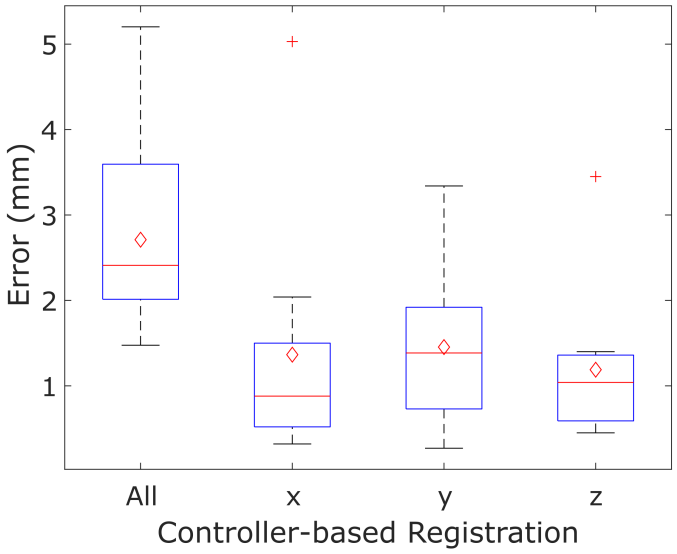


Figure 5.38.: Results of the controller-based registration. Mean values are visualized as red diamond markers, outliers as red cross markers.

The automated surface matching method is evaluated offline by determining the registration error between both point clouds after matching.

5.5.2.1. Accuracy of the Registration with the Controller

The controller-based registration is performed using an XBox game controller. After aligning the virtual skull model with the real OpenHead, ten measurements are performed for each axis. It is assumed that the rotational error is neglectable due to the fact that the OpenHead is placed vertically to the desk in a holding contraption. The base and the boundaries of the OpenHead are used to estimate the misalignment with the digital caliper. For the x-axis this was done using the base connector of the OpenHead. The y-axis error is determined by measuring at the upper part of the skull. The z-error is measured on the back of the skull by looking from the side onto the OpenHead.

The experiments were conducted by two persons who made ten registrations each. The results are depicted in Figure 5.38.

A mean error over all axes of 2.71$\pm$1.18 mm was observed: 1.36$\pm$1.4 mm for the x-axis, 1.45$\pm$0.92mm for the y-axis, and 1.19$\pm$0.87 mm for the z-axis. The mean error over all three axes is larger than for every individual axis, because it is determined over all three dimensions.

5.5.2.2. Accuracy of the Gesture-based Registration

The gesture-based registration is evaluated in the same way as the controller-based registration. The virtual patient model is placed over the real OpenHead phantom by using gestures. The experiments were carried out by two people who each conducted ten registrations. Again, a digital caliper is used to measure the deviation. The results are depicted in Figure 5.39.

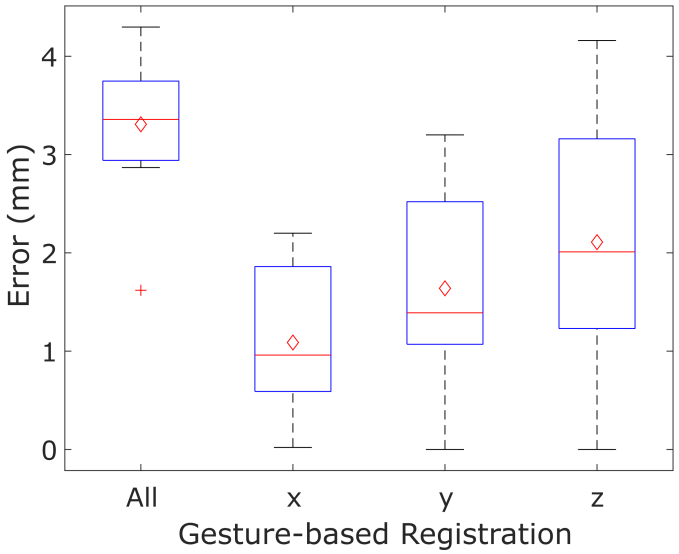


Figure 5.39.: Results of the gesture-based registration: Alignment error over all three axes and for all individual axis are depicted. Mean values are visualized as red diamond markers, outliers as red cross markers.

A registration error of 1.09±0.78 mm was determined for the x axis. 1.64±1.10 mm for the y axis and 2.11±1.29 mm for the z axis. This leads to an overall error of 3.31±0.75 mm.

5.5.2.3. Accuracy of the Registration with the Pointing Device

The pointing device registration is evaluated differently compared to the previous ones. Measuring the misalignment with a digital caliper is error-prone due to the fact that a human must estimate the displacement. A better approach is to conduct a digital measurement and compare it to a known ground truth. A digital measure of the hologram displacement is difficult, as it is not possible to record the virtual world through the lenses of the HoloLens, because it is hard to mimic the behavior and human perception this way. To get better results, measurements are taken on the four points primarily defined on the OpenHead.

The pointing device is used to determine the position of the patient model in space by placing the pointing device on the four recesses as described in Section 5.5.1.2. The distance between these points is known, which provides an accurate ground truth. Although we do not know the absolute position error of the measured points in space, we do know the distance between them and can compare them to the measured value. The distances between the measured positions of the points P1, P2, P3 and P4 are calculated and compared to the known value. To determine the registration error, the virtual patient model of the skull are placed over the OpenHead phantom ten times using the registration method. After saving every point, the position is stored and the distances between the points are determined. The measured distances are displayed in Table 5.13.

The results of the evaluation are depicted in Table 5.13 and in Figure 5.40.

The mean error over all axes is 3.71±3.25 mm.

Points	Distance	Axis	Error & Standard deviation
P1 to P2	65.97 mm	x	5.26±3.94 mm
P1 to P3	37.68 mm	yx	3.02±3.04 mm
P2 to P3	37.35 mm	yx	1.96±1.65 mm
P3 to P4	92.15 mm	yz	4.58±2.76 mm

Table 5.13.: Measurement points with their distances on the OpenHead and results of the evaluation.

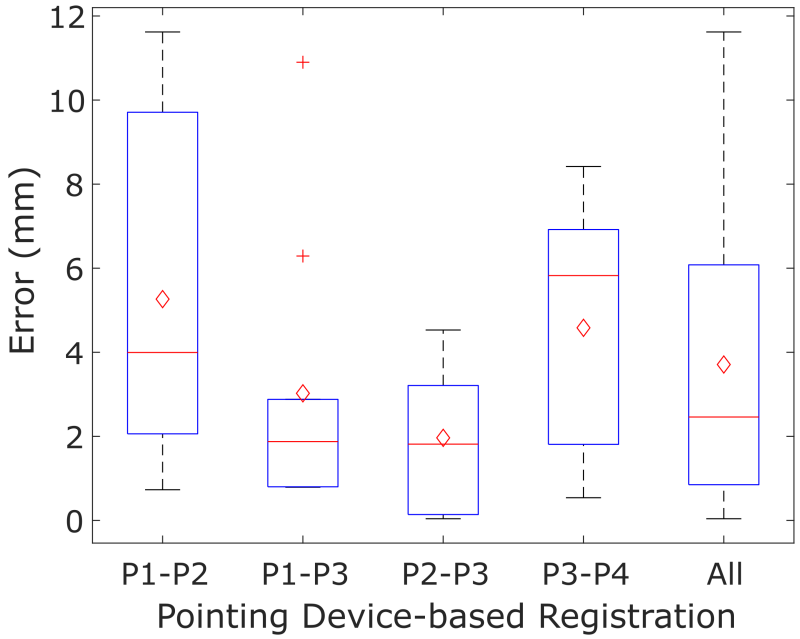


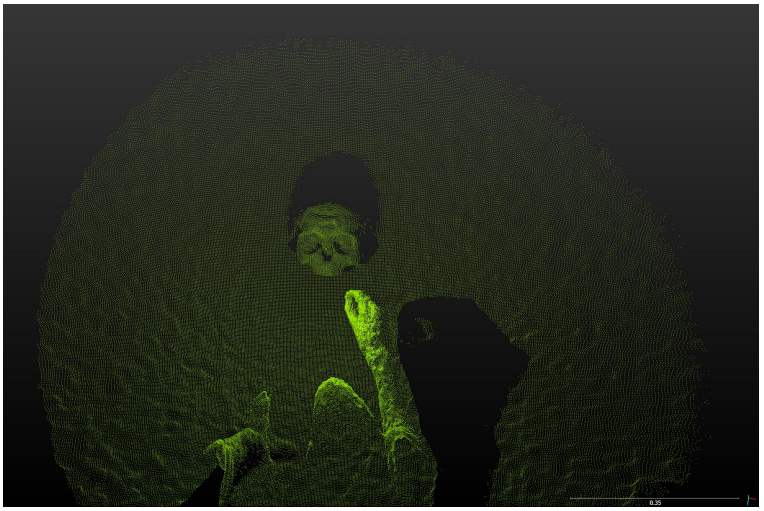
Figure 5.40.: Results of the pointing device-based registration. Mean values are visualized as red diamond markers, outliers as red cross markers.

5.5.2.4. Accuracy of the Registration with Surface Matching

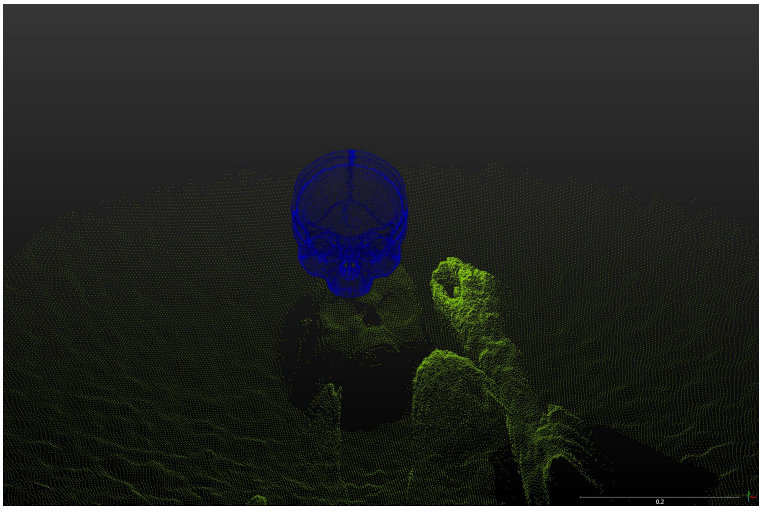
The evaluation of the automated registration using surface matching is evaluated by carrying out ten matching procedures. For this purpose, ten recordings of the short-throw sensors are made and combined to generate a high-resolution point cloud. The distance between the points is approximately 2 mm. An example is depicted in Figure 5.41 a). The patient model (Figure 5.41 b)) is then matched to the recorded point cloud as visualized in Figure 5.41 c). The registration error of the matching result is evaluated by measuring the distance of the matched patient model to the recorded patient anatomy. The distance between five points is measured. The four points defined earlier at the OpenHead are used and a fifth one is defined at the edge of the teeth. The additional measurement point was chosen to get a more distinctive landmark point to perform the measure on.

All ten matchings aligned the model P_S over the recorded OpenHead P_H. Out of ten matchings, eight were aligned sufficiently well, while two matchings had a bigger alignment error and were rotated incorrectly.

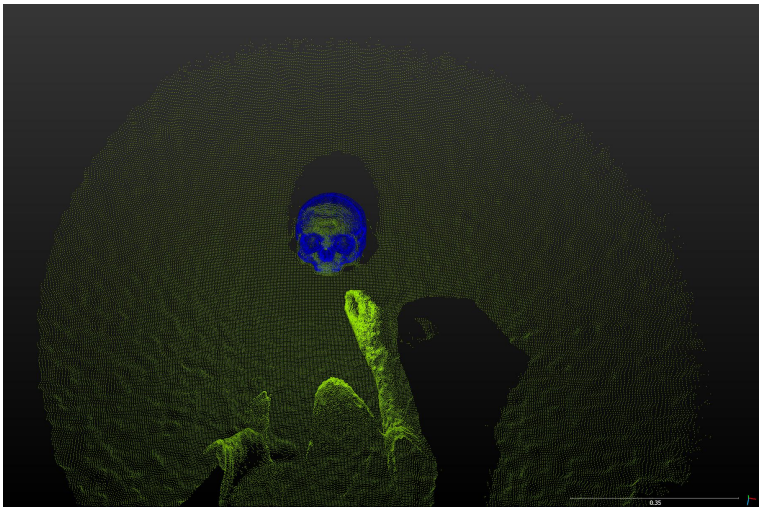
The results of all ten matchings are depicted in Figure 5.42. The misaligned point clouds are numbers 4 and 8 as can clearly be seen by the high error value in the plot. The reason



(a)



(b)



(c)

Figure 5.41.: a) Point cloud P_H recorded from the HoloLens short throw streams (green). b) The patient model point cloud P_S (blue) that will be matched to the recorded point cloud P_H. c) Matched point clouds after alignment procedure.

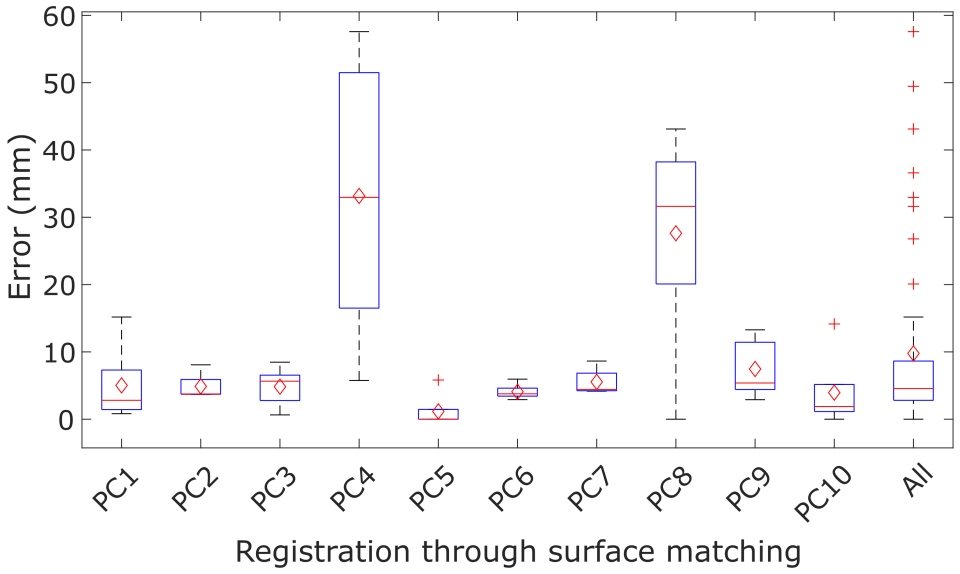


Figure 5.42.: Registration error of the point cloud matchings. All ten point clouds are visualized with numbers PC1 to PC10 and the overall error of all alignment procedures. Mean values are visualized as red diamond markers, outliers as red cross markers.

for the matching error is the run into a local minima due to a high sensor noise of the depth stream.

The registration results of the successful matchings are visualized in Figure 5.43. In the case a matching does not succeed, a slightly different recording angle should be chosen. The registration error was determined to be 9.77±13.43 mm including all registrations. When considering only the successful registrations an error value of 4.62±3.76 mm can be obtained.

5.5.2.5. Comparison of the Accuracy Measurements

In the previous sections all four registration methods have been presented and a detailed evaluation was carried out.

In Figure 5.44 and Figure 5.45 all methods are compared to each other. The controller- and gesture-based manual registration are the most accurate methods. In Figure 5.44 all surface matching results are included, while the failed alignments are for better comparison not included in Figure 5.45.

The worse results of the pointing device-based and surface matching registration methods are presumably due to a high sensor noise of the *short throw depth* sensor. A wavelike scattering noise is depicted in Figure 5.46. In Section 5.3 it was shown that an infrared marker can be tracked at sub-millimeter range. This value was obtained with a HoloLens mounted to a tripod. The tripod was not moving, the scene was static.

When the HoloLens is worn on the head of a user this accuracy drops to a lower accuracy value, due to the movement of the HoloLens. The evaluation of the registration methods that are using the depth stream indicate error values of 2-7 mm with high noise. In the

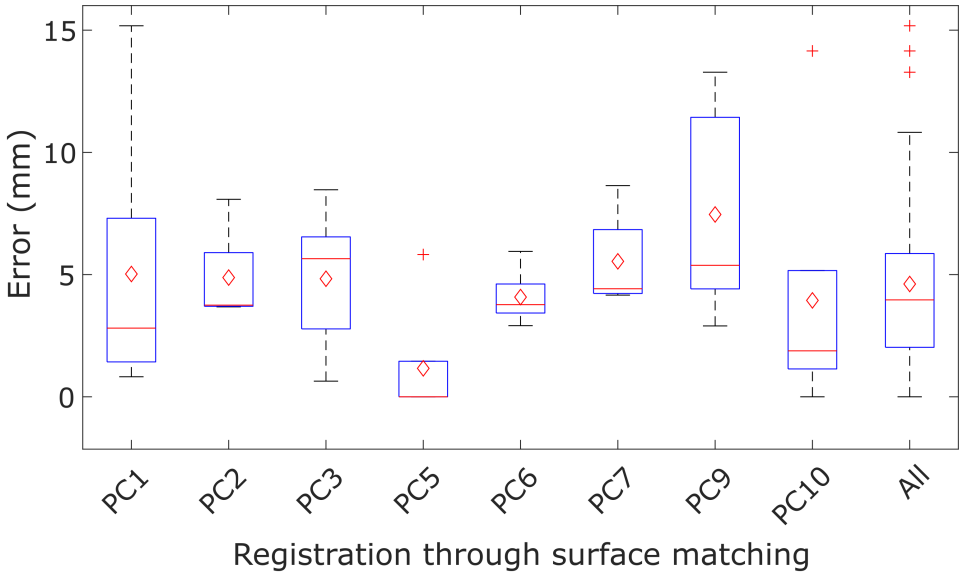


Figure 5.43.: Registration error of the point cloud matchings without misaligned point clouds PC4 and PC8. Mean values are visualized as red diamond markers, outliers as red cross markers.

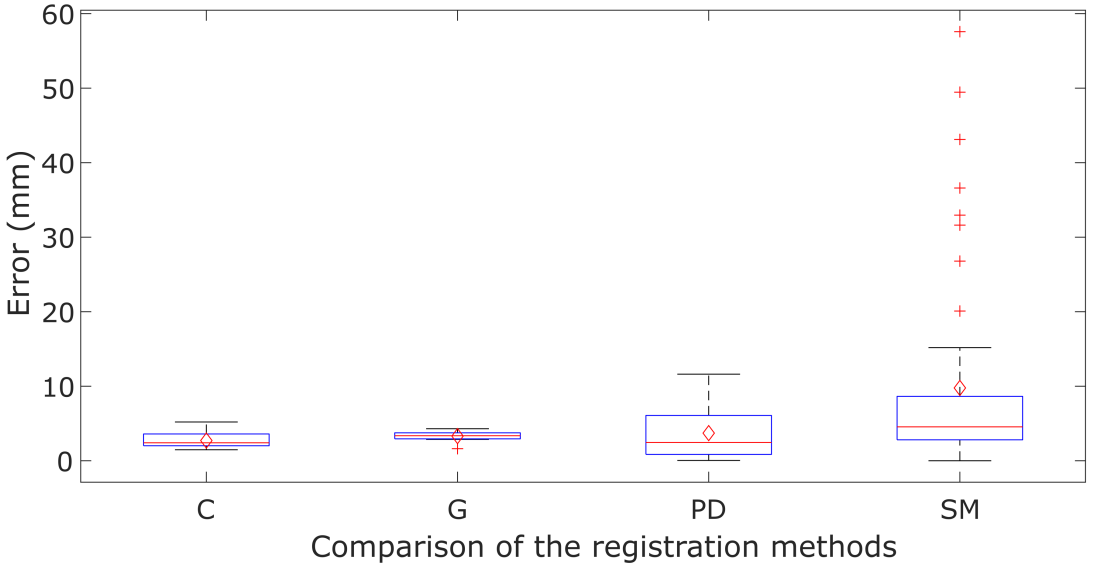


Figure 5.44.: Registration error of all four proposed methods. C = Controller, G = Gesture, PD = pointing device, SM = surface matching. Mean values are visualized as red diamond markers, outliers as red cross markers.

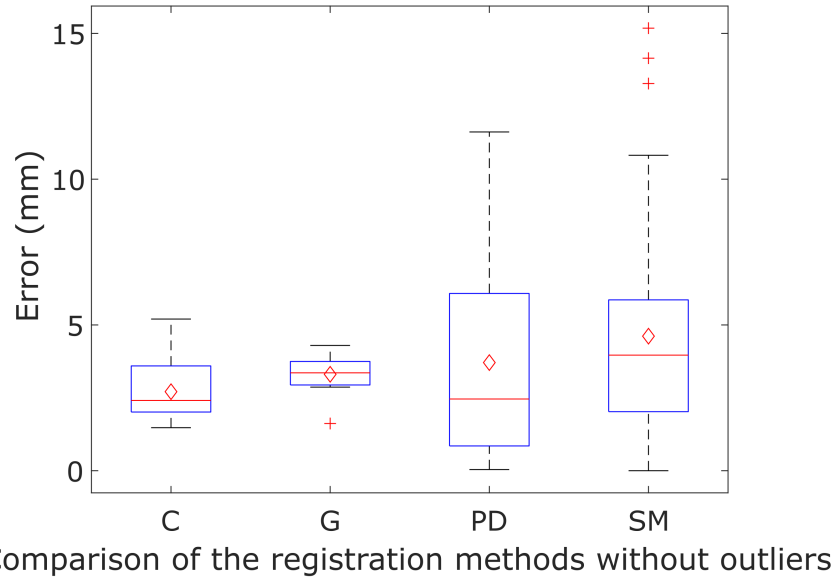


Figure 5.45.: Registration error of all four proposed methods without unsuccessful surface matching results. C = Controller, G = Gesture, PD = pointing device, SM = surface matching. Mean values are visualized as red diamond markers, outliers as red cross markers.

Registration methods	Time
Controller	100.5±63.5 s
Gesture	124.2±46.2 s
Pointing device	28.7±15.1 s
Surface matching	115.8-1227 s

Table 5.14.: Results of the time evaluation of the different registration methods.

future, further evaluations need to be performed to determine the tracking accuracy in relation to the movement of a user.

5.5.2.6. Time Evaluation

The time measurements were conducted in a row by two persons. The registration results were reviewed to ensure that they are in the evaluated range. Generally it must be stated that the time needed for the manual approaches depends strongly on the user's experience with the system. The time evaluations of the manual registration methods were done with two test participants ($n = 2$, computer scientists). The results are depicted in Table 5.14.

The automated registration method using surface matching takes the longest time, depending on the resolution of the point clouds to be matched. A detailed time evaluation is depicted in Table 5.15. The time needed for computation is mainly depending on the size of the voxel grid used. When a higher resolution is chosen, the feature-based approach needs much more time to compute the features and perform the initial alignment.

Voxel grid size	Points	Time
3 mm	4380-5125	1072.2 - 1227 s
4 mm	2719-3320	442.2 - 485.4 s
6 mm	1214-1311	115.8 - 130.8 s

Table 5.15.: Detailed time evaluation of the surface matching in relation to different voxel grid sizes.

5.5.3. Discussion

The manual registration with a game controller is easy to perform and provides a high precision input device. Through an initial tracking of the marker system a coarse alignment of the patient model can be done. The whole registration process relies on the visual perception of the HoloLens wearer. Therefore, a calibration needs to be performed as described in Section 6.1.1.3.

The registration using gestures works basically the same as the controller-based registration. Instead, hand and head gestures are used to align the models to the real phantom. Similar accuracy values can be achieved, although more training is needed to perform the alignment correctly.

The two registration methods using the pointing device or surface matching rely mainly on the quality of the short throw sensors. The infrared markers are tracked within these streams, and the point cloud that the patient model is registered to is generated from the short throw streams. We were able to obtain a resolution of 2 mm distance from point to point. As shown in Section 5.3 an infrared marker tracking accuracy of 0.76 ± 0.65 mm was achieved.

This high accuracy was possible because the center of the infrared markers were interpolated through the tracked pixels that glow bright in the short throw reflectivity stream. These accuracy values can be achieved when the HoloLens is fixed to a tripod.

Initial experiments indicate that the tracking accuracy decreases to 2-7 mm per infrared marker due to motion noise in the short throw streams, when the HoloLens is worn on the head. The high noise in the *short throw depth* stream is the biggest error source in the registration process when using the pointing device or the automated method via surface matching. This lowered tracking accuracy while the HoloLens is worn leads to a registration accuracy of 3.71 ± 3.25 mm for the pointing device and 4.62 ± 3.76 mm for the surface matching. The slighlty better registration accuracy of the pointing device method is presumably due to a better tracking performance of the infrared spheres, because sensor noise can better be minimized, when the center of the infrared sphere is determined.

It can be assumed that these accuracy values are better when the sensors on the HoloLens are updated in newer versions. The small but constant movement of the human head will remain a challenge. This could be adressed with appropriate algorithms or additional sensors, e.g. with an accelerometer. A further possibility would be a higher frame rate of the depth sensor.

The same applies to the registration over surface matching as the results are comparable to those of the pointing device. Again, the noise of the short throw sensors and their limited resolution leads to a limitation of the registration accuracy. Overall the results are good as eight out of ten matchings succeeded. If a higher resolution is chosen, the two misaligned

point clouds can be matched with the OpenHead model sufficiently. This comes at the price of longer time that is needed to perform the matching procedure. Future work will investigate how to speed up this workflow and to perform matching in a reasonable amount of time.

When evaluating the results of the surface matching in more detail, it can be stated that there are some irregularities in the measured data. Measurements were performed on five points to compare both point clouds. In one evaluation a nearly perfect matching was executed, with a detected misalignment on just one measurement point. This leads to the conclusion that every recorded point in a frame of the short throw depth sensor has a uncertainty of a few mm. When a planar surface is recorded the points are scattered, which leads to an uneven surface as depicted in Figure 5.46. This scatter noise correlates with the distance of an object from the HoloLens. In the working distance of this work, this error is determined to be 2.07 ± 1.98 mm, but can increase to several cm for distances over one meter. The error value was measured in the depicted point cloud.

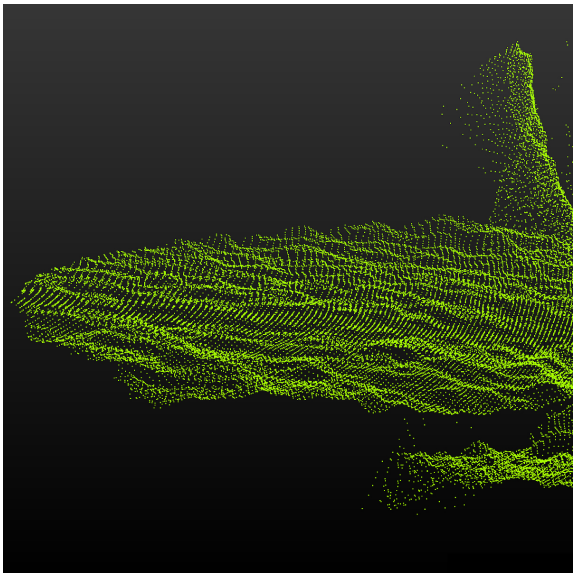


Figure 5.46.: Wavelike scattering noise of the recorded point cloud from the short throw depth stream.

The registration methods were evaluated on the OpenHead phantom printed from ASA, which represents a static surface and therefore ideal conditions of a matching of a patient model generated from prerecorded image data. The human skull with its skin is slightly deformable, especially in the jaw region. When translating the methods to a real clinical environment this problem would need to be addressed, too. The standard of care would be to fix fiducial markers to the skull of the patient prior to the CT scan and then use a pointing device to perform a registration.

5.6. Chapter Conclusion

In this chapter, methods for scene registration were presented. First the experimental setup was presented that was used to evaluate the proposed methods.

Then possible markers for patient tracking were presented and evaluated in regard to a possible usage in this work. As an outcome the Aruco, AprilTags, Vuforia and infrared

markers were chosen to be the most promising ones and evaluated further. The Vuforia markers are not open source but their tracking accuracy has already been published in literature. Aruco and AprilTags perform both sufficiently well, but Aruco markers can be tracked twice as fast as AprilTags. In the beginning of this work infrared marker tracking was not possible because access to the short throw sensors was restricted by the hardware provider. It was initially decided to implement the Aruco marker tracking on the HoloLens 1, which seemed to be the most promising candidate. However, no sophisticated framerate could be reached on the HoloLens. Therefore, it was decided to use the Vuforia marker tracking to derive the position of the patient.

After the research mode of the HoloLens 1 was introduced in 2019, sensor access was possible and a solution was presented to track infrared markers directly with the device. Two different approaches were investigated to do so. The first method utilizes the two *time of flight* sensor streams *str* and *std*. The second approach uses the environmental cameras *lf* and *rf*. Both approaches were evaluated against each other and it was decided to use the *time of flight* sensor. The accuracy is nearly equal but no additional light emitting LEDs are needed. An error value of 0.76 mm at a distance of 40-60 cm was measured. High framerates on the HoloLens of 55-60 fps are reached.

With the presented method it is possible to use well established and sterilizable infrared markers in the operating room to track the relative motion between the patient and the HoloLens.

To mount the markers to the patient, a marker system was developed. Various configurations are possible. The most important part of the marker system that is responsible for the reattachment error is the used connection between the marker base and the carrier. In a first version, a screw attachment was used. A reattachment accuracy of 0.18 ± 0.06 mm was reached. The second version uses a magnet attachment and yields better accuracy values of 0.018 ± 0.01 mm without sterile drape between base and carrier. When using a sterile drape as patient coverage this error increases, but can be minimized with a known drape thickness to 0.054 ± 0.019 mm. It was shown that the marker system remains stable even during vibrations and small movements of a patient.

In the last section, four methods for marker system to patient registration were presented. Three manual registration methods were proposed: via a XBox controller, a pointing device or gestures. Additionally, an automated method for the registration was presented using surface matching utilizing the short throw sensors. The evaluation showed that registration with a controller yielded the most accurate results and was simple to perform.

6. Intraoperative Assistance

In this chapter, the three different levels of augmented reality-based surgical assistance are presented. The structure is shown in Figure 6.1.

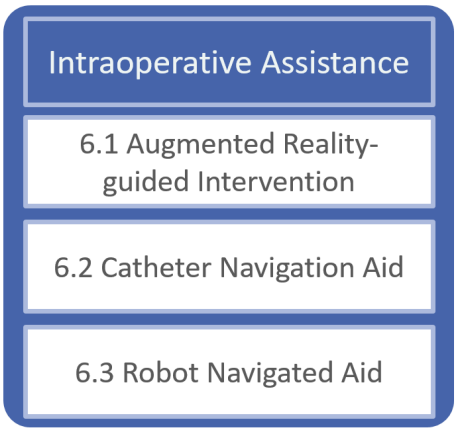


Figure 6.1.: Overview of chapter 6.

In the first stage, only the HoloLens is used to assist a surgeon during an intervention with augmented reality by superimposing the patient models over the patient's head (Section 6.1). The second stage uses an additional catheter navigation aid that is tracked with an infrared marker to additionally provide information about the puncture angle and Kocher point location to the surgeon (Section 6.2). In the last stage a robot is used to guide the catheter navigation aid to enable a more stable and accurate puncture trajectory (Section 6.3).

The three stages of intraoperative assistance were all validated with simulated ventricular punctures in preliminary and preclinical experiments. All experiments were conducted on the OpenHead phantom and its corresponding five interchangeable ventricular systems as presented in Section 5.1.1. The first stage, the AR-guided intervention was evaluated in a preclinical environment by expert neurosurgeons, while for the other two a proof of concept was done in preliminary experiments in the OP:Sense environment with novice non-medical experts having a technical background.

6.1. Augmented Reality-guided Intervention

In this section the first stage of the intraoperative assistance is presented.

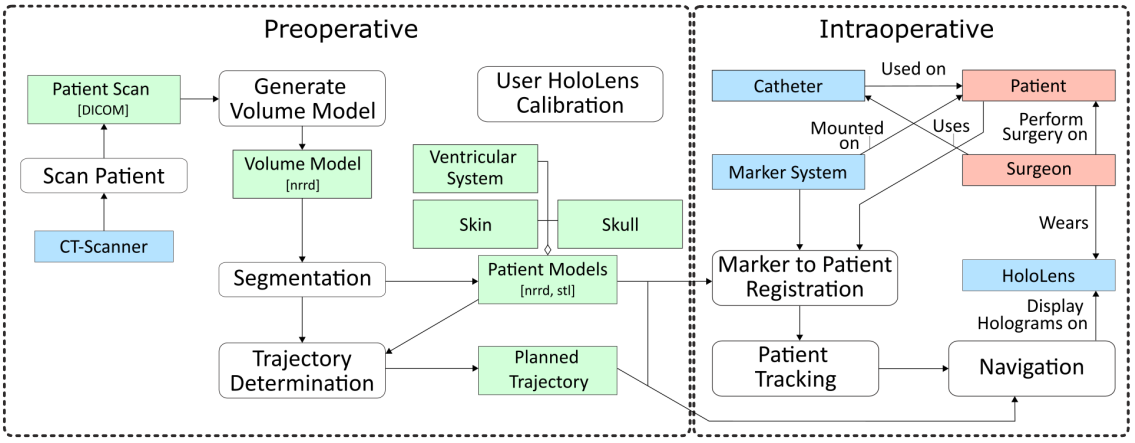


Figure 6.2.: Overview of the complete process of the proposed system. In the preoperative stage the patient models are generated, by segmenting the patient scans. After that the optimal puncture trajectory is determined. Before a surgeon can use the HoloLens it must be calibrated to the individual user. In the intraoperative phase the patient models are superimposed over the patient to support a surgeon during an intervention. Adapted from [119].

6.1.1. The HoloLens Application for Intraoperative Assistance

The HoloLens application for augmented reality support is started by a user when beginning the AR-guided ventriculostomy. The components described in Sections 4 and 5 are used in this application. The generated patient models and puncture trajectories are superimposed over the patient to guide a neurosurgeon during the intervention. The scene registration methods are used to register the patient to the HoloLens and to track the marker during the procedure and compensate for movement.

An overview of the preoperative and intraoperative process is given in Figure 6.2.

6.1.1.1. Workflow of the HoloLens Application

The workflow of the application is depicted in Figure 6.3. First a user can decide if they want to start a new intervention or if they want to load a previously saved patient to marker registration. The latter can be used to restart the HoloLens during an intervention or after a break. The *new registration* option performs an initial alignment by tracking the marker once. This leads to a registration accuracy of up to a few centimeters, depending on the position of the marker system. In the registration step one of the four methods as described in Section 5.5 can be used to perform the marker to patient registration.

When the registration step is completed the intraoperative assistance is executed. The medical expert can choose between different visualization modi. The different patient models can be switched on and off. It is possible to visualize the ventricular system, the skin, the skull, the target points inside the ventricular system, the Kocher's points and the puncture trajectories.

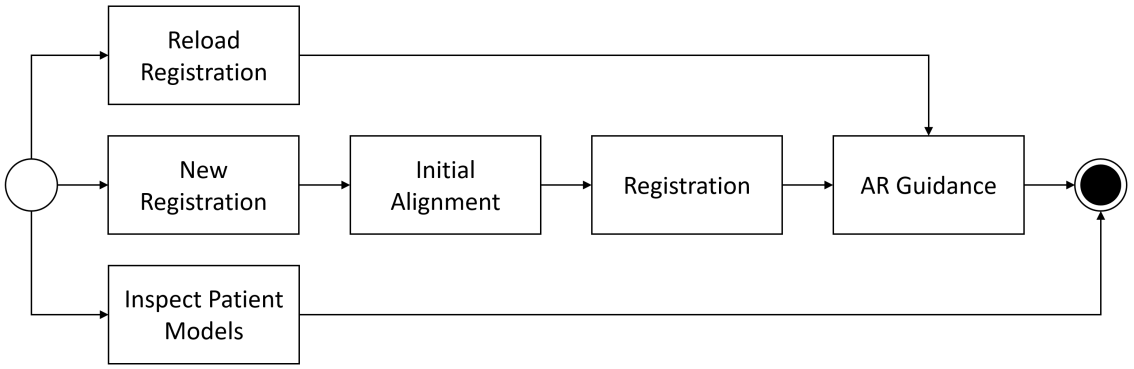
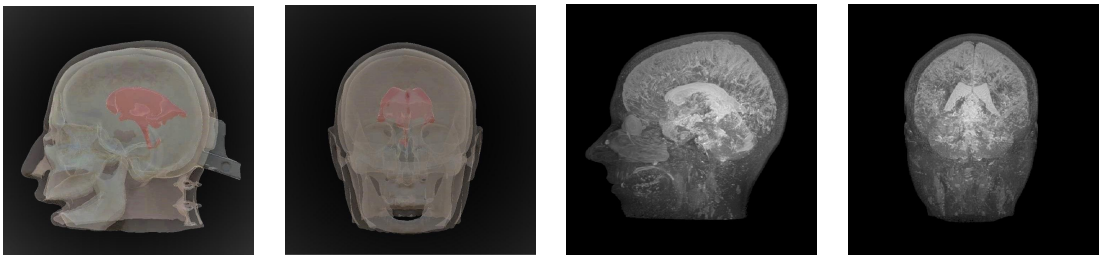


Figure 6.3.: The process of the HoloLens application to support a user with augmented reality.

6.1.1.2. Rendering Techniques

There are two possible ways to visualize patient models on the HoloLens. The standard way is to use the standard shader provided by Unity 3D to visualize surface models to the user. With the implementation of a ray trace shader it is also possible to render volume models on the HoloLens 1. Examples are visualized in Figure 6.4.



(a) (b) (c) (d)

Figure 6.4.: a,b) Surface models (Synbone phantom). c,d) Volume models (real patient) [111].

The surface models are stored as *obj* files and added as an asset to the Unity project. The volume files are stored as *nrrd* files and converted to a binary file representing a three dimensional array. The binary file is also saved as an asset in the Unity project. Both assets can then be accessed from the appropriate shader. A detailed evaluation regarding the runtime behaviour was carried out. Each model dataset consisted of three models: the skin, the skull and the ventricular system. Properties of the used surface models are summarized in Table 6.1. For reduction of the vertex count Blender (v.2.79b) was used. Two patient model datasets of different size were investigated.

The same applies to the volume models. Also different volume model sizes were evaluated. The models were reduced using the *unu* resample tool with default parameters [202]. Applications running on HoloLens are limited to a memory allocation of approximately 900 MB. When trying to render the original size (100%) of the volume models, 950 MB was allocated in RAM and the application crashed.

In Table 6.2 the evaluated volume models are depicted.

To render the volume models on the HoloLens a ray trace shader was implemented, based on the approach introduced in the work of Amanatides et al. [8]. The shader has been

	Model 1 (real patient data)				Model 2 (Synbone phantom head)			
	Skin	Skull	Ventricles	Total	Skin	Skull	Ventricles	Total
Vertex count	4 595	8 103	4 068	16 766	17 633	42 310	5 977	65 950
Triangle count	8 942	16 255	8 149	33 346	34 994	83 805	11 922	130 721
Size (in kB)	280	500	248	1 028	1 154	2 839	369	4 362

Table 6.1.: Details of the two combined surface models that are used for evaluation (skin, skull and ventricular system). Model 1 was created from real patient data. Model 2 was generated out of a CT-scan of the Synbone phantom [111].

	Head 100%	Head 75%	Head 50%
Dimension (in voxel)	512 x 160 x 512	384 x 120 x 384	256 x 80 x 256
Voxel size (in mm)	0.488 x 1.000 x 0.488	0.653 x 1.333 x 0.653	0.980 x 2.000 x 0.980
Binary file size	160 MB	67.5 MB	20 MB

Table 6.2.: Volume models: The volume model was generated from a CT-scan of a real patient and later on reduced to 75% and 50% of its size [111].

implemented to mimic a CT scan to make it possible to look *through* a patient's head and inspect the ventricular system.

The method utilizes a fast voxel traversal algorithm that sends rays through the three-dimensional volume to project the scene onto a two-dimensional plane. The algorithm calculates the pixel value based on the Hounsfield units of the voxels that lie on the ray.

Using all voxels on one ray would be too computational expensive. Therefore, the voxels taken into account are limited to a certain number. In this work 16 or 96 sample points on the rays are evaluated in regard to the rendering performance of the volume model visualization. The more sample points are used, the more computational expensive is the calculation of the two-dimensional projection. The mean value of all voxel values is calculated, also taking their neighbourhood into account (all 27 connected voxels) to determine the pixel value. A visualization of the volume models is depicted in Figure 6.4 c and d.

The scene displaying the surface or volume models were started and measurements were made. The evaluation was performed using the built-in performance tracker of the HoloLens. The evaluation results are the average of a one minute measurement. The RAM utilization was determined after the scene was loaded. In Table 6.3 the results of the evaluation are depicted. The SoC power utilization describes the system on a chip power usage averaged over one minute [150]. The SoC power utilization may represent inaccurate values when the framerate is below 60 fps.

	Volume models					Surface models	
	Head 100%	Head 75%		Head 50%		Model 1	Model 2
Samples per ray	-	16	96	16	96	-	-
FPS	-	54	13.5	52	13	60	46
CPU utilization (in %)	-	11	3	10	2.5	12	9
RAM (commit size) (in MB)	-	507.2	509.4	271.3	271.1	162.8	164.7
SoC power utilization (in %)	-	100	95	61	61	97	116

Table 6.3.: Render performance of the volume and surface models on the HoloLens [111].

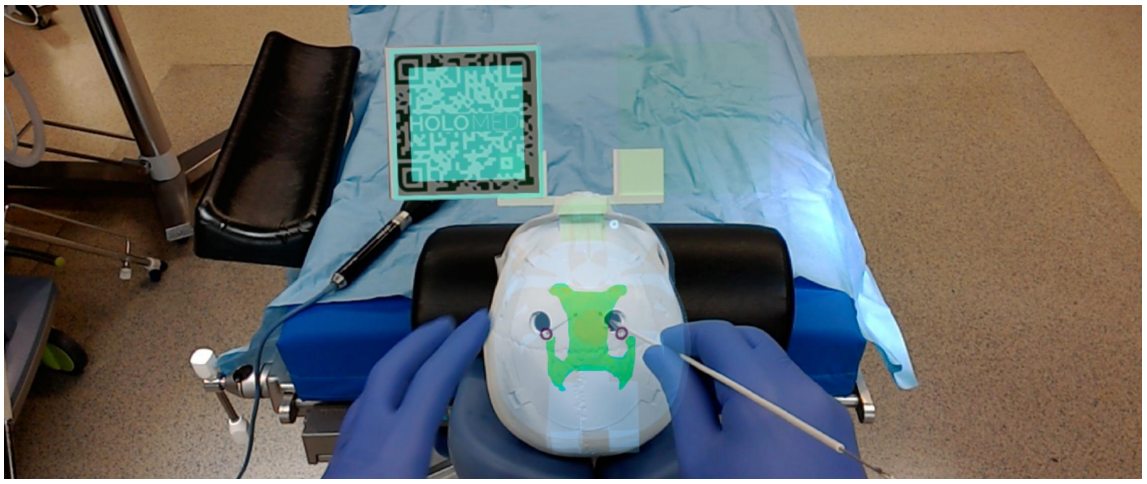


Figure 6.5.: Visualization of the intraoperative assistance.

Discussion of the Hologram Visualization

Both models can be visualized on the HoloLens using an appropriate shader with an acceptable speed as depicted in Table 6.3. The volumetric models need to be of a usable size else the maximum RAM allocation is exceeded. It is also important not to choose a to high 'samples per ray' count to achieve an acceptable framerate. The volume model visualization is especially suitable to view the patient models before an intervention to study the patient anatomy. Due to its limited framerate of 54 fps this rendering method should not be used during an operation. The surface models can be rendered much faster on the HoloLens with a stable framerate of 60 fps when the model size is not too high (roughly < 4 MB).

6.1.1.3. Augmented Reality Guidance

In the last step the intraoperative assistance is activated. A scene from the experimental validation is shown in Figure 6.5.

The hidden structures that lie within the patient's head, i.e. the Kocher's points, the ventricular system, the optimal puncture points inside the VS and the puncture trajectories are visualized as navigation aid.

The patient surface models are generated within 6.9±0.6 seconds, the automated path determination is calculated in 57.0±3.4 s. The generated surface models can then be accessed from the network. Looking at the complete surgical workflow, a navigation aid is possible in under five minutes. The neurosurgeon can then place the catheter based on the augmented holograms for a successful puncture of the ventricular system.

The representation of the characteristics of the vizualised patient models were designed with three expert neurosurgeons. An evaluation was performed to determine suitable colors, their combination and alpha values for optimal usability even under various light conditions present in the operating room. A detailed user study was carried out. Different visualizations were tested in a realistic OR environment and rated regarding their usability.

Three example color combinations are shown in Figure 6.6. The figures can be misleading as the visibility when seen through the HoloLens is different. A user may choose from

different visualization templates to customize the provided navigation aid. In the experiments, the OpenHead was used, therefore just the model of the skull is used and shown in Figure 6.6, whereas the skin is neglected. Theoretically, all models could be switched on or off, but four combinations are preselected for the user. The skull in combination with the ventricular system or just the ventricles can be visualized. Additionally, the puncture trajectories can be turned on and off. When just the ventricles are used the user has a clearer view on the operation situs, while the turned on skull gives more orientation help. The marker system is also visualized to the user during the complete procedure to verify a proper tracking functionality. The tracked marker position is indicated with a light blue rectangle that glows brighter when the marker is detected in one frame. A shader from the Microsoft Mixed Reality Toolkit is used that implements a transparent optic of the patient models. In addition, the hardware buttons on the left side of the HoloLens may be used to control the brightness of the holograms manually to give a surgeon full and additional control of the visualization brightness.

After the registration process a fine calibration of the user to the marker system is performed. As the HoloLens 1 does not provide eye tracking, this needs to be done manually. The wearer aligns the edges of the marker system with the edges of the hologram. In most cases the hologram positions are well aligned and just a small displacement is visible. A wrong position of the HoloLens on the wearers head can lead to a misalignment of more than 1 cm. In the HoloLens configuration the interpupillary distance (IPD) is set for every user individually to visualize the holograms at the correct position.

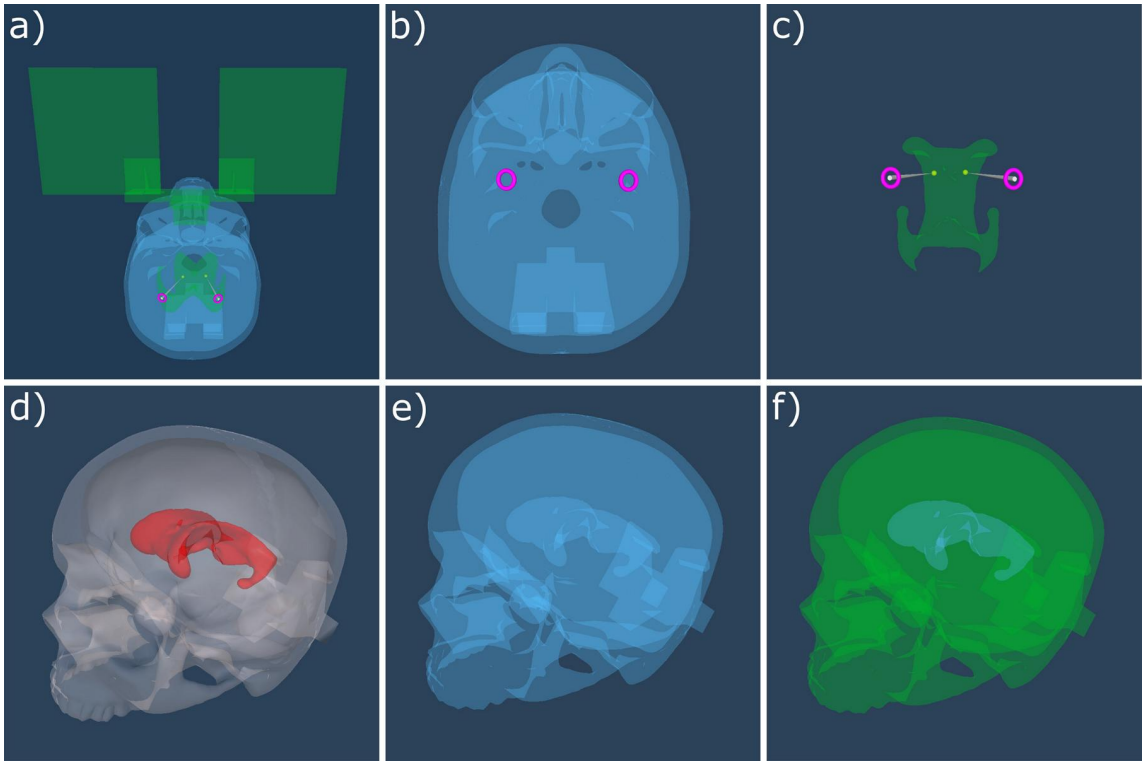


Figure 6.6.: a) Final representation of the patient model visualization with included Vuforia marker system, b) representation of the Kocher's points in magenta and the skull in blue, c) visualization mode with just the VS, the trajectories (grey) and optimal points (yellow) visible, d-e) different color representations of the patient models that were evaluated in regard to visibility [119].

6.1.2. Preliminary Experiments

Eight non-medical novices with technical background were asked to perform ten ventricular punctures each on the OpenHead phantom. One catheter placement was done for every left and right ventricle of the five designed ventricular systems of the OpenHead, as introduced in Section 5.1.1 and visualized in Figure 6.7 a).

6.1.2.1. Experimental Validation

The preliminary experiments were conducted in the OP:Sense laboratory environment (see Section 5.1.2). The participating subjects (n=8) were non-medical novices with technical background. One OpenHead was used with predrilled burr hole trepanations of 11 mm diameter in the detachable skull cap at the Kocher's points. The novices were not asked to localize and burr the Kocher's point trepanation.

All five ventricular systems were modeled from soft modeling clay with the printed moulds as depicted in Figure 6.7 a and b. The surface was modeled with a mould cap with integrated pins of the reference points so that two small indentations are left after the modeling process, one in each ventricle, respectively. The indentations are filled with a small amount of soft modeling clay of white color. The subjects had to consecutively place a disposable skewer in the left and right ventricle of every ventricular system, simulating catheter placement. After both sides were punctured the mould that is holding the ventricular system formed from modeling clay was taken out and replaced with the next one until all five ventricular systems were punctured. The participants punctured with no resistance through air, as no filling of the OpenHead was used. In contrast the preclinical experiments were conducted on OpenHead phantoms filled with agar gel to simulate brain tissue as described in Section 6.1.3. A filling was not possible for the preliminary experiments as no CT scanner was available. After the experiment, all 5 ventricular system moulds were prepared for digitalization with a 3D line scanner. When the ventricles were hit by the skewer an indentation was created. To measure the distance error to the marked reference points the new intendations are also filled with a small amount of modeling clay of different color. A 3D line scanner (25xx-100, resolution in z: 12 μm, x: 0.15 mm, y: 0.1 mm, Micro-Epsilon, Germany) was used to digitalize the results. The euclidian distance is then measured in the point cloud from the reference point to the puncture result. When the ventricle is not successfully punctured, the puncture is counted as a non hit.

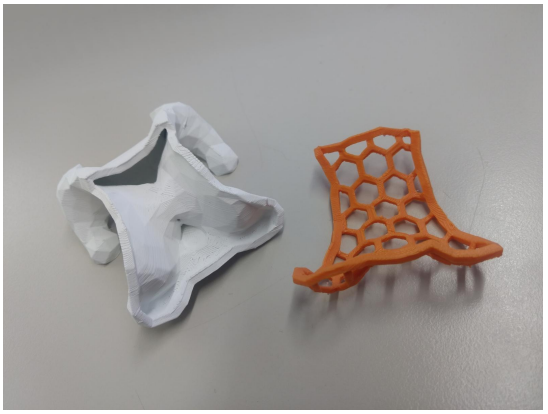
The evaluation metric is visualized in Figure 6.8.

The OpenHead used in the experiments has a fixed mount for the marker system, so that the transformation T_{MS}^{P} is known. This is done to separate all potential error sources and excluding the registration error.

6.1.2.2. Results

The results of the preliminary experiments of the non-medical novices with technical background are visualized in Table 6.4. For each of the five ventricular systems the mean error, the standard deviation and the success rate is depicted for the left and right ventricles. The success rate describes the amount of correctly placed catheters. The overall success rate of all catheter placements was 91.25% with a mean error of 5.6 ± 2.8 mm.

(a)



(b)

Figure 6.7.: a) Five OpenHead phantoms with all modeled ventricular system moulds stacked in. b) Ventricular system mould for taking and modeling the clay to receive the desired shape.

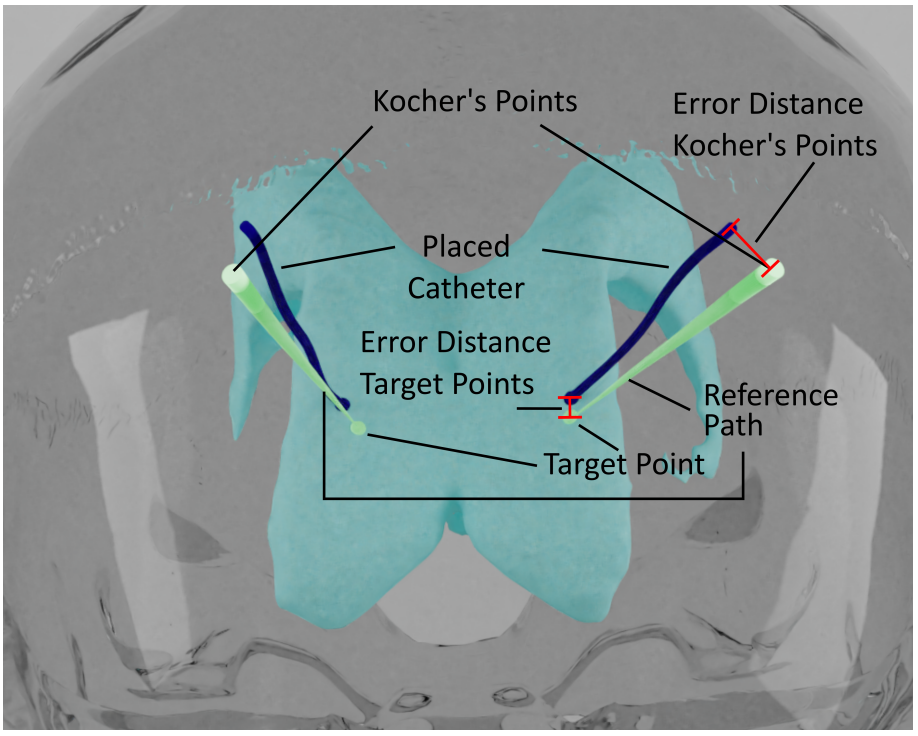


Figure 6.8.: Evaluation metric of the puncture experiments for determination of the placement error. The reference path is compared to the placed catheter. Preliminary experiments: The pentration points on the ventricular system are compared to the annotated reference points. Preclinical experiments: Additionally, the chosen Kocher's points are compared to the annotated points. All values given in mm. Adapted from [119].

The highest error values ocurred when puncturing the left ventricle of the third ventricular system with 8.8 ± 3.1 mm. However, all placed catheters were rated as success as they would have resulted in a drainage of cerebrospinal fluid in a real surgery. The displacement was in y-direction on the ventricle. Low success rates were observed for the 5th ventricular system (left: 87.5%, right: 62.5%), which is the most difficult one to puncture as described in Section 5.1.1. This is due to its narrow shape and its steep surface. The lowest displacement error was found for the right of the second ventricular system with a mean error of only 4.0 ± 1.8 mm, which is a large and symmetric ventricular system.

A graphical representation of the puncture results is depicted in Figure 6.9.

The qualitative evaluation of the system that supported the novices with augmented reality showed good acceptance ratings. The anatomical structures that were superimposed over the OpenHead were visualized clearly without clutter. It was also appreciated that the brightness was adjustable because every person perceives the holograms differently. This way it was possible to react to differing light conditions. It was also rated as good that the system was intuitively controllable.

Rated negative was the small field of view of the HoloLens 1 and the sometimes unsatisfactory contrast of the hologram colors.

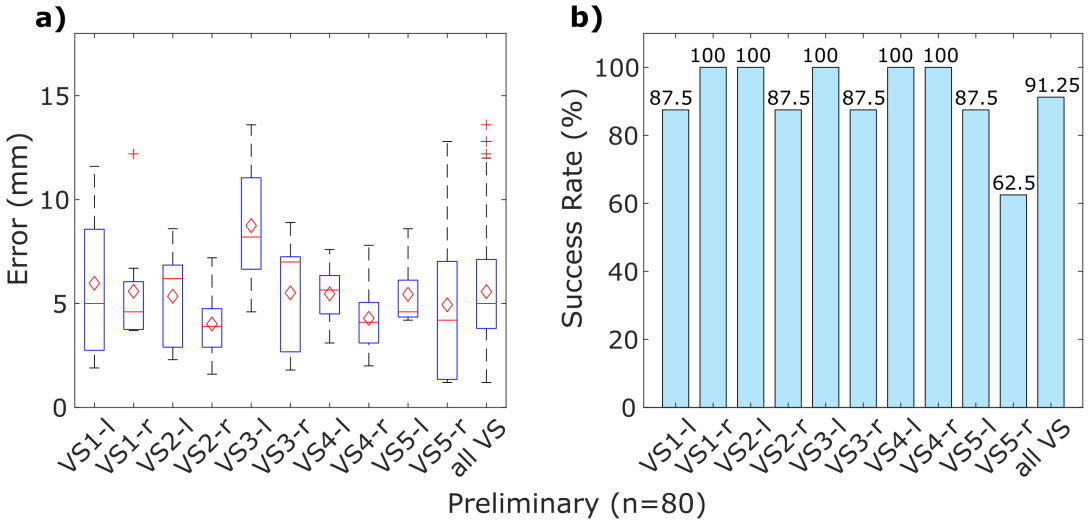


Figure 6.9.: Preliminary experiments: a) Puncture results for all five ventricular systems, b) Success rate. Mean values are visualized as red diamond markers, outliers as red cross markers.

Table 6.4.: Preliminary experiments: Distance to reference points (mean error), standard deviation (SD) and overall success rate for all 5 left and right ventricles. Adapted from [119].

Novice (n=80)	VS1L	VS1R	VS2L	VS2R	VS3L	VS3R	VS4L	VS4R	VS5L	VS5R	All VS
Mean error (mm)	6.0	5.6	5.4	4.0	8.8	5.4	5.5	4.3	5.4	4.9	5.6
SD (mm)	3.6	2.9	2.3	1.8	3.1	2.8	1.4	1.9	1.6	4.7	2.8
Success rate (%)	87.50	100	100	87.5	100	87.5	100	100	87.50	62.50	91.25

6.1.3. Preclinical Experiments

Preliminary experiments demonstrated the general feasibility of AR-guided interventions. In a next step, the ability of the system to support a surgeon during a neurosurgical intervention in a preclinical environment was evaluated. Two experiments were carried out.

First, complete ventriculostomies were performed after the standard of care without augmented reality support. This was done to evaluate the performance of the participating neurosurgeons in the phantom experiments. Then experiments were conducted where the neurosurgeons used the proposed system supported with augmented reality. The control experiments according to the standard of care were conducted by 11 neurosurgeons (8 neurosurgeons and 3 neurosurgical residents). A total of $n = 110$ ventriculostomies were carried out. The planning of the procedures were done with an in-house medical imaging system.

The augmented reality supported experiments were conducted with a total of 11 neurosurgeons (8 neurosurgeons and 3 neurosurgical residents). Three test series were conducted. Ten neurosurgeons participated in the first test series. Three neurosurgeons participated in all three test series. One experiment was not valid due to a failure of the system, therefore, one neurosurgeon participated in two test series. In every test series, a neurosurgeon was asked to puncture the left and right ventricle of all five OpenHead phantoms. This led to a total of $n = 100$ ventriculostomies in the first test series. In the second and third test series $n = 40$ ventriculostomies were carried out, respectively. A total of $n = 180$ ventriculostomies were performed.

The second and third test series were conducted to investigate the influence of experience with the system to the evaluation results and to identify the learning curve.

After an experiment a participating neurosurgeon was asked to fill out a questionnaire to determine the level of acceptance and usability of the AR system. Additionally, they were asked to rate the five ventricular systems according to their difficult level, in other words how hard they think it is to hit the presented ventricular system correctly.

6.1.3.1. Experimental Setup

The preclinical evaluation was done with medical expert neurosurgeons in the university hospital in Günzburg. Nearly the same experimental setup was used as described in Section 6.1.2.1. Instead of using just one OpenHead, five printed skulls were used with one ventricular system each. The preparation of the OpenHead phantom is depicted in Figure 6.10. The phantoms were filled with agar gel at a concentration of 0.5% to simulate brain tissue after cooling and hardening [38].

The neurosurgeons were asked to puncture all ten venricles with and without AR support. The standard of care experiments were performed without AR-glasses. The AR-guided procedure was carried out as described in Section 6.1.1.2. As for the preliminary experiments also a fixed marker position was used and no registration needed to be performed. The surgeons wore the AR glasses during the whole procedure and placed the catheters aided by the visualized trajectory and the reference points. An XBox controller (Microsoft Corp., Redmont, WA, USA) was used to control the user interface. Conventional neurosurgical catheters were used to puncture the ventricles. Additionally, the neurosurgical experts were asked to determine the position of the Kocher's points on the detachable skull cap. The burr hole trepanation was then drilled with a surgical drill with diameter 11 mm (Stryker Corp., Kalamazoo, Michigan, U.S.). The same skull cap was used on all five OpenHead phantoms. When one ventricular system was punctured, the skull cap was detached and attached to the next head. Care had to be taken not to remove the placed catheters in the process.

After all five OpenHead phantoms were punctured, they were placed on a bracket and a CT scan (Siemens Healthcare, Erlangen, Germany) was made of all.

The metric described in Figure 6.8 was used to determine the puncture error in the CT data. In the preliminary experiments a 3D laser scanner was used to measure the displacement error on the surface of the punctured ventricles. The volumetric CT data enables a more accurate investigation and determination of the displacement error. A reference CT scan was made from a specially prepared OpenHead. The Kocher's points were marked with a piece of a catheter and the reference points in every ventricular system as well. The neurosurgical catheters have a high Hounsfield unit, comparable to bone, and thus were easy to detect and localize. The puncture results were then compared to these reference points.

Both, the experiments according to the standard of care and those supported through AR-guidance, were evaluated in the same way.

The coordinate system of the patient model is aligned with the patient. The elongated shape of the ventricular system is following the y-axis. The x-axis is defined from ear to ear of the patient. The two insertion angles defining the catheter placement are depicted

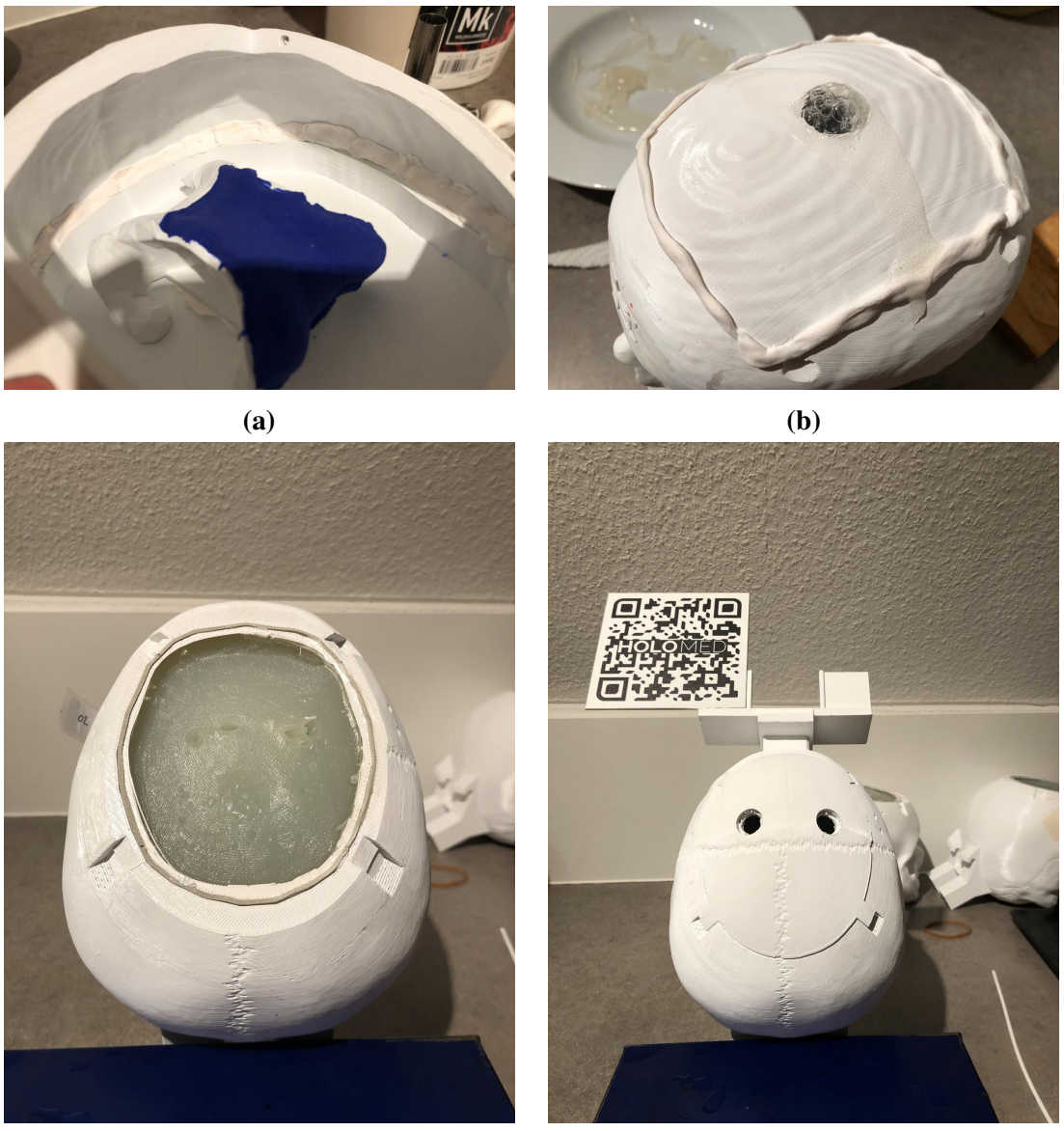


Figure 6.10.: a) OpenHead phantom with inserted ventricular system mould and modeled clay. b) OpenHead phantom after filling with agar. c) Agar after it cooled and solidified. d) OpenHead with attached Vuforia marker system and detachable skull cap with drilled burr hole trepanation at the Kocher's points.

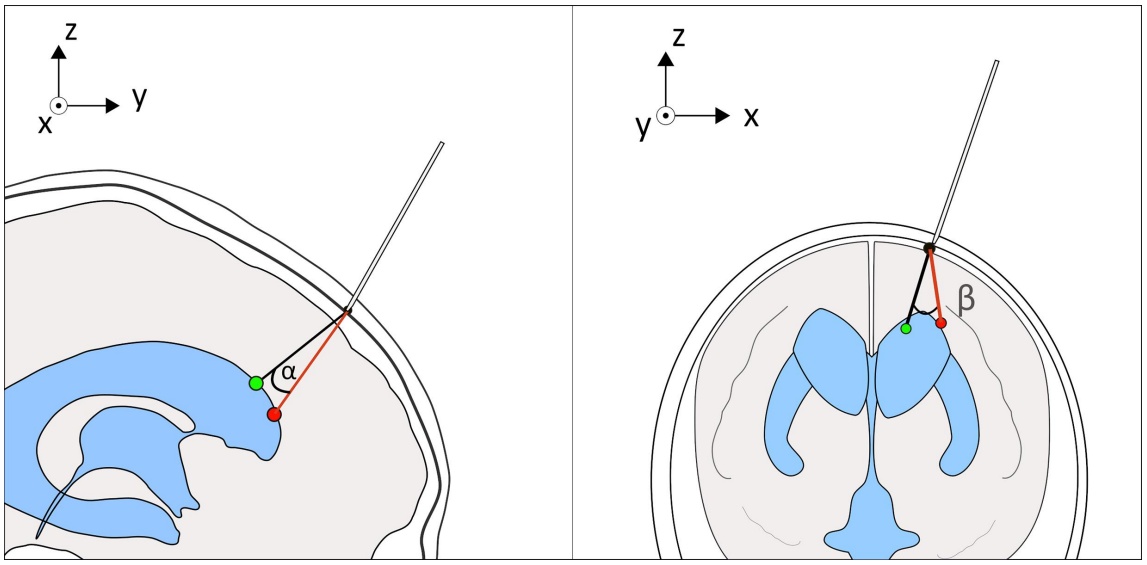


Figure 6.11.: Representation of the two insertion angles that define the puncture trajectory of the catheter placement. In the left upper corner of the left and right figure the coordinate system of the patient model is depicted.

in Figure 6.11. The angle α is defined between the z and y-axis and the angle β between z and x-axis. A change of the α angle changes the end point of the insertion following the y-axis of the ventricular system and its elongated shape. If, on the other hand, the angle β is changed, the end point changes in the direction of the x-axis. If the angle β is wrongly chosen, this is more serious than a wrongly chosen angle α, because the ventricular system is only 1-2 cm wide in x-direction. Due to the elongated shape of the ventricular system larger displacements are possible in y-direction.

6.1.3.2. Results of the Standard of Care Experiments

The results of the standard of care experiments of the medical experts are presented in Table 6.5 and depicted in Figure 6.12. For all ten ventricles of the five ventricular systems the mean error, standard deviation and success rate are visualized for the catheter placements inside the ventricular system. The evaluation of the displacement error of the Kocher's points determination is presented in Table 6.6 and visualized in Figure 6.13. A puncture is rated as a success if the ventricular system is perforated. The overall success rate of the standard of care experiments is determined as 72.73% with an error of 6.6 ± 3.1 mm. The Kocher's points were determined with an accuracy of 8.45 ± 3.0 mm with 100% success rate.

Table 6.5.: Standard of care experiments (no AR-support): Mean error, standard deviation (SD) and overall success rate for all 5 VS (left and right ventricles).

SoC ($n_c = 110$)	VS1L	VS1R	VS2L	VS2R	VS3L	VS3R	VS4L	VS4R	VS5L	VS5R	All VS
Mean error (mm)	8.7	6.8	5.2	5.3	8.0	4.4	7.1	6.6	6.6	5.1	6.6
SD (mm)	2.8	3.7	1.7	2.6	5.3	2.6	2.9	2.3	1.9	1.8	3.1
Success rate (%)	90.91	90.91	100	54.55	72.73	36.36	100	63.64	72.73	45.45	72.73

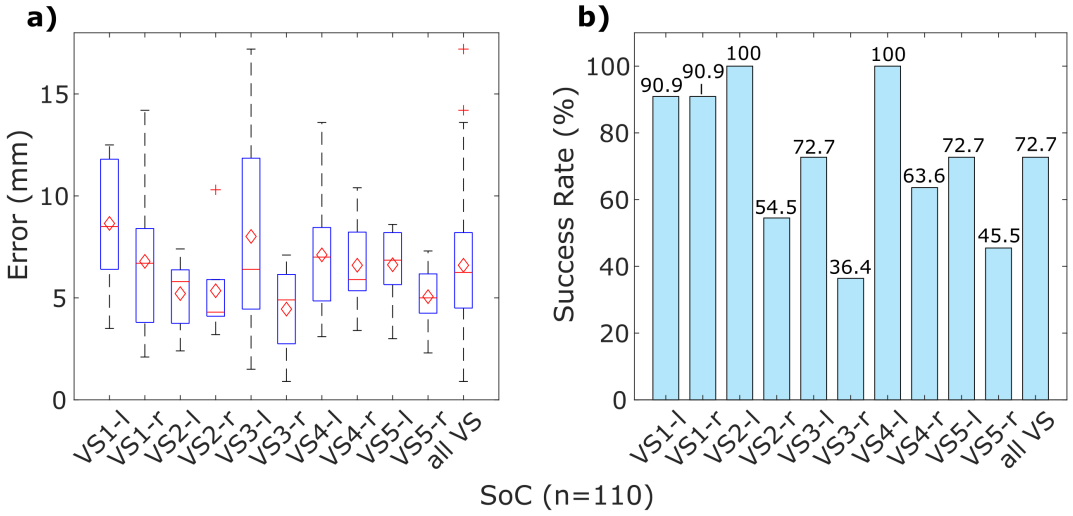


Figure 6.12.: Standard of care experiments: a) Results of the catheter placements for all five ventricular systems, b) Success rate. Mean values are visualized as red diamond markers, outliers as red cross markers.

Table 6.6.: Standard of care experiments (no AR-support): Mean error, standard deviation (SD) and success rate for the left and right side Kocher's points (K) determination ($n = 22$, 11 participants, both sides).

SoC ($n_c = 22$)	K-l	K-r	All K
Mean error (mm)	7.7	9.2	8.45
SD (mm)	3.0	3.1	3.0
Success rate (%)	100	100	100

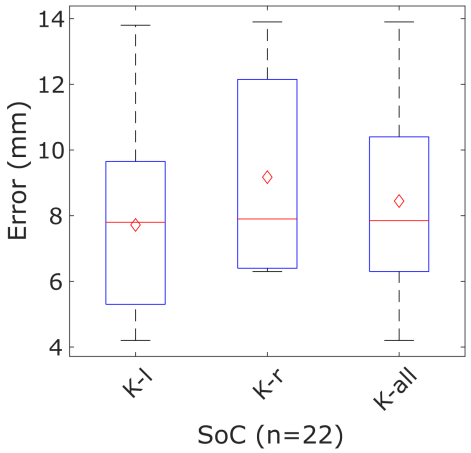


Figure 6.13.: Standard of care experiments: Results of the Kocher's points determination for the left and right side and overall values. Mean values are visualized as red diamond markers.

6.1.3.3. Results of the AR-guided Experiments

In a next step, ventriculostomies were performed with augmented reality support of the proposed system. The results of the experiments are presented in Table 6.7. The mean error, standard deviation and puncture success rates are visualized in Figure 6.14 for the

catheter placement inside the ventricular system. The ventricles were punctured with an accuracy of 4.8 ± 2.5 mm in 81.7 % of cases. The results for the Kocher's points determination are presented in Table 6.8 and depicted in Figure 6.15. The Kocher's points were found in 100% with a mean error of 5.4 ± 2.4 mm.

Table 6.7.: AR-guided experiments: Mean error, standard deviation (SD) and overall success rate for all 5 VS (left and right ventricles). All test series $n = 180$. Adapted from [119].

AR-guided ($n_{AR} = 180$)	VS1L	VS1R	VS2L	VS2R	VS3L	VS3R	VS4L	VS4R	VS5L	VS5R	ALL VS
Mean error (mm)	3.9	4.6	5.0	5.8	5.4	4.4	5.7	4.7	4.1	3.6	4.8
SD (mm)	1.5	2.3	3.0	2.7	3.4	2.7	2.1	2.3	1.8	2.0	2.5
Success rate (%)	88.9	88.9	100	83.3	72.2	55.6	94.44	88.9	88.9	55.6	81.7

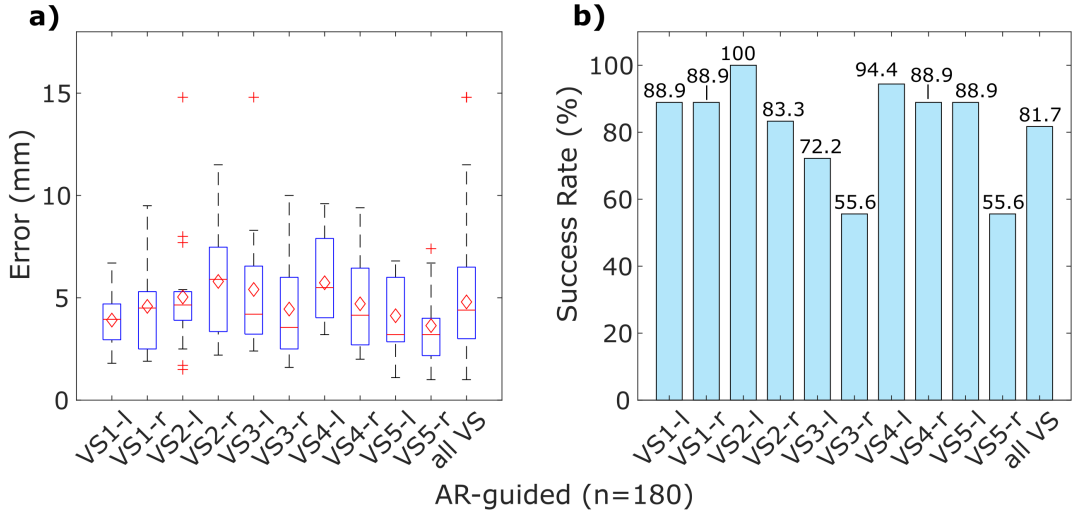


Figure 6.14.: AR-guided experiments: a) Results of the catheter placements for all five ventricular systems, b) Success rate. Mean values are visualized as red diamond markers, outliers as red cross markers. Adapted from [119].

Table 6.8.: AR-guided experiments: Mean error, standard deviation (SD) and overall success rate for determination of both Kocher's points (K). All test series $n = 36$. Adapted from [119].

AR-guided ($n_{AR} = 36$)	K-l	K-r	K
Mean error (mm)	5.5	5.3	5.4
SD (mm)	2.7	2.2	2.4
Success rate (%)	100	100	100

The experiments were carried out by eleven medical experts who punctured in one to three test series ten ventricular systems each. Three neurosurgeons punctured three times, one of them two times and seven once. This resulted in the participation of 10 neurosurgeons in the first test series and four in the second and third test series. One experiment of test series one was not rated as there was a problem with the visualization.

In Table 6.7 and Figure 6.14 the results of all 180 punctures are depicted, not divided into test series. Now, a closer look is taken at the puncture performance in relation to the test series.

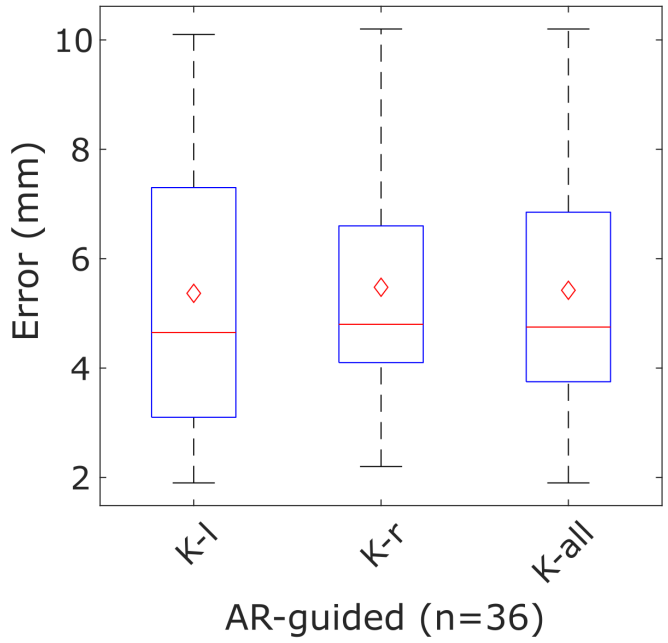


Figure 6.15.: AR-guided experiments: a) Results of the Kocher's points determination for both sides. Mean values are visualized as red diamond markers. Adapted from [119].

Table 6.9.: AR-guided experiments - Test series one: Mean error, standard deviation (SD) and overall success rate for all 5 VS (left and right ventricles).

$n^1_{AR} = 100$	VS1L	VS1R	VS2L	VS2R	VS3L	VS3R	VS4L	VS4R	VS5L	VS5R	ALL VS
Mean error (mm)	3.0	5.3	5.9	7.5	4.6	7.8	6.6	5.4	4.2	3.5	5.3
SD (mm)	0.8	3.0	3.3	2.4	1.1	2.0	2.3	2.8	1.7	2.3	2.6
Success rate (%)	80	80	100	70	50	30	90	80	100	50	73

Table 6.10.: AR-guided experiments - Test series two: Mean error, standard deviation (SD) and overall success rate of the AR-guided evaluation for all 5 VS (left and right ventricles).

$n^2_{AR} = 40$	VS1L	VS1R	VS2L	VS2R	VS3L	VS3R	VS4L	VS4R	VS5L	VS5R	ALL VS
Mean error (mm)	4.3	4.3	2.8	3.8	5.2	2.8	5.2	3.3	4.5	4.0	4.0
SD (mm)	1.8	1.3	1.8	2.2	2.6	0.9	2.0	1.0	1.9	-	1.8
Success rate (%)	100	100	100	100	100	100	100	100	75	25	90

The results of the first test series are depicted in Table 6.9 and Figure 6.16. A puncture accuracy of 5.3 ± 2.6 mm was reached with a success rate of 73%.

Table 6.10 and Figure 6.17 show the results of the second test series. The ventricles were penetrated with a puncture accuracy of 4.0 ± 1.8 mm and a success rate of 90%.

Table 6.11.: AR-guided experiments - Test series three: Mean error, standard deviation (SD) and overall success rate of the AR-guided evaluation for all 5 VS (left and right ventricles).

$n^3_{AR} = 40$	VS1L	VS1R	VS2L	VS2R	VS3L	VS3R	VS4L	VS4R	VS5L	VS5R	ALL VS
Mean error (mm)	5.2	3.5	5.2	4.8	6.7	3.2	4.3	4.7	3.3	3.7	4.5
SD (mm)	1.2	1.4	1.9	2.0	5.8	2.0	1.1	2.2	2.6	2.4	2.5
Success rate (%)	100	100	100	100	100	75	100	100	75	100	95

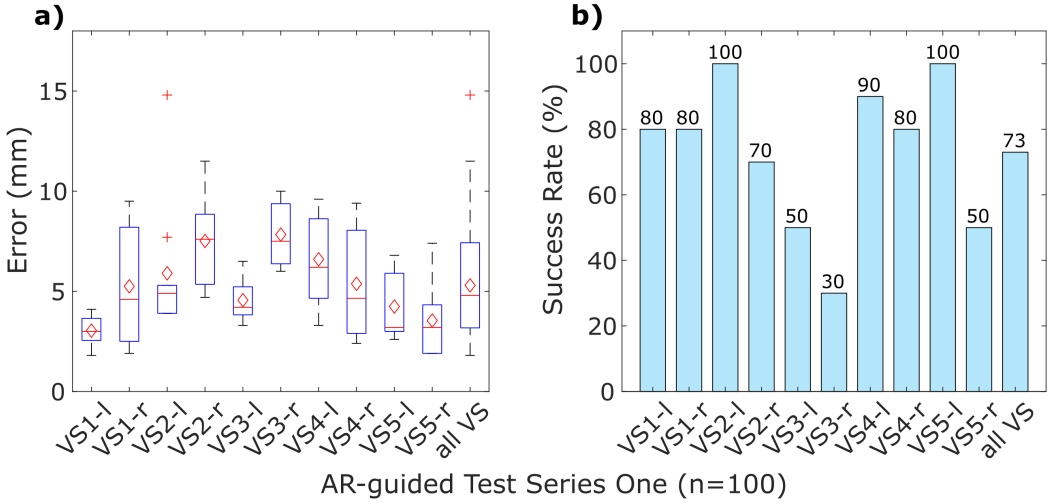


Figure 6.16.: AR-guided experiments - Test series one: a) Results of the catheter placements for all five ventricular systems, b) Success rate. Mean values are visualized as red diamond markers, outliers as red cross markers.

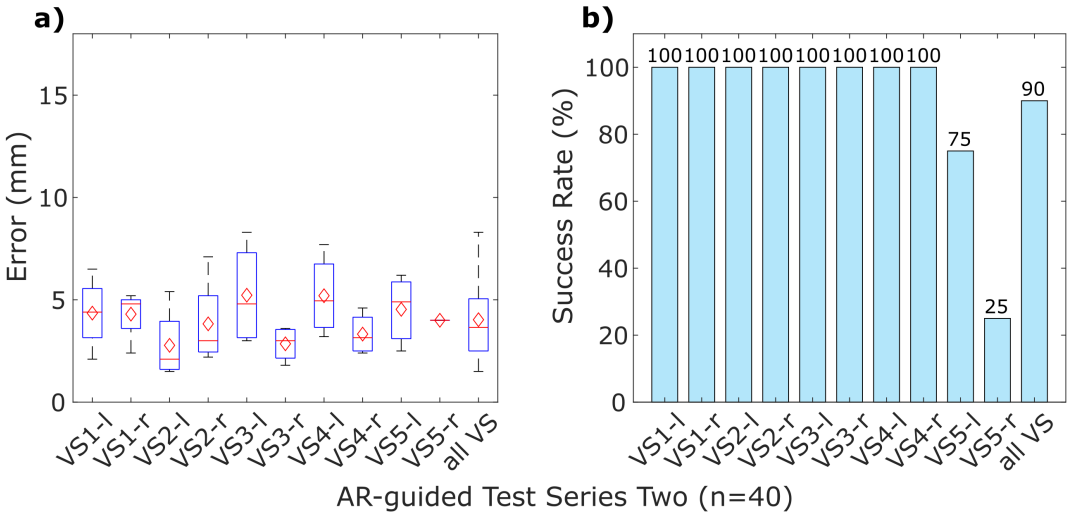


Figure 6.17.: AR-guided experiments - Test series two: a) Results of the catheter placements for all five ventricular systems, b) Success rate. Mean values are visualized as red diamond markers, outliers as red cross markers.

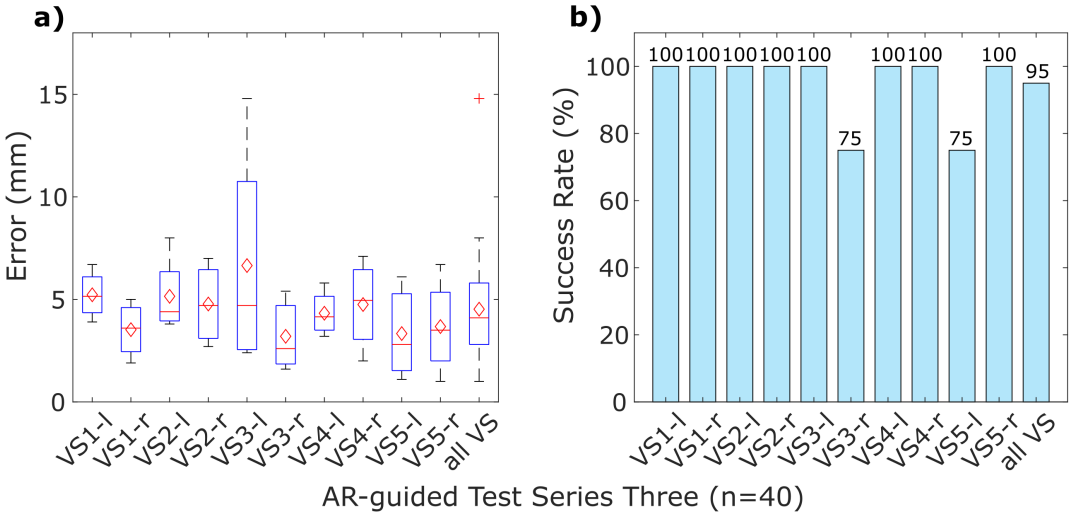


Figure 6.18.: AR-guided experiments - Test series three: a) Results of the catheter placements for all five ventricular systems, b) Success rate. Mean values are visualized as red diamond markers, outliers as red cross markers.

Table 6.11 and Figure 6.18 show the results of the third test series. In the last test series a success rate of 95% and a puncture accuracy of 4.5 ± 2.5 mm were reached.

The comparison shows that the success rate increased throughout the test series and that there is a steep learning curve when using the proposed system. Already, after the second test series a siginificantly better success rate is observed. The mean error also decreased but did not improve from test series two to three. A possible explanation for this is, that the participants learned how to use the system and minimized the catheter placement error in x-direction (β). This is much easier to accomplish than to minimize the error in y-direction. Due to the elongated shape the error in y-direction does not have such a significant impact than in x-direction (α). The errors and the success rates are depicted in Figure 6.19.

The Kocher's points determination accuracy could also be improved over the test series. In the first test series an accuracy of 5.5$\pm$2.73 mm was achieved. In the second test series this could be improved to 5.14$\pm$2.11 mm. In the last test series the mean error was determined to be 4.33$\pm$1.79. The results are depicted in Figure 6.20.

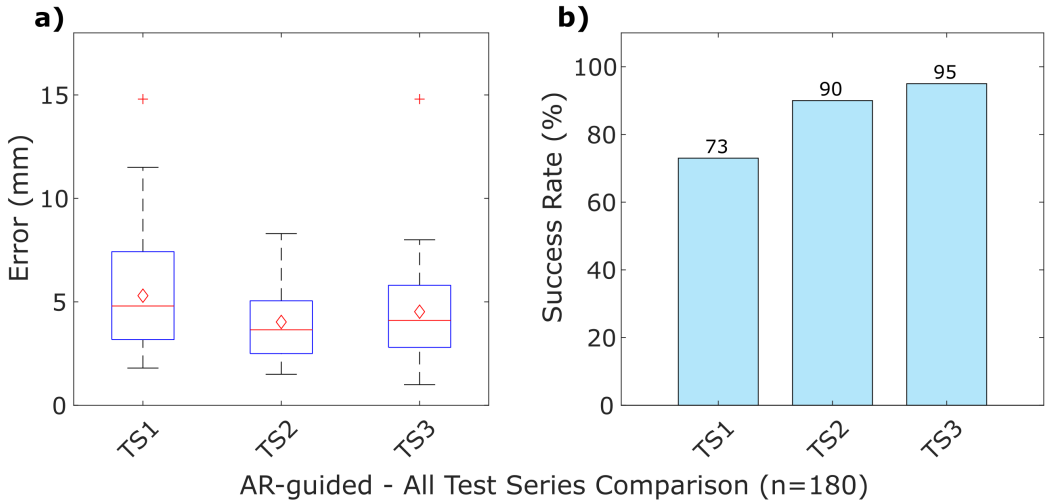


Figure 6.19.: AR-guided experiments - All Test series comparison: a) Results of the catheter placements for all five ventricular systems, b) Success rate. Mean values are visualized as red diamond markers, outliers as red cross markers.

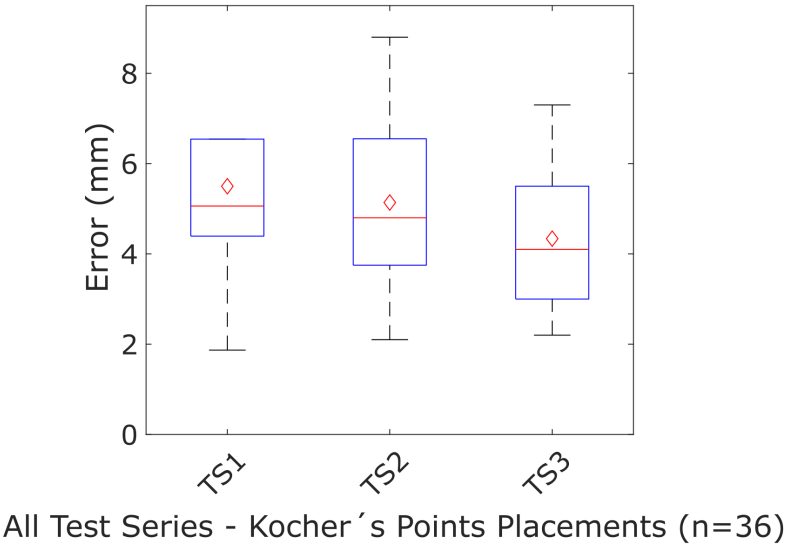


Figure 6.20.: AR-guided experiments - All Test series comparison: Results of the Kocher's points determination. Mean values are visualized as red diamond markers, outliers as red cross markers.

After the puncture experiments the medical experts filled out a user questionnaire for a quantitative evaluation of the system for AR-guided interventions. The surgeons were asked to rate the following aspects:

- The system is convenient to use
- The system has the potential to improve the accuracy of my cranial procedure
- The system would speed up my workflow
- I would use the system in the operating room
- I believe my patients would benefit from the system
- If the system was a product, I would buy it.

They were asked to rate each aspect as "strongly agree", "agree", "neutral", "disagree" or "strongly disagree". The results are depicted as a Likert scale in Figure 6.21.

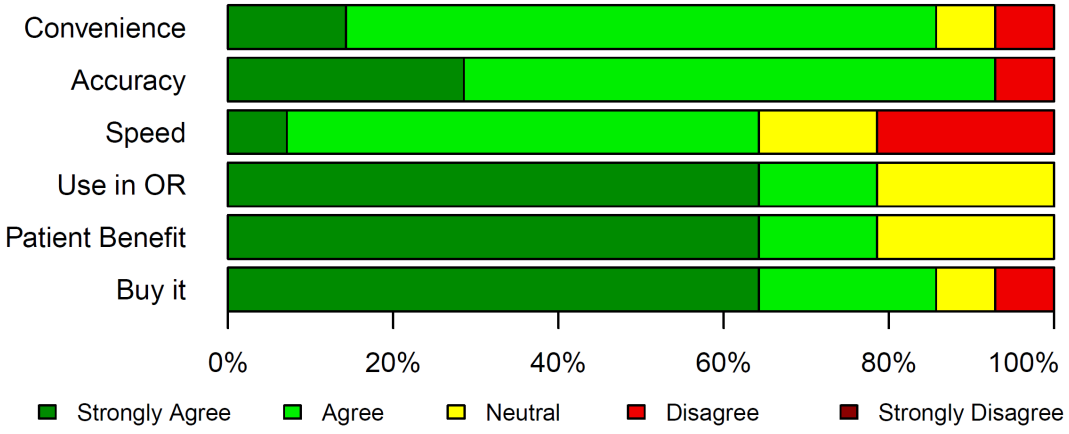


Figure 6.21.: Results of the expert user acceptance survey for the AR navigation aid (ARNA). Users answered to statements on a 5 point Likert scale: 1) ARNA is convenient, 2) ARNA improves accuracy of ventriculostomy, 3) ARNA speeds up my workflow, 4) I would use ARNA in the OR, 5) My patients would benefit from ARNA, and 6) If ARNA was a product I would buy it. [119]

The convenience of the navigation aid has been approved by 85.6% of the participating surgeons (*strongly agree* or *agree*) and 92.9% feel that the system has the potential to improve accuracy of cranial procedures. A lower number of surgeons (64.2%), however, believe that AR support would speed up the surgical workflow. Furthermore, 78.6% of surgeons in the survey would use the navigation aid in the OR and think that patients would benefit from the system. In summary, 85.7% would consider purchasing the system if it was a marketed product.

Additionally, the neurosurgeons were asked to rate the ventricular systems according to their puncture difficulty. They were asked to rate the difficulty as "very easy", "easy", "neutral", "challenging" and "very challenging". The results are depicted in Figure 6.22.

Each of the first two ventricle systems (VS1, VS2) were rated *easy* and *very easy* by 84,6% of the surgeons. VS3 was considered *challenging* by all surgeons. Regarding VS4,

46.2% of the surgeons rated it *easy* and *neutral*, while 7.7% viewed it as *challenging*. The fifth and last ventricle system (VS5) was considered *challenging* by 76,9% and even *very challenging* by 7.7%.

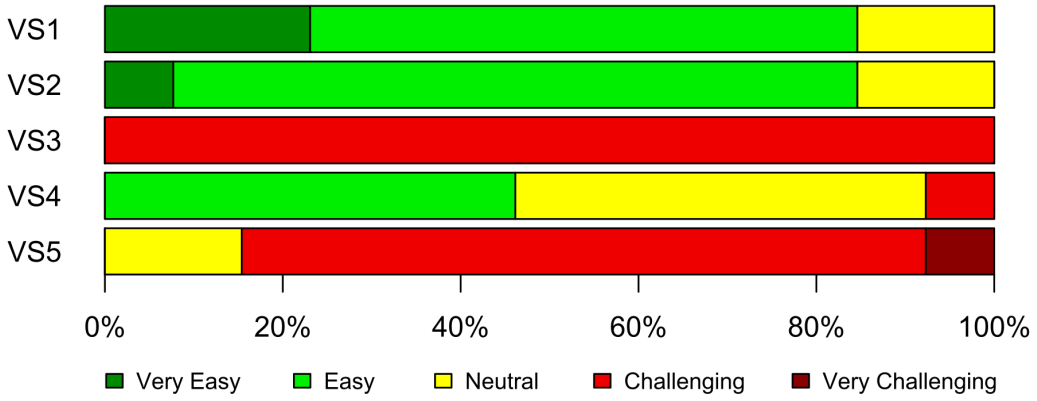


Figure 6.22.: The five ventricular systems used throughout the experiments rated according to their difficulty.

The results are confirming the puncture experiments, as for all catheter placements the third and fifth ventricles were the most difficult to puncture (i.e. with the highest errors).

6.1.3.4. Discussion

The preliminary experiments have proven the general feasibility of AR-guided ventriculostomies. One limitation with the laboratory experiments was that the subjects punctured through air inside the head phantom. This made it possible to freely change the insertion vector until the left or right ventricle was perforated. In contrast to the preclinical experiments where this was not possible, because the skull was filled with agar gel. The OpenHead was filled with agar gel which made it impossible to change the insertion vector once inside the skull. Therefore, it was important to choose a good initial insertion vector because the movement is limited to two degrees of freedom, allowing only forward motion of the catheter.

The comparison of the standard of care to the AR-guided experiments show a significantly higher success rate and puncture accuracy. A 9% better succes rate is reached and a higher accuracy of approximately 2 mm. This is an increased accuracy that can make the difference between a successful and a non-successful ventricular puncture considering that the ventricular system is only 1 to 2 cm wide.

A fixed marker position was used, so the registration error was minimized and not taken into account in the experiments. When looking at the results of the registration presented in Section 5.5 an additional error needs to be considered.

When the system should be used in a real clinical scenario, several additional factors need to be considered. The registration of a real patient could be rendered more difficult than on a static phantom, although the human head is relatively rigid. The standard of care includes usage of fiducial markers attached to the human skull prior to the CT. These markers could then be used in combination with the pointing device to perform a precise

registration. Also, brain shift limits accuracy during a real intervention, although this mainly happens when larger portions of the skull are removed. Brain shift describes the displacement of the brain over time.

The test series showed a steep learning curve, from 73% in the first trial to 95% in the third test series. This shows that significantly higher succes rates can be achieved with AR-support, but the surgeon needs to get confident with the system through tracking.

6.2. Navigated Intervention

The biggest limitation with the presented system for AR-guided interventions is that a good initial trajectory needs to be chosen, because it is not possible to change the angle once inside the brain. At this point just a forward motion is possible.

To overcome these shortcomings, possibilities are investigated how a surgeon can be supported in this process. Therefore, a guidance aid was investigated that is tracked with an infrared marker target with the HoloLens, that is referred to as catheter navigation aid (CNA) in this work.

6.2.1. Catheter Navigation Aid

The catheter navigation aid is depicted in Figure 6.23 b and c. It has a small tube that is used to guide the catheter to its target point inside the ventricle. A standard of care reinforced neurosurgical catheter is depicted in Figure 6.23 a.

When the CNA is tracked through the HoloLens the transformation from the HoloLens coordinate system to the CNA T_H^{CNA} needs to be derived. The coordinate source of the catheter navigation aid is in its tip. To do so, the following equation is used:

$$T_H^{CNA} = T_H^{MT} \cdot T_{MT}^{CNA} \tag{6.1}$$

The marker target is tracked with the HoloLens to derive T_H^{MT}. The transformation from the marker target to the tip of the CNA T_{MT}^{CNA} is known through the CAD data of the marker target and the navigation aid.

The tip of the navigation aid is placed in the trepanation of the Kocher's point K so that the tip of the navigation aid C_t equals the position of the Kocher's point. The system knows the exact position of the Kocher's points and the target points T inside the ventricular system because the models are registered to the scene. The insertion vector $\vec{a}$ is derived from the Kocher's point K to the target point T inside the ventricular system. The second vector $\vec{b}$ is calculated from the entry point on the tube C_e to the tip C_t of the navigation aid.

The angle between $\vec{a}$ and $\vec{b}$ is constantly derived and visualized to the user. The goal is to manually minimize this angle (α) to get the optimal puncture trajectory.

$$\alpha = cos^{-1} \cdot \frac{(\vec{a} \cdot \vec{b})}{(|\vec{a}| \cdot |\vec{b}|)} \tag{6.2}$$

(a)

(b)

(c)

Figure 6.23.: a) Neurosurgical catheter. b) Catheter navigation aid (sideview). c) Top view.

The coordinate system and the described vectors are visualized in Figure 6.24.

A second option is to derive the angle between the vector $\vec{a}$ and the vector from the tip of the navigation aid C_t to the target point T inside the ventricular system as depicted in Figure 6.24, for the left and right ventricle respectively. This way the displacement of the Kocher's point is taken into account and the navigation aid is always aiming towards the target points.

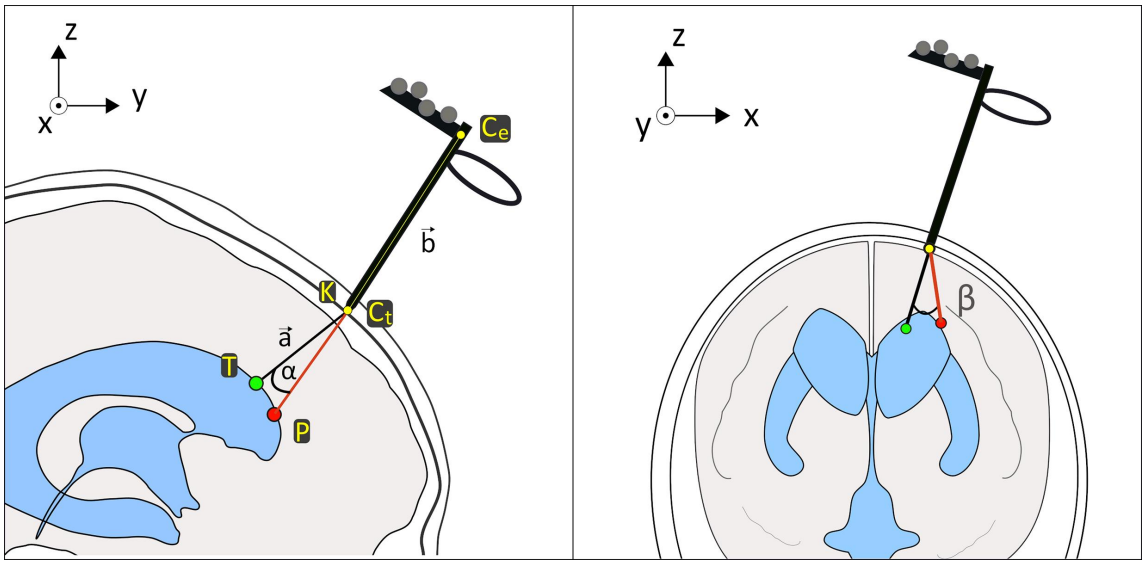


Figure 6.24.: The used coordinate system in combination with the CNA.

6.2.2. Experimental Setup

The same experimental setup was used as presented in Section 6.1.2.1. Additionally, the catheter navigation aid was used to visualize the insertion angle to the user. Due to the experimental setup the experiments are comparable to the evaluation with medical experts, because the catheter navigation aid was used to determine a good initial puncture trajectory of the catheter without the possibility to change the angle when inside the skull. The setup is depicted in Figure 6.25.

Two novice non-medical experts with technical background performed $n = 50$ ventricular punctures on the standard five ventricular systems. The results were again digitilized with a 3D line scanner.

6.2.3. Results

In Table 6.12 the results of the navigation aid supported experiments are summarized. Stated are the mean error to the reference points, the standard deviation and the overall success rate of the punctures.

Table 6.12.: CNA-guided experiments: Mean error, standard deviation (SD) and overall success rate of the catheter navigation aid and AR-guided evaluation for all 5 VS (left and right ventricles).

CNA-guided ($n = 50$)	VS1L	VS1R	VS2L	VS2R	VS3L	VS3R	VS4L	VS4R	VS5L	VS5R	ALL VS
Mean error (mm)	4.0	3.1	3.2	2.7	2.9	3.7	2.9	2.1	2.5	4.3	3.1
SD (mm)	2.8	1.9	1.0	1.6	2.3	1.6	1.9	1.3	2.2	0.7	1.8
Success rate (%)	100	100	100	100	100	100	100	100	100	80	98

The mean error was found to be 3.1 ± 1.8 mm with an overall success rate of 98%. A graphical representation of the experiment results is depicted in Figure 6.26.

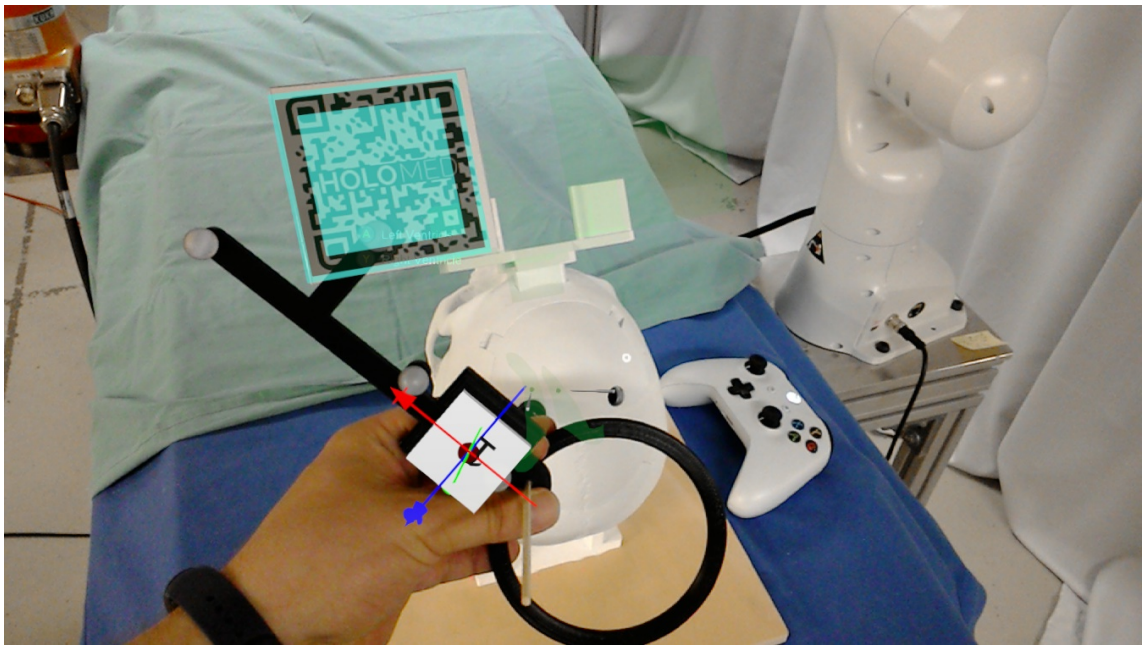


Figure 6.25.: Experimental Setup using the catheter navigation aid.

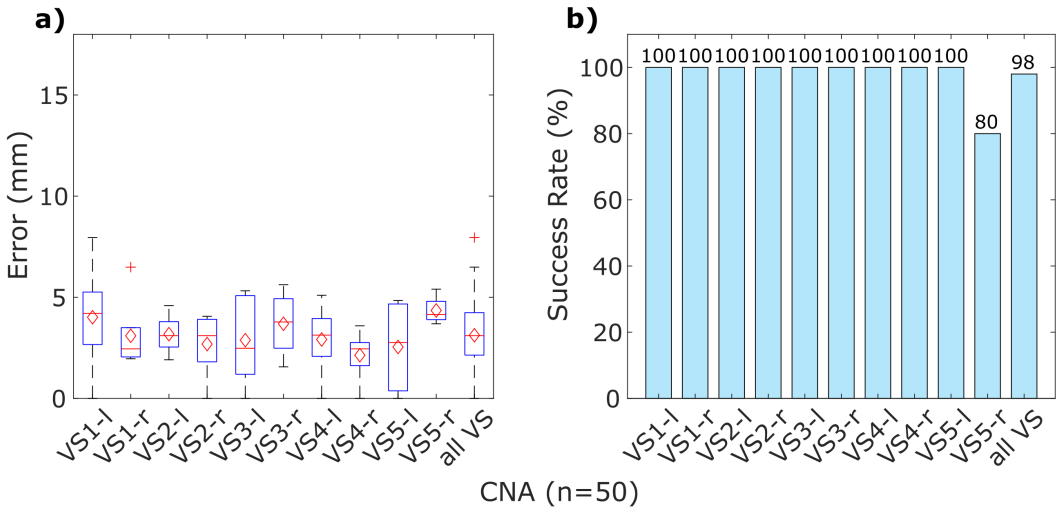


Figure 6.26.: CNA-guided experiments: a) Results of the catheter placements for all five ventricular systems, b) Success rate. Mean values are visualized as red diamond markers, outliers as red cross markers.

6.2.3.1. Discussion

Compared to the AR-guided experiments performed by medical experts the catheter navigation aided experiments yielded a 16.3% better success rate. The mean error was also significantly lower with 3.1 ± 1.8 mm. A limitation is that the navigation aid is manually manipulated, rendering it difficult to puncture the ventricles and holding the navigation aid at the same time. During the experiments no adjustment of the puncture trajectory was done. This way the experiments were comparable to the expert AR-guided evaluation.

The analysis of the puncture results showed that with the help of the CNA the puncture angle β could be minimised. With the assistance of the catheter navigation aid it is possible to adjust the real tube of the CNA with the virtual puncture trajectory. It is much more difficult to adjust angle α. It would be necessary to inspect the CNA from the side or above, which is not possible when manually manipulating the CNA. The infrared marker tracking is not robust and accurate enough to just rely on the visualized puncture angle.

6.3. Robot Assisted Intervention

In the third stage of intraoperative assistance, AR guidance is extended with a robot guiding the catheter navigation aid (RNA). The user can control the robot with a Microsoft XBox 360 controller to set the insertion angle and can then puncture the ventricular system in a consecutive step. This allows a much more stable placement of the catheter. This is to overcome the problems with holding the manually guided catheter navigation aid.

As a robot a Franka Emika Panda is used. All software components are implemented in the *Robot Operating System* (ROS) middleware as nodes.

6.3.1. Robot Control

The Panda robot is integrated into the OP:Sense environment and an altered version of the catheter navigation aid is mounted to it, as seen in Figure 6.27. The coordinate system of the robot is configured in a way that a user can intuitively control the robot when standing in front of the OR table. The user manually places the tip of the robot navigated aid in the center of the burr hole trepanation at the left or right Kocher's point. This is done via force controlled hand guidance of the Franka Panda robot. Then the mode is switched to the controller-based guidance and the user is able to teleoperate the robot with the XBox 360 controller. The used coordinate system is depicted in Figure 6.28. The angle of the RNA is intuitively controlled with the control sticks of the controller.

The puncture path is displayed similar as for the CNA to the user in the AR glasses. Additionally, the virtual puncture path is visualized over the navigation aid and over the patient model. Both trajectories must be aligned with each other to set the optimum insertion angle. After the robot has been brought into position, the user can then manually place the catheter through the RNA.

The Panda robot is controlled over the *libfranka* library on the client site. *Franka_ros* integrates the control functionality of the *libfranka* into ROS *Control* and represents the interface to ROS. See the robot control diagram in Figure 6.29.

Figure 6.27.: Franka Emika Panda with attached robot navigated aid.

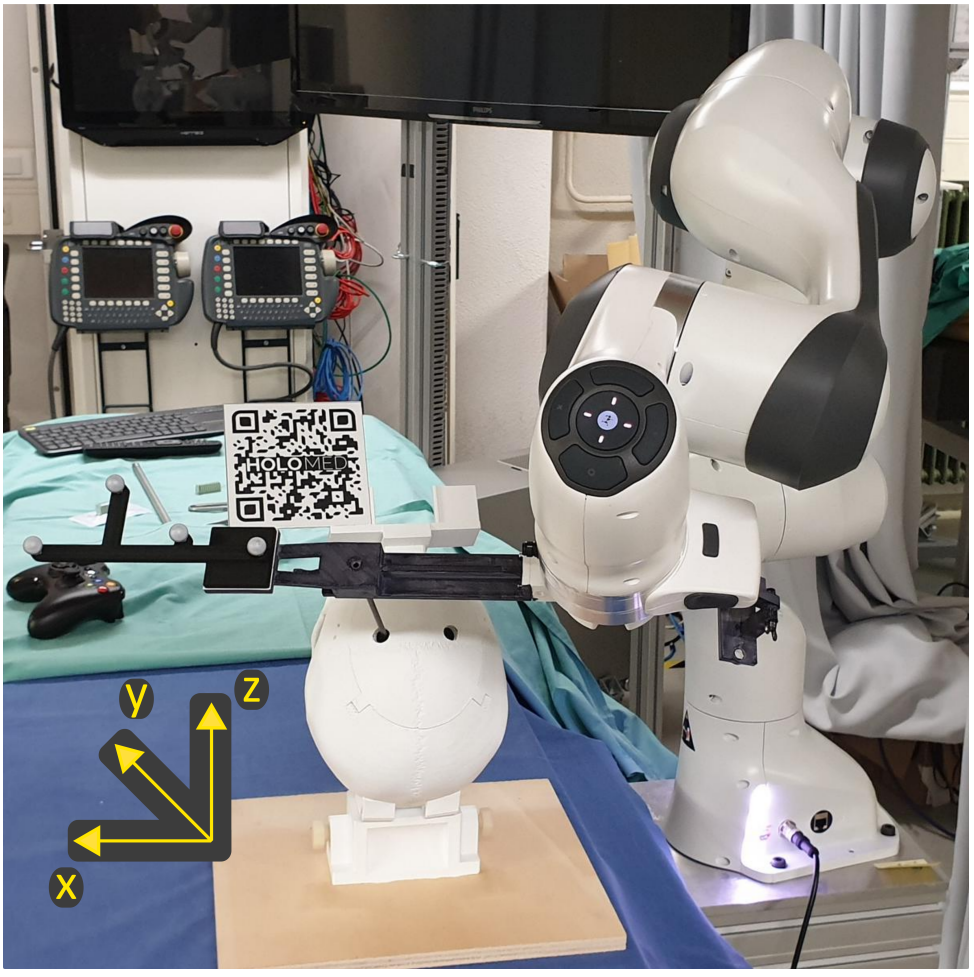


Figure 6.28.: Integrated Franka Panda robot in OP:Sense and used coordinate system.

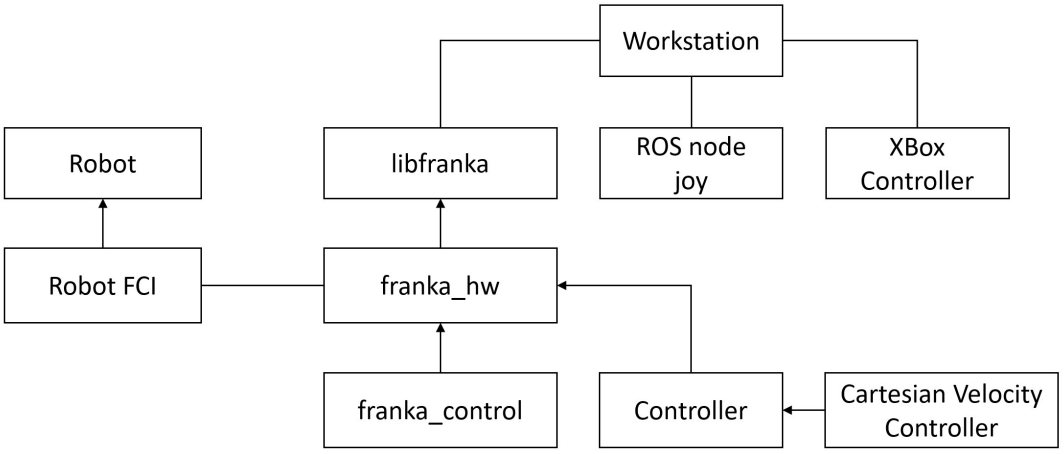


Figure 6.29.: Control diagram of the Franka Panda robot.

6.3.2. Experimental Validation

The experimental setup as presented in Section 6.1.2.1 is expanded with a robot for catheter navigation aid manipulation. The cathter navigation aid is therefore remodeled to make it possible to be mounted to the robot as depicted in Figure 6.27.

Two novice non-medical experts with technical background performed $n = 50$ ventricular punctures on the presented five ventricular systems. The results were again digitilized with a 3D line scanner.

6.3.3. Results

The results of the robot supported ventricular punctures are summarized in Table 6.13 and in Figure 6.30. The mean error, the standard deviation and the overall success rates are shown.

Table 6.13.: RNA-guided experiments: Mean error, standard deviation (SD) and overall success rate of the robot-guided evaluation for all 5 VS (left and right ventricles).

Robot-guided ($n = 50$)	VS1L	VS1R	VS2L	VS2R	VS3L	VS3R	VS4L	VS4R	VS5L	VS5R	ALL VS
Mean error (mm)	3.5	2.9	2.4	1.4	0.9	1.1	1.9	1.6	2.2	1.6	1.9
SD (mm)	0.9	0.7	0.8	1.8	1.0	0.8	1.0	1.0	1.5	0.3	1.2
Success rate (%)	100	100	100	100	100	100	100	100	100	100	100

It was determined that the mean error of the catheter placements was 1.9±1.2 mm. All ventricles were punctured correctly.

6.3.3.1. Discussion

The robot-guided catheter navigation aid yielded significantly better results than the AR-guided and CNA-guided experiments. However, one limitation is, that only a small amount of experiments were performed. Experience from the other experiments showed that most puncture results are in a certain range and do not differ much from subject to

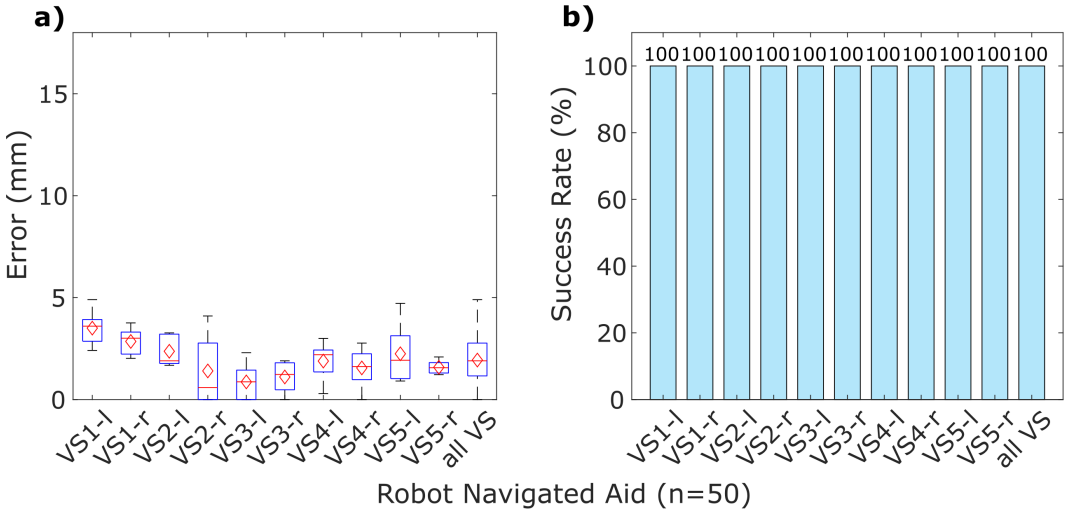


Figure 6.30.: RNA-guided experiments: a) Results of the catheter placements for all five ventricular systems, b) Success rate. Mean values are visualized as red diamond markers, outliers as red cross markers.

subject. All reference points were nearly perforated directly, as an error of 2-3 mm means that they are placed in near range to the annotated points.

The punctured ventricular systems are depicted in Figure 6.31.

Figure 6.31.: RNA-guided experiments: Results of the catheter placements. White modeling clay: reference points, green modeling clay: puncture results.

The RNA experiments are also comparable to the preclinical expert evaluations as the puncture path of the catheter could not be altered when inside the patient's head.

6.4. Overall System Error

The total system error consists of several error sources that add up and can be described as the catheter placement error as depicted in Figure 6.8. The error describes the deviation

on the surface of the ventricular system from the optimal points to the puncture results. It is challenging to determine which factors are responsible for the catheter placement error and to which extent. These factors can be identified, but not precisely separated from each other, because it is challenging to measure each individual factor.

One factor is the patient model to patient registration error ϵ_{reg} that was determined in Section 5.5. The controller-based registration yielded an accuracy of 2.71±1.18 mm. The HoloLens to wearer calibration error ϵ_{cal} cannot be determined directly because it is included in the registration error. It is determined by measuring the displacement after the registration. The calibration error could also change while using the HoloLens to some extent, because the HoleLens can move slightly on the wearer's head. The marker tracking error of the Vuforia marker tracking method is measured to be 0.31±0.38 mm. The reattachment accuracy of the marker system was determined to be 0.18±0.06 mm.

Additionally there are some other factors that can lead to a higher displacement. First there is the hologram fluctuation error ϵ_{fluc} that is describing the stability of the holograms present in the spatial map of the HoloLens and the ability to visualize the holograms on the visor of the HoloLens. This error is included in the registration error as it is not possible to separate it from the experienced displacement of the patient models. Then there is the marker system production error ϵ_{ms}, which can be up to 0.1 mm, but can be minimized with controlled production techniques. A sub error of this is the printing error of the used marker, or in case of the infrared spheres, also a construction error. The correctness of the construction mainly depends on the used printers (2D and 3D), while 3D printers have larger printing errors due to the higher complexity.

The registration error (ϵ_{reg}), the marker tracking error (ϵ_m) and the reattachment error (ϵ_{re}) determine the hologram visualization error ϵ_v.

$$\epsilon_v = \epsilon_{reg} + \epsilon_m + \epsilon_{re} \tag{6.3}$$

The calibration error ϵ_{cal} and the fluctuation error ϵ_{fluc} are included in the error sources as described above. The overall hologram visualization error ϵ_v was evaluated to be 3.2±1.62 mm.

The complete error equation is as follows:

$$\epsilon_{place} = \epsilon_{reg} + \epsilon_{cal} + \epsilon_m + \epsilon_{re} + \epsilon_{fluc} + \epsilon_{ms} + \epsilon_h \tag{6.4}$$

To all above mentioned error sources an additional error ϵ_h needs to be added, that describes the human error of manually placing a catheter to a non-visible target. Even when the registration would be perfect, so that ϵ_v is zero, a certain placement error occurs. The manual placement error is dependent on the skill of a surgeon and can not clearly be stated.

The catheter placement error (ϵ_{place}) has been investigated for all three stages of the AR guidance. The individual mean error values were presented in this chapter and are depicted in Table 6.14.

6.5. Discussion and Comparison of the Three Stages

In this chapter three consecutive stages for AR guidance were presented. In the basic augmented reality-based intraoperative assistance, the patient models are superimposed over the patient during an intervention. The patient is thereby fully covered and just the operation situs is visible. To be able to track the patient during the intervention, a marker system is used that is registered to the patient preoperatively. Four different registration methods were presented in Section 5.5. Their advantages and disadvantages were briefly discussed and a detailed evaluation was carried out. In the experimental evaluation a fixed registration of the marker system to the OpenHead phantom was used. This would comply with the standard of care, where fiducial markers are attached to the skull prior to a CT. The pointer could then be used to perform the registration.

When looking at the registration errors the most important displacement to be considered would be the one in x-direction of 1.36 ± 1.4 mm, when using the controller-based registration. The coordinate system is depicted in Figure 2.4. The error in y- and z-direction is less significant. The surgeon is looking directly along the insertion vector, the z-direction. A misalignment in z-direction renders holograms just a little bit larger or smaller and can therefore be neglected. The y-direction is following the elongated shape of the ventricular system. Therefore larger displacements in this direction still lead to a perforation of the ventricular system. The only issue can be that more unfavourable parts of the brain are penetrated. Also considering the additional error in x-direction the catheter still would have penetrated the surface of the ventricular system in nearly all cases of the preclinical experiments (1st stage) that were rated as a successful ventriculostomy.

One major limitation with the HoloLens 1 is the insufficient sensor quality which limits the tracking accuracy of infrared markers to 2-7 mm when worn by a user. Vuforia markers are trackable with higher accuracy but are also bigger in size (10 cm in the distance of 40 - 60 cm for neurosurgical interventions). The basic AR guidance was rated as very good by the participating eleven neurosurgeons as described in the evaluation in Section 6.1.3. Still, challenges remain with the hardware of the HoloLens 1 as the field of view was rated as too small or the visualization of the holograms are sometimes not sufficient and can lead to headaches or an unnatural way of perception.

In comparison to the standard of care, a nearly 10 percent higher success rate was reached, which clearly shows the potential of augmented reality support during an intervention. The major challenge in placing the catheter is to be able to get a good initial insertion vector, because the angle could not be changed once inside the brain of a patient or inside the agar of the phantom. If problems occur at this point, no adjustment is possible.

It is also very likely that the human is one of the biggest error sources during the procedure even if the holograms are displayed at the correct position. This was also the reason why a fixed position of the marker system was used in the experiments. This way the visualization error was reduced to a minimum, but still no 100% success rate was reached.

To counter the problem of a bad initial insertion angle, a catheter navigation aid (CNA) was constructed that guides the catheter after an angle is set.

On top of the CNA the insertion vector is displayed to the user so he or she can choose a proper insertion vector by minimizing the angle error. The CNA-guided experiments

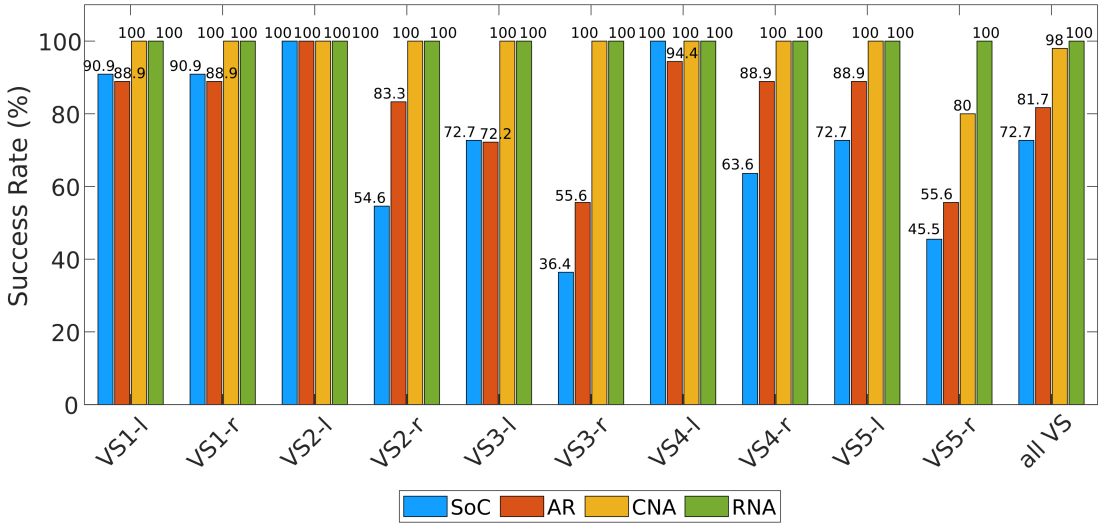


Figure 6.32.: All success rates of all ten ventricles for all three AR-guidance stages.

were evaluated by novice non-medical experts, but the experimental design is comparable to the design of the first stage preclinical experiments as it was not possible to change the insertion vector of the catheter inside the phantom. One problem with the CNA guidance was the manual manipulation of the CNA, which was error prone. A user needed to manipulate the device in space and simultaneously puncture the ventricular system. The manual hand-guided manipulation is not sufficiently steady considering the fact that even small deviations of the angle can lead to a big displacement error. A 16.3% higher success rate was reached in comparison to the AR-guided preclinical experiments. The mean error was also significantly lower with 3.1 ± 1.8 mm. One factor for the better success rate was a lower mean error on the x-axis, that could be detected.

In the final stage the catheter navigation aid is mounted to a robot to compensate the non steady handguided manipulation. Again better results were achieved. The mean error was determined to be 1.9 ± 1.2 mm, which is less then half the error of the preclinical experiments (AR-guided, stage one). All ventricles were punctured correctly. These are values determined in only 50 catheter placements, but the experience from the other experiments show that most of the punctures of one test series lie within the same error value range.

Accuracy values of all catheter placements of the three stages in comparison are visualized in Figure 6.33. Success rates of the catheter placements of all three stages are depicted in Figure 6.32.

As it can be seen better puncture results are reached over the three stages. The most difficult to puncture ventricular systems are the left and right ventricles of VS 3 and VS 5. Ventricular system VS 3 is shifted to the side for several mm and has very thin ventricles. For ventricle 3-right only 36.4% of catheters were placed correctly in the standard of care experiments. With AR-guidance 55.6% of punctures were rated successful and 100% when the CNA and RNA was used. The robot navigated aid yielded in all experiments a 100% success rate.

Ventricle 5-right was the most difficult one to puncture. This is due to the fact that it is very narrow in x-direction and has a steep surface. Even when the catheter is placed in a range of 2.5 mm, it is possible to puncture not correctly. When the AR-guidance of the CNA is used only slightly better results are achieved. With the RNA-guided placement,

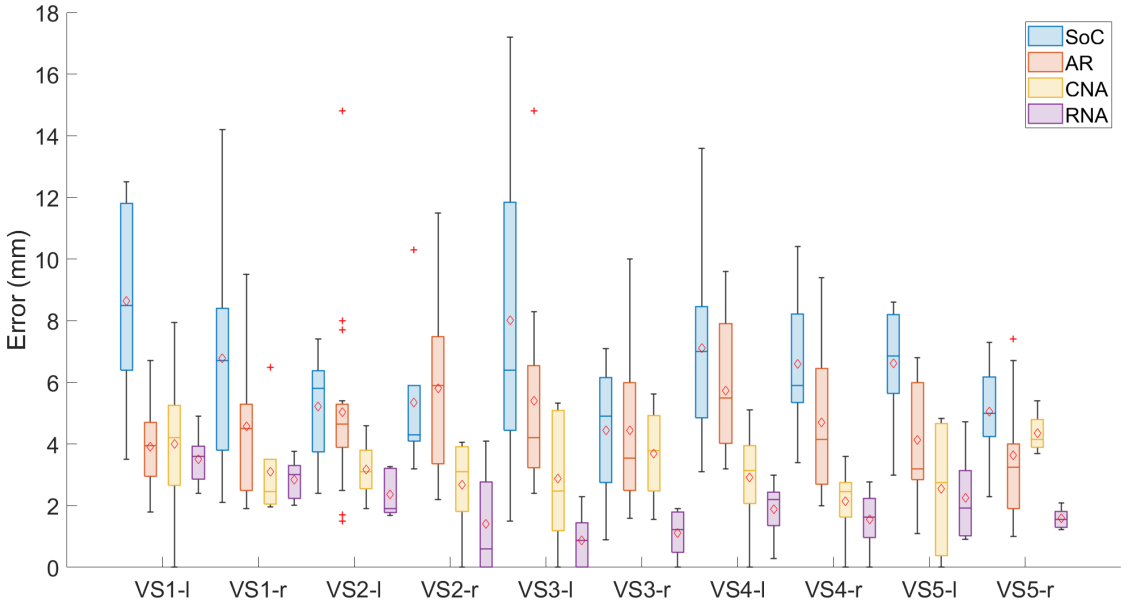


Figure 6.33.: Results of all accuracy values of all puncture experiments in comparison. Mean values are visualized as red diamond markers, outliers as red cross markers.

the ventricle was hit correctly. This shows that limitations regarding the AR-guidance and the CNA were adressed in the RNA-guided experiments.

All puncture errors are shown in Figure 6.33. The diagram shows the errors for each of the ten ventricles in relation to the puncture method used in comparison.

A comparison of the mean accuracies and overall success rates is depicted in Figure 6.34. It also shows that AR-guidance lead to a lower mean error and higher success rate. The comparison of the CNA and robot-guided experiments to the AR-guided is only conditionally possible as the AR-guided were performed by medical experts. It can be assumed that they generally reach a higher success rate than non-medical experts, due to their extensive training. Therefore, it is very likely that they would perform better when using the CNA.

The mean error results need to be carefully interpreted as they just reflect the accuracy of the catheter placements that were rated as correct. Placements that not perforated the ventricular system and were rated as misplaced were not included in the error calculated. This would not have been possible for the CNA and robot-guided experiments because no CT scanner was available.

6.6. Chapter Conclusion

In this chapter three consecutive stages of augmented reality support have been presented. In the first stage the generated patient models are superimposed over the patient to support a surgeon in the placement of the catheter. Scene registration is based on the presented marker system of Section 5.4. In the second stage, the challenge of a good initial puncture angle is addressed using a catheter navigation aid that is manually manipulated to reach an optimal insertion vector. To enable more stable manipulation of the catheter navigation

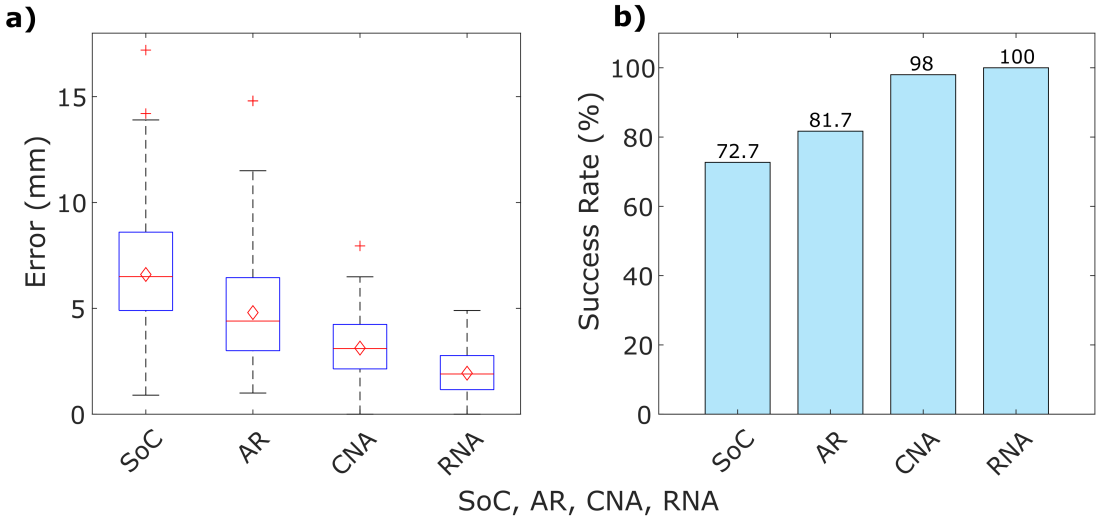


Figure 6.34.: Results of all puncture experiments in comparison. Mean values are visualized as red diamond markers, outliers as red cross markers.

aid, it was mounted to a robot that can be controlled by the surgeon. The robot can then be telemanipulated to set the correct insertion angle.

All accuracy values and success rates are depicted in Table 6.14. Throughout the consecutive stages higher success rates and lower mean error values were observed.

Table 6.14.: Accuracy, standard deviation (SD) and overall success rate for all three stages.

Guidance method	Mean error (mm)	SD (mm)	Success rate (%)
Standard of care ($n = 110$)	6.6	3.1	72.7
AR-guided ($n = 180$)	4.8	2.5	81.7
CNA-guided ($n = 50$)	3.1	1.8	98.0
Robot-guided ($n = 50$)	1.9	1.2	100.0

7. General Discussion

The different proposed methods of this work are discussed in detail in the corresponding sections. In this general discussion we will discuss the results of this work from a broader perspective and connect them to the established research questions of this work.

How can suitable medical knowledge be automatically retrieved and usefully applied to plan linear paths for the neurosurgical interventions ventriculostomies and biopsies?

In Chapter 4 methods are presented to perform automatic planning of the ventricular puncture as well as an expansion to additionally consider the individual risk structures of a patient. To plan ventriculostomies, first vital structures of the patient anatomy must be determined. The intervention is performed by opening the skull at the Kocher's points and puncture the ventricular system with a neurosurgical catheter. A target point inside the ventricular system must thereby be perforated that lies in the first quarter of the ventricular system.

The segmentation and the determination of the relevant points is medical knowledge used to create algorithms that apply this knowledge to automatically derive the puncture path. In Section 4.1, a method for automated segmentation of the ventricular system, the skull and the skin are proposed. This utilizes knowledge of the Hounsfield unit of cerebrospinal fluid and its distribution in the brain. The proposed method reaches a sufficiently good segmentation accuracy (F1 score) of over 98% when no pathologies are present. We also proposed an enhancement of our algorithm to segment blood clots inside the brain and the ventricular system, which further could be used for the drug treatment of these blood clots. The proposed method is also capable of segmenting the skull and the skin. Some patients suffer from a calcification of the plexus choroideus, which has the same Hounsfield Unit as bone, making it hard to segment the complete ventricular system. This leads to parts of the horn of the two big lateral ventricles that are not segmented correctly. However, for the ventricular puncture these structures are not of importance as they lie in the last quarter of the ventricular system that are irrelevant for the procedure. Estimating the borders of the ventricular system in the calcified area to derive the complete shape is a challenge. Advanced methods would be necessary to also segment these structures. Here machine learning methods could be applied. Due to the limited resolution of a CT scan it is also difficult to segment the third and fourth ventricular system because of its narrow structure. Very often even trained experts cannot find these structures in the CT data. However for the ventricular puncture the two big lateral ventricles are the most important ones. With the segmented structures, we are now able to derive the puncture path fully automated.

In Section 4.2, a method based on statistical shape models is proposed. Statistical shape models are a powerful tool to model all shape variations of a shape class. In our case, this is the skull and the ventricular sytem. Two statistical shape models were generated from sample data and a standard gaussian process. The optimal points were defined on the statistical shape models by using medical expert knowledge. In 93.9% of cases, a

valid trajectory could be derived. The largest challenge are misaligned patient models and different scanned parts of the patient head. However, the shape models showed a good robustness against the differing anatomy of patients. In the future the preprocessing could be automated to obtain correctly oriented and cutted patient scans. An alternative could be the use of different statistical shape models in relation to the scanned parts of the patient head. In the future, it must also be investigated how a valid puncture path can be derived when the patient's head is severely damaged or deformed, e.g. after an accident and the suffering of a traumatic brain injury. In such a case, it would be necessary to digitally reconstruct the head, derive valid Kocher's points and project these onto the deformed patient surface anatomy. Also specialized statistical shape models for various pathologies could be a solution to enhance fitting results.

The derivation of the optimal puncture trajectory is based on digitized medical expert knowledge but does not take the individual patient anatomy into account. Therefore, an expansion of the automated trajectory planning was presented in Section 4.3. Initially a risk-based path planning was shown for tumor biopsies, but this could be directly used also for the ventricular puncture. In the evaluation of the risk-based path planning lower risks were found when the risk projection was used. One tumor target was at the anatomic position of the ventricular system. The tumor moved the ventricular system to the side. In the puncture path was a high risk area down to the target, which led to a direct penetration of that risk area. This high risk area was found in the fMRI data of the dataset that was used. Common medical knowledge would have been that there is none such high risk area. Here, the proposed method clearly showed its advantage.

Another interesting idea is to derive a valid area with the statistical shape models for determination of the optimal Kocher's points. With the help of an additional assumption of the risk structures from all trajectories in the derived Kocher's area, the path with the lowest risk could be determined. One challenge with such an approach is that there is no standard patient set of image data that is acquired for every patient. Especially in emergency situations only a CT scan is available. The ventricular puncture is also performed as part of planned operations where a wide range of image data is available.

As a summary, medical knowledge can be used to create algorithms for automated puncture path determination with minimized risk for the patient. It is necessary in a first step that these knowledge is integrated in the algorithms, so that it is possible to further on access this knowledge without the need of an intervention of a surgeon. Once the algorithms are created, it is possible to retrieve this medical knowledge automatically.

Which algorithmic methods may be applied to perform an accurate neurosurgical scene registration?

In Chapter 5 methods were proposed to perform an accurate scene registration for neurosurgical interventions. All entities that are present in the HoloLens application need to be registered to the HoloLens coordinate system in order to process them in the system. In the proposed system these are the marker system that is registered to the patient, the pointer, the catheter navigation aid (CNA) and the robot navigated aid (RNA).

An extensive evaluation of several marker tracking methods was performed to compare them to each other. There is a large number of possible marker tracking methods that could be used to continuously derive the position of the patient. However, a lot of solutions are

not widely used, so that just a few possible candidates remain. In our case, these were ArUco, AprilTags, Vuforia and infrared marker tracking.

The best candidate was the ArUco tracking library as it enables fast and accurate tracking. In addition, the library is free to use. However, it was not possible to implement the solution on the HoloLens with acceptable performance. This shows that the HoloLens 1 is still a development and research device with room for improvement. Also, there are limited options available to investigate the limitations. One assumption is the access to the camera stream of the HoloLens as a bottle neck. The advantage when working with the HoloLens is that there is a large scientific community that is working with the device. Additionally, all solutions of the Unified Windows Platform (UWP) are available for HoloLens development. For example, it is easily possible to use the Microsoft XBox One controller with the HoloLens.

As an alternative it was decided to use Vuforia marker tracking. The tracking accuracy is high at a stable high framerate of 25 fps. A negative point here is that the library is not free to use in commercial settings. Perspectively, if this library is used in a commercial system, this could lead to high fees which would make the product much more expensive.

With the research mode of the HoloLens it became possible to implement infrared marker tracking on the device. High accuracy of under 1 mm were realized at a high framerate, when the HoloLens was mounted on a tripod and was not moved in space. A general problem of augmented reality became apparent. The accuracy dropped from 0.76 mm to over 2-7 mm when the HoloLens was worn by a person. It needs to be investigated in the future if this observation is due to the sensor quality or related to the registration of the different HoloLens streams to each other. This should be evaluated on the HoloLens 2.

When using augmented reality hard- and software components need to be integrated perfectly with each other to minimize possible error sources. The marker system, to which the markers are mounted supports the process change from the non-sterile to the sterile phase at a high reassembly accuracy.

Three manual and one automated registration methods for patient to marker registration were presented. Again, one limitation was the sensor quality of the HoloLens, that were recognized when registering with the pointer or when analyzing the point cloud for the automated registration via surface matching. The used algorithms are dependent on the quality of the sensors. Therefore, it can be stated that the maximum possible accuracy in relation to the sensor quality was reached.

An accurate scene registration can be achieved with an integration of hard- and software components. A robust marker system and a reliable marker that can be tracked with high accuracy is the base of the patient tracking. The sensor quality of modern augmented reality system is still a major challenge. The maximum possible tracking accuracy directly correlates to the sensor quality.

How can these methods and the results of the automated operation planning be used to implement an augmented reality-based intraoperative assistance?

Three different stages of AR-based assistance have been proposed in this work. The generated patient models and the automatically derived optimal trajectories are used in all stages to support a surgeon in the placement of the catheter.

One observation was that even when the patient models are superimposed at the correct position, a human is often not capable to place a catheter perfectly at the visualized target due to motoric limitations. This was one of the outcomes of the preliminary experiments performed by non-medical experts with technical background. The subjects punctured through air and a fixed marker system position was used, therefore an ideal registration. The puncture success rate was high with 91.25%, but a rather large error was recognized of 5.6±2.8 mm. The preclinical evaluation with medical experts yielded a lower success rate of 81.7% but a higher accuracy of 4.8 ± 2.5 mm. The experts punctured through agar gel to simulate brain matter, which required a good initial puncture angle, as it was not possible to change the angle after the catheter was inside the skull. This shows that manual manipulation by humans is a major source of error.

Control experiments were conducted by the expert neurosurgeons to obtain comparable results following the standard of care without using augmented reality. An accuracy of 6.6±3.1 mm and a success rate of 72.7% was reached. This clearly shows that a higher accuracy and success rate can be reached with AR-support.

The question arose how the human error can be minimized further to obtain even better results. One possibility is an extensive training of each surgeon to reach better results. We have investigated the performance of four participating surgeons in the AR-supported preclinical experiments. A steep learning curve was observed from 73 % to 95% success rate within three test series of n=100 for the first test series and n=40 for the second and third test series.

To get a better initial puncture angle, a catheter navigation aid (CNA) was introduced. Proof of concept experiments showed a success rate of 98%. One limitation with the infrared marker tracked CNA was the not ideal tracking accuracy due to sensor noise. In the future, it should be investigated if the tracking accuracy in newer versions of the HoloLens is better. A placement accuracy of 3.1±1.8 mm was observed. The major limitation of the CNA is the manual simultaneous manipulation of the device.

To optimize the manipulation, the CNA was mounted to a robot to enable a more stable puncture aid. The robot-guided CNA was telemanipulated by a human subject. Experiments showed that a higher accuracy of 1.9±1.2 mm was reached and a 100% success rate. Although these results must be confirmed in a realistic scenario with medical experts, the preclinical AR-guided evaluation showed comparable results to the non-medical experts study. Therefore, it can be assumed that medical experts would reach at least similar values.

An augmented reality-based intraoperative assistance can be achieved when all components are integrated in each other to support a user in the best possible way. Automation of the planning process enables a fast and reliable assistance that is accessable in under five minutes. It is thereby crucial that the planning is executed in a short period of time. The experiments showed a higher success rate and a lower mean error that can lead to a better outcome for the patient. Most importantly the system must be accepted by the users, the neurosurgeons. Only when a user feels comfortable using the system, higher accuracies can be achieved.

Key question: Can augmented reality help to optimize the accuracy of ventriculostomies and biopsies?

The experiments clearly show that a significant higher accuracy can be reached for the placement of catheters for the ventricular puncture in preclinical studies performed on

the OpenHead phantom when a surgeon is supported through augmented reality. The question arise if these results are transferable to a real clinical scenario and what possible challenges will be.

Li et al. [127] show similar results as for the preclinical evaluation presented in Section 6.1.1.3. No patient tracking was used. This confirms the results, although questions remain how patient movement was compensated.

The next step would be to perform cadaver or porcine ex-vivo studies to evaluate the accuracy of placements in a real clinical environment. The aspects that would be necessary to investigate are especially the marker system mounting and the registration to the subject. On the realistic Synbone phantom with attached silicone skin this worked sufficiently well, as the evaluation has shown. First preliminary experiments on human skin (not shown in this work) showed similar results. However, the registration of the marker system to the patient could be challenging. In general, the registration of the patient models to the Synbone phantom are comparable as the behaviour of the silicone is similar to the human skin. Problems occurred in the area of the jaw as it is not fixed and the mouth can be opened or closed.

This can lead to a different status of the patient model as for the real patient. The other parts of the skull are rigid and represent the real patient sufficiently enough due to the high resolution of modern CT scanners. When trying to register the models to the patient's head, this did not pose a problem.

Two preliminary experiments in a clinical environment were performed with two Synbone phantoms. The complete procedure was performed including the registration of the marker system to the Synbone phantom and mounting of the marker system. All four ventriculostomies were performed correctly. This provides a good initial assessment that the system can be translated into clinical reality and potentially increase accuracy and safety of future neurosurgical procedures.

8. Conclusion

8.1. Summary

In this work, the potential of augmented reality-based assistance for neurosurgical interventions is investigated on the example of the ventricular puncture. The system takes into account the automation of the complete surgical process from planning to surgical navigation. The placement of the catheter is still manually performed by a medical expert. For every patient, individual models are generated from CT image data, which includes the skin, the skull and the ventricular system. A segmentation accuracy of 98%, as determined by the F1 score, was achieved. An optimal trajectory is planned from these patient-individual models by determing the Kocher's points and the target points inside the ventricular system utilizing statistical shape models. A valid trajectory can be planned in 93.9% of cases.

As an expansion, path planning on the basis of patient-specific risk structures was shown on the example of tumor biopsies. The approach can easily be adopted to every linear puncture scenario. It was shown that a significantly lower risk can be reached when a risk visualization is shown to the user. When the planning according to the standard of care was used without risk visualization two risk structures were directly perforated. With our proposed method no risk structure was damaged.

A marker system has been developed to track the patient during an intervention. The most important feature is the support of the process change from the non-sterile to the sterile phase with very high accuracy of 0.18 ± 0.06 mm. Multiple registration methods have been proposed, from which the manual registration method via a XBox controller was the most accurate one.

Three consecutive stages of AR-guidance have been implemented and experimentally evaluated. For better comparison, experiments according to the standard of care were performed, which yielded a success rate of 72.7%. Preclinical AR-guided catheter placements reached significantly better results with 81.7% success rate. To overcome the shortcomings of manual catheter placement a catheter navigated aid was introduced which led to a 98% success rate. However, limitations with hand-guided manipulation of the aid remained. For this reason the aid was mounted to a robot and a 100% success rate was reached in a proof of concept experiment.

8.2. Outlook

Methods for automated path planning have been presented and were enhanced with a risk-based path planning approach. At the moment it is not the standard of care to gather a standard set of imaging data of a patient. This could greatly enhance the safety of

neurosurgical procedures in the future. A full set of preoperative imaging data would shift the automated path planning from generalizing medicine to a patient-specific treatment.

It is planned to combine the algorithms for the automated path planning with machine learning algorithms in the future to enable an automated preprocessing of the data, like scaling, rotating, translating and cutting. Certain pathologies can lead to parts of the ventricular system not being segmented correctly, like calcifications and blood clots. An expansion was presented to segment blood clots inside the ventricular system, but machine learning approaches could be more flexible and robust against certain pathologies. With modern deep learning techniques it would be possible to reconstruct the surface of the ventricular system or a deformed skull. The presented algorithms for automated path planning use statistical shape models to derive the optimal puncture points. This could also be done by a landmark detection based on neural networks. The statistical shape model itself could be enhanced with machine learning, e.g. to perform the fitting step with a neural network. The goal would be rather not to follow an end-to-end learning approach, but a combination of medical expert knowledge with powerful machine learning methods.

One limitation of this work is the hardware in form of the HoloLens 1 and its limited sensor quality. Therefore, tracking of infrared marker spheres and the registration with surface matching was limited. A further limitation is the small field of view and an unnatural way of seeing. Meanwhile, the second version of the HoloLens has been released. Better hardware could lead to much better results and need to be investigated in future research.

The HoloLens 1 had not the performance to run powerful machine learning algorithms. Therefore, they were not used in this work. In Appendix A, two works are presented that use neural networks to track the used catheter and the surgical staff. The goal would be to use these algorithms directly on the HoloLens to enable a scene analysis directly computed on the device. This promising approach would render it possible to be independent from any other system.

In this work, the medical experts place the catheter with AR support in three consecutive stages, but no tracking of the catheter is done. We have shown that this is possible with a semantic scene analysis by segmenting the camera frame. The algorithms were executed offline, because the performance on the HoloLens 1 was not sufficient. A second option would be to stream all data to a workstation in the OP:Sense environment. The algorithms are then used as a service. Considering all streams of the HoloLens one frame package can have the size of more than 3 MB. When investigating the whole workflow the network speed is still not high enough to reach an acceptable framerate.

The algorithms for semantic segmentation of the catheter run at 14 fps and the people tracking utilizing OpenPose at 21 fps on a high speed workstation. One challenge is also the high demands for processing of machine learning methods based on neural networks.

The here proposed methods could be also used in other scenarios like the treatment of tumors or deep brain stimulation. Further, the system could be used to train neurosurgeons to perform better and reach higher accuracy rates during surgery.

Bibliography

[1] Anforderungen an die Hygiene bei der Aufbereitung von Medizinprodukten. *Bundesgesundheitsblatt - Gesundheitsforschung - Gesundheitsschutz*, 55(10):1244–1310, October 2012.

[2] DIN EN 285:2016-05, Sterilisation- Dampf-Sterilisatoren- Groß-Sterilisatoren; Deutsche Fassung EN 285:2015. Technical report, Beuth Verlag GmbH, 2016. URL `https://www.beuth.de/de/-/-/237560921`. Visited on 06/07/2021.

[3] DIN EN 556-1 Berichtigung 1:2006-12, Sterilisation von Medizinprodukten- Anforderungen an Medizinprodukte, die als "STERIL" gekennzeichnet werden - Teil 1: Anforderungen an Medizinprodukte, die in der Endpackung sterilisiert wurden; Deutsche Fassung EN556-1:2001, Berichtigungen zu DIN EN 556-1:2002-03; Deutsche Fassung EN 556-1:2001/AC:2006. Technical report, Beuth Verlag GmbH, 2016. URL `https://www.beuth.de/de/-/-/94092737`. Visited on 01/06/2021.

[4] Daniel F Abawi, Joachim Bienwald, and Ralf Dorner. Accuracy in optical tracking with fiducial markers: an accuracy function for artoolkit. In *Third IEEE and ACM International symposium on mixed and augmented reality*, pages 260–261. IEEE, 2004.

[5] Yuichiro Abe, Shigenobu Sato, Koji Kato, Takahiko Hyakumachi, Yasushi Yanagibashi, Manabu Ito, and Kuniyoshi Abumi. A novel 3d guidance system using augmented reality for percutaneous vertebroplasty. *Journal of Neurosurgery: Spine*, 19(4):492–501, Oct 2013.

[6] Absolute Reports. Global Surgical Navigation Systems Market 2019: by Manufacturers, Regions, Type and Application, Forecast to 2024. Technical report, Absolute Reports, 2020. Visited on 10/05/2020.

[7] Mohammed H AlSharif, Mohamed Saad, Mohamed Siala, Hatem Boujemaa, Tarig Ballal, and Tareq Y Al-Naffouri. High accuracy acoustic estimation of multiple targets. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3534–3538. IEEE, 2018.

[8] John Amanatides and Andrew Woo. A fast voxel traversal algorithm for ray tracing. 1987.

[9] Amit Chourasia and Jan Klein. 2010 IEEE Visualization Contest, 2019. URL `http://sciviscontest.ieeevis.org/2010/`. Visited on 14/08/2021.

[10] B Armstrong, Thomas Verron, Lee A Heppe, Ravindra M Karonde, Jim Reynolds, and Karl Schmidt. Rgr-6d: low-cost, high-accuracy measurement of 6-dof pose from a single image. *Milwaukee (WI): Manuscript, University of Wisconsin*, 2007.

[11] Bradley Atcheson, Felix Heide, and Wolfgang Heidrich. Caltag: High precision fiducial markers for camera calibration. In *VMV*, volume 10, pages 41–48. Citeseer, 2010.

[12] Syed S. Azeem and T. C. Origitano. Ventricular catheter placement with a frameless neuronavigational system: A 1-year experience. *Operative Neurosurgery*, 60:243–48, 04 2007.

[13] Ehsan Azimi, Zhiyuan Niu, Maia Stiber, Nicholas Greene, Ruby Liu, Camilo Molina, Judy Huang, Chien-Ming Huang, and Peter Kazanzides. An interactive mixed reality platform for bedside surgical procedures. In Anne L. Martel, Purang Abolmaesumi, Danail Stoyanov, Diana Mateus, Maria A. Zuluaga, S. Kevin Zhou, Daniel Racoceanu, and Leo Joskowicz, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2020*, pages 65–75, Cham, 2020. Springer International Publishing.

[14] Ronald T. Azuma. A survey of augmented reality. *Presence: Teleoperators and Virtual Environments*, 6(4):355–385, Aug 1997.

[15] Andrej Babinec, Ladislav Jurišica, Peter Hubinskỳ, and František Duchoň. Visual localization of mobile robot using artificial markers. *Procedia Engineering*, 96: 1–9, 2014.

[16] Michael Bajura, Henry Fuchs, and Ryutarou Ohbuchi. Merging virtual objects with the real world. In *Proceedings of the 19th annual conference on Computer graphics and interactive techniques - SIGGRAPH '92*. ACM Press, 1992.

[17] Habib Y. Baluwala, Duane T. K. Malcolm, Jessica W. Y. Jor, Poul M. F. Nielsen, and Martyn P. Nash. Automatic landmark detection using statistical shape modelling and template matching. In Barry Doyle, Karol Miller, Adam Wittek, and Poul M.F. Nielsen, editors, *Computational Biomechanics for Medicine*, pages 75–82, Cham, 2015. Springer International Publishing.

[18] Joel S Beckett, Bilwaj Gaonkar, Diana Babayan, Justin Mathew, David McArthur, Noriko Salamon, Neil Martin, Isaac Yang, and Luke Macyszyn. Autonomous trajectory planning for external ventricular drain placement. *Operative Neurosurgery*, 15(4):433–439, 2018.

[19] Kimon Bekelis, Tarek A Radwan, Atman Desai, and David W Roberts. Frameless robotically targeted stereotactic brain biopsy: feasibility, diagnostic yield, and safety. *Journal of neurosurgery*, 116(5):1002–1006, 2012.

[20] Abdelkrim Belhaoua, Alexandre Kornmann, and Jean-Pierre Radoux. Accuracy analysis of an augmented reality system. In *2014 12th International Conference on Signal Processing (ICSP)*, pages 1169–1174. IEEE, 2014.

[21] Filippo Bergamasco, Andrea Albarelli, Emanuele Rodola, and Andrea Torsello. Rune-tag: A high accuracy fiducial marker with strong occlusion resilience. In *CVPR 2011*, pages 113–120. IEEE, 2011.

[22] Sylvain Bernhardt, Stéphane A. Nicolau, Luc Soler, and Christophe Doignon. The status of augmented reality in laparoscopic surgery as of 2016. *Medical Image Analysis*, 37:66–90, Apr 2017.

[23] Paul J Besl and Neil D McKay. Method for registration of 3-d shapes. In *Sensor fusion IV: control paradigms and data structures*, volume 1611, pages 586–606. International Society for Optics and Photonics, 1992.

[24] Mark Billinghurst, Adrian Clark, and Gun Lee. A survey of augmented reality. *Foundations and Trends® in Human–Computer Interaction*, 8(2-3):73–272, 2015.

[25] James Booth, Anastasios Roussos, Stefanos Zafeiriou, Allan Ponniah, and David Dunaway. A 3d morphable model learnt from 10,000 faces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5543–5552, 2016.

[26] Sanne M. B. I. Botden, Sonja N. Buzink, Marlies P. Schijven, and Jack J. Jakimowicz. ProMIS augmented reality training of laparoscopic procedures face validity. *Simulation in Healthcare: The Journal of the Society for Simulation in Healthcare*, 3(2):97–102, 2008.

[27] Jack E Bresenham. Algorithm for computer control of a digital plotter. *IBM Systems journal*, 4(1):25–30, 1965.

[28] Ellen JL Brunenberg, Anna Vilanova, Veerle Visser-Vandewalle, Yasin Temel, Linda Ackermans, Bram Platel, and Bart M ter Haar Romeny. Automatic trajectory planning for deep brain stimulation: a feasibility study. pages 584–592, 2007.

[29] Thorsten M Buzug. *Einführung in die Computertomographie: mathematisch-physikalische Grundlagen der Bildrekonstruktion*. Springer-Verlag, 2011.

[30] Silvain Bériault, Fahd Al Subaie, D. Louis Collins, Abbas F. Sadikot, and G. Bruce Pike. A multi-modal approach to computer-assisted deep brain stimulation trajectory planning. *International Journal of Computer Assisted Radiology and Surgery*, 7(5):687–704, 2012.

[31] John Canny. A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, (6):679–698, 1986.

[32] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Openpose: realtime multi-person 2d pose estimation using part affinity fields. *arXiv preprint arXiv:1812.08008*, 2018.

[33] Julie Carmigniani, Borko Furht, Marco Anisetti, Paolo Ceravolo, Ernesto Damiani, and Misa Ivkovic. Augmented reality technologies, systems and applications. *Multimedia Tools and Applications*, 51:341–377, 2010.

[34] Bruce Carse, Barry Meadows, Roy Bowers, and Philip Rowe. Affordable clinical gait analysis: An assessment of the marker tracking accuracy of a new low-cost optical 3d motion analysis system. *Physiotherapy*, 99(4):347–351, 2013.

[35] Hung-Li Jason Chen, Faramarz F Samavati, Mario Costa Sousa, and Joseph Ross Mitchell. Sketch-based volumetric seeded region growing. In *SBM*, pages 123–129, 2006.

[36] Wenan Chen, Rebecca Smith, Soo-Yeon Ji, Kevin R Ward, and Kayvan Najarian. Automated ventricular systems segmentation in brain ct images by combining low-level segmentation and high-level template matching. *BMC medical informatics and decision making*, 9(1), 2009.

[37] Yang Chen and Gérard Medioni. Object modelling by registration of multiple range images. *Image and vision computing*, 10(3):145–155, 1992.

[38] Zhi-Jian Chen, George T Gillies, William C Broaddus, Sujit S Prabhu, Helen Fillmore, Ryan M Mitchell, Frank D Corwin, and Panos P Fatouros. A realistic brain tissue phantom for intraparenchymal infusion studies. *Journal of neurosurgery*, 101(2):314–322, 2004.

[39] Amit Chourasia and Jan Klein. 2010 IEEE Visualization Contest, 2021. URL `http://sciviscontest.ieeevis.org/2010/`. Visited on 06/07/2021.

[40] Dimitrios Chytas, Michael-Alexander Malahias, and Vasileios S. Nikolaou. Augmented reality in orthopedics: Current state and future directions. *Frontiers in Surgery*, 6, Jun 2019.

[41] Slicer Open Source Community. *3D Slicer image computing platform*, 2021. URL `https://www.slicer.org`. Visited on 29/05/2021.

[42] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Active shape models - their training and application. *Comput. Vis. Image Underst.*, 61(1):38–59, Jan 1995.

[43] Novarad Corp. Novarad imaging software, 2021. URL `https://www.novarad.net/`. Visited on 29/05/2021.

[44] Carlos Correa and Kwan-Liu Ma. Size-based transfer functions: A new volume exploration technique. *IEEE transactions on visualization and computer graphics*, 14(6):1380–1387, 2008.

[45] Carlos Correa and Kwan-Liu Ma. The occlusion spectrum for volume classification and visualization. *IEEE Transactions on Visualization and Computer Graphics*, 15 (6):1465–1472, 2009.

[46] Carlos D Correa and Kwan-Liu Ma. Visibility-driven transfer functions. In *2009 IEEE Pacific Visualization Symposium*, pages 177–184. IEEE, 2009.

[47] Carlos D Correa and Kwan-Liu Ma. Visibility histograms and visibility-driven transfer functions. *IEEE Transactions on Visualization and Computer Graphics*, 17(2):192–204, 2010.

[48] Fabrizio Cutolo, Antonio Meola, Marina Carbone, Sara Sinceri, Federico Cagnazzo, Ennio Denaro, Nicola Esposito, Mauro Ferrari, and Vincenzo Ferrari. A new head-mounted display-based augmented reality system in neurosurgical oncology: a study on phantom. *Computer Assisted Surgery*, 22(1):39–53, Jan 2017.

[49] Mohammad A Dabbah, Sean Murphy, Hippolyte Pello, Romain Courbon, Erin Beveridge, Stewart Wiseman, Daniel Wyeth, and Ian Poole. Detection and location of 127 anatomical landmarks in diverse ct datasets. In *Medical Imaging 2014: Image Processing*, volume 9034. International Society for Optics and Photonics, 2014.

[50] Alejandro de León-Cuevas, Saúl Tovar-Arriaga, Arturo González-Gutiérrez, and Marco Antonio Aceves-Fernández. Risk map generation for keyhole neurosurgery using fuzzy logic for trajectory evaluation. *Neurocomputing*, 233:81–89, 2017.

[51] E De Momi, C Caborni, F Cardinale, G Casaceli, L Castana, M Cossu, R Mai, F Gozzo, S Francione, L Tassi, G Lo Russo, L Antiga, and G Ferrigno. Multi-trajectories automatic planner for StereoElectroEncephaloGraphy (SEEG). *International Journal of Computer Assisted Radiology and Surgery*, 9(6):1087–1097, 2014.

[52] Joseph DeGol, Timothy Bretl, and Derek Hoiem. Chromatag: A colored marker and fast detection algorithm. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1472–1481, 2017.

[53] Weiwei Deng, Fang Li, Manning Wang, and Zhijian Song. Easy-to-use augmented reality neuronavigation using a wireless tablet PC. *Stereotactic and Functional Neurosurgery*, 92(1):17–24, 2014.

[54] Data Format Working Group (dfwg). Definition of nifti File Format, 2021. URL `https://brainder.org/2012/09/23/the-nifti-file-format/`. Visited on 14/08/2021.

[55] S. Diepenbrock, J. Prassni, F. Lindemann, H. Bothe, and T. Ropinski. 2010 IEEE Visualization Contest Winner: Interactive Planning for Brain Tumor Resections. *IEEE Computer Graphics and Applications*, 31(5):6–13, 2011.

[56] David H Douglas and Thomas K Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: the international journal for geographic information and geovisualization*, 10(2): 112–122, 1973.

[57] Robert A Drebin, Loren Carpenter, and Pat Hanrahan. Volume rendering. *ACM Siggraph Computer Graphics*, 22(4):65–74, 1988.

[58] George H. Dunteman. *Principal component analysis*. Sage, 1989.

[59] Marco Esposito, Benjamin Busam, Christoph Hennersperger, Julia Rackerseder, An Lu, Nassir Navab, and Benjamin Frisch. Cooperative robotic gamma imaging: Enhancing us-guided needle biopsy. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 611–618. Springer, 2015.

[60] C. Essert, C. Haegelen, F. Lalys, A. Abadie, and P. Jannin. Automatic computation of electrode trajectories for Deep Brain Stimulation: a hybrid symbolic and numerical approach. *International journal of computer assisted radiology and surgery*, 7 (4):517–532, 2012.

[61] R. Estaña. *Entwicklung und Test eines moiré-basierten hochgenauen Positionserfassungssystems*. Logos-Verlag, 2006.

[62] Andriy Fedorov, Reinhard Beichel, Jayashree Kalpathy-Cramer, Julien Finet, Jean-Christophe Fillion-Robin, Sonia Pujol, Christian Bauer, Dominique Jennings, Fiona Fennessy, Milan Sonka, et al. 3d slicer as an image computing platform for the quantitative imaging network. *Magnetic resonance imaging*, 30(9):1323–1341, 2012.

[63] Pedro F Felzenszwalb and Daniel P Huttenlocher. Efficient graph-based image segmentation. *International journal of computer vision*, 59(2):167–181, 2004.

[64] Mark Fiala. Artag, a fiducial marker system using digital techniques. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 590–596. IEEE, 2005.

[65] Andrew W Fitzgibbon. Robust registration of 2d and 3d point sets. *Image and vision computing*, 21(13-14):1145–1153, 2003.

[66] Paul M Foreman, Philipp Hendrix, Christoph J Griessenauer, Philip GR Schmalz, and Mark R Harrigan. External ventricular drain placement in the intensive care unit versus operating room: evaluation of complications and accuracy. *Clinical neurology and neurosurgery*, 128:94–100, 2015.

[67] Alejandro F. Frangi, Wiro J. Niessen, Koen L. Vincken, and Max A. Viergever. Multiscale Vessel Enhancement Filtering. In William M. Wells, Alan Colchester, and Scott Delp, editors, *Medical Image Computing and Computer-Assisted Intervention - MICCAI'98*, volume 1496 of *Lecture Notes in Computer Science*, pages 130–137. Springer, Berlin, Heidelberg, 1998.

[68] T. Frantz, B. Jansen, J. Duerinck, and J. Vandemeulebroucke. Augmenting microsoft's hololens with vuforia tracking for neuronavigation. *Healthcare Technology Letters*, 5(5):221–225, 2018.

[69] Sergio Garrido-Jurado, Rafael Muñoz-Salinas, Francisco José Madrid-Cuevas, and Manuel Jesús Marín-Jiménez. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition*, 47(6):2280–2292, 2014.

[70] Andre Gaschler, Maximilian Springer, Markus Rickert, and Alois Knoll. Intuitive robot tasks with augmented reality and virtual obstacles. In *IEEE International Conference on Robotics and Automation (ICRA)*, June 2014.

[71] Tobias Gass, Gabor Szekely, and Orcun Goksel. Multi-atlas segmentation and landmark localization in images with large field of view. In Bjoern Menze, Georg Langs, Albert Montillo, Michael Kelm, Henning Müller, Shaoting Zhang, Weidong (Tom) Cai, and Dimitris Metaxas, editors, *Medical Computer Vision: Algorithms for Big Data*, pages 171–180, Cham, 2014. Springer International Publishing.

[72] Oliver P. Gautschi, N. R. Smoll, M. Kotowski, B. Schatlo, M. Tosic, B. Stimec, J. Fasel, K. Schaller, and P. Bijlenga. Non-assisted versus neuro-navigated and xperct-guided external ventricular catheter placement: a comparative cadaver study. *Acta Neurochirurgica*, 156(4):777–785, Apr 2014.

[73] Miltiadis Georgiopoulos, John Ellul, Elisabeth Chroni, and Constantine Constantoyannis. Efficacy, safety, and duration of a frameless fiducial-less brain biopsy versus frame-based stereotactic biopsy: a prospective randomized study. *Journal of Neurological Surgery Part A: Central European Neurosurgery*, 79(01):031–038, 2018.

[74] Ian J Gerard, Marta Kersten-Oertel, Kevin Petrecca, Denis Sirhan, Jeffery A Hall, and D Louis Collins. Brain shift in neuronavigation of brain tumors: a review. *Medical image analysis*, 35:403–420, 2017.

[75] F. Ghesu, B. Georgescu, Y. Zheng, S. Grbic, A. Maier, J. Hornegger, and D. Comaniciu. Multi-scale deep reinforcement learning for real-time 3d-landmark detection in ct scans. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(1):176–189, 2019.

[76] Jacob T. Gibby, Samuel A. Swenson, Steve Cvetko, Raj Rao, and Ramin Javan. Head-mounted display augmented reality to guide pedicle screw placement utilizing computed tomography. *International Journal of Computer Assisted Radiology and Surgery*, 14(3):525–535, Jun 2018.

[77] Ziv Gil, Vitaly Siomin, Liana Beni-Adani, Liat Sira, and Shlomi Constantini. Ventricular catheter placement in children with hydrocephalus and small ventricles: the use of a frameless neuronavigation system. *Child's Nervous System*, 18(1):26–29, Feb 2002.

[78] Alexandra J. Golby. *Image-Guided Neurosurgery*. Elsevier Science, Amsterdam, 2015.

[79] Suk-Woo Ha, Michael Koller, and Gerald Göllner. Sterilisation. In Erich Wintermantel and Suk-Woo Ha, editors, *Medizintechnik: Life Science Engineering*, pages 113–125. Springer, Berlin, Heidelberg, 2009.

[80] E.M. Haacke, S. Mittal, Z. Wu, J. Neelavalli, and Y.-C.N. Cheng. Susceptibility-weighted imaging: Technical aspects and clinical applications, part 1. *American Journal of Neuroradiology*, 30(1):19–30, 2009. URL `http://www.ajnr.org/content/30/1/19`.

[81] Felix G Hamza-Lup, Jannick P Rolland, and Charles Hughes. A distributed augmented reality system for medical training and simulation. *arXiv preprint arXiv:1811.12815*, 2018.

[82] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2 edition, 2004.

[83] Peter Hastreiter, Christof Rezk-Salama, Grzegorz Soza, Michael Bauer, Günther Greiner, Rudolf Fahlbusch, Oliver Ganslandt, and Christopher Nimsky. Strategies for brain shift evaluation. *Medical image analysis*, 8(4):447–464, 2004.

[84] P. C. Herghelegiu, V. Manta, R. Perin, S. Bruckner, and E. Gröller. Biopsy Planner – Visual Analysis for Needle Pathway Planning in Deep Seated Brain Tumor Biopsy. *Computer Graphics Forum*, 31(3pt2):1085–1094, 2012.

[85] Di-hui Hong, Gang-min Ning, Ting Zhao, Mu Zhang, and Xiaoxiang Zheng. Method of normal estimation based on approximation for visualization. *Journal of Electronic Imaging*, 12(3):470–477, 2003.

[86] Chung-Ching Hsia, Yuan-Hao CHen, Hao-Yiang Wu, and Ming-Ying Liu. The misplacement of external ventricular drain by freehand method in emergent neurosurgery. *Acta Neurol Belg*, 111:22–28, 2011.

[87] Runzhen Huang and Kwan-Liu Ma. Rgvis: Region growing based techniques for volume visualization. In *11th Pacific Conference onComputer Graphics and Applications, 2003. Proceedings.*, pages 355–363. IEEE, 2003.

[88] David R Huyette, Benjamin J Turnbow, Christian Kaufman, Dale F Vaslow, Benjamin B Whiting, and Michael Y Oh. Accuracy of the freehand pass technique for ventriculostomy catheter placement: retrospective assessment using computed tomography scans. *Journal of neurosurgery*, 108(1):88–91, 2008.

[89] Patrick Hübner, Kate Clintworth, Qingyi Liu, Martin Weinmann, and Sven Wursthorn. Evaluation of hololens tracking and depth sensing for indoor mapping applications. *Sensors*, 20(4):1021, Feb 2020.

[90] D. Inoue, B. Cho, M. Mori, Y. Kikkawa, T. Amano, A. Nakamizo, K. Yoshimoto, M. Mizoguchi, M. Tomikawa, J. Hong, T. Sasaki, and M. Hashizume. Preliminary study on the clinical application of augmented reality neuronavigation. *Journal of Neurological Surgery Part A: Central European Neurosurgery*, 74(02):071–076, Feb 2013.

[91] Siti Salmi Jamali, Mohd Fairuz Shiratuddin, Kok Wai Wong, and Charlotte L. Oskam. Utilising mobile-augmented reality for learning human anatomy. *Procedia - Social and Behavioral Sciences*, 197:659–668, Jul 2015.

[92] Palle ET Jorgensen and Myung-Sin Song. Entropy encoding, hilbert space, and karhunen-loève transforms. *Journal of Mathematical Physics*, 48(10), 2007.

[93] David Joseph Tan, Federico Tombari, Slobodan Ilic, and Nassir Navab. A versatile learning-based 3d temporal tracker: Scalable, robust, online. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 693–701, 2015.

[94] Udaya K Kakarla, Louis J Kim, Steven W Chang, Nicholas Theodore, and Robert F Spetzler. Safety and accuracy of bedside external ventricular drain placement. *Operative Neurosurgery*, 63(suppl_1):162–167, 2008.

[95] Willi A Kalender. X-ray computed tomography. *Physics in Medicine and Biology*, 51(13):29–43, Jun 2006.

[96] Carolien Kamphuis, Esther Barsom, Marlies Schijven, and Noor Christoph. Augmented reality in medical education? *Perspectives on Medical Education*, 3(4): 300–311, Jan 2014.

[97] Xin Kang, Mahdi Azizian, Emmanuel Wilson, Kyle Wu, Aaron D. Martin, Timothy D. Kane, Craig A. Peters, Kevin Cleary, and Raj Shekhar. Stereoscopic augmented reality for laparoscopic surgery. *Surgical Endoscopy*, 28(7):2227–2235, Feb 2014.

[98] Kari Karhunen. *Über lineare Methoden in der Wahrscheinlichkeitsrechnung*, volume 37. Universitat Helsinki, 1947.

[99] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, Jan 1988.

[100] Hirokazu Kato and Mark Billinghurst. Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In *Proceedings 2nd IEEE and ACM International Workshop on Augmented Reality (IWAR'99)*, pages 85–94. IEEE, 1999.

[101] Takakazu Kawamata, Hiroshi Iseki, Takao Shibasaki, and Tomokatsu Hori. Endoscopic augmented reality navigation system for endonasal transsphenoidal surgery to treat pituitary tumors: Technical note. *Neurosurgery*, 50(6):1393–1397, Jun 2002.

[102] A.P. King, P.J. Edwards, C.R. Maurer Jr., D.A. de Cunha, D.J. Hawkes, D.L.G. Hill, R.P. Gaston, M.R. Fenlon, A.J. Strong, C.L. Chandler, A. Richards, and M.J. Gleeson. A system for microscope-assisted guided interventions. *Stereotactic and Functional Neurosurgery*, 72(2-4):107–111, 1999.

[103] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[104] Gabriel Kiss, Cameron Lowell Palmer, and Hans Torp. Patient adapted augmented reality system for real-time echocardiographic applications. In *Workshop on Augmented Environments for Computer-Assisted Interventions*, pages 145–154. Springer, 2015.

[105] Inc. Kitware. Insight toolkit, 2021. URL `https://itk.org/`. Visited on 14/08/2021.

[106] Inc. Kitware. Visualization toolkit, 2021. URL `https://vtk.org/`. Visited on 14/08/2021.

[107] Jens Kleesiek, Gregor Urban, Alexander Hubert, Daniel Schwarz, Klaus Maier-Hein, Martin Bendszus, and Armin Biller. Deep mri brain extraction: A 3d convolutional neural network for skull stripping. *NeuroImage*, 129:460–469, 2016. URL `https://www.sciencedirect.com/science/article/pii/S1053811916000306`.

[108] Joe Kniss, Gordon Kindlmann, and Charles Hansen. Multidimensional transfer functions for interactive volume rendering. *IEEE Transactions on visualization and computer graphics*, 8(3):270–285, 2002.

[109] Ralf A. Kockro, Yeo Tseng Tsai, Ivan Ng, Peter Hwang, Chuangui Zhu, Kusuma Agusanto, Liang Xiao Hong, and Luis Serra. DEX-RAY. *Neurosurgery*, 65(4): 795–808, Oct 2009.

[110] PieterL Kubben and RemirS. N Sinlae. Feasibility of using a low-cost head-mounted augmented reality device in the operating room. *Surgical Neurology International*, 10(1):26, 2019.

[111] Christian Kunz, Michal Hlaváč, Max Schneider, David Puljiz, Steffen Peikert, and Bjoern Hein. A system for augmented reality guided ventricular puncture using a HoloLens: Design, implementation and initial evaluation. In *CURAC 2018 - Tagungsband : 17. Jahrestagung der Deutschen Gesellschaft für Computer- und Roboterassistierte Chirurgie (CURAC), Leipzig, 13 – 15 September 2018*, pages 132–137. Universität Leipzig, 2018.

[112] Christian Kunz, Vera Genten, Pascal Meissner, and Bjoern Hein. Metric-based evaluation of fiducial markers for medical procedures. In *Medical imaging 2019: Image-guided Procedures, Robotic Interventions, and Modeling, San Diego, CA, February 16-21, 2019. Ed.: B. Fei*, Proceedings of SPIE, page 97. Society of Photo-optical Instrumentation Engineers (SPIE), 2019.

[113] Christian Kunz, Pit Henrich, Paul Maria Scheikl, Michal Hlaváč, Max Schneider, Woern Heinz, Bjoern Hein, and Franziska Mathis-Ullrich. Fast Volumetric Auto-Segmentation of Head CT Images in Emergency Situations for Ventricular Punctures. In *CURAC 2019, Reutlingen*, pages 41–46, 2019.

[114] Christian Kunz, Max Schneider, Michal Hlaváč, Dominik Zenth, Christian Pylatiuk, and Bjoern Hein. Augmented reality-assisted ventricular puncture with marker-based scene registration. *International Journal of Computer assisted Radiology and Surgery*, 14(Suppl. 1):S151–S152, 2019. Springer.

[115] Christian Kunz, Franziska Mathis-Ullrich, and Bjoern Hein. Augmented reality in the operating room for neurosurgical interventions. In *The Upper-Rhine Artificial Intelligence Symposium, 13th may, Karlsruhe*, 2020.

[116] Christian Kunz, Paulina Maurer, Fabian Kees, Pit Henrich, Christian Marzi, Michal Hlaváč, Max Schneider, and Franziska Mathis-Ullrich. Infrared marker tracking with the HoloLens for neurosurgical interventions. *Current Directions in Biomedical Engineering*, 6(1), 2020.

[117] Christian Kunz, Maximilian Gerst, Pit Henrich, and Franziska Mathis-Ullrich. Multimodal risk-map for navigation planning in neurosurgical interventions. *New Trends in Medical and Service Robotics*, 93, 2021.

[118] Christian Kunz, Maximilian Gerst, Pit Henrich, Max Schneider, Michal Hlaváč, Andrej Pala, and Franziska Mathis-Ullrich. Multimodal Risk-based Path Planning for Neurosurgical Interventions. *Journal of Medical Devices*, Jan 2021.

[119] Christian Kunz, Michal Hlaváč, Max Schneider, Andrej Pala, Pit Henrich, Birgit Jickeli, Heinz Woern, Bjoern Hein, Rainer Wirtz, and Franziska Mathis-Ullrich. Autonomous planning and intraoperative augmented reality navigation for neurosurgery. *IEEE Transactions on Medical Robotics and Bionics*, 2021.

[120] Michael H Kurniawan, Suharjito, Diana, and Gunawan Witjaksono. Human anatomy learning systems using augmented reality on mobile application. *Procedia Computer Science*, 135:80–88, 2018.

[121] Mauro Sylos Labini, Christina Gsaxner, Antonio Pepe, Juergen Wallner, Jan Egger, and Vitoantonio Bevilacqua. Depth-awareness in a system for mixed-reality aided surgical procedures. In *International Conference on Intelligent Computing*, pages 716–726. Springer, 2019.

[122] Vasileios Lahanas, Constantinos Loukas, Nikolaos Smailis, and Evangelos Georgiou. A novel augmented reality simulator for skills assessment in minimal invasive surgery. *Surgical Endoscopy*, 29(8):2224–2234, Oct 2014.

[123] Shouren Lan, Lisheng Wang, Yipeng Song, Yu-ping Wang, Liping Yao, Kun Sun, Bin Xia, and Zongben Xu. Improving separability of structures with similar attributes in 2d transfer function design. *IEEE transactions on visualization and computer graphics*, 23(5):1546–1560, 2016.

[124] Lung W. Lau, Xinyang Liu, William Plishker, Karun Sharma, Raj Shekhar, and Timothy D. Kane. Laparoscopic liver resection with augmented reality: A preclinical experience. *Journal of Laparoendoscopic & Advanced Surgical Techniques*, 29(1):88–93, Jan 2019.

[125] Denis Le Bihan, Jean-François Mangin, Cyril Poupon, Chris A Clark, Sabina Pappata, Nicolas Molko, and Hughes Chabriat. Diffusion tensor imaging: concepts and applications. *Journal of Magnetic Resonance Imaging: An Official Journal of the International Society for Magnetic Resonance in Medicine*, 13(4):534–546, 2001.

[126] Marc Levoy. Display of surfaces from volume data. *IEEE Computer graphics and Applications*, 8(3):29–37, 1988.

[127] Ye Li, Xiaolei Chen, Ning Wang, Wenyao Zhang, Dawei Li, Lei Zhang, Xin Qu, Weitao Cheng, Yueqiao Xu, Wenjin Chen, et al. A wearable mixed-reality holographic computer for guiding external ventricular drain insertion at the bedside. *Journal of neurosurgery*, 131(5):1599–1606, 2018.

[128] Christopher RP Lind, Amy MC Tsai, Christina J Lind, and Andrew JJ Law. Ventricular catheter placement accuracy in non-stereotactic shunt surgery for hydrocephalus. *Journal of Clinical Neuroscience*, 16(7):918–920, 2009.

[129] He Liu, Edouard Auvinet, Joshua Giles, and Ferdinando Rodriguez y Baena. Augmented reality based navigation for computer assisted hip resurfacing: A proof of concept study. *Annals of Biomedical Engineering*, 46(10):1595–1605, May 2018.

[130] Laurent J Livermore, Ruichong Ma, Stana Bojanic, and Erlick AC Pereira. Yield and complications of frame-based and frameless stereotactic brain biopsy–the value of intra-operative histological analysis. *British journal of neurosurgery*, 28(5):637–644, 2014.

[131] M Loève. Chapter 27. concept of conditioning. *Loève, M. Probability Theory*, 2: 3–12, 1978.

[132] S Scott Lollis and David W Roberts. Robotic catheter ventriculostomy: feasibility, efficacy, and implications. *Journal of neurosurgery*, 108(2):269–274, 2008.

[133] Alberto López-Cerón and José M Cañas. Accuracy analysis of marker-based 3d visual localization. In *XXXVII Jornadas de Automatica Workshop*, page 8, 2016.

[134] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. *ACM siggraph computer graphics*, 21(4):163–169, 1987.

[135] Yi Lu, Cecil Yeung, Alireza Radmanesh, Robert Wiemann, Peter M Black, and Alexandra J Golby. Comparative effectiveness of frame-based, frameless, and intraoperative magnetic resonance imaging–guided brain biopsy techniques. *World neurosurgery*, 83(3):261–268, 2015.

[136] Marcel Luethi and Ghazi Bouabene. *Statistical Shape Models FutureLearn web course*, 2021. URL https://www.futurelearn.com/courses/statistical-shape-modelling. Visited on 10/05/2021.

[137] Alexander Selvikvåg Lundervold and Arvid Lundervold. An overview of deep learning in medical imaging focusing on mri. *Zeitschrift für Medizinische Physik*, 29(2):102–127, 2019.

[138] Marcel Lüthi, Thomas Gerig, Christoph Jud, and Thomas Vetter. Gaussian process morphable models. *IEEE transactions on pattern analysis and machine intelligence*, 40(8):1860–1873, 2017.

[139] Lüthi, Marcel. *A Machine Learning Approach to Statistical Shape Models with Application to Medical Image Analysis*. PhD thesis, University of Basel, 2010. URL https://edoc.unibas.ch/1194/1/thesis-el.pdf.

[140] Longfei Ma, Zhe Zhao, Fang Chen, Boyu Zhang, Ligong Fu, and Hongen Liao. Augmented reality surgical navigation with ultrasound-assisted registration for pedicle screw placement: a pilot study. *International Journal of Computer Assisted Radiology and Surgery*, 12(12):2205–2215, Aug 2017.

[141] Meng Ma, Pascal Fallavollita, Ina Seelbach, Anna Maria Von Der Heide, Ekkehard Euler, Jens Waschke, and Nassir Navab. Personalized augmented reality for anatomy education. *Clinical Anatomy*, 29(4):446–453, Dec 2015.

[142] Shuaiyifan Ma and Zijian Zhao. A new method of surgical tracking system based on fiducial marker. In *Annual Conference on Medical Image Understanding and Analysis*, pages 886–896. Springer, 2017.

[143] Nader Mahmoud, Óscar G. Grasa, Stéphane A. Nicolau, Christophe Doignon, Luc Soler, Jacques Marescaux, and J. M. M. Montiel. On-patient see-through augmented reality based on visual SLAM. *International Journal of Computer Assisted Radiology and Surgery*, 12(1):1–11, Jul 2016.

[144] Andreas Maier, Stefan Steidl, Vincent Christlein, and Joachim Hornegger. *Medical imaging systems: An introductory guide*. Springer, 2018.

[145] M. Martin, F. v. d. Camp, and R. Stiefelhagen. Real time head model creation and head pose estimation on consumer depth cameras. In *2014 2nd International Conference on 3D Vision*, volume 1, pages 641–648, 2014.

[146] Keisuke Maruyama, Eiju Watanabe, Taichi Kin, Kuniaki Saito, Atsushi Kumakiri, Akio Noguchi, Motoo Nagane, and Yoshiaki Shiokawa. Smart glasses for neurosurgical navigation by augmented reality. *Operative Neurosurgery*, 15(5):551–556, Jan 2018.

[147] Antonio Meola, Fabrizio Cutolo, Marina Carbone, Federico Cagnazzo, Mauro Ferrari, and Vincenzo Ferrari. Augmented reality in neurosurgery: a systematic review. *Neurosurgical Review*, 40(4):537–548, May 2016.

[148] Microsoft. Hololensforcv c#/c++ library. https://github.com/microsoft/HoloLensForCV, 2019. Visited on 06/07/2021.

[149] Microsoft. Getting started with the mrt, 2020. URL `https://microsoft.github.io/MixedRealityToolkit-Unity/Documentation/GettingStartedWithTheMRTK.html`. Visited on 05/06/2021.

[150] Microsoft. Using the windows device portal, 2021. URL `https://docs.microsoft.com/en-us/windows/mixed-reality/develop/platform-capabilities-and-apis/using-the-windows-device-portal`. Visited on 14/08/2021.

[151] Yasushi Miyagi, Fumio Shima, and Tomio Sasaki. Brain shift: an error factor during implantation of deep brain stimulation electrodes. *Journal of neurosurgery*, 107(5):989–997, 2007.

[152] Holger Mönnich, Heinz Wörn, and Daniel Stein. Op sense—a robotic research platform for telemanipulated and automatic computer assisted surgery. In *AMC 2012*, pages 1–6. IEEE, 2012.

[153] Fabio Müller, Simon Roner, Florentin Liebmann, José M. Spirig, Philipp Fürnstahl, and Mazda Farshad. Augmented reality navigation for spinal pedicle screw instrumentation using intraoperative 3d imaging. *The Spine Journal*, Oct 2019.

[154] Michael Müller, Marie-Claire Rassweiler, Jan Klein, Alexander Seitel, Matthias Gondan, Matthias Baumhauer, Dogu Teber, Jens J. Rassweiler, Hans-Peter Meinzer, and Lena Maier-Hein. Mobile augmented reality for computer-assisted percutaneous nephrolithotomy. *International Journal of Computer Assisted Radiology and Surgery*, 8(4):663–675, Mar 2013.

[155] Leonid Naimark and Eric Foxlin. Circular data matrix fiducial system and robust image processing for a wearable vision-inertial self-tracker. In *Proceedings. International Symposium on Mixed and Augmented Reality*, pages 27–36. IEEE, 2002.

[156] Hirenkumar Nakawala, Giancarlo Ferrigno, and Elena De Momi. Toward a knowledge-driven context-aware system for surgical assistance. *Journal of Medical Robotics Research*, 2(03), 2017.

[157] Nassir Navab, Sandro-Michael Heining, and Joerg Traub. Camera augmented mobile c-arm (CAMC): Calibration, accuracy study, and clinical applications. *IEEE Transactions on Medical Imaging*, 29(7):1412–1423, Jul 2010.

[158] Nikhil V. Navkar, Nikolaos V. Tsekos, Jason R. Stafford, Jeffrey S. Weinberg, and Zhigang Deng. Visualization and Planning of Neurosurgical Interventions with Straight Access. In Nassir Navab and Pierre Jannin, editors, *Information processing in computer-assisted interventions*, volume 6135 of *Lecture Notes in Computer Science*, pages 1–11. Springer, Berlin, 2010.

[159] Binh P Nguyen, Wei-Liang Tay, Chee-Kong Chui, and Sim-Heng Ong. A clustering-based system to automate transfer function design for medical image visualization. *The Visual Computer*, 28(2):181–191, 2012.

[160] Christopher Nimsky, Oliver Ganslandt, Simon Cerny, Peter Hastreiter, Günther Greiner, and Rudolf Fahlbusch. Quantification of, visualization of, and compensation for brain shift using intraoperative magnetic resonance imaging. *Neurosurgery*, 47(5):1070–1080, 2000.

[161] Christopher Nimsky, Oliver Ganslandt, Peter Hastreiter, and Rudolf Fahlbusch. Intraoperative compensation for brain shift. *Surgical neurology*, 56(6):357–364, 2001.

[162] Hiroyuki Ogawa, Seiichirou Hasegawa, Sachiyuki Tsukada, and Masaaki Matsubara. A pilot study of augmented reality technology applied to the acetabular cup placement during total hip arthroplasty. *The Journal of Arthroplasty*, 33(6):1833–1837, Jun 2018.

[163] O. Oktay, W. Bai, R. Guerrero, M. Rajchl, A. de Marvao, D. P. O'Regan, S. A. Cook, M. P. Heinrich, B. Glocker, and D. Rueckert. Stratified decision forests for accurate anatomical landmark localization in cardiac images. *IEEE Transactions on Medical Imaging*, 36(1):332–342, 2017.

[164] Edwin Olson. Apriltag: A robust and flexible visual fiducial system. In *2011 IEEE International Conference on Robotics and Automation*, pages 3400–3407. IEEE, 2011.

[165] Shinji Onda, Tomoyoshi Okamoto, Masaru Kanehira, Shuichi Fujioka, Naoki Suzuki, Asaki Hattori, and Katsuhiko Yanaga. Short rigid scope and stereo-scope designed specifically for open abdominal navigation surgery: clinical application for hepatobiliary and pancreatic surgery. *Journal of Hepato-Biliary-Pancreatic Sciences*, 20(4):448–453, Dec 2012.

[166] Brent R O'Neill, Dennis A Velez, Ernest E Braxton, Donald Whiting, and Michael Y Oh. A survey of ventriculostomy and intracranial pressure monitor placement practices. *Surgical neurology*, 70(3):268–273, 2008.

[167] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics*, 9(1):62–66, 1979.

[168] Michael G. C. Pellen, Liam F. Horgan, J. Roger Barton, and Stephen E. Attwood. Construct validity of the ProMIS laparoscopic simulator. *Surgical Endoscopy*, 23 (1):130–139, Jul 2008.

[169] T. Pohlen, A. Hermans, M. Mathias, and B. Leibe. Full-Resolution Residual Networks for Semantic Segmentation in Street Scenes. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3309–3318, July 2017.

[170] PTC. Vuforia marker. `https://www.ptc.com/de/products/augmented-reality`, 2021. Visited on 06/07/2021.

[171] Xiaohua Qian, Yuan Lin, Yue Zhao, Xinyan Yue, Bingheng Lu, and Jing Wang. Objective ventricle segmentation in brain ct with ischemic stroke based on anatomical knowledge. *BioMed Research International*, 2017, 2017.

[172] Clemens Raabe, Jens Fichtner, Jürgen Beck, Jan Gralla, and Andreas Raabe. Revisiting the rules for freehand ventriculostomy: a virtual reality analysis. *Journal of neurosurgery*, 128(4):1250–1257, 2018.

[173] Ihsan Rabbi and Sehat Ullah. A survey of augmented reality challenges and tracking. *ACTA GRAPHICA*, 24:29–46, 02 2013.

[174] Tausif Rehman, Atiq ur Rehman, Rushna Ali, Amina Rehman, Hassaan Bashir, Salima Ahmed Bhimani, Hoan Tran, and Sidra Khan. A radiographic analysis of ventricular trajectories. *World neurosurgery*, 80(1-2):173–178, 2013.

[175] Ingerid Reinertsen, Frank Lindseth, Christian Askeland, Daniel Høyer Iversen, and Geirmund Unsgård. Intra-operative correction of brain-shift. *Acta neurochirurgica*, 156(7):1301–1310, 2014.

[176] Fabian Reister, Christian Kunz, Max Schneider, and Torsten Kroeger. Deep learning based 3D pose estimation of surgical tools using a RGB-D camera at the example of a catheter for ventricular puncture. In *CURAC 2018 - Tagungsband : 17. Jahrestagung der Deutschen Gesellschaft für Computer- und Roboterassistierte Chirurgie (CURAC), Leipzig, 13 – 15 September 2018*, pages 28–33. Universität Leipzig, 2018.

[177] Mario Rincón-Nigro, Nikhil V. Navkar, Nikolaos V. Tsekos, and Deng Zhigang. GPU-accelerated interactive visualization and planning of neurosurgical interventions. *IEEE computer graphics and applications*, 34(1):22–31, 2014.

[178] David W. Roberts, John W. Strohbehn, John F. Hatch, William Murray, and Hans Kettenberger. A frameless stereotaxic integration of computerized tomographic imaging and the operating microscope. *Journal of Neurosurgery*, 65(4):545–549, Oct 1986.

[179] Francisco Romero-Ramirez, Rafael Muñoz-Salinas, and Rafael Medina-Carnicer. Speeded up detection of squared fiducial markers. *Image and Vision Computing*, 76, 06 2018.

[180] Szymon Rusinkiewicz and Marc Levoy. Efficient variants of the icp algorithm. In *Proceedings third international conference on 3-D digital imaging and modeling*, pages 145–152. IEEE, 2001.

[181] R. B. Rusu, N. Blodow, and M. Beetz. Fast point feature histograms (fpfh) for 3d registration. In *2009 IEEE International Conference on Robotics and Automation*, pages 3212–3217, 2009.

[182] Radu Bogdan Rusu. *Semantic 3D Object Maps for Everyday Manipulation in Human Living Environments*. PhD thesis, Computer Science department, Technische Universitaet Muenchen, Germany, October 2009.

[183] Radu Bogdan Rusu and Steve Cousins. 3D is here: Point Cloud Library (PCL). In *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, May 9-13 2011.

[184] Radu Bogdan Rusu and Steve Cousins. *Voxel Grid Filter*, 2021. URL `https://pointclouds.org/documentation/tutorials/voxel_grid.html`. Visited on 03/05/2021.

[185] Artur Sagitov, Ksenia Shabalina, Leysan Sabirova, Hongbing Li, and Evgeni Magid. Artag, apriltag and caltag fiducial marker systems: Comparison in a presence of partial marker occlusion and rotation. In *ICINCO (2)*, pages 182–191, 2017.

[186] Asita Sarrafzadeh, Nicolas Smoll, and Karl Schaller. Guided (ventri-guide) versus freehand ventriculostomy: study protocol for a randomized controlled trial. *Trials*, 15(1):478, Dec 2014.

[187] Pascal Sati, Ilena C. George, Colin D. Shea, María I. Gaitán, and Daniel S. Reich. Flair: A combined mr contrast technique for visualizing white matter lesions and parenchymal veins. *Radiology*, 265(3):926–932, 2012. PMID: 23074257.

[188] Dieter Schmalstieg, Anton Fuhrmann, Gerd Hesina, Zsolt Szalavári, L Miguel Encarnaçao, Michael Gervautz, and Werner Purgathofer. The studierstube augmented reality project. *Presence: Teleoperators & Virtual Environments*, 11(1):33–54, 2002.

[189] HG Schnack, HE Hulshoff Pol, William Frans Christiaan Baaré, MA Viergever, and RS Kahn. Automatic segmentation of the ventricular system from mr images of the human brain. *Neuroimage*, 14(1):95–104, 2001.

[190] Max Schneider, Christian Kunz, Andrej Pal'a, Christian Rainer Wirtz, Franziska Mathis-Ullrich, and Michal Hlaváč. Augmented reality–assisted ventriculostomy. *Neurosurgical Focus*, 50(1), 2021. Publisher: American Association of Neurological Surgeons.

[191] Alice Segato, Valentina Pieri, Alberto Favaro, Marco Riva, Andrea Falini, Elena De Momi, and Antonella Castellano. Automated Steerable Path Planning for Deep Brain Stimulation Safeguarding Fiber Tracts and Deep Gray Matter Nuclei. *Frontiers in Robotics and AI*, 6:70, 2019. URL `https://www.frontiersin.org/article/10.3389/frobt.2019.00070`.

[192] Alice Segato, Chiara Di Vece, Sara Zucchelli, Marco Di Marzo, Thomas Wendler, Mohammad Farid Azampour, Stefano Galvan, Riccardo Secoli, and Elena De Momi. Position-based dynamics simulator of brain deformations for path planning and intra-operative control in keyhole neurosurgery. *IEEE Robotics and Automation Letters*, 2021.

[193] Petr Sereda, A Vilanova Bartroli, Iwo WO Serlie, and Frans A Gerritsen. Visualization of boundaries in volumetric data sets using lh histograms. *IEEE Transactions on Visualization and Computer Graphics*, 12(2):208–218, 2006.

[194] Petr Sereda, Anna Vilanova, and Frans A Gerritsen. Automating transfer function design for volume rendering using hierarchical clustering of material boundaries. In *EuroVis*, pages 243–250, 2006.

[195] Ángela Di Serio, María Blanca Ibáñez, and Carlos Delgado Kloos. Impact of an augmented reality system on students' motivation for a visual art course. *Computers & Education*, 68:586–596, Oct 2013.

[196] IWO Serlie, Roel Truyen, Jasper Florie, Frits Post, Lucas van Vliet, and Frans Vos. Computed cleansing for virtual colonoscopy using a three-material transition model. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 175–183. Springer, 2003.

[197] Reuben R. Shamir, Leo Joskowicz, Idit Tamir, Elad Dabool, Lihi Pertman, Adam Ben-Ami, and Yigal Shoshan. Reduced risk trajectory planning in image-guided keyhole neurosurgery. *Medical physics*, 39(5):2885–2895, 2012.

[198] Mahesh B Shenai, Marcus Dillavou, Corey Shum, Douglas Ross, Richard S Tubbs, Alan Shih, and Barton L Guthrie. Virtual interactive presence and augmented reality (VIPAR) for remote surgical assistance. *Operative Neurosurgery*, 68:200–207, Mar 2011.

[199] Oskar Škrinjar, Arya Nabavi, and James Duncan. Model-driven brain shift compensation. *Medical image analysis*, 6(4):361–373, 2002.

[200] Krishna Prasad Soundararajan and Thomas Schultz. Learning probabilistic transfer functions: A comparative study of classifiers. In *Computer Graphics Forum*, volume 34, pages 111–120. Wiley Online Library, 2015.

[201] SourceForge-Teem. Definition of NRRD File Format, 2021. URL `http://teem.sourceforge.net/nrrd/format.html`. Visited on 14/08/2021.

[202] SourceForge-Teem. Utah nrrd utilities, 2021. URL `http://teem.sourceforge.net/unrrdu/`. Visited on 14/08/2021.

[203] Detlev Stalling, Malte Westerhoff, Hans-Christian Hege, et al. Amira: A highly interactive system for visual data analysis. *The visualization handbook*, 38:749–67, 2005.

[204] M. Strowitzki, Y. Komenda, R. Eymann, and W. I. Steudel. Accuracy of ultrasound-guided puncture of the ventricular system. *Child's Nervous System*, 24(1):65–69, Jan 2008.

[205] Anna Syberfeldt, Magnus Holm, Oscar Danielsson, Lihui Wang, and Rodney Brewster. Support systems on the industrial shop-floors of the future – operators' perspective on augmented reality. volume 44, pages 108–113, 05 2016.

[206] Leila Besharati Tabrizi and Mehran Mahvash. Augmented reality–guided neurosurgery: accuracy and intraoperative application of an image projection technique. *Journal of Neurosurgery*, 123(1):206–211, Jul 2015.

[207] OpenCV team. *OpenCV*, 2020. URL `https://opencv.org/about/`. Visited on 05/06/2021.

[208] U. W. Thomale, T. Knitter, A. Schaumann, S. A. Ahmadi, P. Ziegler, M. Schulz, and C. Miethke. Smartphone-assisted guide for the placement of ventricular catheters. *Child's Nervous System*, 29(1):131–139, Jan 2013.

[209] Ahmed K Toma, Sophie Camp, Laurence D Watkins, Joan Grieve, and Neil D Kitchen. External ventricular drain insertion accuracy: is there a need for change in practice? *Neurosurgery*, 65(6):1197–1201, 2009.

[210] Ari Trachtenbert. Computational methods in coding theory. Master's thesis, University of Illinois at Urbana-Champaign, 1996.

[211] Sachiyuki Tsukada, Hiroyuki Ogawa, Masahiro Nishino, Kenji Kurosaka, and Naoyuki Hirasawa. Augmented reality-based navigation system applied to tibial bone resection in total knee arthroplasty. *Journal of Experimental Orthopaedics*, 6 (1), Nov 2019.

[212] F-Y Tzeng, Eric B Lum, and K-L Ma. An intelligent system approach to higher-dimensional classification of volume data. *IEEE Transactions on visualization and computer graphics*, 11(3):273–284, 2005.

[213] O Vaillancourt, A Boré, G Girard, and M Descoteaux. A Fiber Navigator for Neurosurgical Planning (NeuroPlanningNavigator). In *Paper presented at IEEE VIsualization. November. Utah, USA*, 2010.

[214] Tristan P C van Doormaal, Jesse A M van Doormaal, and Tom Mensink. Clinical accuracy of holographic navigation using point-based registration on augmented-reality glasses. *Operative Neurosurgery*, 17(6):588–593, May 2019.

[215] Frederick Van Gestel, Taylor Frantz, Mumtaz Hussain Soomro, Shirley A Elprama, Cedric Vannerom, An Jacobs, Jef Vandemeulebroucke, Bart Jansen, Thierry Scheerlinck, and Johnny Duerinck. Augmented reality-assisted neurosurgical drain placement (araned). *Acta Neurochirurgica. Supplementum*, 131:267–273, 2021.

[216] Rick Van Krevelen and Ronald Poelman. A survey of augmented reality technologies, applications and limitations. *International Journal of Virtual Reality*, 9:1, 06 2010.

[217] P. Vávra, J. Roman, P. Zonča, P. Ihnát, M. Němec, J. Kumar, N. Habib, and A. El-Gendi. Recent development of augmented reality in surgery: A review. *Journal of Healthcare Engineering*, 2017:1–9, 2017.

[218] Iris SC Verploegh, Victor Volovici, Iain K Haitsma, Joost W Schouten, Clemens M Dirven, Johan M Kros, and Ruben Dammers. Contemporary frameless intracranial biopsy techniques: Might variation in safety and efficacy be expected? *Acta neurochirurgica*, 157(11):2011–2016, 2015.

[219] Iván Villanueva-Naquid, Carlos Soubervielle-Montalvo, Ruth M. Aguilar-Ponce, Saúl Tovar-Arriaga, Juan C. Cuevas-Tello, Cesar A. Puente-Montejano, Marcela Mejia-Carlos, and Jaime G. Torres-Corzo. Risk assessment methodology for trajectory planning in keyhole neurosurgery using genetic algorithms. *The International Journal of Medical Robotics and Computer Assisted Surgery*, 16(2), 2020.

[220] Anna Maria von der Heide, Pascal Fallavollita, Lejing Wang, Philipp Sandner, Nassir Navab, Simon Weidert, and Ekkehard Euler. Camera-augmented mobile c-arm (CamC): A feasibility study of augmented reality imaging in the operating room. *The International Journal of Medical Robotics and Computer Assisted Surgery*, 14 (2), Dec 2017.

[221] Daniel Wagner and Dieter Schmalstieg. Artoolkitplus for pose tracking on mobile devices. 2007.

[222] Leo L Wang, Hao-Hua Wu, Nadir Bilici, and Rebecca Tenney-Soeiro. Gunner goggles: Implementing augmented reality into medical education. *Studies in health technology and informatics*, 220:446–449, 2016.

[223] Eiju Watanabe, Makoto Satoh, Takehiko Konno, Masahiro Hirai, and Takashi Yamaguchi. The trans-visible navigator: A see-through neuronavigation system using augmented reality. *World Neurosurgery*, 87:399–405, Mar 2016.

[224] Thomas J Wilson, William R Stetler, Wajd N Al-Holou, and Stephen E Sullivan. Comparison of the accuracy of ventricular catheter placement using freehand placement, ultrasonic guidance, and stereotactic neuronavigation. *Journal of neurosurgery*, 119(1):66–70, 2013.

[225] Graeme Woodworth, Matthew J McGirt, Amer Samdani, Ira Garonzik, Alessandro Olivi, and Jon D Weingart. Accuracy of frameless and frame-based image-guided stereotactic brain biopsy in the diagnosis of glioma: comparison of biopsy and open resection specimen. *Neurological research*, 27(4):358–362, 2005.

[226] Graeme F Woodworth, Matthew J McGirt, Amer Samdani, Ira Garonzik, Alessandro Olivi, and Jon D Weingart. Frameless image-guided stereotactic brain biopsy procedure: diagnostic yield, surgical morbidity, and comparison with the frame-based technique. *Journal of neurosurgery*, 104(2):233–237, 2006.

[227] Jing-Ren Wu, Min-Liang Wang, Kai-Che Liu, Ming-Hsien Hu, and Pei-Yuan Lee. Real-time advanced spinal surgery via visible patient model and augmented reality system. *Computer Methods and Programs in Biomedicine*, 113(3):869–881, Mar 2014.

[228] C. Xu and J. L. Prince. Gradient vector flow: a new external force for snakes. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 66–71, June 1997.

[229] Andy Zeng, Kuan-Ting Yu, Shuran Song, Daniel Suo, Ed Walker, Alberto Rodriguez, and Jianxiong Xiao. Multi-view self-supervised deep learning for 6D pose estimation in the Amazon Picking Challenge. In *Robotics and Automation (ICRA), 2017 IEEE International Conference on*, pages 1386–1383. IEEE, 2017.

[230] J. Zhang, M. Liu, and D. Shen. Detecting anatomical landmarks from limited medical imaging data using two-stage task-oriented deep neural networks. *IEEE Transactions on Image Processing*, 26(10):4753–4764, 2017.

[231] Q. Zhang, M. Alexander, and L. Ryner. Multimodality Neurological Data Visualization With Multi-VOI-Based DTI Fiber Dynamic Integration. *IEEE Journal of Biomedical and Health Informatics*, 20(1):293–303, 2016.

[232] Yefeng Zheng, David Liu, Bogdan Georgescu, Hien Nguyen, and Dorin Comaniciu. 3d deep learning for efficient and robust landmark detection in volumetric data. In Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 565–572, Cham, 2015. Springer International Publishing.

[233] Zoran Zivkovic. Improved adaptive Gaussian mixture model for background subtraction. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 2, pages 28–31. IEEE, 2004.

List of Figures

List of Tables

Appendix

A. Scene Analysis

A.1. Instrument Tracking

This content is taken from Reister et al. [176] and is presented in abbreviated form.

Assistance systems in the operating room can significantly improve the chances of a successful surgery. With the help of augmented reality, normally hidden structures can be visualized. This work addresses the necessary pose estimation of surgical tools. The tools in the color image of the camera are segmented using Deep Learning and then fused with the depth image to get a three-dimensional representation. The solution was implemented and later on evaluated using the example of a catheter for ventricular punctures.

A.1.1. Experimental Setup

An Intel RealSense SR300, statically mounted on a tripod, was used to provide color and depth data. A Synbone phantom head was used as a patient phantom. To create different lighting conditions a surgical lamp was utilised. A workstation was used with an Intel Core i7-8700K (3,7GHz), 64GB RAM and a Nvidia GeForce 1080 Ti GPU.

A.1.2. Segmentation of Colored Images

A Full-Resolution Residual Network based on the FRRN-A architecture proposed in [169] was used. A categorical cross entropy was taken as a cost function, only considering the worst pixels as suggested by Pohlen et al. [169]. Then a backpropagation was performed using the Adam optimizer [103], leading to an accurate segmentation for fine image structures [176]. The segmentation was carried out on the rgb image.

The segmentation was later on fused with the depth stream to perform a 3D projection and to cluster voxels belonging to the catheter. The segmentation pipeline is visualized in Figure A.1.

A.1.3. Generation of Training Data

Based on the method of Zeng et al. [229] the whole scene was labeled initially as background. A single object added to the scene was then segmented by means of color and depth, as depicted in Figure A.2. A Gaussian mixture-based approach [233] was chosen to handle tool shadows under illumination changes. To train the model, three datasets

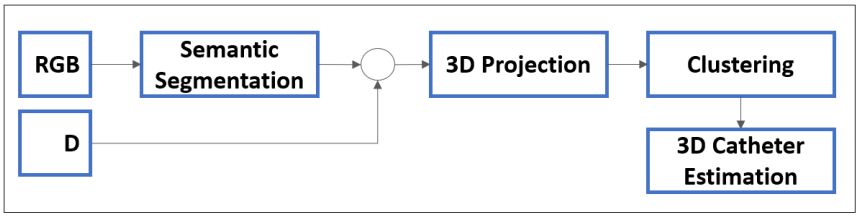


Figure A.1.: Segmentation pipeline [115].

were created as described in Table A.1 and shown in Figure A.3, using different scenes, lighting conditions and viewpoints. Each dataset contains approximately 1500 images with the catheter being partially or fully visible [176].

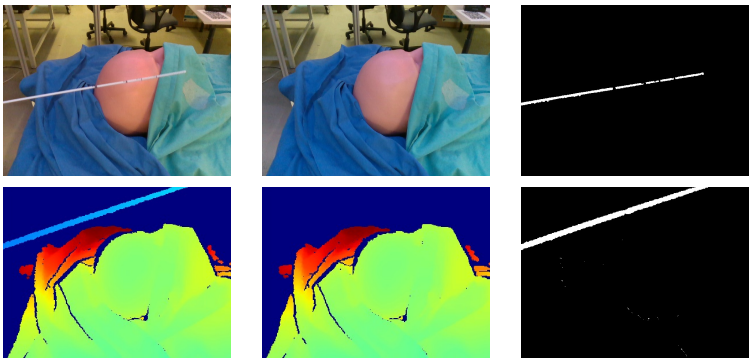


Figure A.2.: Top: Catheter segmentation based on the color image. Bottom: Segmentation based on the depth stream. From left to right: Image with catheter, background image and segmentation. The same camera position was used for all images [176].

Dataset	1	2	3
Scene	Desk	Surgery	Surgery with additional items
Training images	5000	5000	4000
Validation images	2000	1500	1500

Table A.1.: Scenarios used for segmentation [176].

Figure A.3.: Exemplary images of the generated datasets [176].

A.1.4. 3D Pose Estimation

A point cloud of the tool is created based on the segmented part of the image and the depth image. A density based clustering algorithm is applied to identify multiple objects of the same type. A catheter is modeled as a cylinder, the eigenvector corresponding to the greatest eigenvector indicates the direction of the cylinder [176].

A.1.5. Segmentation Evaluation

For each dataset, precision, recall and F1-Score are used as metrics for validation (Figure A.4, Table A.2). Figure A.5 shows representative training results. The first two columns show how well the catheter is recognized in a crowded scene. Column three shows that the trained network produces a better prediction than the training data offers and column four shows that poor training data can lead to poor predictions [176].

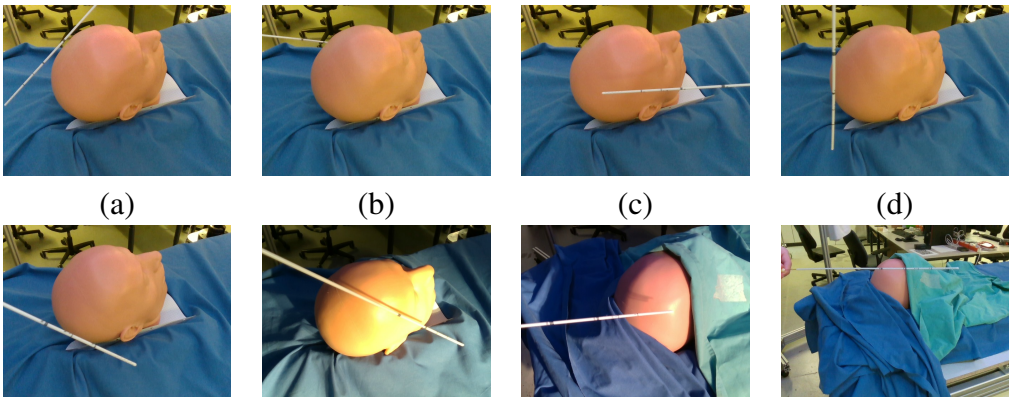


Figure A.4.: Validation datasets. Images corresponding to the datasets listed in Table A.2 [176].

Dataset	Surgery 1 (a)	Surgery 2 (b)	Catheter horizontal (c)	Catheter vertical (d)	Catheter diagonal (e)	Bright (f)	Bright, patient covered (g)	Increased distance (h)
Precision	84.1%	64.0%	77.1%	47.7%	81.0%	59.0%	72.9%	11.5%
Recall	58.7%	61.9%	51.2%	19.2%	31.0%	15.4%	43.1%	55.8%
F1-Score	69.2%	62.9%	61.6%	27.4%	44.9%	24.4%	54.1%	19.0%

Table A.2.: Evaluation results for semantic segmentation [176].

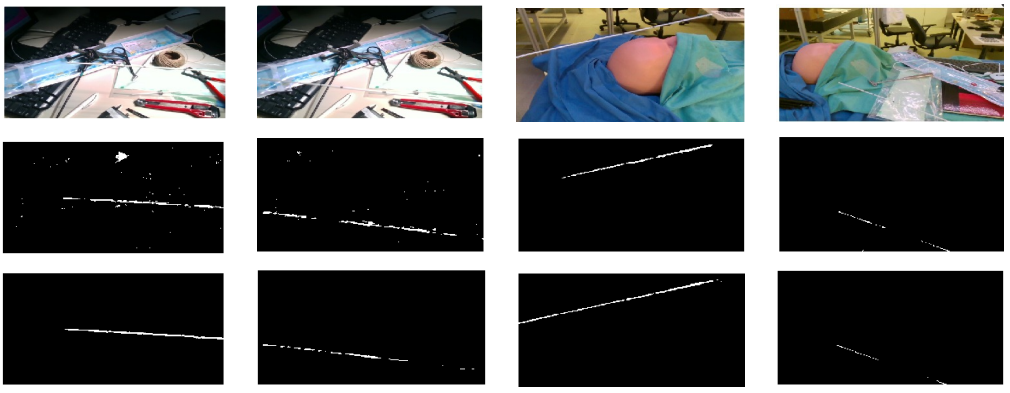


Figure A.5.: Selected samples for training. Top: RGB image. Center: training data. Bottom: Prediction [176].

A.1.6. Model Fitting Performance

To evaluate how well the catheter length of 34.5 cm is recognized we recorded 500 frames. During the whole time the catheter was fully visible. The length prediction is pretty stable

when the catheter is placed parallel in the front of the camera. The length estimation becomes worse when the catheter is moved towards and away from the camera, closer to the image border or is rotated away from the camera [176].

A.1.7. Discussion and Conclusion

The semantic segmentation was able to classify fine image structures like the catheter for ventricular puncture. To improve the performance issues, the training dataset could be extended by automatically creating more images using a camera mounted to one robot arm and the tool to another and slightly change positions of both for each image. The used depth camera doesn't handle dark and reflective surface well and could be replaced by a time-of-flight sensor together with a color camera. A Kalman filter was used to reduce the input size to achieve higher performance. The system could be augmented to track also the surgeon, the patient, the operating table and other tools [176].

The system is able to perform an accurate pose estimation of a surgical tool using a color image in combination with a depth image. The semantic segmentation works well in complex situations. This approach could be combined with an augmented reality-based surgical navigation system to present the surgeon additional information about the surgery and the status of the patient [176].

A.2. OR Staff Tracking

This content is taken from Kunz et al. [115] and is presented in abbreviated form.

The presented system integrates augmented reality into the digital operating room (OR). It enables the surgeon to see obscured structures during neurosurgical operations in realtime. The system tracks the patient on the OR table, medical instruments and the medical staff to provide a better understanding of the current situation in the operating room [115].

The goal is to integrate the proposed augmented reality system [111] into the surgical research platform for robot-assisted surgery OP:Sense [152]. To track the patient on the operating table, a two-stage process was implemented. To determine the coarse position of the patients's head on the OR table, first a YOLO v3 net is applied to reduce search space. A robot with an RGB-D sensor mounted on it, is used to capture a point cloud of the OR table. After the YOLO net has found the coarse position of the patient and marked it with a bounding box, a two-step surface matching approach is used to determine the exact position of the patient. When the surface matching procedure is completed, the marker, mounted on the patient's head, is tracked to determine its position. The registration matrix can then be calculated once the positions of the patient and the marker are known. In addition, the puncture catheter is tracked to determine its insertion depth into the brain. Tracking the medical staff enables the system to react to the current situation, e.g. passing of an instrument [115].

OP:Sense is our digital operating room (illustrated in Figure 5.5a).

It consists of an OR table with a Kuka LWR4 and a Franka Panda robot attached to it. The following sensors are attached to the ceiling rack: an ARTTrack 2 system consisting of six IR cameras and four Microsoft Kinect sensors (Figure 5.5b). A custom-build phantom skull and a patient phantom head from Synbone were used during the experiments.

For AR visualization, the surgeon wears a Microsoft HoloLens. For patient tracking Vuforia markers are used. The Robot Operating System (ROS), a middleware for robotic platforms, is the base of OP:Sense [115].

Alternatively, the Kinect RGB image streams can be used to detect the patient's head with the YOLO v3 CNN. When this is not working, the mounted camera on the Franka Panda is used, as it is closer to the patient. With additional information from the depth stream, the 3D position is determined. Retro-reflective markers of the ARTtrack system are used to track the OR table and the robots. The key points on the bodies of the medical staff are tracked using the OpenPose [32] library. To integrate OpenPose in the OP:Sense ROS environment, the available ROS nodes have been modified. The architecture is presented in Figure A.6 [115].

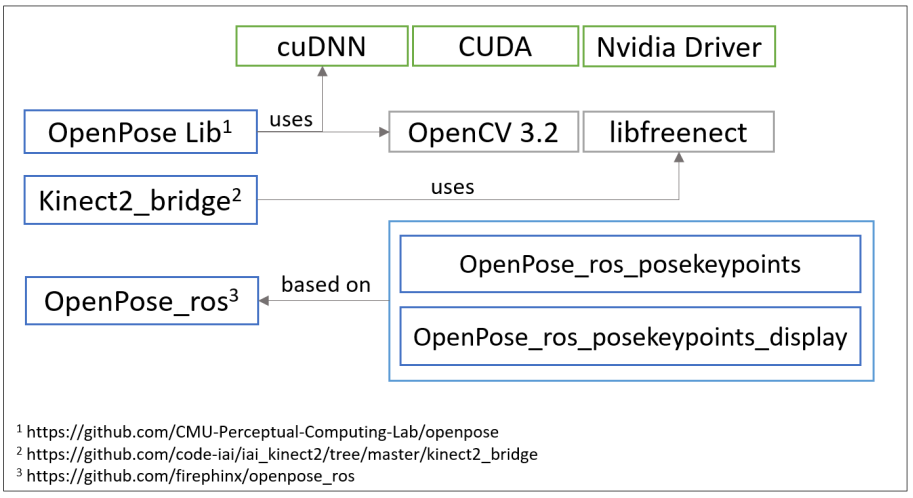


Figure A.6.: Architecture of the OpenPose library implementation in ROS for integration in OP:Sense [115].

A phantom head was placed on the OR table to evaluate the systems performance. First the coarse detection of the patient's head with the YOLO v3 net was evaluated. Figure A.7 shows multiple secenarios with changing occlusion and lighting conditions. The evaluation results are shown in Table A.3. Then the position is determined via surface matching. The system detects the patient with an accuracy between 3 and 7 mm. Under normal OR conditions the results are satisfying. Under bright lighting conditions the solution performance drops. The reason for this are large flares on the phantom made of plastic, which prevent the YOLO v3 net to find the patient's head. Normally, these large flares do not occur in every stream when using the four Kinect cameras, so an overall robust detection is possible.

The OpenPose [32] library is used to track body parts and joint positions of the visible medical personnel. Initially, only a qualitative evaluation was performed, which yielded good results, as shown in Figure A.8. A frame rate of 21 fps was reached on a workstation with an Intel i7-9700k, 32 GB RAM and a GeForce 1080 Ti [115]. The ready-to-use detection algorithm OpenPose produced good results even with the medical stuff wearing surgical clothing [115].

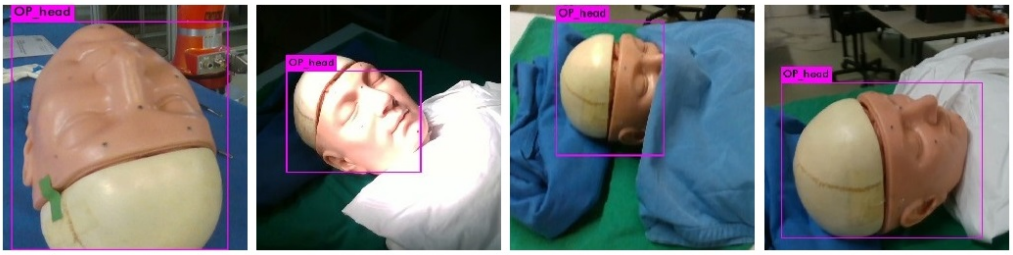


Figure A.7.: Coarse determination of the patient's head on the OR table with YOLO v3 [115].

	Precision	Recall	F1-Score	average IoU	mAP
Normal OR conditions	92%	99%	95%	67.59%	90.35%
Occlusion	99%	93%	96%	75.77%	90.86%
Strong Illumination	62%	66%	64%	41.01%	62.23%
Illumination and Occlusion	65%	51%	57%	41.83%	45.62%

Table A.3.: Evaluation of the position determination of the phantom head with YOLO v3 [115].

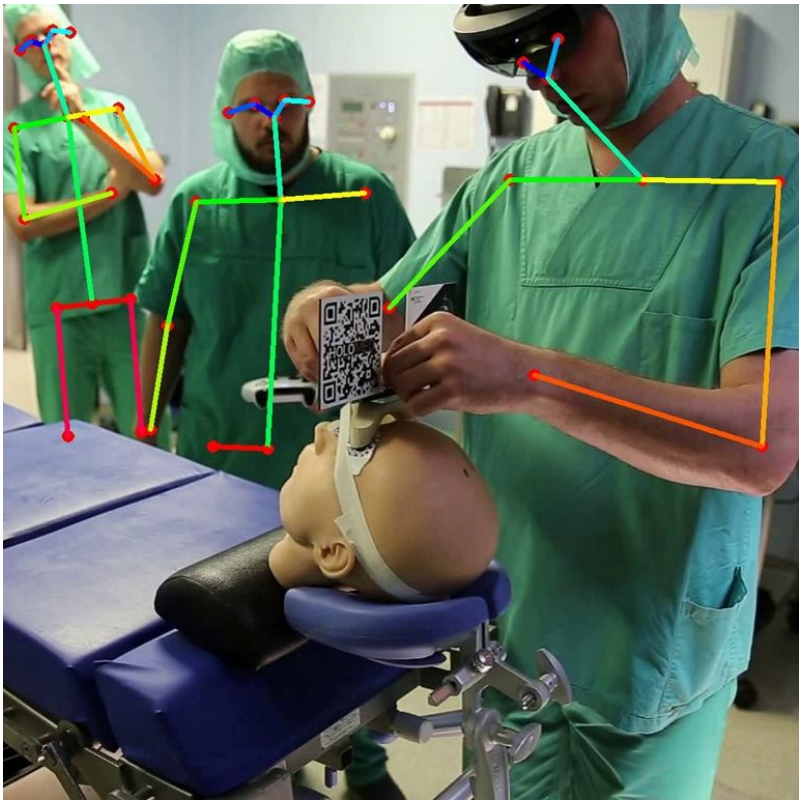


Figure A.8.: Results of the medical staff tracking [115].