GPU Accelerated Visualization for Neurostimulation Therapy

Von der Fakultät für Ingenieurwissenschaften,
Abteilung Informatik und Angewandte Kognitionswissenschaft
der Universität Duisburg-Essen

zur Erlangung des akademischen Grades

Doktor der Naturwissenschaft *(Dr. rer. nat.)*

genehmigte Dissertation

von

Andre Waschk
aus
Dorsten

Gutachter: Univ.-Prof. Dr. rer. nat. Jens Krüger
Gutachter: Univ.-Prof. Dr. rer. nat. Josef Pauli
Tag der mündlichen Prüfung: 24.02.2023

Declaration

Hiermit versichere ich an Eides statt durch meine Unterschrift, dass die vorgelegte Dissertation gemäß §9 der Promotionsordnung der Fakultät für Ingenieurwissenschaften der Universität Duisburg-Essen vom 6. August 2015 eine selbstständig durchgeführte und eigenständig verfasste Forschungsleistung darstellt und ich keine anderen als die angegebenen Hilfsmittel und Quellen benutzt habe. Alle Stellen, die wörtlich oder sinngemäß aus anderen Schriften entnommen sind, habe ich als solche kenntlich gemacht. Die Arbeit hat weder in gleicher noch in ähnlicher Form einem anderen Prüfungsausschuss vorgelegen.

Die Strafbarkeit einer falschen eidesstattlichen Versicherung ist mir bekannt, namentlich die Strafandrohung gemäß §156 StGB bis zu drei Jahren Freiheitsstrafe oder Geldstrafe bei vorsätzlicher Begehung der Tat bzw. gemäß §161 Abs.1 StGB bis zu einem Jahr Freiheitsstrafe oder Geldstrafe bei fahrlässiger Begehung

Andre Waschk
March 2023

Abstract

Visualization plays an important role in many parts of our daily lives. It surrounds us wherever we are in various forms and shapes. Data is everywhere, and visualizing it can be crucial for fast decision-making processes. Even critical parts of our society, such as medicine, rely on different imaging modalities daily. Many imaging techniques have been around for decades and are used during patient assessment. Techniques such as CT or MRI have been established as a standard in medical care and are seen as indispensable for various treatment forms. These techniques are not only seen as indispensable for patient assessment, but they are also essential for some treatment forms, such as neurostimulation therapy. In neurostimulation therapy, the goal is to implant electrodes within the deep areas of the patient's brain to suppress malfunctioning cells within the targeted tissue. Therefore, imaging techniques such as CT and MRI are crucial to locating the correct target tissue. Even though these imaging techniques have been around for decades, they are still primarily used as two-dimensional slice views as they were introduced to the market at the beginning.

However, starting from the mid-1990s, significant progress has been made in computer visualization due to the introduction of dedicated hardware in the form of graphics processing units. Over time these processors became more and more freely programmable to the user and opened a vast space for applications. While the consumer market adopted the advancements in computer graphics in the form of video games or movies, imaging techniques in the clinical field mainly stayed the same.

Especially techniques for visualizing three-dimensional volume data, such as CT or MRI data, have become more accessible over time and could provide a more realistic assessment to doctors in the field of neurostimulation therapy by providing a more in-depth visualization of the three-dimensional structures of the brain. Also, new technologies such as modern virtual reality systems allow combining three-dimensional imaging data with an immersive visualization for more insight. Furthermore, the computational power of modern graphics processors allows the real-

time processing of daily medical data for fast access to the underlying information. However, these possibilities have been largely neglected by clinical practice.

This dissertation will present different techniques and systems that aim to aid doctors in their daily tasks by providing new visualization domains currently not in clinical use. Therefore, the presented research will focus on two aspects of neurostimulation therapy, the preoperative planning phase of the intervention and the intraoperative visualization of various data domains in real time. While imaging data is indispensable for planning the intervention, it is mostly not found during the surgery. More so, do doctors rely on additional modalities in the form of microelectrode recording to verify the current position of the electrode. The planning does not utilize the full capabilities of three-dimensional volume data and primarily relies on two-dimensional slice views of the data. Within this research, systems have been developed for a) providing spatial access to the complete volumetric data set by utilizing modern VR hardware and b) bringing the imaging data into the oval for more insight during crucial decision-making processes.

With the solution found in this work, surgeons can better assess the patient's condition by exploring the recorded data in an immersive environment and planning the intervention in a realistic spatial space. Furthermore, does the research support practitioners with intraoperative visualization of fused CT, MRI, and microelectrode recording data. While up to this point, practitioners had to rely on the information gained during the planning stage and could only utilize the intraoperative recorded neural activity for cross-validation, the presented solution provides a more in-depth analysis of the data utilizing different inputs of information to aid the practitioner.

Zusammenfassung

Visualisierung spielt in vielen Teilen unseres täglichen Lebens eine wichtige Rolle. Sie begleitet uns, wo auch immer wir sind, in verschiedensten Formen. Daten sind allgegenwärtig und diese visuell darzustellen für eine schnelle Beurteilung spielt in vielen Bereichen eine wichtige Rolle. Auch wichtige Bereiche wie die Medizin verlassen sich auf eine Vielzahl von Datenmodalitäten, die in der täglichen Praxis visualisiert werden. Viele bildgebende Techniken in der Medizin sind seit Jahrzehnten im Nutzen und unterstützen Ärzte in der Beurteilung der gesundheitlichen Lage ihrer Patienten. Techniken wie CT oder MRI haben sich als Standardverfahren etabliert und sind nicht mehr wegzudenken. Nicht nur werden diese Techniken für eine grundlegende Begutachtung der aktuellen Situation genutzt, viel mehr sind diese auch essenziell für einige medizinische Verfahren. So sind bildgebende Verfahren unabdingbar für Neurostimulationstherapien, um einen genauen Einblick in die tiefen Hirnregionen zu erhalten, die so nicht einsehbar sind. Innerhalb von Neurostimulationstherapien werden kleine Elektroden tief in das Hirn des Patienten implantiert, welche über kleine Stromstöße Fehlfunktionen im Hirn unterbinden sollen. Um nicht blind diese Elektroden zu platzieren, sind bildgebende Verfahren unabdingbar. Obwohl die bildgebende Verfahren seit Jahrzehnten vorhanden sind, ist es bis heute noch immer der Standard diese als zwei dimensionale Schichten zu betrachten, so wie diese damals auf dem Markt eingeführt wurden.

Jedoch gab es seit Mitte der 1990er Jahre einen signifikanten Umbruch im Bereich der Computer Visualisierung. Immer schneller werdende dedizierte Grafikprozessoren erschienen auf dem Markt, mit immer mehr Freiheiten diese zu programmieren und für eine Vielzahl von Problemen zu nutzen. Während der Konsumbereich immer mehr auf moderne Visualisierungen baut in Form von Videospielen, Filmen und Co., so hat sich in der Medizin die traditionelle Bildgebung fest verankert.

Gerade Verfahren zur Visualisierung von dreidimensionalen Volumen, wie unter anderem CT oder MRI, wurden über die Jahre immer zugänglicher und können gerade im Bereich der Neurostimulationstherapie den Ärzten einen deutlich natür-

licheren Einblick in die Problematik geben. Auch neue Technologien wie moderne Virtual Reality Systeme bieten in Kombination mit den bildgebenden Verfahren Möglichkeiten einen Einblick in die tiefen Hirnstrukturen zu bekommen, der vorher nicht möglich war. Zusätzlich bietet die pure Rechenleistung von modernen Grafikprozessoren die Möglichkeit eine Vielzahl von medizinischen Daten in Echtzeit auszuwerten und zu visualisieren.

In dieser Dissertation werden Techniken und Systeme vorgestellt, die Ärzten mittels Visualisierung in ihrer täglichen Arbeit unterstützen sollen. Die vorgestellte Forschung fokussiert sich dabei auf zwei Kernbereiche der Neurostimulationstherapie; der Präoperativen Planung des Eingriffes und der Intraoperativen Visualisierung. Während bildgebende Verfahren unabdingbar für die Planung sind, können diese oftmals nicht innerhalb OP eingesehen werden. Zusätzlich werden weitere Modalitäten innerhalb der OP in Form von Mikroelektrode Aufnahmen eingeführt, welche sich hervorragend auswerten und mit den bereits vorhandenen Daten verbinden lassen. Die Planung selbst, basiert oftmals auf einzelnen Schichtbildern von komplexen dreidimensionalen Gebilden, welche der Arzt in seinem Kopf selbst wieder mental zusammenfügen muss. Innerhalb dieser Forschung wurden Verfahren entwickelt, die moderne virtuelle Systeme nutzbar machen, um diese Strukturen in einer Umgebung zu erkunden, die eine bessere räumliche Darbietung der Problematik bieten.

Mit den Lösungen, die in dieser Arbeit zu finden sind, können Ärzte eine genauere Bestandsaufnahme der Situation innerhalb einer immersiven dreidimensionalen Umgebung machen und auch in dieser Umgebung den eigentlich Eingriff sehr real planen. Weiterhin unterstützt die präsentierte Forschung Ärzte innerhalb des OP-Saals durch eine in Echtzeit fusionierte Visualisierung von CT, MRI und den intraoperativ aufgenommenen Hirnaktivitäten. Während Ärzte bisher nur basierend auf der präoperativen Planung und der intraoperativen Hirnaufzeichnung entscheiden mussten, ob Sie das richtige Gewebe bereits erreicht haben, können Sie mit dem präsentierten System eine korrekte Anschauung aller Daten erhalten und bekommen zusätzlich noch eine vom Computer berechnete Einschätzung der Daten.

Acknowledgements

Many people supported me throughout the years and stayed by my side to finish this work.

First, thank all my friends who provided me with the strength and endurance to keep working on this thesis over the last few years. While I was often at the point where my energy and motivation were close to hitting the ground, they helped me to buckle up and keep finishing what I had started. While the list of people would be too long to include on this page, I would love to mention at least a handful of people that never let me down. Thanks to Delia von Calle, Daniel Preuß, Pascal Breuer, and Maurice Miklis.

Special thanks go to my supervisor Jens Krüger, for all the support, trust, and encouragement to keep pushing and moving forward. Thank you, Jens, for allowing me to work not only for you but also with you. Thanks for the discussions, ideas, and laughs over the years.

Moreover, I also want to thank all my colleagues at the University of Duisburg-Essen and my fellow students I worked with over the years. Special thanks go out to Alexander Schiewe and Andrey Krekhov, who introduced me to the group during my time as a bachelor. Furthermore, thanks to my (former) colleagues, including Sebastian Cmentowoski, Katharina Emmerich, Michael Michalski, Daniel Preuß, Karl-Michael Schorer, and our secretary Sandra Palmer.

I also want to thank my beloved dog Otis, who often slept on my feet while I was working on this thesis and kept me happy all the time. Thank you, little buddy.

Lastly, I would love to thank Katharina Landwehr. Even though our ways have parted, you are a beautiful person and will always keep a special place in my heart. You motivated me to finish this dissertation and kept me believing in a better future once this chapter of my life is closed. I will miss the times we laughed and talked and our shared dreams. Even though our doors have closed for the time being, sometimes doors will reopen, and both of us will be surprised who we will get to know after some time. I wish you the best, and thank you for everything.

Table of contents

Chapter 1

Introduction

Visualizations have accompanied us for thousands of years. Equally long, the term visualization has been used for various types of imaging. Visualizations can be found everywhere and are used as a medium for communication [109]. The definitions of *visualization* are vast but have one in common; visualization is about creating an image to communicate a more profound message not directly accessible by the data itself. While nowadays, people often think of computer-generated images when talking about visualization, due to its age, visualization can be something other than computer generated. Techniques used for visualization are far older than modern computers and include maps, plots, or even drawings created by hand.

With the rise of the modern computer and its capabilities to generate images, computer-generated visualization gained more and more interest from researchers. In its beginning, visualization was limited to clusters of computers. The computational power was limited, and the complex algorithms that had to be computed for the desired visualization made those extensive and expensive clusters mandatory. Only a handful of research facilities could afford these clusters, which as a result, made visualizations based on computer graphics far less accessible to the broad public.

In the mid-1990s, this hardware limitation was overcome due to the rapid growth in the computational power of personal computers. The introduction of dedicated hardware for creating digital images, a Graphics Processing Unit (GPU), and a shift to a freely programmable rendering structure provided more and more opportunities for the applicability of computer-generated visualization. Since then, the growth in computational capabilities of modern computer hardware has not stopped.

With the vast accessibility to personal computers, more and more fields acknowledged the benefits that visualization could bring to daily life. While various fields

have adopted visualization techniques, some still rely on older traditional methods of data visualization methods due to diverse reasons.

One of these fields is daily clinical practice. Each day, data is collected in manifold forms. These can be imaging data of Computed Tomography (CT) or Magnetic Resonance Imaging (MRI) scans, clinical data collected from simulations or analysis, or recording data from muscles and brain activities. Many techniques used to collect these data are older than modern computer-supported visualization or as old as the first computer-based visualization systems.

While all of the visualization techniques currently in practice have established themself as reliable due to the long-time training of doctors, with recent advancements in hardware and algorithms, computer graphics can provide more understanding of the data and provide more insight than before.

The research found within this dissertation presents GPU accelerated visualization methods for daily clinical practice. In particular, the research focuses on neurostimulation therapy and visualization systems that support surgeons in their daily decision-making processes. Neurostimulation therapy has proven to be an excellent treatment for various movement disorders. It relies heavily on medical imaging and intraoperative data analysis, making it a good candidate for supportive visualization systems.

1.1 Motivation

In the following, I would like to outline the motivation for this dissertation. Visualization, and specific scientific visualization, is an exciting topic. Like the old saying, *A picture is worth a thousand words* visualizations can help to convey information that could stay hidden to the naked eye within the raw and unfiltered data. Techniques developed for scientific visualization can offer more in-depth insight into data and can be helpful or even mandatory.

Scientific visualization approaches have been adopted in the medical field for a long time. However, it is often the case that the gold-standard visualization found in many medical fields is rather old, and the adoption rate for newly developed techniques could be faster compared to other industries. While the experience practitioners have with the available techniques and the reliability of these long-established systems can explain the slow technological progress, especially when human lives depend on it, many treatments can adopt newer visualization approaches without bringing the patient at risk.

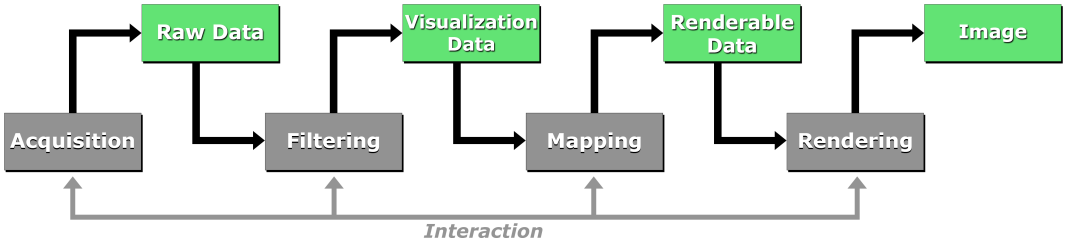


Fig. 1.1 The visualization pipeline is a widely adopted concept. The main stages are presented in this figure. The lower row depicts processes between stages, whereas the upper row represents the data states.

One of these fields is neurostimulation therapy, a treatment for various brain malfunctions investigated over the last half-century. In daily clinical practices, all kinds of patient data are collected, including, for example, imaging data of the patient, movement recordings, or data about brain activity. The data is used for assessing the patient, planning interventions, and post-operative care. In the grand scheme of medicine, neurostimulation therapy and the applied treatments are a relatively new field, still open to new methods and visualizations that aid in improving patient treatment.

The applied processing and visualization of the data are rudimentary and based on techniques used in other medical domains for ages. The possibilities provided by the collected data and the fascinating idea of improving patients' lives motivated this research. As a result, this dissertation focuses on visualization solutions for neurostimulation therapy in daily clinical practice by supporting practitioners in tedious tasks using state-of-the-art rendering techniques and hardware. The presented solutions assist surgeons daily and provide better and faster treatment to their patients.

1.2 Scientific Visualization

While visualization predates the modern computer by hundreds or even thousands of years, if we include, for example, prehistoric cave paintings [173], modern, computer-based visualization, in comparison, has a relatively short but distinguished history. Visualization in scientific computing as a discipline was introduced mainly in the 1980s [323]. With Workstations powerful enough for scientists to directly display graphics on the fly and supercomputers that could simulate complex phenomena not even thinkable of only a few years prior, it became apparent that computer-based visualization had a shining future.

Visualization aims to explain observations, makes predictions based on underlying data, and tries communicating theories [323]. While it is often divided into two disciplines, namely Scientific Visualization and Information Visualization, the boundaries between them are often more fluid than expected [239]. The main difference often broad up is the type of data visualized. Scientific visualization is based on physical data acquired by sensors, simulations, or databases, where the data stands in relationship to the real world. On the other hand, Information Visualization primarily uses abstract data to communicate abstract ideas. However, it is common that visualization techniques found in Information Visualization are adopted for Scientific Visualization.

A key concept of Scientific Visualization is turning the raw data into a visual representation that can be interpreted. This process is known as the visualization pipeline [119], depicted in Figure 1.1. Transforming unfiltered raw data into a representative structure plays a significant role in this dissertation.

Scientific visualization has much in common with modern computer graphics and owes much of its success to the rising popularity of other fields that utilize computer graphics, such as video games or movie effects that require graphical computational power, pushing the market forward. In contrast, these two fields try to communicate an experience to the user, utilizing computer graphics to display stunning visual effects. In scientific visualization, the goal is not to sell an experience but to display complex data for knowledge gain.

1.3 Neurostimulation Therapy

Neurostimulation therapy is a widely adopted form of treatment for diverse neuropsychological diseases. The term comprises multiple different types of treatments [99], for example, Motor Cortex Stimulation [290, 291], Transcranial Magnetic Stimulation [122], Spiral Cord Stimulation [200], or, as most important for this thesis, Deep Brain Stimulation (DBS) [226].

The first investigations in this field date back to the mid-20th century when John Farquhar Fulton [294] observed that the modulation of specific regions of the cerebral cortex could reduce anxiousness. Based on his findings, different forms of lobotomy treatments came up and were executed during the mid-1950s. However, this type of neurological treatment was already abandoned in the 1960s due to various side effects. The most noticeable side effect was the extreme shift in the patient's personality [88] due to the harmed brain tissue.

Even though lobotomy treatment has been canceled on a broad scale, the findings of these early days motivated many researchers to further investigate the treatment of deeper brain regions. One of these early pioneers was Robert G. Heath [127, 128, 126]. Heath aimed to understand and treat psychiatric disorders by implanting intracranial electrodes into the brain. For this purpose, Heath conducted multiple clinical studies and observed in one of them that the stimulation of specific regions of the brain could induce analgesia, the absence of pain [35]. The results of these early years reduced Deep Brain Stimulation as a treatment method only applied for chronic pain.

However, it took only a short time until researchers were interested in other neurological diseases and the applicability of DBS. One of the most prominent forms of movent disorders treated by DBS is Parkinson's Disease, which is also primarily featured in this thesis. Irving S. Cooper discovered the applicability of DBS for treating Parkinson's Disease in 1989 [81]. It is reported that Cooper had to halt during one of his surgeries due to a stroke within the thalamus of his patient. Later on, after his patient awakened, he discovered that his patient had suddenly been freed from various tremor effects due to the stimulation of the subthalamic nucleus. These astonishing findings for treating neurological movement disorders lead to decades of new surgical techniques for the stimulation of different brain regions [71, 28].

For some time, the interest in DBS surgeries was lost due to the introduction of levodopa replacement therapy [218], a medical treatment for Parkinson's disease without the need for any surgery. After it was apparent that long-time use of levodopa resulted in various side effects [224, 194], the interest in DBS rose once again.

More and more methods were developed for long-term DBS resulting in the electrode leds used today. Back in the rising days of DBS surgeries, the most significant concerns were the effects of misplaced leds in ablative techniques, which resulted in DBS replacing this method altogether [27, 26]. Modern stereotactic and image-based techniques [240, 79] were introduced to the planning phase as a measure of counteracting these concerns. For intraoperative verification, surgeons use real-time Micro-Electrode Recording (MER) [230]. MER provides neurophysiological data on the patient current brain activity, hence improving the certainty in the placement of the electrode led.

Due to its history and protracted development, DBS has established itself as one of the best treatment methods for multiple movement disorders [226]. Compared to other mentioned neurostimulation therapies, DBS has been approved for most types of disorders by many significant medical instances [320].

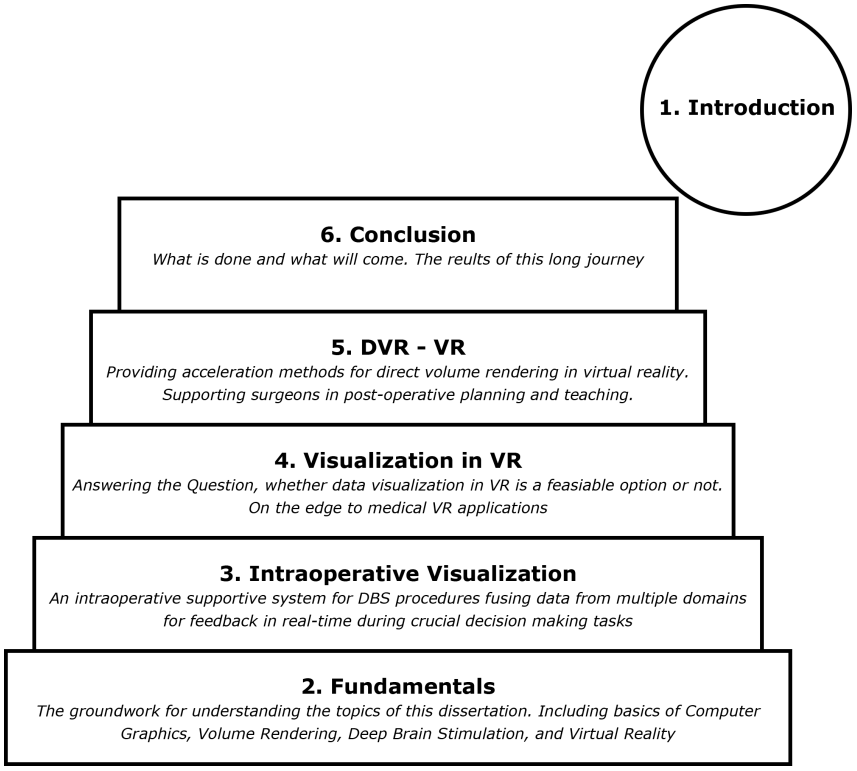


Fig. 1.2 The structure of this dissertation is designed as a pyramid. Starting from the bottom: Fundamental concepts necessary to understand the research areas are presented. Afterward, the research fields and results are reproduced. The final chapter concludes the work of this dissertation and presents future research topics.

1.4 Structure, Contributions and Publications

In the following section, this dissertation will focus on three aspects. The first is a description of the structure and how to read this thesis. The following part will focus on the overall contribution of the presented research, and the last part will outline the included publications and how these are embedded into this work.

The overall structure of this thesis is displayed in Figure 1.2. The presented structure is based on different topics of the included publications. The beginning chapters of this thesis introduce fundamental topics such as computer graphics, volume visualization, deep brain stimulation, and medical imaging. Each of these topics is essential for a better understanding of the research topics and contextualizing the presented results. After presenting the general background of this work, this thesis will focus on various visualization topics of medical data and specific neurostimulation procedures. The first topic covered is intraoperative visualization for DBS procedures. Key aspects of this research were the real-time analysis and

classification of brain activity in combination with interactive and understandable visualizations following the principles of the visualization pipeline.

Afterward, the dissertation will introduce a long and profound literature review on the topic of Virtual Reality (VR) and scientific visualization within immersive environments. The motivation behind this literature review was to investigate whether and under what constraints scientific visualization can be helpful in fully immersive systems. This chapter was part of an ongoing state-of-the-art report still in progress. The results of this review are the motivational foundation for the following topic of medical visualization in virtual environments.

Based on the findings of data visualization in immersive environments, the last topic focuses on Direct Volume Rendering (DVR) on VR systems using a Head-Mounted Display (HMD) as an output device. An important key aspect of this topic is the acceleration of DVR to fit VR systems. Two methods have been researched throughout this dissertation and are presented in detail. The problems of DVR on HMD systems are covered, and the question answered whether it is helpful for the context of neurostimulation therapy and the tedious planning phase or not.

Finally, a conclusion to the presented topics is given, including a discussion and outlook on future work.

Contribution

The presented research topics in this dissertation contribute to neurostimulation procedures, in particular deep brain stimulation procedures, in the form of a variety of visualization systems and rendering solutions to help practitioners in their daily tasks of treating patients with various movement disorders.

Chapter 3 introduces an intraoperative rendering and analysis system for DBS procedures. It can be used during the tedious decision-making tasks of tissue detection and electrode placement by providing new forms of visual feedback on the already present data. The system was presented at multiple international conferences [308, 312, 313], including computer science and medical conferences. Furthermore, it was awarded the best paper award on symposia for medical imaging systems [308]. The key focus of the system is the analysis and classification of microelectrode recording data. It provides a fast and reliable approach to tissue type detection with various forms of visualization for cross-validation by experts. As a visualization system, it is mandatory to provide surgeons with visual feedback during the procedure. The system supports various forms of data processing, not only for

the MRI data but also for preoperative imaging data. The imaging data in the form of CT and MRI scans are fused with the analyzed MRI data and results in a fast-to-interpret visualization. The fusion of CT and MRI data introduced the necessity to develop a real-time data registration approach using the GPU for acceleration. As a side product of the classification system, the system also provides a CT and MRI registration method, exceeding commonly available solutions.

With the rise of Head-Mounted-Displays on the commodity market, there was also an increased interest in VR applications for the medical field. A long and ongoing literature review of already published solutions was conducted to investigate the applicability of VR in scientific visualization. With the interest of enhancing the preoperative planning phase with supportive VR software, the question had to be answered, what constraints must be followed while designing such a system, and what research was already presented. The current state of this literature review is presented in this dissertation within Chapter 4. Based on the finding of this review process, is the motivation for the system and acceleration techniques presented in the following.

With the knowledge of the constraints and boundaries of VR and scientific visualization, the decision was made to design a rendering system for the preoperative planning phase of DBS surgeries that utilize already available imaging data. Chapter 5 focuses on the acceleration of direct volume rendering in virtual environments. Based on already established knowledge, the acceleration methods were designed to provide a fluid experience and reduce the possibility of cybersickness symptoms. Both approaches were published at international conferences [310, 309]. The resulting acceleration method is tailored toward state-of-the-art HMD devices and independent from the used graphics hardware. The technique is scalable to various hardware setups, be it low-end consumer hardware or high-end workstations.

To summarize the contribution of this dissertation: The presented systems and solutions in this work provide a significant impact on DBS surgeries in terms of the preoperative planning phase by providing a fully immersive visualization system for better spatial orientation and for the intraoperative use by offering a classification and visualization approach for the location of the target tissue, aiding the practitioner during their decision-making tasks.

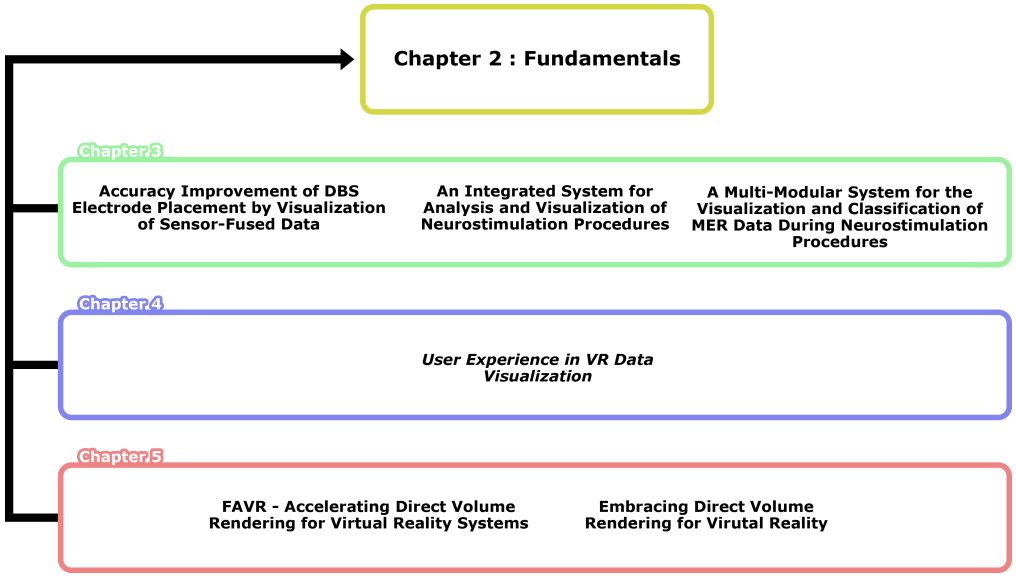


Fig. 1.3 Each research chapter is based on various publications released in this field, while the fundamentals chapter includes the merged related work from all publications.

Publications

This thesis is based on the publications listed below. Figure 1.3 displays the correlation between each chapter and the corresponding research or publications. Over the time working on the topics found in this dissertation, additional papers were published that were excluded from this work due to not fitting into the context of this work.

Each publication found in this dissertation is joint work with my supervisor Jens Krüger, and other researchers and field experts. The chapters are based on the already published content. Whenever I felt that chapters could be more clarified by additional information, I improved the already available material with additional explanations, examples, and results.

As I am the first author for all the included publications, the presented research was mainly contributed by myself. The presented results originated from an iterative process and were, therefore, published in multiple waves in different venues as the research for each topic progressed.

1. **Andre Waschk**, *Yaroslav Parpaley, and Jens Krüger*. "Accuracy Improvement of DBS Electrode Placement by Visualization of Sensor-Fused Data". In: *Proceedings of the GPU Technology Conference, 2016*

The content of this publication is reproduced in Chapters 2, and 3 in combination with other here-listed publications on this topic. The already published information on this topic is extended within this dissertation.

2. **Andre Waschk**, *Yaroslav Parpaley, and Jens Krüger*. "An Integrated System for Analysis and Visualization of Neurostimulation Procedures". In: *Proceedings of the Supercomputing Workshop on Medical Image Analysis and Visaulization, 2016*
★ *winner best-paper award*

   The content of this publication is reproduced in Chapters 2, and 3 in combination with other here-listed publications on this topic. The already published information on this topic is extended within this dissertation.

3. **Andre Waschk**, *Yaroslav Parpaley, and Jens Krüger*. "A Multi-Modular System for the Visualization and Classification of MER Data During Neurostimulation Procedures". In: *EMBC, 2021*
DOI : 10.1109/EMBC46164.2021.9631088

   The content of this publication is reproduced in Chapters 2, and 3 in combination with other here-listed publications on this topic. The already published information on this topic is extended within this dissertation.

4. **Andre Waschk**, *Jens Krüger*. "FAVR - Accelerating Direct Volume Rendering for Virtual Reality Systems". In: *IEEE Visualization Conference (VIS), 2020*
DOI : 10.1109/VIS47514.2020.00028

   The content of this publication is reproduced in Chapters 2, 4 and 5 in combination with other here-listed publications on this topic. The already published information on this topic is extended within this dissertation.

5. **Andre Waschk**, *Jens Krüger*. "Embracing Direct Volume Rendering for Virutal Reality". In: *WSCG, 2022*
ISBN : 978-80-86943-33-6 pages 128-134

   The content of this publication is reproduced in Chapters 2, 4 and 5 in combination with other here-listed publications on this topic. The already published information on this topic is extended within this dissertation.

The following articles were also published during the time of this dissertation but are excluded. While the topics of these publications are fascinating, they are not relevant for this dissertation.

1. **Andre Waschk**, *Jens Krüger*. "Automatic route planning for GPS art generation". In: *Proceedings ACM SIGGRAPH Asia 2018 (Technical Briefs), Publisher: ACM, December 2018*
   DOI : 10.1145/3283254.3283275
   ★ *selected as top technical brief for journal publication*

2. **Andre Waschk**, *Jens Krüger*. "Automatic route planning for GPS art generation". In: *Computational Visual Media, September 2019, Volume 5, Issue 3), Publisher: ACM, December 2018*
   DOI : 10.1007/s41095-019-0146-z

Chapter 2

Fundamentals

Within this upcoming chapter, the reader will find various topics which can be seen as the fundamentals necessary to understand a large amount of the covered research topics found throughout this work. The core topic of the selected research focuses on scientific visualization, with a strong emphasis on volume visualization. As scientific visualization is deeply connected with computer graphics topics, the first portion of this chapter will focus on a brief history of computer graphics, the programmable graphics pipeline, and visualization techniques used in this dissertation.

Specifically, this chapter will begin with a brief history of computer graphics, the rising need for dedicated GPUs over the years, and the shift from a fixed rendering pipeline to a freely programmable graphics pipeline. The rapid advancements in computer graphics and computational power available to researchers benefitted the field of scientific visualization with an increasing interest in researching various rendering techniques throughout the decades to visualize the increasing number of collected data. Numerous fundamental computer graphics algorithms were developed in the early 1970s to 1980s, and a considerable number of these are still essential today and are found in many systems. For the reader of this work, it can be helpful to gain some common knowledge of the beforementioned topics as they help to understand the following research topics better. Especially the shift in programming hardware of the late 1990s and early 2000s dramatically changed and broadened the application fields of computer graphics in the scientific field.

Following the fundamentals of computer graphics and the programmable rendering pipeline, the focus is shifted toward the topic of volume visualization. As a large portion of the presented research focuses on volume visualization, it is crucial to understand some of the basic concepts; such as the differences between indirect

and direct volume rendering, the typical rendering pipeline for real-time volume visualization, and how scalar values found in most volumetric data sets are mapped to colors displayed on the screen. A significant emphasis is laid on direct volume rendering and how these concepts are realized on state-of-the-art GPUs. While being a computationally expensive approach, direct volume rendering offers a fast and reliable approach for generating images of sparse volumetric data sets in real time. Due to its complexity, it is essential to introduce acceleration methods that have been researched over the years. Most of these acceleration methods are applied in the presented systems and a fundamental for most volume rendering systems.

As volume visualization needs volumetric data to be visualized, this chapter will also present a brief overview of typical volumetric data sets used in this work. The main target area of this research is the medical treatment procedure known as DBS. Therefore, the data that has to be visualized also results from this area. In particular, data acquired from CT and MRI scans are presented. Both data domains play an essential role in DBS as they are necessary to create a treatment plan for the patient, and neither of them can be left out as both have personal strength in presenting different tissue types. For this work, a brief history of both techniques will be presented as additional information about both domains' individual strengths and drawbacks.

The last topic that is covered in this chapter is virtual reality. While VR seems to be out of place as a topic at first, as the previous topics largely surrounded the medical space, VR plays a significant role in the presented topics. A significant portion of the research found in this work focuses on the visualization of data, especially medical data, in an immersive virtual environment. As the term virtual reality covers a large field, it is essential to introduce the topic briefly. As this work aims to convey the benefits of visualizing data in an immersive environment and present the advantages of exploring medical data in an immersive context, it is crucial to explain terms such as immersion, presence, and engagement. Furthermore, it can be helpful to present different types of VR and discuss cybersickness and how it can be avoided.

The presented information is collected from various sources referenced across the chapter. Since this chapter only offers a brief overview of each topic, it is recommended to the interested reader to dive deeper into the referenced material if more detailed information on a specific topic is desired.

Fig. 2.1 The famous *Utah Teapot* by Martin Newell [288] lid with three different lighting models. Left: Flat Shading (single color per triangle) Middle: Gouraud Shading (per vertex lighting) Right: Phong Shading (per fragment lighting).

2.1 Computer Graphics

What most people reading this work understand under the term computer graphics is properly the generation or synthesizing of digital images based on geometrical data [248]. One of the first occasions this term was broad up was by William Fetter in 1960, when he was asked to describe what he was doing at Boeing at the time [248]. Besides simply generating any kind of image with the support of computer hardware and algorithms, more often, the accuracy of those generated images is of interest.

While the term computer graphics has been around since the 1960s, it was in the 1970s that computer graphics started to grow, and many foundations were laid out for what we know today under this term. Before this time, computer graphics were out of reach for most users and even researchers. The hardware required to compute images was too expensive, and users had to pay for time on large clusters or invest in costly computers for this task. Even though computer graphics and research in this field were expensive during this period, many of the essential methods and techniques still used today were developed back then. Especially in the late 1960s, researchers at the University of Utah developed illumination models (Figure 2.1), surface algorithms, or rendering techniques for polygonal surfaces, including widely known names such as Phong Bui-Tuong [228], Henri Gouraud [113], James Blinn [37], and Edwin Catmull [56].

The last mentioned, Ed Catmull, is also known for generating one of the first 3D animations that have found their way into movie production. In the 1970s, Edwin Catmull and Fred Parke [55] animated and rendered a model of a human hand, unprecedented to this day (Figure 2.2). Their animation was later used in the film Futureworld [167], making it the first computer-generated animation on the big screen. Back in the day, the generation of this short animation clip was far more complicated than nowadays. Since displays could not keep the image long enough

Fig. 2.2 One of the earliest 3D animations created by Catmull et al. developed during his Ph.D. thesis [57]. The model was a real world sculpture. Each vertex was manually picked and triangulated on the sculpture before being scanned for digital recreation.

on the screen, long-time exposure imaging of Cathode-ray tube (CRT) displays was mandatory to capture single frames of the animated hand. Afterward, the pictures of each frame had to be stitched together to generate the final video in post-production.

Based on the experience of early research in computer graphics and the computational findings, it became clear that special, dedicated hardware was mandatory to accelerate the operations of these applications. The first step toward dedicated graphics hardware originated between the 1970s and the late-1980s. The released hardware focused on two-dimensional bitmaps of text and shapes to accelerate the video output and was found in many video game consoles of the corresponding generation [13].

The rise of 3D acceleration hardware only started a few decades ago. With improvements in manufacturing in the 1990s, more graphics accelerators appeared on the market. In the early- to mid-1990s, the first 3D accelerators entered the commodity market. It only took a few years until most personal computers included 3D acceleration hardware and made it possible for anyone to generate computer graphics on commodity hardware. One of the biggest GPU manufacturers of nowadays also entered the market in the late 1990s, Nvidia [199], and introduced the term GPU to the broad public.

In the early 1990s, the programming language OpenGL [154] appeared and became famous for writing code for 3D accelerator cards. The introduction of OpenGL spared users from writing complex assembler code and made it more accessible to new developers. Back in the day, OpenGL only provided a set of fixed functions and basic lighting models to accelerate and process polygonal models. This fixed set of functions became known as the fixed-function pipeline. Due to the strive

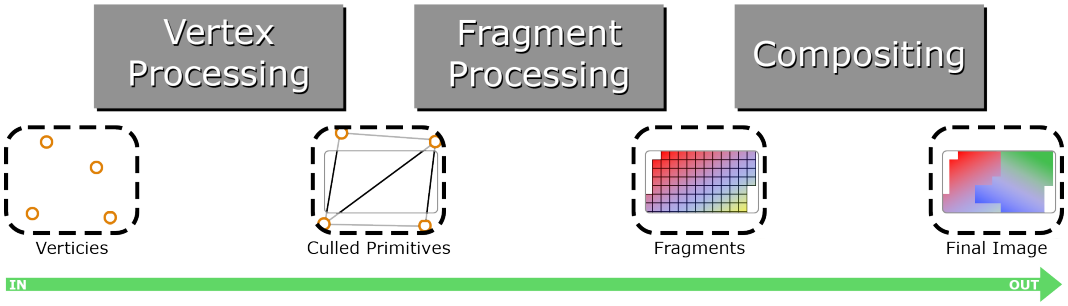


Fig. 2.3 The incoming set of vertices are getting processed within the Vertex Processing stage resulting in culled primitives. These primitives get rasterized and result in multiple fragments which get composited with already stored data for a final image.

of consumers and developers for more realistic images, the fixed-function pipeline of early 3D accelerators was fast abandoned for a fully programmable graphics pipeline.

The shift from 2D accelerators to 3D accelerators and from a fixed function pipeline to a fully programmable rendering pipeline in the late 1990s laid the foundation for many research aspects in this dissertation. Thanks to the flexibility of modern GPUs, we can use the computational power of its hardware for various problems that scale on parallel processing units, including volume visualization.

2.1.1 Programmable Graphics Pipeline

GPUs are included in nearly every modern personal computer. Compared to the Central Processing Unit (CPU), which primarily consists of a few cores needed to compute various general algorithms, a GPU can be seen as a highly efficient and optimized piece of hardware with multiple thousands of processors designed to generate computer graphics. While CPUs are mostly faster on a core-to-core basis, they are not designed to compute computer graphics and linear algebra on the level GPUs, whose architectures are based on parallel processes.

The most significant innovation of the last decades for GPUs was shifting from a fixed-function pipeline to a fully programmable rendering pipeline (Figure 2.3). By switching to a programmable rendering pipeline, developers could now write customized code that gets executed on the GPU. The flexibility of customizable code provided several advantages compared to the traditional fixed-function pipeline and opened the door for various new applications. The access to thousands of processors made GPUs far superior to CPUs for many use cases in which programs

can be executed in parallel. However, GPUs and the rendering pipeline have several limitations in how we can access and use the available resources.

First, GPUs are designed to generate raster images of polygonal geometry. Therefore complex geometry, such as NURBS [246] or subdivision surfaces [58], must be tessellated into small triangles to be processed. For the visualization of three-dimensional scalar volume data, as used in this dissertation, other solutions have to be found without tesselating the data itself. Nevertheless, modern GPUs are flexible enough to provide plenty of options to visualize scalar field data.

This chapter will further outline a brief overview of the rendering pipeline and the processing stages found within this pipeline. While the term "*programmable pipeline*" could indicate that every part of the pipeline is programmable or interchangeable, the order in which these stages are executed remains fixed and can be described as seen in Figure 2.3. Just specific stages of the pipeline can execute self-written code, while other parts rely on parameters that can be changed but are still fixed.

The input for the rendering pipeline is a stream of vertex data representing a polygonal model. Once the vertices have been processed, the result is a raster image in the local memory of the GPU, which can be displayed on the screen or used for additional computations.

As implemented on most modern GPUs, the rendering pipeline consists of three stages: vertex processing, fragment processing, and Compositing. The following sections explain each stage briefly with its functionailty.

Vertex Processing

The *vertex processing* stage has the function to process the incoming stream of ordered vertices. It returns a set of *geometric primitives* for the rasterization and further processing on a per-fragment level. The operations executed in this stage are so-called *per-vertex operations*. Each vertex within the stream has a set of attributes, such as the *model-space position*, the *normal vector*, *texture coordinates*, *color values*, and several additional values determined by the programmer.

Within the vertex program, the vertex data, in particular the position and normal, undergoes a handful of linear transformations from the original model space to the world space of the scene and, subsequentially, to the camera space of the screen. These linear transformations include computations such as scaling, shearing, rotation, or translation and are mostly solved using precomputed *4 x 4* matrices such as the well

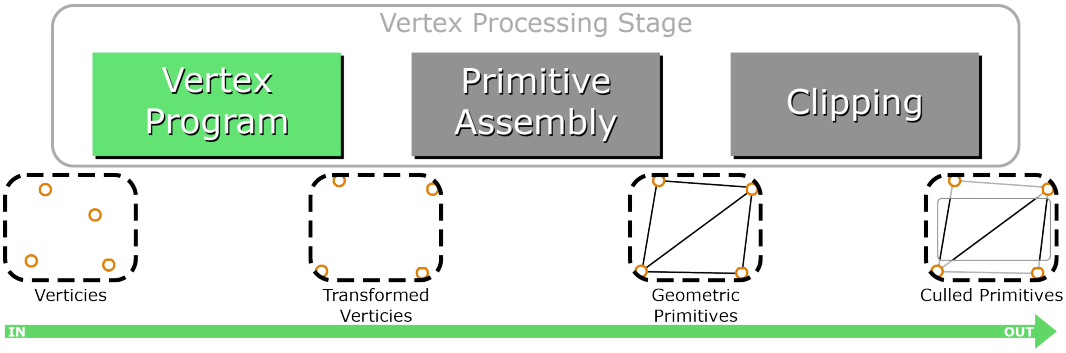


Fig. 2.4 The incoming stream of ordered verticies gets transformed in the correct view space of the screen. After transformation geometric primitives are build and clipped to the dimensions of the screen. The vertex program is a custom, user written, code that gets executed on every single fragment.

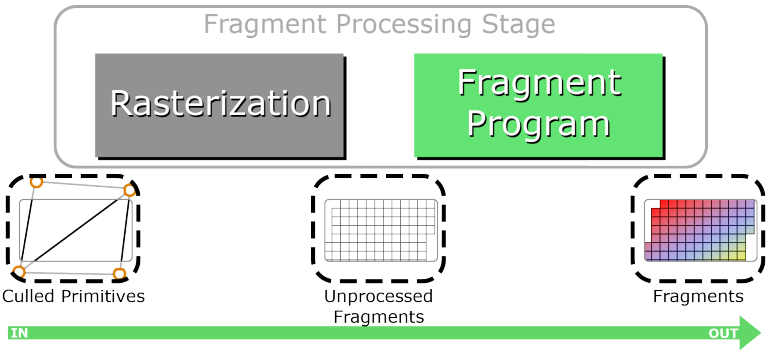


Fig. 2.5 During the rasterization stage the incoming primitives get transformed into a set unprocessed fragments that are covered on the screen. During the fragment program a custom program gets execute to determine the final color of the fragment.

know *model*, *view*, and *projection* matrices. For fewer computations, these matrices are often combined into a single *modelviewprojection* matrix if the computation allows it.

Once the vertices have been transformed into camera space, the primitive assembly groups sets of vertices to geometric primitives, which can range from points to lines or, most likely, triangles. These geometric primitives are clipped and culled to the corresponding screen space and handed over to the fragment processing stage for rasterization.

Fragment Processing

The first step of the fragment processing stage is the rasterization of the screen-space primitives. Therefore, the rasterizer generates fragments for every pixel of

the target image covered by the primitive. These fragments include the interpolated attributes provided by the primitive, including data such as world space position, normal of the vertex, texture coordinates, depth value, stencil value, and much more. The interpolated fragment data is further processed in the fragment program, a custom-written program to compute all kinds of algorithms. Technically the fragment program is an optional step of the rendering pipeline. If it is neglected, the final color of the fragment is unknown, while other attributes, such as the depth or stencil value, are passed on. Often the fragment program is used to compute a realistic-looking lighting model for the color of the final pixel. However, it can also solve problems like edge detection or shadow maps. The computationally expensive traversal of scalar volumetric data is also executed within the fragment program on a per-fragment level.

Texture data can be fetched within the fragment program, including 1D, 2D, and 3D textures, including the three-dimensional scalar field data mandatory for volume visualization (see Section 2.2)

The outgoing result of fragment processing is a set of pixels covered by the incoming primitive with the corresponding color values and attributes. Since multiple primitives, and therefore multiple fragments, can contribute to a single pixel of the final image, it is necessary to compose the data.

Compositing

The final fragments generated by the fragment processing stage have to be written to the frame buffer. The frame buffer represents the final image, a two-dimensional array with the stored pixel data and their corresponding attributes of color, alpha, and depth values. Newly generated fragments are written to the frame buffer and modify the stored data. How the already stored information is modified is defined by a set of tests a fragment must pass. The tests applied consist of the alpha, stencil, and depth tests. After leaving the fragment processing stage, all these tests are executed in a set order and are known as frame buffer operations.

The **alpha test** compares the incoming α value against a specified reference value. Its original idea was to immediately discard completely transparent fragments, with an α value of 0, skipping the write operations to the frame buffer. Since then, other usages for the alpha test have been found, making it useful for a handful of applications.

The **stencil test** can be seen as a test against a specified mask for the cut-out of the set stencil values. This mask is stored in a separate portion of the frame buffer and is used for various effects in computer graphics.

The order in which primitives are generated is somewhat arbitrary. Therefore it can and will happen that fragments are generated that are already covered by another primitive. To determine which fragment is in front, the **depth test** compares the positional information of two fragments based on their depth value. Commonly, lower depth values indicate fragments closer to the camera; therefore, fragments with larger depth values than the already stored ones can be discarded due to being occluded.

As mentioned, the frame buffer operations are typically executed after fragment processing. However, there is an exception for the depth test. Since occluded fragments get discarded by the depth test, it is desirable to exclude the execution of the fragment program for pixels that do not contribute to the final image. Fragment programs can be expensive in terms of execution time. Utilizing early z-tests, the rendering pipeline can skip the execution of the fragment program if it determines that the depth test would result in discarding the current fragment. However, developers can not determine whether early z-tests get executed or not. The graphics driver determines the execution of the early z-test depending on the fragment program and the used hardware.

Once a fragment passes all tests, it can be written to the memory. For transparent objects, alpha blending can be applied. Alpha blending provides various functions that determine how the color values of new fragments and the already stored pixel should be blended based on the α.

After these steps, the final image is stored within the frame buffer. This frame buffer can be displayed on the screen, reused as texture for other fragment programs, or read back to the CPU's main memory.

2.2 Volume Rendering

Volume visualization describes a set of techniques in scientific visualization for rendering three-dimensional scalar fields. The scalar field data used in this dissertation is commonly acquired from medical imaging, such as Computed Tomography, Magnetic Resonance Imaging, and Diffuse Tensor Imaging. However, volume data can also be acquired by simulations or other forms of sensor data. Each of these

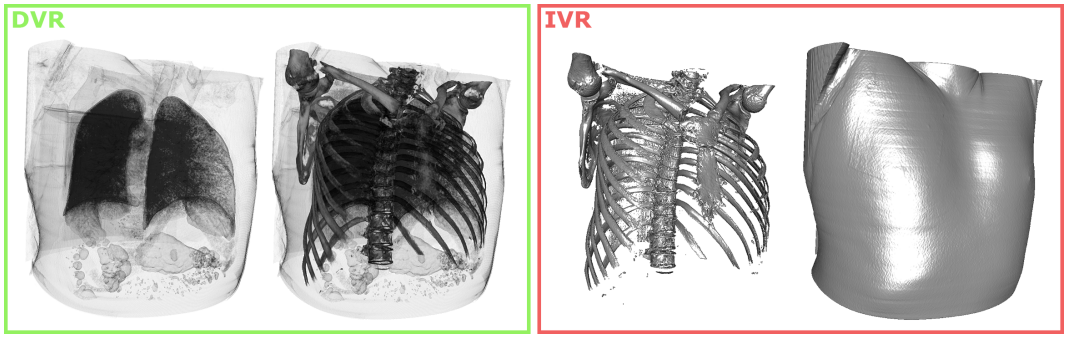


Fig. 2.6 The rendered images display two variants of volume visualization. The two left images use direct volume rendering with a transfer function, providing inside into the raw data without any preprocessing. The two right images represent indirect volume rendering using iso-surface extraction, generating one single polygon mesh of a given iso-value.

medical data domains plays a part in the crucial planning phase of neurostimulation procedures.

Within scientific visualization, volume visualization is often differentiated between two types of algorithms; Indirect Volume Rendering (IVR) and Direct Volume Rendering (DVR).

Both types of volume visualization aim to generate a two-dimensional image of the complex three-dimensional scalar field. While IVR requires the tedious task of extracting triangulated mesh structures using isosurface algorithms, DVR allows directly visualizing the raw data without any preprocessing.

Even though IVR requires preprocessing, the outcoming meshes are faster to render than any direct volume rendering approach. Especially since modern GPUs are tailored toward processing a large number of triangles simultaneously, as established in section 2.1. Nevertheless, the meshes used for IVR provide much less information about the scalar field. Due to the extraction of isosurfaces, users must know what kind of data they want to see before any three-dimensional visualization is available. Since the set iso-value predefines the extracted surfaces, other information stored in the data set gets lost in this process.

On the other hand, direct volume rendering visualizes the raw data and uses a transfer function (Section 2.2.3) to adjust the visualization on the fly. Working with the raw data comes with an increased computational cost but can be seen as worthwhile due to the increased insight into the data (Figure 2.6).

The reader can find background information about DVR, essential rendering, and acceleration techniques in the following sections. Most of the following are based on

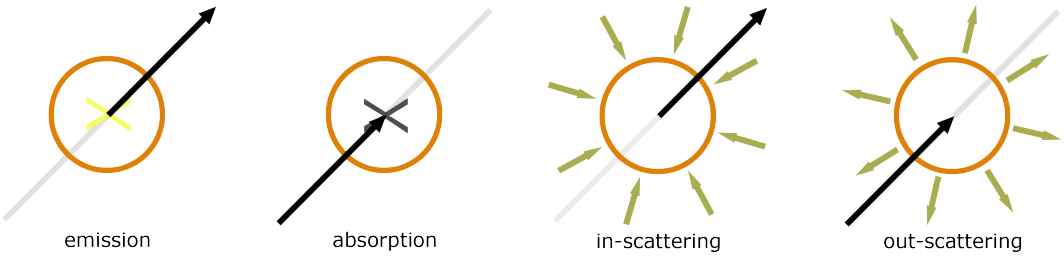


Fig. 2.7 The radiance is affected based on the participating media. Media can emit light, absorb incoming light rays, scatter incoming light from various directions into another direction or distribute incoming light from one direction in multiple new directions.

the book Real-Time Volume Graphics by Engel et al. [120], and I highly recommend reading the named book for more detailed information on the presented topic.

2.2.1 Mathematical Background

Volume rendering follows the principles of geometric optics and the traversal of light across a straight line through different mediums. Geometric optics assumes that the ray of light is propagated within the volume through interaction with the participating medium. The property used to describe the light energy is the radiance I, defined by the radiative energy per unit area, per solid angle, and per unit of time. During traversal of the volume, three effects can influence the radiance along the ray (Figure 2.7). These are:

- **Emission:** Material which by itself emits light energy, increasing the radiance of the light ray.
- **Absorption:** Material that absorbs radiative energy and converts it into heat
- **Scattering:** Differentiated into in- and out-scattering material, where in-scattering material adds more energy by redirecting energy into the ray, and out-scattering material removes energy by distributing energy into other directions.

Solutions considering the complete equation of light transport are computationally intensive and mostly not feasible for interactive volume visualization. In many applications, as for this dissertation, simplified models are used only considering the emission and absorption effects reducing the complexity for faster results.

The equation necessary to understand volume rendering is the well-known volume-rendering integral. Written in its differential form, it can be defined by Equation 2.1 and is visualized in Figure 2.8.

Fig. 2.8 Incoming radiance enters the volume at $s = s_0$. While traveling through the volumetric structure, the radiance gets absorbed dependent on the material. At $s = D$, the remaining radiance leaves the volume and is projected to the eye.

$$I(D) = I_0 T(s_0, D) + \int_{s_0}^{D} q(s) T(s, D) ds \tag{2.1}$$

The incoming radiance at position $s = s_0$ is defined by I_0. It describes the radiance entering the volume from behind and is attenuated by the volume's transparency term. The result $I(D)$ represents the radiance leaving the volume at $s = D$ in the camera's direction. The integral in Equation 2.1 describes the emitted light within the volume, defined by the source term $q(s)$, attenuated by the modeled transparency term $T(s_1, s_2)$. The transparency term $T(s_1, s_2)$, as further described in Equation 2.2, defines how far light may travel within the medium before it is fully absorbed and is also known as the optical depth. Small values for the optical depth indicate a relatively transparent material, while high values correspond to opaque material encapsulated within the absorption coefficient k.

$$T(s_1, s_2) = e^{-\int_{s_1}^{s_2} k(t) dt} \tag{2.2}$$

In most applications, the volume rendering integral can not be computed analytically, and numerical solutions are applied to approximate the integral as close as possible. Typically, the integral is discretized and split into several smaller subsequent intervals s_i. The location s_i on the ray range between s_0, the starting location, and $s_n = D$, the endpoint of the ray. In most applications using a discretized solution of the rendering integral, the radiance contribution is also known as the color contribution, noted as

$$C_i = \int_{s_{i-1}}^{s_i} q(s) T(s, s_i) ds \tag{2.3}$$

with $T_i = T(s_{i-1}, s_i)$ being the transparency function. With $c_0 = I(s_0)$, the volume rendering integral in Equation 2.1 can be written as

$$I(D) = \sum_{i=0}^{n} C_i \prod_{j=i+1}^{n} T_j \tag{2.4}$$

The most common approach, as it is used for most texture-based volume rendering systems, is a Rieman sum to approximate the volume rendering integral. the light ray passing through the volume is divided in n equidistant segments of length $\Delta x = (D - s_o)/n$. The color contribution is defined by $c_i \approx q(s_i)\Delta x$ and the transparency of the ith segment is defined by $T_i \approx e^{-k(s_i)\Delta x}$, where the transparency term T_i is often replaced by the opacity term $\alpha_i = 1 - T_i$.

The discretized volume rendering integral in Equation 2.4 can be further simplified by splitting it into a handful of more straightforward operations, also known as compositing schemes. Two compositing schemes have been established over the years: front-to-back compositing and back-to-front compositing. Instead of the accumulated radiance I or the contributed radiance c as found in the previous Equations, the color value C is now represented by a vector storing the individual red, green, and blue color channels and are premultiplied with the corresponding opacity value α. The color value C is subdivided into C_{dst}, the target color stored in memory, and C_{src}, the newly sampled source color of the volume.

The natural way of those two compositing schemes is the back-to-front composition. It is similar to how natural light travels from its emitting source into the receiver's direction. Transferring the back-to-front concept to volume rendering can be defined as a ray starting from the backside of the volume and traveling through the volume toward the target camera. Similar to the illustration in Figure 2.8. While traversing the volume by discrete equidistant steps, the color values of the volume are accumulated until the exit position at the front of the volume is reached. Mathematically the compositing function can be defined by

$$C_{dst} \leftarrow (1 - \alpha_{src})C_{dst} + C_{src} \tag{2.5}$$

This equation can be interpreted as follows; The final color C_{dst} of the ray is the result of the already accumulated color value C_{dst} weighted with the source's transparency value α_{src} added by the incoming source color value C_{src}.

The alternative compositing scheme, the front-to-back approach, inverts the sampling direction. Instead of following the light ray from its origin toward the

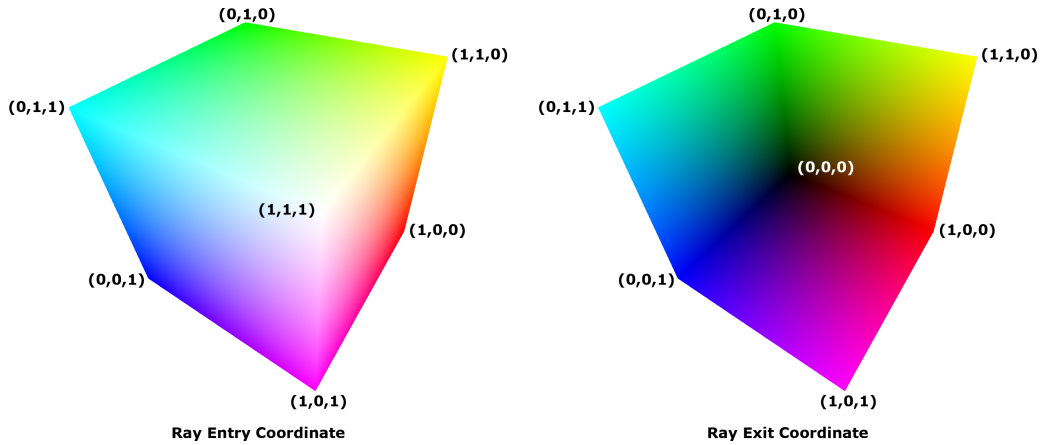


Fig. 2.9 Bounding geometries are typically used for ray-caster approaches, the ray's direction is computed by subtracting the color of a fragment found in the back face image from the color of the fragment in the front face image. Adopted and recreated from the work of Krüger et al. [160]

terminating location, this approach starts at the camera and traces rays into the scene through the volume. This approach can be defined by the Equations:

$$\begin{aligned} C_{dst} &\leftarrow C_{dst} + (1 - \alpha_{dst}) + C_{src} \\ \alpha_{dst} &\leftarrow \alpha_{dst} + (1 - \alpha_{dst}) + \alpha_{src} \end{aligned} \tag{2.6}$$

Compared to the back-to-front approach, the front-to-back scheme needs to keep track of the accumulated opacity. While, at first glance, this additional storage requirement can be seen as a significant drawback of the front-to-back approach, in reality, it is one of the critical pieces of information making many acceleration methods feasible and possible.

2.2.2 Techniques

There are two widely spread concepts for the visualization of scalar volumetric data, Indirect Volume Rendering (IVR) and Direct Volume Rendering (DVR). This dissertation mainly focuses on utilizing DVR for visualizing volumetric data sets. While IVR results in much faster visualizations, it is bound to mandatory preprocessing stages and provides less flexible insight into the original raw data than DVR.

DVR can be grouped into different visualization techniques. The two most prominent ones are slice-based and ray-casting approaches. Of these two, slice-based algorithms are the older ones and were mainly used at the time when the

fixed-function pipeline was still in use. The volume rendering solutions found in this dissertation are based on the more modern ray-casting approach. While this section will not provide an in-depth explanation of slice-based solutions, some benefits, and drawbacks, compared to ray-casting, have to be mentioned. Readers can find more details on slice-based volume rendering in the book by Engel et al. [120].

For a long time, slice-based renderers have been adopted to a large degree. Due to being compatible with the fixed function pipeline, it was for a long time the only solution for volume rendering on GPUs. Even today, it is often used as a fallback method for devices that do not provide enough computational performance for ray-casting approaches or do not support the mandatory shader functions. Its success was further supported by the fast access to texture memory of GPUs and the built-in bilinear and trilinear interpolation methods. Compared to CPU solutions, the processing performance of pixels in an image was much faster on GPUs and their rasterization units, making slice-based volume renderer a good solution for the time.

Nevertheless, slice-based approaches had several drawbacks. Due to being an object-space approach, the number of slices rendered is determined by the resolution of the volumetric data set. Most slice-based solutions are back-to-front approaches, implicating that many fragments are immediately overwritten on successive rendered slices (Figure 2.10). Furthermore, the back-to-front process limits the integration of many acceleration methods that have established themself as the gold standard for ray-casting solutions (Section 2.2.2).

Ray Casting

The fundamental concepts of ray-casting algorithms have been around since the 1980s [174, 175]. However, these concepts were traditionally computed on CPUs as GPUs were not advanced enough, or even an idea, to provide powerful parallel processors. It took several years until ray-casting could be adopted on GPUs. As mentioned in Section 2.1, it was only in the late 1990s that the first GPUs with programmable rendering pipelines appeared on the market, and just in 2003, the first implementations of GPU accelerated ray-casters were published [160]. Most algorithms for GPU ray-casters build upon long-time published CPU methods adopted to the parallel streaming processors of GPUs.

The fundamental idea of ray-casting is to evaluate the volume-rendering integral directly by tracing rays through the volume. Compared to the natural direction of light, ray-casting does not follow a back-to-front approach chasing the rays from

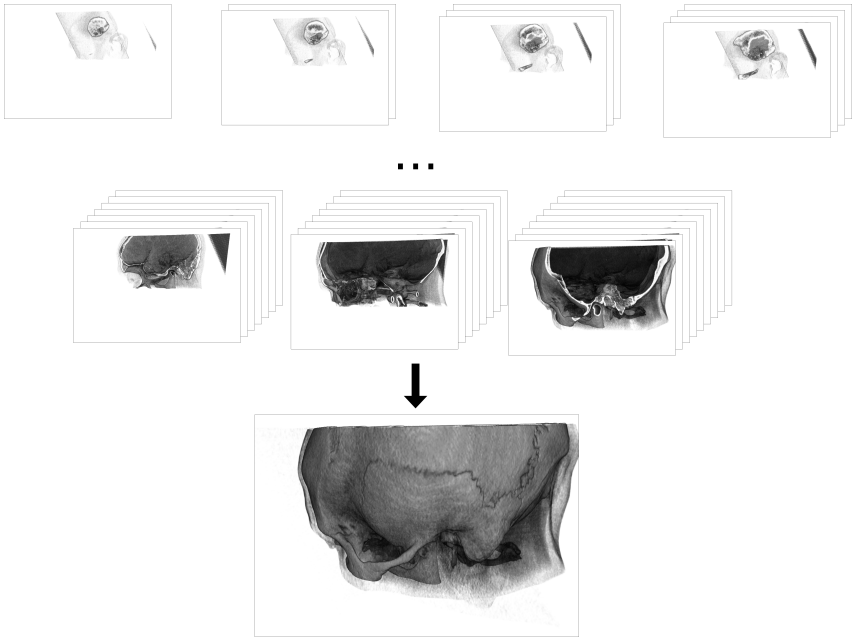


Fig. 2.10 Slice-based volume renderers often use a back-to-front compositing scheme. This scheme results in many fragments that get processed during the first steps of the image processing that get overwritten by later processed slices introducing a considerable overhead to this approach.

the light source to the camera. Instead, rays are emitted from the camera position, shot through each screen pixel, and traced through the volume, making ray-casting a front-to-back algorithm. During the volume traversal, the data is sampled at discrete steps along the ray. The sampled value of the scalar field has to be mapped to a color value C for the already described compositing schemes. Transferring from a scalar data value to a four-dimensional color value (red, green, blue, and alpha) is done by transfer functions.

Due to being a front-to-back compositing scheme, the newly sampled color values C_{src} on each step along the ray influence the accumulated and stored color and transparency values separately. Compared to back-to-front solutions, as found often in slice-based renderers, ray-casting approaches provide several opportunities for integrating acceleration methods using the separately stored transparency values. Some of the most common acceleration methods are found in Section 2.2.2.

The following components describe the ray-casting algorithm:

- **Ray Set-up:** The first step is to set up the ray according to the current camera position. Most approaches use a bounding cube enclosing the entire volume and mapping the cube's corners to possible texture coordinates of the 3D

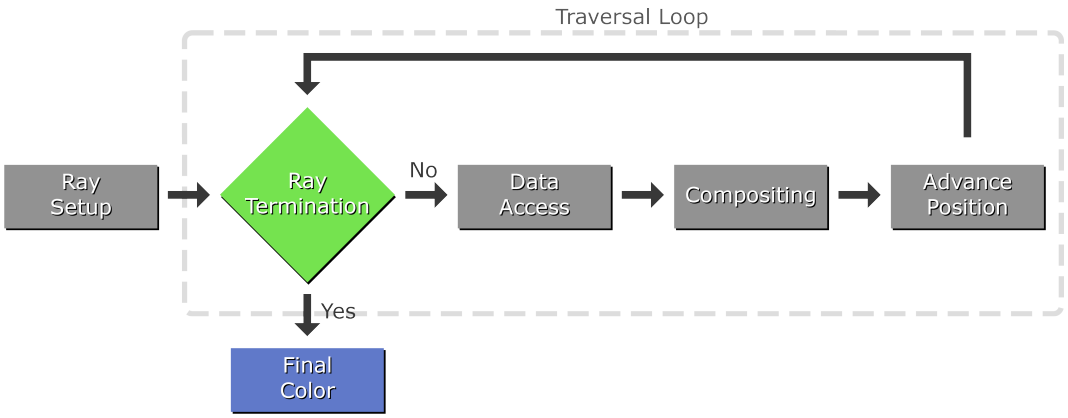


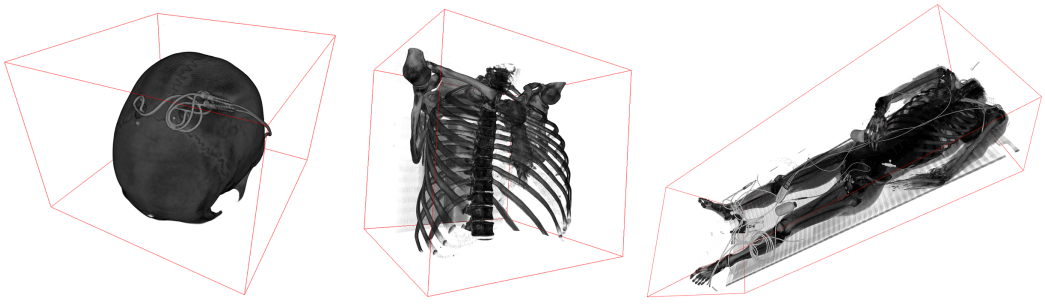
Fig. 2.11 Volume ray-casting consists of the ray setup stage to determine the starting position and direction of the ray and a traversal loop that samples, composites, and advances the ray position. In the end, the final color of the fragment is returned.

texture. The entry and exit coordinates of each pixel can be taken from the stored bounding geometry, and the ray's direction is computed accordingly.

- **Traversal Loop:** The primary iteration loop of any ray-caster. It is used to sample the ray at discrete intervals and compute the volume-rendering integral. The following steps get executed during each iteration of the loop.
  - **Data Access:** The scalar field is sampled at the current ray position, possibly involving interpolating the values found in the cartesian grid. The sampled value is classified based on transfer functions to determine the correct color and opacity value.
  - **Compositing:** based on the front-to-back compositing scheme, the stored color, and opacity values are updated with the current data.
  - **Advance Ray Position:** The following sampling location on the ray is computed. Depending on the implemented acceleration methods, this can be either a straightforward discrete step on the ray, a mechanism to skip a more significant part of the ray, or a reduced step on the ray for more detailed sampling.
  - **Ray Termination:** The loop terminates once the ray leaves the data set. Whether the following sampling location is still inside the data set can be determined by comparing the already traveled length of the ray with the length between the entry and exit position of the fragment. Additonal termination conditions can also be added to the loop—for

example, early-ray termination tests (see Section 2.2.2) or a maximum number of iterations. If the ray does not terminate, the loop is started again with the data access.

Acceleration Methods



(a) (b) (c)

Fig. 2.12 Various examples of direct volume rendering results from medical data. a) CT scan after a DBS procedure for verifying the led position. b) thorax scan for the assessment of injuries. c) Full-body scan with additional contrast agent.

Direct volume rendering is a computationally expensive visualization approach for three-dimensional scalar fields to approximate the volume-rendering integral. Over time, various acceleration methods have been established that are used in most rendering systems. Different acceleration methods are used for the visualization systems found in this dissertation. Therefore, the following will briefly present an overview of the acceleration methods.

The ray traversal is the most costly part of direct volume rendering. It can consist of thousands of texture fetches, dynamic branches, gradient computations, and lighting models. Many acceleration methods have focused on per-ray acceleration methods to reduce the complexity of the ray traversal. Some essential techniques are early ray termination, empty space skipping, and a group of progressive rendering systems.

- **Early Ray Termination:** Due to traversing the volumetric data from front-to-back, it is straightforward to add an additional stopping criterion besides leaving the volume. As the opacity value is accumulated during the traversal and updated on each step, it is possible to evaluate how opaque the fragment is. Suppose the opacity value comes close to a set limit, typically a value close

to 1.0. In that case, the traversal loop can be terminated as additional steps along the way would not influence the final color.

- **Empty Space Skipping:** Many fragments computed during the ray-traversal do not contribute to the final image and only consist of *transparent space*. The mapping from scalar volume data to a four-component color-opacity vector often results in sampled positions with an opacity of 0. As demonstrated in Figure 2.12 most of the fragments inside the bounding geometry are empty, while only a handful of fragments represent the extracted object. While sampling all these *empty* positions does not harm the quality of the image, it does cost a lot of computations and time.

  A widely used solution for this problem is empty-space skipping. To fastly compute, with only a handful of samples, whether a given fragment would result in a transparent color value or not, an additional space partitioning data structure is introduced to the ray-traversal step. These data structures, for example, an octree of the complete three-dimensional volume data, store a minimum and maximum value within each node. During ray-traversal, the renderer can sample the octree, determine the current position's minimum and maximum value, and look up whether any scalar value within this range would result in a non-transparent value (Figure 2.13).

  While the inclusion of early ray-termination can be solved by including a few lines of code checking the already accumulated opaqueness of the fragment, implementing empty-space skipping is much more complex. Empty space skipping requires the computation of the space partitioning data prior to the visualization, additional computations for each fragment determining entry and exit coordinates for each cell on each level of the data structure that is traversed and can further be altered by changes to the transfer function. Even though these steps have to be part of the visualization and processing pipeline, the performance improvements of the technique during real-time visualization make up for it.

- **Progressive Rendering:** There has always been a race between the size of volumetric data sets and the available memory on GPUs. While the available memory on GPUs is constantly growing, it could never keep up with the resolution of data to be visualized. Progressive rendering is another concept adopted in the desktop space to make visualization of large-scale data sets

feasible. It should not be seen as an acceleration method in raw computational speed but more as a technique to provide immediate feedback and interaction to the user while still working on the final image.

Another concept often named when talking about progressive rendering is so-called out-of-core visualization, which for simplicity, can be described as a visualization of data that does not fit in the GPUs memory at once. Progressive out-of-core renderers use various pre-computed levels of details (different detailed resolutions of the original dataset) to generate the final image. During interaction with the system, the renderer falls back to a lower resolution of the data set that still fits in the memory and can be visualized interactively. Once the interaction with the system comes to hold, the system gradually pages in higher resolutions of the data set for a more detailed image.

These acceleration methods are found in this research or are mentioned due to several reasons that make them not feasible for the underlying rendering goals.

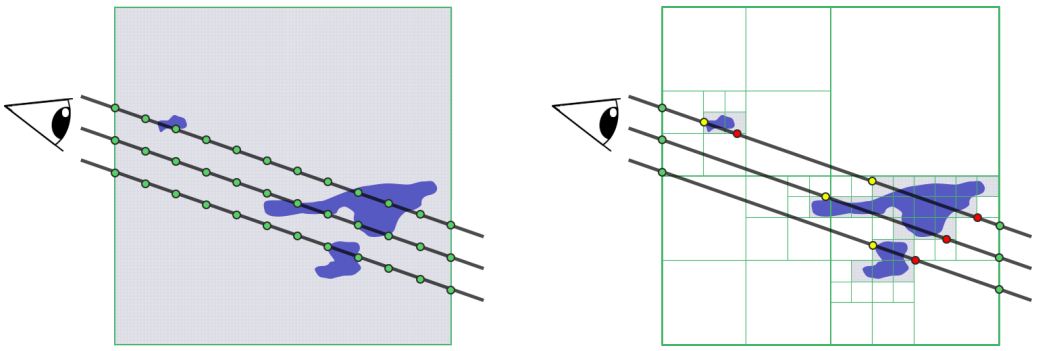


Fig. 2.13 Without acceleration methods such as Empty Space Skipping, during the ray-traversal, the volume has to be sampled up to thousands of times without the knowledge of whether the volume's traversed portion contains any data. (left) For the acceleration of the ray-traversal loop, additional space partitioning structures, such as octrees, can be utilized. Storing the minimum and maximum values of the cells data makes it possible to evaluate large chunks of the data with only a handful of samples, minimizing the total number of mandatory samples ray evaluation (right).

2.2.3 Transfer Functions

A crucial part of direct volume rendering is transferring the scalar volume data to properties to solve the rendering integral. A Transfer Function (TF) is one of the most commonly used methods. TFs map scalar values of the data set to a four-dimensional vector consisting of the color information C (red, green, and blue) and

the corresponding opacity value α. TFs are primarily implemented as lookup tables in the form of textures in GPU memory for fast access and editing of the function.

While there are higher dimensional solutions for TFs, this work mainly focuses on one-dimensional solutions. During runtime, the ray-caster samples and interpolates the scalar volumetric data. The sampled value is used as an index for the one-dimensional lookup texture. The systems presented in this work are tailored toward clinical practice. Within clinical practice, so-called windowing functions have established themself as reliable solutions to determine which kind of data should be visualized. These windowing functions are also known as the smooth-step function in other fields and are described by equation 2.7.

$$S(x) = \begin{cases} 0 & x \leq 0 \\ 3x^2 - 2x^3 & 0 \leq x \leq 1 \\ 1 & 1 \leq x \end{cases} \tag{2.7}$$

The smooth-step function, as found in literature, is defined by three parameters, the position on the function x, the left edge e_0, and the right edge e_1. For values of $x < e_0$ the function has to returns 0.0 and for values $x > e_1$ the function returns 1.0. In between e_0 and e_1, the function is interpolated by applying a Hermit polynom

In practical use, the smooth-step function is defined by only two parameters, the center c of the slope and the gradient f. Both parameters are truncated on a range of 0.0 and 1.0 and make it straightforward to visualize and understand the function, as demonstrated in Figure 2.14

Thanks to the simplicity of Transfer Functions, it is possible to simulate the results of IVR approaches. By setting every opacity value in the TF to 0.0, except for one value which has to be 1.0, an iso-value can be simulated. The resulting image comes close to the extracted iso-surfaces of IVR systems. However, since these results are not extracted polygon meshes but the result of ray-traversal, it is computationally much more expensive than a real IVR solution.

2.3 Deep Brain Stimulation

For a long time, Deep Brain Stimulation (DBS) has established itself as an excellent treatment option for different neurological disorders [189]. As the research of this work aims to provide intuitive and new visualization techniques for the daily clinical use of DBS, it is highly recommended to know some essential background about

Fig. 2.14 Smoothstep functions are a straightforward solution to define a transfer function for direct volume rendering systems and are often found in the medical field.

the state-of-the-art procedure. Advances in technological and surgical techniques helped DBS replace ablative procedures when applicable. Especially applying DBS to treating essential tremors and Parkinson's disease (PD) could improve tremor control by stimulating the Subthalamic Nucleus (STN) or internal segments of the Globus Pallidus (GPi). Over time, the number of diseases to which DBS has been applied expanded and ranges from chronic pain [293, 123, 123], over several forms of headache [172, 196, 108] to neuropsychiatric disorders [18, 132, 197]. However, the most prominent application field still surrounds PD and other tremor-related diseases. As DBS does not require a destructive lesion, meaning it does not harm any available brain tissue, it should be possible to reverse the intervention if needed by future therapies. Furthermore, the treatment parameters can be changed postoperatively to adjust them to the patient's needs, making DBS highly flexible regarding changes in the patient's symptoms. However, DBS is not a cure for these diseases but a treatment of the symptoms while the disease still progresses.

Over the upcoming sections, this dissertation will take a brief look at the currently implemented DBS procedure with its mandatory steps for patient treatment and outline the most crucial data domains utilized during these processes.

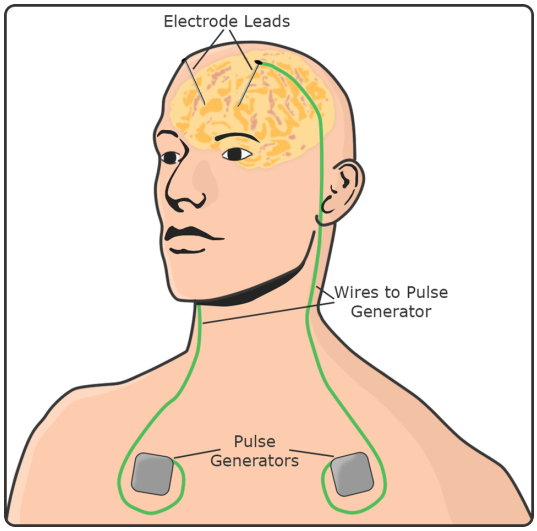


Fig. 2.15 During the DBS intervention, two electrode leads are implanted in the patient's brain at a tissue location suppressing various symptoms of the treated diseases. Two wires below the skin connect the leads to the pulse generators implanted in the patient's chest.

2.3.1 Procedure

A complete DBS procedure consists of three stages; the preoperative assessment and planning, the surgery necessary for implanting the electrodes into the patient's head, and the postoperative care mandatory to adjust the treatment parameters.

A highly critical aspect of DBS is the preoperative assessment of the patient. Patient selection is still a vital aspect of the treatment since side effects can still occur in intracranial hemorrhage and damage to other brain structures. However, the probability of neurologic deficits is seen as lower than with different approaches [302], making the benefit-to-risk ratio favorable in many cases. During the assessment stage, the goal is to identify patients that could benefit from the intervention the most considering the possible risks associated with DBS. Part of this selection process is to determine whether the movement disorders can be led back to one of the targeted brain tissues, whether the intervention could improve the symptoms, and estimate the risk for the patient. After assessing a potential candidate, they are informed about their benefit-to-risk analysis and alternative treatment options.

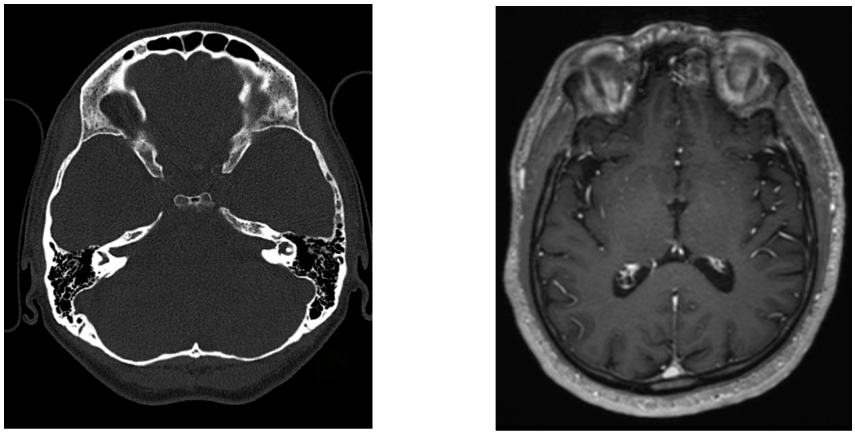
Part of this preoperative stage is also the planning of the intervention. DBS is a stereotactic surgery requiring stereotactic localization systems to identify the targeted tissue [304, 111]. Part of this localization process is medical imaging data in the form of MRI, and CT acquired preoperatively. More information about MRI and CT

can be found in Sections 2.3.2 and Section 2.3.2. Both data domains are mandatory for the planning phase as they display different tissue types. While the MRI data usually displays more information about the internal brain structure, it can not be used in combination with the stereotactic localization system made out of metal. On the other hand, the CT scanners have no problems with metallic objects, allowing surgeons to create scans of the patient's head while wearing this localization system. As both data domains need to be colocated with each other, surgeons have to fuse both data domains during the preoperative planning phase. This task can be tedious and time-consuming. A Part of this work presents a GPU accelerated approach for the fusion of both data domains giving surgeons more time to locate the target tissue (See Section 3.2.4). Once the surgeons have located the targeted tissue, they can create a treatment plan, including angles and entry markers of the patient's scalp. This process can be time-consuming overall and must be done immediately before surgery.

The second stage of DBS consists of the surgery itself. Utilizing the stereotactic localization system and the treatment plan, the entry points of the leads undergo local anesthesia, keeping the patient awake during the intervention. A small hole is made in the skull for the placement process. In many cases, the placement process is aided by MER in addition to the preoperatively created treatment plan. The electrodes of the MER are moved toward the targeted location in small incremental steps while recording the neural activity of the tissue in real-time. Surgeons can listen to these real-time recordings on the connected recording device and base their decisions on the sounds they hear and the preoperative treatment plan. Years of experience are mandatory to distinguish between different tissue types only on sound. During the research presented in Chapter 3, a visualization system was developed, providing various types of visualizations of the recorded data to aid surgeons in this process.

Furthermore, the recorded data were classified using a supervised learning approach, providing more certainty to surgeons in this critical stage. After each step, the patient is conducted about his well-being. Once the electrodes are getting close to the possible target tissue, tiny electric impulses are emitted to get the first assessment of possible symptom improvements. The patient has to fulfill several tests during this stage to identify the best possible location for the lead. After identifying the final location of the electrode, extensions are added, and a pulse generator is surgically implanted in the patient's chest.

During the last stage of DBS, the postoperative care, the settings of the implanted pulse generator are fine-tuned. Over several sessions and repeated over the years,



Computed Tomography Magnetic Resonance Imaging

Fig. 2.16 Two example images of CT and MRI. While the CTimage provides more information about the bone structures of the patient, no details about the brain tissue are visible. On the right side, the MRIdata presents much more details about the structure of the brain but less information about the dense structures of the body.

patients undergo tests to adjust the individual strength of the emitted impulses. This ongoing fine and re-tuning are mandatory as the nature of the central nervous system is dynamic, and the disease is still progressing. While at the beginning of DBS, these adjustment sessions were primarily done within the hospital due to the hours it took to find the correct parameters, thanks to treatment advancements, many of the adjustments can now be made remotely within miniutes, not requiring the patient to be physically within the hospital.

2.3.2 Medical Data of DBS

As the primary goal of this dissertation is to close a gap between data domains of neurostimulation procedures and the visualization available to surgeons in clinical practices, it is crucial to introduce a brief background of these data domains. The primary data types in the covered research topics originate from the medical domain. Each data domain plays a crucial role in DBS and is found in various procedure steps. While at first glance, it may seem redundant to some people to include different data domains, such as CT and MRI, each has personal benefits that the other can not provide. Where CT has its benefits in providing fast and accurate images in a larger resolution compared to MRI techniques, it still has the drawback of exposing patients to harmful X-ray radiation.

On the other hand, MRI does not use harmful X-rays during image acquisition but does take longer to compute and provides a lower resolution. Due to the nature

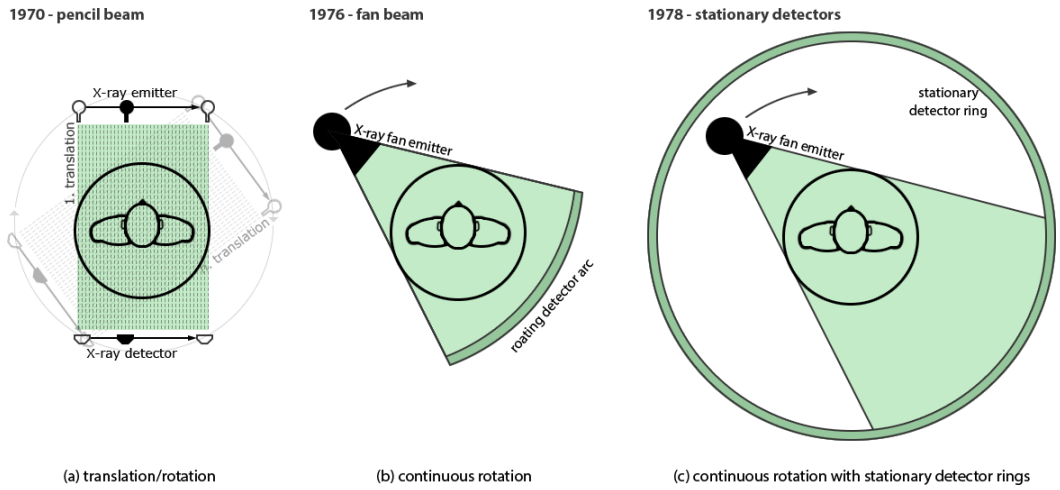


Fig. 2.17 Generations of CT scanning approaches based on the publication of Kalender [145]. Pen beams were used in the early version of CT scanners (a) using single emitters translated across one axis before the complete device gets rotated for sequential scans. The approximated scanning duration was above 24h. Later, fan beams were used with rotating detectors arcs (b), accelerating the scanning procedure to several seconds. Two years later, the stationary detectors were introduced (c) to push the scanning times of singles slides below 5ms.

of both imaging techniques, they provide inside into different tissues. Where MRI shines in displaying *softer* tissue, it losses detail in more *dense* material, which on the other hand, is clearer to see in CT scans. Furthermore, MRI does not allow any magnetic objects to be present during the scan, which makes it impossible for DBS to acquire volumetric imaging data, including the localization system. Also, post-surgery patients cannot enter a MRI scanner due to the implanted electrodes and pulse generators.

As a result, both imaging techniques are necessary for the medical field and are often not used only by themself but in addition to each other. CT and MRI data will appear in several topics of this dissertation. Therefore, the following Sections will present a brief historical overview and functionality of these techniques with information about their use-case within DBS. In addition to these two imaging domains, MER will be introduced as it plays a significant role in DBS and the research found in Chapter 3.

Computed Tomography

Computed Tomography was the first modern imaging technique in the medical field and was first introduced in 1972 by Sir Godfrey Hounsfield [135]. While, at its

time, the reconstruction of measured X-ray data was a novelty, it is an essential part of today's clinical practice. Around the mid-80s, field experts assessed CT as a dead technology [145] due to the upcoming and less harmful MRI technology. However, CT could present itself as a vital part of practical clinical use. With the shift from slice-by-slice scanning to spiral scanning technologies at the beginning of the 90s [145], CT was finally capable of creating complex three-dimensional scans of the patient quickly. The increasing amount of data also finally led to the need for adequate visualization for volumetric data sets, such as DVR.

From a simplified technological standpoint, modern CT scanners use X-ray emitters rotating rapidly around the patient while collecting the remaining intensity on the other end of the scanner. The collected data is used to generate cross-sectional images of the patient based on the measured intensity on the detector side. For image generation, the measured data is reprojected onto a two-dimensional image plane considering the angle of the emitted X-rays and the received intensity [145]. The reconstruction of the image can be solved using one of several available reconstruction methods [211]. These algorithms are often classified by one of the following types: iterative techniques [232], back-projection [325], and Fourier reconstruction [273].

While modern CT scanners are fast at constructing three-dimensional data sets, the first available hardware could not even consider creating three-dimensional data. In the beginning, CT scanners were limited by the available technology. First-generation CT Systems used X-ray emitters located on a sliding gantry that only utilized a limited amount of parallel pen emitters and detectors [145]. These emitters and detectors were translated from the start to the end of the gantry and rotated afterward for repeated measurements of various angles (Figure 2.17 a). It was reported that images of a single slice took up to 300s, in which the patient had to be located still within the tube. Over time more advanced technologies were introduced, such as fan beams allowing the use of a higher number of detectors on the receiving end (Figure 2.17 b). The revolution to CT scanning technology came from fourth-generation scanners around 1978. Rotating fan beams could finally be combined with an encircling stationary set of detectors, accelerating the overall scanning procedure by magnitudes (Figure 2.17 c).

The next giant leap in CT scanning technology came in the 1990s when spiral CT scanning was introduced [143, 144]. Up to this point, CT was only used to create single-slice images of the patient, as no motion was allowed during the scanning procedure. With the introduction of a continuous scanning of the patient along the z-axis, or longitudinal axis, it was finally possible to create full volumetrical images

of the patient within a short time [144]. Since then, technological advancements have mainly focused on further accelerating the scanning speed and increasing the image quality [106, 115, 3]. Nowadays, CT scanners can produce complete volumetric patient data sets within minutes and in a resolution that was unthinkable at the beginning times of CT.

As mentioned, CT is based on the emission and detection of X-rays. These X-rays pass the subject and get absorbed by the penetrated media, where the media's density significantly impacts the remaining intensity of the emitted ray. While the reconstructed images represent a complete slice, or even volume, of the patient, the receiving detectors only acquire information about the remaining radiation of a single ray that passed the subject. The beforementioned reconstruction techniques are crucial to process the data and achieving images that the doctors can interpret. While explaining these reconstruction techniques would go beyond the scope of this work, it is fascinating and worth a read.

It was already mentioned that CT plays a crucial role in neurostimulation procedures, such as DBS. Especially during the location process of DBS and the post-operative validation of the lead position, CT is indispensable. Using CT imaging data, including the patient wearing the localization system, surgeons can precisely identify positions within the reference frame. However, as outlined, CT is more prominent for making dense tissues visible within the imaging data. These dense structures include the localization system or bones but not the brain's soft tissue. While contrast media could be used to highlight some parts of the brain, other imaging techniques are more helpful for this task, such as MRI.

Magnetic Resonance Imaging

Magnetic Resonance Imaging (MRI) is an imaging technology in the clinical field utilizing magnetic properties of the human body to generate images of any body part [29]. The first research that later led to MR imaging can be traced back to the mid-1950s. During this time, a physicochemical phenomenon was discovered, later known as Nuclear Magnetic Resonance (NMR) [121], which built the fundamentals for MRI. Over decades more and more researchers worked on this phenomenon for various use cases [242]. The research finally resulted in the first full-body MRI scanners in the early-1970s.

While the overall functionality of an MRI scanner is complex and fills entire books, a brief overview of the functionality is more than sufficient for this work to

understand what type of data can be found within MR images and what not. For the interested reader, I would suggest the book by Weishaupt et al. [317] to get more information on MRI scanning.

While placed in an MRI scanner, the human body is surrounded by a strong magnetic field. This magnetic field influences the hydrogen protons in the human body, aligning them with the orientation of the scanner's axis resulting in a magnetic vector. While the magnetic field influences these hydrogen protons, radio waves are added to the system, deflecting the aligned protons in other directions. Once the radio waves are removed by turning the emitter off, the deflected protons will return to their resting state, defined by the scanner's magnetic field. While returning to their resting state, the protons emit a radio wave which can be recorded by the MR scanner receiver coils surrounding the body. As with CT, the recorded data can be used to reconstruct a cross-sectional image. Various frequencies of radio waves can be used to record different tissue types as each tissue in the human body resonates on a different frequency and provides different relaxation times.

An MRI scan results in two images. Once the added radio wave is turned off, two times are measured. The first time measured is the time it takes for the hydrogen protons to realign with the scanner's magnetic field, also known as *T1*. The second time measured is the time it takes for the axiel spin of the protons to return to their resting state *T2*. As both results differ in their measurements, some tissues are better to identify in one or the other image. Both images are often consulted in medicine to minimize doctors' uncertainty. Multiple scans are executed with different pulses for various types of tissues to generate a fully volumetric MRI scan.

For a short time, MRI was seen as a death sentence to CT imaging as it could provide more information about different tissue types while not exposing patients to harmful X-rays. However, as both technologies result in different images, focusing on different tissues, both have established themself in daily clinical practice. Even though MRI does not use X-rays, it is limited in its approach due to not being compatible with magnetic objects within the scanner, as this can be highly dangerous for the patient. By not being compatible with magnetic objects, the applicability of MRI is limited in various fields, such as the emergency department, when doctors can not be sure if patients have any magnetic implants or not. Also, in DBS, MRI can not be applied after the surgery is completed. The implanted leads and pulse emitters should not be put into an MRI scanner. Therefore surgeons have to fall back to CT scans to verify the placement of the electrodes. However, for the planning of the surgery, MR imaging is indispensable. MRI provides better imaging data

for tissues with high levels of hydrogen protons, as found in the human brain. In contrast, CT imaging treats brain tissue more or less the same, making it impossible to distinguish between different tissue types.

Micro-Electrode Recording

Besides the two introduced imaging techniques, which play a crucial role, particularly during the planning phase of DBS, another technology has to be introduced for the presented research of this work. This technology is MER, a technology, in contrast to the imaging techniques, that is used intraoperatively to minimize the risk of misplaced electrode leads. A large portion of the research presented in Chapter 3 focuses on MER, the processing of the data, the real time classification of the data, and its visualization. Therefore, it should be seen as mandatory to have a brief overview of MER and its used cases. A large portion of the presented information is adapted from the book of Pouratian et al. [230], which is a collection of in-depth articles to the field of stereotactic neurosurgery.

Most easily, MER can be described as a technique for the intraoperative recording of neural activity during the placement process of electrode leads for the stimulation of deep brain regions mandatory for DBS. The recorded neural activity differs depending on the region of the brain and is distinguishable for trained experts during the surgery. As the target location is planned preoperatively using the described imaging data of MRI and CT, MER can aid surgeons during the placement process as a form of validation layer for the absent imaging information during the actual insertion of the leads.

As described in Section 2.3 DBS works by implanting macro electrodes within the malfunctioning brain tissue and stimulating the area to improve the symptoms of various movement disorders. While the final goal is to implant a single electrode into the correct tissue, validating the location utilizing MER is first mandatory. After preparing the patient, anesthetizing the scalp, and drilling the hole, guiding tubes are inserted into the opening. Utilizing these guiding tubes, up to five recording microelectrodes are inserted into the patient's brain on each hemisphere. Shortly after exiting the guiding tube, the first recordings can be seen on the connected MER recording hardware. After validating that the recording electrodes are functioning correctly, they are moved in submillimeter steps toward the planned target location ranging between 0.1 and 0.5 mm. During this process, neural activity is recorded in the form of spontaneous reactions and the completion of tasks given to the patient.

Once the target area is reached by the electrode and sufficient recording information is obtained, including first test stimulations, the recording electrodes are removed and replaced with the final macro electrode.

2.4 Virtual Reality

Over the last decade, Virtual Reality (VR) has found its way into the broad consumer market. While VR is not a new technology, due to the recent development of affordable end-user hardware revitalized the interest in the technology, and many application fields have aimed to incorporate VR since then. The term Virtual Reality was first used by Jaron Lanier in 1987 [30]. VR is often described as a technology for generating multi-sensory simulations of a digital environment [255]. Within VR, users are surrounded by an immersive and interactive environment through visual, auditory, tactile, and other sensory perception [156]. In Computer Science, terms such as virtual reality, virtual environment, or immersive environments are often used interchangeably. Many definitions of VR [260, 74, 20, 170, 270] often emphasize the impression of users being in a realistic-looking world with possibilities of interaction methods representing real-world interactions. According to Sherman et al. [260], there are four critical elements for a VR experience: a virtual world, immersion, sensory feedback, and interactivity.

While VR has found its comeback in the consumer market, it was long present in various application fields for decades. The first VR applications were mainly designed as simulators for pilots or military scenarious [30]. The shift to home users with new technologies also shifted the applications. Today's VR not only imitates the real world but also allows users to explore entirely virtual worlds independent of the rules and bounds of reality. Smith et al. [270] divides applications in virtual environments into two categories: realistic virtual worlds and magical virtual worlds. Within their publication, they defined realistic virtual worlds as simulations of the real world for training purposes. Magical virtual worlds are conceptualized not to follow the rules of reality, meaning they have no implications outside the virtual environment. Examples of magical virtual worlds besides video games are scientific applications such as data or scientific visualization.

According to Slater et al. [266], an ideal immersive VR system has to serve multiple modalities, such as visual, auditory, or tactile, and support input information from tracking hardware. Instead of displaying a virtual environment to the user without any user information, the computer has to consider the correct viewport of

the user based on the position and orientation of the user's head. The system must react to every positional change of the user and present immediate feedback. Slater states that ideally, the tracking should include tactile and force feedback, heat, and smell.

According to Mandal et al. [190], the main difference between traditional displays on desktop setups and VR is the stereoscopic projection and the peculiar features of immersion, presence, and interactivity. VR allows users to experience a detailed virtual environment, providing a more extensive and immersive experience than a traditional desktop display. The term immersive is often interpreted ambiguously as different depths of immersion exist. As stated by Fons et al. [107] VR can be differentiated by the level of immersion. These are fully-immersive VR, semi-immersive VR, and non-immersive VR.

Non-immersive systems can be easily employed without special VR hardware and allow users to explore the presented content based on the device's capabilities. For example, phones can be used for non-immersive virtual reality; users can tilt the phone while exploring a 360-degree environment [299]. Semi-immersive VR systems are, for example, Fish Tank VR systems. Stereo images of a three-dimensional digital scene are displayed on a two-dimensional display. The perspective of the projection is determined by additional hardware mounted to the user's head [86]. Modern fully-immersive VR systems typically have a set of displays directly mounted to the user's head and support full tracking of the user's position in real-time. Due to the advancements and cost of these Head-Mounted Display (HMD) systems, other fully immersive systems, such as Cave Automatic Virtual Environments (CAVEs) [77], have lost popularity.

2.4.1 VR Displays

Over the decades, multiple different types of VR systems have been introduced. While many of them are classified as immersive systems, they often differ in their degree of immersion. Virtual environments can be displayed on HMDs (see Figure 2.18), one or many monitors, or one or many projectors. To be classified as an immersive system, these systems have to support head and hand tracking to enable the user to explore, interact with, and observe the virtual environment.

Sutherland et al. [281] introduced the first head-mounted display design in the '1960s', and ever since then, the opportunities and capabilities of immersive VR have progressed. From the Sensorama machine, a cost-intensive multi-sensory

simulator invented in 1962, to the first CAVE [77] in 1992 and then the rise of VR gaming technology for the masses with the start of Oculus [217] and HTC Vive [137]. Fully immersive VR has never stopped evolving, with more and more companies providing HMD systems. The unresolved problem since the early 2010s has been that traditional and now outdated techniques such as CAVEs require complex, cost-intensive, and unhandy hardware [84] while providing the advantage of offering a wider field of view in comparison to an HMD system.

CAVEs, also known as Cave Automatic Virtual Environments (see Figure 2.19), are VR environments in which the user is surrounded by up to six walls in a few square meters in which projectors and tracking devices are installed. Due to their design, CAVEs are considered immersive or hybrid-immersive environements [103]. Compared to closed HMD systems, CAVEs offer an extended view in the peripheral field and offer users the possibility to see their own body, typically not perceivable while wearing an HMD.

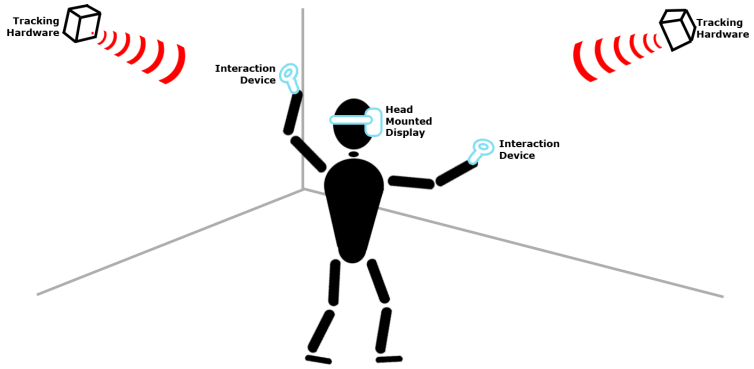


Fig. 2.18 A typical VR system consists of a head-mounted display, trackable interaction devices, and a tracking system. The most common systems use stationary tracking hardware to locate the HMD and interaction devices. Other systems use so-called inside-out tracking moving the cameras to the HMD for tracking.

Mestre et al. [202] compared HMD and CAVE systems in terms of their level of immersion. While HMDs seem to offer a better and more immersive experience, especially compared to classic four-wall CAVEs, advanced CAVE systems, such as the CAVE2 from Febretti et al. [103], six-walled or even tunnel-like CAVEs can compete in this regard. Especially the aspect of still visually seeing the own body is pointed out as a big positive to CAVE systems. The main difference between CAVE and HMDs systems remains the cost difference. Thanks to mobile devices, for instance, GearVR or the Oculus Quest, which do not require high-end computers, VRs using an HMDs can be affordable for nearly everyone [12, 186].

Modern VRs systems commonly allow users to explore spatial surroundings freely, only restricting movement based on the physical boundaries of the real world. However, this freedom in motion introduces constraints for these systems in the form of necessary high update rates and low response times. Not achieving these mandatory constraints result in the possible appearance of cybersickness symptoms 2.5. While the actual point at which high response times or low update rates of the display influence the performance and health of the user varies from person to person, once falling below this threshold, cybersickness symptoms appear rapidly.

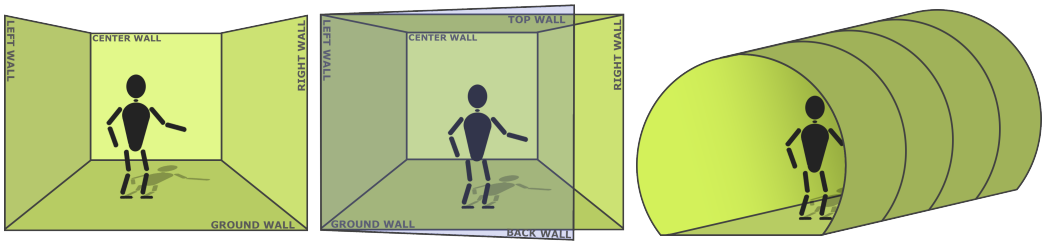


Fig. 2.19 CAVEs exist in various forms. They are often used as a four-wall setup (left) covering the center, left, right, and bottom of the room. Alternatively, six-walled CAVES (center) are built like a cube, surrounding the user, and one wall functions as an entrance door. Another approach often found as a CAVE setup are tunnels (right) spanning the setup in a spherical form around the user.

2.4.2 Interaction and Tracking

Interaction and real-time user tracking are two significant aspects of any immersive VR system. As interaction possibilities within VR differ significantly from traditional desktop systems utilizing mice and keyboards as input devices, alternative approaches for interaction with an immersive virtual environment have to be considered. Many seated VR systems rely on controllers primarily used for video games for interaction with the environment. However, modern VR systems shifted toward supporting full-body tracking hardware in the form of trackable devices such as VR controllers spatial recognized by the VR system and other trackable gadgets mountable to the body or third-party hardware.

VR systems such as the HTC Vive or Oculus Rift mostly rely on controllers as interaction devices. However, the design has shifted from a traditional layout of a single physical device held with both hands to separate controllers for each hand. These controllers try to track single-finger interactions, simulating grips and pointing gestures, and can track the spatial position of the user's hands. Besides supporting

input possibilities such as joysticks and buttons found on controllers of most home consoles, direct mapping of traditional movement interactions to a VR environment seem unintuitive for many users. Hence, other interaction techniques and metaphors have established themself as the gold standard for VR user interaction.

The most promising interaction techniques in many VR applications are based on imported behavioral schemes of the real world, which describe the adaption of everyday interactions (e.g., pulling a doorknob) and transferring these schemes onto the virtual environment. According to Fuchs et al. [110], this interaction technique is the most intuitive and fastest to learn. The behavioral interface is a mixed entity comprising both a physical mechanism and an acquired schema that is transferred and adapted into a virtual environment. While behavioral interfaces are the most promising techniques concerning the user experience and learning rate, these interfaces are often difficult to employ since many physical mechanisms cannot be transfered directly into the VR environment. Fons et al. [107] explored how interaction with volumetric models in VR can become more intuitive and found that user experience benefits from so-called interaction metaphors. The design of modern VR controllers can directly support these interaction metaphors since they are designed for the recognition of single-finger movements, hence allowing users to simulate grab or point gestures. Most VR controllers support a higher Degree of Freedomt (DoF) thanks to the additional tracking capabilities than traditional systems. The DoF of VR controllers represent the ability of a controller to move in space while being spatially tracked. Most VR controllers support up to six DoF (Figure 2.20).

Besides physical VR controllers, gesture tracking has found its way to commodity-available VR systems. Initially, gesture tracking was achieved using the Leap Motion controller mounted in front of the HMD [66, 14]. New systems such as the Oculus Quest support inside-out and gesture tracking out of the box without any additional hardware added to the system. These systems use two cameras and LEDs to reconstruct and track virtual models of sensed objects. The user's hands can be digitally reconstructed based on the recorded data for gesture interaction. However, a downside of this approach is the camera's field of view. For gesture tracking to work, the hands must be visible to the system. In the case of the Leap Motion, the field of view comprises 135-degree and more or less one meter in depth. Once the hands leave the field of view recorded by the systems, no knowledge is present about the current interaction state of the user.

Motivated by the exploration of hand gestures and controllers, LaViola et al. [164] aimed to make use of the broader range of virtual locomotion techniques and explored rather unusual techniques for its time, such as walking, leaning, bending, and turning. Locomotion describes forms and methods in which users move within the virtual environment. They state that widget-based controls, for instance, an Oculus™ Controller, are still the most popular option but could overwhelm users in complex applications. They developed a step-world-in-miniature (WIM) navigation. The user employs gestures to access a WIM to navigate large distances and familiar body-motion navigation for shorter distances. Over the decades and with newer tracking hardware, more and more research was established in different locomotion techniques. Since locomotion plays a crucial role in Cybersickness, various techniques have come up to counteract the chance for symptoms to appear [69].

To summarize, virtual reality systems' most popular interaction techniques are gesture controls or trackable controllers such as the one from Oculus™ with six DoF. Researchers have found that intuitive interactions can be created by adapting mental schemes close to real-world interactions. Furthermore, controller-based interaction may lead to better results than gesture tracking when the task becomes more abstract.

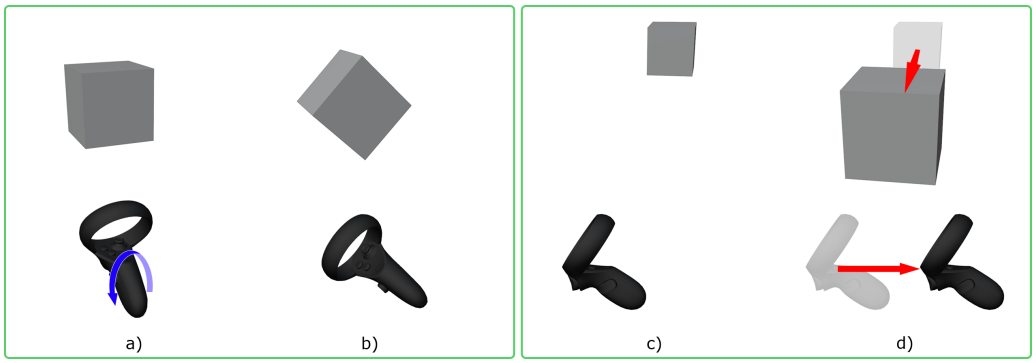


Fig. 2.20 VR Controllers support an extensive range of DoF. For example, controllers can be rotated while mapping the rotation angle to the shape to spectate it from various angles (a-b). Other examples would be translation interaction (c-d), moving the controller toward the user can translate the object for a more detailed view.

2.4.3 Immersion, Presence and Engagement

Immersion, *presence*, and *engagement* are three terms that frequently come up when talking about VR. In VR gaming especially, these terms are omnipresent and define the quality and reception of the VR game. The question arises whether and how

immersion, *presence*, and *engagement* are related to the efficiency and effectiveness of a user-centered scientific visualization system.

Immersion is the most relevant characteristic of VR, mainly fully immersive VR, having various definitions. Most definitions are united by the statement that *immersion* describes the extent to which a virtual reality system provides physical inputs to various human sensory modalities that result in an illusion of reality [190, 268, 155]. In simpler terms, *immersion* describes how well a system can provide the illusion of a virtual world without considering the user's subjective perception. Kim et al. [155] extend the term and interpret *immersion* as the extent of the user's flow state that arises from the user's *engagement* with the virtual environment. From a different point of view, Singer et al. [264] stress the dependence on individual perception and reaction to the virtual world. Rosenbaum et al. [247] emphasize that due to the use of stereoscopic vision and motion parallax, immersive environments may imitate our natural three-dimensional vision and subsequently may lead to an advanced perception of physical *immersion*.

In 1997, Slater et al. [269] defined the concept of *immersion* as the sum of the characteristics of a virtual environment. Based on several parameters, virtual reality systems could be compared in terms of *immersion*. In 2009, Slater et al. [266] summarized these parameters to measure the quality of the experience, which include the frame rate, the overall extent of tracking, tracking latency, the quality of the images, the field of view, the visual quality of the rendered scene, the dynamics, and the range of sensory modalities accommodated.

Presence within the context of VR is defined as one's sense of being in the virtual world. In the literature, researchers have agreed that *presence* refers to the user's subjective feeling of being in the virtual world instead of their actual physical location. Various definitions can be found in research. One of them is the feeling of being in an environment, even when the user is physically in another [264, 321]. Alternatively, *presence* is defined as the state of consciousness of being situated in the virtual environment [268], and lastly, as the user's subjective, context-dependent psychological response to the virtual world [40]. The feeling of *presence* is considered as perceptual and not cognitive because human perception identifies the actions and objects and inducts automatic reactions to the changes in the environment [31, 267]. Dwyer et al. [98] have defined three types of *presence*:

- **spatial presence**, which refers to the definitions above
- **social presence**, which becomes relevant in collaborative data visualization

- **self-presence**, which means the psychological state in which virtual selves are experienced as the actual self and are perceived as a reconstructed version of oneself [171]

Even VR experts use the term *presence* interchangeably with *immersion* incorrectly. Whereas *immersion* means the objective level of sensory fidelity provided by the virtual reality system, *presence* relates to the human reaction to its subjective perception of the level of given *immersion*. Still, they are logically inseparable and empirically strongly associated [265]. Singer et al. [264] agree that there is no obvious or straightforward relationship between *presence* and *immersion*. However, an exciting field of research could be to determine if *immersion* influences *presence* and task performance. Cooper et al. [72] found a significant negative relationship between the objective and subjective *presence* measured. Also, the performance and the feeling of *immersion* improved when multimodal sensory feedback was provided. In data visualization, this finding is limited as Cooper et al. also found that auditory and tactile cues influenced users' perceived *presence*, which has yet to be applied in VR.

Lastly, *engagement* is a term that repeatedly occurs when discussing VR visualization's benefits. *Engagement* in VR environments means the user's state of mind during a focused experience and is often linked with a loss of sense of time. When a user's *engagement* is high, awareness of the virtual situation is amplified, and other aspects of the real environment are neglected [78]. Dorph et al. [93] defined *engagement* as the users' focus, participation, and persistence within a task and stated that it subsequently is linked to adaptive or self-regulated learning. They divided *engagement* into three subcategories:

- **behavioral engagement**, which focuses on the physical focus on the task
- **cognitive engagement**, which refers to thought and attention
- **affective engagement**, which includes the emotional experience.

Engagement is related to user performance, especially in educational or academic applications [11].

In conclusion, *immersion*, *presence*, and *engagement* have proven to be robust characteristics of VR. All three attributes of VR are linked to user satisfaction, learning, and performance enhancement. It remains to be determined how scientific

Cybersickness		Symptoms	
Eye strain	*Sweating*	*Disorientation*	*Nausea*
Headache	*Dryness of mouth*	*Vertigo*	*Vomitting*
Pallor	*Fullness of stomach*	*Ataxia*	

Table 2.1 Possible Cybersickness symptoms described by LaViola. [165].

visualization can benefit from *immersion*, *presence*, and *engagement*. Nonetheless, desktop displays cannot achieve the same level of *immersion*, *presence*, and *engagement*, as shown in the study of Lu et al. [187].

2.5 Cybersickness

A common problem of virtual reality are occurring Cybersickness (CS) symptoms due to prolonged exposure to the VR system. As research has identified, these symptoms can be up to three times stronger than symptoms of simulator sickness [272] and appear polysymptomatic, which describes the presence of multiple symptoms at the same time [236]. A study by Cobb et al. [258] indicates that roughly 80 % of users experience cybersickness symptoms within ten minutes of VR usage.

The symptoms covered by Cybersickness are manifold. LaViola et al. [165] listed several possible symptoms covered in Table 2.1. Cybersickness symptoms vary from person to person. While the VR system and software can be a limiting factor, supporting the appearance of Cybersickness symptoms, for example, an increased latency between the displayed image and the reaction of the system to user inputs [274] or a low refresh rate of the displayed image due to system overload, several aspects besides hardware limitations can influence Cybersickness. Studies have found that users' age plays an important role in the appearance of Cybersickness [82], as other studies also indicated that women are more prone to Cybersickness symptoms than men [33].

While up to this point, there is no single concrete reason for why cybersickness symptoms occur, four theories have established themself in the research community.

Sensory Conflict Theory

The sensory conflict theory was introduced by Reason and Brand in 1975 [235], making it one of the oldest and most accecpted [206] explanations for Cybersickness. The concept behind this theory is the conflict between the visual and vestibular

systems during the usage of VR hardware. While experiencing a virtual environment, the visual feedback provided to the user does not match the sensory input of the user. For example, users can perceive a movement in the virtual world while exploring a digital landscape displayed on the screen of the HMD while still sitting in the real world. The visual feedback of movement does not correspond to the vestibular system's state of still being seated in place, resulting in a mismatch that could result in Cybersickness symptoms. While widely accepted in the community, criticism came up [279] since the theory does not explain how and why users can get used to the sensory mismatch over time, nor does it provide any prediction on the type of symptoms that may occur.

Postural Instability Theory

Another theory that aims to explain the appearance of Cybersickness was introduced by Riccio et al. [241] in 1991. They described the postural instability theory, which relies on the human desire to maintain an upright posture. In this theory, two terms are essential to reduce the chance of Cybersickness. The first one is posture control, the human ability to maintain a stable posture at any given time. The second one is posture stability, defined as the difference in postural control over time. The less postural control a human can maintain over time, the less their postural stability for this given time frame.

Based on this theory, an absence of posture stability for a prolonged time can introduce a rapid growth in cybersickness symptoms. As users adjust their posture based on the virtual environment perceived during HMD applications, their posture control gets influenced from the outside, reducing their posture control.

Poison Theory

In 1977 Treisman [289] introduced the poison theory. Based on this theory, Cybersickness can be seen as an evolutionary survival mechanism. The displayed images provided by the VR system are a form of sensory hallucination perceived as a poison by the body. As a result, the body introduces Cybersickness symptoms, such as nausea, to eliminate the consumed poison.

The criticism of this theory is its vague formulation considering how and if nausea appears [165]. Furthermore, does this theory not explain the large spectrum of symptoms covered by Cybersickness, nor is this theory proven to any extent [82].

Rest Frame Construct

One of the newest theories regarding Cybersickness is the Rest Frame Construct introduced by Prothero [231] in 1998. At its core, this theory describes that humans tend to strongly perceive stationary objects as a reference frame in their surroundings. If the virtual environment does not provide any fix points as a reference for orientation, Cybersickness symptoms are more likely to appear. Lin et al. [180] tested this approach by introducing stationary clouds as fixed points within a virtual environment.

2.5.1 Reduction of Cybersickness

Different techniques can be applied to VR systems and applications to reduce the appearance of Cybersickness symptoms. These techniques can be categorized into visual methods, design options, and additional hardware. As Cybersickness symptoms are mainly occurring due to a mismatch of a user's visual and physical experience, each technique aims to minimize the sensory mismatch.

Visual Methods

To provide the best possible immersive visual environment, offering the largest field of view (FoV) achievable by the hardware is desirable. While an increased field of view can enhance the perceived presence of a user, it also increases the perceived vection [22], which again increases the sensory conflict. One of the most effective solutions is the field of view reduction (FOVR) [118]. By narrowing the field of view while experiencing and VR environment utilizing an HMD, the vection in the peripheral vision gets reduced, resulting in lower chances for Cybersickness. While simply narrowing the field of view by fading the peripheral vision to black is often not the most desirable option for spatial orientation, other approaches tested an increased blur in the peripheral field of view [118, 182].

Further research in the field of image reduction to lower Cybersickness symptoms investigated asymmetric field of vision [324, 5]. They increased the field of view on the dominant eye while using a fade-to-black approach for the non-dominant one. Additionally, more of the bottom area of the image is displayed to the user as these tend to orient themself based on the ground.

Other visual approaches besides the field-of-view focus on blurring mechanisms for horizontal rotations [50], saliency blurring [213] approaches utilizing deep learn-

ing algorithms to filter important objects, or reference frames helping users to locate themself in the scene [318].

Design Options

Several aspects of VR software can be considered regarding the possible occurrence of Cybersickness symptoms during the design process of the application. One of the most significant aspects is the form of locomotion supported by the system. As most systems support a room-scale VR experience, the question often arises about how environments that extend the physical space are realized. Several concepts exist to solve this problem. One of the most prominent ones is the concept of teleportation [41], allowing users to move to another location within the visual field instantly without physically moving. While this concept is often used to increase the possible reachable area, instant transportation to a new place can lead to brief disorientation and sensory conflict. Another approach often used is the redirected walking [277, 234]. This concept alters the walking direction between the physical and virtual worlds to increase the reachable area without reorientation seamlessly. Various approaches to this concept have been researched, evaluating adjustments in walking angles or walking speed. Such concepts need to consider the presented speed and direction. Especially a sensory mismatch between the perceived displayed movement speed of the virtual environment, and the real movement speed can again result in Cybersickness symptoms. Other design concepts that can influence the degree of Cybersickness are often tailored toward the design of the virtual environment. Ang et al. [15] investigated the influence of uneven terrain on Cybersickness symptoms. During their study, users could explore three different worlds freely, each with varying levels of unevenness. The result of their research indicated that flatter levels resulted in a better score in regard to Cybersickness. Another concept that has been researched is the degree of realism. Several groups [83, 229, 287] have studied how low realism can benefit Cybersickness. The results of these groups indicated that an increased realism comes with an increased risk for Cybersickness to appear.

Hardware Solutions

Regarding the reduction of Cybersickness symptoms, various hardware approaches can be seen as applicable. The most prominent solution that can help avoid Cybersickness is advancements in HMD technology. As mentioned, the mismatch between

the visual and vestibular systems plays a significant role in Cybersickness. Much of this mismatch comes from latency between the VR system and the user. Over recent years, improvements in tracking hardware and GPU performance helped to lower this problem. Advancements in display technology with increasing resolutions and refresh rates of HMD displays increased the immersion of the VR system and helped reduce Cybersickness Symptoms due to the improved image quality.

Besides the VR systems, other hardware, such as omnidirectional treadmills, entered the market over time. However, Wehden et al. [316] investigated how far treadmills can influence Cybersickness symptoms and concluded that omnidirectional treadmills achieved no significant benefit compared to methods such as redirected walking.

Chapter 3

Data Analysis and Visualization for Neurostimulation Procedures

3.1 Introduction and Problem Statement

Neurostimulation therapy [117] has established itself as a suitable treatment for various movement disorders. In particular, Deep Brain Stimulation (DBS) [227] has shown to be effective in treating the symptoms of a variety of affective and movement disorders like Parkinson's disease [252], essential tremor [116], dystonia [300], or Tourette syndrome [225]. DBS is used when traditional medications fail to control the symptoms. Compared to other neurosurgical procedures, such as thalamotomy [253], DBS does not harm nerve cells and instead blocks malfunctioning signals within the target area.

To treat the beforementioned movement disorders, DBS aims to implant electrodes within the target tissue in the patient's brain, controlled by a programmable pulse generator implanted into the patient's chest. The implanted electrodes emit tiny electrical impulses stimulating the otherwise malfunctioning brain tissue. A typical DBS procedure consists of three phases: preoperative assessment and planning, insertion of the electrode, and postoperative care.

The planning phase is a mandatory step of DBS to locate the target tissue for the procedure, in the case of Parkinson's disease, the STN [276], and to determine the electrode's exact placement and trajectory. To locate the target, doctors use preoperatively taken CT [53] and MRI [327] scans and translate the planned location into actual angles and penetration depth for the surgery harming as little healthy

brain tissue as possible. During the surgery, the computed angles are used to set up the stereotactic frame [111] for precise placement of the electrodes.

During a typical DBS surgery, none of the preoperative imaging data is present, and surgeons must rely on the precomputed planned setup. Micro-Electrode Recording (MER) [44] is used as an aiding tool to verify the current electrode position. MER is a real-time system allowing surgeons to gain feedback on the current brain activity of the awake patient. The recorded data is often represented by its time-dependent waveform and can be listened to as auditive feedback on the currently penetrated tissue. During the procedure, surgeons must fuse the auditive feedback with the preoperative imaging data to gain a complete overview of the current state of the surgery. This mental fusion of both data domains is a tedious task requiring surgeons to be highly experienced with both domains and have years of knowledge. While the fusion of both domains requires much training and experience, procedures using MER significantly improve the placement accuracy over procedures that are only based on imaging data alone.

In the postoperative phase, the pulse emission of the implanted electrodes is fine-tuned to achieve the best treatment for the patient. This setup process can be done in the clinic or at home and provides considerable flexibility for the patient.

3.1.1 Contribution

The current state-of-the-art DBS surgery [43] heavily relies on the surgeon's capability to build a mental model of the current electrodeposition and the classification of different tissue types based only on the auditive feedback provided by the MER. The auditive signal can be visualized as a wave representation during recording, but most surgeons rely on listening to the signal. Based on the auditive and visual representation of the MER, surgeons have to decide if the targeted tissue is reached or not. Currently, available classification software on the market that supports surgeons in this challenging task is still in its infancy.

While the preoperative imaging and placement technology is highly developed, intraoperative real-time imaging and analysis software [312] has not been integrated into the daily procedure. Even though imaging data is mandatory for planning these procedures, to this point, no effort was given to integrating these domains into the surgery itself.

The goal of the presented research is to support surgeons during the placement task of the DBS procedure. It utilizes a machine-learning classification approach

for tissue type detection based on the recorded MER signal. Further, it provides various visualization techniques to provide visual feedback, which is currently not present during the procedure and will support surgeons in their decision-making tasks. With the presented real-time analysis and classification approach, the system aims to improve the surgeon's confidence in their decision-making process. The implemented visualization fuses the intraoperative electrode recording of the MER with the preoperative imaging data and reduces the mental processing power needed during the surgery. The complete system is highly flexible and can be easily integrated into the daily DBS procedure.

3.2 System

The presented system and research results support doctors during DBS surgeries and decision-making tasks by adding additional visual feedback methods to the already established procedure. Looking at the data practitioners have currently available during the implantation of the electrodes, it became clear that visual feedback could be constructive for relieving the mental processing mandatory during every surgery. Not only does the system display visualization to the surgeon currently unavailable during the procedure, but the system is also capable of classifying the recorded tissue activity using a straightforward neural network. The upcoming sections of this dissertation will detail the solutions found for intuitive intraoperative visualization and classification.

Structurally the following part will first focus on micro-electrode recording, the acquisition, and the transfer of the data, followed by the processing and classification. Furthermore, the results of the classification processes are evaluated to determine the system's capabilities of determining the correct target tissue.

The second part of this chapter focuses on visualizing the collected data. The implemented system supports two distinct visualization types. The first visualization type is tailored toward directly visualizing the MER data and the classification results. The design of the visualization leans on already well-established visualization systems found for the MER recording enhanced with the results of processing and classification stages of the system. The second visualization space found in the system focuses on merging the preoperative imaging data with the intraoperative MER data. The results presented by this type of visualization were priorly only achievable as mental models based on the experience of practitioners.

3.2.1 Tissue Type Classifcation

A significant portion of the research in this work is the processing and classification of the patient's brain activity in real-time. The upcoming sections will focus on data acquisition, processing, and classification processes, to achieve the results presented in Section 3.3. The complete data pipeline is automated during run-time and requires no user input by the practitioners.

Data Acquisition

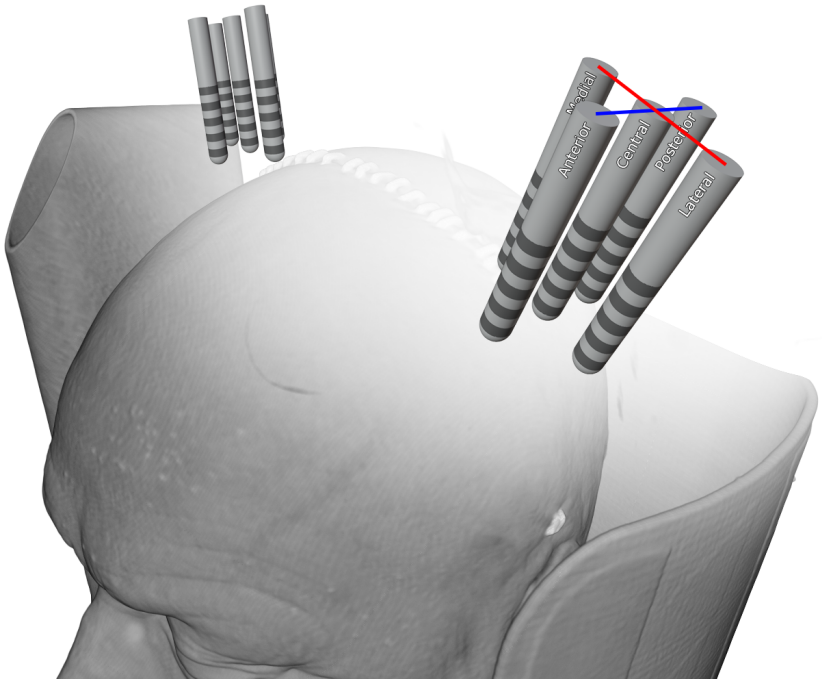


Fig. 3.1 Up to five recording electrodes can be used for each hemisphere during the location process. These are arranged as displayed in the Figure above and named by medical conventions.

The first step of the classification pipeline is to gain access to the recording data during the placement process. Multiple recording electrodes are injected into the patient's brain for the location process of the DBS surgery. Each recording electrode is displaced by several millimeters (see Figure 3.1). By using multiple recording electrodes to determine the final location of the permanent emission electrode, uncertainty can be minimized introduced due to possible errors in imaging data and human uncertainty during the planning phase.

The number of recording electrodes and data depends on the MER system and the surgeon's personal preferences. Most systems support up to five recording electrodes,

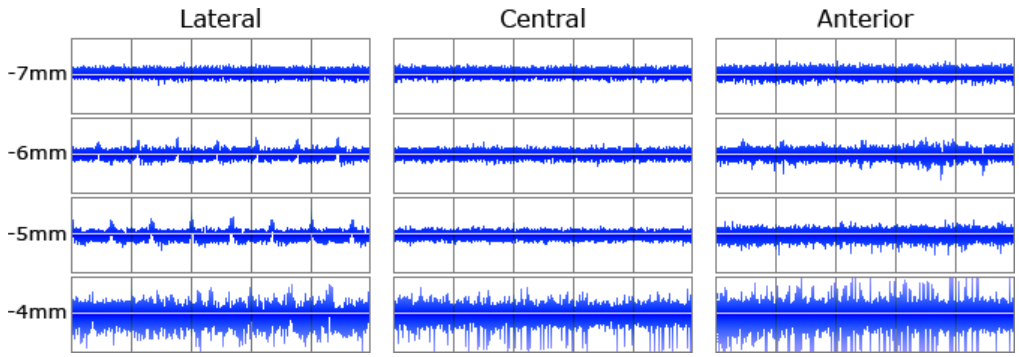


Fig. 3.2 The gold-standard visualization typically available during DBS procedures is the waveform representation of the currently penetrated tissue's low- and high-pass filtered data. For each inserted electrode, the waveform is displayed with its corresponding distance to the planned target location.

whereas in clinical practices, most surgeons use between two and four electrodes during the recording process. As mandatory for DBS surgery, the patient is awake to record the brain activity of several motoric tasks during the procedure. The recording electrode provides direct feedback on the brain activity transmitted to the recording system using thin wires. On the recording system, the data is displayed in the form of a time-dependent waveform (Figure 3.2).

In the case of this project, practitioners used three of the five electrodes that could be used during the placement procedure. Figure 3.1 displays the possible arrangements of MER electrodes. In this research, central, lateral and anterior probes were used.

The used recording system samples the brain with a frequency of 20 kHz, providing an extensive frequency range for further processing and classification. During the procedure, the recording system transfers the raw recording data directly to the presented supporting software using an ethernet connection between both devices. The data protocol between both devices is vendor-dependent but straightforward to implement.

The actual process of recording the patient's brain activity varies from hospital to hospital and heavily depends on the preferences of the surgery team. However, the overall recording process can be described as follow. The first data recording starts roughly 10 to 15 mm away from the planned target tissue. A single recording can range between 2 and 20 ms, depending on how long practitioners need to listen to the signal to interpret the brain activity. Once the data recording stops, the electrode is moved along the planned trajectory for a step size between 0.5 mm and 1.5 mm. The last recording is usually taken between 5 to 10 mm behind the planned location. As

mentioned, the parameters of starting location, end location, step size, and recording duration vary between hospitals, systems, practitioners, and patients.

For this research, some parameters were fixed. Therefore the first recording location was located 10 mm before the preplanned target location and the last 5 mm behind the target tissue. The recording duration was at least 7 seconds, with no upper bounds for the surgeons. The step size between recordings was fixed to 1 mm as it was the standard for the participating practitioners. While moving the electrode from one location to the next, the recording stopped removing noise introduced due to the movement of the electrode itself.

For the classification pipeline of this system, the set of parameters results in 15 recordings for each used electrode. Since three electrodes are used in parallel for each hemisphere, the system can work with 90 recordings during one DBS procedure. Each recording provides a data length of at least 7 seconds sampled at 20 kHz. The recorded data is further processed during the next step and later classified based on a supervised learning approach.

Data Processing

The MER device's recorded data represents the patient's brain activity during the procedure at the given location of the electrode. It is helpful to process the raw data for a better insight into the recorded activity. By consulting highly experienced practitioners, a processing pipeline was integrated into the intraoperative system focusing on parameters most helpful during the procedure with the primary goal of removing frequency ranges that practitioners interpret as noise.

The recorded signal is transmitted as a 16 bit unsigned short data stream, mapping the 20 kHz signal to values between 0 and 65535. As displayed in Figure 3.3 especially, low frequencies make it complicated to interpret the nuances of the signal. Based on the feedback provided during the integration of the system and already consumer MER systems, low- and high-pass filters were integrated to reduce the signals frequency range. The chosen frequency band left in the signal range from 250 Hz to 1 kHz. As also displayed in Figure 3.3, the difference between the unfiltered and filter data representation is visible and already provides an easier differentiation between brain tissues for skilled practitioners.

While for experts, the filtered signal already provides a better overview of the currently penetrated brain tissue, other visualization forms can provide additional insight into the data. The system transforms the filtered signal into its frequency

domain by applying a Discrete Fourier Transformation (DFT) [146] to offer an additional visualization domain to the practitioners. Figure 3.3 presents the waveform representation and the frequency domain side by side. Where practitioners can quickly glance at the overall tissue activity using the waveform representation, the frequency domain provides more detailed information about which signal frequencies are the most active. Based on the frequency range between 250 Hz and 1000 Hz, the result of the DFT represents 750 individual frequencies which can be linked to various tissue types based on the activity and electrode location.

During the procedure, the processed data is displayed to the surgeons in real-time within the system and supports the decision-making process. Furthermore, the classification of the target tissue, in this case, the STN, utilizes the result of the frequency transformation as input for the neural network.

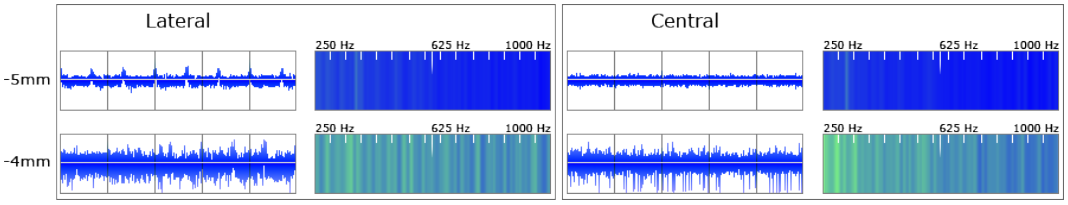


Fig. 3.3 Next to the waveform, the extracted and color-coded frequencies are presented. During the procedure, doctors can see at a glance which frequencies are more present in the signal than others.

Classification

One of the system's biggest perks is its ability to classify the recorded and filtered tissue activity during a DBS surgery. The classification approach chosen for this system is based on perceptron networks with supervised learning utilizing the years of experience of DBS practitioners.

The current implementation mainly focuses on detecting the subthalamic nucleus (STN) to treat Parkinson's disease symptoms. While DBS can be utilized for other neurological malfunctions, it requires different tissues to be stimulated, which are not covered by this network. The input data of the network utilizes the 750 frequency values obtained during the processing stage of the signal.

As for this use case, a single-layer perceptron with 750 input parameters could be too much to determine a separation between the STN and other tissues; the overall input dimension was reduced to 150. To reduce the input dimension, a window of 10 neighboring frequencies was used to build an average value. Instead of simply

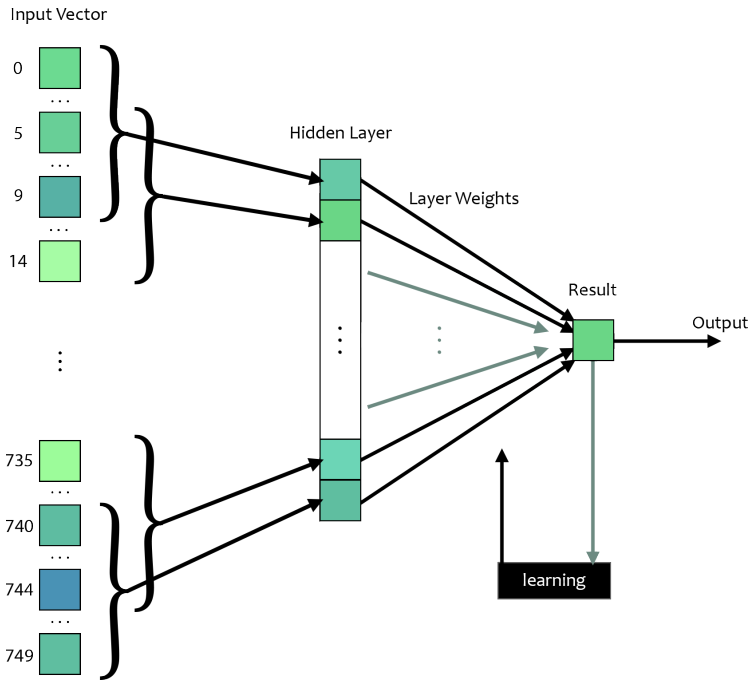


Fig. 3.4 The used CNN takes as input the frequency domain computed by the Fourier Transformation. For the input layer, ten frequencies are folded to a single value. The individual weights for each value are learned based on the output value and the pre-classified expected result.

building an average of every 10 consecutive values, the window is shifted by 5 instead of 10 values, as displayed in Figure 3.4. The overlap introduced due to this approach helped create a correlation between the network's neighboring input values and increased the network's robustness.

As typical for a perceptron, the function computed to determine the classification of the input data is described in Equation 3.1.

$$o = \begin{cases} 1 \mid \sum_{i=0}^{n} weight_i * input_i + \theta > 0 \\ 0 \mid else \end{cases} \tag{3.1}$$

Since a perceptron network is a linear classifier between two classes, defined by its output value of 0 for not being in the class, and 1 for being in the class, the system has to learn the weights for each of the 150 input values. In the case of the STN classification, 1 describes an input data set that correlates to STN activity, while 0 can be related to any other type of tissue.

A supervised learning approach is used to train the weights of the perceptron network. As the current gold standard requires, practitioners already have to classify the MER recording based on auditive feedback during the procedure. During training sessions, the already supervised data of prior procedures are used to train the network. Based on the network's output and the experts' expected classification, the individual weights are adjusted. The integrated perceptron follows the learning function described in Equation 3.2.

$$weight_i^{next} = \begin{cases} \text{correct classification} \mid weight_i^{old} \\ \text{result 1 but 0 expected} \mid weight_i^{old} - \alpha * input_i \\ \text{result 0 but 1 expected} \mid weight_i^{old} + \alpha * input_i \end{cases} \tag{3.2}$$

The learning rate of the network can be adjusted by α, where too high values can result in overfitting to individual input data and too low values in a network that does not converge against a minimum in a reasonable time. The achieved results of the integrated network for classifying the STN can be found in Section 3.3.1.

3.2.2 Intraoperative Visualization

The presented system supports various visualization types that aim to support the practitioners in their tedious task of tissue classification. While the current gold standard of intraoperative visualization for MER procedures is still the display of the raw signal, the supplementary visualization domains provided by the system provide more insight that should, even at a glance, increase the certainty of doctors. While practitioners have been trained over the years to interpret the raw signal, visual feedback can guide them to reach better results and help new surgeons understand the complex task of DBS procedures. For this goal, the system not only visualizes the MER device's filtered signal as it is already found on state-of-the-art hardware, but it also provides real-time visualization of the frequency domain obtained from the Fourier transformation, an interactive three-dimensional and slice-based visualization of the preoperative-imaging data. Furthermore, the classification process results are displayed within the MER visualization and the imaging data.

The desktop interface of the system is split into two interlinked visualization domains (Figure 3.5). One visualization area focuses on displaying the MER data in real-time. Alternatively, the frequency domain can be displayed as a color-

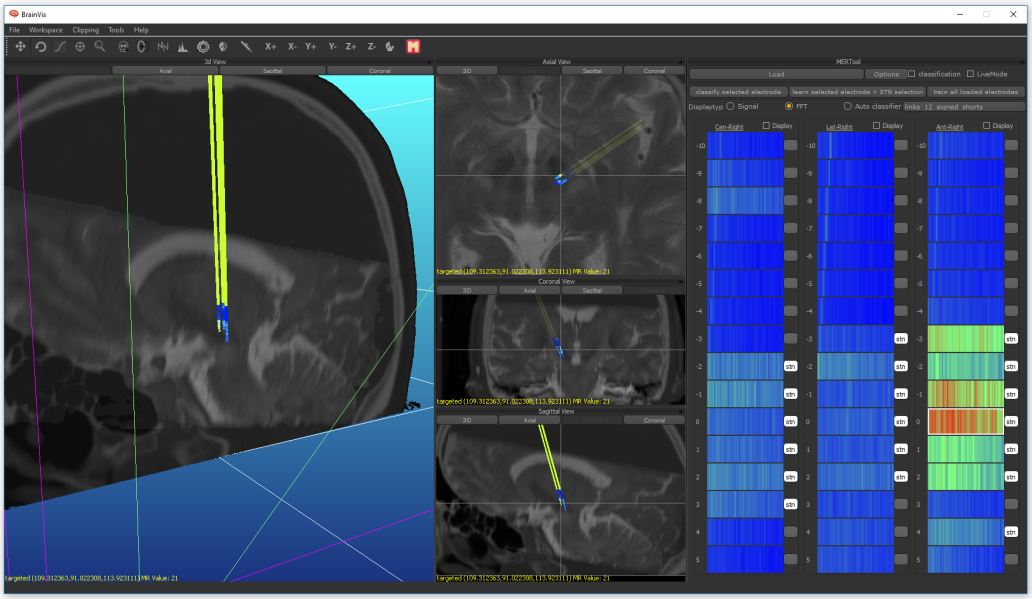


Fig. 3.5 The user interface of the system is separated into two spaces. The left part of the interface displays the interactive visualization of the preoperative CT and MRI data enriched with the MER results. The right side represents the processed and analyzed MER data in real-time.

coded spectrum while providing information on the classification result. The other part of the interface is used to visualize an interactive three- and two-dimensional representation of the preoperative imaging data. The imaging data is further enhanced using the results of the MER processing pipeline. For the visualization, the CT and MRI data is rendered using direct volume rendering to maintain as much insight into the data as possible. Since both data domains have to be displayed simultaneously, a part of this research was focused on accelerating the registration process of both data domains far beyond available consumer solutions.

As the system should provide as much guidance during the electrode placement process, the MER and imaging visualization domains are linked, making it straightforward to reference one data point in the other visualization domain.

MER Visualization

To be seen as an intuitive and valuable visualization for intraoperative use, the MER data visualization has to be understandable by a fast glance at the displayed images. For this purpose, the integrated systems provide two different types of visualization, as they are displayed in Figure 3.7.

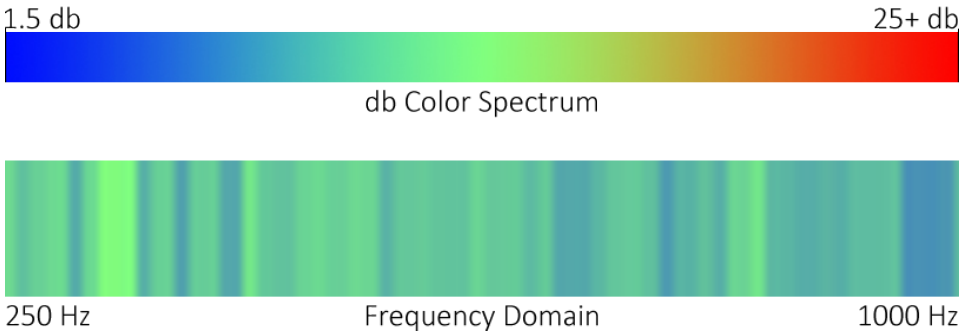


Fig. 3.6 The MER signal is transformed into its frequency domain. Frequencies between 250Hz and 1000 Hz are visualized within the software using the displayed color-coding.

The first visualization integrated utilizes the already well-established and known representation form of the signal represented in its waveform after applying the low- and high-pass filters. As practitioners know from the MER devices, the continuous recording of the tissue is directly displayed. As displayed in Figure 3.7 each inserted measuring electrode is shown next to the other in a single row. In contrast, the row represents the current distance from the target location marked by the millimeter description on the side. By integrating an already established and known visualization type, surgeons do not need to switch between different devices to gain all information they need.

As mentioned before, an additional visualization of the frequency domain is added to the system. Since humans often interpret colors faster than numbers, the individual frequencies found in the data are color-coded based on their dB value (Figure 3.6). The shown images represent the extracted frequency range from 250 Hz to 1000 Hz, sorted from small frequencies on the left to larger frequencies on the right side of the image. The color map for this purpose uses a linear blending between the three primary color channels of red, green, and blue. As demonstrated in Figure 3.6 low dB values are mapped to the blue, where green represents values in the middle and red high values. Values exceeding the displayed scale are clipped to the maximum. The displayed image is updated in real-time as the data is recorded.

Where the waveform representation can display a continuous signal over time, the frequency images are used to evaluate individual frequency ranges. As learned during the training of the classification network, it became clear that different tissue types react with different amplitudes in specific frequencies. Color coding these frequency ranges help surgeons during the procedure due to faster indicate specific tissue types compared to the waveform representation.

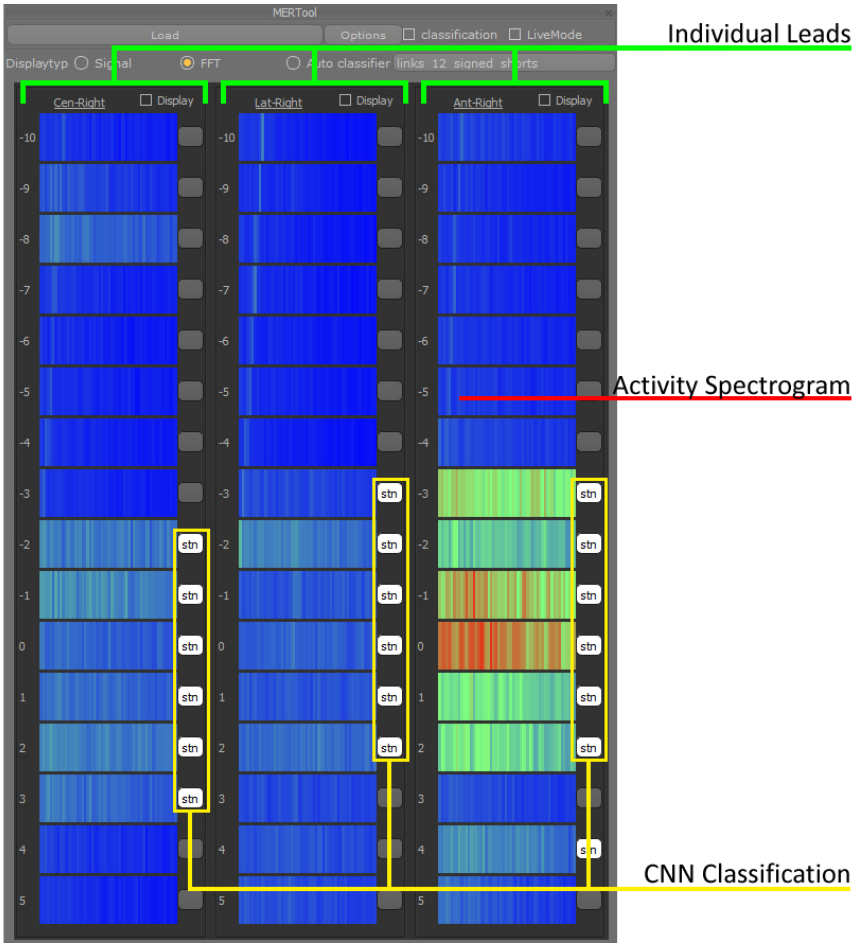


Fig. 3.7 Close-up of the MER-Widget: The number of displayed columns depends on the number of used recording leads. The tool allows switching between the visualization of the spectrogram and waveform. The automatic classification is displayed in real-time next to each recording.

The last part added to the individual visualization of each electrode is a tiny field representing the result of the classification network. During recording, the system evaluates and classifies the data using the perceptron network. If the network detects the targeted tissue, it lets the practitioner know by displaying the label "STN". By hovering over this label, more information is displayed, for example, the certainty of the classification. Surgeons can overwrite the classification result, triggering a learning procedure of the system to adjust the perceptron network.

3.2.3 Image Data Visualization

Besides visualizing, processing, and classifying the MER data in real-time, the presented system also supports an interactive visualization of the preoperatively

taken MRI and CT imaging data in the form of 2D axis-aligned slice views and a 3D volume visualization. The visualization includes the results of the MER processing pipeline. It will aid practitioners by relieving some of the cognitive load necessary to link the MER feedback to the spatial imagining data of the patient's brain.

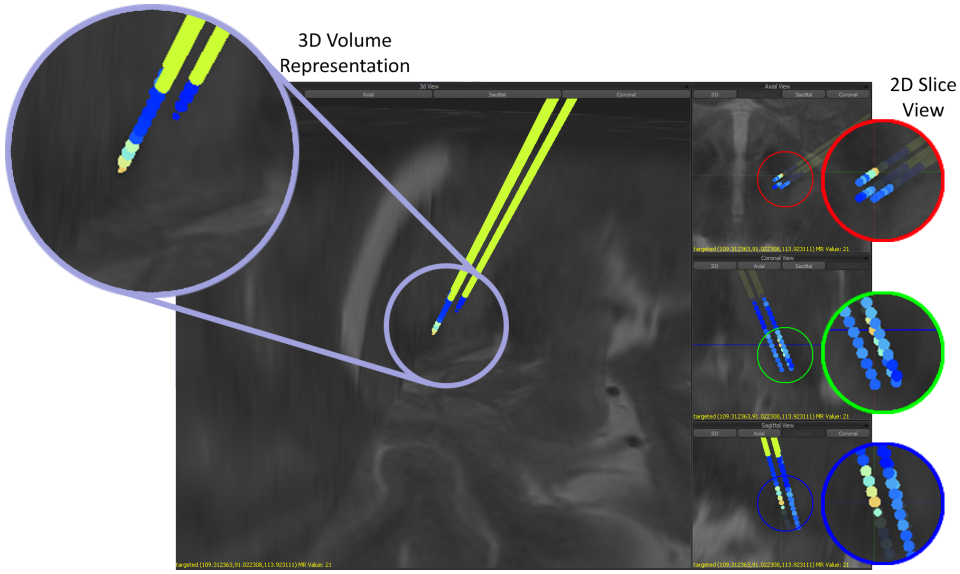


Fig. 3.8 The classified MER data is represented in the 3D and 2D visualization of the system. Surgeons can cross-validate the findings using the pre-operative imaging data and the intraoperative recordings.

The visualization has to fulfill a handful of constraints to be complementary for intraoperative use. One of the most critical points is the registration of both data set domains. The CT scans used for DBS procedures are taken directly before the surgery and include a stereotactic frame allowing surgeons to reference any point within. This frame is also used to reference the current location of the MER electrode during the placement procedure. However, CT scans are less detailed regarding brain tissue, requiring an additional MRI scan to detect the target area. Therefore, it is mandatory to precisely fuse the CT, and MRI scans to make correct predictions about the location of the electrode and the penetrated tissue. Fusing both data sets can be costly, time-consuming, and often requires manual adjustments. Practitioners can not afford to lose this much time during the procedure. Hence the system supports a GPU-accelerated registration technique for CT and MRI scans. Section 3.2.4 describes the approach in more detail.

Besides registering the two data sets, several other tasks must be fulfilled. Among other things, these would be the extraction of the stereotactic frame from the CT scan to recreate the reference coordinate system (Figure 3.9) or methods that integrate

an automatic selection of viewing parameters, such that practitioners do not have to interact with the system manually.

Fig. 3.9 (a) The preoperative CT imaging data includes information of the stereotactic frame used as a base coordinate system for the procedure. (b) By focusing on the most considerable quantities in the data set, it is possible to extract the structures of the frame. (c) The system uses the extracted information to locate the position of the MER electrode during the placement process.

Due to the experience with 2D axis-aligned slices, the rendering system supports the visualization of each vital view, including the axial, lateral, and sagittal axis. During the procedure, each axis automatically focuses on the current electrode position and highlights the position, including a graphical representation of the recording data.

For the 3D visualization, direct volume rendering [175, 161] was applied. Visualizing three-dimensional volumetric data can be very computationally expensive but provides various benefits to other methods, such as indirect volume rendering techniques. Section 2.2 provides the fundamental background to volume visualization as it is used in this research topic.

While the system and the visualization should work independently, not requiring inputs and adjustments by the practitioner, it supports various techniques for interaction and filtering of the visualized data. One critical aspect of visualizing volumetric data is occlusion, especially when working with multiple volumetric data sets and lead geometry. Clipping methods must be used to gain a clear view of the current electrode position, removing occluding data from the view. A solution commonly used is a clip plane oriented along the axis of the volume. Another approach is clipping geometry in the form of cubes, spheres, or other geometric shapes. In the medical context, cubes are often found as clipping geometry. Focusing on a location within the data set removes a cubical shape between the location and the viewing

position, as shown in Figure 3.5 and Figure 3.8. The benefit of this approach is that practitioners can still reference each axis of the data made visible due to the clipping geometry.

1D transfer functions (Section 2.2.3) are integrated into the system [184] to adjust the visualization further. These functions are commonly known as windowing functions by most practitioners.

As mentioned, the MER pipeline's results enrich the volumetric data. For each recording, the system stores the location within the stereotactic frame. The 2D and 3D visualization utilizes the positional information and the classification results to display geometric primitives in the form of little spheres at the corresponding location. As displayed in Figure 3.8, these spheres are color-coded by one of two options. The first option is the averaged dB value of all 750 stored frequencies. Similar to the MER visualization, the color-coding helps to obtain a fast overview of the overall activity of the tissue. The second option is a binary color-coding based on the result of the neuronal network. Results labeled as STN are color-coded in white, while other results are encoded in black.

3.2.4 Image-Data-Fusion

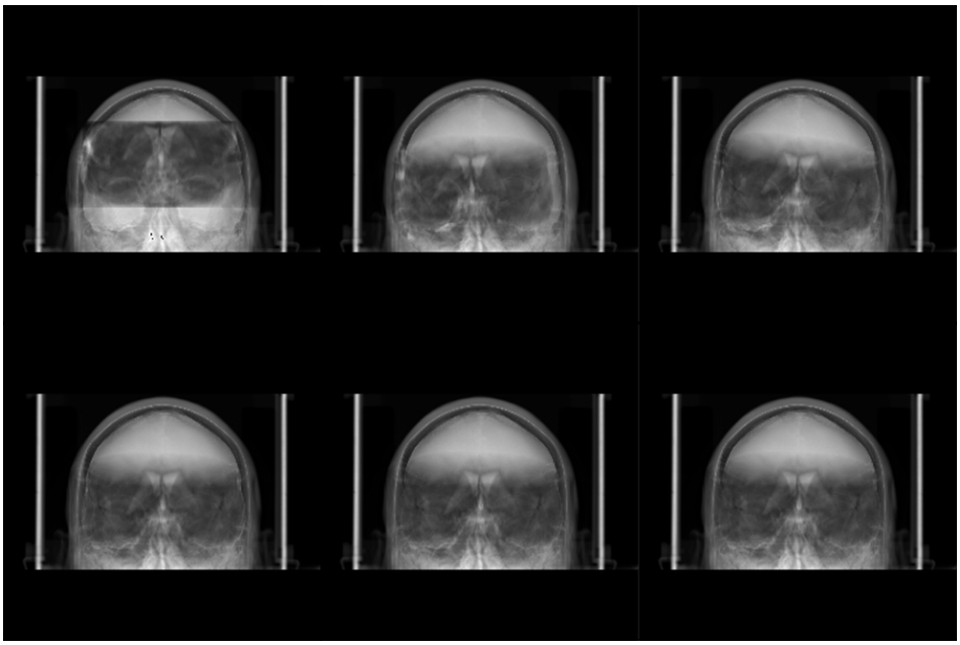


Fig. 3.10 Using the GPU accelerated method based on a gradient descent approach the system automatically fuses the MRI and CT images in milliseconds. Top left: starting position; Bottom right: Final registration.

Fusing or Registering two different data domains, such as CT and MRI, can be a tedious and complicated task. The approach used for this system is heavily automized and relies on GPU acceleration techniques for fast and reliable results. Compared to solutions in clinical practice, the integrated approach presents more accurate results according to expert users' eyes and is magnitudes faster than available CPU solutions.

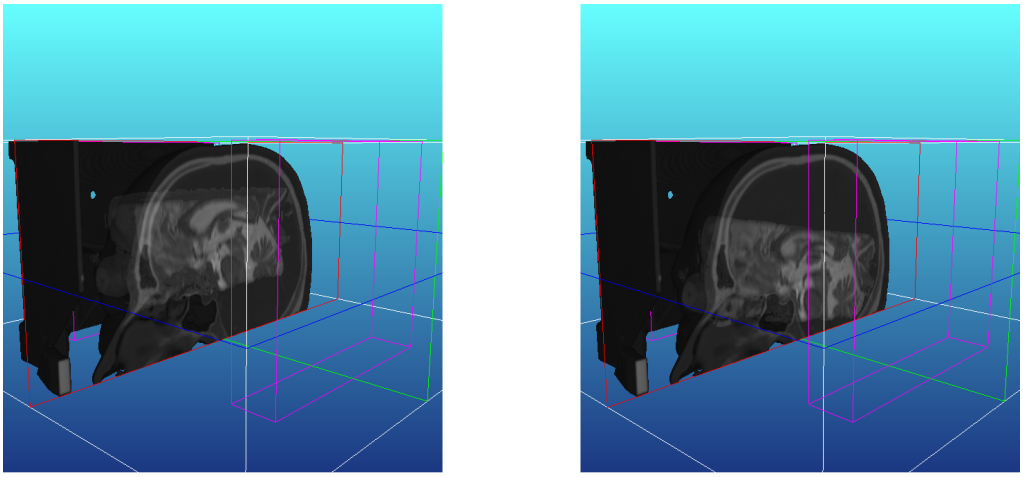
The fundamental idea behind the registration process is similar to a gradient-descent approach, looking for the minimum difference between both data sets. Using the parallel computing power of GPUs, it is straightforward to compute the difference between both data sets. The GPU can traverse both sets and subtract the individual values by scaling and positioning both data sets on the same axis. Folding the resulting 2D image on the GPU is another way to compute a single score for the current position quickly.

While computing a single score for both data set given a single position can be fast, the challenge lies in locating the best possible fit. By assuming the CT scan to represent the target data, the MRI scan can be placed in an infinite number of locations within it by adjusting the position and rotation. In contrast, the scaling of both data sets can be assumed to be fixed based on parameters provided by the scanner.

A gradient-descent approach aims to locate a minimum by evaluating the gradient of a function. Similar to such an approach, the implemented system utilized the difference between both data sets to evaluate the next possible position. During the registration process, the system first scores the current position of both data sets. Afterward, 12 possible candidate positions are computed, including a shift in the positive and negative x, y, and z coordinates, as well as a rotation along each of these axes.

After evaluating the score of each candidate, the results are compared to the current position. The results are enqueued into a system tracking the already computed scores for later referencing. The registration process is continued as long as one of the candidates is associated with a smaller score compared to the current location, making the smaller candidate the following evaluated location. This approach will converge against a locale minimum, with no possible candidate offering a smaller score.

While this approach results in a minimum difference between both data sets, the result is not guaranteed to be the global minimum or the desired registration result. The detection of the correct minimum is a fiddley act resolving around parameters



(a) Initial Positioning of Both Data Sets (b) Positioning after Registration

Fig. 3.11 The initial position of both data sets is located in the origin of the rendered scene (a). After computing the correct alignment of both data sets, they are displayed accordingly in 3D and 2D (b).

such as the distance between evaluated candidates. A too-small distance between the current location and the possible candidates could result in a local minimum not even close to the desired result; too-large distances could overshoot and miss the target.

Several tweaks have been added to the system to increase the likelihood of converging against the correct result. The first one is a dynamic step size for the distance of possible candidates. Starting with more considerable distances helps to overcome local minima at the beginning of the process. By tracking the scores of successive locations, the step size can be adjusted to become smaller from iteration to iteration. The exact step size and adjustment depend on the resolution of the data sets presented to the system, where larger data sets, including more of the patient's head, can converge faster than data sets only including partial information about the patient's brain.

Another tweak that significantly increased the system's robustness is the exclusion of specific values within the CT scan. Within CT scans, bone structures are significantly prominent in the imaging data, which are hard to see in MRI images. Excluding Hounsfield Units [256, 2] that are tailored toward bone structures resulted in much better results.

Finally, the last tweak added to the system is a random selection of the starting position and starting step size. Since the overall registration performance far exceeds commodity solutions, as it is displayed in the evaluation in Section 3.3.3, the system

computes between 4 and 10 results based on random starting locations. At the end of each run, the difference between the results and the position is evaluated and compared until a set threshold is reached.

3.3 Evaluation

The presented system includes a variety of features that can be evaluated. Some of these features will be dissected and evaluated in the following section based on different aspects. The first topic covered in this section is the perceptron network used to classify the recorded MER data. The exact training process will be explained with detailed insight into the accuracy of the classifier. The classification topic is followed by an evaluation of the rendering environment with a significant focus on expert feedback for intraoperative use. The last topic evaluated in this section is the GPU-accelerated registration of CT and MRI data sets. Besides comparing the performance of the approach to other systems, it is also evaluated against the manual registration results of skilled practitioners.

3.3.1 Classification Results

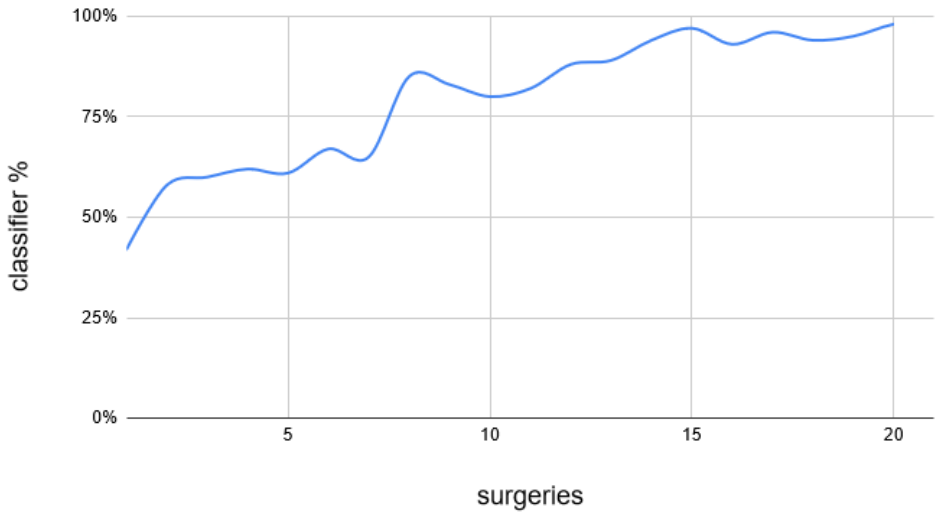


Fig. 3.12 The classification network was trained after each surgery utilizing the newly recorded data supported by the supervised classification of field experts. After each training session, the best network was used as a candidate for the upcoming procedure. After ĩ4 sessions, the network reached a classification accuracy of 90%, which was furthermore validated in the following surgeries.

The neural network described in Section 3.2.1 was trained using a supervised learning approach. While working on the system, DBS procedures were recorded weekly, including supervised classification of the STN by the executing practitioner. For the here presented results, a total of 1890 pre-classified recordings were gathered. Each data set represented a recorded tissue activity between 7 and 13 seconds. The entire recording was transformed for the training set using a Discrete Fourier Transformation. To further increase the training set, 5-second-long snippets were picked out of the signal at random starting locations and transformed into their frequency representation, resulting in slightly different frequency bands compared to the complete recordings. Over the training process, the network could be trained on 3780 pre-classified inputs.

The best possible classification results were noted each time new data was added to the training set. The best possible network is defined by the total number of correctly classified data sets and the minimum summed error of the wrong classified inputs. Figure 3.12 displays the achieved classification accuracy of the network after every training session. In the beginning, the network hovered around a correct classification rate between 60% to 70%. Especially the first weeks of training resulted in many overfitted networks. Some of these networks already achieved classification rates of 85% but introduced extreme error sums for the wrongly classified inputs.

A considerable spike in accuracy could be achieved after the eighth session was added to the system, while after 14 sessions, the network achieved an accuracy of 90%. The network, to this point, trained on all available recording sets, reached a validation rate of 98.6%.

This network was later evaluated in 6 additional surgeries in which practitioners first classified the recorded MER data by themself and afterward compared their decision to the result of the network. In two of these 540 recordings, the practitioner first decided differently than the neural network. In both cases, the practitioner did not classify the tissue as STN when the network did. However, After further investigation of the data, the surgeon concluded that both cases were STN, resulting in the best treatment symptoms for the patient in one of those cases.

3.3.2 Implications of Intraoperative Visualization

A primary goal of this research topic was to provide a supportive tool to practitioners to reduce the cognitive load during DBS electrode placement. Practitioners' feedback has to be considered to conclude if this goal could be achieved.

During the development of the system and after reaching set milestones, feedback from the practitioners was collected through free-text questionnaires. The questionnaire included three aspects of the system; The MER visualization, the Classification process, and the visualization of the imaging data. For each of these topics, practitioners were asked to answer whether the system supported their decision-making process or not. In free text, surgeons could provide information on what parts of the system could be enhanced and what parts already significantly benefit their workflow. As expected, this feedback had a significant influence on the development of the system, ranging from feature requests to changes in the color mapping of the frequency domain.

While at the beginning of this project, practitioners mainly focused on feature requests in their feedback, including automatic camera positioning for the visualization or clipping methods similar to commodity planning software, over the weeks, the feedback shifted toward the classification results and the visualization benefits. More often, it was noted that the color-coded visualization was faster to interpret than the waveform signal of the MER device. The classification was highlighted after some weeks since it was perceived as "pretty accurate". The visualization was mostly highlighted for its cross-referencing between the MER recording and the preoperative imaging data. On several occasions, it was pointed out that the three-dimensional visualization, enriched with the electrode's geometry, helped reduce the uncertainty present after evaluating the MER signal.

After a few months, the questionnaire was adjusted for a final round. Doctors should assess whether the system at its current state enhanced the overall DBS procedure compared to procedures without using the classification and visualization system. The feedback obtained in this round indicated that practitioners preferred the intraoperative visualization of the MER's spectrogram and the volumetric visualization since both types of visualization present a significant relief in the cognitive load required by the surgeon. For the classification, mixed results were achieved. While doctors were impressed by the network's accuracy, some did not see the benefit since they themself came to the same result. However, these doctors could see the system as beneficial for critical recordings. Especially doctors with less experience saw the classification system as a beneficial training tool while leaving the final decision to themself.

The feedback gained through these questionnaires should only be seen as preliminary results. Several of the participating doctors accompanied the development of the system from the beginning helping to design the system to their needs. For

	Set #1	Set #2	Set #3	Set #4	Set #5
Practitioner 1	*173.5 s*	*182.4 s*	*121.8 s*	*277.3 s*	*184.8 s*
Practitioner 2	*202.1 s*	*198.4 s*	*157.3 s*	*290.5 s*	*174.3 s*
Medical Workstation	*62.1 s*	*53.5 s*	*41.7 s*	*31.6 s*	*47.1 s*
Single GPU Path	*0.213 s*	*0.284 s*	*0.174 s*	*0.285 s*	*0.391 s*
Full GPU Fusion	***0.981 s***	***1.873 s***	***0.576 s***	***1.327 s***	***2.839 s***

Table 3.1 The registration of a CT and a MRI data set were compared regarding the task completion time. Two practitioners manually fused both data domains to be subjectively precious enough for daily tasks. The medical workstation tested represents the hard- and software available at the evaluation time. The results were compared to the automated GPU-accelerated data set registration.

a more profound study, the system has to be evaluated in several medical facilities over the long term.

3.3.3 Registration Evaluation

Section 3.2.4 focused on the process of fusing MRI and CT data domains. For the evaluation of the system, two aspects have been investigated; The task completion time and the precision of the resulting registration.

The results of the GPU-accelerated approach are compared to registration results acquired from field experts doing this task on a daily based, fully manual. Most surgeons elaborate that, in medical practices, the tedious registration task is done by hand since available software solutions either provide unreliable results, require additional manual adjustments, or take too long to converge for time-critical interventions.

The registration parameters provided by field experts include placement information, including the translation and rotation values which describe the placement of the MRI data set within the CT image. These values are comparable to the values returned from the GPU registration method. Besides comparing the difference in position, the absolute error of both setups is also compared.

The registration processes were tested on five randomly selected and anonymized CT and MRI scans (Table 3.1). Two practitioners participated in this trial. They had to manually fuse the data sets as they would for any surgery using their preoperative planning software. The time was noted together with the positional and rotational information. Afterward, the software solution available within the planning software was tested as a reference for later comparison to the presented approach. As the

field experts were present during this step, feedback was collected on the registration quality. In the end, the proposed GPU-accelerated method was tested and compared.

Figure 3.13 presents the results of the measured timing. As expected, manual registration cannot compete with other solutions, with a median of 196.24 seconds across all five tested data sets. The automated approach found in the planning software could provide results within an average of 47,2 seconds. By far faster than the manual approach, but still not satisfying to the practitioners, as it will be outlined in the precision evaluation of this section. Finally, the GPU-accelerated approach of this research has been evaluated. Two different setups are displayed in Figure 3.13. The first is the execution of a single registration loop resulting in a fast approximation of the target registration. The second one is the result of the repeated execution of the registration process based on a random starting position, as mentioned in Section 3.2.4. For a single registration loop on the tested data sets, the approach converges within 238 ms. The second test evaluated the repeated execution of the registration pipeline. On average, the registration process finished after 1519 ms. On data sets with a small registration error, the repeated loop exits the function earlier than on data sets with a more considerable error between individual registration parameters.

While the execution time of the presented approach far exceeds commodity solutions and manual registration, the quality of the registration has to be considered too. Table 3.2 displays the difference in space between each solution compared to the manual registration of practitioners. Only the translation could be compared, as the automated approach of the medical workstation did not provide rotational information. Manual registration is seen as the gold standard as surgeons feel most confident in their skills when it comes to the task of domain registration. The results

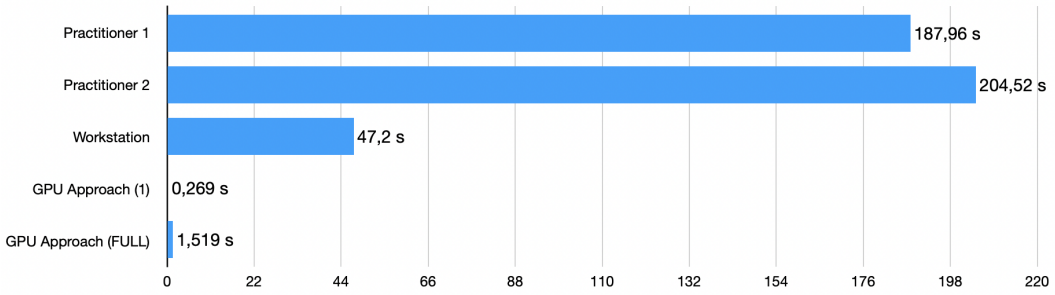


Fig. 3.13 The Figure displays the average task completion time across all five tested data sets in seconds. As visible, the GPU-accelerated approach increases the overall registration.

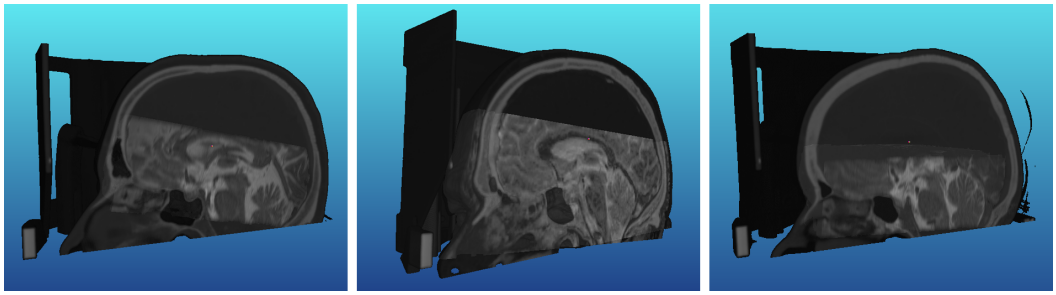


Fig. 3.14 The Figure displays three example registrations computed by the system in a three-dimensional visualization. Each of the shown registrations was evaluated by practitioners and passed their evaluation.

in Table 3.2 demonstrate that automated approaches found in preoperative planning software vastly diverge from manual solutions. The results missed the actual target setup by several mm or even cm, making them often not feasible for surgeons to be considered as a possible solution. Based on the collected feedback, only one of the solutions achieved by the software was of interest to the practitioners, as the others were too far off and required manual adjustments. While one of the single loop GPU registrations displayed in Tbale 3.2 also missed the target setup to a large degree, the other results came close enough to the results provided by the practitioners.

By presenting the results to field experts, it could be concluded that the resulting registrations came close enough to the manual solutions that the difference could be seen as a margin of error. As a closing thought to the evaluation of this registration approach, while the difference to manual registration is held to a minimum, the expertise of field experts shall not be neglected regarding the health and well-being of patients. Therefore, field experts should also overlook the results and adjust them if necessary. However, the increased performance compared to available software solutions should enhance the overall registration process.

3.4 Preliminary Conclusion

Throughout this chapter, an assistance system for neurostimulation therapy was introduced. The system is tailored toward intraoperative use and focuses on supporting the electrode placement by offering interactive real-time visualization and classification of the patient's data. Several challenges had to be tackled to create a system that enhances the already established procedure without introducing additional work to the practitioners.

	Workstation Deviation	*Single Path Deviation*	*Full Fusion Deviation*
Set #1	(-4.21 , -0.19 , -1.97)	(-0.84 , 0.93 , -0.72)	(-0.03 , -0.13, 1,23)
Set #2	(-3.88 , 1.83 , -0.94)	(1.84 , -1.52 , 0.55)	(0.42 , -0.80 , -0.21)
Set #3	(1.98 , -11.94 , 5.12)	(-3.85 , 9.87 , 3.84)	(1.09 , -2.68 , -0.33)
Set #4	(-25.01 , -38.77 , 9.22)	(11.94 , -8.15 , 7.28)	(1.01 , -0.09 , 1.19)
Set #5	(3.00 , -1.43 , 4.74)	(1.17 , -0.96 , 2.24)	(1.18 , -0,92 , 2.11)
Abs. Avg-	*(7.62 , 10.83 , 4.40)*	*(3.93 , 4.29 , 2.93)*	*(0.75 , 0.92 , 1.01)*

Table 3.2 The table above displays the difference in mm on each axis compared to the manual image fusion of field experts. The table shows that the presented GPU accelerated approach is closer to the surgeon's manual results compared to state-of-the-art available hardware in the clinic.

The recorded MER data had to be processed in real time to present immediate feedback to the practitioner. The processed and transformed data was mapped to a color representation of the individual frequencies found within the signal. After the first feedback from practitioners, this type of visualization provides a faster indication of the tissue type than the traditional waveform representation of the MER devices.

Furthermore, the processed data is classified on its tissue type using a supervised learned perceptron network. The implemented network provides a real time classification of the subthalamic nucleus. Based on the achieved results, the classification exceeds available methods and provides convincing results to the surgeons helping them in their decision-making process.

To further aid surgeons in the process of electrode placement, the system supports visualizing preoperatively acquired imaging data in the form of CT and MRI data. The patient's imaging data plays a crucial role in the planning phase of the surgery but is not present during the procedure. The presented system brings these data domains into the surgery. Using volume visualization techniques and traditional slice-based visualizations, surgeons can cross-validate the findings of the MER device with the imaging data. Including visualizations of the MER results in the 3D and 2D visualization makes it much more straightforward for surgeons to understand the recorded data.

Another feature of this system is an accelerated registration system for CT and MRI data. As the registration of both data domains is necessary for the planning of the surgery and the intraoperative visualization, a solution was implemented exceeding state-of-the-art registration software with much higher accuracy.

As the feedback of participating practitioners indicates, the presented results of the system seem reliable and helpful during the electrode placement. Especially practitioners with less experience appreciated the tissue classification, which helped them in decision-making. The intraoperative visualization was additional support in multiple surgeries where practitioners were unsure about the placement location based only on MER results.

3.5 Discussion and Outlook

This chapter introduced a system that aids neurosurgeons in their daily clinical task of DBS electrode placement. It was developed in close relationships with practitioners and has overgone several iterations. Based on feature requests and feedback, the system was adjusted to the practitioner's needs and now provides as much insight as possible. While it received positive feedback for its ease of use and the enhanced certainty it provides during the procedure, several aspects of the system could be enhanced in future versions.

The neural network presented in this chapter can only detect the subthalamic nucleus and is trained based on data provided by the participating practitioners. While the network results are promising, and practitioners are optimistic about the classification, it should be considered that the trained network is tailored toward the provided classification data of the participating neurosurgeons. In the future, the number of surgeons providing data should be expanded to increase the robustness of the network. Increasing the number of participating neurosurgeons makes the network less likely to be overfitted to only a small group of people and their classifications.

Furthermore, the network is currently only a single perceptron layer. While it can still provide an excellent classification of the data, the future will show how likely it is to keep the network to this level. It is expected that other neural networks will provide better results, especially once other tissue types should be classified. The single-layer perceptron can only be used to classify a single tissue type. For multiple tissues, additional perceptron layers must be trained and compared to each other for a final judgment of the correct tissue type.

While this kind of classification is still in its infancy for DBS surgeries, the results are promising and should be further investigated. Nevertheless, it should be mentioned that the final decision still has to be left to the surgeon. Another aspect that should be considered is other neurostimulation therapies that do not use MER

to detect the target tissue. Other inputs, such as motor neuron activity or patient task performance, should be added to the system and considered for classification approaches.

Considering the visualization aspect of the system, it utilizes well-established visualization methods that were absent during the surgery to this point. Visualizing CT and MRI data can help surgeons in their daily tasks and offer another layer of validation during the placement of the electrode. Another domain that could enhance the visualization for DBS procedures is diffuse tensor imaging (DTI) [168]. While not included in the current system, it should be straightforward to be added to the visualization if it is desired and beneficial.

The registration process presented in this chapter far exceeds the quality of software available in clinical practice. Not only is the placement of this approach more accurate, but it is also much faster. However, the registration, or image fusion approach, should be tested with more data sets. Not only should the number of data sets tested be increased, but more should different resolutions of CT and MRI scanners be evaluated. The approach results in a local minimum between both data sets, which the imaging quality can influence.

Chapter 4

Benefits of Virtual Reality for Data Analysis

Due to the prolonged cooperation with practitioners, and the insight gained into neurostimulation procedures, the question arose whether other stages of neurostimulation therapy could benefit from enhanced visualization techniques. While mobile visualization has been researched for the postoperative care [303], the gold standard for the preoperative planning phase still relies on workstations with two-dimensional flat desktop monitors. The typical data during the planning phase are three-dimensional volumetric recordings of the patient acquired from CT or MRI scanners. Despite the desire to analyze the data in a three-dimensional space, the application of flat monitors reduces the data to be a projection onto a two-dimensional canvas. A suitable solution for spatial visualization is seen in virtual reality hardware providing immersive visualization and interaction in a stereoscopic environment. While virtual reality hardware has become more available to the broad public at a low cost, most application fields for VR have to fall back to video games [166, 326] or manufacturing [177]. Before diving deeper into the concepts and ideas of how virtual reality could enhance the preoperative planning phase of neurostimulation therapy, a long and extensive literature review has been conducted on the question of how VR is commonly used in the field of scientific visualization and data visualization. The upcoming chapter reflects on this literature review and presents the fundamentals that led to the ideas and motivation for the later topic of volume visualization within VR space.

4.1 Introduction

In our modern society, data surrounds us all the time. As the digitalization of our lives advances continuously, data is becoming more and more critical not just to data analysts and scientists but also to educators and personal use. As the amount of data collected increases at a high rate, the capacity to analyze data has not been able to keep up with this pace [150]. Concurrently, VR technologies have conquered the mass market, increasing access to VR tools for a broader audience. In particular, VR applications for smartphones or portable VR tools contribute to the accessibility of VR technology. Even big technology companies such as Meta, formally known as Facebook, restructured their company and now focus on the metaverse [203, 85], which can be seen as a digital, and entirely virtual universe existing next to our natural world.

The interpretation of data is based not only on the mental processing of the raw and unfiltered data alone. Due to the amount of data collected daily, transformations, visualizations, and interactions with the data have also become highly necessary. Therefore, computer-based visualization systems for all kinds of specific use cases have been developed over the years. Nevertheless, researchers have stressed that those systems alone cannot accomplish the scientific inside into the data without human interpretation [306]. Following the saying "a picture is worth a thousand words," it is mandatory not only to visualize the data but more understand the human perception and interpretation of it. Human pattern recognition skills are highly developed and may even outperform technology in detecting anomalies in some aspects [73]. The importance of our visual skills becomes apparent when considering that our vision is the dominant sense, which makes up 70 percent of all sensor receptors in the human body [192]. Instead of improving technologies to become better than the visual pattern recognition of humans, the objective of many scientific types of research and technological developments has been to enhance the synergies of these two.

One research field that focuses on the visualization and understanding of data is visual analytics, which is a key in decision-making [250, 188]. As an iterative process consisting of collecting information, pre-processing data, representing data, and providing interaction [286], it is also helpful in terms of scientific visualization when it comes to understanding VRs applicability to fields such as volume visualization. Researchers have shown that VR can provide a better user experience [204, 14, 51, 249] while analyzing visual data. As a result, the leading question of this chapter

is whether VR may have more advantages than just a better user experience in comparison to desktop visualization and how far it could benefit the field of scientific visualization.

The following sections investigate which application fields, tools, and data structures may be improved by VR. So far, much research has dealt with concrete aspects of VR, such as the role of depth cues or input devices. Simpson et al. [263] have found various opportunities and challenges concerning, for instance, display hardware, rendering, haptics, interaction, telecollaboration, software standards, and interoperability. Additionally, researchers have struggled to find enhancements of the user's performance in immersive analytics-related tasks compared to non-immersive approaches, despite the many reports of better user experience and satisfaction [92, 51, 233, 87]. Motivated by these controversial results, this work aims to examine in more detail if VR can be more than an exciting user experience. It is challenging to answer this question in general, considering that visual data analysis is mostly demanded by experts such as scientists and data analysts who need specific solutions for their tasks. Thus, this review will focus on the possibilities of VR for scientific data and non-professional applications.

Regarding the conquest of personal data concerning the amount of data collected and used for professional applications, it is essential to emphasize scientific data visualization and Big Data analytics. Also, professionals are more familiar with the data they are analyzing than the broader population and see data with different eyes.

This chapter is structured in the following way. First, it is crucial to share an overview of recent developments in data visualization, types of visualizations, and current application fields for immersive and non-immersive visualization environments before considering the benefits and drawbacks of both sides. Afterward, desktop-based visualization systems and solutions are compared to VR Visualization systems based on the effects of VR and immersive interaction possibilities on performance and usability. Finally, this chapter will conclude with the challenges, opportunities, and suitable applications for VR and proposals for future work.

4.2 Data and Data visualization

Humans are naturally, and intuitively spatial thinkers and analysts who automatically identify irregularities and patterns in their environment [195]. Many scientists, engineers, and software developers in data visualization have taken advantage of the ability of humans to process their environment by exploring the fields of data

visualization. Data visualization aims to support the users by visualizing the collected data during data exploration [198]. According to Keim et al. [149], the idea of visual data exploration is to present the data graphically, enabling the user to gain insight into the data and directly interact with the data. Visual data exploration is most suitable for datasets whose content is not fully known and whose exploration aim is unclear. According to Shneiderman et al. [261], visual data exploration typically consists of three steps: overview first, zoom and filter, and details-on-demand. There is a need for visual data analysis because several data analysis methods depend on the geometry and topology of the data distribution in the relevant dimension. If the user applies algorithms without seeing and understanding the patterns behind the data, the algorithm may lead to erroneous interpretation and thus cause misleading results. Visualization is the connection between quantitative data and human pattern recognition skills, which enables humans to comprehend data and constructs they cannot visualize in their minds. Successful data visualization is a technique that demands a careful balancing of function and form. A minimalist graph could decrease the interest of the viewer; visualization, too overloading, could fail at conveying the necessary content as the human short-term memory can remember between three and seven items simultaneously [214].

Thus, data and visualization have to synergize, which is the art of data visualization. To evaluate the effectiveness of a data visualization, measures such as efficiency, accuracy, and memorability have been defined [328, 38].

According to Keim et al. [149], the main advantage of visual data exploration in comparison to statistical data mining algorithms or machine learning is, for one, the ability of visual data exploration to deal with heterogeneous or noisy datasets [237]. For another, he names the advantage that visual data exploration can be applied more intuitively and thus requires less knowledge of the complex mathematical or statistical background.

4.2.1 Ever Growing Data

After describing the needs and advantages of data visualization compared to statistical data mining, the question arises of whether and how data, in its various forms, should be presented in virtual reality. Data surrounds and accompanies us nearly every second. The data growth rate has expanded incredibly within the last decade due to several factors, such as the Internet of Things and sensors in every domain. Systems and scanners evolve, collecting not only more data but much more detailed data in

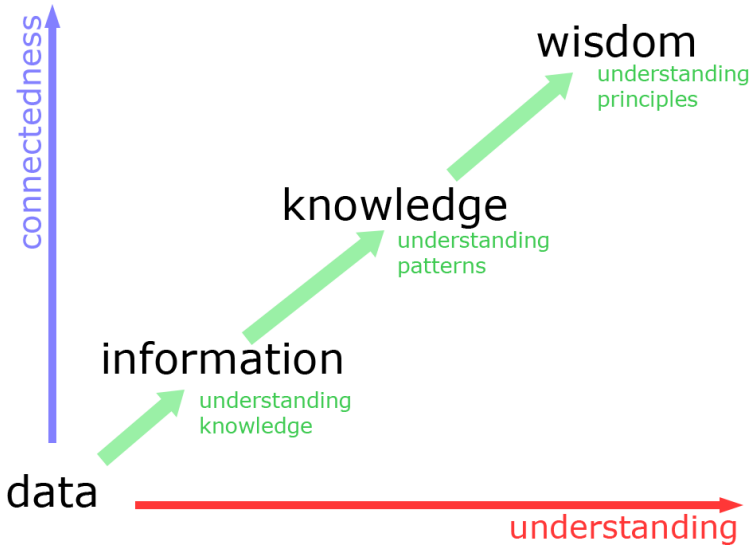


Fig. 4.1 This Figure represents the D-I-K-W model based on Bellinger et al. [25]

every domain. A term that often comes up regarding the amount of data that has to be tackled is Big Data. As the size of data increases, it becomes more relevant to provide intuitive visualization systems that support scientists and the broad public to understand the underlying information. The upcoming sections will highlight models and frameworks underlining the need for visualization systems. However, before going into detail about the evolution of data and its visualization, it is essential to clarify the term 'data' in more detail.

In many disciplines, the term 'data' is frequently used to mean 'information' or 'knowledge,' such as in psychology or medical sciences but foremostly and most relevantly in computer science and engineering. Since the term 'data' is used in many disciplines but not always in the same context or meaning, the term is not implemented consistently and sometimes is used in a conflicting way. Often, data and information refer roughly to the same purpose, e.g., 'data processing' and 'information processing. Cleve et al. [67] describe data as an accumulation of symbols with its own syntax and divide data into unstructured, semi-structured, and structured data, which can be seen as the stations of the data-mining process as well as the type of their structure when being collected. Knowledge is described as the result of the connection between the information and the ability to extract its meaning. Chen et al. [60] took this approach and proposed a more advanced look at these three terms. They discussed that for data visualization, especially the term data, information, and knowledge should be interpreted differently depending on the given context. When talking about data in a digital context, they define it as a

"computerized representation of models and attributes of real or simulated entities", whereas in a cognitive space and according to Ackoff et al. [4], it is seen as a symbol.

Chen et al. [60] highlight the influence of the computational process when defining information. Bellinger et al. [25] extend these three terms to 'wisdom', which means the ability to extract new knowledge from previously held knowledge. They stress the step of "synthesizing" new knowledge. They describe wisdom as systematic and summarize it as evaluated understanding. Figure 4.1 shows the transitions from data to information, to knowledge, and lastly to wisdom (D-I-K-W). Bellinger et al. [25] extended the original D-I-K model from Ackoff et al. [4] to stress that the process of understanding is crucial in each step but not an independent step of the process. The connectedness here describes how humans have to synthesize new data from the data already held, as mentioned before. As VR has a remarkable influence not only on recognizing and discovering but also in understanding patterns, which is the central and omnipresent aspect of a user's way to knowledge and wisdom, the D-I-K-W Model is of significant relevance when designing VR applications for data visualization and understanding.

Various definitions exist of the term 'Big Data', which has led to people still wondering what Big Data actually is. For example, Dumbill et al. [97] defined it as data that outperforms the processing capacity of traditional database systems. He describes data as too extensive, dynamic, or out of existing conventions and restrictions of database architectures. A large amount of data originates from social networks, geological sensors, audio streams, the financial market, medical systems, etc. As shown in Figure 4.2, the first step of Big Data visualization is the acquisition. Data from one dataset can originate from various sources and thus has to be structured in the next step by parsing and reduced by filtering mechanisms. In the last step, the refinement, structured data can be presented in graphs, enabling the user to interpret the data and extract hidden information [59]. Dumbill et al. [97] also state that techniques have to be adjusted to process data. Dwyer et al. [98] agree, saying that the amount and complexity of the data available to us exceed the human ability to process and use data in decision-making without technological help. Simultaneously, human pattern recognition skills remain indispensable and often exceed the capacity of mining algorithms [73].

The so-called 3Vs (velocity, volume, and variety) describe significant aspects of Big Data and are the three elements that differ from 'normal' data. As the volume of data and its importance for information technology, business, and data science increases, scientists are also concerned with a fourth V: veracity. Banik et al. [21]

describe the four terms as follows: Data volume is the amount of data available to the user and does not necessarily mean the physical ownership of the data but the access to it. As the quantity of data increases, the value of the individual datasets reduces depending, for example, on type, richness, and actuality. Data variety describes the richness of the data, e.g., images and text. Challenges Banik and Bandyopadhyay mention concerning data variety and their analysis are possible incompatible formats, non-aligned data structures, and inconsistent data semantics.

The speed at which the data is created, streamed, or aggregated is called data velocity. The enormous increase in velocity may cause so-called ingest issues, which refer to obtaining and importing data for immediate use or storage in a database. Lastly, the term data veracity can be defined as the combination of the noise, biases, and abnormality in data that describe the data's quality. Veracity can be regarded by its origin and its content. While the quality of the source is related to its trustworthiness, the quality of the data itself is a significant factor in the quality of the data analysis.

According to Chawla et al. [59], the biggest challenge of data visualization is how we handle and display today's large data volume. Traditional visualization tools might fail at this challenge because they were not designed for such large datasets. Thus, the dimensions that display these datasets must be chosen carefully. Data visualization also faces limitations, such as perceptual scalability, real-time scalability, and interactive scalability [6]. Perceptual scalability means the ability of human eyes to extract relevant information from a large amount of data. At the same time, screens are limited in their ability to display information, e.g., because of their limited size or pixels. Real-time scalability is another limitation of traditional data visualization, which describes the issue that most visualization systems are not capable or just not designed to handle and significantly interact in real time. This limitation might include connectivity, storage, or data processing capabilities. In particular, scaling operations with terabytes of data can cause system freezes. This problem occurs in many systems, regardless of whether they are VR systems.

4.2.2 Benefits of Data Visualization

The visualization of data in its various forms within immersive virtual environments achievable by modern VR hardware is an interdisciplinary field that consists of aspects from data visualization, VR, computer graphics, human-computer interaction, and visual analytics. Especially the field of visual analytics is essential to data visu-

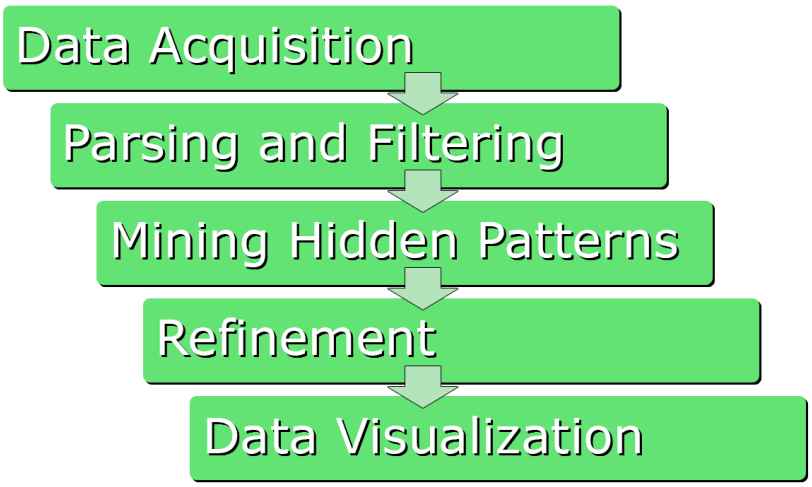


Fig. 4.2 Data visualization process pipeline based on Chawla et al. [59]. The acquired data undergoes several stages until it is finally displayed in the form of a visualization. This concept is similar to the Visualization Pipeline referenced in Section 1.2

alization in virtual reality. The main goal of visual analytics is to minimize barriers between people and the data they are analyzing by supporting data understanding and decision-making [98]. In general, the term "visualization" describes the process of revealing hidden information and patterns to the eye, following the subsequent output of that process. The terms' data visualization' and 'visualization' will be used interchangeably following the approach of Grainger et al. [114] to include all visual representations of data that aim to make information accessible and understandable via presentation, exploration, or analysis. Cairo et al. [54] state that the aesthetic appeal of the visualization is a subordinated goal. Visualizations should be considered engineering tools, serving their primary objective rather than the artwork. Keim et al. [150] expressed the general aim of visual analytics in finding a way to make use of the information overload introduced due to every growing size of data sets. He emphasized that data visualization is more than just a final representation of a data set. It aims to process data and make it transparent for an analytic discourse. As these processes become accessible and understandable, users are not just left with the results of the data analysis but are enabled to review their results critically.

Before going into detail about visualization methods, multiple terms must be distinguished as they are related and sometimes used interchangeably. These terms include visual data mining, visual data analytics, information visualization, and visual analytics.

Simoff et al. [262] summarize the definitions of visual data mining as the process of interaction and analytical reasoning with the visualized data. They emphasize that the process leads to discovering patterns in the given dataset, which may originate from unstructured abstract data or the output from data mining algorithms. In comparison, Thomas et al. [286] describe visual analytics as a science that focuses on analytical reasoning facilitated by interactive graphical interfaces. Keim et al. [147] extend this definition, stating that visual analytics combines computational analysis mechanisms and interactive visualizations to support understanding, reasoning, and decision-making. Cliquet et al. [68] point out that visual analytics, compared to visual data mining, focuses on human analysts. In contrast, visual data mining is more concerned with computational processes.

According to Mazza et al. [198], observing visualized content is a cognitive activity connected with constructing mental models. Mental models are internal representations of the user's environment and help them understand the data. These mental representations of the data emerge from preattentive processing, which enables the perception of visual stimuli into the sensory memory without focused attention. Larkin et al. [162] described in their paper "Why a diagram (sometimes) is worth ten thousand words", that solving physics problems with diagrams in comparison to equivalent text descriptions leads to better results. They identified three properties that cause this effect: Locality, minimizing labeling, and perceptual enhancement. The first factor, locality, describes that each element in a visual representation owns a physical space. If two different parts in the visualization have to be processed simultaneously, they should be represented in two ways but still positioned close to each other. Secondly, Larkin and Simon list minimizing labeling, which refers to the human ability to recognize information without detailed textual information. Symbolic representations are often immediately understood, while textual information must be processed actively. The last factor that influenced the physicists' increased effectiveness is perceptual enhancements. That humans can process a large amount of perceptual interference via visual representations enables them to identify relationships and dependencies between data.

Simoff et al. [262] present a detailed model of not just the data visualization process but also the resulting analytical reasoning and concentrate on the role of sense-making, as shown in Figure 4.3. They stress that various visualization techniques contribute to the success of the visual data-mining process and its interaction. The success relies not only on the number of visualization techniques but also their design consistency, the ability to display the data attributes onto visualizations in an

interactive way, and the contributions of these functionalities to support the reasoning process. As seen in Figure 4.3, Simoff et al. emphasize the central function of the interaction with the visualization and refer here to Keim et al.'s [149] classification of interaction or distortion techniques depending on the data type to be visualized. In Simoff et al., the visual processing pipeline is central as the human is involved in each step as the visual analyst. Data mining algorithms may support the user in these steps (a) before any visualization technique is applied and (b) after each interaction with the data.

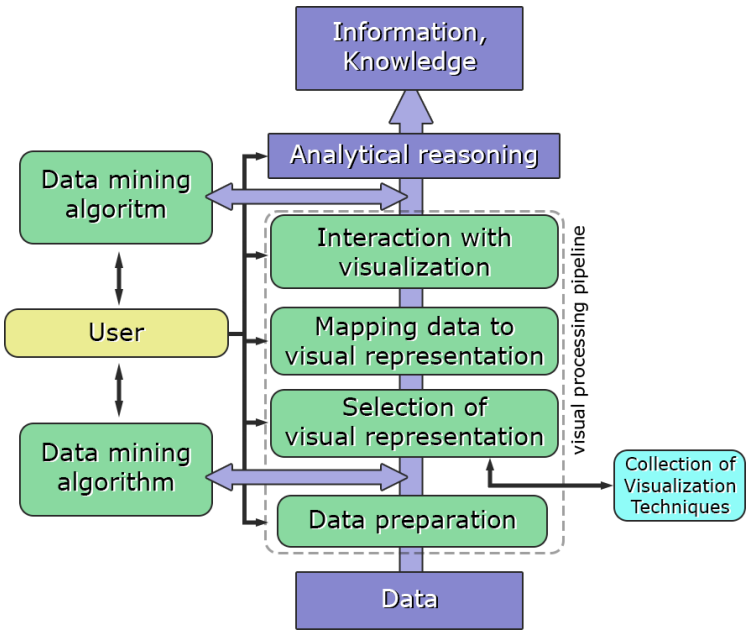


Fig. 4.3 Figure based on Simoff et al. [262]. Visual data mining is often understood as a human-centered interactive analytical and discovery process. Similar to the Visualization Pipeline (Figure 1.1), raw data has to be processed and broad into a structure that can be visualized before any knowledge can be gained.

4.2.3 Traditional Data Visualization

The discussion of whether data visualization is more efficient in 2D or 3D has been interesting for many years. For instance, graph-orientated analysts prefer 2D visualization due to a perceptual and navigational conflict when using 2D screens and 2D input devices while interacting with 3D objects [131]. As possibilities for visual computing grew, scientific visualization researchers quickly adapted to new technologies. Moreover, hybrid and VR systems reached the field of data visualizations since 3D data visualizations had proven themselves in certain domains

such as geospatial data [178, 179, 138, 49], weather data [14] or urban-planning [87, 301]. In contrast to these spatially related data visualizations, information visualization researchers tended to be cautious about using 3D representations for abstract data. Subsequently, many analysts, researchers, and scientists focused on 2D data visualization displayed on desktop computers.

Nevertheless, research started to explore the benefits and opportunities of touch interfaces [95], and tangible graphics [141]. These explorative approaches aligned with the "unbridled enthusiasm" [208] of visualization researchers between 1985 and 1995 for 3D representations [292]. At nearly the same time, the first 3D visual workstations appeared and became more popular and accessible to the mass market, e.g., by Silicon Graphics [9]. Many data visualization researchers agreed that 3D visualizations with their linear perspective, shading, and shadow benefits could compete with or even out-compete conservative 2D representations on 2D displays [193].

Rosenbaum et al. [247] also discuss that position is probably the most frequently used attribute to map data because human visual perception relies on location and distance to identify relations between objects. Nevertheless, with larger and more complex data sets, especially in their dimensionality, the limits of 2D representations come to light. As mentioned before, data is becoming more complex, extensive, and high-dimensional, requiring the shift to 3D visualizations. The additional spatial dimension for value encoding in 3D visualizations has benefits with which 2D representations cannot compete. Furthermore, 3D visualizations shift the viewing process from a cognitive task to a perceptional task, speeding up the perception process [157]. Reda et al. [237] state that when extensive data sets are represented and layered into 3D instead of a 2D plane, the visual clutter can also be reduced, which may improve the interpretation of the data.

Various visualization techniques can be used to visualize data. Beyond standard 2D/3D techniques, these visualization techniques include x-y(-z) plots, bar charts, line graphs, and more. The choice of data visualization is based on the kind of data given, which can be divided by its dimensionality. For example, 1D data, such as temporal data; 2D data, such as geographical maps; or 3D data, such as medical imaging data. Depending on the underlying data, the type of visualization required can vary heavily. Data resolving on imaging information can need significantly different visualization techniques than empirical or statistical data. Relational tables extend the better-known pivot-table interface by using a more prosperous, expressive set of graphical displays and interpreting visual specifications as a precise sequence

of relational database operations. Other kinds of data are text and hypertext, which can be visualized in, e.g., a ThemeRiver [125], hierarchies and graphs, algorithms, and software. ThemeRiver was developed to study the value of the river metaphor for detecting patterns, relationships, and trends in the themes of a document collection.

Keim et al. [149] classify visualization techniques into (a) standard 2D/3D displays, such as bar charts and x-y plots, and (b) geometrically transformed displays, e.g.landscapes and parallel coordinates [129], which are a widely used visualization technique for multivariate data and high-dimensional geometry. Parallel coordinates have became a popular visualization for exploratory data analysis [314] and visual multi-dimensional geometry [139]. Furthermore, Keim lists icon-based displays, which range from visualizations with individual icons to forming texture and color patterns via an overlay of countless icons. Chen et al. [61] find that using various icons in one position may be an effective and efficient technique to visualize large high-dimensional datasets. Lastly, Keim names (c) dense pixel displays and (d) stacked displays. Dense pixel techniques are developed to represent each dimension value on a colored pixel and to cluster the pixels associated with one dimension into adjoining areas [148], and they have been used for Data Mining of mobile sensors[245].

Based upon the work of Wang et al. [305], Ali et al. [10] present another classification of visualization techniques not based on the type of data but its the field of research. Table 4.1 based on Wang et al. [305] shows some of the most popular visualization techniques and when or when not to use them. In the first place, they name treemaps. Treemaps are space-constrained visualizations of a hierarchical structure that use enclosures to represent trees and subtrees in different sizes and colors, building up sequenced nested rectangular areas. In treemaps, nodes are visualized as rectangles; if a node has a child, the rectangle is further divided. Long et al. [185] compares different kinds of treemaps and found that 3D treemaps have potential. They are especially beneficial because of their high space efficiency and scalability. Secondly, Ali et al. name circle packing, which is similar to treemaps, but instead of rectangles, circles are represented, which is not considered as efficient as treemaps. Following circle packing, sunburst visualizations are used for hierarchies but in a circular arrangement with different layers on the visualizations, representing the hierarchy's levels. As next, Ali et al. named, similar to Keim Parallel, coordinates, which are suitable for most dataset attributes as seen in Table 4.1. Ali et al. consider stream graphs such as the ThemeRiver appropriate for large datasets but not for dynamic or heterogeneous data sets. Lastly, circular network diagrams fit

Method Name	Large Data Volume	Data Variety	Data Dynamics
Treemap	Y	N	N
Circle packing	Y	N	N
Sunburst	Y	N	Y
Parallel coordinates	Y	Y	Y
Stream graph	Y	N	Y
Circular network diagram	Y	Y	N

Table 4.1 Attributes of visualization techniques by Wang et al. [305]

best for relative data because various entities are located in a circle and linked based on their relativeness. Circle network diagrams have, e.g., been used for visualizing social networks [42].

So far, various kinds of data and their fitted type of visualization have been introduced. For abstract data, another visualization type is required [151, 233, 247]. The data for this kind of visualization often consists of numerous dimensions and cannot be mapped naturally and efficiently as standard visualization techniques, e.g., (x, y) plots, line- and bar-charts, are limited in this regard. Therefore, Keim [147] suggests that more modern visualization techniques are suitable, such as parallel coordinates [129], treemaps [183], and glyph-based [61, 298] data representations.

To summarize, there are diverse visualizations, just as different data types exist. Other data analysis methods, approaches, and visualizations depend on the data distribution's spatial dimensions, geometry, and topology. A thoughtless application of algorithms and visualization tools may lead to false results and assumptions. Furthermore, the review of visualization techniques to this point revealed that many traditional 2D techniques could be modified to work in 3D models, which can be efficient when the given task deals with the spatial structure of the dataset.

4.2.4 Towards Immersive Data Visualization

As mentioned in Sections 2.4.1 and 2.1, it was in the late 1960s [281] when the first HMDs were researched and the late 1990s when the first fully programmable 3D accelerators entered the marked, enabling real-time rendering and processing of large data sets. This development was followed by the first IEEE VR conference series and their initial workshops on 3D user interfaces in 2006 [98, 1]; the possibilities seemed endless. Despite technological progress, researchers were stuck on conventional visualization techniques and improved classical WIMP (Windows-Icons-Menus-

Pointers) interfaces rather than proceeding toward immersive visualizations. Until 1994, there were many pilot systems, but except for simulators and gaming or immersive movie applications, VR was not yet found in any other fields [45].

The first ambitious research on VR started with cone trees and the horizontal variant cam trees from Robertson et al. [244], which were developed to visualize hierarchical data. As CAVEs are immersive systems that were used earlier than HMDs, the first VR applications have been applied in CAVEs, for instance, in [282, 207]. A significant milestone in VR history was achieved [76, 75] when they introduced the first scientific collaboration space in a CAVE. The goal was to visualize and interact with presented research data in various areas such as living spaces, the universe, galaxies, and brain structure. With the rise of cheaper VR technologies, such as stereoscopic displays [322], a foundation was laid for implementing more accessible VR-based visualizations that took into account the associated advantages.

In his work, Cliquet et al. [68] already mentioned the research of Azzag et al. [17], and Nagel et al. [209] who extended existing techniques such as scatterplots to be supported in and 3D virtual environment. Since these early works, several adaptations of already established visualization methods have been carried over to VR, such as the work of Wijayasekara et al. [319], or Ferey et al. [104] visualizing maps within a 3D CAVE setup. Several researchers have shown that various benefits of VR support people in their task to understand the data better [8, 64, 307, 315]. They demonstrated that, with suitable depth cues, the 3D perception could improve the comprehensibility of a dataset, make it possible to differentiate or identify complex abstract representations and models, and, lastly, make it easier to move objects mentally. It was finally the introduction of good quality and affordable HMDs, such as Oculus Rift™ and HTC Vive, that led to the new era of VR. The following sections will outline how modern CAVEs and HMD systems may be utilized to visualize collected data beneficial for the user's knowledge finding.

4.3 Desktop versus VR for Data Visualization

Over the upcoming sections, various use cases of data visualization are investigated. The structure of the upcoming section splits the topic into three parts. First, visualization tools are presented, which are tailored toward traditional workstation setups utilizing 2D desktop monitors but intend to visualize higher dimensional data by projecting it to the lower dimensional output domain. Afterward, state-of-the-art VR systems for data exploration and knowledge gain are presented. As the interac-

tion with various forms of data in stereoscopic 3D environments requires different interaction metaphors than the widely adopted WIMP concept, the importance of real-world metaphors for interaction is outlined. In the following approaches for desktop and VR data visualization are compared, and the impact of VR on the field of data visualization is analyzed.

4.3.1 Desktop Data Visualization of 3D Data

Due to their ease of access and inexpensiveness, desktop computers and large 2D displays have been the gold standard for data visualization for most analysts and scientists. Depending on the research domain, data visualization on desktop workstations usually does not necessarily require high-performance hardware compared to typical VR systems. While some fields of scientific visualization can demand an efficient and powerful workstation for the underlying algorithms to be computed, most of the time, commodity-available workstations are sufficient for most desktop data visualization scenarios. Therefore, desktop workstations appear more attractive for most use cases than costly VR systems, especially CAVE setups. Nevertheless, the discussion of whether 3D data visualizations are effectively displayed on 2D screens is still ongoing. Many visualizations still rely on traditional widgets, such as 2D bar and pie charts, or the slicing of 3D medical imaging data into 2D image planes, which are not suited for spatially related data, according to Tufte et al. [292].

Furthermore, data analysts and scientists often prefer mouse and keyboard systems since most visualization systems still rely on complex WIMP interfaces that require precise interaction. Nowadays, desktop data visualizations are ubiquitous and appear in different technical domains, for instance, information visualization, scientific visualization, and geographic visualization, and functional domains such as statistical graphics, information graphics, and data journalism [34]. Over the last few decades, 3D visualizations worked their way into our everyday life, for instance, interactive 3D globe representations such as Google Earth in the 2000s. They have also proven themselves useful for geo-spatial visualizations [36] or software visualization [284]. Bleisch et al. [36] discussed the possibilities of 3D visualizations on flat displays in their paper about connected 2D and 3D visualizations. Their study showed that combining 3D visualization of datasets with 2D displays offers the possibility of using 3D only when this specific way of representation is considered advantageous or effective, supporting the user to gain new insights or draw conclusions. Moreover, with only one dataset, researchers can decide whether data should be displayed in

2D or 3D representations so that they do not have to rely on one model for data exploration.

Teyseyre et al. [284] concluded that 3D visualizations on flat screens aim to create visualizations that try to get as close as they can to real-world metaphors or improve efficiency in terms of space usage by adding an extra dimension. This kind of visualization allows users to rotate and relocate the 3D objects within a 3D space. Tufte et al. [292] state that when the third dimension is not necessarily needed, a 3D presentation on a 2D display should not be preferred to avoid perceptual overload. In agreement, Parker et al. [222] found the most significant disadvantage of 3D visualizations on a 2D display to be the intrinsic 3D navigation problem. It is problematic to navigate within 3D spaces or with 3D objects when a 2D input device like a mouse or keyboard is used [284], and therefore, users may get disoriented [133]. For example, tasks performed on 3D objects within a 3D space, such as manipulating the position and orientation of an object, involve six degrees of freedom. On setups using 2D displays, most input devices, such as a mouse, only provide two degrees of freedom that should be mapped to a 3D environment.

In addition, the need for methods that support the user to navigate in a 3D space arose because of the increased movement and interaction possibilities enabled by more degrees of freedom and large scalings in 3D spaces [7]. Teyseryre and Campo propose that if interfaces overcome the WIMP-dogma and specialized hardware, such as haptic displays, gaze tracking, and motion tracking, 3D techniques would be established in 2D screen applications. 3D software visualization could exploit its potential in non-immersive environments [89].

4.3.2 State-of-the-Art Immersive Data Visualization

A brief overview of the latest developments and achievements in VR data visualization is presented in the following. In 2014, Andersen et al. [14] analyzed how using a VR environment might help to visualize large weather data sets in their comparative study. For interaction, they switched from traditional controllers to motion-tracking hardware in the form of the Leap Motion. They used a rendering method of transparent-colored polygonal meshes for a better perception of individual weather layers. The mesh structure was extracted from a three-dimensional scalar volume since direct visualization of the volumetric data was too expensive for the time being. The extracted data represented a height map of the scalar field.

A novel approach for the use of VR was proposed by Donalek et al. [92], who focused on collaboration in virtual worlds while analyzing data. They experimented with different approaches to multi-dimensional data representation by varying several coordinates, colors, sizes, transparencies, and shapes of data points. Additionally, they tested different textures, orientations, rotations, and more to find the best number of data dimensions. Also, they added new features for a collaborative, multi-user visual data exploration to their tool iViz [92], proving another potential of VR. The iViz application can be used with the Oculus Rift™ and works with Leap Motion for natural interaction without needing physical controllers. Since the environment is not necessarily large-scale, users can remain seated.

Nevertheless, as Andersen et al. [14] pointed out, holding their hands up while gesturing may be tiring for users. The MICA-Experiment (Meta-Institute for Computational Astrophysics) [91] from 2012, aimed to explore the possibilities of collaborative work around data visualization in a virtual world. In the virtual world, lectures were held online, and data visualization tools were implemented, as described by Farr et al. [102], and Nakasone et al. [210]. Although the technology is not the latest in Farr and Nakasone's collaborative AstroPhysics virtual world, both showed promising results, which built the foundation for further development of the platform by Djorgovski et al. The results of their systems demonstrated an increased interest in students and novice users to data exploration in teaching lectures.

All three studies used variations of OpenSim, enabling the user to be embodied by an avatar, which made communication more natural. Farr et al. added a gravitational N-body simulator to the virtual world, focusing on scientific visualization. In contrast, Nakasone et al. found the capabilities of their tool to be efficient in wayfinding through landmarks, advanced data selection in 3D space, and natural communication between users.

Another novel approach that differs from the types of VR presented before was that proposed by Chen et al. [62]. Unlike visualization, which immerses the human in the data, Chen et al. intended to let the user become data points. They called this approach "Be the Data". The approach is based on the benefits of the embodiment of an avatar and might engage users in the usual unexciting data exploration. Be the Data differs from Be In the Data by the perspective from which the user can view the data. Multiple users assemble the data representation by representing the data points in one virtual environment.

BrunhartLupo et al. [47] focused on the potential of parallel planes in VR. They adopted the concept of Parallel Planes into a fully interactive, immersive environment.

They, too, followed the idea that the purpose of the interface should be to explore the existing data further and enable the user to examine the system's parameter spaces interactively, creating new parallels within the immersive environment. Users can select or brush the planes' areas and get a more detailed insight into the highlighted observations. Unfortunately, they could not conduct a study to test their approach to parallel planes.

Nevertheless, with yet another focus, Millais et al. [204] used Parallel Planes for their comparative study of 2D screens and stereoscopic VR systems. They surveyed to compare 2D data visualizations with two approaches in multi-dimensional visualizations in VR. The leading questions of their study were whether the type and system of visualization influence users' approach to data exploration, the extent of insight they gain, and what role user experience and depth of understanding play. They first worked with a 3D scatter plot, allowing participants to observe the data from various perspectives. They used the approach Be the Data by Chen et al. [62] to motivate users to engage, explore, and interact with the data as part of the dataset and move as data points within the visualization. Finally, they allowed users to explore more factors for the provided data, including mood, productivity, sleep, music listening, and physical activity. It must be mentioned that Millais et al. let their participants get used to the VR visualization tools in different training stages to prevent differences in performance due to missing experience and comparison to the desktop setup. Nonetheless, there were no significant differences in the measured data explorations workload and performance between 2D and immersive visualizations.

A more generic study was conducted by Raja et al. [233]. Even though the study may not be the most recent, it extended the boundaries of VR application methods. They tested different abstract datasets in three display conditions and used a VR tool that could easily be modified to the different kinds of display technologies. Thus, this study stresses that VR might not be limited by its data type but by technological limitations due to budget or hardware. DiBenigno et al. [87] adapted this concept in 2021 with their newly developed data storytelling platform "Flow Immersive". Their tool encourages experts and non-expert users to comprehend complex health data. Above all, the visualization tool showed promising results concerning engagement and understanding, especially when the user had a high perception of immersion. According to DiBeningo, this may be caused by the effect that the visualization becomes more immersive as the time spent in the virtual world and, subsequently, engagement increases.

DiBenigno et al. were inspired by the findings and the novel approach of Cicalo et al. [65] who found a symbiosis of health data with environmental data to help understand environmental-health linkages but did not expand the approach to immersive visualization. DiBenigno et al. aimed to enhance 3D data storytelling and generate an extensible, generalizable visualization interface for health data that is not limited to specific locations and is accessible to multiple users through multiple devices and platforms. A similar approach has recently been followed by Sharma et al. [257]. Based on the approach of Brundhart-Lopus' Parallel Planes [47], they also explored the use of VR for Covid-19 data. Instead of focusing on the platforms, they aimed to identify how objects in the virtual environment can be depicted by following an analyst's mental model and enhancing their situational awareness. The exploratory study showed that the users better understood the data, leading to more human-centric situational awareness insights using an HMD instead of a mobile device. A desktop application was not compared.

In terms of access, Butcher et al. [52] developed a prototype of a web-based VR platform for data visualization, which lets users view the data within CAVEs, HMDs, and mobile devices. Like Raja et al. [233] and DiBenigno et al. [87], they wanted to empower a broader audience to interact with data but also aim to engage developers to build new tools and interfaces for data visualization.

To conclude this section, VR has shifted from an application mainly focused on scientists and data analysts to an application domain that should inspire novice users to explore and collaborate within immersive virtual environments. New data visualization approaches, such as Be the Data or Parallel Planes, brought new perspectives. New tools have driven accessibility to a point where non-professionals may easily engage with VR tools.

4.3.3 Impact of VR to Data Visualization

As established in Section refsubsec:VRhardware, modern VR systems enable users to explore virtual environments in an immersive and complex way, leading to the feeling of presence and engagement often not achievable on traditional desktop displays. The leading question for this section is, how do modern VR systems, presented by HMDs and CAVES, influence the data analysis process?

Mania et al. [191] focused their research on the topic of memory in immersive and non-immersive spaces. In their study, participants were either in an actual 15-minute seminar or in one of two VR simulations of a university lecture displayed on

a traditional desktop monitor or an HMD. They discovered that immersion achieved by HMDs is correlated significantly with better recall for easy memory tasks. It is deducted that while analyzing data in VR, users can improve their performance by recalling more information during decision-making processes.

The results of Rosenbaum et al. [247] align with these findings and also demonstrated that immersion might be a factor contributing to the success of VR. In their study, they treated the individual position of the user as a data point within the data space to increase their immersion during the exploration of the surrounding data. They referred to the relevance of depth cues and immersion based on Raja et al.'s [233] work for the effectiveness of abstract VR. Cho et al. [64] found out that depth cues, mainly 3D visual perception, may be correlated with the effectiveness of the depth perception, which then may enhance the recognition and perception of 3D visualizations. The depth cues that have been studied the most are ocular (accommodation and vergence), pictorial (shading, size, and occlusion), dynamic (motion cues), and stereopsis (binocular disparity). Raja et al. proposed that a high degree of physical immersion might allow an enhanced perception of dimensionality, higher levels of task performance, and greater user satisfaction when visualizing 3D data sets. Participants had to find data points according to color or value in data sets and determine trends and clusters. They found that a higher degree of physical immersion results in faster task completion.

According to Witmer et al. [321], the performance of analysts in a virtual environment may be correlated to the amount of presence perceived. Additionally, Busch et al. [51] showed in a comparative study in which they examined the impact of a virtual versus a natural environment on usability and learnability that presence also affects user experience, which may have an impact on user satisfaction.

In contrast to Andersen's [14] results, Cooper et al.'s [72] research indicated that a user's subjective experience might stand in relation to the overall task performance. Their study conducted that users felt more presence during VR interaction and showed a faster completion time. These results may be suspected based on the nature of the tasks in Cooper et al.'s study since participants were in a mechanical repairing simulation.

In terms of immersion, Chen et al. [62] showed with their Be the Data approach that immersion enhances a user's engagement and thus might be a valuable tool for data novices. This type of visualization was based on "Andromeda" published by Self et al. [254], which utilized 2D displays. According to Chen et al., this kind of display has limitations since immersion is an essential factor contributing to user

engagement. Hence, this new approach of VR might be beneficial for novices of data analytics or non-professionals which will be inspired and encouraged to learn more about a particular dataset or data visualization but could limit experts during data exploration.

Engagement is considered a factor as well. For example, Millais et al. [204] suspect engagement as an influence on their result, that participants reported a similar number of insights in all conditions, HMD, CAVE, and desktop but showed fewer inaccurate insights in a VR environment. They assume that participants might be more attentive and meticulous during their analysis task in a VR environment because they are more satisfied and engaged by the technology. Another factor they suggest is the novelty of the visualization technique for some users, which made them more curious and engaged.

Van Dam et al. [297] analyzed the role of depth perception in spatial apprehension of the data set. According to their paper, the immersive context may help the user perceive the data with kinesthetic depth, leading to a better understanding of 3D structures and spatial relationships. When the virtual environment fits the data visualization in its 3D attributes, such as size, distance, and angle judgments, it is easier and more natural for the user to utilize these attributes for the analysis. They describe this shift from looking at a 2D image of a 3D environment on a conventional 2D desktop to 'being in' the dataset. Thus, spatial judgments are made based on the user's body movement and depth perception. Mazza et al. [198] agree that data attributes are encoded in VR by preattentive attributes such as color, form, spatial position, and movement. Additionally, the stereoscopic depth provided by binocular vision can be used to perceive depth in visualizations.

Bayyari et al. [23] presented their research to discuss the impact of situational awareness on understanding data visualization. Participants were either in an experimental condition with a monitor or a one, two, or three-walled CAVE and counted objects in a virtual landscape by their textural characteristics. The results indicate that immersive displays lead to better results in a counting task that depends on the resolution and perceptional factors. They conclude that immersive visualizations outperform 2D displays because immersive environments support better visualization comprehension. Pausch et al. [223], and Robertson et al. [243] conducted similar research and found that users performed better using HMDs than standard desktop interfaces. Pausch et al. found that VR can help users remember where they have and have not looked because VR users build a better mental frame of reference for the space and avoid redundant searching.

A factor not addressed so far is the embodiment, which is meant as a participative status that describes the level of perception that the technology is an extension or part of oneself [94, 169]. Ball et al. [19] reported that these attributes are especially applicable when the data visualization is represented on large-scale displays and promotes physical involvement such as walking and grabbing.

4.4 Application fields for Immersrive Visualization

There can be no generic answer about which application field, immersive or non-immersive visualization, is more appropriate or helpful as the visualizations often are as unique as the domain in which it is applied. Nevertheless, there seem to be different tendencies between information and scientific visualization and spatial versus non-spatial data. Whereas scientific visualizations usually examine vast amounts of scientific data from sensors, simulations, and laboratory tests, information visualization is defined more generally as "the communication of abstract data relevant in terms of action" [150]. Research about visualization in immersive environments mainly focuses on spatial data [138]. Abstract data is used less frequently for VR since the apparent relationship between a 3D environment and the data is missing, which is often considered the main reason for its application. As the benefits of spatial-related data are easier to understand, two representative examples proving that VR may be more suitable than desktop displays will be given.

Andersen et al. [14] developed and tested a weather visualization tool and compared the performances of users in three different display/interface combinations. The meteorological data consisted of four scalar fields: terrain, height, temperature, rainfall, and humidity. Polygonal meshes were used as the primary tool in their visualization. They enabled the presentation of the data in a spatial context while the user could see different layers of meshes due to transparencies. This approach combines the benefits of the spatial relation of the data and the simultaneous display of diverse weather data. Their study did not prove significant differences in accuracy and effectiveness. However, when participants were asked to rate the level of comprehension they acquired while using the data visualization tools, VR tools achieved better results than desktop tools. A similar approach was presented earlier by Helbig et al. [130].

Broucke et al. [46] approached the city metaphor, which displays data as buildings on a map in an obvious way by using it within VR for real-time heterogeneous Smart City data. Their study showed significantly lower levels of perceived frus-

trations and higher levels of perceived data intuition for users who used the VR application and not a web-based desktop system. In terms of perceived workloads, no differences were measured.

Compared to the immersive visualization of spatial data, often referred to as scientific data, not much is known about data visualization of abstract data in virtual environments. The basic idea of the approach proposed by Rosenbaum et al. [247] is the exceptional support of location changes and spatial understanding in immersive environments. This additional aid allows users to effortlessly position and relates to the data visualized.

Time and spatially related data have already proven to be beneficially represented in VR. As discussed in section 4.2.1, multi-dimensional complex data sets bring new challenges, which have already been studied [221, 6, 205, 283, 10]. Olshannikova et al. [221] stress that processing methods for 'normal' and less complex data cannot be applied to multi-dimensional complex data, as the data size has to be reduced immensely but also be reasonable and applicable for VR. Brandes et al. [42] explored a novel visualization technique called "visione" to facilitate the exploration of social networks. They attempted to make complicated analysis and data handling techniques more transparent, intuitive, and accessible. In a methodological approach for social sciences, Brandes and Wagner developed a social network analysis consisting of graph-theoretic concepts that aim to describe, understand, explain, and eventually predict or design social structures. Moran et al. [205] introduced such a tool and intended to visualize the information in a realistic 3D setting. Analysts could experience enhanced situational awareness and draw better in-depth conclusions. Their approach was used for geospatial but also abstract Twitter data. Participants in the associated study were able to navigate and explore the data set by moving around and zooming in/out. Users could identify or select data to get a detailed insight and filter or query data to spectate data that matches certain criteria in an isolated view.

Vincur et al. [301] proposed an approach to visualize software based on a 3D city metaphor, resembling the implementation of Broucke et al. [46] to explore object-oriented software systems in an immersive environment. The tool supports the user in identifying interfaces, abstract classes, methods, and code related to the methods used in the software. Both approaches brought a new perspective on the city metaphor and transferred its spatial benefits to abstract and complex data.

A study from 1997 presented by van Teylingen et al. [298] designed an interactive VR tool, 'Virtual Data Visualizer', to visualize and analyze data from computational fluid dynamics simulations and molecular dynamics simulations of materials systems.

Users can transfer data variables to glyph elements organized hierarchically in classes within the virtual environment. This solution resembles Chen et al.'s [61] proposal that high-dimensional data can benefit from using icons in a 3D space. The user can modify the data organization and the mappings to the glyphs from within the virtual environment.

As reviewed in section 4.3.2, VR has proven itself to be suitable in collaborative data visualization [92, 91, 136]. Avatar-based tools such as OpenSim or long-established chat systems showed good communication results and enabled the collaborators to take different perspectives on the data set simultaneously. Also, remote work has achieved new perspectives, simplifying data analysis for multiple users in a co-located space where the users can see the exact virtual location of their co-worker's avatar. Collaborative data visualization tools can be used in many application fields, no matter if the focus is on scientific visualization as in the studies of Djorgovski et al. [91] or on information visualization as described in the paper by Donalek et al. [92]. Their MICA experiment created a new scientific institution, focusing on exploring VR and virtual world technologies for science, scholarship, and education. Both approaches showed that educational use could benefit from co-located data exploration and motivate students to explore data visualization opportunities. Combined with the findings of Rosenbaum et al. [247], VR proved itself to be an all-rounder tool regarding the domain used. Whether the visualization technique is appropriate is a question of visual analytics in general, not of immersion.

Ferey et al. [104] proposed the software solution Genome3DExplorer to explore the vast streams of data in genome analysis. As biologists rely on the knowledge received from data mining processes of the raw data, they need to apply bio-informatic data mining tools, which are often over-strained by the amount of data. Genomic databases are heterogeneous in format and quantity but contain specific patterns such as proteins and genes. Users can compare different genome datasets by numerical mapping measurements between two genomic objects of interest by edge length and using predefined colors.

Ferey et al. [104] pointed out that this kind of representation is major for 2D visualizations because, i.e., the user can directly see several chromosome arms in the center of the global gene duplications, while other genes can be easily identified as peripheral, which is not that obvious on the desktop visualization. They found immersive visualization to be powerful because it enables the exploration of large quantities of data in a synthetic way, giving a global vision of heterogeneous data structure. They emphasize the ability to identify local and global topological char-

acteristics compared to 2D display applications. Other approaches of VR for data mining have been presented as well and yielded good results [17, 209, 319, 315].

4.5 Differences in User Interaction

Interaction possibilities between immersive and desktop-based visualizations primarily differ concerning their interfaces. Desktop-based environments usually use abstract interfaces such as mice and keyboards to control the translation of movements and direct changes. In virtual reality, abstract interfaces may also be used, but in contrast to desktops, VR systems can also use realistic representations of mental models.

Scientific data visualizations and explorations are commonly limited by 2D interaction technology, such as a mouse, keyboard, or touch controls. These interaction possibilities set the boundary, causing data interaction to be restricted within 2D space. In particular, Section 4.2.1 pointed out the most significant problems data visualization has to face. According to [10], the most crucial aspect of a data visualization is its ability to be interactive in real-time. While working with the data, users must be able to interact with the visualization. The interaction option to select subsets of the data or only visualize a particular region has to be provided by supporting methods like hovering, zooming, or selection queries.

The significant differences between interactive desktop visualization and VR systems are listed in Table 4.2 based pm Mulder et al. [207]. They compared interaction with a traditional desktop setup with steering in CAVEs. The aspects of user interaction listed in Table 4.2 consider both viewing and manipulation. The comparing entries indicate that virtual environments may provide more intuitive viewing and manipulation methods. Thus, virtual environments can be expected to enhance user participation.

According to Mulder et al. [207], the most significant difference between the user interaction of both display methods is the viewport and object manipulation. More than just classifying the benefits of VR, Mulder et al. developed viewing and manipulation techniques within a CAVE system based on steering transferred to VR in analytic data applications. In the interface editor implemented in their approach, basic navigation functionalities such as viewing direction, scene rotation, or scene scaling were realized using six DoF wands with three buttons and a joystick. The additional DoF compared to the keyboard or mouse, assumed here as the typical input devices for desktop solutions, allows spatial interactions such as rotating the

Interactive 3D Graphics	Virtual Environments
Screen used as canvas	Displays used as 3D space
Tele-manipulation	Direct manipulation
3D to 2D mapping of user's actions	3D to 3D mapping of user's actions
No multiple simultaneous interactions	Multiple simultaneous interactions
Users always aware of interaction	Interaction is transparent

Table 4.2 Major differences in user interaction and type of display between interactive 3D graphics on desktop versus virtual environments from CruzNeira et al. [75] and further developed by Mulder et al. [207]

datasets or the object and make it easier and more intuitive for the user to manipulate the visualization.

Based on Dimara et al.'s [90] review of data visualization, and the previously mentioned studies, interaction techniques for mapping and presentation can be applied to the ongoing comparison as follows. They conclude that when comparing general human-computer interaction (HCI) and data visualization interaction, the most significant differences are related to entities, focus, intent, and interaction flexibility. The main difference concerning entities means that whereas in HCI, humans and computers interact with each other, in immersive data visualization, the human directly interacts with the data. The goal is for users to forget they are interacting with a system.

In contrast, immersive data visualization tools had to develop novel techniques that were conceptualized from the beginning to facilitate knowledge and consequently support acquiring wisdom or decision-making [285]. Instead of minimizing effort, VR systems aim to support the construction of mental models [198]. Nevertheless, applications that can be considered multi-platform visualization tools, as presented in [87], show that one tool can provide both.

DiBenigno et al. [87] demonstrated with their multi-platform VR tool Flow two clusters of answers that emerged when participants were asked why and how the tools differ from traditional visualizations. The responses were mainly (a) quality of interaction or (b) learning. Flow was designed to enable the user not just to zoom in and out of the data point but to ensure that data points do not appear and disappear when an operator is chosen. The points steadily moved from one perspective to another to allow users to follow a linear storyline. Participants could click on the data point at each step and pull up extensional details to enhance interactivity. Participants experienced flow as a richer interaction due to the storytelling aspects

with personalized experiences. The immersion achieved by the system made them watch the data longer and understand it from a new perspective. Moreover, the influence of the user experience in VR motivated the participants to keep engaging with the dataset.

Research has also explored to what extent and how gesture-controlled interactions in VR may be more efficient, immersive, and intuitive than conventional input devices such as a mouse, keyboard, or tangible interfaces when applied to 3D objects are manipulation [238, 80, 95]. In terms of rotating and translating an object in a 3D space, it was found in the research of Reifinger et al. [238] that rotation is performed at the slowest using a mouse or a keyboard. When the user performed a combination of translation and rotation, gesture-based interaction was measured as the fastest interaction technique. Moreover, the study of Reifinger et al. demonstrated that gestures result in the most immersive and intuitive interaction with the lowest mental workload. Tangible interfaces from the survey of Dangeti et al. [80] have proven to extend and complement existing inputs for 3D modeling. Nevertheless, gesture-based techniques may be more exhaustive for the user, and creating a well-designed gesture set is not trivial [14].

Brushing and marking data points, or even sketching models out of data visualizations, can be approached differently than VR raycasting [204] or hand gestures. Bimber et al. [32] proposed sketch-based, multi-layered interaction on calligraphic interfaces to ensure the naturalness of the interaction and reduce the user's body movement. In contrast to non-immersive desktop solutions for sketching interaction with data, immersive environments offer less constrained sketching, which is caused mainly by the possibility of one- or two-handed 3D interaction, 3D navigation, and stereoscopic visual perception. Bimber et al. acknowledge that sketch-based interaction has not yet reached generic suitability for many application domains and should be employed in specific domains. Overall, their proposal is another step away from WIMP interfaces and thus should be considered a helpful complement to existing data interaction techniques in VR.

Santos et al. [249] conducted an extensive study to compare navigational performance in HMD setups versus desktop setups and focused on usability and interaction. For the desktop condition, a mouse was used to select the moving direction, and two keyboard keys were used for moving forward or backward. In the immersive condition with an HMD, the participants' head movements and the mouse buttons were used to control the walking direction. No controllers were used, which could be perceived as unusual and affect the results. Consequently, their results showed

that participants were generally satisfied with the immersive environment. Most performed better while using the desktop for the task. Since VR systems are primarily displayed in relatively small environments, navigation may not strongly relate to analysis performance. However, the VR experience was seen as intuitive and natural, but natural head and body tracking should be preferred to have navigation as natural as possible.

Lee et al. [169] explored the interaction possibilities of data visualization in general and proposed that, as data becomes more complex, interaction design has to be more natural and consider new approaches. This overview pointed out how interaction techniques have evolved from classic WIMP interfaces towards more cognition-oriented approaches. It shall be proposed that data interaction should be designed with novel and natural interaction techniques. The main criticism of desktop interfaces, primarily for WIMP, is that users are "drowning in functionality" [169] due to the overload of menus, buttons, and the lack of intuitive use.

4.6 Challenges of Immersive Data Visualization

In terms of challenges that VR has to face regarding applications are the selection of appropriate application domains and the right choice of task that can benefit from VR. Based on these addressed issues in Section 4.4, it is now time to discuss additional challenges that VR has to face regarding data visualization.

The most prominent problem that most people directly associate with VR is the possibility of Cybersickness. As outlined in Section 2.5 the research community [191, 271, 164, 163] has discussed this issue mainly in the context of gaming. However, these symptoms can also occur during data visualization. As both domains share the same technology and limitations, Cybersickness can be an essential topic that has to be covered for any application domain of VR. Up to 95% of users who stay within a virtual environment experience a certain level of post-exposure symptoms [271]. While the degree of symptoms can vary dramatically from user to user, avoiding any of these symptoms is desirable. Especially when we consider VR in research, medicine, or industrial use, any form of Cybersickness can lead to the users not being capable of fulfilling their duty for the rest of the day. Since most Cybersickness factors are based on the conflict between the vestibular and visual perceptions and occur after too long exposure to the virtual environment, these differences should be eliminated whenever possible. One solution to this problem could be to design a data visualization system that limits the exposure time, counteracting the benefit of

increased user satisfaction during data exploration. Hence, data visualization should be designed to not have to stay in the system for too long.

Another challenge that VR has to face is related to collaboration and communication. Donalek et al. [92] showed that VR in a collaborative space can be helpful. However, they could never confirm their preliminary results with a comprehensive study. It is yet to be determined whether communication in VR can be conducted naturally. The knowledge acquisition process can be accomplished collaboratively in VR quite successfully, but face-to-face situations allow researchers to use their communicative skills. Most recent studies are exploring the possibilities of mimicking natural conversation [280]. Personal contact, such as person-to-person meetings for discussions and decision-making, should not be neglected as long as communication within VR cannot compete with these natural meetings. Desktop setups face the same issue but cannot benefit from novel findings and developments in digital communication. Thus, VR can be considered more useful for team data analytics tasks. Khadka et al. [153] found communication in multi-user VR tools for data visualization to be a problem as well and developed a tool similar to the one proposed by Donalek et al. [92]. They found in their preliminary study that collaboration within their system can improve task effectiveness and scientific workflow. They propose that the advantages of real-time and synchronous interaction across multiple types of VR systems contribute to their promising results.

Besides communication-related issues or Cybersickness, data visualization in immersive environments is controversial. Today, only a few people have regular access to and knowledge of VR systems. The research presented in [101] predicted that in the US, 15.7% of the population would use VR, which does not imply that these 15.7% have access regularly. The low number of VR users indicates that the users are either experts or novices, which presents VR with the challenge of conceptualizing tools suitable for both extremes or developing various tools adapted to the users' skills. On the other side, desktop-based data visualizations are suitable for both user types and can be adapted more quickly to user needs [54].

As outlined in section 4.2.1, data visualization has to deal with several challenges due to the 3V's of data. Concerning data variety, possible incompatible formats, non-aligned data structures, and inconsistent semantics were listed [21]. Olshannikova et al. [221] see scaling as the biggest problem for immersive data visualization and propose that using other motion detection wearables could improve scaling operations from the interaction-related point of view. Visual representation and level of detail are two aspects that should be handled by the visualization and its

interaction and cannot be approached by navigational capabilities. Suitable tools must intelligently combine visualizations of chosen details of the data representation on the one hand and a global overview on the other hand to accommodate immense streams of data. In desktop or other 2D display applications, scaling can be conducted intuitively by scrolling or zooming on touch interfaces. However, when looking at the hardware aspects, desktop applications have no benefits or disadvantages compared to VR since both visualization systems may rely on the same computer as a database and process unit. Besides veracity, as the name indicates, today's increasing data confronts VR with its large amount to be visualized. Even after applying data processing pipelines as presented in Figure 4.2 and Figure 1.1, the diversity and quantity of data could still be overloaded. Data visualization tools and techniques must scale with the size and dimensionality of the input data and quickly adapt to changes requested by the user.

Users can easily switch between tabs or windows in desktop applications, while immersive solutions have to deal with this issue another way. Moran et al. [205] proved VR to be highly adaptable when confronted with complex Twitter data by integrating filters and dynamic queries in a large-scale environment. Clusters and queries can be simply arranged next to the users, who have unlimited space for arranging filtered data around them. Moreover, the relevant data patterns and relationships can be presented on different levels of detail and with relevant data and visual abstraction levels. Whether these interaction and navigation techniques can handle volume-based challenges and desktop tools or better has to be determined in a separate, comparative study.

Another issue immersive data analytics tools encounter is the display of interfaces and metaphors since, in a virtual environment, WIMP interfaces are more difficult to implement or should at least be reduced to a minimum [179, 39]. VR systems need to be easily used and interacted with by the user to enable a maximized focus on the task at hand, avoiding overly technical, complex, or potentially distracting user interfaces. Additionally, the 'gulf of execution' [215] is an issue that must be mentioned. The gulf of execution is a phenomenon that describes the difference between the intentions of a user and the actions the system offers. Since VR aims to reduce visual interfaces to a minimum, as discussed before, users might be confronted with missing buttons or menus they are used to from desktop interfaces. For example, isolated search tasks [14, 223] or pattern recognition tasks [92, 19] which do not require complex and rich WIMP-interfaces benefit from post-WIMP interfaces. So far, interaction techniques developed and presented in section 4.5 aim

to support the seamless, intuitive visual communication with the system as far as suitable for the task. Simultaneously, user feedback should be processed and reacted to as intelligently as possible, avoiding long learning processes for interaction or time needed to explore interaction possibilities.

In summary, VR may face more challenges than traditional desktop visualization due to unfamiliarity or the conflict between maximizing immersion and interaction simultaneously. Nevertheless, this discussion proved VR to be a versatile tool that can be deployed for nearly every data domain when adapted reasonably. Hence, adaptability should be the primary goal when developing new tools and immersive systems. There cannot be one tool that solves all challenges mentioned, but scientists and developers should be encouraged to propose versatile and adaptable visualization tools that can be changed due to the user's needs and task specifications.

4.7 Conclusion and Outlook

In the last sections, the characteristics of virtual reality systems and the challenges of modern data analytics were reviewed, and solutions to the problems arising from combining 3D data visualizations in VR were proposed. The recent research highlighted in this review showed that VR offers new and exciting possibilities for data visualization, which should be further researched. The overview of 3D data visualization within desktop applications pointed out that a presentation of data in non-immersive Environments can be reasonable and, above all, is more available to non-professional users. Nonetheless, its traditional use is a disadvantage for VR because users are used to WIMP interfaces and expect to interact in VR in a way they usually interact with desktop computers. VR interfaces that overcome conventional interface's dominance are confronted with similar effects due to unfamiliarity.

Since 3D visualizations should be used only when suitable [292], tools that can include 2D and 3D visualizations are a proposed solution [36] that might address this challenge. In combination with multi-platform proposals, [87, 153], 3D data visualization would not have to be implemented either within an immersive or non-immersive system but could be utilized and applied individually and be context-dependent. Isolated fully immersive data visualization solutions of the latest state of art proved that VR is not just a potentially and hypothetically powerful tool but is used in several domains and has not quite reached its full abilities. Approaches presented in this summary also pointed out the importance of collaborative tools and platforms within VR systems. VR can play a significant role in academic, educational, and

professional domains thanks to the novel multi-platform and multi-user solutions mentioned. Moreover, when novel communication possibilities emerge and settle, they can enhance the naturalness of communication in VR, and immersive solutions may outperform current data visualization processes.

In terms of interaction possibilities in VR versus desktops, it was outlined that immersive visualization tools have various interaction possibilities to offer: from controllers to wands to data gloves to, most naturally, gesture control. The DoF allows the user to gain insights from a different perspective with natural movements as a standalone factor. Tilting the head to vary the view is more intuitive than moving a mouse or pressing keys. Even though the mouse and keyboard appear to be more familiar, regular use of VR might eliminate this advantage of traditional input devices.

Researchers pointed out that 3D might not be suitable for comparative data analysis because of perceptual issues such as distortion arising from a perspective view [198] and occlusion [16]. Nevertheless, confusion and cognitive overload can occur when the visualization is not well-prepared and conceptualized for 3D representation [142]. 3D visualizations in immersive environments are not as limited as 2D desktops by utilizing motion and relocation of the virtual objects by interaction. They are valuable tools for decision-making due to their perceptual advantages.

Also, navigation is a factor in VR systems that improves performance in specific tasks for immersive analytics. Studies showed that VR displays were preferred in data analysis tasks, primarily because users appreciate the ability to move freely within the environment and the visualization [14, 223, 205, 249] allowing humans to gain a better spatial comprehension of the relationship between different factors, variables, and objects within a visualization. Still, a connection has not been found between navigational user satisfaction and performance, accuracy, or speed [14]. In terms of interaction, it was highlighted that the realism of VR interaction techniques might enhance usability, immersion, and, subsequently, user satisfaction. It was also discussed how VR cannot replace traditional desktop-based data exploration and handling but will enrich the existing visualization approaches by novel means for visual representation and data interaction possibilities.

Furthermore, it was found that the advantages of VR become more prominent with the growing complexity and size of the datasets [104]. When the data analysis includes the exploration of non-linear relations and substructures, VR has turned out to be more advantageous in comparison to 2D displays[209]. Conclusively, VR proves to be a powerful tool in data exploration but relies on other application fields

on either intuitive interaction techniques or learning sessions to support the user in becoming familiar with the interaction possibilities.

As data visualization has always been important in decision-making [250, 188], this review revealed that immersive environments could improve and support decision-making processes as well. The user is assisted in the decision-making process by marking important structures, identifying and understanding cluster structure, and the intrinsic dimensionality of data within VR. It has been found that, in contrast to representations that are not applied in space or in time, physical action can immediately enable the user to perform analytical reasoning and decision-making [285]. The relation between VR and decision-making may also be underlined by the finding that immersion influences memory within the simulation [191]. Besides helping the user draw conclusions from the visualized data set, immersive approaches also support the acquisition process. Subsequently, the user's enhanced interests and engagement offer new possibilities for extracting knowledge from the visualized information. As a result, VR is a considerable tool for decision-making, especially within data sets linked to space or time.

As the most outstanding characteristic of VR, immersion has been the driving force concerning the user experience and user understanding of the data. Hinckley et al. stated that "... people do not innately understand three-dimensional reality, but rather they experience it" [134] back in 1994, and the findings of this review support his statement. Immersion can contribute significantly to experiencing the objects and data displayed in 3D compared to 3D objects displayed on a 2D desktop monitor. Hence, the user's participation is enhanced, and time and space cues can be interpreted more naturally [207]. Immersion also causes a strong feeling of presence and engagement, leading users to be more attentive and meticulous during their analysis tasks, which may result in fewer mistakes.

Several studies were presented that compared immersive versus non-immersive environments using the same set of tasks. These studies lack experimental control, as the two different types of presentation differ significantly. Factors supporting immersion, such as input device, display resolution, range of interaction possibilities, the field of view, and user's posture, have been potentially helpful but complicate the generic comparison of VR versus non-VR setups.

A burden that must be mentioned is that in VR and especially in applications for experts, studies are conducted with small groups of participants and possess an exploratory character. The explorative character of these studies makes it complicated to conclude potential tools that apply to the general public and are extensible. On the

other hand, it can be assumed that most participants are either students or associates of the institution, implying that they are most likely familiar with the field of research or data visualization itself.

The question of why 3D models cannot simply be displayed on desktops has been answered as follows: For one, visualizing a 3D object on a 2D desktop can cause issues such as inadequate depth cues [233, 64]. Especially for multi-dimensional data, traditional 2D desktop systems can hinder visualization. VR takes advantage of immersion, enhancing realistic effects, spatial understanding, and orientation. VR has the potential to demonstrate a more effective presentation of data and to offer higher user satisfaction [14, 249, 233], better accuracy [223], and increased depth of insights.

Coming back to the initial question of whether VR can enhance data visualization, insight into the role of user experience in data visualization was provided. The research presented in this work indicated that VR enhances, in many cases, the user experience but struggles to find detailed results concerning the influence of immersion on the performance of visualization and analytical tasks. It was found that whether performance is enhanced by immersive tools is a task-related issue and depends on the user's needs and experience. Most articles found that VR satisfies the user more than a desktop application [233, 187, 51] and Vr is often associated with better user experience [136, 87, 204]. The impact on performance depends on several factors, such as whether the user is exploring the data set to gain first insights [104] or search-specific tasks [223]. When the task does not benefit from the advantages of depth cues and perspective, VR often achieves results similar to non-immersive approaches or sometimes cannot compete; when the task requires abstract and complex interactions, the user does not yet know how to execute. Researchers have found that even a little experience in VR can improve interaction times in VR [212]. Thus, several introductory sessions may eliminate the factor of non-familiarity with VR systems. Nevertheless, as immersion enhances user experience, it fascinates, motivates, and inspires data analysis novices, leading them to engage further with the data set. The outcome of this conclusion is that a broader audience than expected may be eager to engage with, visualize and understand data.

Throughout this review, it was elaborated that VR must not be seen as a general solution to the prevailing challenges of data visualization but can count as a supplementary tool. Moreover, three approaches can be considered and followed:

1) the further development of multi-platform and possibly multi-user environments for the non-professional and educational domains, 2) adaption of immersive

interfaces to natural and intuitive interaction possibilities, and 3) continuous pursuit of a distinction between spatial- and abstract data.

As summarized above, VR proved its positive impact on user satisfaction and delivered promising but still specific and controversial results in terms of performance. Cooper et al. [72] and Andersen et al. [14] showed contradictory results concerning the influence of user experience on overall task performance. It would be interesting to find out whether Cooper et al.'s results, which indicated that an enhanced sense of presence might improve user performance, could be projected into a setting similar to that proposed by Andersen et al.. As Cooper et al. concentrated on multi-sensory feedback and Andersen on the immersion provided by the display, combining these two could satisfy the user's experience and improve effectiveness by enhancing immersion and making interaction more natural. More generically, it would also be helpful if versatile approaches, such as the one from Sharma et al. [257], were tested in a more extensive study, considering spatial and abstract data. In that case, the remaining question of whether VR can be useful for different data types could be answered more elaborately.

Despite the limited potential concerning graphics in fully immersive displays, most VR displays, such as HMDs, can render stereo. Even if the display is not stereoscopic, e.g., GearVR, the six DoF of VR input devices and the possibilities to use natural tracking of body, hands, and head are so strongly combined with the graphics rendered that the display seems to be in 3D. Mobile VR and affordable HMDs provide new application possibilities for non-professionals and personal users. As pointed out before, VR holds the ability to motivate novices to interact with data in a way that desktops could not inspire non-professionals. With more accessible VR devices, VR could conquer personal applications or educational fields with smaller budgets than professional institutions.

In terms of interaction, intuitive designs should be pursued. So far, Leap Motion and other gesture-tracking hardware did not prove themselves. Therefore, research should focus on controllers or, if suitable, pens or wands as long as gesture-tracking cannot compete with the accuracy of the other input devices. Nevertheless, it can be proposed that applications with an explorative character that do not demand high accuracy can be realized with natural hand gesture interaction. Studies should examine this application in particular.

To pursue a distinction between spatial and abstract data, the final proposal does not imply that a VR tool should not be designed to display one of the data types exclusively. For instance, the adaptable and versatile approaches that already exist

by Millais et al. [204] should be developed further. Millais et al. never conducted a confirmatory study with a more prominent and representative group. Such a study could prove that introductory sessions in VR could eliminate the issue of unfamiliarity.

Chapter 5

Direct Volume Rendering on HMD Systems

Direct Volume Rendering can be a beneficial visualization type for various medical fields. It provides direct insight into the data without the need for tedious preprocessing stages and can be used to render images resampling the real world as close as possible. The following chapter focuses on the research of direct volume rendering on state-of-the-art head-mounted displays. The content comprises two publications [310, 311] presented at international conferences. Direct Volume Rendering is an exciting topic for immersive virtual environments due to the spatial awareness generated by displaying the three-dimensional volumetric data in a stereoscopic environment. Compared to desktop monitors, VR can introduce more intuitive interaction with the data, resampling natural metaphors understandable even to non-expert users. However, the computational cost of DVR and the constraints of high update rates for HMD devices make DVR on HMDs a challenging combination. The following chapter provides two acceleration approaches that utilize fundamental characteristics of HMDs and human vision to significantly improve rendering performance for this domain.

5.1 Introduction

Scientific visualization and, in particular, volume rendering has been of considerable interest for immersive virtual reality applications for a long time [219, 100, 63, 152, 216, 48]. Due to the high cost and setup requirements, it has long been reserved for an elite group of researchers with access to the necessary resources. With the recent

developments in consumer VR systems, scientific visualization in VR environments has become of broader interest and has attracted many researchers' attention.

Different approaches to scientific visualization, particularly DVR, have been proposed for immersive environments over the years. Before the introduction of HMD systems, CAVES and stereoscopic displays were the go-to standards for immersive scientific visualization (See Section 2.4.1). As early as 2006, the research of Kratz et al. [159] presented high-quality volume rendering on GPUs for immersive stereoscopic displays and offered a collaborative approach for multiple users on the same screen. They reported that they could maintain update rates of roughly 10 Hz on their setup, which is far below what is commonly accepted even on desktop systems nowadays. CAVES and stereoscopic displays were also used in the work of Shen et al. [259]. Their tool MediVis was designed as an immersive and explorative visualization system for volumetric data sets. They rendered two viewports for stereoscopic data visualization, each with a resolution of 640x640 pixels. They reported an average update rate of 13 Hz for the visualization on the stereoscopic display, where the CAVE setup they presented was limited to 10 Hz.

While the quality of the rendered images was of acceptable high quality even compared to today's standard, the refresh rate and resolution of these systems were far below the requirements of HMD systems of our time. The increased rendering performance of modern GPUs will uplift the reported performance of these systems. However, considering the increase in resolution of modern HMD systems, the fragment-bound nature of Direct Volume Rendering, and the research of cybersickness symptoms, it is still a challenge to achieve interactive and immersive volume rendering systems for prolonged use.

Hanel et al. [124] researched the mandatory features of virtual environments to be suitable for DVR in 2014. The most crucial aspect of VR systems they established was the necessity of real-time head-tracking with extra low response times to minimize the chance of cybersickness symptoms. In addition, they found a minimum update rate of 60 Hz to be the lowest boundary for short-time use, while higher update rates are much more favorable.

In a first attempt to integrate DVR in current HMD scenarios, the group around Scholl et al. [251] elevated the minimum required update rate to 90 Hz to compensate for cybersickness symptoms. They addressed that a straightforward rendering approach is insufficient for most VR setups and provided an interactive virtual environment with the integration of DVR but had to make compromises to satisfy their constraints of 90 Hz. One of these compromises was their limitation on the sampling

rate. They limited the sampling rate to 150 samples along a single ray to increase the rendering performance, which could quickly lead to various artifacts due to the undersampling of the data. To still provide a visually pleasing result, they only tested their system with smaller data sets than commonly available in daily clinical practice.

Another approach that aimed to integrate volume visualization within immersive environments was that of Usher et al. [295]. Their team focused on a visualization tool for neuron tracing utilizing direct volume rendering in VR. While providing a valuable tool in an immersive environment, they still had to make compromises. The large scale of their data could not be visualized in real-time with sufficient refresh rates on HMDs. They decided to limit themselves to visualize smaller bricks of the original volumetric data to maintain higher refresh rates at a sufficient sampling rate.

Chheang et al. [63] demonstrated virtual reality in the medical domain by introducing software to plan liver surgery in VR collaboratively. To maintain high refresh rates with their visualization, they utilize indirect volume rendering to render the volumetric data as surface structures instead of DVR. Their research demonstrates the applicability of VR and volumetric data in the medical domain. Based on their results, it is assumed that a direct representation of the underlying data would be highly beneficial once the performance constraints are handled.

Another approach to accelerate rendering in VR was described by Meng et al. [201]. They presented a ray-casting approach in which the actual rendering resolution gets reduced on the non-dominant eye, reducing the system's total fragment load. While the calibration has to be done for every user, this approach can also present a practical approach for DVR acceleration by only introducing some visible artifacts in the user's fovea.

5.1.1 Constraints for DVR in Immersive Environments

The development of Virtual Reality applications comes with several constraints to the system. The biggest problem VR faces is the appearance of Cybersickness symptoms (see Section 2.5). Besides the individual sensitivity to cybersickness symptoms, the VR hardware and software can influence the symptoms' appearance.

Most often, cybersickness symptoms like nausea or dizziness occur due to the mismatch between physical input and visual feedback. On a hardware level, manufacturers aim to reduce the latency between the display of the rendered image and the sensory input by introducing HMDs with faster displays and tracking hardware with less overhead.

	Display Resolution	*Refresh Rate*	DoF	Tracking Type
Oculus Rift	2160x1200	90 Hz	6 DoF	Adjacent Reality Tracker
Oculus Rift S	2560x1440	80 Hz	6 DoF	inside-out tracking
Oculus Go*	2560x1440	72 Hz	3DoF	internal gyroscope
Oculus Quest*	2880x1600	72 Hz	6 DoF	inside-out tracking
Meta Quest 2*	3664x1920	120 Hz	6 DoF	inside-out tracking
HTC Vive	2160x1200	90 Hz	6 DoF	lighthouses with IR
HTC Vive Pro	2880x1600	90 Hz	6 DoF	lighthouses with IR
HTC Vive Pro Eye	2880x1600	90 Hz	6 DoF	lighthouses with IR
HTV Cosmos	2880x1700	90 Hz	6 DoF	inside-out tracking
Valve Index	2880x1600	144 Hz	6 DoF	lighthouses with IR

Table 5.1 This table lists currently available HMD systems from various vendors. All of them offer displays with refresh rates and resolutions beyond the still most prominent 60Hz and 1080p found on desktop computers. Even stand-alone devices (*marked by* *) provide higher refresh-rates and resolutions while offering much less performance.

As introduced in Section 2.4.1, current HMD systems on the low-end utilize displays with an update rate of 72 Hz on stand-alone mobile devices and 90 Hz on systems requiring separate workstations. On the high-end, modern HMD systems have already adopted even faster displays with 144 Hz or even 165 Hz.

For the DVR acceleration methods present in the following, the lower boundary of 90 Hz, as presented by Scholl et al. [251], will be considered as the minimum update rate of the system. Since the application field is the clinical practice, it is mandatory to minimize the occurrence of cybersickness symptoms for practitioners in service.

Another aspect crucial for the applicability of DVR in a virtual reality context is the resolution of HMDs. Displays build into modern VR systems provide a native resolution of at least 2160 x 1200 pixels on first-generation devices and 2880 x 1600 pixels on the state-of-the-art product line (see Table 5.1).

Comparing those values against the most distributed displays in the desktop market using the Steam Hardware Survey [296], it stands out that most desktop monitors in use are still running on slower 60 Hz panels with a resolution of 1920 x 1080 pixels. The higher resolution and update rate in VR systems require more powerful hardware than is often available on commodity hardware.

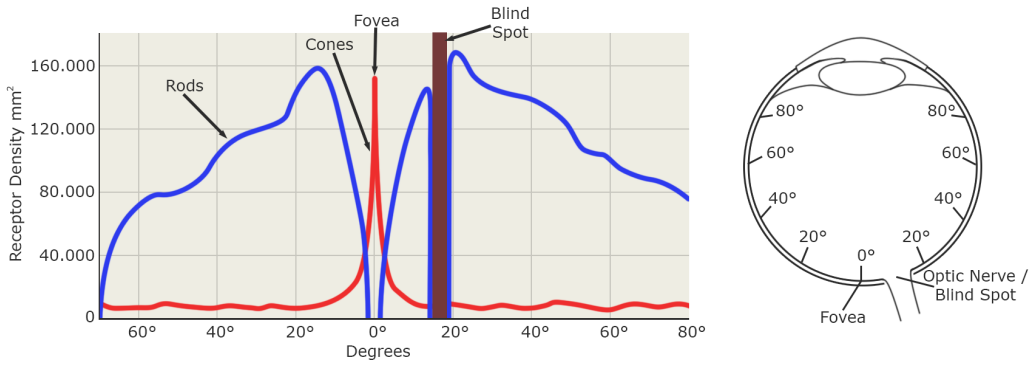


Fig. 5.1 Distribution of cones and rods in the human eye. The fovea centralis indicates the point of the sharpest sight. No receptors are located within the blind spot. Figure based on images found in [112].

DVR in its core, as described in Section 2.2.2, is an algorithm that heavily scales with the number of processed pixels. Even on desktop systems, providing visualization systems for volumetric data sets with interactive frame rates is often challenging. While the achievable update rate depends not only on the resolution of the display and the available rendering performance, it also depends on the size of the data set. As mentioned before, the first attempts for DVR in VR [251, 295] had to limit the sampling rate during ray-casting or reduce the volume's size by only visualizing smaller bricks of the data set.

Since HMDs provide an increased resolution compared to desktop displays and require a higher update rate to reduce the possibilities for cybersickness symptoms to appear, making a fragment-bound algorithm such as DVR feasible for those setups seems challenging.

5.1.2 Visual Perception

Characteristics of the human visual system play a significant role in the design of the presented acceleration methods.

Light rays enter the human eye by passing the cornea lens, where they get refractured and flipped by 180°. Once light rays pass this step, they hit the retina within the eye [112]. The entering light rays stimulate the photoreceptors located on the retina and get transformed into electrical signals. The brain interprets these signals and creates an image. Within the retina, two types of photoreceptors exist, cones and rods [96]. Each of these photoreceptors has a different purpose regarding human vision. Where cones are used for color and photopic vision, rods are used for scotopic vision and can only perceive achromatic light.

Cones can be further distinguished into three types. The retina has cones for lightrays of short, medium, and long wavelengths. Each of these cones reacts differently to the incoming light, which results in the perception of color.

Within the center of the vision is a small area called the Fovea, depicted in Figure 5.11. The degrees displayed in the Figure represent the angle within the human vision in regards to its distance to the Fovea located in the center. Another vital aspect of the visual system is the blind spot located in each eye. No photoreceptors are located at these locations, and the brain has to fill these gaps with information from the other eye.

The entire visual field of an average human comprises 180° on the horizontal and 130° on the vertical axis [96]. Furthermore, the visual system can be split into three areas: The foveal area consist of only 2° in each direction from the center. The parafoveal area reaches up to 4°, and the peripheral field starts beyond 4°.

Within the foveal area, only the color-sensitive cones can be found. No rods are located within this area. Furthermore, the density of cones is the highest within the foveal area resulting in the sharpest part of the visual system. The number of cones falls rapidly with rising angles in the field of vision but keeps a steady level in the peripheral field. As visualized in Figure 5.11 the density of rods starts to rise once leaving the foveal area of the retina and once again falls off with rising angles in the field of vision.

Within the foveal area, the human vision is at its sharpest [24, 70, 96, 278]. The critical part here is that, while at 0° we achieve 100% of the perceived sharpness, already at 5°, the visual system loses up to 50% of its visual sharpness [96].

Based on these characteristics of the visual system and the fact that commodity HMDs only use fixed lenses, reducing the possible focal point to a narrow field in the center, It is not mandatory to render every fragment at its full resolution.

5.1.3 Contribution

This chapter contributes by introducing two acceleration methods for direct volume rendering on immersive virtual reality systems. Both acceleration methods build on the characteristics of the visual system and the limitations of state-of-the-art HMD systems. The final acceleration method provides a stable refresh rate with no or minimal perceivable artifacts to the user.

Compared to solutions of GPU manufacturers, the presented approaches are hardware independent and can easily be adapted to any VR system. The presented

solutions reduce the rendering pipeline's overall fragment load to increase the system's responsiveness while focusing on delivering an unnoticeable reduction in image quality.

The approaches target the set refresh rate of 90 Hz for modern VR experience and are scalable to lower-end hardware. The first presented approach focuses on a fixed acceleration setup precomputed to executing the application. At the same time, the latter method introduces an adaptive system for the rendering pipeline, which helps to maintain the update rate at the sacrifice of possible quality reduction.

5.2 Motivation and Concept

The following acceleration methods for volume visualization on state-of-the-art HMDs are motivated by the need to reduce the chance of Cybersickness symptoms. Especially in the medical field, practitioners should not suffer from nausea or dizziness. While immersive volume visualization has been of interest for a long time in the clinical field, it is often avoided due to the rapid and unpredicted appearance of Cybersickness symptoms. As presented in Section 2.5, high latency between the generated image and the visual feedback can lead to Cybersickness symptoms. This latency can either be due to the long response time of the VR systems sensors to be processed by the software or a too high computational load on the visualization pipeline. Since volume visualization by itself is already a computationally expensive visualization method, it is often seen as not feasible for VR systems.

Conceptually the acceleration methods in the following are based on the observation that many of the pixels presented on state-of-the-art HMDs do not contribute to the perceived visual quality. The visual system underlies a non-uniform distribution of perceived sharpness and color due to the distribution of rods and cones found in the human eye (see Section 5.1.2). Therefore, only a tiny portion of the actual rendered image can be perceived as sharp while using an HMD. This concept is well researched on desktop rendering systems and is known as foveated rendering. Similar to the visual systems, available commodity HMDs utilize fixed lens setups distorting the incoming image (Figure 5.11) and reducing the possibility of manually shifting the focus point while using an HMD. The distortion introduced by the fixed lens setup plays an equally important role in the overall perceived image quality and should be considered in the rendering pipeline.

By utilizing the non-uniform characteristics of human vision and HMDs, this research aims to provide acceleration methods that achieve high refresh rates on

commodity graphics hardware to explore volumetric data sets using HMD devices. To be suited for broad acceptance in clinical practice, it is essential to not only target high-end workstations often found in research facilities but more to provide a solution that scales down to available hardware.

Visualization of three-dimensional scalar volume data can be classified as either direct or indirect volume rendering (Section 2.2). Both techniques are widely adopted in many scientific fields and are used daily. Indirect volume rendering has already been implemented in VR systems [63] and is based on extracting proxy surfaces from the volumetric data. These surfaces are often represented by polygonal meshes, which can be rendered very efficiently on modern GPUs. This method's potential downsides are a resource-intense pre-process and a significant loss of information due to the necessary selection of a proxy quantity, e.g., an iso-value. Due to some data sets' fuzzy nature, focusing on a small set of proxy values may make it impossible to use indirect volume rendering.

On the other hand, direct volume rendering, as the focus of this research, can deliver more profound insight into the actual 3D data structure provided to the system. It is flexible in its visualization due to the use of transfer functions (Section 2.2.3), which can be adjusted in real-time. DVR is widely adopted in domains where detailed information of the raw data is crucial. Even though direct volume rendering provides more insight and does not rely on time-consuming pre-processing, indirect volume rendering is the dominant visualization method in virtual reality setups.

The downside of DVR, while providing much more information to the users, is the computationally expensive ray-traversal stage of the core algorithm (Section 2.2.2), which has to be evaluated for every single fragment covered by the volumetric data. This per-fragment evaluation makes DVR a heavily fragment-bound problem, scaling with the total number of fragments covered on display.

For the last decades, research was primarily focused on accelerating the per-fragment ray-traversal to achieve interactive frame rates for DVR where the term "interactive" diverge between desktop and VR applications. Hierarchical data structures are often used to speed up the ray computation by skipping empty spaces, and opacity thresholds are integrated to terminate rays early. Another well-established technique is progressive rendering, which reduces the quality during motion and generates a high-resolution image once the viewport is fixed.

While the research in this field resulted in algorithms to achieve interactive refresh rates on desktop applications, especially with ever-growing data sets, these refresh rates are usually below the necessary target rate for VR applications. Typically, users

consider a refresh-rates of roughly 20-30 Hz as interactive on desktop applications. In contrast, VR applications, as introduced, target much higher refresh rates of at least 90 Hz to compensate for the risk of cybersickness symptoms.

In addition to targeting higher refresh rates, HMDs tend to have a higher pixel resolution than most desktop monitors. Considering the ratio of fragments covered between an HMD and a desktop monitor, we can assume that, due to the explorative nature of VR, the volumetric data will cover most of the fragments found on an HMD. In contrast, most desktop applications only use a small portion of the screen for the 3D visualization, dramatically reducing the number of fragments covered by the volume. Based on the necessity of high refresh rates, high HMD resolutions, and DVRs fragment-bound nature, the combination of VR and DVR is often seen as a not viable option to be widely adopted.

As motivated to provide an interactive VR experience for the exploration of volumetric data using Direct Volume Rendering, it is mandatory to develop acceleration methods tailored directly toward HMD systems. Instead of accelerating the ray-traversal, the focus has to be on the number of fragments processed during image generation. Of course, all the established algorithms to speed up the ray traversal should be compatible with the presented acceleration method.

Most closely related to this research are gaze-based DVR systems using foveated rendering [176]. As with gaze-based, or foveated rendering systems, the resolution is non-uniformly distributed according to the user's focus point, the fovea. Many desktop systems have already adapted this concept and use eye-tracking hardware to track the user's focus point in real-time. While foveated imaging significantly reduces the fragment load of a rendering system, it has one significant drawback that makes it unsuitable for the acceleration of the direct volume rendering approach. It requires an even higher update rate. As the users' gaze can travel at very high speeds, the image must be updated to follow the center of gaze.

Consequently, foveated imaging is rarely used in practical volume rendering applications, despite the availability of reliable eye-tracking hardware for the desktop market. Nevertheless, this drawback does not have to be of concern when working with current HMD devices. Eye-tracking in HMDs is still in its infancy, and currently, widely available HMDs only use a fixed lens setup, limiting the range of sharp pixels presented to the user. Typically users have a fixed focal point when using HMDs and tend to move the entire head instead of the eyes. The narrow focus achievable by HMDs removes the need for fast eye-tracking hardware while still being able to use a similar approach to most foveated rendering systems.

5.3 Pipeline Overview

The general pipeline of the acceleration method can be described as a deferred rendering approach with additional processing steps added to the process. Both approaches are realized as pre and post-processes to the ray-traversal and can be integrated into any volume rendering system without many changes to the already present system.

At the core of most direct volume rendering systems implemented today is a variation of the GPU-based ray-caster proposed by Krüger and Westermann [161]. For such approaches, a ray's entry and exit position and, consequently, the ray direction are computed first. In the following, the ray traversal is executed based on the obtained ray information. In 2003 Krüger and Westermann had to design their algorithm around several restrictions in graphics hardware. They designed a multi-pass algorithm for both the ray parameter computation as well as the ray traversal itself.

Today, both steps can be integrated into a single pass [275], but still, the computation of ray parameters is primarily independent of the following ray traversal and is not considered performance-critical as the ray traversal is orders of magnitude slower. In this work, the initial ray parameter computation is modified to reduce the input for the ray-caster but leaves the ray traversal itself unchanged. Since the traversal stage is untouched by the acceleration method for HMDs, this approach allows the combination of practically any ray traversal mechanism with the presented method, including recent ray-guided volume rendering systems.

Conceptually, both approaches generate reduced ray-entry and ray-exit buffers stored in texture maps of a lower resolution (see Figure 5.2 (top-left)). These generated textures are subsequently used during the ray-traversal stage (see Figure 5.2 (center)). The result of the ray-traversal is a lower resolution representation of the final image submitted to the HMD. The resulting image is handed back to a post-processing stage for the reconstruction of the original resolution (see Figure 5.2 (bottom)). Hence, the approach can be considered a bracket around existing ray traversal methods.

Since the aim is to provide acceleration methods, which stay independent from the actual ray-caster implementation, the only option to reduce the number of rays evaluated is by modifying the ray-caster input. Therefore, two additional steps were added to the rendering pipeline (see Figure 5.2). As with most multi-pass GPU ray-casters, the first step is to generate the bounding geometry of the data set and

store it as a texture. Instead of providing the generated texture as input to the ray-caster, these approaches add a "fragment reduction stage" prior to the ray evaluation. This stage reorders the fragments of the bounding geometry to reduce the total number of fragments covered and therefore reduces the total number of executions during ray-traversal. Over the research on this topic, two different approaches have been developed to reduce fragments. Depending on the approach presented in the following sections, this stage uses either pre-computed textures specified for the connected HMD or a dynamic lookup function adjustable during run-time.

The resulting buffers of the reduction stage still represent the entire bounding geometry of the data but order them into different resolution layers for performance benefits. The two approaches developed for this process are covered in Sections 5.4, and5.5. Since the ray entry and exit information has not changed, just rearranged, the resulting textures of this stage are still compatible with mainly any ray-caster implementation that follows the rules of Section 2.2 and utilizes entry and exit buffers as ray-traversal input.

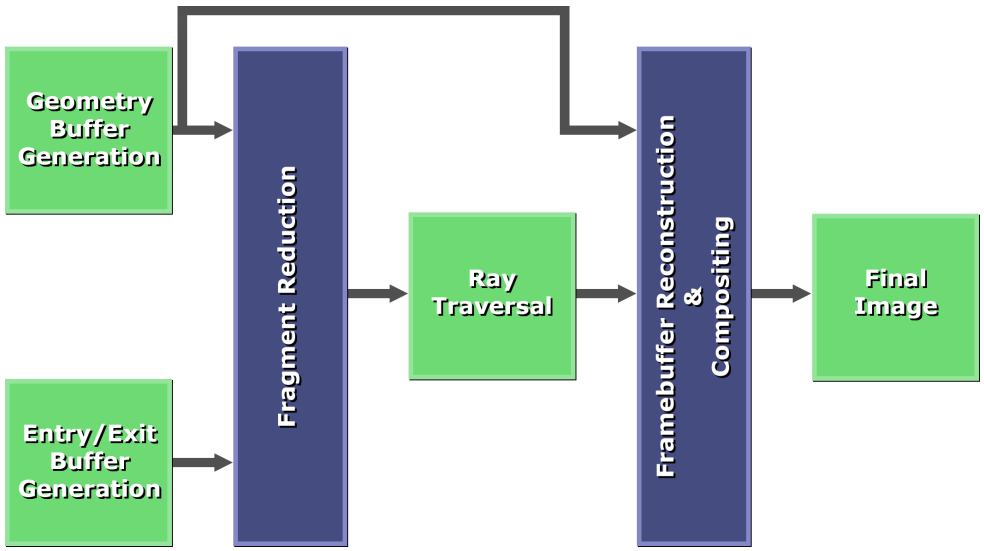


Fig. 5.2 The typical DVR rendering pipeline consists of the generation of entry- and exit-buffers of the volumes bounding box with additional framebuffers of the scene geometry. After ray traversal, the volume buffer is composited with the scene geometry before it is displayed to the user. The rendering pipeline is enhanced for this acceleration approach by adding a pre- and post-process to the ray traversal.

After executing the ray-caster, the result has to be restructured and reconstructed for the final presentation. The output of the ray-caster is based on the resampled input texture of the reduction stage, which is not immediately presentable to the user. Therefore, a reconstruction stage has been added to the pipeline, taking the resulting

images of the ray-caster and recreating a full-resolution image based on the pipeline parameters.

In many volume rendering scenarios on desktop computers, only the volume itself has to be rendered without any other geometry potentially intersecting the volume. Consequently, many rendering systems are built around the simplifying assumption that no other interactions with other geometric primitives have to be considered.

In virtual reality, however, it is crucial to be able to render additional polygonal models, such as the user's avatar, controllers, and a surrounding environment for spatial orientation. Therefore the acceleration methods must consider other geometry information during the processing stage of the incoming ray-entry and ray-exit textures. The pre-processing stage considers the already rendered geometry of the scene by including buffers consisting of the geometric position as input parameters. The incoming spatial information of the geometry is merged with the rendered volume and used to compose the final image with respect to the polygonal scene.

5.4 Precomputed Textures Approach

The first acceleration method presented in this chapter is based on pre-computed textures, representing a fast method for rearranging the entry and exit buffers in a single path without much computational overhead during run-time. This approach was the first tested and evaluated method and the introduction to this research field. It was born out of the need for a fast solution to make DVR feasible on HMD systems, even on older hardware not capable of the newest features. The decision for pre-computed lookup textures was made due to the low execution cost during run-time and fast implementation. The presented method focuses on circular patterns, which approximately correspond to the falloff found in human vision. However, various other patterns could be feasible for the down- and up-sampling textures. These could vary in the form of down-sampling and actual storage in the texture to condense more space.

5.4.1 Fragment Reduction

Research in ray-caster acceleration has often focused on accelerating the solving of the ray-equation for every single fragment or on progressive rendering systems converging toward a final image over time once the viewport is fixed. While the ray-

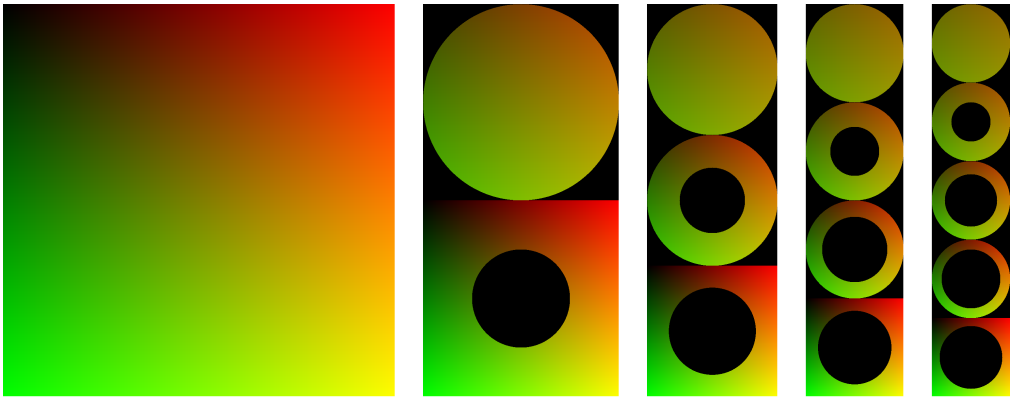


Fig. 5.3 The system computes multiple lookup textures to generate the subsampled, low-resolution version of the entry- and exit-buffers. Besides storing a full-resolution UV-map (**left**), the system can generate arbitrary numbers of subsample textures for each level. The top structure of each level represents the center of the screen, the user's gaze, in its native fragment density. The colors displayed in the textures are used to resample the incoming image at the given UV coordinate.

caster used within this system already supports various per-ray acceleration methods, other solutions had to be found to increase the rendering performance. Progressive rendering was no option in this case since VR systems are in a continuous flow of camera movement due to the head-tracking of the user. As the computation time of DVR scales with the number of fragments processed during ray-traversal, the increased resolution found on state-of-the-art HMDs does not help in increasing the rendering performance. However, the non-uniform distribution of the projected pixels while wearing the HMD does help to introduce another approach for performance improvement.

This solution comes in the form of sets of pre-computed textures for resampling the original input texture into textures of a smaller resolution. During the initialization of the system, multiple subsampling textures are computed to be used during rendering. These subsampling textures depend on the connected HMD and its native resolution. They are used as lookup images, mapping fragments from the source buffer, including the entry and exit coordinates of the bounding geometry, to a target texture, which will then be used during the ray-traversal stage of the rendering pipeline. Depending on the system load, the targeted refresh rate, and the data set size to be visualized, the system can switch between different texture setups, making the overall technique highly scalable to various hardware setups.

Figure 5.3 displays multiple subsampling textures computed by the system. The normalized coordinates of the original input texture are stored within the image's red

and green color channels. In the pre-processing stage, each pixel of the subsampling texture is evaluated according to its stored coordinate and the input texture sampled to generate a new texture. The volume's bounding geometry images are used as input for the pre-processing stage. The Figure demonstrates the spherical patterns used to simulate the circular falloff of the cones in the distance to the focal point of human vision.

For the following, some naming conventions have to be laid out for these textures. In this approach, each of these textures is called a subsampling level, where each level can contain multiple resolution stages. For each subsampling level l, $l+1$ reduction stages are generated in the texture. Level 0 represents a copy function mapping each fragment of the full resolution input to the identical fragment of the output texture, hence reducing no fragments for the ray-caster.

Beginning with subsampling level 1, multiple stages are introduced to the texture, each reducing the incoming texture further in its resolution. As the focal area of the user should maintain its full resolution, the most center pattern at stage 0, top of each level, simply copies the input data to the output without changing anything to the fragment density. The following stages represent a scaled portion of the native full resolution of the input texture, where each stage s scales the original resolution by the factor

$$scaleFactor(s,l) = 1.0 - \frac{s}{1+l}. \tag{5.1}$$

where l is the subsampling level of the texture, and s the exact resolution stage within this texture.

As with increasing subsampling levels, the number of fragments left over from the native resolution falls rapidly. The percentage of remaining fragments for the pictured subsampling levels is displayed in Figure 5.5. Compared to the native resolution, the first subsampling level 1 already reduces the percentage of covered fragments from 100 % to $\tilde{4}0\%$. While the processed number of fragments is more than halved, due to the spherical design focused on the focal point of vision, most of the actual perceived image still remains in its full resolution. The following levels further reduce the total fragment load to 22.42% in level 2, 14.78% in level 3, and 10.62% for level 4. As the number of discarded fragments grows with higher subsampling levels, several artifacts enter the perceivable area of the center of the screen. Different trade-offs between fragment reduction, perceived image quality, and system performance are presented during this technique's Evaluation Section 5.4.3.

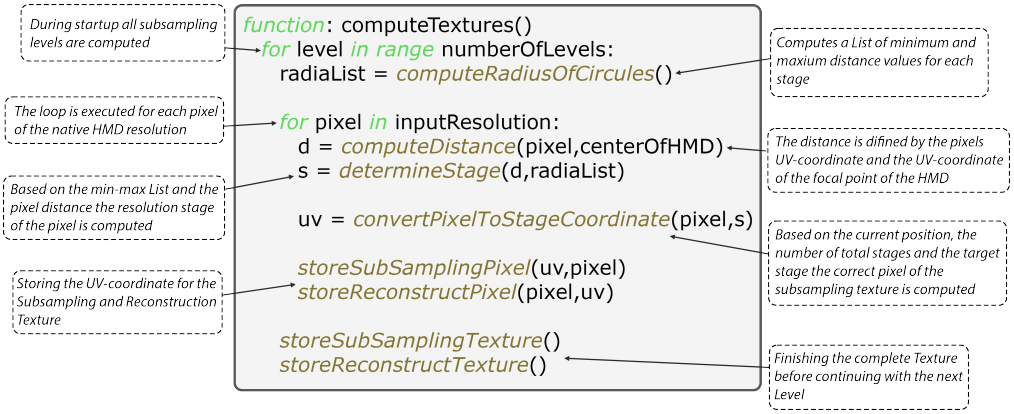


Fig. 5.4 The basic concept of the creation of the subsampling and reconstruction texture.

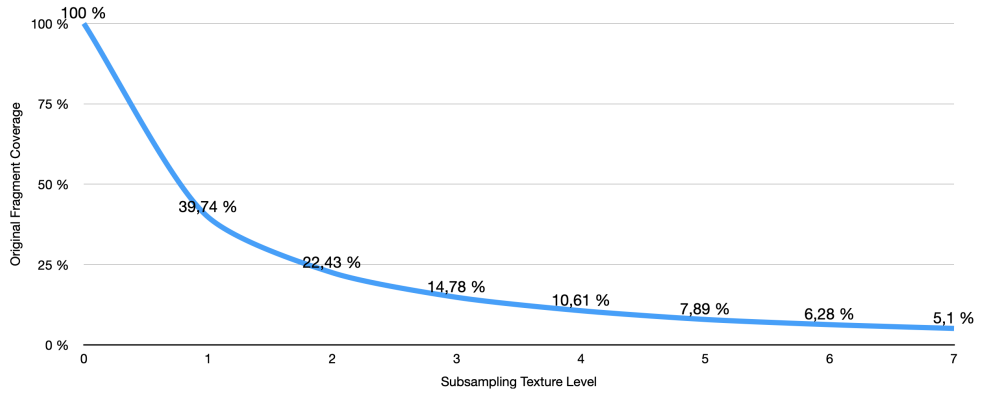


Fig. 5.5 The Figure displays the percentage of remaining fragments compared to the input texture depending on the selected subsampling level. While level 0 corresponds to the native input resolution, starting from level 1, the number of remaining fragments gets reduced significantly.

5.4.2 Texture Reconstruction

The output image of the ray-traversal stage can not directly be presented to the user without reconstructing its original layout. As the preprocessing stage resampled the bounding geometry into the corresponding resolution layers, a texture reversing this step has to be used. As for the subsampling stage, these textures are precomputed at the same time during the initialization of the system. Reconstruction textures corresponding to the already displayed subsampling textures of Figure 5.3 are displayed in Figure 5.6

A bilinear interpolation is applied to fill the missing information during the upscaling of the lower resolution levels. This type of interpolation approximates the missing information by interpolating between the available data points. While

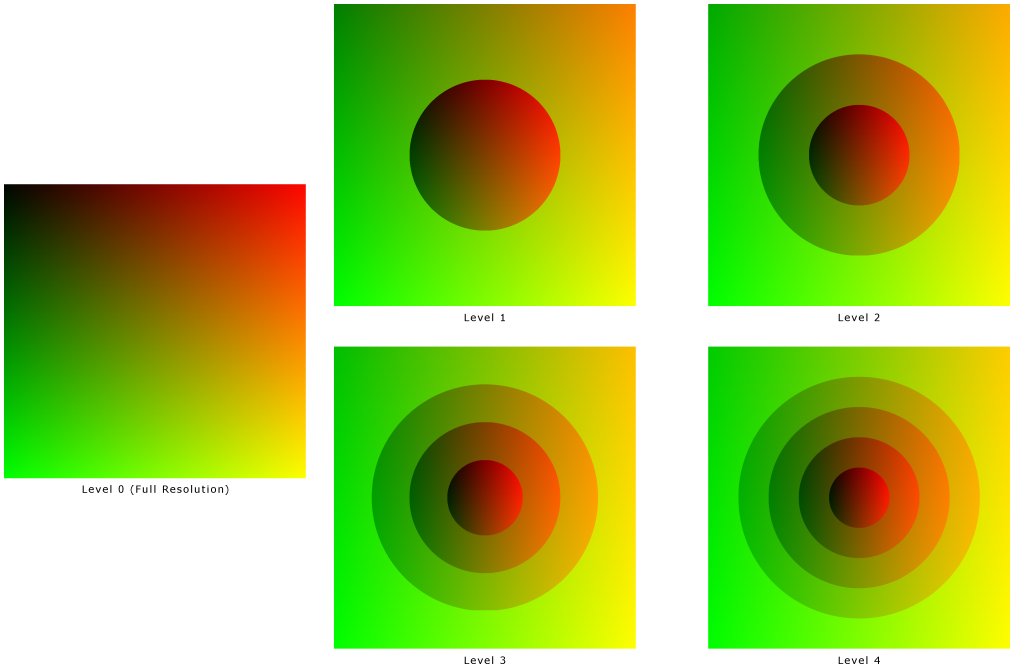


Fig. 5.6 For the reconstruction, a set of textures are mandatory to reverse the initial subsampling stage. For each subsampling texture exists a corresponding reconstruction texture that resamples the result of the ray traversal into a presentable image format.

in recent times, newer upscaling solutions have found their way into modern GPU hardware, such as DLSS on Nvidia GPUs or FSR on AMD hardware, they require more computational performance, which is a limiting factor for this type of application. Additionally, the bilinear interpolation results in a blurry image, which matches humans' already reduced visual capability in the peripheral field. Therefore, a straightforward upscaling approach, such as bilinear interpolation, fits the problem tackled by the acceleration method for HMD devices.

As gaps could appear on edges of neighboring resolution levels during reconstruction, an overlap between levels is introduced. In this overlap region, portions of the image are rendered multiple times during ray-traversal. While this introduces an overhead in rendering performance, compared to the total number of fragments reduced, this overhead is neglectable. However, rendering an overlap area allows for a smooth blending making the different resolution levels less noticeable. More of the image quality evaluation can be found in Section 5.4.3.

5.4.3 Results

It is mandatory to evaluate multiple aspects of the system to verify if the acceleration method is sufficient to achieve the desired performance for DVR on HMD systems. The upcoming section focuses on three critical aspects to assess the value of this approach. The first is the performance of the DVR system. As the goal was to achieve an update rate of at least 90 Hz for commonly available data set sizes in the clinical field, it is mandatory to verify whether this goal could be achieved. The second aspect evaluated is the loss in image quality. The approach lowers the number of processed fragments and upscales the remaining fragments to the native resolution, which reduces image quality. The last aspect that must be verified is the system's applicability in the clinical field. Preliminary tests were conducted, allowing field experts and medical students to evaluate the system's performance, quality, and experience.

Performance

	MRI			*CT*		
	128	*512*	*1024*	*128*	*512*	*1024*
[A] no Acceleration	81,67	31.81	20,38	123.21	68.39	59.26
[B] with Acceleration	121.43	51.31	32.43	164.95	87.84	75.42
[C] FAVR LvL 2	138.33	94.79	51.46	239.38	151.64	101.19
[D] FAVR LvL 4	161.01	101.18	62.08	286.87	210.54	143.37
[E] FAVR LvL 6	193.11	112.30	74.81	300.00	221.12	153.12

Table 5.2 Results of our rendering method compared to a ray-caster without any acceleration methods applied *[A]* and a ray-caster using empty-space skipping and early ray-termination *[B]*. We examined an MRI scan (512x512x792), and a CT scan (512x512x256) and compared different sampling levels *[C,D,E]* to gather more information about the scalability of our approach. The resulting numbers represent Frames per Second (FPS) achieved by each setup.

The update rate and the system's response time play a critical role in HMD systems and the appearance of Cybersickness symptoms. It is necessary to improve the rendering performance and minimize the chance for Cybersicknes to make DVR a feasible visualization technique for HMD devices in the clinical field. While desktop applications with frame rates of 20 - 30 Hz can still be seen as acceptable, these low update rates physically strain the user in virtual reality. The goal was to achieve a refresh rate of at least 90 Hz, the most common refresh rate of displays

found in HMDs. However, it is desirable to design the system to be scalable enough for future HMDs with high displays supporting higher refresh rates.

A workstation equipped with a Ryzen 7 2700X processor, 32 GB DDR4 3200 Mhz memory, and an Nvidia GTX 1080 graphics card was used to evaluate the system's overall performance while visualizing medical data sets found in daily clinical practice. The VR system used during performance tests was an HTC Vive Pro with a 2560 x 1440 pixels resolution.

A fixed set of interactions was recorded to guarantee that each tested technique renders the same set of images over a 3-minute benchmark sequence. The interactions consisted of translation, rotation, and scaling tasks with additional manipulation of the transfer function and real-time editing of the data set. The data sets originated from field experts, which were also incorporated during the preliminary study. They provided multiple data sets, of which two more complex ones were chosen for the performance tests. These two data sets represent commonly found data in daily practice, an MRI scan of the human torso with a resolution of 512 x 512 x 792 voxels and a CT scan of the head with a resolution of 512 x 512 x 256.

Five rendering setups were evaluated, including three fixed sampling rates for each data set. The five rendering setups included a ray-caster implementation without any acceleration method, which provides a baseline for the performance of the used workstation. Afterward, basic acceleration methods were added to the renderer, including empty-space skipping based on an underlying octree structure and early-ray termination. These acceleration methods were also enabled while evaluating the presented acceleration method.

While the rendering pipeline can switch between different precomputed reduction levels on the fly, depending on the previous frame's rendering performance, for evaluating the individual rendering levels, this dynamic adjustment option was disabled. By focusing on distinct reduction levels, it is more straightforward to estimate the system's expected performance. For the resolution levels tested, three options were selected. These three levels were also evaluated regarding the resulting image quality in Section 5.4.3.

In addition to different reduction levels, a variation in the sampling rate has been tested. The sampling rate describes the step size during ray-traversal, where a larger sampling rate corresponds to smaller steps resulting in more accumulated values during the traversal process. The larger the sampling rate, the more costly the ray-traversal becomes during DVR. A reduction in sampling rate is often a chosen option to improve the performance of the DVR system. However, reducing the

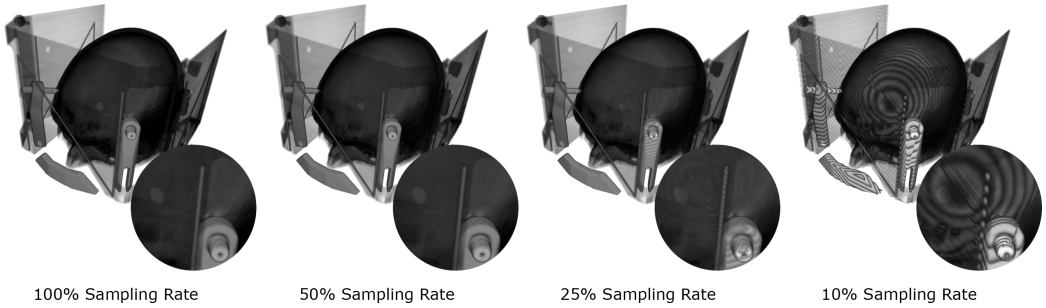


Fig. 5.7 The Figure demonstrates different sampling rates during the ray-traversal stage. Choosing too low sampling rates introduces artifacts in the final image.

sampling rate too much can result in various artifacts due to undersampling the data set (Figure 5.7). Sampling rates of 128, 512, and 1024 samples were tested for each rendering approach and data set. Limiting the refresh rate can be a solution to improve the rendering performance, as seen in the work of Scholl et al. [251].

The results of these tests are displayed in Table 5.2. While the baseline solutions are insufficient to achieve reasonable refresh rates, even low reduction levels of this approach could already improve the performance to exceed the targeted 90 Hz threshold.

The baseline implementation of a ray-caster cannot maintain high enough refresh rates for VR applications. While the CT data set could be rendered with an average of 123.21 Hz, this was only achievable with a massive undersampling of the volume by applying a sampling rate of 128 samples per ray, which can be seen as sampling only each fourth value of the data set. The accelerated ray-caster follows by achieving refresh rates above 90 Hz for both CT and MRI data sets. However, a sampling rate of 128 samples does not result in sufficient image quality for clinical use due to too many artifacts and missing imaging data. To work around the introduction of artifacts due to undersampling, selecting a sampling rate that reads each voxel data passed by the ray at least once is mandatory. In the case of the MRI and CT data set used in this test, a sampling rate of at least 512 samples is necessary. However, neither the baseline ray-caster nor the accelerated ray-caster could keep the refresh rate above 90 Hz while using a suitable sampling rate.

In comparison, even the first levels of the subsampling approach could increase the performance to the point where it can be seen as a feasible option for VR applications. The presented approach could achieve a significant speed-up in rendering performance compared to the other two solutions. For the MRI data set and 512 samples, the ray-caster gained an additional performance between 84.7% on reduction

level 2 and 119.9% on level 6 compared to the optimized and accelerated ray-caster. Even the less demanding CT data set showed a performance increase between 72.2% and 151,7%.

Image Quality

Due to the reconstruction stage necessary after ray-traversal, several upsampling artifacts are introduced to the image. As these artifacts result from the upsampling, they are not present in native full-resolution images as they would be rendered on desktop monitors. Especially in the clinical field, artifacts and errors in imaging data should be avoided. While these errors could be seen as critical, they are mostly not perceivable while using an HMD. As the lenses found in state-of-the-art HMDs are fixed and only provide a small corridor for the user's view, the introduced upscaling artifacts in the peripheral vision can not be perceived.

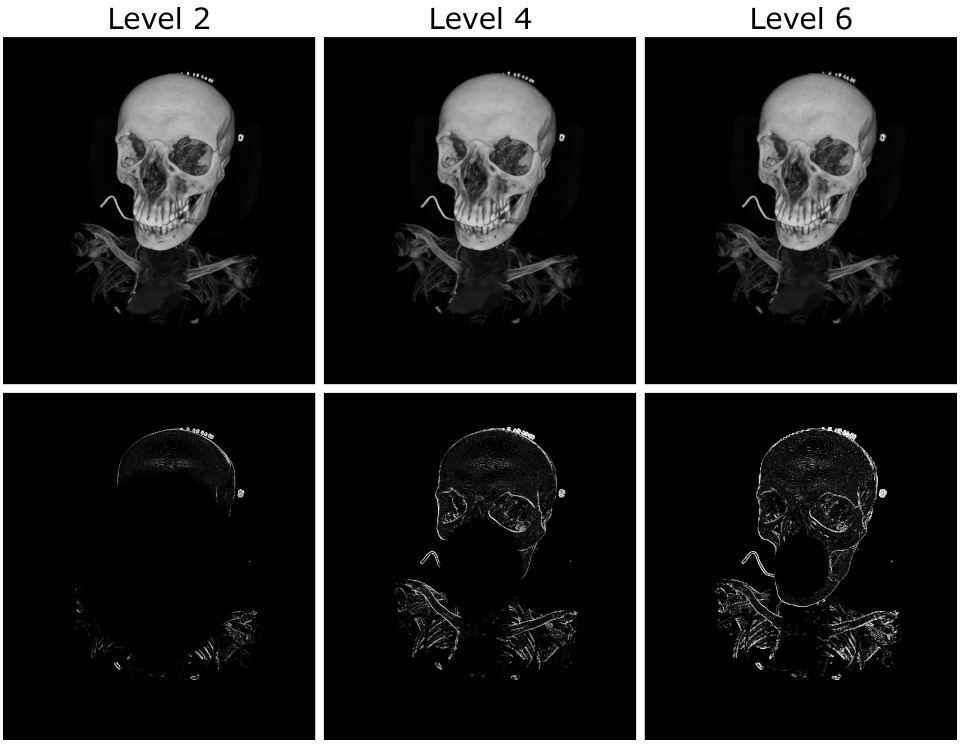


Fig. 5.8 This image series represents three levels of the acceleration approach present in this work. The displayed image is a CT scan of a patient; other data sets can be found in Figures 5.9. The bottom row shows the squared error compared to a native resolution rendering of the same viewport. The corresponding PSNR values are shown in Table 5.3.

Nevertheless, this section evaluates the objective image quality achievable by the approach. The Peak Signal-to-Noise Ratio (PSNR) [158] is computed to quantify the difference between a traditional rendered image on a full-resolution canvas and the result of the reconstructed image. A fixed sequence of images with the identical final resolution are rendered and written to disk. Afterward, the PSNR for each frame is computed. As with the performance evaluation, the same reduction levels of two, four, and six were tested. Based on performance and preliminary tests, moving beyond level six only increased the error in the image without providing benefit in terms of rendering performance and user experience.

	FAVR LvL 2	*FAVR LvL 4*	FAVR LvL 6
Torso	46.44 dB	39.97 dB	38.14 dB
Atery	49.66 dB	44.34 dB	40.04 dB
Head	47.16 dB	31.52 dB	39.99 dB

Table 5.3 A sequence of images (Figure 5.8) was evaluated based on their image quality using PSNR. Based on the results, the objective classification of the reconstructed images is a high-quality lossy video stream. However, as field studies indicate, the loss in quality is not perceived.

Figure 5.8 displays selected images of the rendered sequence. Next to each image is the error image representing the difference of the image compared to a native native resolution rendering. The computed PSNR results are displayed in Table 5.3 with an average result for each data set and subsampling level. As the literature suggests [158], PSNR values above 30 dB are seen as lossy image compression noticeable to the human eye. The analysis indicates that this approach results in images with several artifacts that are noticeable to the user when looking at the image. However, it is important to point out that this quality loss appears in the user's peripheral vision. As the displayed images show, the center portion of the image remains without any difference compared to the native resolution. Therefore, the introduced error is only perceivable to the user as long as these frames are viewed on traditional desktop monitors. Once they are submitted and viewed while wearing an HMD, these errors get lost in the peripheral vision.

While the PSNR value represents an objective measurement of the image quality, the user's subjective perception must be evaluated in more detail. Preliminary tests were conducted to gain insight into the experience using the system and are presented in Section 5.4.3. The section details more information about the user experience and

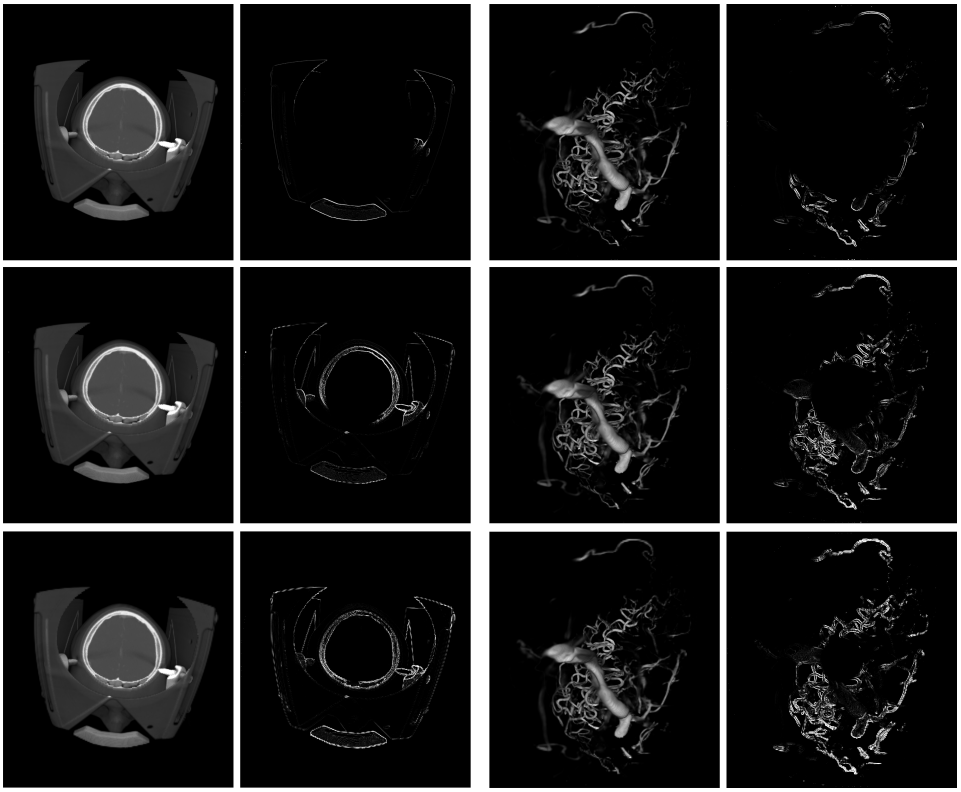


Fig. 5.9 This image series represents three levels of the acceleration approach present in this work. The displayed image is a MRI scan on the left and a contrast enriched CT scan on the right. The bottom right images show the squared error compared to a native resolution rendering of the same viewport. The corresponding PSNR values are shown in Table 5.3.

the feedback collected from field experts and medical students testing the system regarding performance and image quality.

Preliminary Tests

The prior two Sections concentrated on objective measurements of the system's image quality and rendering performance. While it could be established that this acceleration method increases the system's rendering performance, it was also pointed out that the result suffers from artifacts introduced due to the upsampling to the final resolution. In comparison, this section presents the subjective feedback and results of the user's experience of the DVR - VR environment in the form of preliminary field tests.

The fundamental assumption of this acceleration method assumes that artifacts in the user's peripheral vision are not noticeable due to the nature of the human

visual system and the fixed lens setup in HMDs, reducing the focal corridor of the image to a narrow central angle. For the preliminary tests, 13 participants could be conducted, including experts from the field of neurostimulation therapy and medical students. Each participant had experience evaluating CT and MRI data. For the tests, users explored multiple data sets to their liking. The rendering system offers various interaction methods allowing them to translate, rotate, scale, and even edit the data set in real-time. Especially the real-time editing and coloring of the data set introduced an additional system load we had to compensate for during run-time.

At first, to evaluate whether users noticed any difference between the activated acceleration method and a non-accelerated visualization, a small data set of 128^3 voxels was rendered, which was still possible to visualize with at least 90 Hz on a straightforward ray-caster solution approach. During exploration, the system switched randomly between different reduction levels and a native resolution. Afterward, a larger data set was tested with a resolution of 512^3 voxels, resembling the scans the experts have to work with daily. In this case, it was not only the peripheral resolution that was from interest but further the refresh rate of the system.

The assumption that quality loss in the peripheral vision is not noticeable to users during VR exploration is not new. The lenses found in HMDs only project a small area of the screen in its full resolution to the human retina, while other areas get stretched and distorted [220].

As the interviews indicated, none of the users could notice a difference in image quality while exploring the 128^3 voxel data set up to a reduction level of 5; neither did they notice the adjustment at run-time. Starting from level 6, two of our participants could notice a reduced image quality in the center of the screen, which can lead back to the fact that starting from level 6 on, the central full resolution area became too small for the central vision.

Once a soft threshold of reduction level 5 for all participants was found, the larger data sets were explored. Especially data sets with a state-of-the-art resolution of 512^3 voxel were of interest. As outlined in Subsection 5.4.3, a ray-caster without the presented subsampling approach could not maintain refresh rates of 90 Hz. Again, participants explored the volume and edited and colored the data, all without the activated reduction method. All users immediately noticed the system's lower responsiveness, and two participants had to end the study due to dizziness and nausea. After one minute, the system activated the subsampling acceleration, increasing the system's performance. The participating users immediately noticed the gained fluidity, and no additional user had to stop the task.

After interviewing each participant and evaluating the provided feedback, the following results can be recorded. The threshold for this approach of precomputed subsampling textures is at level 5 of the system. Visualizing larger data sets indicates that low refresh rates are more noticeable to the users than the reduction in resolution. Furthermore, inconsistent timing between frames can lead to cybersickness symptoms. The presented approach can help to avoid cybersickness symptoms and, therefore, can be seen as a feasible approach for DVR in VR environments.

Alternative Pattern Storing

While developing this acceleration method for DVR on HMD systems, various forms of precomputed textures have been tested. The textures shown in Figure 5.3 have established themself as the best possible solution for the tested hardware. However, it is observable within the Figures that a large portion of fragments does not contribute to the ray-traversal stage. These fragments are displayed as entirely black in each of the presented Figures. These fragments do not map any entry or exit information of the bounding box texture to the new frame buffer utilized during ray traversal.

As these empty fragments increase the number of invocations of the fragment stage without any result, reducing the number of empty fragments can be seen as desirable. For this purpose, various arrangements for the precomputed textures have been tested. Some of these textures are displayed in Figure 5.10.

Each of these textures was also tested regarding the rendering performance of the system. As the size of these textures is significantly smaller than the chosen option, the rendering performance would be expected to increase. However, the system's performance was considerably slower. The benchmark mentioned in Section 5.4.3 was used to evaluate each storage option's performance. The evaluation results are shown in Table 5.4. Compared to the simple radial function used throughout this

	LvL 2				LvL 4			
	FAVR	A	B	C	FAVR	A	B	C
MRI	94.79	49.22	71.38	74.91	101.18	54.61	74.27	76.81
CT	151.64	55.78	84.17	101.82	210.54	72.12	134.66	152.51

Table 5.4 Different storage textures were tested for fragment reduction. FAVR represents the default storage option as displayed in Figure 5.3. A, B, and C are the patterns found in Figure 5.10. Other storage options reduce the performance of the system significantly.

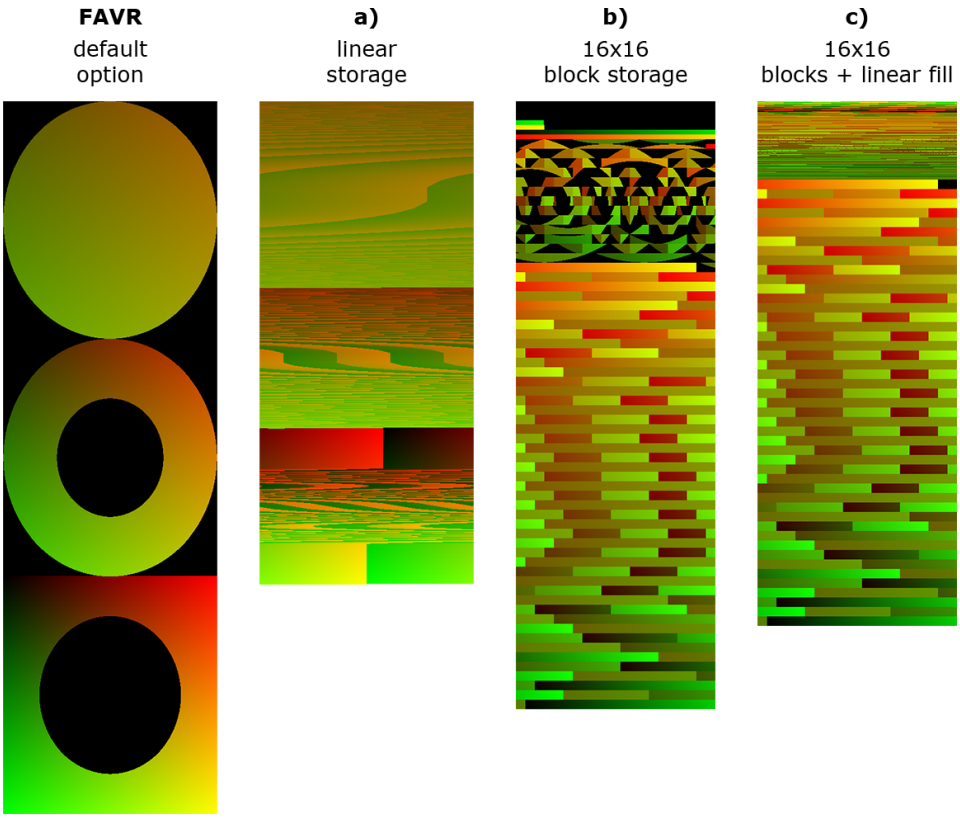


Fig. 5.10 Three alternative storage options for the subsampling textures were tested. FAVR represents the storage option currently used in the system. a) stores the fragment information in a linear list to minimize the size of the texture. b) maintains 16x16 blocks during storage to negate possible cache misses. c) combines the block approach with linear storage for edge areas were no whole blocks could be built.

chapter, these alternative patterns could not provide any uplift. As the number of fragments computed during ray traversal stayed the same for each tested texture, the loss in performance must come from other aspects of the processing pipeline. Based on the first investigations on the used hardware and implementation, it is to assume that cache misses on the driver level lead to more problematic results. However, these assumptions have to be investigated further in future work.

5.4.4 Conclusion of this Approach

Over the last sections, an acceleration technique for Direct Volume Rendering on HMD systems was presented and evaluated. Research in this field [165, 105, 181, 140] showed that high refresh rates and low latency significantly influence the appearance of cybersickness symptoms. Therefore, commodity-available HMD systems offer a higher refresh rate than most desktop monitors currently available,

reducing the chance of Cybersickness symptoms. HMDs also provide a higher resolution compared to most desktop monitors that have to be covered entirely by the virtual scene compared to traditional desktop solutions. Due to DVR being a fragment-bound algorithm, making it computationally expensive with increasing resolutions, it is challenging to design a VR experience utilizing Direct Volume Rendering.

The presented acceleration technique was designed to fit HMD characteristics making DVR a feasible option for interactive exploration. As the increasing number of fragments on HMDs is the limiting factor for most DVR solutions, the acceleration approach aims to reduce the number of processed rays without introducing a perceivable loss in image quality. For this purpose, the system reduces the number of fragments processed in the peripheral field of view by considering the narrow corridor users can focus on while using an HMD. A subsampling and a reconstruction stage based on precomputed textures were added to the rendering pipeline. Both stages were designed to be straightforward for other rendering systems to adapt. As the system's performance can vary during rendering, the system can switch between different reduction levels on the fly.

Evaluation of the acceleration technique indicates that the performance could be increased to a point where DVR becomes a feasible option without reducing the sampling rate of the ray-traversal step. While the image quality analysis reveals a loss in the overall quality of the presented image, preliminary tests have shown that the loss in image quality is not noticeable to the users in typical use cases. Furthermore, the preliminary tests indicated that the increased performance reduced the chance of Cybersickness symptoms.

5.5 Resolution Function Approach

A natural evolution of the acceleration technique presented in Section 5.4 is a shift in how the number of processed fragments is reduced while creating a more adjustable system. The previous method focused on utilizing pre-computed textures as a straightforward solution for resampling the ray-traversal input textures. While the method achieved a significant uplift in the rendering performance, it had to rely on the set of textures computed during the startup procedure. While adjustments during run-time could be made by switching between different reduction levels, these changes were often very conservative due to the limited set of textures.

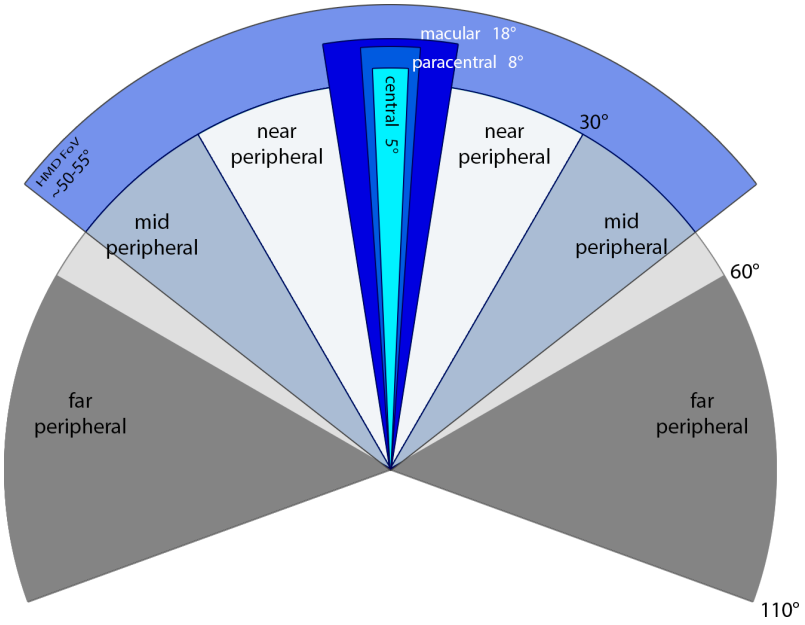


Fig. 5.11 The human visual system is subdivided into numerous fields ranging from the central vision to the far peripheral vision. Outgoing from the central vision, the perceived sharpness, and the color are reduced the further the angle increases. HMDs only cover a small portion of the visual field up to the mid-peripheral vision.

Over the following pages, another approach is presented for accelerating Direct Volume Rendering on HMDs. It is based on the previous system and its rendering pipeline but utilizes a different approach to reducing and reconstructing the frame-buffers. Compared to the limiting pre-computed textures of the previous system, this approach presents a highly flexible reduction function. This function can react with minor adjustments in the different resolution levels during run-time while further being future-proof for upcoming and announced HMD systems supporting eye-tracking hardware.

At its core, this approach follows the same fundamental idea as the previous. At first, the bounding geometry of the volumetric data is generated, which is then used during a preprocess to be resampled to new non-uniformly input textures for the ray-traversal step. Afterward, the result of the ray-traversal is collected and handed to the reconstruction stage before the final image is submitted to the HMD and displayed to the user. However, the pre-computed textures are replaced by a resampling function $f(t_u, t_v, e_x, e_y)$.

Furthermore, post submitting the frames to the HMD, an adjustment system is added to the rendering pipeline, which changes the individual resolution level depending on the current system load.

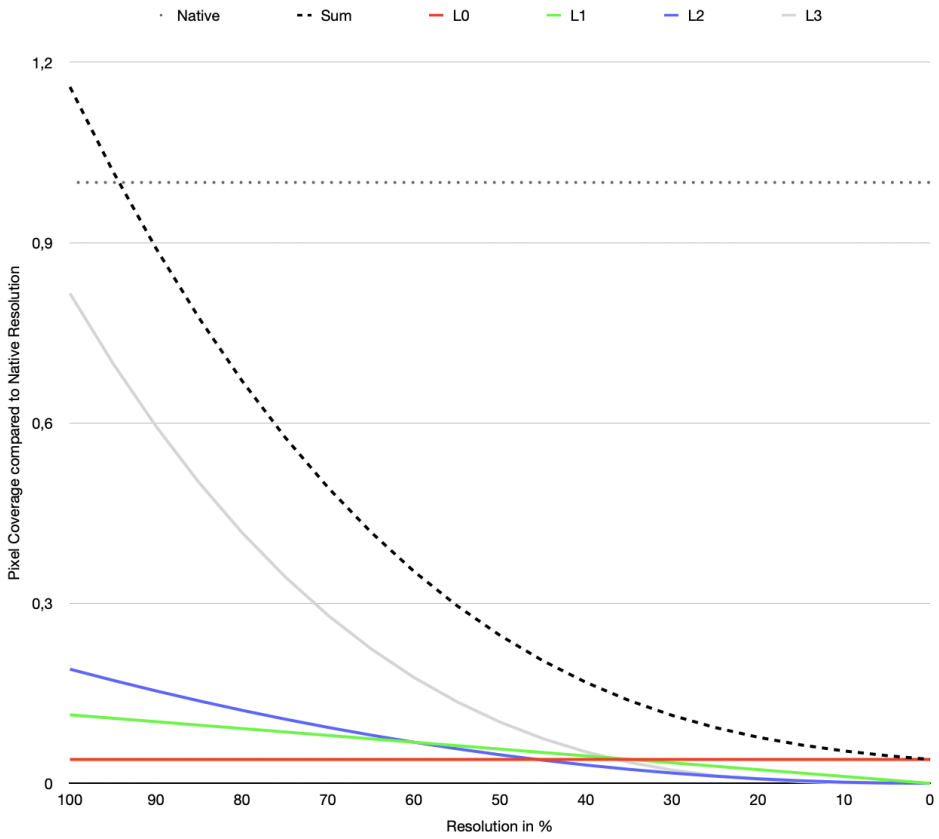


Fig. 5.12 Visualization of the fragment coverage. The visualization is based on the default resolution-distribution-function found in Figure 5.16 on the top. The upper line represents the native resolution of the HMD. The black dotted line the sum of covered fragments depending on the resolution scaling factor described in figure 5.13. On the default 50% scaling, the system renders roughly 25% of the total number of fragments the HMD frame-buffer covers.

5.5.1 Multiresolution Atlas Buffers

As the goal is still to improve the system's performance by reducing the number of primary rays traversed by the ray-caster, the system adapts the spatial ray frequency to the decreasing spatial perception in the peripheral regions of the user's field of view. While foveated rendering systems also follow this approach by utilizing high-speed eye-tracking hardware to locate the current gaze point, in the case of commodity available HMD setups, we do not need to consider the rapid eye movement of the users. Lenses built into HMDs are fixed and only provide a slight angle with the highest projected resolution. However, manufacturers are working on HMD setups supporting eye-tracking hardware, which is, to this point, still in its infancy.

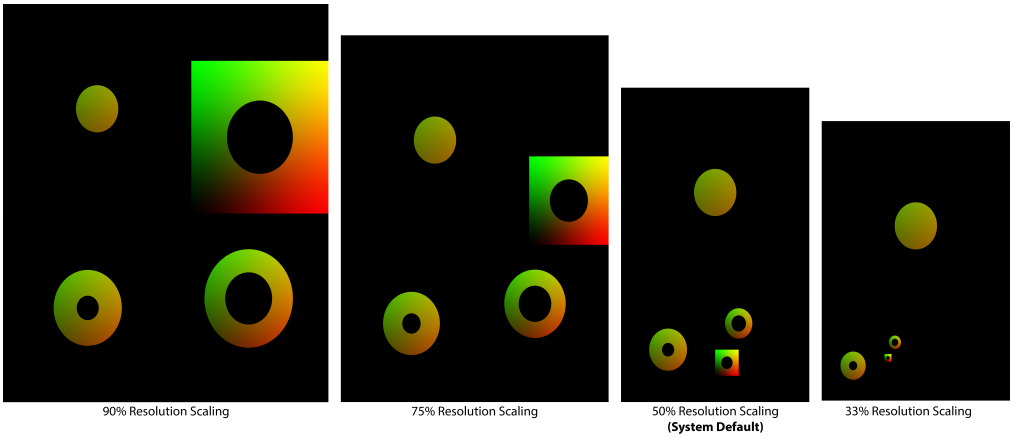


Fig. 5.13 The acceleration method supports arbitrary resolution scaling between subsequent resolution levels. Whereas Mip-Maps usually only keep a 50% scaling from level to level, this approach can utilize various scaling vectors depending on the hardware needs. For the displayed images in this work, the default 50% scaling was chosen.

The system uses discrete levels with progressively lower resolutions for specific image portions to reduce the number of rays. This layered approach is conceptually similar to MIP-Maps (see Figure 5.13). Compared to Mip-Maps, however, the system should support varying ratios between levels that are not necessarily a power of two, further reducing the noticeable difference between levels.

The system has to construct the set of resolution levels, forward the non-uniformly generated input textures to the traversal stage, collect the results from the ray-caster, and finally generate a consistent output image for the display in the HMD.

5.5.2 Implementation

Before going into detail about the implementation of this acceleration approach, it is beneficial first to understand the fundamental mapping between the HMD's frame buffer (s. Figure 5.14 left) and the Multiresolution Atlas Buffer (MAB) (s. Figure 5.14 right).

The MAB is partitioned into multiple regions, whereas each region corresponds to the entire frame buffer of HMD's native display resolution. This concept is similar to the levels of a MIP-map texture; however, in this approach, the difference between the various resolutions is not necessarily a fixed 4:1 ratio; instead, it is defined by a parameter set by the user.

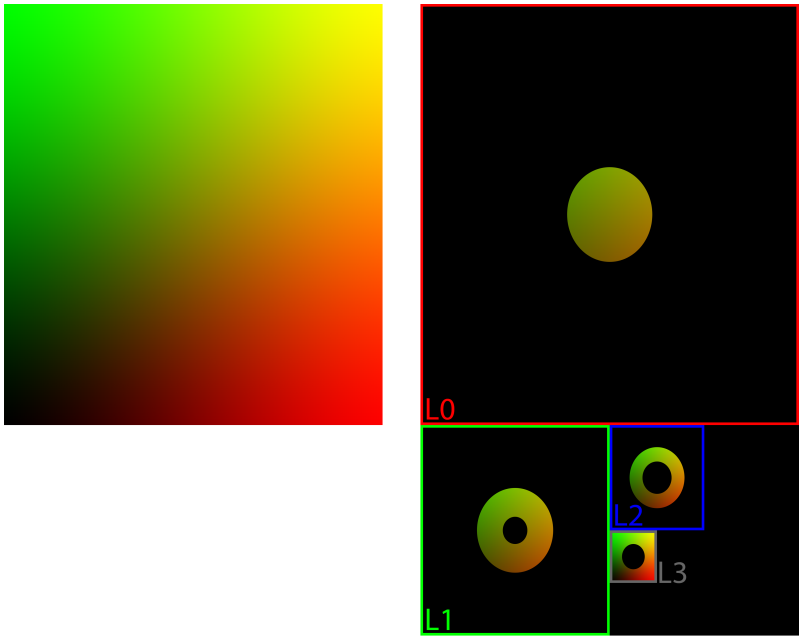


Fig. 5.14 On the Left: Screen-Space frame-buffer as it is used for the HMD. Right: The default MAB setup with 50% resolution scaling (similar to MIP-Maps) and the default resolution distribution. The different resolution levels are color-coded. The black areas in each level are discarded before the ray traversal.

A bi-directional mapping between the two spaces is mandatory to perform the two bracket steps mentioned in Section 5.3, i.e., the computation of multiresolution ray coordinates for the raycaster and the compositing of the multiple resolutions into a single consistent frame buffer image. The mandatory function assigns to each position in the HMD's frame buffer a set of corresponding positions for every resolution region of the MAB and a blending factor for each of these levels. The blending factor determines which fragments must be traveresd during raycasting and which can be discarded. This function can be reversed during compositing to prepare the final image for presentation. In this stage, the blending factors are used to create a smooth transition between neighboring resolution levels.

For example, a frame buffer fragment in the center of the gaze is mapped to a single MAB fragment in the full-resolution region as the blending factor for every other resolution level is set to 0. A fragment in the far peripheral is mapped to a single MAB fragment in a very low-resolution region. Frame buffer fragments between various resolution regions can be mapped to several MAB fragments and are composited according to each MAB-value's blending factors (see Figure 5.14 top-left).

The rendering pipeline is looked upon in reverse order to explain the process better. After the ray traversal is complete, the compositing is initiated by rendering a full-screen-primitive that covers the entire output frame buffer. For each generated fragment, the system has to perform a mapping from the target frame buffer to the MAB. This mapping is computed using two functions. The first function (f_a) takes as input the distance to the center of gaze, based on the screen-space coordinates of the fragment, and outputs a set of blending factors for the MAB, one blending factor for each resolution level inside the MAB. The sum of these blending factors is equal to 1. The second function (f_b) utilizes the screen-space coordinates of the fragment and outputs a set of corresponding coordinates in the MAB. Both functions have to be evaluated to perform the final compositing. For each non-zero entry in the blending-factor-set acquired by function (f_a), a look-up into the MAB is carried out (f_b) to obtain the stored color values. These values are blended into a single color for display using the corresponding blending value.

Before this compositing step, the raycasting is initiated by also rendering a full-screen-primitive. This primitive, however, covers the entire MAB. Hence, the rasterizer generates all fragments for all resolutions of the frame buffer. To select only those MAB fragments that contribute to the compositing, the inverse of the function mentioned above is computed and determines the blend factor for a given MAB fragment. If the value is zero, the fragment is discarded.

The system uses a third function (f_c) that takes as input the MAB coordinates and outputs two quantities to perform the inverse mapping —first, the resolution level of this fragment, and second the corresponding position in the HMD's frame buffer. The frame buffer position is used to determine the distance value for function f_a, where the resolution level is needed to select the correct blending factor returned by f_a.

Notice that the functions f_b and f_c are fixed for a given frame buffer resolution and depend on the connected hardware. These functions are pre-defined during start-up. On the other hand, the function f_a is adjusted during run-time to control the rendering performance vs. quality. Mapping more frame buffer pixels to lower resolutions will increase the rendering speed and is accomplished by shifting the lower resolutions toward the center of gaze (Figure 5.16 bottom). Mapping more frame buffer pixels to the native resolution at the center of the gaze will increase the image quality but reduce the system's performance (Figure 5.16 top).

In the current implementation, the function f_a is realized as a 1D lookup texture similar to a transfer-functions used in direct volume rendering (Figure 5.16). The

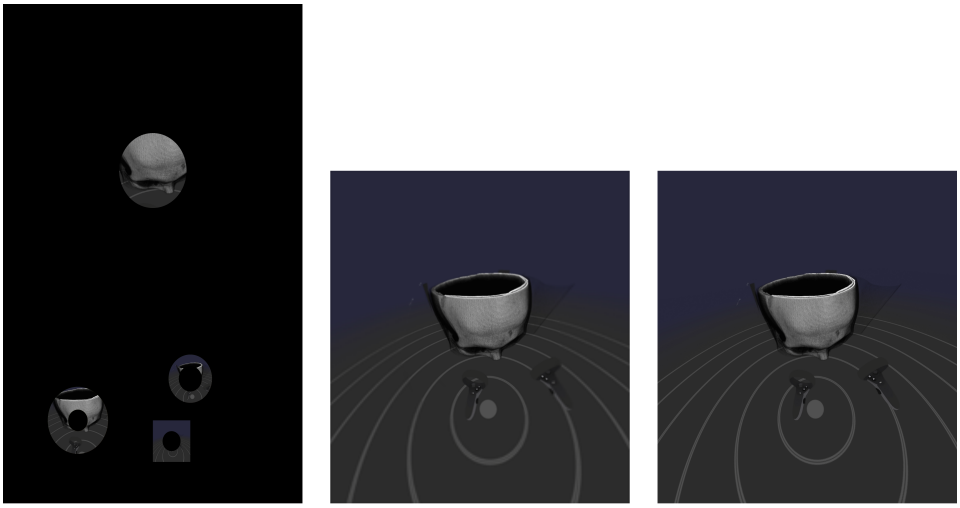


Fig. 5.15 From the left to the right: The result of the ray-caster rendered into the MAB. The full-resolution reconstruction that is submitted to the HMD. A native-resolution result as a comparison.

individual resolution levels of the MAB are mapped to the four color channels of the texture. The individual value of each channel describes the blending factor for the corresponding resolution level. To deliver unnoticeable blending, a smooth-step function is added between adjacent resolution levels (Figure 5.16).

5.5.3 Frame Rate Adaption

The actual frame rate of a DVR system depends on a large variety of parameters. Even small parametric changes can significantly impact the performance during run-time, i.e., changing the transfer function used to sample the data or modifying the view frustum. As described throughout this work, a frame rate below a certain threshold is not desirable for virtual environments. Even small but abrupt dips in the rendering performance can lead to nausea and other cybersickness symptoms.

Especially parameters like the camera position and viewing direction are highly variable when wearing an HMD. The user's real-time tracking can be seen as a benefit of VR systems, but it leads to system performance complications.

Adjusting the rendering parameters on the fly is necessary to maintain a consistently high update rate and to avoid cybersickness symptoms. The rendering method here provides a straightforward adjustment system to guarantee refresh rates above the hardware-defined threshold.

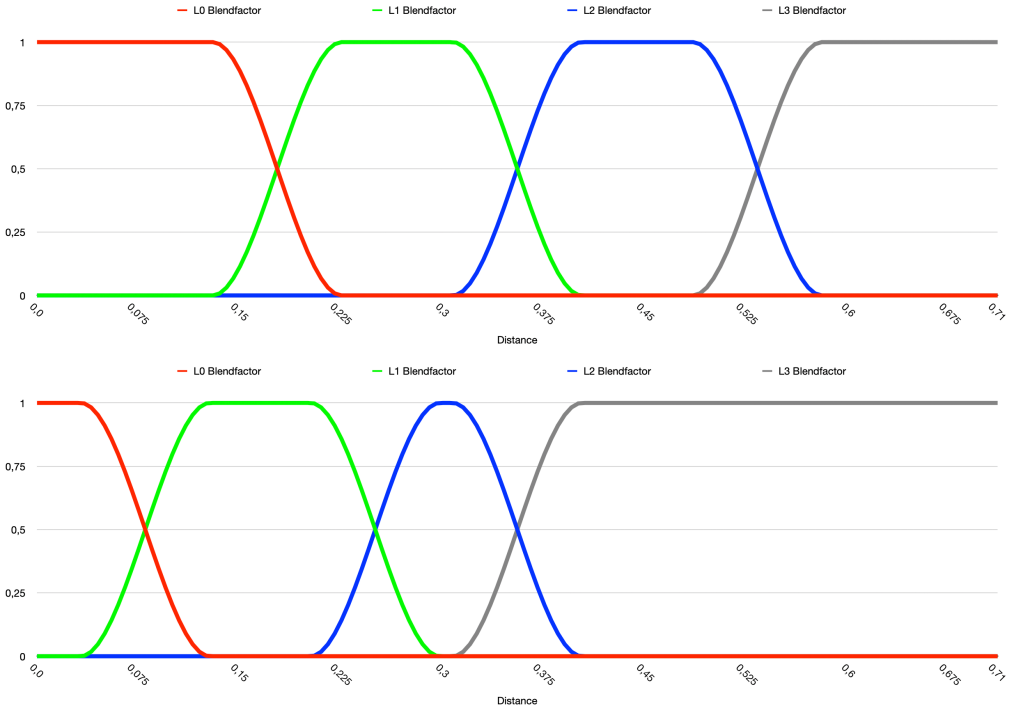


Fig. 5.16 Each line represents a specific resolution-level of the MAB (Red = L0, Green = L1, Blue = L3, gray = L4). The Y-Value gives the opacity value for a fragment at the computed distance from the gaze. Values equal to 0 get discarded. The top functions result in Figure 5.13 5.14. The top chart displays the default distribution used on start-up. The bottom chart displays a fallback function to maximize the performance if dynamic run-time adjustments are not supported.

By tracking the rendering performance and adjusting the distribution of the resolution levels on the fly, the system can maintain its performance before the critical threshold is reached.

The already established function f_a (Section 5.5.2), implemented as a 1D texture, is used in the opening brace as a fast look-up function to determine the corresponding resolution level of a texel and whether or not it has to be traversed by the ray-caster. The number of fragments corresponding to each resolution level can be adjusted by modifying the texture.

The system starts by equally distributing the pixels over every resolution level (Figure 5.16 (a)) and shifts the various levels according to the current system load. By tracking the frame timings of the previous finished frame, the system predicts the assumed load for the upcoming frame and adjusts the resolution distribution accordingly.

The system keeps track of the rendering time of previous n frames completed. In the first step, the system computes the gradient of the completion times of these finished frames. Afterward, the rendering time of the last finished frame and the computed gradient are used to adjust the different resolution levels found in the 1D texture. The previous frame's rendering time is compared against two thresholds—a lower threshold of a minimum acceptable refresh rate and an upper threshold of a sufficiently high refresh rate.

If the performance is below the threshold, the system shifts the resolution distribution to cause fewer rays to be emitted. If the refresh rate is above the higher threshold, the system increases the number of rays. The system considers the gradient of the window of n-frames to compute the increment or decrements in rays.

The current implementation targets a refresh rate of at least 90 Hz or 11.1 ms per frame. Therefore, the upper and lower threshold is set to 10 ms as the lower boundary and 9 ms as the upper boundary. If the system's rendering time is above 10 ms, the number of rays is lowered, while being below 9 ms, the number of rays is increased.

By evaluating the change in rendering time over the last n-frames, the system determines which anchor point between two resolution levels should be moved and how far it should be shifted in the corresponding direction. Minor variations lead to adjustments in the higher resolution levels, only slightly altering the resolution distribution, where high deviations primarily shift the lower resolutions to compensate in favor of the rendering performance (see Figure 5.17).

5.5.4 Results

Two crucial aspects are analyzed to evaluate to which extent the presented acceleration method can be applied to the field of Direct Volume Rendering. At first, the system's rendering performance is tested by comparing a state-of-the-art raycaster implementation without specific acceleration methods for the HMD system to the peripheral resolution adjustment approach presented in the previous sections. As the approach should present a hardware-independent and scalable solution for DVR on HMD systems, it is evaluated on different hardware setups with a range of computational performances. Besides benchmarking the system's rendering performance using a fixed sequence of pre-recorded interactions, preliminary tests were conducted to gain feedback about the perceived image quality and performance during real-world clinical operations.

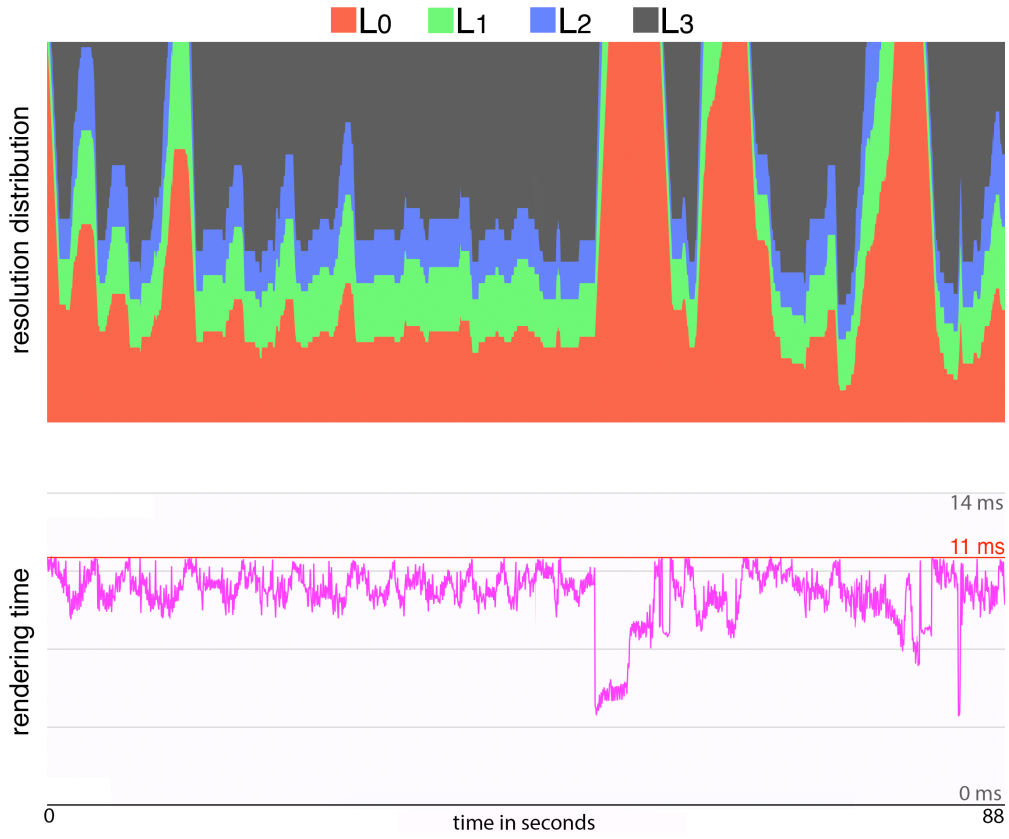


Fig. 5.17 Adjustment of the distribution function is made during run-time. The sequence displayed is the benchmark found in our supplementary material. The top displays the sizes of the different resolution levels and anchor points. The bottom displays the corresponding rendering-time and the 11 ms threshold.

Performance

As often mentioned throughout this work, high refresh rates are mandatory to reduce the possibility of Cybersickness symptoms appearing. Often a threshold of 90 Hz is mentioned as sufficient for most use cases. The previously established acceleration method introduced a multi-resolution rendering pipeline reducing the fragment density in the peripheral field of view while wearing an HMD. The resulting uplift in rendering performance can push the refresh rate beyond the desired 90 Hz threshold. This section presents the results of various benchmarks executed on different hardware setups and different data sets. The presented acceleration method is compared to native resolution approaches and demonstrates the increased rendering performance.

Two hardware setups were tested during the benchmarks to evaluate the method's scalability. As a high-end stand-alone workstation, a computer equipped with a

Ryzen 9 3900X, 32 Gb DDR4 3200 Mhz memory, and an Nvidia RTX 2070 as our graphics processing unit was selected. Such systems are widely accessible to the targeted field and represent the performance practitioners could expect during clinical use. As a second rendering system, a laptop was tested with a mobile Nvidia RTX 2060 with a power limit of 80 Watts. This system was chosen to verify that the system is also useable on a mobile setup and scalable to limited hardware. The VR system used during testing was the HTC Vive Pro, with a native display resolution of 2560 x 1440 pixels and a refresh rate of 90 Hz.

The automated benchmark scene was composed of a set of recorded interactions in the form of rotations, translations, scaling, and modifications in the transfer function. To further extend the testing, four rendering modes were evaluated. The first mode, the baseline, is a full-resolution ray-caster implementation utilizing empty-space skipping and early-ray termination. This mode represents a rendering setup without this chapter's opening and closing braces. The next mode tested is the default resolution level distribution (Figure 5.16) without any automatic adjustments and presents a uniform distribution of the resolutions across all levels. The third mode is a transfer function setup that aims for maximum performance and reduces the resolution much more rapidly than the previous setup. Finally, the system is tested with the automatic resolution adoption enabled. The minimum and average refresh rates on both hardware setups for all testes modes are evaluated.

Two data sets were selected representing state-of-the-art volumes obtained from modern medical scanners for the testing. The first is a CT scan with a resolution of 512 x 512 x 404 voxels, and the second is an MRI scan of 512 x 512 x 392 voxels. The sampling rate was adjusted to be sufficient enough to sample each voxel along the ray at least once to avoid undersampling the data.

The results of the benchmark sequence are displayed in Table 5.5. The results show how this approach can increase DVR performance on HMD setups. The baseline tests on the desktop and mobile/laptop rendering setup could not maintain refresh rates even close to the desired 90 Hz, with an average frame rate of 65 fps on the RTX 2070 and 50 fps on the mobile RTX 2060 across both data sets. Using the default MAB setup without the addition of the dynamic resolution adjustments, the system could maintain an average refresh rate above 90 Hz for both data sets on the RTX 2070, but only on one of them for the RTX 2060. While the average refresh rate increased, the 1% lowest frames are still below the targeted refresh rate of the HMD, which can already lead to nausea and other cybersickness symptoms.

The additional dynamic resolution distribution is added to the system in the last tests to keep the overall refresh rate above the set threshold of 90 Hz. With the complete acceleration method activated, even the mobile RTX 2060, with its limiting power draw, can maintain the necessary refresh rate. While the total resolution rendered on the mobile setup is lower than the desktop system, the loss in image quality was still not perceivable, as the preliminary tests with domain experts demonstrated.

		RTX 2070		RTX 2060m	
		1%	avg	1%	avg
MRI	baseline	23.7	51.1	18.5	37.3
	default MBA	78.8	105.2	65.1	84.9
	max MBA	193.2	208.1	143.0	185.4
	auto MBA	**90.3**	**108.2**	**90.1**	**91.2**
CT	baseline	37.6	79.0	26.8	62.6
	default MBA	80.8	132.9	77.4	97.1
	max MBA	196.1	213.7	148.1	174.2
	auto MBA	**92.9**	**141.6**	**90.9**	**98.8**

Table 5.5 The table displays the rendering system's average performance on two different data-sets and two different hardware-setups. In addition to the average frame-rate, we also display the 1% lowest frame-rates. The RTX 2070 performance is on a desktop system, while the RTX 2060 benchmarks are done on a mobile laptop with an 80 Watts limit of the GPU

Preliminary Tests

While the previous section focused on the objective performance increase achieved by applying the presented solution, this section focuses on the user feedback of the first preliminary tests. For these first preliminary tests, 17 participants were conducted. Each participant had prior experience with Virtual Reality and exploring volumetric data sets. During the tests, the participants could explore various data sets using interaction methods such as translation, scaling, rotation, or transfer-function editing within the virtual environment.

During these tests, two aspects of the system should be investigated:

1. Is the loss in image quality noticeable to the users while exploring the data?
2. Is the rendering performance below the threshold of 90 Hz more noticeable to the user than the change in the image quality?

Users could first explore data sets in a controlled environment to test these two aspects. Within this environment, the MAB setup was designed so that the refresh rate could always stay beyond 90 Hz. During exploration, the system altered the resolution distribution, further reducing the image quality in the peripheral vision. Based on the interviews after each session, none of the participants noticed any change in image quality during the exploration. These statements support the assumption that the rendering resolution can be reduced further in the peripheral vision in HMD scenarios than currently utilized.

More rendering load was put on the system to investigate the second aspect. Increasing the data set's size and the sampling rate requires more computations during ray traversal.

While in its default setup, the MAB, without utilizing the real-time adjustment of the resolution function, could maintain an average refresh rate above 90 Hz, on some occasions and transfer-function setups, the refresh rate could dip to $\tilde{6}0$ Hz. Once the dynamic resolution adaption is toggled on, the refresh rate could be caped to 90 Hz. At this point, we randomized the starting condition for each participant. Either the participants started with a fixed resolution distribution, including dips into low refresh rates, or they started with an adaptive resolution distribution but a consistent refresh rate. After 1 minute of free exploration, the participant switched to the other mode. After collecting the feedback from each participant, no one noticed that the system was changing the overall image quality on the fly. However, all of them noticed that during the fixed MAB setup, the system was less responsive, and at least two of our participants felt nausea after a short time and had to stop the test.

In conclusion, the preliminary tests support the assumption that fluctuating and low refresh rates are more noticeable to users and can rapidly cause Cybersickness symptoms. While it was assumed that some participants would notice the change in image quality, it was astonishing that none of the participants could notice the rendering approach.

5.5.5 Conclusion of this Approach

In conclusion, this section presented an acceleration method for Direct Volume Rendering on VR systems. The technique acts as a brace around preexisting ray-caster implementations by adding additional subsampling and reconstruction stages to the rendering pipeline. The system can provide an interactive exploration of

volumetric data sets at 90 Hz or more by reducing the overall number of fragments processed during the computationally expensive ray traversal.

By offering an adaptable approach for changes in the resolution distribution, the system can easily maintain the targeted refresh rate and avoid critical fluctuations in the frame, which could cause Cybersickness symptoms. The system's scalability also provides an excellent opportunity to reduce the hardware requirement for volume visualization on HMD setups. The tested mobile setup demonstrated that this approach could uplift the rendering performance on lower-end hardware for a high refresh rate VR experience.

Preliminary tests conducted with field experts helped to assess the assumption that the loss in image quality in peripheral vision is not noticeable in the VR context. The tests focused on the difference between consistent image quality and constant refresh rate. They showed that changes in the image quality due to changing resolution zones were not noticeable by the users, while rapid drops in the refresh rate could lead to nausea in a short time.

5.6 Comparison of Both Acceleration Methods

Both acceleration methods outlined in the previous sections aim to provide a fluent user experience while exploring immersive volumetric data sets in a virtual environment. Both techniques also exceed in this task and give options to be scalable to various systems. While in the core, both presented acceleration methods follow the same idea, they vary in execution, and each of them come with several drawback and advantages over the other. This section will outline some of the differences between both techniques and their implications.

The implementation of the subsampling and reconstruction stages is the most obvious difference between both acceleration approaches. Where the first method makes use of precomputed textures as lookup tables during run-time, the second technique uses a more dynamic approach by evaluating a distance function during every single frame. Leaving the flexibility of the second approach aside and just comparing two identical textures with the same distribution of fragments on each level, the actual size of the frame buffer is significantly larger compared to the first approach. The increased texture size of the second technique can be led back to the necessity that the entry input frame buffer must be covered on each resolution level. Depending on the distribution of the distance function across all resolution levels, each level could cover the entire original frame buffer at any given time. As a result,

the texture can not be packed as densely as the fixed precomputed textures of the first approach.

In comparison, the first approach precomputed the reduction textures and, therefore, can pack the textures to a closer fit. As a result, the number of fragment shader executions on the latter approach is far higher than on the first. While the raycaster used for both methods immediately discards every fragment that is not covered by the volumes bounding box, the invocation of the fragment shader itself could cost additional performance. However, comparing the single execution of the fragment shader, with an immediate discard call, to the costly ray-traversal loop makes the increased number of fragments carry no weight. Nonetheless, the increased size in the targeted fragment buffer could be limiting for older hardware. While the size of a single texture does not exceed the size of a modern GPU's memory by any means, this could shift quickly once two or three-generation old hardware is used. This problem could arise when combing the raycaster with growing data sets that could be visualized with a high enough frame rate using this approach, even on older hardware.

Another aspect that differentiates the two methods is the distant function introduced in the second approach. The first approach's precomputed textures do not allow run-time adjustments to the individual rendering zone. As a result, the different resolution distributions that can be interchanged during run-time have to be determined beforehand. While the generated distribution levels were not noticeable in the preliminary tests, the individual levels for each user could vary significantly. Furthermore, is the first technique less flexible in its performance adjustments. If the performance falls below the set threshold, a notable shift in resolution has to be introduced by switching to a lower distribution. On the other side, the distance function found in the second approach can be used to adjust each level individually on the fly. The fine adjustment achievable by this approach can result in a more consistent frame rate with fewer shifts in the full resolution.

While the dynamic adjustment in the second approach results in a more consistent and stable frame rate, it can also introduce a noticeable decrease in perceivable resolution. The resolution in the center could be decreased too far in the case that the rendered data set is too large to be visualized with a native resolution. If this is the case, the system has to shift the distance function to a point where no pixel would be covered with the native display resolution, hence introducing a lower resolution in the gaze area of the display. This result is not desirable for visualizing volumetric

data sets but has to be seen as a compromise to guarantee a high enough refresh rate to avoid cybersickness symptoms.

Regarding rendering performance, both acceleration methods can not be compared based on the numbers presented in Section 5.4.3 and Section 5.5.4. Each technique follows a different goal. The first technique aimed to provide a refresh rate that consistently exceeds the set threshold. In contrast, the second approach was designed to target the set refresh rate while maintaining as much space of the display in its full resolution. Therefore it is to conclude the individual performance of each system and whether the set goals were achieved or not.

5.7 Preliminary Conclusion and Outlook

This chapter presented two acceleration methods applicable to HMD systems for accelerating direct volume rendering algorithms on HMD devices. Each approach offers a significant gain in rendering performance while not being perceivable to the user in most cases.

Each approach has individual benefits and goals. Implementing both methods in most ray-caster solutions is straightforward as they are used as a pre- and post-processing stage of the ray-traversal stage. Additionally, the presented approaches are independent of any hardware specification. While GPU manufacturers have shown similar APIs for a general acceleration of the rendering pipeline using a varying number of fragments rendered across the screen, each of these APIs is bound to a specific set of hardware architecture and graphics cards.

Both techniques can deliver refresh rates more than sufficient for exploring volumetric data sets in immersive environments. In both cases, the preliminary tests have demonstrated that the reduction in spatial resolution in the peripheral vision is less noticeable in HMD scenarios. At the same time, low refresh rates can cause nausea within seconds. The second approach benefits from a dynamic resolution system using the texture-based distance function to maintain a consistent refresh rate. Furthermore, the advancements introduced in the latter technique allow for a fast adaption to upcoming HMDs supporting eye-tracking hardware.

While the setup is scaleable to various hardware setups and can be adjusted to maintain high refresh rates by reducing the image quality, it is to mention that in some configurations, the image quality has to be sacrificed even in the central field of vision.

The upcoming research should focus on optimizing the patterns between various resolution layers. The current state utilizes spherical shapes due to their simplicity and likeliness to the lenses found in HMDs. However, the human visual system does not perceive resolution equally on the horizontal and vertical axis. Furthermore, the preliminary tests should be extended to long-term studies analyzing various aspects of DVR within Virtual Reality. While first feedback indicates that the spatial relationships gained from an immersive VR visualization benefit practitioners in their clinical use, long-term studies and courses evaluating these use cases should be conducted in the future.

Chapter 6

Conclusion

Throughout this work, various research has been presented, introducing new visualization methods to the context of neurostimulation therapy. The presented research integrates the already available data acquired in daily clinical practice into new visualization systems that aid practitioners in several tedious tasks mandatory for DBS procedures to succeed. A significant focus of the research was on systems for real time visualization without disturbing already available workflows.

Typically, neurostimulation procedures can be split into three steps, starting with the preoperative assessment and planning, followed by surgery, including the implantation of macro electrodes into the patient's brain, resulting in the final stage, the postoperative care, including the adjustment of several treatment parameters. The topics found in this work are highly tailored toward the first two steps of neurostimulation therapy but could find their use in other medical domains. More, do the presented systems and solutions target the planning and surgery stages of DBS procedures.

The first step to improving neurostimulation procedures was made at an intraoperative visualization system. During the design process of the system and the current state of DBS procedures, it became clear that additional aiding software could be beneficial for practitioners in locating the correct brain tissue during surgery. An essential role in this location process is MER, which can record neural activity while inserting tiny microelectrodes into the patient's brain. While experienced practitioners can distinguish different brain tissues based on the recorded signal by the MER, it is expected that a degree of uncertainty remains even after years of experience. The developed system presented in Chapter 3 can transfer the raw MER data from the connected computer for more detailed processing. By applying low- and high-pass

filters to the system, surgeons can experience more filtered information with less noise. However, the most significant improvement to the decision-making process based on the MER data was achieved by transferring the filtered MER modality into a frequency domain of the neural activity. By color-coding the recorded intensity of different frequencies, surgeons could already decide at a glance what tissue they are currently penetrating. While the color-coded frequencies already improved the decision-making process of practitioners compared to the raw signals displayed on the MER device, the system went a step further. By utilizing a linear classifier, in the form of a perceptron network, the system could learn, with the help of classified expert data, which kind of signals correspond to the targeted STN. The system achieved high accuracy in STN classification, supporting the practitioners when necessary.

Besides displaying MER data, the system integrates volumetric CT and MRI data for interactive visualization of the electrode placement. By fusing the CT, MRI, and processed MER data into a single three-dimensional visualization, surgeons can see at all times what kind of tissue they are currently penetrating and cross-validate the recorded neural activity with the imaging data. This fused visualization of these three domains was not present to surgeons during the surgery prior to this research. While initially seen as a problem, the registration of CT and MRI data resulted in a GPU-accelerated method acceding commodity solutions in the clinical practice of its time. The registration of both data domains is a mandatory and time-critical step during the procedure. While in practice, this task is often done manually by surgeons as available software needs to be more accurate, the method integrated into the system could convince the participating practitioners.

Throughout the research on this topic, the presented solution was evaluated over multiple interventions and was found as a suitable addition to the current DBS procedure. The system provides more insight into the already available intraoperative MER data by including further processing and classification. The fused three-dimensional visualization lifts some of the cognitive load of surgeons as they can now see the current location of the lead, which aids them in crucial decision-making tasks.

While researching intraoperative visualization systems for DBS procedures, another exciting problem came across in the form of the planning phase for DBS surgeries. The insight into the planning phase opened the question of whether Virtual Reality could enhance certainty during the location process of the STN. Furthermore, the question arose if VR, and its spatial benefits, could improve the learning rate

of medical students in distinguishing different tissue types in a close-to real-world visualization. The immersive experience modern HMD systems can deliver would allow surgeons to investigate the patient's imaging data in a surrounding similar to the real world. Instead of viewing the three-dimensional imaging data on flat two-dimensional screens, surgeons could experience the spatial benefits of looking at the data from every angle in real stereoscopic 3D.

However, up to this point, Virtual Reality was mainly a technology in manufacturing industries or video games. To find the current state of VR in data visualization, a comprehensive, state-of-the-art literature review has been conducted by investigating several VR domains in the data and scientific visualization field. After conducting various research papers presented throughout the chapters of this dissertation and distinguishing the difference between traditional desktop visualization and VR, immersive environments using an HMD system could enhance the insight into data not achievable by traditional desktop monitors.

The current state of data visualization in immersive environments indicates that VR systems can provide a better spatial awareness of the data and provide additional, more intuitive interaction metaphors with the data. Furthermore, the research reviewed throughout Chapter 4 indicates that user satisfaction is significantly higher with VR data visualization software compared to traditional desktop software. However, immersive data visualization comes with several constraints to the system and the hurdle to finding users willing to adapt from traditional WIMP interfaces to more natural VR interfaces. As the corresponding chapter pointed out, most immersive data exploration systems are specifically designed for the data to be visualized to support the researchers of field experts. While generating more general visualization tools is desirable, the past has shown that specially tailored-solutions result in more success.

As for the visualization of medical data in VR, in particular scalar volumetric data, published systems tend to fall back to visualization techniques that are less computationally expensive. To maintain the most insight into the scalar data, DVR has proven to be a suppered solution compared to IVR. However, DVR is a computationally expensive rendering technique, while most VR solutions utilize IVR and extract surface areas from the data to be visualized interactively in VR. Alternatively, some systems tried to provide high-refresh-rate DVR on HMDs but had to limit the volume's size or the ray-traversals quality to upkeep the system's performance.

The research in this work focused on solutions that could accelerate DVR on HMD systems for an immersive preoperative planning and teaching scenario. As

the medical imaging data acquired from CT or MRI scanners represent spatial scalar data, practitioners can benefit from the stereoscopic visualization of HMD systems and examine the data much more closely than possible on desktop systems. However, the demand for DVR s algorithm to be executed in real time with sufficient refresh rates for VR made it challenging to combine both techniques. Another crucial point to consider was how DVR scales with increasing display resolutions. As a fragment-bound algorithm, the ever-increasing resolution of HMDs proposed a challenge for the future of DVR on HMDs.

The resulting solutions found in this work present acceleration methods for DVR on HMD systems that utilize crucial aspects of the human visual system and restrictions of currently state-of-the-art HMDs to achieve VR experience practitioners had not experienced before when it comes to the immersive investigation of medical imaging data. The main concept used is the reduction of processed fragments in the user's peripheral vision while wearing an HMD. As the perceived resolution in the peripheral and far peripheral vision of the human decreases rapidly, it is not necessary to maintain the full resolution of the built-in displays.

Besides objectively evaluating the acceleration methods using rendering benchmarks or metrics for the evaluation of image quality, they were tested in close relationships with field experts in preliminary tests. As expected, the increased performance of the system could reduce the appearance of cybersickness symptoms, which should not occur in any medical facility for doctors on duty. Furthermore, the reduced resolution was not noticeable to the users. Even while cross-validating the perceived quality by turning the system on and off, the practitioners did not notice the lack of image quality in the peripheral field while immediately noticing the lower response time of the system. Even small groups of medical students could test the system and report a better understanding of the relation between different structures found in the human body.

As a conclusion to the results of this dissertation, the presented work could provide benefits to two of three stages of neurostimulation therapy. The placement process of DBS electrodes could be enhanced by including interactive visualizations of patients' imaging data during the surgery and further establishing a processing and classification pipeline of the MER data that aids surgeons during their crucial tasks. These newly introduced intraoperative visualizations lift some of the mental load of practitioners and remove some of the uncertainty that doctors had during this step. The deep dive into state-of-the-art data visualizations for immersive environments led to the ideas and concepts that resulted in unprecedented VR experiences for

practitioners. Acceleration methods for DVR on HMDs were developed that not only aid experienced surgeons by presenting fully immersive visualization of familiar CT and MRI data but also help students learn anatomical structures much more naturally than textbooks.

6.1 Discussion

The presented topics in this work propose promising results regarding supportive systems for neurostimulation therapy. The visualizations integrated into the daily workflow aid practitioners in various tasks. However, some of the outlined results can be discussed regarding their applicability and impact on neurostimulation therapy.

The intraoperative visualization and classification system presented in chapter 3 utilizes one of the simplest forms of neural networks. The perceptron network in this approach is based on a linear function to distinguish between two classes of data, making it a binary classifier. While at the beginning of this research, it was expected that recorded tissue activity could be too complex for a binary classifier, the results after the first training sessions were promising. The network was trained with data collected over multiple months and could achieve a stable classification rate. As perceptrons use a supervised learning approach, the resulting network tends to be overfitted towards the personal classification preferences of the participating practitioners. To counteract this problem, an increased number of practitioners would benefit the learning procedure of the neural network. In addition, perceptrons can only classify two classes of data, making it necessary to train new networks to classify other tissue types.

The recorded tissue activity is transformed into its frequency domain and later visualized using color mapping based on the individual signal strength of each frequency. The investigated frequency range of 250 Hz to 1000 Hz is not proven to be the only frequency range viable for this approach. It was selected due to the personal feedback of practitioners already familiar with other DBS and MER devices. While not clearly visible to the human eye on a display, other frequencies could provide more detailed information to the classification approach. The visualization of the MER data combined with the preoperative CT and MRI data provides a well-received feedback layer to the practitioners. It helps practitioners cross-validate the intraoperative findings with the spatial imaging information generated prior to the intervention. However, the fused visualization of three data domains can result in occlusion problems making it harder to find an optimal few on the target tissue.

The comprehensive literature review on VR data visualization was initially tailored toward scientific visualization topics in VR. However, most of the literature found on the topic of data visualization in immersive environments was found to be tailored towards information visualization. Typical scientific visualization topics are still a niche topic in this application field, whether due to being computationally expensive or other reason that is open for debate. Nevertheless, the input found in data visualization in virtual environments provided enough information to conduct that volume visualization in VR can significantly benefit specific use cases.

It was necessary to introduce acceleration methods directly tailored toward these systems to provide a Virtual Reality experience not ending in Cybersickness symptoms for direct volume rendering on HMD devices. The spatial awareness created due to the three-dimensional stereoscopic visualization helped during preliminary tests to convey various tissue structures to medical students. The overall rendering pipeline consists of a pre and post-processing stage, which can be added to mostly every direct volume rendering system without much effort. The second approach is based on a flexible distance function and can be adjusted on the fly to maintain the desired refresh rate. However, the implemented approach only utilizes the distance to a set point to determine the internal resolution. The human visual system does not scale equally on the horizontal and vertical axis. Therefore, different functions could provide a better solution with a closer fit to the actual human resolution distribution.

6.2 Future Work

As expected, several aspects of the presented topics are interesting for future work.

For the intraoperative visualization system, it is self-evident that the classification network should be enhanced for future versions. The classification could be extended to multiple types of tissue using a more complex deep neural network. The current classifier is only practical for DBS procedures, other types of neurostimulation therapy target other tissues not covered by the current network. Even if other networks are not integrated into the system, other perceptrons for other tissues should be integrated and weighted against each other to determine the tissue with the highest likelihood. However, therefore it would be mandatory for practitioners to provide additional supervised data for other tissue types. During this research, only supervised data for the subthalamic nucleus was provided; hence the approach to detect different types of tissues could not be followed. It is open for future research

to determine if other neural networks would result in better classification and whether the current perceptron keeps its reliability once it is compared to other tissues.

Another aspect that could be further enhanced is the three-dimensional visualization of the fused data domains. The current visualization already provides additional insight into the data and lifts a significant portion of the mental processing during the procedure; an operator must often manually navigate the camera due to occlusion problematics. Since the practitioners themself can not interact with the system while being sterilized, requiring support during the exploration. An automated approach would benefit practitioners during this process. First tests with automated camera locations have been integrated into the system but often do not satisfy the practitioner's needs. As an alternative, hands-free interaction methods, such as gesture control, could be integrated into the system.

Additional data domains such as Diffuse Tensor Imaging (DTI) could be integrated into the system to enhance the visualization further. Also, the visualization of the electrical impulse could be integrated into the visualization simulating the impact once the final electrode is placed.

Talking about the visualization of volumetric data sets in VR, several aspects could be the focus of future research. As the literature review suggested, VR software tailored for specific use cases can be a suitable solution for the given problem. It is still open to how far volumetric visualization can benefit planning and teaching in the medical context. Only limited tests could be conducted during the research and ongoning the Covid-19 pandemic. The preliminary tests for the presented acceleration methods were designed to evaluate the systems regarding rendering performance and visual perceptibility of the reduction patterns. During these tests, interviews with practitioners and students indicated that volumetric visualization in VR can provide insight not achievable on currently available desktop setups or traditional books. However, additional long-term studies are mandatory to evaluate this aspect of the system in the form of teaching classes for medical students or real-world procedure planning.

Several features could be interesting for upcoming researchers for the system itself and its acceleration approach. The current function-based acceleration method already considers the focal point as a parameter. While currently available HMDs do not utilize eye-tracking capabilities, multiple manufacturers have stated to release eye-tracking capable HMDs to the consumer market over the upcoming years. As outlined in chapter 5.2, eye-tracking capable systems on the desktop require even higher refresh rates than HMDs to compensate for rapid eye movement. It will be

interesting to see how eye-tracking is solved for HMDs and how volume visualization can utilize this feature.

Based on the idea of eye-tracking and the implemented distance function for the different resolution levels, it has to be mentioned that the underlying function should be altered in the future. The current solution reduces the horizontal and vertical axis resolution by the same rate. However, the human visual system maintains a higher resolution for longer on the horizontal axis than on the vertical axis. This property would indicate that the resolution in vertical space could be reduced even faster than currently implemented for the presented reduction function.

A vital aspect of the acceleration method is its hardware independency. Over the years, GPU hardware manufacturers have developed proprietary solutions to reduce the number of processed fragments on a driver level. However, these features are limited to specific hardware architectures and are not compatible with different GPU vendors. Nevertheless, it can be assumed that these driver-level features should provide similar or even better performance to the general-purpose solution presented in this dissertation.

Beyond the topics found in this dissertation, other visualization topics could be interesting for neurostimulation procedures. As mentioned at the beginning of this dissertation, neurostimulation procedures consist of an assessment and preoperative planning phase, intervention, and post-operative care. The research topics in this work mainly focus on the planning and intervention stages. While visualization systems for post-operative care have been developed and evaluated over the recent time [303], it would be interesting to utilize the MER data and visualization within those systems. A possible virtual reality tool kit for post-operative care could also be considered. This VR tool kit can provide several tasks to the patients while setting up the final treatment parameters.

6.3 Closing Thoughts

As a closing thought to the research found in this work, it is to say that the results presented in this dissertation indicated a positive impact on the daily tasks of practitioners of neurostimulation therapy. Each of the presented systems aids surgeons in their daily tasks and provides insight that was not presented to this point.

By making new visualization domains accessible to practitioners, such as virtual reality and spatial volume visualization, not only experienced surgeons gain new knowledge about the available data, but also medical students can benefit from

this approach during training. The additional knowledge gained by a stereoscopic three-dimensional visualization helps to relieve some of the mental strain doctors have to go through during the planning and execution of these kinds of interventions.

The intraoperative system provides an additional feedback channel for practitioners during the location procedure for DBS procedures. The presented visualization and classification reduce the uncertainty present during these crucial procedures and enhance the current state of MER-supported electrode placement. While practitioners primarily relied on auditive feedback during the placement process up to this point, the supportive system introduced imaging data and processing to the oval that was absent. Based on the feedback provided by skilled experts, the benefits of intraoperative visualization and classification release much stress to them and practitioners in training.

As a closing statement, it is up to say that the results found in this work prove a promising future for visualization in neurostimulation therapy and can be seen as groundwork for future research topics that can build up on the presented research.

Acronyms

CAVE Cave Automatic Virtual Environment.

CPU Central Processing Unit.

CRT Cathode-ray tube.

CT Computed Tomography.

DBS Deep Brain Stimulation.

DoF Degree of Freedomt.

DVR Direct Volume Rendering.

FPS Frames per Second.

GPi Globus Pallidus.

GPU Graphics Processing Unit.

HMD Head-Mounted Display.

IVR Indirect Volume Rendering.

MER Micro-Electrode Recording.

MRI Magnetic Resonance Imaging.

NMR Nuclear Magnetic Resonance.

PD Parkinson's disease.

PSNR Peak Signal-to-Noise Ratio.

STN Subthalamic Nucleus.

TF Transfer Function.

VR Virtual Reality.

List of figures

List of tables

References

[1] (2006). *Proc. IEEE Symposium on 3D User Interfaces, 3DUI 2006, Alexandria, Virginia, USA, 25-26 March, 2006*. IEEE Computer Society.

[2] Aamodt, A., Kvistad, K. A., Andersen, E., Lund-Larsen, J., Eine, J., Benum, P., and Husby, O. S. (1999). Determination of the hounsfield value for ct-based design of custom femoral stems. *The Journal of Bone and Joint Surgery. British volume*, 81(1):143–147.

[3] Achenbach, S., Ropers, D., Kuettner, A., Flohr, T., Ohnesorge, B., Bruder, H., Theessen, H., Karakaya, M., Daniel, W. G., Bautz, W., et al. (2006). Contrast-enhanced coronary artery visualization by dual-source computed tomography—initial experience. *European journal of radiology*, 57(3):331–335.

[4] Ackoff, R. L. (1989). From data to wisdom. *Journal of applied systems analysis*, 16(1):3–9.

[5] Adhanom, I. B., Griffin, N. N., MacNeilage, P., and Folmer, E. (2020). The effect of a foveated field-of-view restrictor on vr sickness. In *2020 IEEE conference on virtual reality and 3D user interfaces (VR)*, pages 645–652. IEEE.

[6] Agrawal, R., Kadadi, A., Dai, X., and Andres, F. (2015). Challenges and opportunities with big data visualization. In Chbeir, R., Manolopoulos, Y., Mammana, V. P., Modena, E. A., Traina, A., Salviano, O., Badr, Y., and Andres, F., editors, *Proceedings of the 7th International Conference on Management of computational and collective intElligence in Digital EcoSystems*, pages 169–173, New York, NY, USA. ACM.

[7] Ahmed A, E. P., editor (2005). *Automatic camera path generation for graph navigation in 3D*. Citeseer.

[8] Aitsiselmi, Y. and Holliman, N. S. (2009). Using mental rotation to evaluate the benefits of stereoscopic displays. In *IS&T/SPIE Electronic Imaging*.

[9] Akeley, K. (1989). The silicon graphics 4d/240gtx superworkstation. *IEEE computer graphics and applications*, 9(4):71–83.

[10] Ali, S. M., Gupta, N., Nayak, G. K., and Lenka, R. K. (2016). Big data visualization: Tools and challenges. In *2016 2nd International Conference on Contemporary Computing and Informatics (IC3I)*, pages 656–660. IEEE.

[11] Allcoat, D. and von Mühlenen, A. (2018). Learning in virtual reality: Effects on performance, emotion and engagement. *Research in Learning Technology*, 26.

[12] Alliance Studio (2015). Exploring virtual reality data visualization with gear vr. https://www.alliancestudio.in/blog/exploring-virtual-reality-data-visualization-gear-vr/.

[13] Amos, E. (2019). *The game console a photographic history from Atari to xbox*. No Starch Press.

[14] Andersen, B. J. H., Davis, A. T. A., Weber, G., and Wunsche, B. C. (2019). Immersion or diversion: Does virtual reality make data visualisation more effective? In *2019 International Conference on Electronics, Information, and Communication (ICEIC)*, pages 1–7. IEEE.

[15] Ang, S. and Quarles, J. (2022). You're in for a bumpy ride! uneven terrain increases cybersickness while navigating with head mounted displays. In *2022 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pages 428–435. IEEE.

[16] Anupam, V., Dar, S., Leibfried, T., and Petajan, E. (1995). Research report. dataspace: 3-d visualizations of large databases. In *Proceedings of Visualization 1995 Conference*, pages 82–88. IEEE Comput. Soc. Press.

[17] Azzag, H., Picarougne, F., Guinot, C., and Venturini, G. (2006). Vrminer. In Darmont, J. and Boussaid, O., editors, *Processing and Managing Complex Data for Decision Support*, pages 318–339. IGI Global.

[18] Bajwa, R. J., de Lotbinière, A. J., King, R. A., Jabbari, B., Quatrano, S., Kunze, K., Scahill, L., and Leckman, J. F. (2007). Deep brain stimulation in tourette's syndrome. *Movement disorders: official journal of the Movement Disorder Society*, 22(9):1346–1350.

[19] Ball, R. and North, C., editors (2005). *Effects of tiled high-resolution display on basic visualization and navigation tasks*.

[20] Bamodu, O. and Ye, X. M. (2013). Virtual reality and virtual reality system components. In *Advanced materials research*, volume 765, pages 1169–1172. Trans Tech Publ.

[21] Banik, A. and Bandyopadhyay, S. (2016). Big data - a review on analysing 3vs. *Journal of Scientific and Engineering Research*, (3(1)).

[22] Basting, O., Fuhrmann, A., and Grünvogel, S. M. (2017). The effectiveness of changing the field of view in a hmd on the perceived self-motion. In *2017 IEEE symposium on 3D user interfaces (3DUI)*, pages 225–226. IEEE.

[23] Bayyari, A. and Tudoreanu, M. E. (2006). The impact of immersive virtual reality displays on the understanding of data visualization. In Slater, M., Kitamura, Y., Tal, A., Amditis, A., and Chrysanthou, Y., editors, *Proceedings of the ACM symposium on Virtual reality software and technology - VRST '06*, page 368. ACM Press.

[24] Bektaş, K., Çöltekin, A., Krüger, J., Duchowski, A. T., and Fabrikant, S. I. (2019). Geogcd: Improved visual search via gaze-contingent display. In *Proceedings of the 11th ACM Symposium on Eye Tracking Research & Applications*, ETRA '19, New York, NY, USA. Association for Computing Machinery.

[25] Bellinger, G., Castro, D., and Mills, A. (2004). Data, information, knowledge, and wisdom.

[26] Benabid, A. L., Krack, P. P., Benazzouz, A., Limousin, P., Koudsie, A., and Pollak, P. (2000). Deep brain stimulation of the subthalamic nucleus for parkinson's disease: methodologic aspects and clinical criteria. *Neurology*, 55:S40–S44.

[27] Benabid, A. L., Pollak, P., Hoffmann, D., Gervason, C., Hommel, M., Perret, J. E., de Rougemont, J., and Gao, D. M. (1991). Long-term suppression of tremor by chronic stimulation of the ventral intermediate thalamic nucleus. 337:403–406.

[28] Benabid, A. L., Pollak, P., Louveau, A., Henry, S., and de Roúgemont, J. (1987). Combined (thalamotomy and stimulation) stereotactic surgery of the vim thalamic nucleus for bilateral parkinson disease. *Applied neurophysiology*, 50 1-6:344–6.

[29] Berger, A. (2002). How does it work?: Magnetic resonance imaging. *BMJ: British Medical Journal*, 324(7328):35.

[30] Berkman, M. I. (2018). *History of Virtual Reality*, pages 1–9. Springer International Publishing, Cham.

[31] Berkman, M. I. and Akan, E. (2019). Presence and immersion in virtual reality. In Lee, N., editor, *Encyclopedia of Computer Graphics and Games*, pages 1–10. Springer International Publishing.

[32] Bimber, O., Encarnação, L., and Stork, A. (2000). A multi-layered architecture for sketch-based interaction within virtual environments. *Computers & Graphics*, 24(6):851–867.

[33] Biocca, F. (1992). Will simulation sickness slow down the diffusion of virtual environment technology? *Presence: Teleoperators & Virtual Environments*, 1:334–343.

[34] Bishop, I. D., Pettit, C. J., Sheth, F., and Sharma, S. (2013). Evaluation of data visualisation options for land-use policy and decision making in response to climate change. *Environment and Planning B: Planning and Design*, 40(2):213–233.

[35] Bittar, R. G., Kar-Purkayastha, I., Owen, S. L., Bear, R. E., Green, A., Wang, S., and Aziz, T. Z. (2005). Deep brain stimulation for pain relief: A meta-analysis. 12:515–519.

[36] Bleisch, S. and Nebiker, S. (2008). Connected 2d and 3d visualizations for the interactive exploration of spatial information. In *Proc. of 21th ISPRS Congress, Beijing, China. No. 1999. 2008.*, pages 1037–1042.

[37] Blinn, J. F. (1977). Models of light reflection for computer synthesized pictures. In *Proceedings of the 4th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '77, page 192–198, New York, NY, USA. Association for Computing Machinery.

[38] Borkin, M. A., Vo, A. A., Bylinskii, Z., Isola, P., Sunkavalli, S., Oliva, A., and Pfister, H. (2013). What makes a visualization memorable? *IEEE transactions on visualization and computer graphics*, 19(12):2306–2315.

[39] Bowman, D. A., Johnson, D. B., and Hodges, L. F. (2001). Testbed evaluation of virtual environment interaction techniques. *Presence*, 10(1):75–95.

[40] Bowman, D. A. and McMahan, R. P. (2007). Virtual reality: How much immersion is enough? *Computer*, 40(7):36–43.

[41] Bozgeyikli, E., Raij, A., Katkoori, S., and Dubey, R. (2016). Point & teleport locomotion technique for virtual reality. In *Proceedings of the 2016 annual symposium on computer-human interaction in play*, pages 205–216.

[42] Brandes, U. and Wagner, D. (2004). Analysis and visualization of social networks. In Jünger, M. and Mutzel, P., editors, *Graph drawing software*, pages 321–340. Springer.

[43] Breit, S., Schulz, J. B., and Benabid, A.-L. (2004). Deep brain stimulation. *Cell and Tissue Research*, 318(1):275–288.

[44] Brock, L. G., Coombs, J. S., and Eccles, J. C. (1952). The recording of potentials from motoneurones with an intracellular electrode. *The Journal of Physiology*, 117(4):431–460.

[45] Brooks, F. P. (1999). What's real about virtual reality? *IEEE computer graphics and applications*, 19(6):16–27.

[46] Broucke, S. V. and Deligiannis, N. (102019). Visualization of real-time heterogeneous smart city data using virtual reality. In *2019 IEEE International Smart Cities Conference (ISC2)*, pages 685–690. IEEE.

[47] Brunhart-Lupo, N., Bush, B. W., Gruchalla, K., and Smith, S. (2016). Simulation exploration through immersive parallel planes. In *2016 Workshop on Immersive Analytics (IA)*, pages 19–24. IEEE.

[48] Bruno, F., Bruno, S., De Sensi, G., Luchi, M.-L., Mancuso, S., and Muzzupappa, M. (2010). From 3d reconstruction to virtual reality: A complete methodology for digital archaeological exhibition. *Journal of Cultural Heritage*, 11(1):42–49.

[49] Bryson, S. (1996). Virtual reality in scientific visualization. *Communications of the ACM*, 39(5):62–71.

[50] Budhiraja, P., Miller, M. R., Modi, A. K., and Forsyth, D. (2017). Rotation blurring: use of artificial blurring to reduce cybersickness in virtual reality first person shooters. *arXiv preprint arXiv:1710.02599*.

[51] Busch, M., Lorenz, M., Tscheligi, M., Hochleitner, C., and Schulz, T. (2017). Being there again – presence in real and virtual environments and its relation to usability and user experience using a mobile navigation task. *International Journal of Human-Computer Studies*, 101:76–87.

[52] Butcher, P. W. S. and Ritsos, P. D. (2017). Building immersive data visualizations for the web. In *2017 International Conference on Cyberworlds (CW)*, pages 142–145.

[53] Buzug, T. M. (2011). *Computed Tomography*, pages 311–342. Springer Berlin Heidelberg, Berlin, Heidelberg.

[54] Cairo, A. (2013). *The functional art: An introduction to information graphics and visualization*. New Riders.

[55] Catmull, E. (1972). A system for computer generated movies. page 422.

[56] Catmull, E. E. (1974a). *A Subdivision Algorithm for Computer Display of Curved Surfaces.* PhD thesis. AAI7504786.

[57] Catmull, E. E. (1974b). *A Subdivision Algorithm for Computer Display of Curved Surfaces.* PhD thesis. AAI7504786.

[58] Catmull, E. E. (1974c). *A subdivision algorithm for computer display of curved surfaces*. The University of Utah.

[59] Chawla, G., Bamal, S., and Khatana, R. (2018). Big data analytics for data visualization: Review of techniques. *International Journal of Computer Applications*, 182(21):37–40.

[60] Chen, M., Ebert, D., Hagen, H., Laramee, R. S., van Liere, R., Ma, K.-L., Ribarsky, W., Scheuermann, G., and Silver, D. (2009). Data, information, and knowledge in visualization. *IEEE computer graphics and applications*, 29(1):12–19.

[61] Chen, P., Hu, C., Ding, W., Lynn, H., and Simon, Y. (2003). Icon-based visualization of large high-dimensional datasets. In *Third IEEE International Conference on Data Mining*, pages 505–508. IEEE Comput. Soc.

[62] Chen, X., Self, J. Z., House, L., and North, C. (2016). Be the data: A new approach for lmmersive analytics. In *2016 Workshop on Immersive Analytics (IA)*, pages 32–37. IEEE.

[63] Chheang, V., Apilla, V., Saalfeld, P., Boedecker, C., Huber, T., Huettl, F., Lang, H., Preim, B., and Hansen, C. (2021). Collaborative VR for Liver Surgery Planning using Wearable Data Gloves: An Interactive Demonstration. In *Proc. of IEEE Conference on Virtual Reality (IEEE VR)*, Lisbon, Portugal.

[64] Cho, I., Dou, W., Wartell, Z., Ribarsky, W., and Wang, X. (2012). Evaluating depth perception of volumetric data in semi-immersive vr. In Tortora, G., Levialdi, S., and Tucci, M., editors, *Proceedings of the International Working Conference on Advanced Visual Interfaces - AVI '12*, page 266. ACM Press.

[65] Cicalò, E. and Valentino, M. (2019). Mapping and visualisation on of health data.

[66] Clark, A. and Moodley, D. (2016). A system for a hand gesture-manipulated virtual reality environment. In *Proceedings of the Annual Conference of the South African Institute of Computer Scientists and Information Technologists*, SAICSIT '16. Association for Computing Machinery.

[67] Cleve, J. and Lämmel, U. (2020). Grundlagen des data mining. In Cleve, J. and Lämmel, U., editors, *Data Mining*, pages 41–60. De Gruyter Oldenbourg.

[68] Cliquet, G., Perreira, M., Picarougne, F., Prié, Y., and Vigier, T. (2017). Towards hmd-based immersive analytics. In *Immersive analytics Workshop, IEEE VIS*.

[69] Cmentowski, S., Krekhov, A., and Krüger, J. (2019). Outstanding: a multi-perspective travel approach for virtual reality games. In *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*, CHI PLAY '19, pages 287–299, New York, NY, USA. ACM.

[70] Cognolato, M., Atzori, M., and Müller, H. (2018). Head-mounted eye gaze tracking devices: An overview of modern devices and recent advances. *Journal of Rehabilitation and Assistive Technologies Engineering*, 5:2055668318773991. PMID: 31191938.

[71] Cooper, I. S., Upton, A. R. M., and Amin, I. (1980). Reversibility of chronic neurologic deficits. some effects of electrical stimulation of the thalamus and internal capsule in man. 43:244–258.

[72] Cooper, N., Milella, F., Pinto, C., Cant, I., White, M., and Meyer, G. (2018). The effects of substitute multisensory feedback on task performance and the sense of presence in a virtual reality environment. *PloS one*, 13(2):e0191846.

[73] Cox, K. C., Eick, S. G., Wills, G. J., and Brachman, R. J. (1997). Visual data mining: Recognizing telephone calling fraud. *Data Mining and Knowledge Discovery*, 1(2):225–231.

[74] Craig, A. B., Sherman, W. R., and Will, J. D. (2009). *Developing virtual reality applications: Foundations of effective design*. Morgan Kaufmann.

[75] Cruz-Neira, C. (1995). *Virtual reality based on multiple projection screens: The CAVE and its applications to computational science and engineering*. Dissertation.

[76] Cruz-Neira, C., Sandin, D. J., and DeFanti, T. A. (1993). Surround-screen projection-based virtual reality. In Whitton, M. C., editor, *Proceedings of the 20th annual conference on Computer graphics and interactive techniques - SIGGRAPH '93*, pages 135–142. ACM Press.

[77] Cruz-Neira, C., Sandin, D. J., DeFanti, T. A., Kenyon, R. V., and Hart, J. C. (1992). The cave: audio visual experience automatic virtual environment. *Communications of the ACM*, 35(6):64–73.

[78] Csikszentmihalyi, M. and Csikzentmihaly, M. (1990). *Flow: The psychology of optimal experience*. Harper & Row New York.

[79] da Silva, N. M., Ahmadi, S.-A., Tafula, S. N., Cunha, J. P. S., Bötzel, K., Vollmar, C., and Rozanski, V. E. (2017). A diffusion-based connectivity map of the gpi for optimised stereotactic targeting in dbs. 144:83–91.

[80] Dangeti, S., Chen, Y. V., and Zheng, C. (2016). Comparing bare-hand-in-air gesture and object-in-hand tangible user interaction for navigation of 3d objects in modeling. In *Proceedings of the TEI '16: Tenth International Conference on Tangible, Embedded, and Embodied Interaction*, pages 417–421. Association for Computing Machinery.

[81] Das, K., Benzil, D. L., Rovit, R. L., Murali, R., and Couldwell, W. T. (1989). Irving s. cooper (1922-1985): a pioneer in functional neurosurgery. *Journal of neurosurgery*.

[82] Davis, S., Nesbitt, K., and Nalivaiko, E. (2014). A systematic review of cybersickness. IE2014, page 1–9, New York, NY, USA. Association for Computing Machinery.

[83] Davis, S., Nesbitt, K., Nalivaiko, E., et al. (2015). Comparing the onset of cybersickness using the oculus rift and two virtual roller coasters. In *Proceedings of the 11th Australasian conference on interactive entertainment (IE 2015)*, volume 27, page 30. Australian Computing Society Sydney, Australia.

[84] de Vasconcelos, G. N., Malard, M. L., van Stralen, M., Campomori, M., de Abreu, S. C., Lobosco, T., Gomes, I. F., Duarte, L., and Lima, C., editors (2019). *Do we still need CAVEs?*

[85] Decentraland (2022). https://decnteraland.com, Last Acess: 09.02.2022.

[86] Demiralp, C., Jackson, C. D., Karelitz, D. B., Zhang, S., and Laidlaw, D. H. (2006). Cave and fishtank virtual-reality displays: a qualitative and quantitative comparison. *IEEE transactions on visualization and computer graphics*, 12(3):323–330. Comparative Study Evaluation Study Journal Article Research Support, U.S. Gov't, Non-P.H.S.

[87] DiBenigno, M., Kosa, M., and Johnson-Glenberg, M. C. (2021). Flow immersive: A multiuser, multidimensional, multiplatform interactive covid-19 data visualization tool. *Frontiers in psychology*, 12.

[88] Diefenbach, G. J., Diefenbach, D., Baumeister, A., and West, M. (1999). Portrayal of lobotomy in the popular press: 1935-1960. *Journal of the History of the Neurosciences*, 8(1):60–69.

[89] Diehl, S. (2007). *Software Visualization: Visualizing the Structure, Behaviour, and Evolution of Software*. Springer Berlin Heidelberg.

[90] Dimara, E. and Perin, C. (2020). What is interaction for data visualization? *IEEE transactions on visualization and computer graphics*, 26(1):119–129. Journal Article.

[91] Djorgovski, S. G., Hut, P., Knop, R., Longo, G., McMillan, S., Vesperini, E., Donalek, C., Graham, M., Mahabal, A., Sauer, F., White, C., and Lopes, C. (2012). The mica experiment: Astrophysics in virtual worlds.

[92] Donalek, C., Djorgovski, S. G., Cioc, A., Wang, A., Zhang, J., Lawler, E., Yeh, S., Mahabal, A., Graham, M., Drake, A., Davidoff, S., Norris, J. S., and Longo, G. (2014). Immersive and collaborative data visualization using virtual reality platforms. In *2014 IEEE International Conference on Big Data (Big Data)*, pages 609–614. IEEE.

[93] Dorph, R., Cannady, M. A., and Schunn, C. D. (2016). How science learning activation enables success for youth in science learning experiences. *The Electronic Journal for Research in Science & Mathematics Education*, 20(8).

[94] Dourish, P. (2004). *Where the action is: the foundations of embodied interaction*. MIT press.

[95] Drucker, S. M., Fisher, D., Sadana, R., Herron, J., and schraefel, m. (2013). Touchviz. In Mackay, W. E., Brewster, S., and Bødker, S., editors, *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2301–2310. ACM.

[96] Duchowski, A. T. and Duchowski, A. T. (2017). *Eye tracking methodology: Theory and practice*. Springer.

[97] Dumbill, E. (2013). Making sense of big data. *Big data*, 1(1):1–2.

[98] Dywer, T., Marriott, K., Klein, K., Richie, N., Schreiber, F., Stuerzlinger, W., and Thomas, B. H. (2018). Immersive analytics: An introduction. In Marriott, K., Schreiber, F., Dwyer, T., Klein, K., Riche, N. H., Itoh, T., Stuerzlinger, W., and Thomas, B. H., editors, *Immersive Analytics*, pages 1–15. Springer International Publishing.

[99] Edwards, C. A., Kouzani, A., Lee, K. H., and Ross, E. K. (2017). Neurostimulation devices for the treatment of neurologic disorders. In *Mayo Clinic Proceedings*, volume 92, pages 1427–1444. Elsevier.

[100] El Beheiry, M., Doutreligne, S., Caporal, C., Ostertag, C., Dahan, M., and Masson, J.-B. (2019). Virtual reality: Beyond visualization. *Journal of Molecular Biology*, 431(7):1315–1321.

[101] eMarketer (2020). Us virtual and augmented reality users 2020. https://www.emarketer.com/content/us-virtual-and-augmented-reality-users-2020, Last Access: 09.02.2022.

[102] Farr, W. M., Hut, P., Ames, J., and Johnson, A. (2009). An experiment in using virtual worlds for scientific visualization of self-gravitating systems. 10 pages, 4 figures. As accepted by Journal of Virtual Worlds Research.

[103] Febretti, Alessandro and Nishimoto, Arthur and Thigpen, Terrance and Talandis, Jonas and Long, Lance and Pirtle, JD and Peterka, Tom and Verlo, Alan and Brown, Maxine and Plepys, Dana and others, editor (2013). *CAVE2: a hybrid reality environment for immersive simulation and information analysis*. International Society for Optics and Photonics.

[104] Férey, N., Gros, P. E., Hérisson, J., and Gherbi, R. (2005). Visual data mining of genomic databases by immersive graph-based exploration. In Wyvill, G., Arnold, D., and Billinghurst, M., editors, *Proceedings of the 3rd international conference on Computer graphics and interactive techniques in Australasia and South East Asia - GRAPHITE '05*, page 143. ACM Press.

[105] Fernandes, A. S. and Feiner, S. K. (2016). Combating vr sickness through subtle dynamic field-of-view modification. In *2016 IEEE Symposium on 3D User Interfaces (3DUI)*, pages 201–210.

[106] Flohr, T. G., McCollough, C. H., Bruder, H., Petersilka, M., Gruber, K., Süβ, C., Grasruck, M., Stierstorfer, K., Krauss, B., Raupach, R., et al. (2006). First performance evaluation of a dual-source ct (dsct) system. *European radiology*, 16(2):256–268.

[107] Fons, J., Monclús, E., Vázquez, P.-P., and Navazo, I. (2018). Rendering and interacting with volume models in immersive environments. In *Spanish Computer Graphics Conference (CEIG)*. The Eurographics Association.

[108] Franzini, A., Messina, G., Cordella, R., Marras, C., and Broggi, G. (2010). Deep brain stimulation of the posteromedial hypothalamus: indications, long-term results, and neurophysiological considerations. *Neurosurgical focus*, 29(2):E13.

[109] Friendly, M. (2006). A brief history of data visualization. In Chen, C., Härdle, W., and Unwin, A., editors, *Handbook of Computational Statistics: Data Visualization*, volume III. Springer-Verlag, Heidelberg. (In press).

[110] Fuchs, P., Moreau, G., and Guitton, P. (2011). *Virtual Reality: Concepts and Technologies*. Taylor & Francis Ltd.

[111] Galloway, R. and Maciunas, R. J. (1990). Stereotactic neurosurgery. *Critical reviews in biomedical engineering*, 18(3):181–205.

[112] Goldstein, E. B. (2015). *Wahrnehmungspsychologie : der Grundkurs*. Springer, Berlin.

[113] Gouraud, H. (1971). Computer display of curved surfaces. In *Outstanding Dissertations in the Computer Sciences*.

[114] Grainger, S., Mao, F., and Buytaert, W. (2016). Environmental data visualisation for non-scientific contexts: Literature review and design framework. *Environmental Modelling & Software*, 85:299–318.

[115] Grass, M., Manzke, R., Nielsen, T., Koken, P., Proksa, R., Natanzon, M., and Shechter, G. (2003). Helical cardiac cone beam reconstruction using retrospective ecg gating. *Physics in Medicine & Biology*, 48(18):3069.

[116] Groppa, S., Herzog, J., Falk, D., Riedel, C., Deuschl, G., and Volkmann, J. (2013). Physiological and anatomical decomposition of subthalamic neurostimulation effects in essential tremor. *Brain*, 137(1):109–121.

[117] Gross, R. E. and Lozano, A. M. (2000). Advances in neurostimulation for movement disorders. *Neurological Research*, 22(3):247–258.

[118] Groth, C., Tauscher, J.-P., Heesen, N., Castillo, S., and Magnor, M. (2021). Visual techniques to reduce cybersickness in virtual reality. In *2021 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, pages 486–487. IEEE.

[119] Haber, R. B. and McNabb, D. A. (1990). Visualization idioms: A conceptual model for scientific visualization systems. *Visualization in scientific computing*, 74:93.

[120] Hadwiger, M., Kniss, J. M., Rezk-salama, C., Weiskopf, D., and Engel, K. (2006). *Real-Time Volume Graphics*. A. K. Peters, Ltd., USA.

[121] Hall, L. (1964). Nuclear magnetic resonance. *Advances in carbohydrate chemistry*, 19:51–93.

[122] Hallett, M. (2007). Transcranial magnetic stimulation: a primer. *Neuron*, 55(2):187–199.

[123] Hamani, C., Schwalb, J. M., Rezai, A. R., Dostrovsky, J. O., Davis, K. D., and Lozano, A. M. (2006). Deep brain stimulation for chronic neuropathic pain: long-term outcome and the incidence of insertional effect. *Pain*, 125(1-2):188–196.

[124] Hänel, C., Weyers, B., Hentschel, B., and Kuhlen, T. W. (2014). Interactive volume rendering for immersive virtual environments. In *2014 IEEE VIS International Workshop on 3DVis (3DVis)*, pages 73–74.

[125] Havre, S., Hetzler, E., Whitney, P., and Nowell, L. (2002). Themeriver: visualizing thematic changes in large document collections. *IEEE transactions on visualization and computer graphics*, 8(1):9–20.

[126] HEALTH, R. G. (1977). Modulation of emotion with a brain pacemaker treatment for intractable psychiatric illness. 165:300–317.

[127] HEATH, R. G. (1962). Common characteristics of epilepsy and schizophrenia: Clinical observation and depth electrode studies. 118:1013–1026.

[128] HEATH, R. G. (1963). Electrical self-stimulation of the brain in man. 120:571–577.

[129] Heinrich, J. and Weiskopf, D. (2012). State of the art of parallel coordinates.

[130] Helbig, C., Bauer, H.-S., Rink, K., Wulfmeyer, V., Frank, M., and Kolditz, O. (2014). Concept and workflow for 3d visualization of atmospheric data in a virtual reality environment for analytical approaches. *Environmental Earth Sciences*, 72(10):3767–3780.

[131] Herman, I., Melancon, G., and Marshall, M. S. (2000). Graph visualization and navigation in information visualization: A survey. *IEEE transactions on visualization and computer graphics*, 6(1):24–43.

[132] Hernando, V., Pastor, J., Pedrosa, M., Peña, E., and Sola, R. (2008). Low-frequency bilateral hypothalamic stimulation for treatment of drug-resistant aggressiveness in a young man with mental retardation. *Stereotactic and functional neurosurgery*, 86(4):219–223.

[133] Herndon, K. P., van Dam, A., and Gleicher, M. (1994). The challenges of 3d interaction: a chi'94 workshop. *ACM SIGCHI Bulletin*, 26(4):36–43.

[134] Hinckley, K., Pausch, R., Goble, J. C., and Kassell, N. F. (1994). A survey of design issues in spatial input. In Szekely, P., editor, *Proceedings of the 7th annual ACM symposium on User interface software and technology - UIST '94*, pages 213–222. ACM Press.

[135] Hounsfield, G. N. (1973). Computerized transverse axial scanning (tomography): Part 1. description of system. *The British journal of radiology*, 46(552):1016–1022.

[136] Hrimech, H., Alem, L., and Merienne, F. (2011). How 3d interaction metaphors affect user experience in collaborative virtual environment. *Advances in Human-Computer Interaction*, 2011:1–11.

[137] HTC (2019). Vive discover virtual reality beyond imagination.

[138] Huang, B., Jiang, B., and Li, H. (2001). An integration of gis, virtual reality and the internet for visualization, analysis and exploration of spatial data. *International Journal of Geographical Information Science*, 15(5):439–456.

[139] Inselberg, A. (2009). Data mining and other applications. In Inselberg, A., editor, *Parallel Coordinates*, pages 379–427. Springer New York.

[140] Iskenderova, A., Weidner, F., and Broll, W. (2017). Drunk virtual reality gaming: Exploring the influence of alcohol on cybersickness. In *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*, CHI PLAY '17, pages 561–572, New York, NY, USA. ACM.

[141] Jansen, Y., Dragicevic, P., Isenberg, P., Alexander, J., Karnik, A., Kildal, J., Subramanian, S., and Hornbæk, K. (2015). Opportunities and challenges for data physicalization. In Begole, B., Kim, J., Inkpen, K., and Woo, W., editors, *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 3227–3236. ACM.

[142] Johnson-Glenberg, M. C. (2018). Immersive vr and education: Embodied design principles that include gesture and hand controls. *Frontiers in robotics and AI*, 5:81.

[143] Kalender, W., Seissler, W., and Vock, P. (1989). Single-breath-hold spiral volumetric ct by continuous patient translation and scanner rotation. In *Proceedings of the 75th anniversary scientific assembly and annual meeting Radiological Society of North America (Abstracts)*.

[144] Kalender, W. A. (1994). Spiral or helical ct: right or wrong? *Radiology*, 193(2):583–583.

[145] Kalender, W. A. (2006). X-ray computed tomography. *Physics in Medicine & Biology*, 51(13):R29.

[146] Kaptan, H., Ayaz, M., and Ekmekçi, H. (2014). Effect transformation of the micro electrode recording (mer) data to fast fourier transform (fft) for the main target nucleus determination for stn-dbs. *Acta informatica medica : AIM : journal of the Society for Medical Informatics of Bosnia & Herzegovina : casopis Drustva za medicinsku informatiku BiH*, 22(6):411–412.

[147] Keim, D., Andrienko, G., Fekete, J.-D., Görg, C., Kohlhammer, J., and Melançon, G. (2008a). Visual analytics: Definition, process, and challenges. In Kerren, A., Stasko, J. T., Fekete, J.-D., and North, C., editors, *Information Visualization*, volume 4950 of *Lecture Notes in Computer Science*, pages 154–175. Springer Berlin Heidelberg.

[148] Keim, D. A. (2000). Designing pixel-oriented visualization techniques: Theory and applications. *IEEE transactions on visualization and computer graphics*, 6(1):59–78.

[149] Keim, D. A. (2002). Information visualization and visual data mining. *IEEE transactions on visualization and computer graphics*, 8(1):1–8.

[150] Keim, D. A., Mansmann, F., Schneidewind, J., Thomas, J., and Ziegler, H. (2008b). Visual analytics: Scope and challenges. In Simoff, S. J., Böhlen, M. H., and Mazeika, A., editors, *Visual Data Mining: Theory, Techniques and Tools for Visual Analytics*, pages 76–90. Springer Berlin Heidelberg.

[151] Keim, D. A. and Ward, M. O. (2002). Visual data mining techniques. In Berthold, M., editor, *Intelligent Data Analysis: An Introduction*, pages 2–27. Springer.

[152] Kenngott, H. G., Pfeiffer, M., Preukschas, A. A., Bettscheider, L., Wise, P. A., Wagner, M., Speidel, S., Huber, M., Nickel, F., Mehrabi, A., and Müller-Stich, B. P. (2021). Imhotep: cross-professional evaluation of a three-dimensional virtual reality system for interactive surgical operation planning, tumor board discussion and immersive training for complex liver surgery in a head-mounted display. *Surgical Endoscopy*.

[153] Khadka, R., Money, J. H., and Banic, A. (2018). Evaluation of scientific workflow effectiveness for a distributed multi-user multi-platform support system for collaborative visualization. In *Proceedings of the Practice and Experience on Advanced Research Computing*, pages 1–8.

[154] Kilgard, M. J. and Akeley, K. (2008). Modern opengl: its design and evolution. In *ACM SIGGRAPH ASIA 2008 courses*, pages 1–31.

[155] Kim, G. and Biocca, F. (2018). Immersion in virtual reality can increase exercise motivation and physical performance. In Chen, J. Y. and Fragomeni, G., editors, *Virtual, Augmented and Mixed Reality: Applications in Health, Cultural Heritage, and Industry*, volume 10910 of *Lecture Notes in Computer Science*, pages 94–102. Springer International Publishing.

[156] Knierim, P., Kosch, T., Schwind, V., Funk, M., Kiss, F., Schneegass, S., and Henze, N. (2017). Tactile drones - providing immersive tactile feedback in virtual reality through quadcopters. In Mark, G., Fussell, S., Lampe, C., Schraefel M.,

Hourcade, J. P., Appert, C., and Wigdor, D., editors, *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, pages 433–436. ACM.

[157] Knight, C. and Munro, M. (1999). Comprehension with[in] virtual environment visualisations. In *Proceedings Seventh International Workshop on Program Comprehension*, pages 4–11. IEEE Comput. Soc.

[158] Korhonen, J. and You, J. (2012). Peak signal-to-noise ratio revisited: Is simple beautiful? In *2012 Fourth International Workshop on Quality of Multimedia Experience*, pages 37–38.

[159] Kratz, A., Hadwiger, M., Fuhrmann, A. L., Splechtna, R., and Bühler, K. (2006). Gpu-based high-quality volume rendering for virtual environments.

[160] Kruger, J. and Westermann, R. (2003). Acceleration techniques for gpu-based volume rendering. In *Proceedings of the 14th IEEE Visualization 2003 (VIS'03)*, VIS '03, pages 38–, Washington, DC, USA. IEEE Computer Society.

[161] Krüger, J. and Westermann, R. (2003). Acceleration techniques for gpu-based volume rendering. In *Proceedings of the 14th IEEE Visualization 2003 (VIS'03)*, VIS '03, page 38, USA. IEEE Computer Society.

[162] Larkin, J. H. and Simon, H. A. (1987). Why a diagram is (sometimes) worth ten thousand words. *Cognitive Science*, 11(1):65–100.

[163] LaViola, J. J. (2000a). A discussion of cybersickness in virtual environments. *ACM SIGCHI Bulletin*, 32(1):47–56.

[164] LaViola, J. J., Feliz, D. A., Keefe, D. F., and Zeleznik, R. C. (2001). Hands-free multi-scale navigation in virtual environments. In Hughes, J. F. and Séquin, C. H., editors, *Proceedings of the 2001 symposium on Interactive 3D graphics - SI3D '01*, pages 9–15. ACM Press.

[165] LaViola, Jr., J. J. (2000b). A discussion of cybersickness in virtual environments. *SIGCHI Bull.*, 32(1):47–56.

[166] LaViola Jr, J. J. (2008). Bringing vr and spatial 3d interaction to the masses through video games. *IEEE Computer Graphics and Applications*, 28(5):10–15.

[167] Lazarus III, P., Aubrey, J., and Heffron, R. (1976). *Futureworld*. American International Pictures.

[168] Le Bihan, D., Mangin, J.-F., Poupon, C., Clark, C. A., Pappata, S., Molko, N., and Chabriat, H. (2001). Diffusion tensor imaging: Concepts and applications. *Journal of Magnetic Resonance Imaging*, 13(4):534–546.

[169] Lee, B., Isenberg, P., Riche, N. H., and Carpendale, S. (2012). Beyond mouse and keyboard: Expanding design considerations for information visualization interactions. *IEEE transactions on visualization and computer graphics*, 18(12):2689–2698.

[170] Lee, E. A.-L. and Wong, K. W. (2014). Learning with desktop virtual reality: Low spatial ability learners are more positively affected. *Computers & Education*, 79:49–58.

[171] Lee, K. M. (2004). Presence, explicated. *Communication Theory*, 14(1):27–50.

[172] Leone, M., Franzini, A., and Bussone, G. (2001). Stereotactic stimulation of posterior hypothalamic gray matter in a patient with intractable cluster headache. *New England Journal of Medicine*, 345(19):1428–1429.

[173] Leroi-Gourhan, A. (1982). *The dawn of European art: an introduction to Palaeolithic cave painting*. CUP Archive.

[174] Levoy, M. (1989). *Display of Surfaces from Volume Data*. PhD thesis, University of North Carolina at Chapel Hill.

[175] Levoy, M. (1990). Efficient ray tracing of volume data. *ACM Trans. Graph.*, 9(3):245–261.

[176] Levoy, M. and Whitaker, R. (1990). Gaze-directed volume rendering. In *Proceedings of the 1990 Symposium on Interactive 3D Graphics*, I3D '90, page 217–223, New York, NY, USA. Association for Computing Machinery.

[177] Liagkou, V., Salmas, D., and Stylios, C. (2019). Realizing virtual reality learning environment for industry 4.0. *Procedia Cirp*, 79:712–717.

[178] Lin, A. Y.-M., Novo, A., Weber, P. P., Morelli, G., Goodman, D., and Schulze, J. P. (2011). A virtual excavation: Combining 3d immersive virtual reality and geophysical surveying. In Bebis, G., Boyle, R., Parvin, B., Koracin, D., Wang, S., Kyungnam, K., Benes, B., Moreland, K., Borst, C., DiVerdi, S., Yi-Jen, C., and Ming, J., editors, *Advances in Visual Computing*, volume 6939 of *Lecture Notes in Computer Science*, pages 229–238. Springer Berlin Heidelberg.

[179] Lin, C.-R. and Loftin, R. B. (1998). Application of virtual reality in the interpretation of geoscience data. In Shieh, J. M. and Yang, S.-N., editors, *Proceedings of the ACM symposium on Virtual reality software and technology 1998*, pages 187–194. ACM Press.

[180] Lin, J. J.-W., Abi-Rached, H., Kim, D.-H., Parker, D. E., and Furness, T. A. (2002a). A "natural" independent visual background reduced simulator sickness. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 46(26):2124–2128.

[181] Lin, J. J.-W., Duh, H. B.-L., Abi-Rached, H., Parker, D. E., and Furness, T. A. (2002b). Effects of field of view on presence, enjoyment, memory, and simulator sickness in a virtual environment. *Proceedings IEEE Virtual Reality 2002*, pages 164–171.

[182] Lin, Y.-X., Venkatakrishnan, R., Venkatakrishnan, R., Ebrahimi, E., Lin, W.-C., and Babu, S. V. (2020). How the presence and size of static peripheral blur affects cybersickness in virtual reality. *ACM Transactions on Applied Perception (TAP)*, 17(4):1–18.

[183] Linsen, L. and Behrendt, S. (2011). Linked treemap: a 3d treemap-nodelink layout for visualizing hierarchical structures. *Computational Statistics*, 26(4):679–697.

[184] Ljung, P., Krüger, J., Gröller, E., Hadwiger, M., Hansen, C. D., and Ynnerman, A. (2016). State of the Art in Transfer Functions for Direct Volume Rendering. *Computer Graphics Forum*.

[185] Long, L. K., Hui, L. C., Fook, G. Y., and Wan Zainon, W. M. N. (2017). A study on the effectiveness of tree-maps as tree visualization techniques. *Procedia Computer Science*, 124:108–115.

[186] Lu, A., Huang, J., Zhang, S., Wang, C., and Wang, W. (2016). Towards mobile immersive analysis: A study of applications. In *2016 Workshop on Immersive Analytics (IA)*, pages 25–30. IEEE.

[187] Lu, F., Yu, D., Liang, H.-N., Chen, W., Papangelis, K., and Ali, N. M. (2018). Evaluating engagement level and analytical support of interactive visualizations in virtual reality environments. In *2018 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 143–152. IEEE.

[188] Lurie, N. H. and Mason, C. H. (2007). Visual representation: Implications for decision making. *Journal of Marketing*, 71(1):160–177.

[189] Lyons, M. K. (2011). Deep brain stimulation: current and future clinical applications. In *Mayo Clinic Proceedings*, volume 86, pages 662–672. Elsevier.

[190] Mandal, S. (2013). Brief introduction of virtual reality & its challenges. *International Journal of Scientific & Engineering Research*, 4(4):304–309.

[191] Mania, K. and Chalmers, A. (2001). The effects of levels of immersion on memory and presence in virtual environments: a reality centered approach. *Cyberpsychology & behavior : the impact of the Internet, multimedia and virtual reality on behavior and society*, 4(2):247–264.

[192] Marieb, E. N. and Hoehn, K. (2007). *Human anatomy & physiology*. Pearson Benjamin Cummings, 7th ed. edition.

[193] Marriott, K., Chen, J., Hlawatsch, Marcel, Itoh, Takayuki, and Nacenta, Miguel A., Reina, Guido, Stuerzlinger, Wolfgang (2018). Time to reconsider the value of 3d for information visualisation. In Marriott, K., Schreiber, F., Dwyer, T., Klein, K., Riche, N. H., Itoh, T., Stuerzlinger, W., and Thomas, B. H., editors, *Immersive Analytics*, pages 25–55. Springer International Publishing.

[194] Marsden, C. D. and Parkes, J. D. (1976). "on-off" effects in patients with parkinson's disease on chronic levodopa therapy. 307:292–296.

[195] Mary K. Kaiser (1991). Knowing. In Ellis, S. R., editor, *Pictorial Communication in Virtual and Real Environments*, pages 43–46. Taylor & Francis, Inc.

[196] May, A., Leone, M., Boecker, H., Sprenger, T., Juergens, T., Bussone, G., and Tolle, T. R. (2006). Hypothalamic deep brain stimulation in positron emission tomography. *Journal of Neuroscience*, 26(13):3589–3593.

[197] Mayberg, H. S., Lozano, A. M., Voon, V., McNeely, H. E., Seminowicz, D., Hamani, C., Schwalb, J. M., and Kennedy, S. H. (2005). Deep brain stimulation for treatment-resistant depression. *Neuron*, 45(5):651–660.

[198] Mazza, R. (2009). *Introduction to Information Visualization*. Springer London.

[199] McClanahan, C. (2010). History and evolution of gpu architecture. *A Survey Paper*, 9:1–7.

[200] Melzack, R. (2018). Spinal cord stimulation. *Functional Neurosurgery and Neuromodulation*, 4:43.

[201] Meng, X., Du, R., and Varshney, A. (2020). Eye-dominance-guided foveated rendering. *IEEE Transactions on Visualization and Computer Graphics*, 26(5):1972–1980.

[202] Mestre, D. R. (2017). Cave versus head-mounted displays: Ongoing thoughts. *Electronic Imaging*, 2017(3):31–35.

[203] Meta (2022). https://about.facebook.com/meta/, Last Acess: 09.02.2022.

[204] Millais, P., Jones, S. L., and Kelly, R. (04202018). Exploring data in virtual reality. In Mandryk, R., Hancock, M., Perry, M., and Cox, A., editors, *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*, pages 1–6. ACM.

[205] Moran, A., Gadepally, V., Hubbell, M., and Kepner, J. (2015). Improving big data visual analytics with interactive virtual reality. pages 1–6.

[206] Mousavi, M., Jen, Y. H., and Musa, S. N. B. (2013). A review on cybersickness and usability in virtual environments. In *Current Trends in Ergonomics*, volume 10 of *Advanced Engineering Forum*, pages 34–39. Trans Tech Publications Ltd.

[207] Mulder, J. D., van Liere, R., and van Wijk, J. J. (1998). Computational steering in the cave. *Future Generation Computer Systems*, 14(3-4):199–207.

[208] Munzner, T. (2014). *Visualization analysis and design*. CRC press.

[209] Nagel, H. R., Granum, E., and Bovbjerg, S., Vittrup, M. (2008). Immersive visual data mining: The 3dvdm approach. In Simoff, S. J., Böhlen, M. H., and Mazeika, A., editors, *Visual Data Mining: Theory, Techniques and Tools for Visual Analytics*, pages 281–311. Springer Berlin Heidelberg.

[210] Nakasone, A., Prendinger, H., Holland, S., Miura, K., Hut, P., and Makino, J. (2009). Astrosim: collaborative visualization of an astrophysics simulation in second life. *IEEE computer graphics and applications*, 29(5):69–81.

[211] Natterer, F. (2001). *The mathematics of computerized tomography*. SIAM.

[212] Nelson, L., Cook, D., and Cruz-Neira, C. (1999). Xgobi vs the c2: Results of an experiment comparing data visualization in a 3-d immersive virtual reality environment with a 2-d workstation display. *Computational Statistics*, 14(1):39–51.

[213] Nie, G.-Y., Duh, H. B.-L., Liu, Y., and Wang, Y. (2019). Analysis on mitigation of visually induced motion sickness by applying dynamical blurring on a user's retina. *IEEE transactions on visualization and computer graphics*, 26(8):2535–2545.

[214] Norman, D. A. (2013). *The design of everyday things*. Basic Books, revised and expanded edition edition.

[215] Norman, D. A. and Draper, S. W. (1986). *User Centered System Design; New Perspectives on Human-Computer Interaction*. L. Erlbaum Associates Inc., USA.

[216] Novotny, J., Tveite, J., Turner, M. L., Gatesy, S., Drury, F., Falkingham, P., and Laidlaw, D. H. (2019). Developing virtual reality visualizations for unsteady flow analysis of dinosaur track formation using scientific sketching. *IEEE Transactions on Visualization and Computer Graphics*, 25(5):2145–2154.

[217] Oculus (2019). Oculus rift: Vr-headset.

[218] Olanow, C. W. (2008). Levodopa/dopamine replacement strategies in parkinson's disease-future directions. 23:S613–S622.

[219] O'Leary, P., Jhaveri, S., Chaudhary, A., Sherman, W., Martin, K., Lonie, D., Whiting, E., Money, J., and McKenzie, S. (2017). Enhancements to vtk enabling scientific visualization in immersive environments. In *2017 IEEE Virtual Reality (VR)*, pages 186–194.

[220] Oliver (2018). The display resolution of head-mounted displays, revisited.

[221] Olshannikova, E., Ometov, A., Koucheryavy, Y., and Olsson, T. (2015). Visualizing big data with augmented and virtual reality: challenges and research agenda. *Journal of Big Data*, 2(1). PII: 31.

[222] Parker, G., Franck, G., and Ware, C. (1998). Visualization of large nested graphs in 3d: Navigation and interaction. *Journal of Visual Languages & Computing*, 9(3):299–317.

[223] Pausch, R., Proffitt, D., and Williams, G. (1997). Quantifying immersion in virtual reality. In Owen, G. S., Whitted, T., and Mones-Hattal, B., editors, *Proceedings of the 24th annual conference on Computer graphics and interactive techniques - SIGGRAPH '97*, pages 13–18. ACM Press.

[224] Pederzoli, M., Girotti, F., Scigliano, G., Aiello, G., Carella, F., and Caraceni, T. (1983). L-dopa long-term treatment in parkinson's disease: Age-related side effects. 33:1518–1518.

[225] Pedroarena-Leal, N. and Ruge, D. (2017). Toward a symptom-guided neurostimulation for gilles de la tourette syndrome. *Frontiers in Psychiatry*, 8:29.

[226] Perlmutter, J. S. and Mink, J. W. (2006a). Deep brain stimulation. *Annual review of neuroscience*, 29:229.

[227] Perlmutter, J. S. and Mink, J. W. (2006b). Deep brain stimulation. *Annual Review of Neuroscience*, 29(1):229–257. PMID: 16776585.

[228] Phong, B. T. (1975). Illumination for computer generated pictures. *Commun. ACM*, 18(6):311–317.

[229] Pouke, M., Tiiro, A., LaValle, S. M., and Ojala, T. (2018). Effects of visual realism and moving detail on cybersickness. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pages 665–666. IEEE.

[230] Pouratian, N. and Sheth, S. A. (2020). *Stereotactic and Functional Neurosurgery: Principles and Applications*. Springer Nature.

[231] Prothero, J. D. (1998). *The role of rest frames in vection, presence and motion sickness*. University of Washington.

[232] Qi, J. and Leahy, R. M. (2006). Iterative reconstruction techniques in emission computed tomography. *Physics in Medicine & Biology*, 51(15):R541.

[233] Raja, D., Bowman, D., Lucas, J., North, C., and Tech, V. (2004). Exploring the benefits of immersion in abstract information visualization. *8th Int'l Immersive Projection Technology Workshop (IPT '04)*.

[234] Razzaque, S. (2005). *Redirected walking*. The University of North Carolina at Chapel Hill.

[235] Reason, J. and Brand, J. J. (1975). *Motion sickness / J. T. Reason, J. J. Brand*. Academic Press London ; New York.

[236] Rebenitsch, L. and Owen, C. (2016). Review on cybersickness in applications and visual displays. *Virtual Reality*, 20.

[237] Reda, K., Febretti, A., Knoll, A., Aurisano, J., Leigh, J., Johnson, A., Papka, M. E., and Hereld, M. (2013). Visualizing large, heterogeneous data in hybrid-reality environments. *IEEE computer graphics and applications*, 33(4):38–48. Journal Article Research Support, U.S. Gov't, Non-P.H.S.

[238] Reifinger, S., Laquai, F., and Rigoll, G. (2008). Translation and rotation of virtual objects in augmented reality: A comparison of interaction devices. In *2008 IEEE International Conference on Systems, Man and Cybernetics*, pages 2448–2453.

[239] Rhyne, T.-M., Tory, M., Munzner, T., Ward, M., Johnson, C., and Laidlaw, D. H. (2003). Information and scientific visualization: Separate but equal or happy together at last. In *Visualization Conference, IEEE*, pages 115–115. IEEE Computer Society.

[240] Ricciardi, G., Tommasi, G., Nicolato, A., Foroni, R., Bertolasi, L., Beltramello, A., Moretto, G., Tinazzi, M., Gerosa, M., and Longhi, M. (2015). The role of 3t magnetic resonance imaging for targeting the human subthalamic nucleus in deep brain stimulation for parkinson disease. 76:181–189.

[241] Riccio, G. E. and Stoffregen, T. A. (1991). An ecological theory of motion sickness and postural instability. *Ecological Psychology*, 3(3):195–240.

[242] Rinck, P. A. (2008). A short history of magnetic resonance imaging. *Spectroscopy Europe*, 20(1):7–10.

[243] Robertson, G., Czerwinski, M., and van Dantzich, M. (1997). Immersion in desktop virtual reality. In Robertson, G. and Schmandt, C., editors, *Proceedings of the 10th annual ACM symposium on User interface software and technology - UIST '97*, pages 11–19. ACM Press.

[244] Robertson, G. G., Mackinlay, J. D., and Card, S. K. (1991). Cone trees. In Robertson, S. P., Olson, G. M., and Olson, J. S., editors, *Proceedings of the SIGCHI conference on Human factors in computing systems Reaching through technology - CHI '91*, pages 189–194. ACM Press.

[245] Rodrigues, P. P. and Gama, J. (2010). A simple dense pixel visualization for mobile sensor data mining. In Gaber, M. M., Vatsavai, R. R., Omitaomu, O. A., Gama, J., Chawla, N. V., and Ganguly, A. R., editors, *Knowledge Discovery from Sensor Data*, pages 175–189. Springer Berlin Heidelberg.

[246] Rogers, D. F. (2001). *An introduction to NURBS: with historical perspective*. Morgan Kaufmann.

[247] Rosenbaum, R., Bottleson, J., Liu, Z., and Hamann, B. (2011). Involve me and i will understand!–abstract data visualization in immersive environments. In Bebis, G., Boyle, R., Parvin, B., Koracin, D., Wang, S., Kyungnam, K., Benes, B., Moreland, K., Borst, C., DiVerdi, S., Yi-Jen, C., and Ming, J., editors, *Advances in Visual Computing*, volume 6938 of *Lecture Notes in Computer Science*, pages 530–540. Springer Berlin Heidelberg.

[248] Salomon, D. (2011). *The computer graphics manual*. Springer Science & Business Media.

[249] Santos, S., Dias Paulo, B., Angela, P., Jan-Willem, B., Carlos, F., Samuel, S., and Joaquim, M. (2008). Head-mounted display versus desktop for 3d navigation in virtual reality: a user study. *Multimedia Tools and Applications*, 41(1):161.

[250] Savikhin, A., Maciejewski, R., and Ebert, D. S. (2008). Applied visual analytics for economic decision-making. In *2008 IEEE Symposium on Visual Analytics Science and Technology*, pages 107–114. IEEE.

[251] Scholl, I., Suder, S., and Schiffer, S. (2018). *Direct Volume Rendering in Virtual Reality*, pages 297–302.

[252] Schuepbach, W., Rau, J., Knudsen, K., Volkmann, J., Krack, P., Timmermann, L., Hälbig, T., Hesekamp, H., Navarro, S., Meier, N., Falk, D., Mehdorn, M., Paschen, S., Maarouf, M., Barbe, M., Fink, G., Kupsch, A., Gruber, D., Schneider, G.-H., Seigneuret, E., Kistner, A., Chaynes, P., Ory-Magne, F., Brefel Courbon, C., Vesper, J., Schnitzler, A., Wojtecki, L., Houeto, J.-L., Bataille, B., Maltête, D., Damier, P., Raoul, S., Sixel-Doering, F., Hellwig, D., Gharabaghi, A., Krüger, R., Pinsker, M., Amtage, F., Régis, J.-M., Witjas, T., Thobois, S., Mertens, P., Kloss,

M., Hartmann, A., Oertel, W., Post, B., Speelman, H., Agid, Y., Schade-Brittinger, C., and Deuschl, G. (2013). Neurostimulation for parkinson's disease with early motor complications. *New England Journal of Medicine*, 368(7):610–622. PMID: 23406026.

[253] Schuurman, P. R., Bosch, D. A., Bossuyt, P. M., Bonsel, G. J., van Someren, E. J., de Bie, R. M., Merkus, M. P., and Speelman, J. D. (2000). A comparison of continuous thalamic stimulation and thalamotomy for suppression of severe tremor. *New England Journal of Medicine*, 342(7):461–468. PMID: 10675426.

[254] Self, J. Z., House, L., Leman, S., and North, C. (2015). Andromeda: Observation-level and parametric interaction for exploratory data analysis.

[255] Seth, A., Vance, J. M., and Oliver, J. H. (2011). Virtual reality for assembly methods prototyping: a review. *Virtual Reality*, 15(1):5–20.

[256] Shapurian, T., Damoulis, P. D., Reiser, G. M., Griffin, T. J., and Rand, W. M. (2006). Quantitative evaluation of bone density using the hounsfield index. *International Journal of Oral & Maxillofacial Implants*, 21(2).

[257] Sharma, S., Bodempudi, S. T., and Reehl, A. (2021). Real-time data analytics of covid pandemic using virtual reality. In Chen, J. Y. C. and Fragomeni, G., editors, *Virtual, Augmented and Mixed Reality*, volume 12770 of *Lecture Notes in Computer Science*, pages 106–116. Springer International Publishing.

[258] Sharples, S., Cobb, S., Moody, A., and Wilson, J. (2008). Virtual reality induced symptoms and effects (vrise): Comparison of head mounted display (hmd), desktop and projection display systems. *Displays*, 29:58 – 69.

[259] Shen, R., Boulanger, P., and Noga, M. L. (2008). Medvis: A real-time immersive visualization environment for the exploration of medical volumetric data. *2008 Fifth International Conference BioMedical Visualization: Information Visualization in Medical and Biomedical Informatics*, pages 63–68.

[260] Sherman, W. R. and Craig, A. B. (2018). *Understanding Virtual Reality*. Elsevier Science.

[261] Shneiderman, B. (1996). The eyes have it: a task by data type taxonomy for information visualizations. In *Proceedings 1996 IEEE Symposium on Visual Languages*, pages 336–343. IEEE Comput. Soc. Press.

[262] Simoff, S. J., Böhlen, M. H., and Mazeika, A. (2008). Visual data mining: An introduction and overview. In Simoff, S. J., Böhlen, M. H., and Mazeika, A., editors, *Visual Data Mining*, Lecture Notes in Computer Science, pages 1–12. Springer Berlin Heidelberg.

[263] Simpson, R. M., LaViola, J. J., Laidlaw, D. H., Forsberg, A. S., and van Dam, A. (2000). Immersive vr for scientific visualization: a progress report. *IEEE computer graphics and applications*, 20(6):26–52.

[264] Singer, M. J. and Witmer, B. G. (1999). On selecting the right yardstick. *Presence: Teleoperators and Virtual Environments*, 8(5):566–573.

[265] Slater, M. (2003). A note on presence terminology. *Presence connect*, 3(3):1–5.

[266] Slater, M. (2009). Place illusion and plausibility can lead to realistic behaviour in immersive virtual environments. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 364(1535):3549–3557.

[267] Slater, M. (2018). Immersion and the illusion of presence in virtual reality. *British journal of psychology*, 109(3):431–433.

[268] Slater, M., Linakis, V., Usoh, M., and Kooper, R. (1996). Immersion, presence and performance in virtual environments. In Green, M., Fairchild, K., and Zyda, M., editors, *Proceedings of the ACM Symposium on Virtual Reality Software and Technology - VRST '96*, pages 163–172. ACM Press.

[269] Slater, M. and Wilbur, S. (1997). A framework for immersive virtual environments (five): Speculations on the role of presence in virtual environments. *Presence: Teleoperators and Virtual Environments*, 6(6):603–616.

[270] Smith, R. B. (1987). Experiences with the alternate reality kit: an example of the tension between literalism and magic. In Carroll, J. M. and Tanner, P. P., editors, *Proceedings of the SIGCHI/GI conference on Human factors in computing systems and graphics interface - CHI '87*, pages 61–67. ACM Press.

[271] Stanney, K. M. and Kennedy, R. S. (1998). Aftereffects from virtual environment exposure: How long do they last? *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 42(21):1476–1480.

[272] Stanney, K. M., Kennedy, R. S., and Drexler, J. M. (1997). Cybersickness is not simulator sickness. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 41(2):1138–1142.

[273] Stark, H., Woods, J., Paul, I., and Hingorani, R. (1981). Direct fourier reconstruction in computer tomography. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 29(2):237–245.

[274] Stauffert, J.-P., Niebling, F., and Latoschik, M. (2020). Latency and cybersickness: Impact, causes, and measures. a review. *Frontiers in Virtual Reality*, 1.

[275] Stegmaier, S., Strengert, M., Klein, T., and Ertl, T. (2005). A simple and flexible volume rendering framework for graphics-hardware-based raycasting. In *Fourth International Workshop on Volume Graphics, 2005.*, pages 187–241.

[276] Steigerwald, F., Müller, L., Johannes, S., Matthies, C., and Volkmann, J. (2016). Directional deep brain stimulation of the subthalamic nucleus: A pilot study using a novel neurostimulation device. *Movement Disorders*, 31(8):1240–1243.

[277] Steinicke, F., Bruder, G., Jerald, J., Frenz, H., and Lappe, M. (2008). Analyses of human sensitivity to redirected walking. In *Proceedings of the 2008 ACM symposium on Virtual reality software and technology*, pages 149–156.

[278] Stewart, E. E. M., Valsecchi, M., and Schütz, A. C. (2020). A review of interactions between peripheral and foveal vision. *Journal of Vision*, 20(12):2–2.

[279] Stoffregen, T. A. and Riccio, G. E. (1991). An ecological critique of the sensory conflict theory of motion sickness. *Ecological Psychology*, 3(3):159–194.

[280] Suda, M. and Oka, M. (2021). Evaluation of the effect of mimicry on facial expression in avatar-mediated communication. In *The 2021 Conference on Artificial Life*. MIT press.

[281] Sutherland, I. E. (1968). A head-mounted three dimensional display. In *Proceedings of the December 9-11, 1968, fall joint computer conference, part I on - AFIPS '68 (Fall, part I)*, page 757. ACM Press.

[282] Symanzik, J., Cook, D., Kohlmeyer, B. T., and Cruz-Neira, C. (1996). Dynamic statistical graphics in the cave virtual reality environment. In *Proc. Dynamic Statistical Graphics Workshop*, pages 41–47.

[283] Teras, M. and Raghunathan, S. (2015). Big data visualisation in immersive virtual reality environments: Embodied phenomenological perspectives to interaction. *ICTACT Journal on Soft Computing*, 05(04):1009–1015.

[284] Teyseyre, A. R. and Campo, M. R. (2008). An overview of 3d software visualization. *IEEE transactions on visualization and computer graphics*, 15(1):87–105.

[285] Thomas, B. H., Welch, G. F., Dragicevic, P., Elmqvist, N., Irani, P., Jansen, Y., Schmalstieg, D., Tabard, A., ElSayed, N. A. M., Smith, R. T., and Willett, W. (2018). Situated analytics. In Marriott, K., Schreiber, F., Dwyer, T., Klein, K., Riche, N. H., Itoh, T., Stuerzlinger, W., and Thomas, B. H., editors, *Immersive Analytics*, volume 11190, pages 185–220. Springer International Publishing.

[286] Thomas, J. J. and Cook, K. A. (2006). A visual analytics agenda. *IEEE computer graphics and applications*, 26(1):1–190.

[287] Tiiro, A. (2018). Effect of visual realism on cybersickness in virtual reality. *University of Oulu*, 350.

[288] Torrence, A. (2006). Martin newell's original teapot. In *ACM SIGGRAPH 2006 Teapot Copyright restrictions prevent ACM from providing the full text for the Teapot exhibits*, pages 29–es.

[289] Treisman, M. (1977). Motion sickness: An evolutionary hypothesis. *Science*, 197(4302):493–495.

[290] Tsubokawa, T., Katayama, Y., Yamamoto, T., Hirayama, T., and Koyama, S. (1991). Chronic motor cortex stimulation for the treatment of central pain. In *Advances in Stereotactic and Functional Neurosurgery 9*, pages 137–139. Springer.

[291] Tsubokawa, T., Katayama, Y., Yamamoto, T., Hirayama, T., and Koyama, S. (1993). Chronic motor cortex stimulation in patients with thalamic pain. *Journal of neurosurgery*, 78(3):393–401.

[292] Tufte, E. R. (1985). The visual display of quantitative information. *The Journal for Healthcare Quality (JHQ)*, 7(3):15.

[293] Turnbull, I. M., Shulman, R., and Woodhurst, W. B. (1980). Thalamic stimulation for neuropathic pain. *Journal of neurosurgery*, 52(4):486–493.

[294] Tye, S. J., Frye, M. A., and Lee, K. H. (2009). Disrupting disordered neurocircuitry: Treating refractory psychiatric illness with neuromodulation. 84:522–532.

[295] Usher, W., Klacansky, P., Federer, F., Bremer, P., Knoll, A., Yarch, J., Angelucci, A., and Pascucci, V. (2018). A virtual reality visualization tool for neuron tracing. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):994–1003.

[296] Valve (2019). Steam-hard- & softwaresurvey.

[297] van Dam, A., Laidlaw, D. H., and Simpson, R. M. (2002). Experiments in immersive virtual reality for scientific visualization. *Computers & Graphics*, 26(4):535–555.

[298] van Teylingen, R., Ribarsky, W., and van der Mast, C. (1997). Virtual data visualizer. *IEEE transactions on visualization and computer graphics*, 3(1):65–74.

[299] Ventura, S., Brivio, E., Riva, G., and Baños, R. M. (2019). Immersive versus non-immersive experience: Exploring the feasibility of memory assessment through 360° technology. *Frontiers in psychology*, 10:2509.

[300] Vidailhet, M., Vercueil, L., Houeto, J.-L., Krystkowiak, P., Benabid, A.-L., Cornu, P., Lagrange, C., Tézenas du Montcel, S., Dormont, D., Grand, S., Blond, S., Detante, O., Pillon, B., Ardouin, C., Agid, Y., Destée, A., and Pollak, P. (2005). Bilateral deep-brain stimulation of the globus pallidus in primary generalized dystonia. *New England Journal of Medicine*, 352(5):459–467. PMID: 15689584.

[301] Vincur, J., Navrat, P., and Polasek, I. (2017). Vr city: Software analysis in virtual reality environment. In *2017 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C)*, pages 509–516. IEEE.

[302] Volkmann, J. (2004). Deep brain stimulation for the treatment of parkinson's disease. *Journal of clinical neurophysiology*, 21(1):6–17.

[303] Vorwerk, J., Janson, A., Schiewe, A., Butson, C. R., and Krüger, J. (2016). Mobile computational steering for interactive prediction and visualization of deep brain stimulation therapy. In *Proceedings of the Supercomputing Workshop on Medical Image Analysis and Visaulization*.

[304] Walton, L., Hampshire, A., Forster, D. M., and Kemeny, A. A. (1996). A phantom study to assess the accuracy of stereotactic localization, using t1-weighted magnetic resonance imaging with the leksell stereotactic system. *Neurosurgery*, 38(1):170–178.

[305] Wang, L., Wang, G., and Alexander, C. A. (2015). Big data and visualization: Methods, challenges and technology progress. *Digital Technologies*, 1(1):33–38.

[306] Ware, C. (2004). *Information Visualization: Perception for Design: Second Edition*. Elsevier Science, fourth edition edition.

[307] Ware, C. and Mitchell, P. (2008). Visualizing graphs in three dimensions. *ACM Transactions on Applied Perception*, 5(1):1–15.

[308] Waschk, A., , and Krüger, J. (2016a). Accuracy improvement of dbs electrode placement by visualization of sensor-fused data. In *Proceedings of the GPU Technology Conference 2016*.

[309] Waschk, A. and Krueger, J. (2022). Embracing Raycasting for Virtual Reality. *Computer Science Research Notes*, 3201:128–134.

[310] Waschk, A. and Krüger, J. (2020). Favr - accelerating direct volume rendering for virtual reality systems. In *2020 IEEE Visualization Conference (VIS)*, pages 106–110.

[311] Waschk, A. and Krüger, J. (2022). Favr - accelerating direct volume rendering for virtual reality systems. In *WSCG 2022: full papers proceedings: 30. International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision*, pages 128–134.

[312] Waschk, A., Parpaley, Y., and Krüger, J. (2016b). An integrated system for analysis and visualization of neurostimulation procedures. In *Proceedings of the Supercomputing Workshop on Medical Image Analysis and Visaulization*.

[313] Waschk, A., Parpaley, Y., and Krüger, J. (2021). A multi-modular system for the visualization and classification of mer data during neurostimulation procedures. In *2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pages 6281–6284.

[314] Wegman, E. J. (1990). Hyperdimensional data analysis using parallel coordinates. *Journal of the American Statistical Association*, 85(411):664.

[315] Wegman, E. J. and Symanzik, J. (2002). Immersive projection technology for visual data mining. *Journal of Computational and Graphical Statistics*, 11(1):163–188.

[316] Wehden, L.-O., Reer, F., Janzik, R., Tang, W. Y., and Quandt, T. (2021). The slippery path to total presence: How omnidirectional virtual reality treadmills influence the gaming experience. *Media and Communication*, 9(1):5–16.

[317] Weishaupt, D., D., K. V., and Marincek, B. (2014). *Wie funktioniert MRI? Eine einführung in Physik und funktionsweise der Magnetresonanzbildgebunn*. Springer Medizin.

[318] Wienrich, C., Weidner, C. K., Schatto, C., Obremski, D., and Israel, J. H. (2018). A virtual nose as a rest-frame-the impact on simulator sickness and game experience. In *2018 10th international conference on virtual worlds and games for serious applications (VS-Games)*, pages 1–8. IEEE.

[319] Wijayasekara, D., Linda, O., and Manic, M. (2011). Cave-som: Immersive visual data mining using 3d self-organizing maps. In *The 2011 International Joint Conference 2011*, pages 2471–2478.

[320] Williams, N. R. and Okun, M. S. (2013). Deep brain stimulation (dbs) at the interface of neurology and psychiatry. 123:4546–4556.

[321] Witmer, B. G. and Singer, M. J. (1998). Measuring presence in virtual environments: A presence questionnaire. *Presence: Teleoperators and Virtual Environments*, 7(3):225–240.

[322] Woods, A. J., Holliman, N. S., Favalora, G. E., Kawai, T., and Sommer, B. (2018). Stereoscopic displays and applications. *Electronic Imaging*, 2018(4):476.

[323] Wright, H. (2007). *Introduction to scientific visualization*. Springer Science & Business Media.

[324] Wu, F., Bailey, G. S., Stoffregen, T., and Suma Rosenberg, E. (2021). Don't block the ground: Reducing discomfort in virtual reality with an asymmetric field-of-view restrictor. In *Symposium on Spatial User Interaction*, pages 1–10.

[325] Xu, M. and Wang, L. V. (2005). Universal back-projection algorithm for photoacoustic computed tomography. *Physical Review E*, 71(1):016706.

[326] Yildirim, C., Carroll, M., Hufnal, D., Johnson, T., and Pericles, S. (2018). Video game user experience: to vr, or not to vr? In *2018 IEEE Games, Entertainment, Media Conference (GEM)*, pages 1–9. IEEE.

[327] Young, S. W. (1987). *Magnetic resonance imaging: Basic principles*. Raven Press, United States.

[328] Zhu, Y. (2007). Measuring effective data visualization. In Bebis, G., Boyle, R., Parvin, B., Koracin, D., Paragios, N., Tanveer, S.-M., Ju, T., Liu, Z., Coquillart, S., Cruz-Neira, C., Müller, T., and Malzbender, T., editors, *Advances in Visual Computing*, pages 652–661. Springer Berlin Heidelberg.

DOI: 10.17185/duepublico/78357
URN: urn:nbn:de:hbz:465-20230428-142630-6